

UNIVERSIDAD CATOLICA DE SANTA MARIA

FACULTAD DE CIENCIAS FISICAS Y FORMALES

**PROGRAMA PROFESIONAL DE INGENIERÍA MECÁNICA,
MECÁNICA- ELÉCTRICA Y MECATRONICA**



**"DISEÑO E IMPLEMENTACION DE UN CONTROLADOR PID
DE TEMPERATURA CONTROLADO A DISTANCIA PARA UN
PROTOTIPO DE INCUBADORA"**

TESIS PARA OBTENCIÓN DE TITULO DE:

INGENIERO MECATRÓNICO

PRESENTADO POR:

BACH. MIGUEL ÁNGEL RAMÍREZ LAZO

AREQUIPA - PERÚ

2013



“Al vencer sin obstáculos se triunfa sin gloria”

RESUMEN

El control automático es de vital importancia en el mundo de la ingeniería, además de resultar imprescindible en sistemas robóticos o procesos de manufactura moderna entre otras aplicaciones, se ha vuelto esencial en operaciones industriales, uno de ellos es el control temperatura.

Hay lugares donde es necesario mantener una temperatura constante o en donde es necesario el poder variar la temperatura de cierto recinto hasta un punto determinado. Esta temperatura a su vez debe estar siendo supervisada continuamente.

La presente tesis está estructurada en líneas generales de la siguiente manera:

Capítulo I.- Se determina el ¿Qué?, ¿Dónde?, ¿Cuándo? ¿Por qué? y ¿Cómo?, también se determinan los objetivos específicos, su justificación los alcances y limitaciones del proyecto que se ha realizado.

Capítulo II.- Contiene todo el material teórico que se puede incluir para la realización de esta investigación y el desarrollo del sistema.

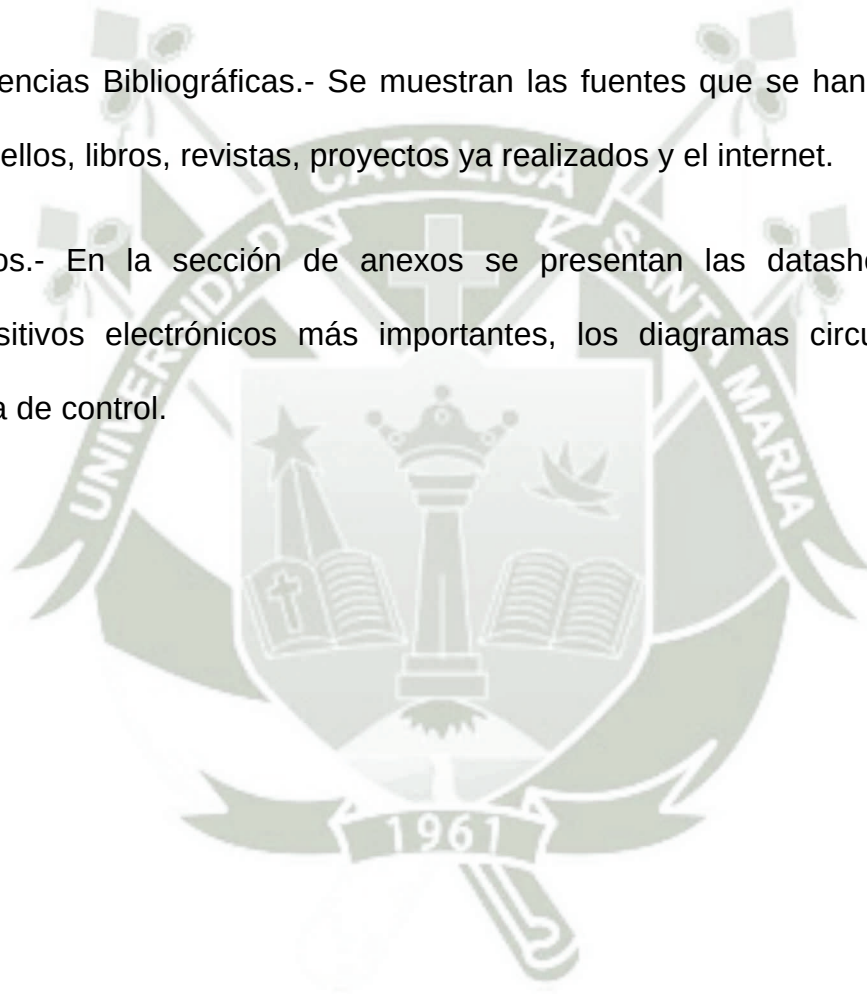
Capítulo III.- Trata sobre la construcción, el diseño y la simulación del controlador, así como también de la etapa de comunicación remota y el manejo por computador de la temperatura.

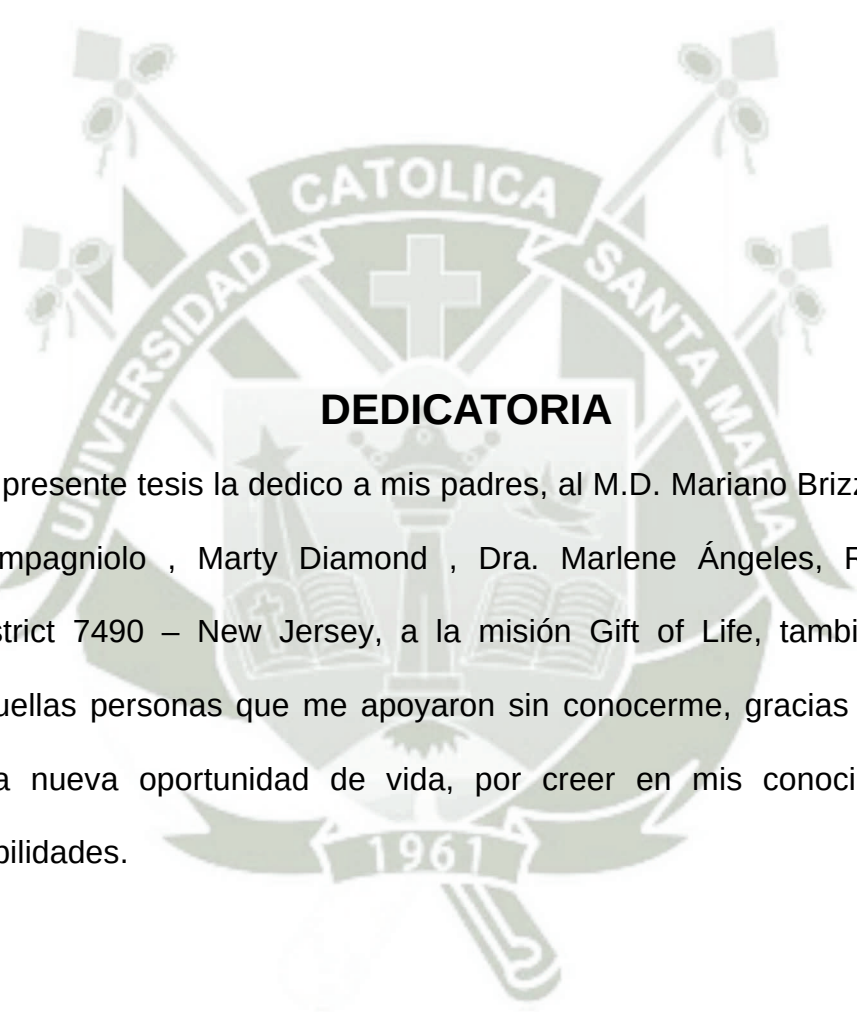
Capítulo IV.- Aquí se muestra el análisis económico del proyecto, se incluyen todos los costos, costos de componentes, implementación, también incluye ingeniería y costos adicionales.

Conclusiones y Recomendaciones.- Aquí se da las conclusiones a la que se llegó el proyecto como también opiniones personales o aclaraciones para la mejora del prototipo si se usara en algún futuro.

Referencias Bibliográficas.- Se muestran las fuentes que se han consultado, entre ellos, libros, revistas, proyectos ya realizados y el internet.

Anexos.- En la sección de anexos se presentan las datasheets de los dispositivos electrónicos más importantes, los diagramas circuitales, y la tarjeta de control.





DEDICATORIA

La presente tesis la dedico a mis padres, al M.D. Mariano Brizzio , Glenda Campagnuolo , Marty Diamond , Dra. Marlene Ángeles, Rotary Club District 7490 – New Jersey, a la misión Gift of Life, también a todas aquellas personas que me apoyaron sin conocerme, gracias por darme una nueva oportunidad de vida, por creer en mis conocimientos y habilidades.

AGRADECIMIENTOS

A Dios, por darme una nueva oportunidad de vivir, y tener alrededor a personas bondadosas, honestas y desinteresadas.

A mi madre Eliana por brindarme su apoyo incondicional en todo momento de mi vida, por ser la persona que me acompañó en una travesía, un país muy lejano que ni ella ni yo pensábamos viajar, para que el día de hoy pueda escribir esta tesis.

A mi padre Luis y a mis hermanos por confiar en mí, además de darme la confianza para poder desarrollarme como profesional.

A mi abuelita Lourdes, mi mamá Lula, por darme la motivación especial en mi vida, protegerme, defenderme, ser bondadosa y además de disfrutar siempre de su buena sazón.

A Margarita Zea por confiar siempre en mí y darme esa luz de esperanza que siempre siento al estar con ella.

A los profesores que me acompañaron en toda mi vida estudiantil ya que con su esmero y dedicación me formaron académicamente y moralmente hasta llegar a ser una persona responsable.

A todas aquellas personas que con su compañerismo me dieron motivación para terminar mi carrera y esta tesis.

INTRODUCCIÓN

El presente proyecto de tesis es el diseño y la construcción de un prototipo de incubación controlado a distancia para la industria avícola. El prototipo se encarga de crear las condiciones ambientales de temperatura de los huevos, poder monitorearlo y controlarlo remotamente que permitan el desarrollo embrionario de los mismos, así como su posterior nacimiento en buenas condiciones.

El presente trabajo de investigación “Diseño e implementación de un controlador PID de temperatura controlado a distancia para un prototipo de incubadora” es para uso en la industria avícola, surgió como respuesta a las diversas necesidades que actualmente existen dentro de sector, en donde los pequeños avicultores tienen que comprar huevos en regiones muy apartadas de los centros industriales donde transportar cualquier animal es difícil, acarreando costos y pérdidas a los pequeños empresarios.

El prototipo debe mantener la temperatura a 37.5° C durante todo el proceso de incubación, también se debe de observar el comportamiento de temperatura con respecto al tiempo y de poder mandar mensajes de texto para alertar cualquier situación que perjudique el proceso de incubación.

Palabras Claves: Diseño y Construcción, Prototipo, Controlador de temperatura, remotamente, avicultores.

ABSTRACT

This project proposes the design of an incubation prototype which is remote controlled for the poultry industry. The prototype is responsible for creating environmental conditions for the eggs, monitoring it remotely, allowing embryonic development and their subsequent birth in a good condition. This project "Design and implementation of a PID temperature controller, remote controlled, for a prototype incubator" is focused in the poultry industry. This idea emerged in response to the diverse needs that currently exist within the area, where small farmers should buy eggs in isolated regions in industrial centers where transporting any animal is difficult, carrying costs and as a consequence the loss of profits for the business. The prototype should keep the temperature at 37.5 °C during the incubation process. It should also monitor the temperature behavior related to time and be able to send text messages to alert any situation that prevents the incubation process.

Keywords: Design and Construction, Prototype, temperature controller, remotely, poultry.

INDICE

RESUMEN	1
DEDICATORIA	3
AGRADECIMIENTOS	4
INTRODUCCIÓN	5
ABSTRACT	6
CAPÍTULO I: PLANTEAMIENTO TEÓRICO	
1.1 Introducción	15
1.2 Antecedentes del problema	16
1.2.1 Incubacion artificial	17
1.2.2 Control Proporcional – Integral – Derivativo (P, I, D)	17
1.3 Hipótesis	18
1.4 Objetivos	18
1.4.1 Objetivo general.....	18
1.4.2 Objetivos específicos	19
1.5 Justificación e importancia.....	19
1.6 Limites y alcances.....	20
1.7 Descripción funcional del sistema.....	21
1.7.1 Sensor de temperatura	22
1.7.2 Lámparas incandescentes	22
1.7.3 PC control y monitoreo	23
1.7.4 Módulos GSM	23
1.7.5 Control	23
1.7.6 Actuador	23
1.7.7 Transductor.....	23
CAPÍTULO II: MARCO TEÓRICO	
2.1 Control de temperatura	26
2.1.1 Control de temperatura de la incubación	26
2.1.1.1 Factores que influyen en nacimiento temprano	27
2.1.1.2 Factores que influyen en nacimiento tarde o atrasado	28
2.1.2 Instrumentos de medición de temperatura.....	28
2.1.3 Sensores de temperatura	29

2.1.3.1 Sensor LM35	29
2.2 Incubación artificial	31
2.2.1 Incubación en la actualidad	33
2.2.2 Tipos de incubadoras	35
a) Incubadora Neonatal.	35
b) Incubadora en Microbiología.	36
c) Incubadora Comercial.	36
2.3 Sistemas de control y controlador Proporcional – Integral -Derivativo (PID)	37
2.3.1 Objetivos de los sistemas de control	40
2.3.2 Controlador PID	41
2.3.3 Aplicaciones de las acciones básicas usadas en control de procesos	43
2.3.3.1 Métodos de sintonización de controladores PID	43
2.4 Matlab	52
A.Toolboxes	53
B.Simulink	54
C.Otros	54
2.4.1 Identificación de sistemas – Toolbox Ident	55
2.5 Red GSM	56
2.5.1 Módem GSM	59
2.6 Visual C#	60
2.6.1 Interfaz Hombre Maquina (HMI)	61
2.6.1.1 Funciones del HMI	61
CAPÍTULO III: DISEÑO, SIMULACIÓN Y CONSTRUCCIÓN DEL PROTOTIPO	
3.1 Construcción del prototipo	65
3.1.1 Construcción de cámara térmica	65
3.1.2. Modelo de la planta a considerar para la identificación	69
3.1.3 Fundamentos Teóricos	70
3.1.3.1 Transferencia de calor	70
3.1.3.2 Identificación de sistemas	70
3.1.3.2.2 Identificación paramétrica.	72
3.1.4 Modelo matemático de la planta	72
3.1.5 Modelo matemático del actuador	75

3.1.6 Herramienta para la identificación del sistema.....	79
3.1.7 Diseño de los controladores.....	79
3.1.8 Diseño y selección del sistema de calentamiento.....	81
3.1.9 Función de transferencia	95
3.1.9.1 Función de transferencia del actuador.....	96
3.1.9.1.1 Función de transferencia del circuito de disparo.....	96
3.1.9.1.2 Función de transferencia del circuito de potencia.....	96
3.1.9.2 Función de transferencia de la planta.....	105
3.1.9.3 Función de transferencia del sensor	110
3.1.10. Diseño del control PID	113
3.1.10.1 Sintonización de las constantes PID	114
3.1.11 Diseño de los controladores digitales	125
3.1.11.1 Control PID a base de microcontrolador	125
3.1.11.2 Discretización del control PID.....	126
3.1.11.2.1 Comportamiento de control discreto PID	130
3.1.11.2.2 Implementación de la ley de control.....	132
3.2 Comunicación GSM	143
3.2.1 Comunicación de módulos.....	143
3.3 Interfaz grafica de control por computadora.....	147
CAPITULO IV: ANÁLISIS ECONÓMICO DEL PROTOTIPO DE INCUBADORA	
4.1 Costos de los componentes para la tarjeta	156
4.1.1 Costo de la cámara térmica	159
4.1.2 Costo de ingeniería	159
4.2 Costo total del prototipo de incubadora controlada remotamente	161
CAPITULO V: RESULTADOS DE PROYECTO	
5.1 Puesta en funcionamiento y pruebas.....	163
5.1.1 Puesta en funcionamiento	163
5.1.2 Prueba final.....	165
5.1.3 Comparación de temperaturas.....	166
CONCLUSIONES	170
RECOMENDACIONES	171
REFERENCIAS BIBLIOGRAFICAS.....	173

ÍNDICE DE FIGURAS

Fig.1 Funcionamiento del Prototipo de Incubadora.....	22
Fig.2 Esquema de conexión del LM35	31
Fig.3 Polluelos saliendo del nido.....	32
Fig.4 Incubadora de Alta Tecnología.	34
Fig.5 Incubadoras Neonatales.	36
Fig.6 Incubadora Bacteriológica.	36
Fig.7 Incubadora de Huevos.	37
Fig.8 Sistema de Control Básico.....	38
Fig.9 Tipos de comportamiento PID.....	42
Fig.10 Descripción de estabilización para el prototipo de incubadora.....	43
Fig.11 Característica de la salida del sistema con factor de decaimiento de un cuarto en la oscilación.	45
Fig.12 Diagrama de bloques de un sistema en lazo abierto, y representación gráfica de la entrada al mismo (escalón unitario)	45
Fig.13 Respuesta del sistema en lazo abierto ante entrada escalón unitario, y medición experimental de los parámetros K_{LA} , L y T.....	46
Fig.14 Cuadro de Dialogo PID Tunner	49
Fig.15 Tabla de parámetros y rendimiento PID Tunner	49
Fig.16 PID ajustado con exceso de cero.....	50
Fig.17 Respuesta de PID a lazo cerrado	51
Fig.18 Cuadro de diálogo de PID actualizado.....	51
Fig.19 Respuesta del sistema a Lazo cerrado	52
Fig.20 Izq. Toolbox Sisotool . Der. Toolbox Fuzzy Logic	55
Fig.21 Interfaz Simulink	54
Fig.22 Real time workshop usando microcontrolador.....	54
Fig.23 Inicialización Ident.....	55
Fig.24 Módem GSM Wavecom Fastrack Supreme 20	59
Fig.25 Antena GSM conectada al modem	60
Fig.26 Interfaz gráfica para el control de temperatura del prototipo	63
Fig.27 Cámara térmica.....	67
Fig.28 Localización de lámparas incandescentes y sensor de la cámara térmica (Vista Frontal)	68
Fig.29 Localización de ventilador (Vista Lateral derecha).....	68

Fig.30 Simplificación de la planta real para el modelado	69
Fig.31 Proceso de identificación	70
Fig.32 Modelamiento de la cámara térmica	72
Fig.33 Modelamiento del Actuador	75
Fig.34 Energía requerida en el proceso	83
Fig.35 Pared de incubadora.....	87
Fig.36 Simulación de cruce por cero de la planta	92
Fig.37 Circuito de cruce por cero	93
Fig.38 Circuito de potencia	93
Fig.39 Circuito de sensado de temperatura	94
Fig.40 Diseño de PCB.....	95
Fig.41 Diagrama de bloques del actuador	96
Fig.42 Comportamiento de Voltaje eficaz de salida	98
Fig.43 Uso de cftool para la linealización.....	99
Fig.44 Resultado de la ganancia lineal	99
Fig.45 Simulación de Etapa de Potencia con un primer ángulo de disparo	102
Fig.46 Simulación de Etapa de Potencia con un segundo ángulo de disparo.....	103
Fig.47 Simulación de Etapa de Potencia con un tercer ángulo de disparo	103
Fig.48 Ventana de inicio de Toolbox Ident	106
Fig.49 Ident Importando el data Object para calcular el modelo	106
Fig.50 Ident calculando el Modelo del Proceso.....	107
Fig.51 Ident .Condiciones iniciales para cálculo de Función de Transferencia	107
Fig.52 Aproximación de la Planta con Ident.....	108
Fig.53 Valores de Función de Transferencia	108
Fig.54 Comportamiento del filtro de primer orden.....	112
Fig.55 Diagrama de Bloques del control PID	114
Fig.56 Sistema de Control de Temperatura en simulink (PID continuo).....	114
Fig.57 Bloque de PID controller	115
Fig.58 Autoajuste PID por PID tuner	116
Fig.59 Ventana de aproximación del sintonizado de PID tuner.....	116
Fig.60 Constantes del Controlador PID encontrado.....	117
Fig.61 Comportamiento del controlador PID	117
Fig.62 Comportamiento del controlador en función de la temperatura y el ángulo de disparo	123

Fig.63 Simulación del controlador PID continuo	125
Fig.64 Diagrama de bloques de un sistema control digital.....	126
Fig.65 Sistema de control de temperatura PID discreto.....	131
Fig.66 Simulación del controlador PID discreto	131
Fig.67 Pines ATmega 144.....	135
Fig.68 Interfaz gráfica del prototipo de incubadora	147
Fig.69 Cámara térmica.....	147
Fig.70 Interior de la Cámara térmica.....	147
Fig.71 Posición del ventilador	147
Fig.72 Cámara térmica a lazo abierto	147
Fig.73 Comportamiento del controlador en prueba Final.....	166
Fig.74 Comparación de temperaturas de control implementado vs. Controlador diseñado por computador	167

ÍNDICE DE TABLAS

Tabla 1. Tipos de controladores para diversos Procesos industriales	43
Tabla 2. Parámetros de los controladores tipo PID según el primer método de Ziegler-Nichols.....	47
Tabla 3. Características de la madera	66
Tabla 4. Componentes del sistema de control de temperatura	85
Tabla 6 . Energía requerida en el proceso	93
Tabla 7 . Datos obtenidos del controlador en prueba final	166
Tabla 8 . Comparación de temperaturas de control implementado vs. Controlador diseñado por computador	93

ÍNDICE DE ANEXOS

Anexo 1. Sensor de Temperatura LM35
Anexo 2. ATmega644
Anexo 3. Wavecom FASTRACK Supreme 20
Anexo 4. Detector de Cruce por cero MOC3021
Anexo 5. MAX232
Anexo 6. Amplificador 2N2222
Anexo 7. Triac BT136

Anexo 8. Octoacoplador 4N25

Anexo 9. Diodo 1N40001

INDICE DE PLANOS

Plano 1. Vista isométrica Cámara Térmica

Plano 2. Cuerpo de Cámara Térmica

Plano 3. Tapa de Cámara Térmica

Plano 4. Bisagra, detalle de instalación

Plano 5. Configuración placa de control

Plano 6. Circuito detección de cruce por cero

Plano 7. Configuración ATmega644

Plano 8. Etapa de Potencia

Plano 9. Circuito Fuente de alimentación

Plano 10. Circuito sensado de temperatura

Plano 11. Interfaz de comunicación





CAPÍTULO I: PLANTEAMIENTO TEORICO

CAPÍTULO I: PLANTEAMIENTO TEÓRICO

1.1 INTRODUCCIÓN

En la actualidad el avance tecnológico ha dado paso a la inclusión de nuevos sistemas de control incorporando tecnología digital para la resolución de diversos tipos de control.

Dado el amplio uso de los controladores PID (Proporcional-Integrativo-Derivativo) en diversos procesos de control, el uso de microcontroladores para el desarrollo de este tipo de aplicaciones ha tomado fuerza gracias a la incorporación de lenguajes de alto nivel que facilitan ampliamente este tipo de implementaciones.

Hay lugares donde es necesario el mantener una temperatura constante ò en donde es necesario el poder variar la temperatura de cierto recinto hasta un punto determinado. Esta temperatura a su vez debe estar siendo supervisada continuamente.

Conociendo el valor de temperatura a la que se encuentra la planta, es necesaria alguna acción de control que opere dentro del área para poder manejar la variable temperatura hasta un punto de referencia requerido.

Los controladores de temperatura constituyen una de las aplicaciones de este tipo de controlador como diseño de los sistemas de control y de mucha utilidad en diferentes campos de aplicación, además se ha propuesto controlarlo a distancia.

1.2 ANTECEDENTES DEL PROBLEMA

El rendimiento y productividad de la avicultura , su explosión hacia el consumo masivo de pollo en Perú ha sido el producto de la aplicación de este tema de investigación.

Antiguamente el pollo se producía de forma muy rudimentaria, partiendo de las gallinas y gallos puestos en un corral, la gallina se encarga de poner el huevo, empollarlo, esperar a que naciera y su consecutiva crianza y engorde hasta alcanzar la edad adecuada para su consumo.

Esta forma de producción siempre fue útil y ha sido usada durante muchos años pero actualmente y debido a la mejora de procesos y la necesidad de ser más competitivos se ha hecho una transformación total de esta industria haciendo uso de las nuevas tecnologías de la automatización.

Las empresas avícolas son focos de alta tecnología y las incubadoras son solo el comienzo de una gran cadena de producción que actualmente crece a pasos agigantados.

Gracias a la utilización de las incubadoras avícolas se pueden producir grandes cantidades de pollos para engorde a precios razonables permitiendo que más familias se alimenten bien a precios menores.

La utilización de las incubadoras de gran tamaño y alta productividad se da en las grandes industrias de producción de pollo, pero existen pequeños avicultores los cuales difícilmente podrán acceder a esta tecnología.

1.2.1 INCUBACION ARTIFICIAL

Se denomina incubadora a un equipo que tiene la función común de crear un ambiente con la temperatura y la humedad adecuada para el crecimiento o reproducción de seres vivos. En casos extraordinarios al anterior se puede tener incubadoras para seres vivos mamíferos como por ejemplo las incubadoras para bebés prematuros o con enfermedades o para algunos animales que necesitan mantenerse a una buena temperatura.

Los principales tipos de incubadora son los que se utilizan en neonatología, las de uso en microbiología y las destinadas a la reproducción de especies incluyendo la producción comercial de huevo.

En las últimas décadas, se han descubierto nuevas y mejores formas de la fabricación de incubadoras de huevos de gallinas..

Otra de las formas para incubar son las revolucionarias incubadoras con controles de variables adecuadas para incubación de las diferentes especies de aves de corral, las cuales cuentan con la mejor tecnología de control para tener un excelente ambiente. Con esto se tiene mejor calidad de incubación y por consecuencia mayor producción avícola.

1.2.2 CONTROL PROPORCIONAL – INTEGRAL – DERIVATIVO (P, I, D)

Los controladores PID son los más utilizados (se estima que al menos un 90% de los reguladores empleados en la industria son PID's Este tipo de controladores no se ha visto desplazado por los modernos algoritmos de control fruto del desarrollo de las áreas de electrónica e informática (como son el Fuzzy Logic o el uso de redes neuronales). Su amplio uso radica en las dos

grandes ventajas que proporcionan estos controladores, como la robustez y las intuitivas relaciones entre sus parámetros y la respuesta del sistema.

El control PID es un controlador proporcional con acción derivativa y con acción integral simultáneamente superpuestas. Se dice que un sistema y su controlador tienen un comportamiento inestable cuando después de un tiempo razonable de funcionamiento y sin ocurrir perturbaciones externas, la temperatura permanece fluctuando en forma oscilatorio ya sea con un periodo regular o errático.

En un recinto, en el cual se tengan protocolos de control de temperatura, aire, humedad, etcétera; se requiere un control que responda más rápido y efectivamente a una respuesta a los cambios que tenga tal recinto; éste puede ser desde una recámara hasta una gran bodega donde se tenga materia que se pueda dañar con los cambios de estado. Los algoritmos están diseñados para que el operador sea el que especifique el modo del control según su necesidad y sus propias especificaciones.

1.3 HIPÓTESIS

¿Será posible poder controlar la temperatura a distancia con un controlador PID para un prototipo de incubadora?

1.4 OBJETIVOS

1.4.1 OBJETIVO GENERAL

Diseñar, construir e implementar un prototipo de incubación de huevos con control automático y controlado a distancia para la industria avícola. El equipo se encargará de crear la condición ambiental de temperatura óptima, así

también de poder gobernarlo remotamente con un interfaz fácil de supervisar y monitorear el proceso.

1.4.2 OBJETIVOS ESPECÍFICOS

- Diseñar e implementar un controlador PID para un sistema térmico basado en microcontrolador.
- Diseñar e implementar la etapa de adquisición de datos necesaria para la medición de la variable controlada: temperatura.
- Diseñar e implementar la etapa de potencia necesaria para el manejo del actuador.
- Desarrollar un programa capaz de entrar en comunicación con el microcontrolador para adquirir los datos y mantenerlo a la temperatura fijada lo más constante posible.
- Desarrollar una interfaz gráfica para la visualización y el manejo de temperatura del prototipo de incubadora.
- Construir el prototipo de la cámara térmica para la realización de pruebas.

1.5 JUSTIFICACIÓN E IMPORTANCIA

El presente proyecto de tesis surgió como respuesta a las diversas necesidades que actualmente existen dentro de sector avícola dirigido a la pequeña y micro empresa en la ciudad de Arequipa, donde los pequeños

avicultores cuentan con escaso personal que se encuentre presente en todo el proceso de incubación monitoreando y controlando la temperatura , además de innovar un proceso que no ha presentado mejoras en los últimos años poniendo a prueba un prototipo en donde una sola persona supervisaría más de una sola incubadora y lograr automatizar un conjunto éstas a bajo precio, controlándolas a distancia, continuamente y con el menor número de recursos humanos.

Esta tesis de investigación tiene la importancia en el desarrollo de un proceso telemático, además como primer paso para que en el futuro se desarrolle el control de temperatura ya no de aves sino de implementarlas en incubadoras neonatales.

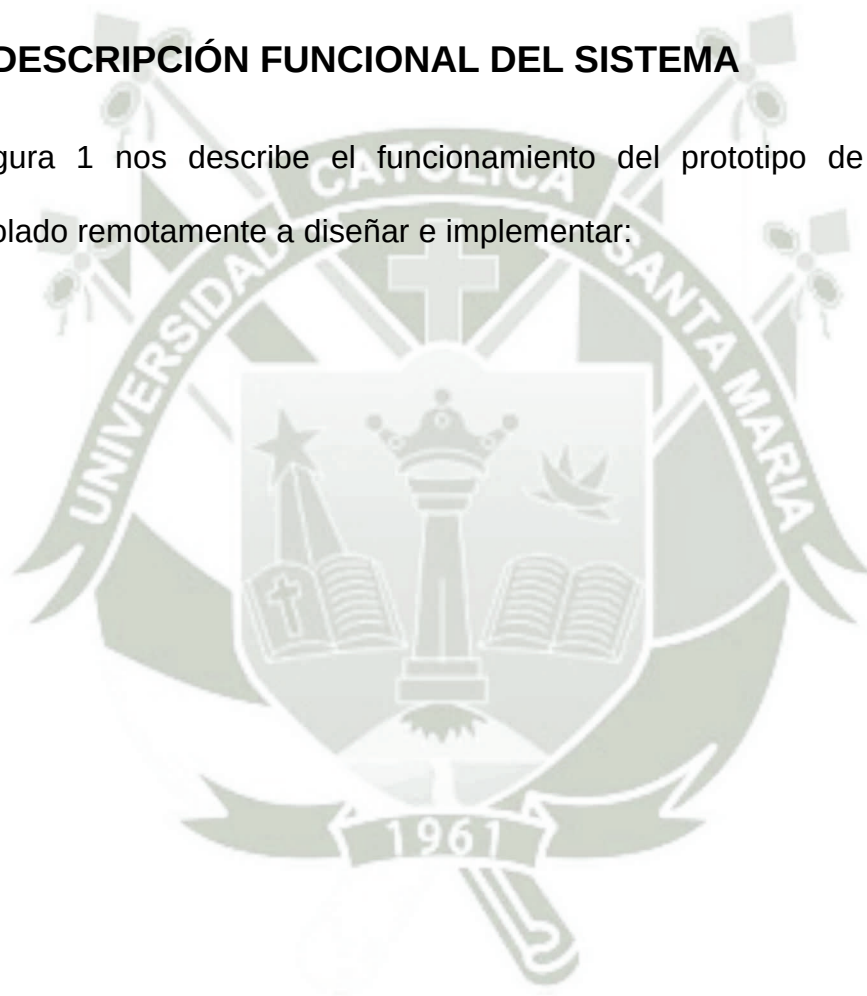
1.6 LIMITES Y ALCANCES

- En la presente tesis nos enfocaremos únicamente en la variable *temperatura*, simularemos e implementaremos una planta para que podamos manipular y ver cómo se comporta esta variable
- El proyecto utilizará la tecnología GSM, ya que la comunicación que haremos entre la incubadora a la PC controlada remotamente usará esta tecnología.
- Proponer una solución a los problemas del control a distancia, cuando éste se quiera realizar en distancias muy lejanas, al usar esta tecnología GSM se debe optimizar la transmisión de datos siendo más rápido y más confiables.

- Se utilizará el método más certero y eficaz para el control a distancia.
- El proyecto no ahondará temas de telecomunicaciones avanzado ni brindará nuevas ecuaciones físicas, se usarán las existentes.
- Utilizando herramientas, se realizarán las programaciones en PC y en el microcontrolador de la placa de control.

1.7 DESCRIPCIÓN FUNCIONAL DEL SISTEMA

La figura 1 nos describe el funcionamiento del prototipo de incubadora controlado remotamente a diseñar e implementar:



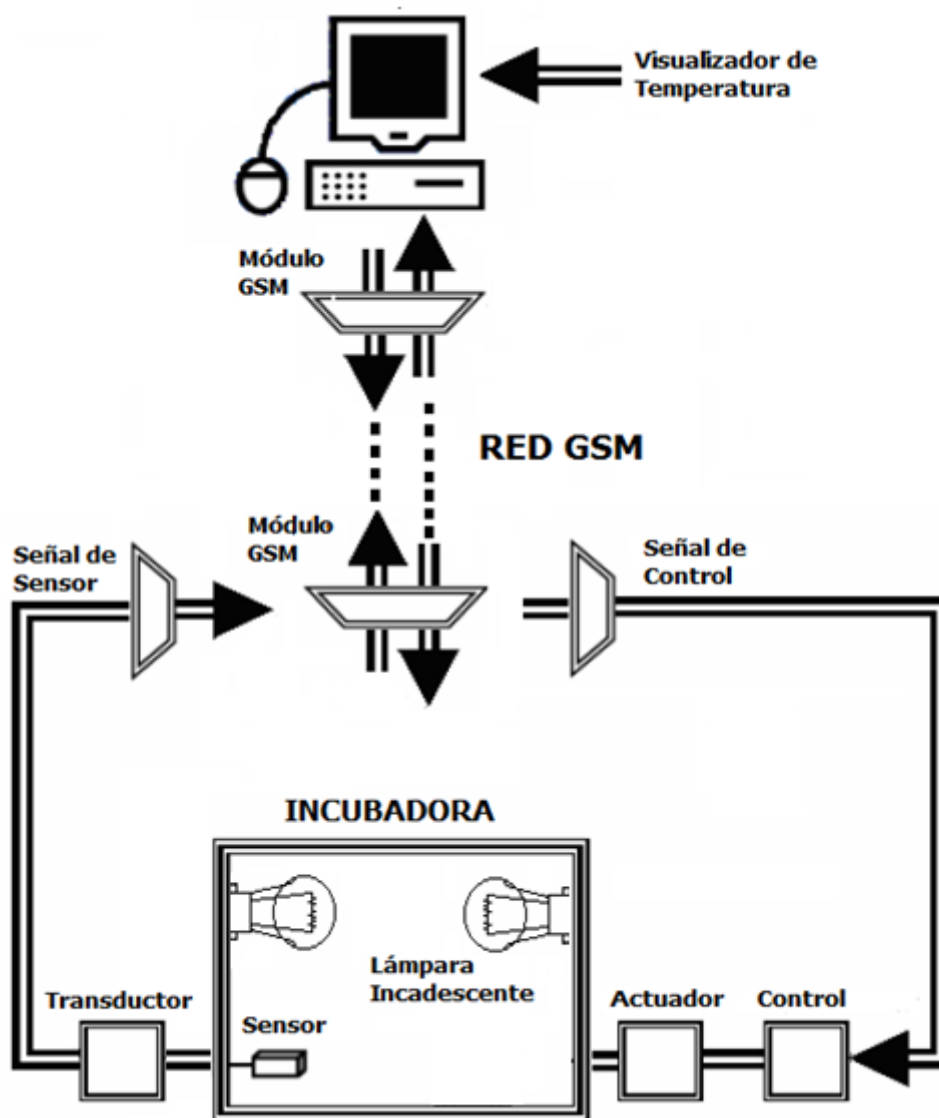


Fig. 1. Funcionamiento del Prototipo de Incubadora

1.7.1 SENSOR DE TEMPERATURA

Es el encargado de percibir la variación de temperatura y así ofrecer datos del comportamiento de la temperatura al usuario para su posterior análisis.

1.7.2 LÁMPARAS INCANDESCENTES

Es la fuente de calor del prototipo, va a ser controlado en su potencia para mantener la temperatura lo más cercana posible al Set - Point.

1.7.3 PC CONTROL Y MONITOREO

Este bloque comprende un programa desarrollado en Visual C sharp para controlar la temperatura automáticamente, insertar el Set Point y monitorear su comportamiento.

1.7.4 MODULOS GSM

Estos equipos permiten la comunicación remota ente la PC y la etapa de control., también es el encargado de enviar remotamente los datos obtenidos del sensor de temperatura a la PC.

1.7.5 CONTROL

En este bloque se implementa en el microcontrolador, el cual lleva un programa que controla toda la planta , en el control PID se hace el sintonizado para poder mantenerla en la temperatura fijada (Set-point).

1.7.6 ACTUADOR

Este bloque convierte y adapta las señales digitales entregadas por el circuito de control, encargado de regular la intensidad de la lámpara incandescente.

1.7.7 TRANSDUCTOR

Es el encargado de transforma la señal física (señal térmica) en una señal digital (voltaje). Para que los datos sean entregados con éxito del sensor de temperatura a la PC.

El proyecto funciona de la siguiente manera:

1. En la PC se introduce manualmente un set point, la señal del set point es enviada desde la PC por medio un módulo gsm a otro módulo que está conectado a la etapa de control implementada en el microcontrolador.
2. En el microcontrolador se implementa un programa por medio de un controlador que analiza la señal de error y envía una señal de cruce por cero al circuito de potencia, el mismo que regular la corriente por medio de señales PWM que circula por la bombilla eléctrica y variar su intensidad.
3. Luego de un tiempo determinado el sistema logra estabilizarse con un error de estado estable.
4. El sensor detecta la temperatura (set point), mediante un transductor convertirá la señal térmica en señal de voltaje la cual se envía por los módulos gsm y compara y tiene que mantener la temperatura lo más cercana al set point.

CAPÍTULO II: MARCO TEÓRICO



CAPÍTULO II: MARCO TEÓRICO

2.1 CONTROL DE TEMPERATURA

La temperatura es la propiedad que posee un cuerpo o sustancia que determina si están en equilibrio térmico. Este equilibrio térmico se produce cuando no existe transferencia de calor de un cuerpo hacia otro.

Un sistema de control de temperatura, obtiene su señal del ambiente mediante un sensor y la señal obtenida es tratada, ya sea analógicamente o digitalmente (según el tipo de señal que se utilice). Todo esto pasa por un sistema de control, el cual toma por sí mismo la decisión de activar o desactivar, aumentar o disminuir, el proceso que estará encargando de mantener la temperatura ambiente en un sistema térmico. Por ejemplo, en un control de un refrigerador o de un horno eléctrico, el sistema de control se encarga de disminuir la temperatura, en el caso del refrigerador es tratar de mantener un nivel de enfriamiento y en el caso del horno eléctrico es aumentar y disminuir según sea necesario o establecer un nivel de calefacción que se requiera para cada alimento o valores deseados que dependen del usuario.

2.1.1 CONTROL DE TEMPERATURA DE LA INCUBACIÓN

La temperatura corporal de la gallina varía ligeramente entre los primeros días de incubación y los últimos, siendo de unos 36.5°C al comienzo, hasta algo más de 39.5°C al final del proceso. No obstante, en la práctica esta pequeña diferencia puede obviarse y mantener todo el proceso con temperatura regulada a 37.5°C con los mismos resultados.

El proceso de formación y nacimiento de los polluelos es muy sensible a la temperatura de permanencia del huevo, de tal forma que puede decirse que si la temperatura se mantiene por debajo de 36°C por largos períodos la eficiencia de la incubación se reduce y puede incluso ser cero. Peor es el caso cuando la temperatura sobrepasa los 40°C, con esta temperatura el proceso se deteriora y casi todos los polluelos mueren o los huevos se pudren.

Se ha observado que el rango de temperatura es muy estrecho, lo que significa que un factor muy importante para el éxito de la incubación radica en utilizar un sistema de regulación de la temperatura que sea sensible al intervalo de 1°C, para así lograr que toda la incubación se realice entre 37 °C y 38°C

2.1.1.1 FACTORES QUE INFLUYEN EN NACIMIENTO TEMPRANO

- Períodos de precalentamiento muy largos
- Incubación de huevos muy temprano. Muchas horas de incubación
- Incorrecta temperatura y humedad de incubadoras.
- Sitios muy calientes dentro de incubadora.
- Ventilación incorrecta
- Cambios de temperatura estacionales afectando el ambiente de la incubadora

2.1.1.2 FACTORES QUE INFLUYEN EN NACIMIENTO TARDE O ATRASADO

- Incubación de huevos muy tarde
- Incorrecta temperatura y humedad de incubadoras
- Ventilación incorrecta
- Cambios de temperatura estacionales afectando el ambiente de la incubadora
- Huevos que han sido almacenados por largos períodos de tiempo
- Huevos que han sido almacenados a temperaturas muy bajas
- Enfermedad y problemas de fertilidad

2.1.2 INSTRUMENTOS DE MEDICIÓN DE TEMPERATURA

La clase de tarea de medición determina el tipo de sensor. El sensor de temperatura más idóneo se selecciona según los siguientes criterios:

- Rango de medición
- Exactitud
- Diseño según el lugar de medición.
- Tiempo de respuesta
- Resistencia

En el mercado se dispone de una gran variedad de elementos sensores e instrumentos de medición de la temperatura para ofrecer el más adecuado,

como son: los termopares o termocuplas, RTD, termistores, termopilas, Sensores IC (Circuitos Integrados), etc.

2.1.3 SENSORES DE TEMPERATURA

Un sensor es un dispositivo que realiza la conversión física, como temperatura, distancia, presión, velocidad; a un valor que es más fácil medir. Los sensores de temperatura son los elementos primarios de medición de variables del proceso, siendo utilizados para la lectura e indicación y para la transformar la variable medida en una señal eléctrica.

Existen varios tipos de sensores de temperatura como se han mencionado para la medición, la cual sirve para la adquisición de datos para el proceso de control que se requiera. El que se usa en el proyecto de investigación es el LM35.

2.1.3.1 SENSOR LM35

Es un sensor de temperatura integrado de precisión, cuya tensión de salida es linealmente proporcional a la temperatura en grados centígrados ($^{\circ}\text{C}$). El LM35 por lo tanto tiene una ventaja sobre los sensores de temperatura lineal calibrada en grados Kelvin; que el usuario no está obligado a restar una gran tensión constante para obtener grados centígrados.

El LM35 no requiere ninguna calibración externa o ajuste para proporcionar una precisión típica de $+ 1.4^{\circ}\text{C}$ a temperatura ambiente y $+ 3.4^{\circ}\text{C}$ a lo largo de su rango de temperatura (de -55°C a 150°C). El dispositivo se ajusta y calibra durante el proceso de producción. La baja impedancia de salida lineal

y la precisa calibración inherente, permiten la creación de circuitos de lectura o control especialmente sencillos. El LM35 puede funcionar con alimentación simple o alimentación doble (+ y -).

Requiere sólo 60 μ A para alimentarse, y bajo factor de auto-calentamiento, menos de 0,1°C en aire estático. El LM35 está preparado para trabajar en una gama de temperaturas que abarca desde los -55 °C bajo cero a 150 °C.

2.1.3.1.1 CARACTERÍSTICAS DEL LM35

- Calibrado directamente en grados Celsius (Centígrados).
- Factor de escala lineal de +10 mV/ °C .
- 0,5 °C de precisión a +25 °C.
- Rango de trabajo: -55 °C a +150 °C.
- Apropiado para aplicaciones remotas.
- Bajo coste.
- Funciona con alimentaciones entre 4 V y 30 V.
- Menos de 60 μ A de consumo.
- Bajo auto-calentamiento (0,08 °C en aire estático).
- Baja impedancia de salida, 0,1 W para cargas de 1 mA.
- Diagrama de conexión figura 2..

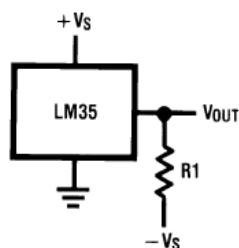


Fig.2. Esquema de conexión del LM35

En el apéndice Sensor LM35, se muestran las curvas de respuestas de este sensor de temperatura.

2.2 INCUBACIÓN ARTIFICIAL

La incubación artificial es un proceso en el cual se trata de recrear lo que la naturaleza hace desde millones de años atrás. Este es un proceso en el cual se trata de empollar huevos de gallina con un aparato que los mantendrá calientes a los que estén fertilizados, en el cual se reemplazó la gallina que incuba por una máquina que hará este mismo proceso.

Uno de los grandes beneficios que se tiene con la incubación artificial es la de empollar una mayor cantidad de huevos, pero esto también tiene un factor en contra: el porcentaje de error que consiste en la proporción de huevos con embrión no desarrollado, muertes de embriones prematuros dentro del huevo o muertes de polluelos después de nacer.

Una gallina en su estado natural y con su instinto mueve los huevos de posición para que así, el embrión no se adhiera a la pared del huevo, dando por lo menos dos vueltas por día a cada huevo. Y en una incubadora se necesita también este proceso, que por lo menos a partir del tercer día se comience con este proceso de volteo de huevos.

Lo anterior se denomina requisito para una incubadora, otros de estos requisitos es la ventilación, para el huevo fertilizado (con embrión) produce CO_2 y tiene que ser eliminado con ventilación pero no con suficiente como para enfriar la incubadora; otro requisito en la humedad, para un huevo fertilizado es necesaria la humedad para que así, se evite que el huevo y el embrión se deshidraten, por ultimo no se tienen que dar vuelta los últimos tres días de eclosión de los huevos para darles la oportunidad de descansar para su nacimiento y después del nacimiento otros tres días de incubación de los polluelos nacidos y posteriormente extraerlos de la incubadora (véase figura 3).



Fig.3. Polluelos saliendo del nido

La incubadora debe desinfectarse completamente para evitar enfermedades en las siguientes incubaciones posteriores, esto debe ser un requisito de mayor prioridad para una incubadora que esté en constante funcionamiento.

Para lo anterior, se trata desde luego recrear las características de que se tiene en una incubación natural.

2.2.1 INCUBACIÓN EN LA ACTUALIDAD

Para favorecer un proceso de incubación correcto habrá que operar durante la fase de crianza, evitando que el huevo permanezca a la intemperie, se exponga al frío intenso, a corrientes de aire, a la luz directa del sol, al calor y a la humedad excesiva; y es por este motivo que se aconseja realizar la recogida de huevos varias veces al día utilizando unas bandejas (portahuevos), las cuales garantizan, gracias a los materiales empleados en su fabricación, una extrema facilidad de limpieza y desinfección.

Habrà que descartar los huevos rotos, agrietados y con la cáscara irregular, los que tienen yema doble o manchas de sangre con forma anómala; y se tendrán que conservar en locales idóneos con temperaturas alrededor de los 14°C - 16°C por no más de siete días, ya que tiempos más prolongados afectarían de forma negativa el porcentaje de eclosión, pero de todos modos, nunca menos de tres días.

Las incubadoras tienen que estar colocadas en ambientes sanos, que se puedan airear con facilidad y a una distancia de al menos 60 cm de las paredes, evitando su exposición directa a los rayos del sol o a fuentes de calor. La temperatura ideal del local donde se encuentren las máquinas deberá hallarse entre los 17°C y los 23°C .

La incubación propiamente mencionada inicia en el momento en el que los huevos son colocados en la incubadora y termina aproximadamente 2/3 días

antes del presunto nacimiento del pollito. Desde este momento empezará la fase de eclosión¹ que terminará con la salida del pollito del cascaron.



Fig. 4. Incubadora de Alta Tecnología.

Fueron los estadounidenses quienes comenzaron a desarrollar nuevos sistemas de producción que resultarían mucho más rentables y viables, las nuevas explotaciones permitirían reducir el precio de venta al mercado de pollos y gallinas (véase fig.4.)

Esta parte del proceso de la producción avícola industrial comienza con la recolección de miles de huevos fertilizados que son almacenados en bandejas e introducidos en cámaras especiales donde se reproducen las condiciones ambientales adecuadas para que los polluelos puedan desarrollarse, la temperatura mantenida es de 37.5 °C y la humedad relativa es del 55% , salvo en los tres últimos días en los que esta se eleva a un 70% a fin de que la cáscara se ablande y los pollitos puedan romperla con más facilidad. El sistema productivo contempla también el giro de los huevos cada

¹ Eclosión: Aparición o salida, especialmente de un animal

hora tal y como los girarían las gallinas en condiciones normales, este proceso de incubación dura unos 19 días.

A los 18,5 días los huevos se retiran de las cámaras de incubación y se verifica que son huevos fertilizados a través de un sensor infrarrojo que detecta la opacidad del contenido, si son traslúcidos indicará que son huevos que no han sido fertilizados y por tanto se retiran de la producción. Todo se aprovecha, estos huevos “defectuosos” se procesarán y convertirán en alimentos para animales.

2.2.2 TIPOS DE INCUBADORAS

Existen tres tipos de incubadoras que se utilizan en diversas áreas de investigación, producción y medicina.

a) Incubadora Neonatal. Este tipo de incubadora están elaboradas con material transparente, acolchonada y esterilizada, para que los bebés prematuros o neonatos (recién nacidos), con calefacción por convección, filtro de aire exterior, ventanas para manipular al paciente, diversos y sofisticados sistemas de monitoreo que incluyen control de peso, respiración, cardíaco y de actividad cerebral; son esenciales en el área de recién nacidos en una clínica (véase fig. 5).



Fig. 5. Incubadoras Neonatales

b) Incubadora en Microbiología. Incubadoras utilizadas en la investigación de microbiología en las cuales se controla como cualquier incubadora, la temperatura y la humedad, pero en este caso para el cultivo de microbiología (véase figura 6).



Fig. 6. Incubadora Bactereológica.

c) Incubadora Comercial. Este tipo de incubadoras están orientadas para la producción de alimentos, principalmente de huevos y crías de aves en la cual se trata de recrear y sustituir la incubación natural por una artificial.

Con incubadoras comerciales o de procesos artificiales se obtienen mejores resultados, con los cuales permite un control automático, de temperatura y

humedad ambiental, mediante ventilación, termostatos y temporizadores. En el mercado existen incubadoras industriales para pocas decenas de huevos hasta miles de huevos a empollar (véase figura 7).



Fig. 7. Incubadora Industrial de Huevos

2.3 SISTEMAS DE CONTROL Y CONTROLADOR PROPORCIONAL – INTEGRAL -DERIVATIVO (PID)

Los sistemas de control son aquellos dedicados a obtener la salida deseada de un sistema o proceso. En un sistema general se tienen una serie de entradas que provienen del sistema a controlar, llamado planta; y se diseña un sistema para que, a partir de estas entradas, modifique ciertos parámetros en el sistema o planta, con lo que las señales anteriores volverán a su estado normal ante cualquier variación (véase figura 8).

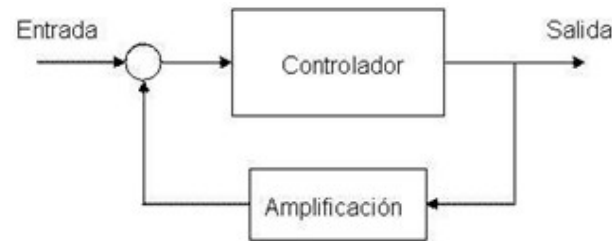


Fig. 8. Sistema de Control Básico

Hay varias clasificaciones dentro de los sistemas de control. Atendiéndose a su naturaleza son analógicos, digitales o mixtos; también a su estructura (número de entradas y salidas), puede ser control clásico o control moderno; lo cual su diseño pueden ser por lógica difusa, o redes neuronales [

Los principales tipos de sistemas de control son:

A. SÍ/NO. En este sistema el controlador enciende o apaga la entrada y es utilizado, por ejemplo, en el alumbrado público, ya que éste se enciende cuando la luz ambiental es más baja que un nivel predeterminado de luminosidad.

B. CONTROL PWM Y TIEMPO DE CICLO. El controlador por modulación por ancho de pulso o PWM (Pulse Width Modulation) es un tipo de control en el que se modifica el ciclo de trabajo de una señal periódica ya sea sinusoidal o cuadrada.

C. PROPORCIONAL (P). En este sistema la amplitud de la señal de entrada al sistema afecta directamente la salida, ya no es solamente un nivel prefijado sino toda la gama de niveles de entrada. Algunos sistemas automáticos de

iluminación utilizan un sistema P para determinar con qué intensidad puede encender lámparas dependiendo directamente de la luminosidad ambiental.

D. PROPORCIONAL DERIVATIVO (PD). En este sistema, la velocidad de cambio de la señal de entrada se utiliza para determinar el factor de amplificación, calculando la derivada de la señal.

E. PROPORCIONAL INTEGRAL (PI). Este sistema es similar al anterior, solo que la señal se integra en vez de derivarse.

F. PROPORCIONAL INTEGRAL DERIVATIVO (PID). Este sistema combina los dos tipos anteriores.

G. REDES NEURONALES. Este sistema modela el proceso de aprendizaje del cerebro humano para aprender a controlar la señal de salida.

El tipo de control utilizado es seleccionado por reglas de control y depende mucho de la planta que se está controlando, en los sistemas de control de procesos que tenían controladores neumáticos, actualmente están siendo reemplazados por sistemas electrónicos, en la actualidad con la experiencia adquirida, algunos autores proponen las siguientes especificaciones de acciones básicas de control

Sistema a controlar acciones básicas a aplicar:

Control de presión de líquidos	P+I
Control de presión de gases	P
Control de Caudal	P+I
Control de Temperatura	P+I+D
Control de Nivel	P
Control de Presión de Vapores	P+I+D

Tabla 1. Tipos de controladores para diversos Procesos industriales

2.3.1 OBJETIVOS DE LOS SISTEMAS DE CONTROL

Un sistema de control debe ser capaz de conseguir su objetivo cumpliendo los siguientes requisitos:

- Garantizar la estabilidad y particularmente ser robusto frente a perturbaciones y errores en los modelos.
- Ser tan eficiente como sea posible, según criterio preestablecido. normalmente este criterio consiste en que la acción de control sobre las variables de entrada, sea realizable, evitando comportamiento brusco e irreal.
- Ser de fácil implementación y cómodo de operar en tiempo real con ayuda de un ordenador.

2.3.2 CONTROLADOR PID

El control PID es un controlador proporcional con acción derivativa y con acción integral simultáneamente superpuestas. Se dice que un sistema y su controlador tienen un comportamiento inestable cuando después de un tiempo razonable de funcionamiento y sin ocurrir perturbaciones externas, la temperatura permanece fluctuando en forma oscilatorio ya sea con un periodo regular o errático.

Siempre se busca que el sistema de control sea estable, pero además, dentro de las condiciones de estabildades existen tres tipos de comportamiento bien definidos: control subamortiguado, control con amortiguamiento crítico y control sobre amortiguado.

El sistema sobreamortiguado tiene una velocidad de respuesta lenta, después de que ocurra una perturbación, el sistema puede tardar en volver al SP, pero la ventaja es que el sistema es muy estable y no adquiere comportamientos oscilatorios indeseables. Esta condición tiende a ocurrir cuando la banda proporcional cuando la banda P_b es más grande de lo necesario. También puede deberse a una constante derivativa D muy grandes, basta acordarse de que la acción derivativa tiene a frenar la temperatura.

En el otro extremo, cuando un sistema se comporta de modo subamortiguado la velocidad de respuesta es muy buena pero pueden ocurrir varias oscilaciones de cierta amplitud antes de que la temperatura llegue a un valor estable. Suele ocurrir esta condición cuando la banda proporcional P_b es

pequeña (se parece a un ON-OFF), la constante derivativa D pequeña y la constante de integración I grande.

El justo medio entre las condiciones anteriores es el amortiguamiento crítico. A esta condición corresponde los valores óptimos de los parámetros P, I, D. en este caso el sistema es bastante estable y la velocidad de respuesta es la mejor que se puede lograr.

$$\text{Salida} = [P + I + D]$$

En términos de ganancias:

$$\text{Salida} = K_p + K_d + K_i$$

En la figura 9 se muestra ejemplos de estos comportamientos.

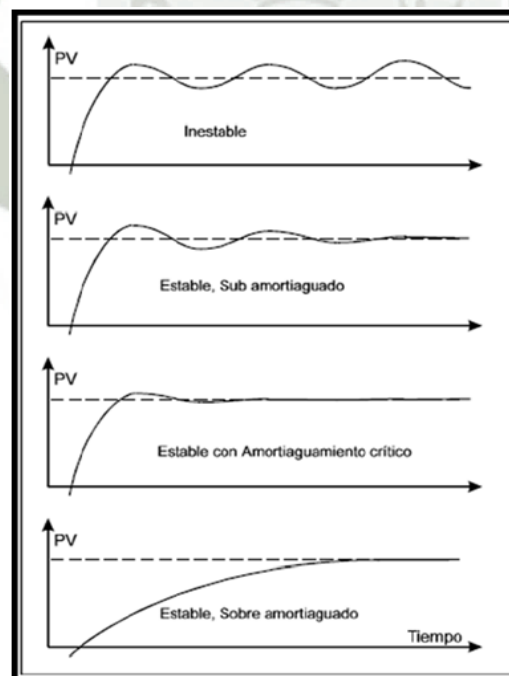


Fig.9. Tipos de comportamiento PID

Los controladores PID son los más utilizados (se estima que al menos un 90% de los reguladores empleados en la industria son PID's o variantes del mismo). Este tipo de controladores no se ha visto desplazado por los modernos algoritmos de control fruto del desarrollo de las áreas de electrónica e informática. Su amplio uso radica en las dos grandes ventajas que proporcionan estos controladores, como la robustez y las intuitivas relaciones entre sus parámetros y la respuesta del sistema.

Para el prototipo de incubadora, el rango del mando de temperatura deberá de estabilizarse a temperatura a 37.5°C .

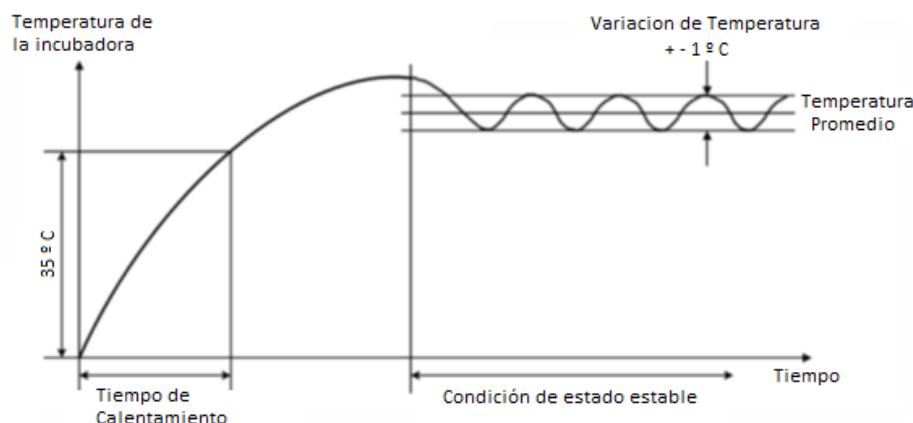


Fig.10. Descripción de estabilización para nuestro prototipo de incubadora

2.3.3 APLICACIONES DE LAS ACCIONES BÁSICAS USADAS EN CONTROL DE PROCESOS

2.3.3.1 METODOS DE SINTONIZACION DE CONTROLADORES PID

La forma más exacta de conocer un sistema, es conociendo su modelo matemático, el diseño de un controlador PID adecuado para que el sistema se

comporte en lazo cerrado de la forma deseada es sencillo, y existen varias técnicas para conseguirlo.

Ahora bien, si no se conoce de forma precisa, no será posible obtener analíticamente el controlador deseado y será necesario recurrir a aproximaciones experimentales para sintonizarlo.

El proceso de obtener los parámetros adecuados para que se cumplan en lazo cerrado, es decir, las especificaciones deseadas en el diseño se conoce como sintonía del controlador.

En esta sección se va a explicar el método de sintonía del PID: Sintonía de Ziegler-Nichols, Primer y Segundo método, en la cual nos basaremos para la sintonización de nuestro controlador.

Estas reglas tienden a limitar el máximo sobre – impulso o decaimiento en un 25%.

2.3.3.1.1 MÉTODO DE SINTONIZACION ZIEGLER - NICHOLS

Existen dos métodos llamados reglas de sintonía de Ziegler-Nichols, y en ambos la característica común es que en general se consigue un factor de decaimiento o de sobre-impulso del 25%, esto es, que el valor de la salida ante entrada escalón decae a un cuarto en un periodo de oscilación.

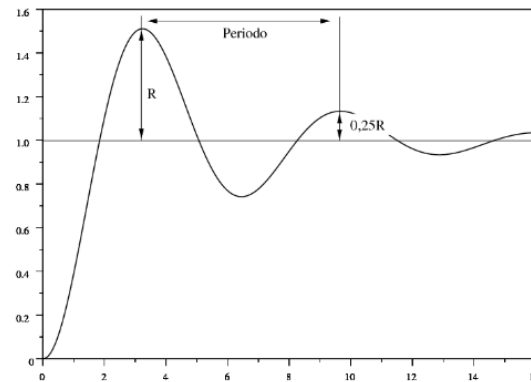


Fig.11. Característica de la salida del sistema con factor de decaimiento de un cuarto en la oscilación.

Ziegler y Nichols sugirieron una serie de reglas para sintonizar PID's basadas en un estudio experimental de múltiples respuestas de sistemas (ante entrada escalón, y estudiando la ganancia de estabilidad crítica en el caso de un sistema con control proporcional).

2.3.3.1.1.1 ZIEGLER – NICHOLS: PRIMER MÉTODO

Llamado también método de Ziegler – Nichols en lazo abierto, en este primer método, se obtiene experimentalmente la respuesta de la planta ante entrada escalón unitario como se ve en la Fig. 12.

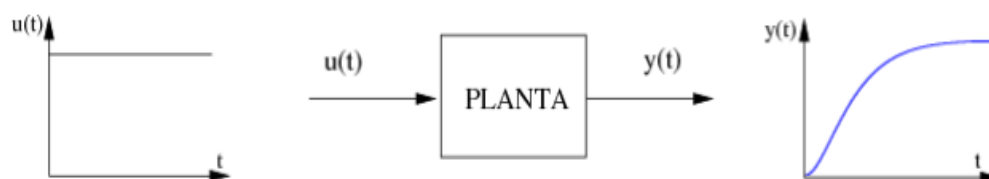


Fig. 12. Diagrama de bloques de un sistema en lazo abierto, y representación gráfica de la entrada al mismo (escalón unitario) y la respuesta a dicha entrada

Si la planta, que por supuesto tiene que ser estable, no tiene ni integradores ni parejas de polos complejos conjugados dominantes, entonces la respuesta del mismo tendrá la forma de S (S-shaped). Como inconveniente de este método se puede decir que no es aplicable a sistemas cuya salida no tenga la forma de S.

Este tipo de salidas se pueden caracterizar con tres constantes (K_{LA} , L y T) que se pueden obtener gráficamente de la salida del sistema en lazo abierto ante entrada escalón unitario (Figura 13).

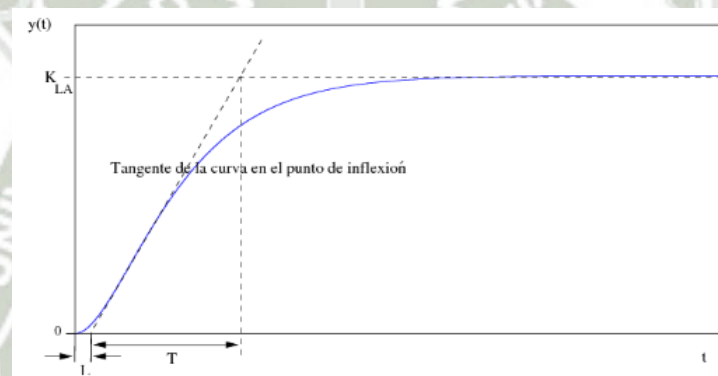


Fig. 13. Respuesta del sistema en lazo abierto ante entrada escalón unitario, y medición experimental de los parámetros K_{LA} , L y T .

La función de transferencia del sistema se puede aproximar entonces por una de primer orden con retardo de la forma:

$$G(s) = \frac{Y(s)}{U(s)} = \frac{K_{LA}e^{-Ls}}{1 + Ts}$$

Ziegler y Nichols² sugirieron que los valores de los parámetros del PID (K, Td y Ti) más adecuados para este tipo de sistemas son los que se muestran en la Tabla 2.

Tipo de controlador	K	Ti	Td
P	$\frac{T}{L}$	∞	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	$2L$	$0.5 L$

Tabla 2. Parámetros de los controladores tipo PID según el primer método de Ziegler-Nichols.

2.3.3.1.2 MÉTODOS COMPUTACIONALES USANDO MATLAB

En la actualidad con el desarrollo de procesadores y computadoras muy potentes es posible el análisis y modelación de sistemas de control o simulaciones, con relativa facilidad y rapidez.

Gracias al software que se ha especializado, para su utilización en ingeniería e investigaciones científicas, por lo que diseñar se hace una tarea menos laboriosa y complicada.

² Los valores obtenidos con cualquiera de los dos métodos de sintonía de Ziegler-Nichols no aseguran un comportamiento óptimo, simplemente son en la mayoría de los casos una buena aproximación al control ideal. Por ello será necesario realizar un reajuste de los parámetros del PID con otros métodos para mejorar la dinámica del sistema realimentado. Existen plantas en donde no se puede sintonizar con este método.

Matlab, nos ofrece una buena variedad de herramientas gráficas, analíticas para el análisis y simulación de sistemas de control, ya sea en tiempo continuo, o en tiempo discreto

2.3.3.1.2.1 AJUSTE AUTOMÁTICO DEL REGULADOR PID BLOQUE SIMULINK (PID TUNER)

Sintonizador PID proporciona un método de sintonización PID de un solo lazo rápido y ampliamente aplicable para los bloques del controlador PID de Simulink. Con este método, se pueden ajustar los parámetros PID para lograr un diseño robusto con el tiempo de respuesta deseado.

2.3.3.1.2.2 DISEÑO PID INICIAL

Cuando se pone en marcha el sintonizador PID, el software calcula un modelo de planta linealizado visto por el controlador. El software identifica automáticamente la entrada y salida de la planta, y utiliza el punto de funcionamiento actual de la linealización. La planta puede tener cualquier orden y puede tener retrasos.

El sintonizador PID calcula un controlador PI inicial de lograr un equilibrio razonable entre el rendimiento y robustez. De forma predeterminada, el rendimiento de rastreo de referencia paso muestra en el diagrama.

La figura 14 muestra el cuadro de diálogo PID Tuner con el diseño inicial:

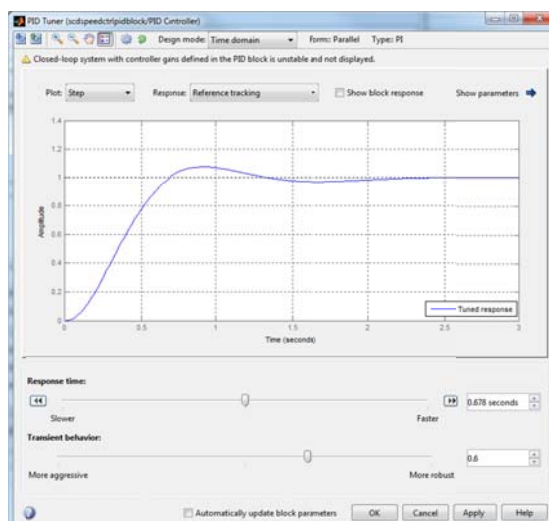


Fig. 14. Cuadro de Dialogo PID Tunner

2.3.3.1.2.3 SINTONIZANDO LOS PARÁMETROS PID

En esta ventana se sintoniza los parámetros del controlador P, I y D, y observando un conjunto de resultados y mediciones de robustez. La figura 15 muestra las tablas de parámetros y el rendimiento:

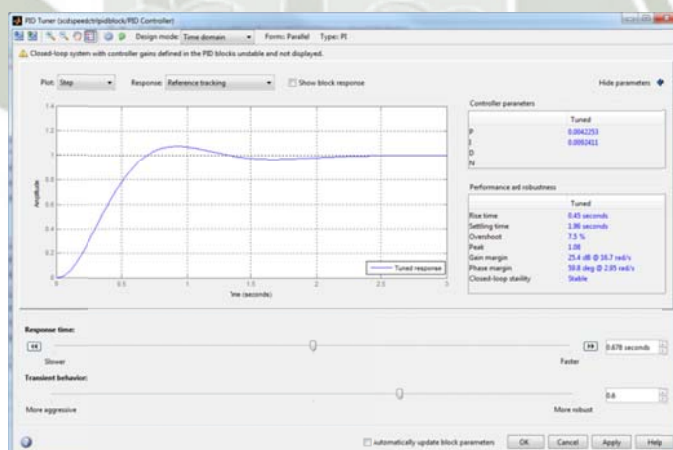


Fig. 15. Tabla de parámetros y rendimiento PID Tunner

2.3.3.1.2.4 AJUSTE DE DISEÑO PID EN EL SINTONIZADOR PID

La sintonización se va logrando deslizando con el control deslizante tiempo de respuesta a la izquierda para aumentar el tiempo de respuesta de bucle cerrado. Tenga en cuenta que cuando se ajusta el tiempo de respuesta, la trama de respuesta y los parámetros del controlador y el rendimiento de actualización de mediciones.

La figura 16 muestra un diseño PID ajustado con un exceso de cero y un tiempo de establecimiento de t segundos. El controlador diseñado se convierte efectivamente en un controlador de sólo integrante.

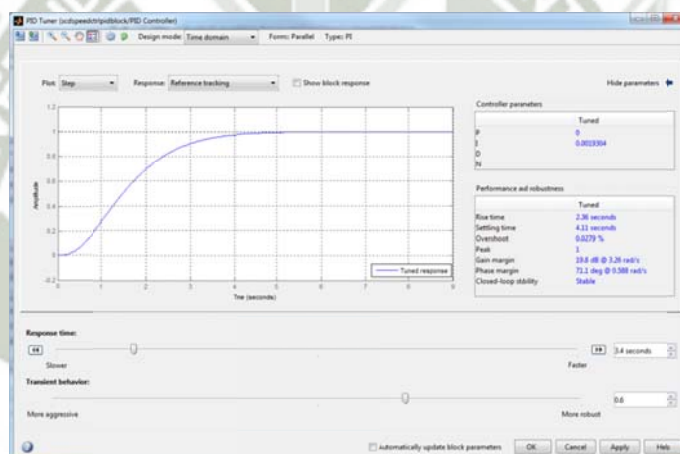


Fig. 16. PID ajustado con exceso de cero

2.3.3.1.2.5 COMPLETAR DISEÑO PID CON PERFORMANCE COMERCIO-OFF

Se necesita asegurarse que la respuesta de control sea más rápida para reducir el tiempo de asentamiento y aumentar la robustez de reducir el exceso. La figura 17 muestra la respuesta a lazo cerrado con diferentes valores:

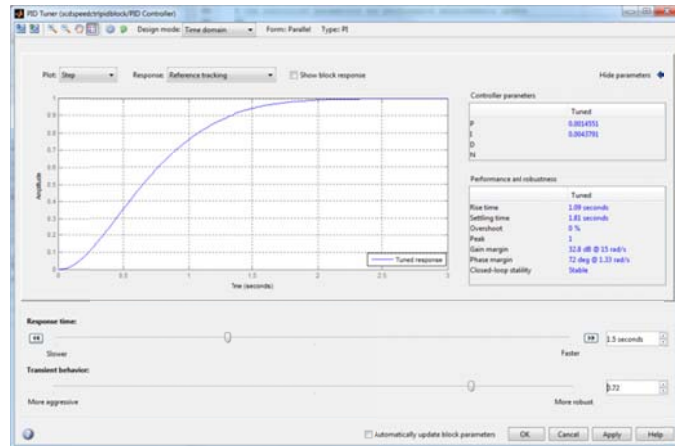


Fig. 17. Respuesta de PID a lazo cerrado

2.3.3.1.2.6 ESCRIBIR LOS PARÁMETROS AFINADOS A BLOCK CONTROLADOR PID

Una vez conformes con el rendimiento del controlador en la planta modelo se guardan los datos actualizados.

La figura 18 muestra el cuadro de diálogo de bloques del controlador PID actualización:

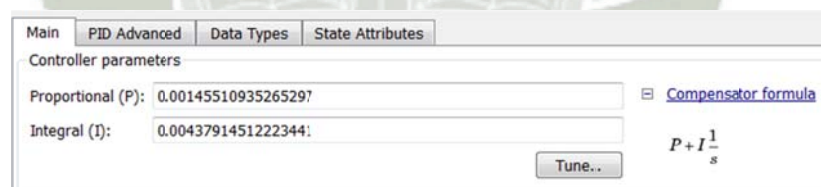


Fig. 18. Cuadro de diálogo de PID actualizado

2.3.3.1.2.7 DISEÑO COMPLETADO

La figura 19 muestra la respuesta del sistema de circuito cerrado:

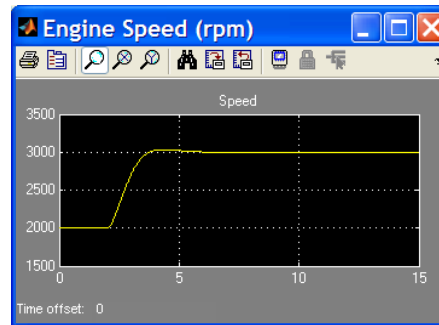


Fig. 19. Respuesta del sistema a Lazo cerrado

Se pondrá énfasis en los resultados obtenidos gráfica y analíticamente, con el fin de analizar el desempeño, y evaluar distintos parámetros de los procesos industriales que se pretenden controlar.

2.4 MATLAB

Matlab es un lenguaje de alto funcionamiento para computación técnica, integra computación, visualización, y programación, en un entorno fácil de usar donde los problemas y las soluciones son expresados en la más familiar notación matemática. Los usos más familiares de Matlab son:

- Matemática y Computación
- Desarrollo de algoritmos
- Desarrollo de algoritmos
- Modelamiento , simulación y prototipado
- Análisis de datos, exploración y visualización
- Graficas científicas e ingenieriles
- Desarrollo de aplicaciones, incluyendo construcción de interfaces graficas de usuario

MATLAB es un sistema interactivo cuyo elemento básico de almacenamiento de información es la matriz, que tiene una característica fundamental y es que no necesita dimensionamiento. Esto le permite resolver varios problemas de computación técnica (especialmente aquellos que tienen formulaciones matriciales y vectoriales) en una fracción de tiempo similar al que se gastaría cuando se escribe un programa en un lenguaje no interactivo como C.

Matlab presenta una familia de soluciones a aplicaciones específicas de acoplamiento rápido llamadas ToolBoxes, para la simulación de sistemas dinámicos, Simulink.

A. TOOLBOXES

Los toolboxes son colecciones muy comprensibles de funciones MATLAB, o archivos de matlab (M-files) que extienden el entorno de MATLAB para resolver clases particulares de problemas, Algunas áreas en las cuales existen toolboxes disponibles son:

- Teoría de control
- Tratamiento de señales
- Inteligencia artificial
- Control de procesos mecánicos, de aviación, automoción, etc.
- Mapeo y tratamiento de imágenes
- Instrumentación y adquisición de datos
- Identificación de sistemas

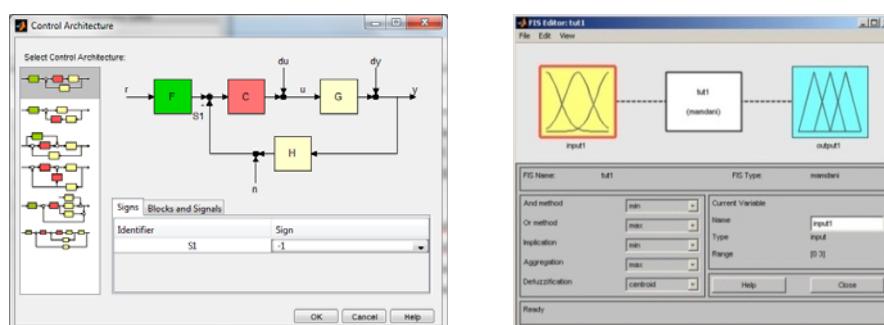


Fig.20. Izq. Toolbox Sisotool . Der. Toolbox Fuzzy Logic

B. SIMULINK

Interfaz gráfica para el modelado y simulación de sistemas

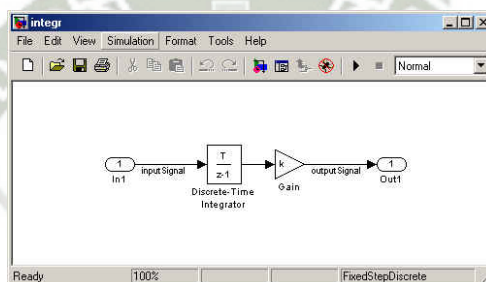


Fig.21. Interfaz Simulink

C. OTROS

Real Time Workshop, xPC Tarjet y desarrollo sobre DSPs y FPGAs

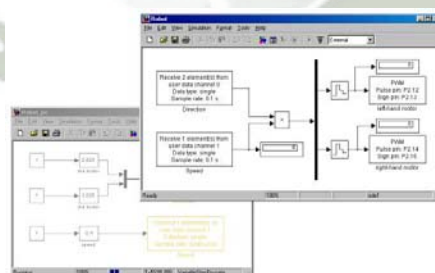


Fig.22. Real Time Workshop usando microcontrolador C167

2.4.1 IDENTIFICACION DE SISTEMAS – TOOLBOX IDENT

Para utilizar el interfaz gráfico de MATLAB para la identificación, se ejecuta el comando “ident”.

En primer lugar, se cargarán los datos del experimento mediante la opción “Import data >Time domain data” y rellenando el formulario para la importación de los datos. En este formulario, el botón “More” permitirá incluir información adicional como los nombres de las variables, unidades, etc. Pulsando el botón “Import”, los datos importados aparecerán en las “Data Views” y como datos iniciales de trabajo, “Working Data”, y datos de validación, “Validation Data”.

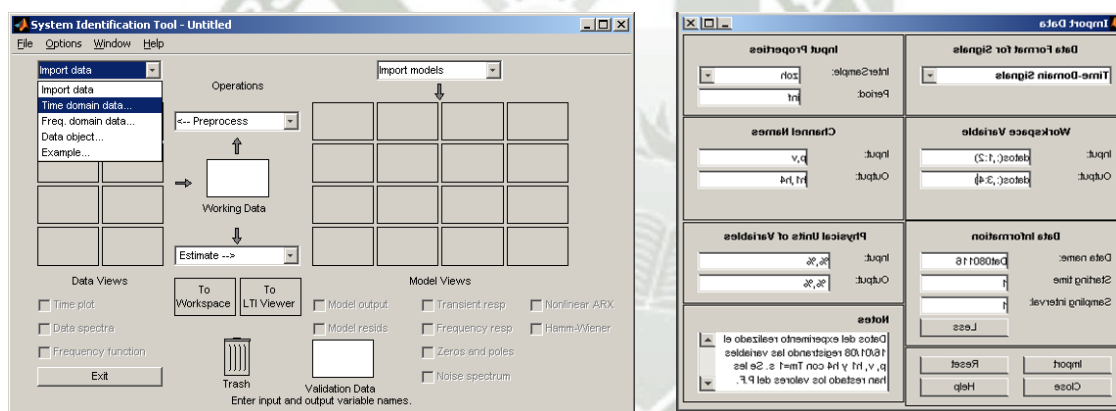


Fig.23 Inicialización Ident

El botón de selección “Time Plot” permite visualizar gráficamente la pareja entrada-salida que se seleccione en el menú “Chanel” de cualquiera de los datos que se tengan en las “Data views”.

Antes de utilizarlos para la estimación de modelos, se preprocesan los datos para seleccionar los datos de trabajo y de validación indicados anteriormente, mediante la opción “Preprocess->Select range”.

Una vez cubierto el formulario de la ventana “Select range” y seleccionados los datos numérica o gráficamente, se pulsa el botón “Insert”. Los nuevos conjuntos de datos aparecerán en las “Data Views” desde donde deberán ser arrastrados con el ratón a las correspondientes casillas de “Working Data” y “Validation Data”

En este punto ya se puede iniciar la estimación de modelos, utilizando la opción “Estimate->Quick start” que proporciona MATLAB o seleccionando la estructura del modelo que se quiere identificar. Por ejemplo, seleccionando “Parametric models”, podemos elegir entre ARX, ARMAX, “State Space”, etc. Una vez definidos los parámetros adecuados para el modelo, como el orden o el nombre que se le va a dar, se calcula el modelo se pulsa el botón “Estimate”.

La ventana “Model output” ofrecerá información de qué modelo se ajusta mejor a los datos de validación mediante un índice que será mejor cuanto mayor sea su valor. Este índice puede ser diferente para cada una de las salidas, que se seleccionan en el menú “Chanel”.

2.5 RED GSM

La red GSM³ es el sistema global para las comunicaciones móviles , es un sistema estándar, libre de regalías, de telefonía móvil digital.

Un cliente GSM puede conectarse a través de su teléfono con su computador y enviar y recibir mensajes por correo electrónico, faxes, navegar por Internet, acceder con seguridad a la red informática de una compañía (red

³ GSM : Del inglés Global System for Mobile communications

local/Intranet), así como utilizar otras funciones digitales de transmisión de datos, incluyendo el servicio de mensajes cortos (SMS) o mensajes de texto.

GSM se considera, por su velocidad de transmisión y otras características, un estándar de segunda generación (2G). Su extensión a 3G se denomina UMTS y difiere en su mayor velocidad de transmisión, el uso de una arquitectura de red ligeramente distinta y sobre todo en el empleo de diferentes protocolos de radio (W-CDMA).

Una red de radiotelefonía celular está conformada por una extensión de territorio cubierto por un conjunto de espacios llamados células y una serie de canales de radio repartidos entre dichas células.

En GSM se distinguen cuatro tipos diferentes de células, son las siguientes:

A. Macrocélulas (Macrocells): Son células de gran tamaño (amplia cobertura) utilizadas en áreas de terreno muy grandes donde la distancia entre áreas pobladas es grande.

B. Microcélulas (Microcells): Son células de escasa extensión de cobertura que se utilizan en áreas donde hay una gran densidad de población. A mayor número de células mayor número de canales disponibles, que pueden ser utilizados por más usuarios simultáneamente.

C. Células selectivas (Selectived Cells): En muchas ocasiones no interesa que una célula tenga una cobertura de 360 grados, sino que interesa que tenga un alcance y un radio de acción determinado. En estos casos es cuando se usan las células selectivas. El caso más típico de uso de células de

este tipo son aquellas que se disponen en las entradas de los túneles, en los cuales no tiene sentido que la célula tenga un radio de acción total (360 grados) sino un radio de acción que vaya a lo largo del túnel.

D. Células Sombrilla (Umbrella Cells): Este tipo de células se utilizan en aquellos casos en los que tenemos un elevado número de células de tamaño pequeño y continuamente se están produciendo cambios (handovers) del terminal de una célula a otra. Para evitar que suceda esto lo que hacemos es agrupar conjuntos de microcélulas, de modo que aumentamos la potencia de la nueva célula formada y podemos reducir el número de handovers que se producen.

Los abonados a la red celular son móviles y por tanto itinerantes. Para comunicar estaciones móviles entre sí es necesario preestablecer ciertas bases que ayuden a identificar y localizar a cada usuario, puesto que no siempre se van a encontrar en una misma célula. Para ello es necesario dotar a cada abonado de una tarjeta de identidad (SIM, Subscriber Identify Module) que permita identificar de forma unívoca a cada cliente de la red. Asimismo es necesario dotar a la red de un centro de autenticación de clientes (AUC, AUthentication Centre) para controlar la identidad de cada abonado una vez que enciende su terminal móvil, y de una base de datos llamada (HLR, Home Location Register) que contenga el registro de los abonados locales, actualizando dicha información dinámicamente.

2.5.1 MODEM GSM

El modem GSM es la parte que se comunicará con el mundo exterior. En la elección del modem, pese a ser lo más importante, no ha hecho falta tener en cuenta muchos parámetros, ya que todos los modem GSM cumplen los mismos requisitos mínimos necesarios.

El modem GSM elegido es del fabricante Wavecomm modelo FASTRACK SUPREME 20, dotado de interfaz RS-232, conector de antena tipo Jack JME, soporte de tarjeta SIM, LED indicador de estado y alimentación mediante un adaptador 220V AC – 12V DC. Se ha elegido este modelo porque en la universidad se dispone de él, conozco su funcionamiento y cumple con los requisitos necesarios para el desarrollo de este proyecto.

En cuanto a soporte tiene un foro en su página web que ayuda a solucionar cualquier inquietud. La marca ya tiene 15 años de experiencia en el mercado de módulos wireless CPU, siendo la pionera actualmente. Tiene más de 11 millones de módulos vendidos en el mundo.



Fig. 24 Módem GSM Wavecom Fastrack Supreme 20

- Banda Dual 900/1800 MHz GSM.
- Comunicación serie a 2.4, 4.8, 9.6 y 14.4 kbps.

- GSM Clase 1 y clase 2 Grupo 3 Fax.
- Short Message Services (SMS)
- Zócalo para tarjeta SIM.
- Compatible con los comandos AT.

Se deberá conectar una antena GSM a la salida del modem. En caso de no hacerlo no se garantizará conectarse a la red con la suficiente calidad.



Fig..25. Antena GSM conectada al modem

2.6 Visual C#

Microsoft Visual C# es un lenguaje de programación diseñado para crear una amplia gama de aplicaciones que se ejecutan en .NET Framework. C# es simple, eficaz, con seguridad de tipos y orientado a objetos. Con sus diversas innovaciones, C# permite desarrollar aplicaciones rápidamente y mantiene la expresividad y elegancia de los lenguajes de tipo C.

Visual Studio admite Visual C# con un editor de código completo, plantillas de proyecto, diseñadores, asistentes para código, un depurador eficaz y fácil de usar, además de otras herramientas. La biblioteca de clases .NET Framework

ofrece acceso a una amplia gama de servicios de sistema operativo y a otras clases útiles y adecuadamente diseñadas que aceleran el ciclo de desarrollo de manera significativa.

C# es un lenguaje de programación orientado a objetos. Tiene que definir clases para representar los tipos de su aplicación y, a continuación, crear los objetos como instancias de estas clases. Una clase puede contener campos, métodos, propiedades e indizadores.

2.6.1 INTERFAZ HOMBRE MAQUINA (HMI)

Un HMI⁴ o Interfaz Hombre - Máquina, como su nombre lo indica, tiene como función que el operario pueda supervisar y controlar su proceso desde una pantalla gráfica.

Fundamentalmente, un terminal sustituye a los antiguos paneles de control, que estaban contruidos principalmente por pulsadores, pilotos, indicadores, y dispositivos de medición.

Un terminal HMI, como parte de un sistema de automatización, puede conectarse a varios equipos al mismo tiempo.

2.6.1.1 FUNCIONES DEL HMI

A menudo, la palabra scada y HMI inducen cierta confusión, cierto es que todos los SCADA`s ofrecen una interfaz gráfica tipo HMI, pero no todos los sistemas de automatización que tienen HMI son SCADA. La diferencia radica

⁴ HMI : De sus siglas en inglés , Human Machine Interface.

en la función de supervisión que pueden realizar estos últimos a través del HMI. Las funciones son las siguientes:

2.6.1.1.1 MONITOREO

Habilidad de obtener y mostrar datos del proceso planta en tiempo real.

2.6.1.1.2 SUPERVISIÓN

Posibilidad de ajustar condiciones de trabajo de la planta o proceso directamente desde el computador además de monitoreo.

2.6.1.1.3 ALARMAS

Reconoce eventos excepcionales dentro del proceso y reportar dichos eventos. Estas alarmas son reportadas basándose en los límites de control preestablecidos.

2.6.1.1.4 CONTROL

Aplicar algoritmos (dentro de microcontrolador) que ajustan los valores del proceso y así mantener estos valores dentro de ciertos límites.

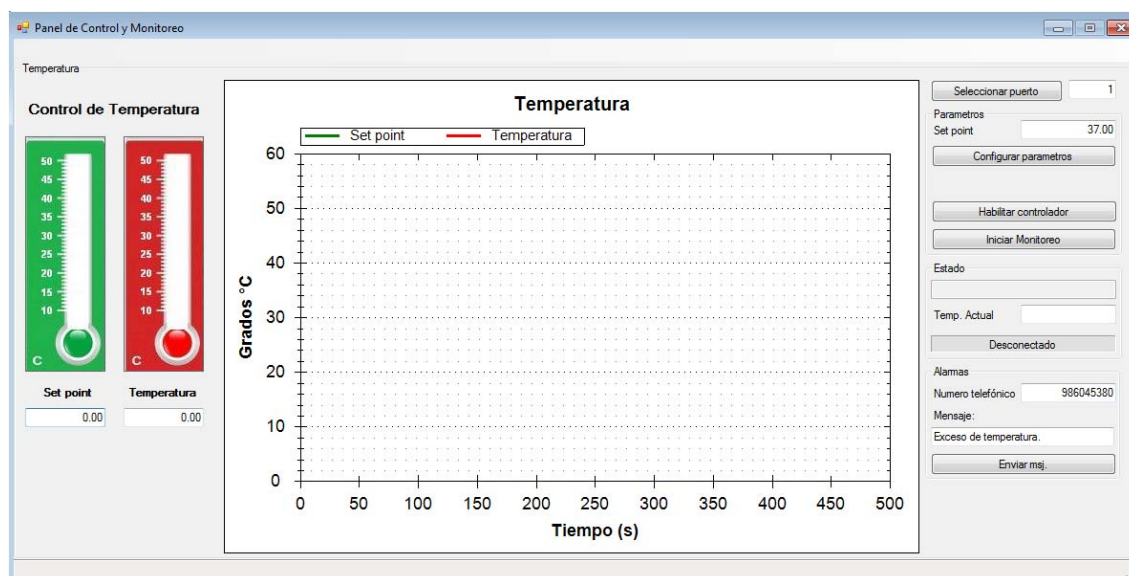


Fig.26. Interfaz gráfica para el control de temperatura del prototipo



CAPÍTULO III: DISEÑO, SIMULACIÓN Y CONSTRUCCIÓN DEL PROTOTIPO

CAPÍTULO III: DISEÑO, SIMULACIÓN Y CONSTRUCCIÓN DEL PROTOTIPO

3.1 CONSTRUCCIÓN DEL PROTOTIPO

El prototipo de incubadora consta del diseño y construcción de la cámara térmica, etapa de potencia, etapa de control y etapa de comunicación así como el diseño, implementación y sintonización del controlador PID.

3.1.1 CONSTRUCCIÓN DE CÁMARA TERMICA

La cámara térmica está construida de madera, se decidió usar madera porque es un material muy común, fácil de trabajar para lograr las dimensiones y formas necesarias, su conductividad térmica es relativamente baja, porque lo que las pérdidas de calor son mínimas cuando el grueso de las paredes es de 20 mm a más, es suficientemente rígida como para que la caja pueda ser manipulada con seguridad, y es fácil de pegar, atornillar y perforar. Además en uno de sus lados se confeccionará una compuerta la cual tendrá una ventana de acrílico para poder supervisar el crecimiento de los pollos. Esto es importante ya que durante el tiempo de incubación necesitaremos un cómodo acceso al interior a fin de realizar ciertas tareas de manipulación que se verá más adelante.

Las propiedades y características que presenta la madera se muestran en la Tabla 3.

PROPIEDADES FÍSICAS		CARACTERÍSTICAS
Nombre comercial		Madera
Peso específico (kg/dm ³)		1.56
Resistencia específica	Seco	2 - 3.5
	Húmedo	1.8 - 2.5
Temperatura de operación constante °C		40
Temperatura pico con tiempo no mayor a dos horas °C		120
Resistencia a la abrasión		Buena
Absorbencia (%)		8.3 - 10.5
Recuperación elástica		92
Elongación (%)	Seco	30
	Húmedo	30
Resistencia a la luz solar		Excelente
pH de operación normal		3 - 9

Tabla 3. Características de la madera

Además debemos de considerar de la madera:

- Material de fácil obtención.
- Fácil de mover y ensamblar.
- Aumenta su resistencia con el aumento del calibre o espesor.
- No requiere tecnología complicada para su uso
- Puede ser cortado y perforado con facilidad.
- Puede pigmentarse con colores sólidos.

La caja de madera sirve para simular un recinto de incubación artificial, ésta tiene las dimensiones de 0.40 m de largo * 0.35 m de ancho y 0.35 m de alto, va a estar operado en el rango de temperatura que comprende los 30°C a 50°C.

La cámara consta de una puerta, para manejar fácilmente la carga (huevos), una vez cerrado esta será lo más herméticamente posible, y además tendrá una ventana de acrílico para poder visualizar el comportamiento de nuestra planta.



Fig.27. Cámara térmica

Las lámparas incandescentes se ubican en la parte superior en ambas paredes, la boquilla cerámica permite la tolerancia de la alta temperatura, y se encuentra fijada con tornillos a la caja; los cables utilizados son 14 AWG THW de siete hilos, el aislante térmico de este cable soporta una temperatura de hasta 90 °C, el ventilador se ubicará en la parte inferior como lo muestra en la figura 29, éste es de 12V solo girará a 4V lo que nos permitirá un calor constante en todo el recinto y no enfriamiento, también en la base de la incubadora irá un recipiente pequeño con agua para simular la humedad.

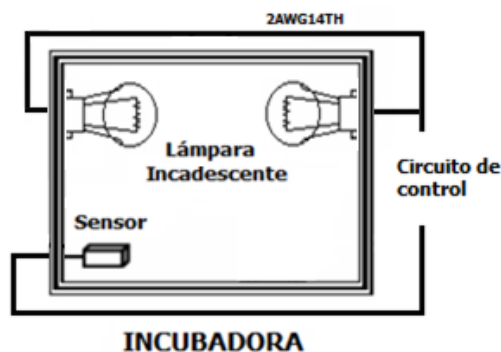


Fig. 28. Localización de lámparas incandescentes y sensor de la cámara térmica
(Vista Frontal)



Fig. 29. Localización de ventilador (Vista Lateral derecha)

El sensor es colocado a unos centímetros cerca al foco para que este tarde menos en percibir los cambios de temperatura. El sensor es el circuito integrado LM35 el rango de operación del sensor es suficiente para el rango de operación de nuestro prototipo como ya habíamos mencionado en líneas anteriores. Además este sensor nos da una salida de voltaje proporcional a la temperatura por lo que esta señal de salida de nuestro sensor será menos compleja adecuarla a nuestra circuitería.

El sensor se localiza en la parte inferior de la cámara de temperatura, figura 28. Este se encuentra atornillado con la parte interior de la cámara, debido a la forma que tiene el sensor. Los pines de éste se encuentran aislados con la utilización de un recubrimiento de acrílico permitiendo el

aislamiento del voltaje de polarización con la lámina de acero y del aumento de temperatura hacia los pines del sensor.

En la pared derecha de la cámara térmica (figura 32) se instaló un ventilador que tiene como objetivo principal mantener un flujo continuo de calor en toda el área de nuestra cámara térmica.

Los cables de alimentación y de la señal del sensor se encuentran aislados térmicamente y en el exterior de la cámara.

3.1.2. MODELO DE LA PLANTA A CONSIDERAR PARA LA IDENTIFICACIÓN

Para realizar el modelo matemático de la planta, al cual se le aplicará el proceso de identificación, se utilizará una simplificación de la planta real la cual se muestra en la figura 30.

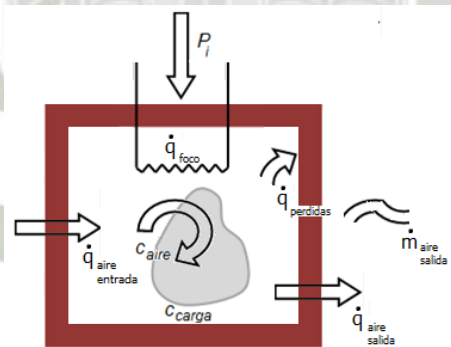


Fig. 30. Simplificación de la planta real para el modelado

Como se ha mencionado anteriormente, la temperatura dentro de la cámara se considerará uniforme, y la temperatura del exterior constante e igual a la temperatura ambiente.

3.1.3 FUNDAMENTOS TEÓRICOS

3.1.3.1 TRANSFERENCIA DE CALOR

Es un proceso por el que se intercambia energía en forma de calor entre distintos cuerpos, o entre diferentes partes de un mismo cuerpo que se encuentran a distintas temperaturas.

El calor se transfiere mediante convección, radiación o conducción. Aunque estos tres procesos pueden tener lugar simultáneamente, puede ocurrir que uno de los mecanismos predomine sobre los otros dos.

3.1.3.2 IDENTIFICACIÓN DE SISTEMAS

Se trata de un método experimental que permite obtener el modelo de un sistema a partir de datos reales recogidos de la planta bajo estudio (figura 31).

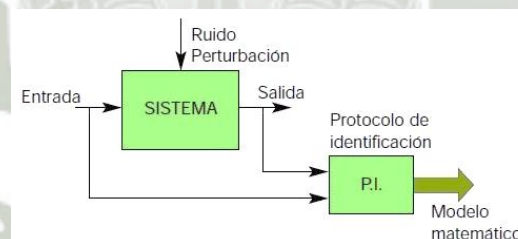


Fig. 31. Proceso de identificación

Identificación es la determinación, en base a la entrada y la salida, de un sistema, dentro de una clase de sistemas especificada, al cual el sistema probado es equivalente.

3.1.3.2.1 Identificación no paramétrica. Los métodos de identificación no paramétricos permiten obtener modelos o representaciones no paramétricas de la planta bajo estudio.

3.1.3.2.1.1 Identificación no paramétrica en el dominio el tiempo.

Mediante esta técnica de identificación se pretende obtener la respuesta al impulso del sistema, o bien la respuesta al escalón del mismo y para ello, debe registrarse la evolución temporal de la salida del sistema tras la aplicación de una señal impulso o escalón.

Obviamente, la imposibilidad de conseguir este tipo de señales en la práctica lleva a utilizar un método indirecto para obtener la respuesta impulsiva, conocido como análisis de la correlación.

3.1.3.2.1.2 Identificación no paramétrica en el dominio de la frecuencia.

En este caso, el modelo resultante es una representación de la respuesta en frecuencia del sistema, obtenida mediante la aplicación de señales de entrada sinusoidales de distintas frecuencias.

Cuando no sea posible aplicar este tipo de entradas, puede recurrirse a la aplicación de un ruido blanco, que permite obtener la respuesta en frecuencia mediante el conocido análisis espectral. Este análisis se basa en la realización de la transformada de Fourier de las funciones de covarianza de la entrada, la salida y la correlación entre la entrada y la salida.

3.1.3.2.2 Identificación paramétrica. Los modelos paramétricos, a diferencia de los anteriores, quedan descritos mediante una estructura y un número finito de parámetros que relacionan las señales de interés del sistema (entradas, salida y perturbaciones).

3.1.4 MODELO MATEMÁTICO DE LA PLANTA

Para describir procesos físicos, se recurre a la conservación de la energía. La condición de equilibrio de los sistemas térmicos establece que el calor administrado a un sistema es igual al calor almacenado por el sistema más el calor liberado por el sistema y más las pérdidas del sistema.

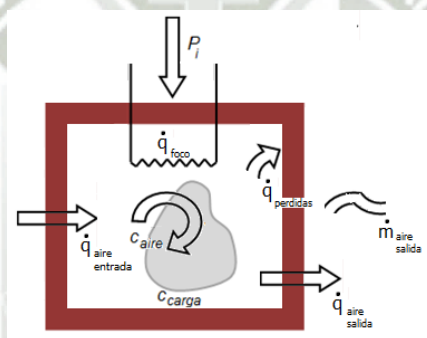


Fig. 32. Modelamiento de la cámara térmica

Por conservación de energía

$$\Delta q_{cam} = q_{entrada} - q_{salida} - q_{perdidas}$$

$$q_{entrada} = q_{aire\ entrada} + q_{foco}$$

Donde:

Δq_{cam} : Calor absorbido por el aire dentro de la camara incubadora

$q_{aire\ entrada}$: Calor del aire que ingresa a la camara incubadora

q_{foco} : Calor entregado porl foco

$q_{aire\ salida}$: Calor del aire que sale de la camara incubadora

$q_{perdidas}$: Calor perdido a traves de las paredes y el techo de la incubadora

$$\Delta q_{cam} = q_{aire\ entrada} + q_{foco} - q_{aire\ salida} - q_{perdidas}$$

Derivando la expresión:

$$\Delta \dot{q}_{cam} = \dot{q}_{aire\ entrada} - \dot{q}_{aire\ salida} + \dot{q}_{foco} - \dot{q}_{perdidas}$$

donde:

i) $\Delta q_{cam} = Ce\ m_{cam}(T - T_o)$

T : Temperatura del aire dentro de la cámara

T_o : Temperatura inicial del aire dentro de la cámara

m_{cam} : masa del aire dentro de la cámara

Ce : Calor especifico del aire

$$\Delta \dot{q}_{cam} = Ce\ m_{cam} \frac{d}{dt}(T - T_o)$$

ii) $\dot{q}_{aire\ entrada} = Ce\ \dot{m}\ (T_{aire\ entrada} - T_{REF})$

$\dot{m}$: flujo masico del aire circulante

$T_{aire\ entrada}$: Temperatura del aire que ingresa a la camara incubadora

T_{REF} : Temperatura del aire para una entalpia igual a cero

De forma análoga:

$$\dot{q}_{aire\ salida} = Ce\ \dot{m}\ (T_{aire\ salida} - T_{REF})$$

iii) $\dot{q}_{foco} = K * Pot$

Pot : Potencia eléctrica que circula por el foco

K : Factor de conversion de energia electrica en calorifica ($K = 0.95$)

$$C_e m_{cam} \frac{d}{dt} (T - T_o) =$$

$$C_e \dot{m} (T_{aire entrada} - T_{REF}) - C_e \dot{m} (T_{aire salida} - T_{REF}) + K * Pot - \frac{(T - T_{amb})}{R}$$

$$C_e m_{cam} \frac{d}{dt} (T - T_o) =$$

$$C_e \dot{m} (T_{aire entrada} - T_{aire salida}) + K * Pot - \frac{(T - T_{amb})}{R}$$

Donde :

$$T_{aire entrada} = T_o$$

$$T_{aire salida} = T$$

Continuando con la resolución :

$$C_e m_{cam} \frac{d}{dt} (T - T_o) =$$

$$C_e \dot{m} (T - T_o) - \frac{(T - T_o)}{R} + K * Pot$$

Despejando : $K * Pot$

$$K * Pot =$$

$$C_e m_{cam} \frac{d}{dt} (T - T_o) + C_e \dot{m} (T - T_o) + \frac{(T - T_o)}{R}$$

$$K * Pot = C_e m_{cam} \frac{d}{dt} (T - T_o) + \left(C_e \dot{m} + \frac{1}{R} \right) (T - T_o)$$

$$K * Pot(t) = Ce m_{cam} \frac{d}{dt}(\Delta T)(t) + \left(Ce \dot{m} + \frac{1}{R} \right) (\Delta T)(t)$$

Aplicando Laplace:

$$K * Pot(s) = Ce m_{cam} s(\Delta T)(s) + \left(Ce \dot{m} + \frac{1}{R} \right) (\Delta T)(s)$$

$$K * Pot(s) = [Ce m_{cam} s + \left(Ce \dot{m} + \frac{1}{R} \right)] (\Delta T)(s)$$

$$\frac{\Delta T(s)}{Pot(s)} = \frac{K}{Ce m_{cam} s + Ce \dot{m} + \frac{1}{R}}$$

$$\frac{\Delta T(s)}{Pot(s)} = \frac{\frac{K}{Ce \dot{m} + \frac{1}{R}}}{\frac{Ce m_{cam}}{Ce \dot{m} + \frac{1}{R}} * s + 1}$$

Considerando $R \rightarrow \infty$ (Caso general)

$$\frac{\Delta T(s)}{Pot(s)} = \frac{\frac{K}{Ce \dot{m}}}{\left(\frac{m_{cam}}{\dot{m}} \right) * s + 1}$$

3.1.5 MODELO MATEMÁTICO DEL ACTUADOR

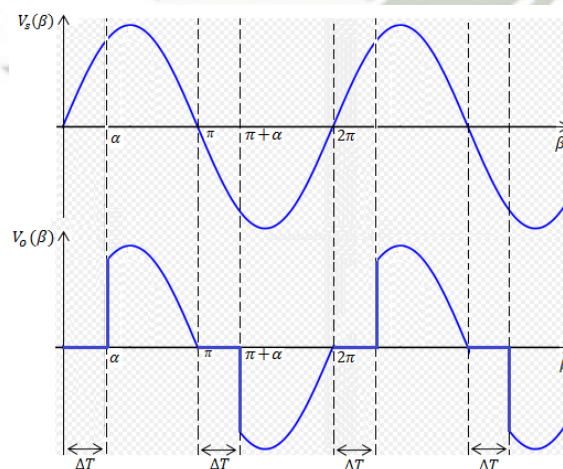


Fig. 33. Modelamiento del Actuador

V_S : Voltaje de Alimentacion de 220 V_{AC}

V_o : Voltaje de Salida de la Etapa de Potencia

V_{RMS} : Voltaje Eficaz de Salida de la Etapa de Potencia

V_{RMS_S} = Voltaje Eficaz de Alimentacion

$$V_{RMS(\alpha)}^2 = \frac{1}{2\pi} \int V_{o(\beta)}^2 d\beta$$

$$V_{o(\beta)} = \begin{cases} 0 & ; 0 < \beta < \alpha \\ \sqrt{2} V_{RMS(s)} \sin\beta & ; \alpha < \beta < \pi \\ 0 & ; \pi < \beta < \pi + \alpha \\ \sqrt{2} V_{RMS(s)} \sin\beta & ; \pi + \alpha < \beta < 2\pi \end{cases}$$

$$V_{RMS(\alpha)}^2 = \frac{1}{2\pi} \left[\int_{\alpha}^{\pi} 2V_{RMS_S}^2 \sin^2\beta d\beta + \int_{\pi+\alpha}^{2\pi} 2V_{RMS_S}^2 \sin^2\beta d\beta \right]$$

$$V_{RMS(\alpha)}^2 = \frac{2V_{RMS_S}^2}{2\pi} \left[\int_{\alpha}^{\pi} \sin^2\beta d\beta + \int_{\pi+\alpha}^{2\pi} \sin^2\beta d\beta \right]$$

$$V_{RMS(\alpha)}^2 = \frac{V_{RMS_S}^2}{\pi} \left[\int_{\alpha}^{\pi} \sin^2\beta d\beta + \int_{\pi+\alpha}^{2\pi} \sin^2\beta d\beta \right]$$

$$I_1 = \int_{\alpha}^{\pi} \sin^2\beta d\beta \quad I_2 = \int_{\pi+\alpha}^{2\pi} \sin^2\beta d\beta$$

Por propiedades trigonométricas $\frac{1-\cos 2\beta}{2} = \sin^2\beta$

$$I_1 = \int_{\alpha}^{\pi} \frac{1-\cos 2\beta}{2} d\beta$$

$$I_1 = \int_{\alpha}^{\pi} \frac{d\beta}{2} - \int_{\alpha}^{\pi} \frac{\cos 2\beta}{2} d\beta$$

$$I_1 = \int_{\alpha}^{\pi} \frac{d\beta}{2} - \int_{\alpha}^{\pi} \frac{\cos 2\beta}{4} d2\beta$$

$$I_1 = \frac{\beta}{2} \Big|_{\alpha}^{\pi} - \frac{\sin 2\beta}{4} \Big|_{\alpha}^{\pi}$$

$$I_1 = \frac{\pi - \alpha}{2} - \left(\frac{\sin 2\pi - \sin 2\alpha}{4} \right)$$

$$I_1 = \frac{\pi - \alpha}{2} + \frac{\sin 2\alpha}{4}$$

$$I_2 = \int_{\pi+\alpha}^{2\pi} \sin^2 \beta \, d\beta$$

$$I_2 = \int_{\pi+\alpha}^{2\pi} \frac{1 - \cos 2\beta}{2} \, d\beta$$

$$I_2 = \int_{\pi+\alpha}^{2\pi} \frac{d\beta}{2} - \int_{\pi+\alpha}^{2\pi} \frac{\cos 2\beta}{2} \, d\beta$$

$$I_2 = \int_{\pi+\alpha}^{2\pi} \frac{d\beta}{2} - \int_{\pi+\alpha}^{2\pi} \frac{\cos 2\beta}{4} \, d2\beta$$

$$I_2 = \frac{\beta}{2} \Big|_{\pi+\alpha}^{2\pi} - \frac{\sin 2\beta}{4} \Big|_{\pi+\alpha}^{2\pi}$$

$$I_2 = \frac{\pi - \alpha}{2} - \left(\frac{\sin 4\pi - \sin(2\pi + 2\alpha)}{4} \right)$$

$$I_2 = \frac{\pi - \alpha}{2} + \frac{\sin 2\alpha}{4}$$

$$V_{RMS(\alpha)}^2 = \frac{V_{RMS(s)}^2}{\pi} [I_1 + I_2]$$

$$V_{RMS(\alpha)}^2 = \frac{V_{RMS(s)}^2}{\pi} 2 * \left[\frac{\pi - \alpha}{2} + \frac{\sin 2\alpha}{4} \right]$$

$$V_{RMS(\alpha)} = V_{RMS(s)} * \sqrt{\frac{\pi - \alpha}{\pi} + \frac{\sin 2\alpha}{2\pi}}$$

Para linealizar el modelo y simplificar el diseño del controlador, usamos la linealización de Taylor

$$V_{RMS(\alpha)} = V_{RMS P.O.} + \frac{\partial}{\partial \alpha} V_{RMS P.O.} (\alpha - \alpha_{P.O.})$$

Hallamos la derivada de $V_{RMS(\alpha)}$

$$\frac{\partial}{\partial \alpha} V_{RMS(\alpha)} = \frac{V_{RMS(s)} * \left(-\frac{1}{\pi} + \frac{\cos 2\pi}{\pi}\right)}{2 * \sqrt{\frac{\pi - \alpha}{\pi} + \frac{\sin 2\alpha}{2\pi}}}$$

$$\frac{\partial}{\partial \alpha} V_{RMS(\alpha)} = \frac{-\frac{1}{\pi} V_{RMS(s)} (1 - \cos 2\alpha)}{2 * \sqrt{\frac{\pi - \alpha}{\pi} + \frac{\sin 2\alpha}{2\pi}}}$$

$$\frac{\partial}{\partial \alpha} V_{RMS(\alpha)} = \frac{-\frac{1}{\pi} V_{RMS(s)} * \sin^2 \alpha}{\sqrt{\frac{\pi - \alpha}{\pi} + \frac{\sin 2\alpha}{2\pi}}}$$

$$\frac{\partial}{\partial \alpha} V_{RMS(\alpha)} = \frac{-\frac{1}{\pi} V_{RMS(s)}^2 * \sin^2 \alpha}{V_{RMS(s)} \sqrt{\frac{\pi - \alpha}{\pi} + \frac{\sin 2\alpha}{2\pi}}}$$

$$\frac{\partial}{\partial \alpha} V_{RMS(\alpha)} = \frac{-\frac{1}{\pi} V_{RMS(s)}^2 * \sin^2 \alpha}{V_{RMS(\alpha)}}$$

$$\frac{\partial}{\partial \alpha} V_{RMS(\alpha)} = V_{RMS(\alpha)_{P.O.}} - \frac{1}{\pi} V_{RMS(s)}^2 * \sin^2(\alpha_{P.O.}) * (\alpha - \alpha_{P.O.})$$

Donde:

$V_{RMS(\alpha)_{P.O.}}$: Voltaje Eficaz de la salida de Potencia para un punto de operacion determinado

$\alpha_{P.O.}$: Angulo de disparo correspondiente al Punto de Operacion

Por lo tanto:

$$V_{RMS(\alpha)} = \left[V_{RMS(\alpha)_{P.O.}} + \frac{\frac{1}{\pi} V_{RMS(s)}^2 * \sin^2(\alpha_{P.O.}) * (\alpha_o.)}{V_{RMS(\alpha)_{P.O.}}} \right] - \left[\frac{\frac{1}{\pi} V_{RMS(s)}^2 * \sin^2(\alpha_{P.O.}) * \alpha}{V_{RMS(\alpha)_{P.O.}}} \right]$$

3.1.6 HERRAMIENTA PARA LA IDENTIFICACIÓN DEL SISTEMA

El proceso de identificación se simplifica utilizando una herramienta implementada en Matlab mediante la ayuda del módulo System Identification (ident). Ident cuenta con una serie de funciones para realizar la siguiente secuencia de operaciones:

- Carga archivo de datos.
- Organiza y filtra los datos, removiendo los valores medios, tendencias y filtrando el ruido. Adicionalmente separa los datos para la identificación y validación
- Realiza el análisis de correlación y espectral.
- Aplica método predictor de error, mediante un modelo paramétrico lineal.
- Realiza proceso de validación comparando gráficamente los resultados. Adicionalmente entrega la respuesta al escalón, al impulso , respuesta a la frecuencia .

3.1.7 DISEÑO DE LOS CONTROLADORES

Al momento de diseñar un sistema de control es indispensable conocer y entender la dinámica de la planta. Una de las etapas transcendentales en el diseño del control es la identificación de la planta, porque con ésta el modelo se tendrá el comportamiento exacto del proceso. Para modelar un sistema de control se debe describir por medio de ecuaciones matemáticas la

relación de la variable de salida, variable que se desea controlar, con la entrada del sistema.

La identificación de sistemas consiste en tratar al sistema como una caja negra y excitarlo con varias señales de entrada y observar el comportamiento del mismo, es decir, las salidas producidas por las múltiples entradas de prueba.

El trabajo que se presenta a continuación es la identificación de sistema del proceso para regular la temperatura del prototipo de incubadora, en el que se busca una incubabilidad⁵ alta.

Se usa el modelamiento matemático (con leyes de la termodinámica, fórmulas de electricidad, etc.), además se dispone de la cámara térmica para hacer la adquisición de datos necesarios (temperatura, voltaje, etc.), las cuales nos ayuda en la identificación de nuestra planta y su comportamiento.

Una vez obtenida la identificación idónea de la planta, se continua con el diseño de un controlador para mejorar su rendimiento, teniendo presente al momento de implementarlo las características y limitaciones reales del sistema.

El control de temperatura en la incubación es muy importante, ya que de éste depende el porcentaje de incubabilidad, el proceso que se va a identificar es el que se lleva a cabo dentro de la cámara, para que así el embrión desarrollo

⁵ Incubabilidad : Es la capacidad de un huevo fértil de producir un pollito vivo y apto, y se mide en porcentajes que están dados por los huevos fértiles cargados en la incubadora y los pollitos nacidos.

y no muera durante el transcurso, lo que se busca es la relación que existe entre la temperatura en su interior y la potencia ingresada a la cámara.

3.1.8 DISEÑO Y SELECCIÓN DEL SISTEMA DE CALENTAMIENTO

Los dispositivos de calentamiento hacen referencia a todos los equipos, aparatos y componentes que permiten mantener la temperatura al interior del cuarto de incubación, dentro de rango de 37.3 °C y 37.7°C, durante el periodo de 21 días que tarda el proceso de incubación.

Es importante aclarar que para lograr mantener constante la temperatura dentro de un rango tan estrecho, es indispensable implantar un control continuo de temperatura el cual garantice mantener las condiciones ambientales en el rango óptimo.

En el cuadro siguiente se enlistan cada uno de los componentes para el calentamiento, así como una breve descripción del mismo.

	Dispositivo	Elemento	Descripción
Calentamiento	Generador de Calor	Lámparas incandescentes	200 W
	Ventilador	Cooler PC	28 m ³ /hr
	Control de temperatura	Controlador	Control PID
		Sensor electrónico	LM35

Tabla 4. Componentes del sistema de control de temperatura

Para el diseño del dispositivo de calentamiento se deben cumplir unos requerimientos mínimos para garantizar la excelente operación del equipo.

Dicho requerimientos se mencionan como:

- Mantener las condiciones ambientales uniformes y constantes al interior del equipo, de temperatura durante todo el tiempo de incubación.
- Suministrar la energía necesaria para satisfacer las necesidades de los embriones durante su crecimiento.

Para la selección, disposición y puesta a punto del sistema de calentamiento es necesario estudiar, todas las causas que generan consumo de energía dentro del equipo, con el fin cuantificarlas, reducirlas y seleccionar los elementos acordes con esas necesidades.

A continuación se ilustra el proceso de diseño y selección de los diversos componentes que incluyen el dispositivo de calentamiento al interior de la incubadora.

3.1.8.1 SELECCIÓN DEL SISTEMA DE CALENTAMIENTO

Para el diseño del sistema de calentamiento se deben tener presentes todas las necesidades energéticas del equipo en operación, así como las pérdidas de energía inmersas en el proceso.

Dichas necesidades energéticas se enumeran:

- Calentar el aire al interior del cuarto desde la temperatura atmosférica ambiental que oscila en los 20 °C hasta la temperatura óptima de incubación de 37,5 °C en un tiempo aproximado de 12 minutos.
- Suministrar la energía requerida por cada huevo para su desarrollo embrionario.
- Restaurar la energía perdida durante el proceso,
- Energía perdida a través de las paredes
- Energía perdida cuando se abren las puertas.

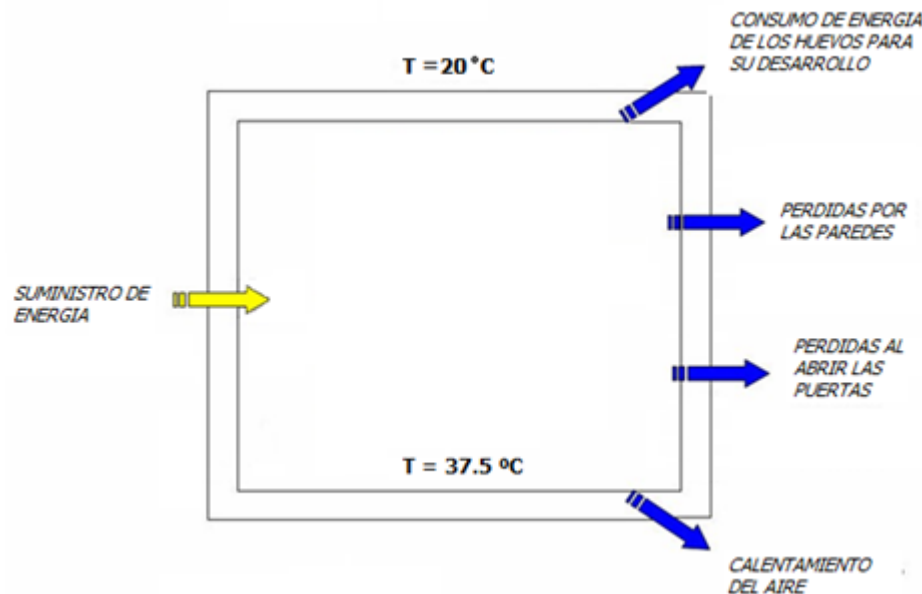


Fig. 34. Energía requerida en el proceso

A continuación se procederá a calcular cada uno de los factores de carga térmica descritos anteriormente.

3.1.8.1.1 ENERGÍA REQUERIDA PARA EL CALENTAMIENTO DEL AIRE EN LA INCUBADORA

- Las dimensiones del cuarto son 40 cm. x 40 cm. x 35 cm.

- Volumen = $40 \times 40 \times 35 = 56.000 \text{ cm}^3 = 0.0560 \text{ m}^3$ de aire en la cámara
- Densidad del aire = 1 Kg/m^3
- Masa total de aire = 0.0560 Kg. de aire en la cámara.

La temperatura inicial del aire se asume como $T_i = 16^\circ\text{C}$, lo que es promedio en la ciudad de Arequipa, pero por efectos de cálculo se toma este valor.

La temperatura final ya está establecida siendo de $T_2 = 37.5^\circ\text{C}$ Haciendo un balance de energía al interior de la cámara:

$$Q = m * (T_2 - T_1) * C_v$$

El calor específico a volumen constante para el aire es de $C_v = 0.718 \text{ KJ/Kg. K}$

Reemplazando valores se tiene $Q = 864.472 \text{ J}$ esta es la energía total que debe suministrarse para lograr condiciones estables.

Además de acuerdo con la operación de la incubadora y el propósito para el cual se destina, el calentamiento no debe ser tan brusco, ya que el choque térmico sobre los huevos puede dañarlos, razón por la cual el tiempo total de calentamiento debe ser alrededor de un cuarto de hora.

Para los cálculos se toma este tiempo de 1 minutos, con el objetivo de tener un cierto rango de confianza sobre los cálculos realizados.

- Tiempo total = 60s
- Energía total = 864.472 J
- Potencia a disipar = $(864.472\text{J} / 60\text{s}) = 14.40 \text{ W}$ en el estado transitorio

CALCULO DE LA ENERGÍA REQUERIDA POR LOS HUEVOS. Para el cálculo de la energía requerida por los huevos para el desarrollo del embrión, se realiza un balance de energía, por medio de la primera ley de la termodinámica.

$$Q = m * (T_2 - T_1) * C_v$$

- La masa se toma como el peso del total de los huevos dentro del equipo, en total 80, cada huevo pesa 70 gr, para un total de 5.6 Kg.
- La temperatura inicial es de 16°C y la temperatura final de 37.5 °C (después de que se ha llegado al estado estable)
- Para el cálculo del C_v (calor específico a volumen constante) del huevo lo aproximamos el C_p (calor específico a presión constante) del agua, de acuerdo los siguientes datos.
- La cáscara del huevo cuya masa es aproximadamente el 11% del total del peso del huevo, su porcentaje de agua es de 5%, para un contenido total de 0.55% de agua; la clara del huevo cuya masa es aproximadamente 58% del peso del huevo, su porcentaje de agua es de 88%, para un contenido total de 51% de agua; la yema de huevo cuya masa es aproximadamente 31% del peso del huevo, su porcentaje de agua es de 68%, para un contenido total de 21% de agua. De acuerdo a estos datos el porcentaje de agua en un huevo es de 72.55% del peso total del huevo. Lo cual permite aproximar un

huevo a las mismas características del agua, solo para efectos de cálculo.

- De acuerdo a esto, el $Cv_{\text{huevo}} = Cp_{\text{agua}} = 4.18 \text{ KJ/Kg. } ^\circ\text{C}$ La energía requerida será de 503.272 KJ de energía.

Además es importante resaltar que el tiempo para que los huevos alcancen la temperatura de $37.5 ^\circ\text{C}$ debe ser lo suficientemente grande para evitar los efectos nocivos de un cambio brusco en temperatura, por lo tanto el tiempo de establecimiento se acepta de 1 – 2 horas hasta alcanzar el estado estable.

- Potencia requerida = $(503272 \text{ J}/3600\text{s}) = 139.79\text{W}$

Calculo de la energía de pérdidas. El cálculo de la energía de pérdidas se realizará en el orden descrito anteriormente.

Energía perdida a través de las paredes. Como la temperatura al interior de la cámara es superior a la ambiental, es importante colocar aislante térmico y así minimizar las pérdidas de energía; en este caso se colocó espuma de poliuretano como aislante., este un material de excelentes propiedades físicas y químicas: excelente aislante térmico, impermeable a la humedad, dimensionalmente estable, resistente a la propagación del fuego, larga vida útil y muy liviano.

Para calcular el espesor de aislamiento se realizó el siguiente procedimiento.

Temperatura ambiente externa (T_1) = $16 ^\circ\text{C}$

Temperatura ambiente interna (T_2) = $37.5 ^\circ\text{C}$

Dimensiones del equipo = 40 x 40 x 35 cm.

Área de la cara de mayor tamaño (A) = 40 x 40 = 0.16 m²

De acuerdo al montaje que se desea realizar y las condiciones ambientales nombradas tenemos la siguiente figura:

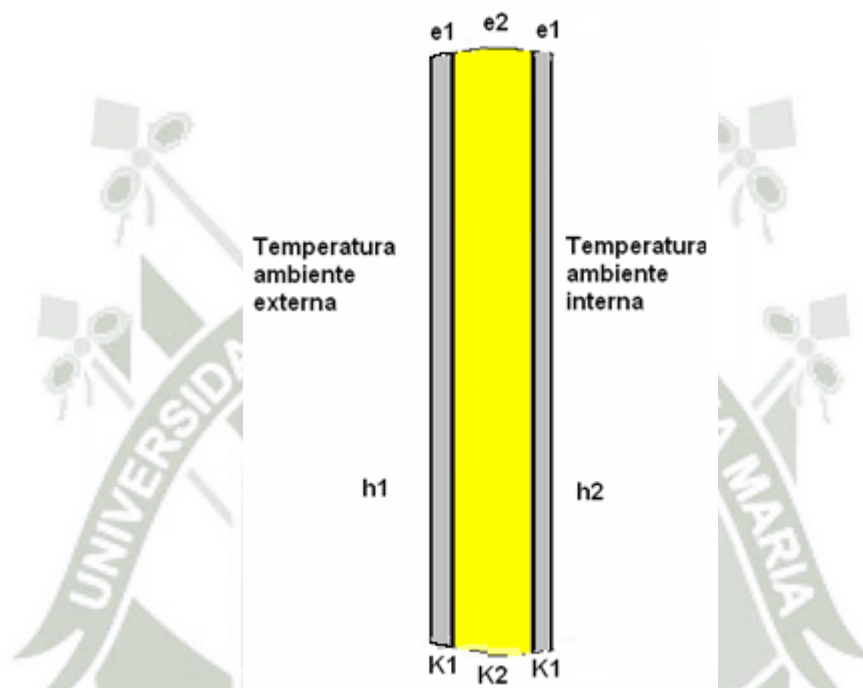


Fig.35 Pared de incubadora

K1 = Conductividad térmica madera

K2 = Conductividad térmica espuma poliuretano

h1 = Coeficiente convección ambiente exterior

h2 = Coeficiente convección ambiente interno

e1 = Espesor madera

e2 = Espesor de aislamiento

La ecuación que relaciona todas estas variables con el calor disipado y el espesor de aislamiento es la siguiente:

$$Q = \frac{T_2 - T_1}{R_{eq}}$$

Donde:

Q = Calor disipado

R_{eq} = Resistencia equivalente al flujo de energía.

En este caso la R_{eq} incluye la resistencia por convección en ambas caras y la resistencia por conducción en las tres paredes.

Por tanto la ecuación se escribe así:

$$Q = \frac{T_2 - T_1}{\frac{1}{h_1 * A} + \frac{e_1}{K_1 * A} + \frac{e_2}{K_2 * A} + \frac{e_1}{K_1 * A} + \frac{1}{h_2 * A}}$$

Definiendo los valores de las variables, tenemos:

- T_1 = Temperatura ambiente externa = 16°C
- T_2 = Temperatura ambiente interna = 37.5°C
- h_1 = Coeficiente convección externo = 10 W/m²°C
- h_2 = Coeficiente convección interno = 20 W/m²°C
- e_1 = Espesor madera = 10 mm = 0.01 m
- K_1 = Conductividad térmica madera = 13 W/mK
- K_2 = Conductividad térmica espuma poliuretano = 0.021 W/mk
- A = Área de una cara lateral = 0.16 m²

Nota: Los valores de h_1 y h_2 se han obtenido de los cuadros, para superficies expuestas al aire a baja velocidad. Las conductividades térmicas de los materiales se encuentran en los libros de transferencia de calor.

De acuerdo a estos valores se necesita encontrar el espesor de aislamiento óptimo con el fin de disminuir la pérdida de energía confrontando esto con el precio del aislante.

Cuando se coloca un aislante térmico de 2 cm, las pérdidas de calor es 9.36W.

Energía perdidas por paredes = 9.36 W

Energía perdida cuando se abren las puertas. Para el cálculo de la energía perdida cuando las puertas del equipo se abrían, fue necesaria la experimentación con el objetivo de lograr datos más confiables.

Durante una de las pruebas de calentamiento hechas en el prototipo, la temperatura se mantenga en estado estable a 37.5 °C como valor de referencia para la temperatura. En este punto se desconectó las lámparas y la puerta fue abierta completamente, los datos obtenidos son:

- El tiempo transcurrido para que la temperatura disminuyera desde el punto de referencia 37.5 °C hasta 30 °C fue de 50 – 60 s en promedio para las mediciones.
- El tiempo transcurrido para que la temperatura disminuyera desde los 30 °C hasta la temperatura ambiente de 16 °C fue de 1.4 – 1.5 minutos en promedio para las mediciones

De acuerdo a esto la energía que debe el sistema reponer, se calcula como:

Energía total perdida en la incubadora hasta el establecimiento de la temperatura con la atmosférica $Q = 864.472 \text{ J}$

El tiempo para que esto ocurriera fue de 50 s, la energía a disipar por la lámpara durante este periodo, es

- Tiempo total = 50s
- Energía total = 864.472 J
- Potencia perdida cuando se abren las puertas es de $(864.472 \text{ J}/50\text{s}) = 17 \text{ W}$

Potencia disipar al abrir puertas = 17 W

Selección de las lámparas incandescentes Las lámparas incandescentes como fuente de energía para el calentamiento de la cámara de incubación, suministra la energía en forma de calor al sistema de acuerdo a la corriente que a través de ella está circulando, con el fin de mantener la temperatura en el valor deseado.

Las lámparas incandescentes como lo mencionamos antes se montan en las paredes de la cámara en sockets y se aprieta por medio de tornillos para una buena fijación.

De acuerdo a esto, las lámparas incandescentes a utilizar deben poder disipar la potencia total requerida por el equipo, calculada anteriormente; esta potencia es:

CARACTERÍSTICAS	POTENCIA
Potencia requerida para calentar el aire de la incubadora	14.40 W
Potencia requerida por los huevos	139.7 W
Potencia perdida a través de la paredes	9.36 W
Potencia perdida cuando se abren las puertas	17 W
	180 W

Tabla 5. Energía requerida en el proceso

La potencia es de 200W para nuestro sistema , así entonces se selecciona dos focos de 100W de la marca Philips.

El sistema de control se encarga de administrar la cantidad de energía liberada dependiendo del estado en que se encuentre, así:

- Estado transitorio: de acuerdo al tiempo programado para lograr los 37.5°C partiendo desde 20 °C
- Estado estable: manteniendo los 37.5°C durante los 21 días. De acuerdo a estos cálculos la capacidad instalada será

Potencia = 200W

Selección del ventilador. El ventilador es un elemento indispensable para el calentamiento del aire al interior del equipo, ya que este renueva las capas de aire adyacentes a la lámpara, permite que todo el aire confinado en el equipo se caliente de forma uniforme evitando puntos fríos.

El ventilador en su operación normal envía aire hacia delante, succionándolo de la parte posterior; el aire de salida del ventilador fluye a través de la

lámpara incandescente, calentándose y distribuyéndose a través de todo el equipo.

Dentro de la gama de ventiladores se escogió un ventilador para PC, de dimensiones de 80 mm. x 80 mm. x 25 mm de 12V , del cual se usará solo 4V para mantener el calor equitativo en toda el área dentro de la cámara.

3.1.8.1.2 CIRCUITOS EMPLEADOS EN PLACA ELECTRONICA DE CONTROL

CIRCUITO DE CRUCE POR CERO

Como lo que se pretende es controlar la potencia suministrada a la lámpara incandescente, esto se hará controlando el paso de corriente un tiempo determinado en cada ciclo de la onda, es decir, controlamos los ángulos de disparos de la onda de $220V_{RMS}$, para esto utilizamos la señal PWM que controla la onda.

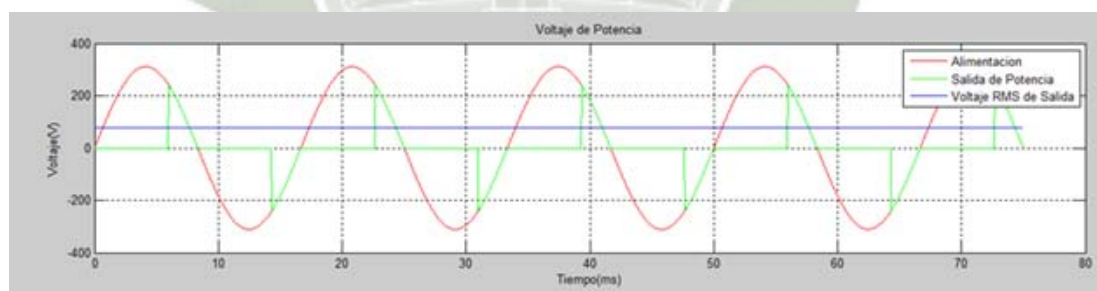


Fig. 36. Simulación de Cruce por cero de la planta

Este circuito se encarga de indicar al microcontrolador en que instante la señal de alimentación AC de 220 V cruza por cero. Esto con el fin que el microcontrolador espere el tiempo suficiente para generar el flanco de disparo para la etapa de potencia. Este circuito podría también haberse implementado

con un Opam, pero tiene como desventaja de no ofrecer un aislamiento eléctrico el cual pone en riesgo a la placa de control.

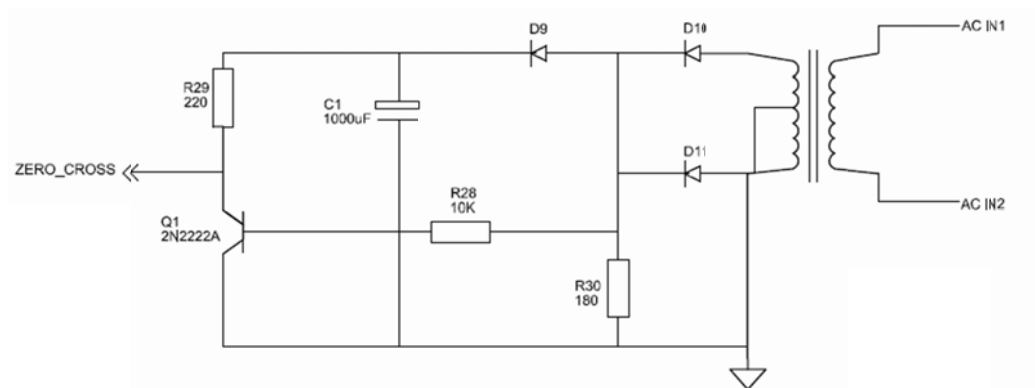


Fig. 37. Circuito de cruce por cero

CIRCUITO DE POTENCIA

Se encarga de la de la conversión de la señal proveniente del microcontrolador para realizar el control de fase de la señal de voltaje alterno de 220 V(vendría ser básicamente un convertidor DC-AC).Se debe resaltar que la señal del microcontrolador no se conecta directamente al optotriac ya que la corriente que requiere es demasiada alta (alrededor de 20 mA), a comparación de la corriente del ATmega (4mA). Los valores de las resistencias y el optotriac se obtuvieron a partir del datasheet del MOC 3021

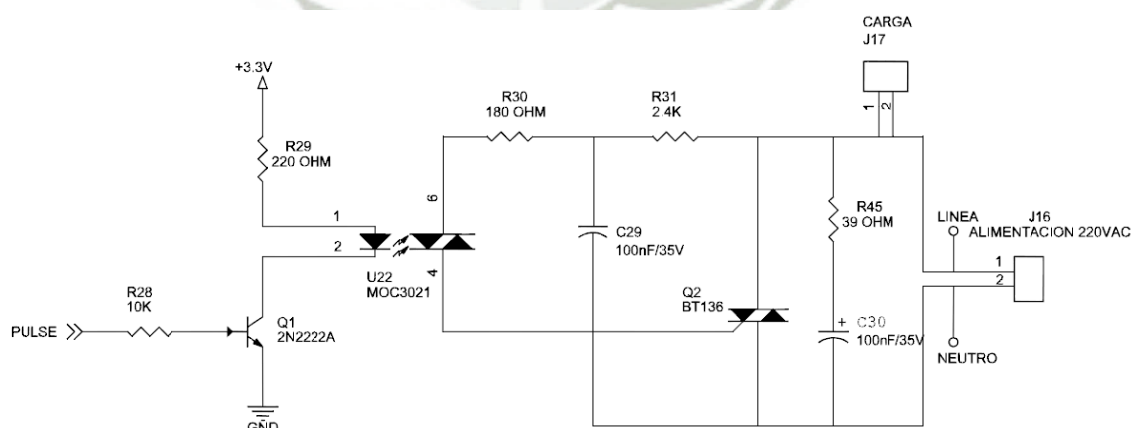


Fig. 38. Circuito de potencia

CIRCUITO SENSADO DE TEMPERATURA

Este circuito se divide en tres partes. La primera esta compuesta por el LM35 que se encarga de hacer la conversión de temperatura a voltaje con una ganancia de $0.01^{\circ}\text{C}/\text{mV}$, la segunda etapa es la del filtrado compuesto por un circuito R-C en serie y en bypass (los valores de estos se obtuvieron a partir del datasheet del LM35). La tercera y última etapa viene la etapa de amplificación compuesta por el LM358 que se encarga de elevar el nivel de voltaje que sale del filtro con una ganancia, además cuenta con un potenciómetro de referencia para poder eliminar cualquier offset que estuviera presente.

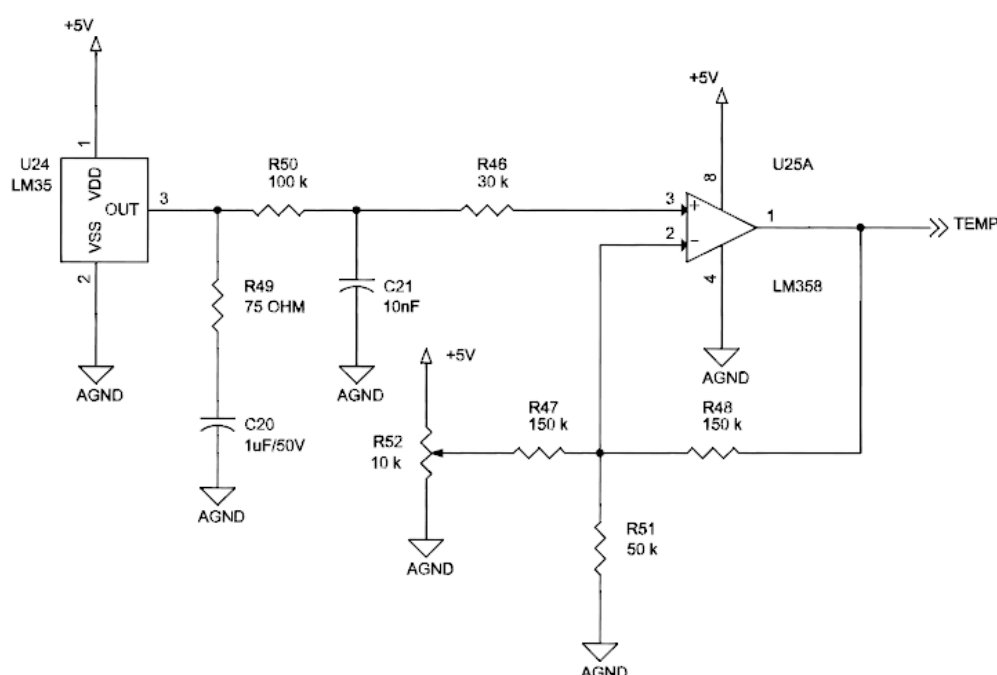


Fig. 39. Circuito sensado de temperatura

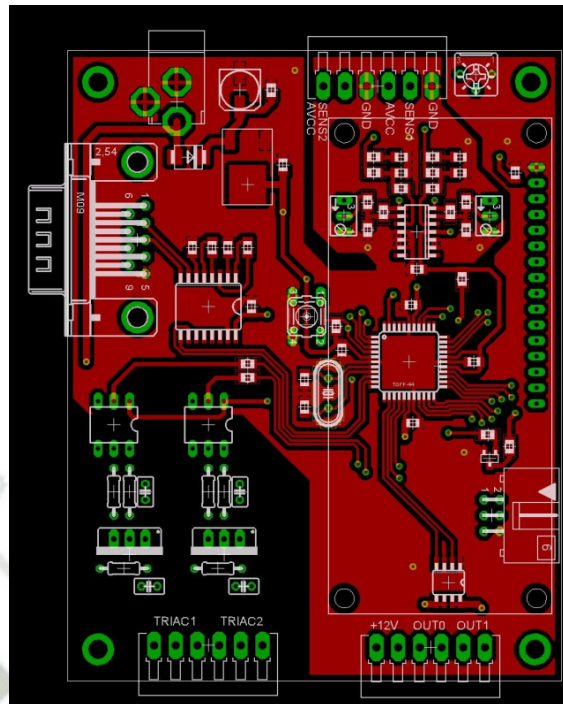


Fig. 40. Diseño de Tarjeta PCB

3.1.9 FUNCIÓN DE TRANSFERENCIA

Una función de transferencia es un modelo matemático que a través de un cociente relaciona la respuesta de un sistema (modelada) a una señal de entrada o excitación (también modelada). Se usan las funciones de transferencia para caracterizar las relaciones de entrada y salida de un sistema.

En el prototipo de incubadora se tomara en cuenta las siguientes funciones de transferencia:

- Función de transferencia del actuador.
- Función de transferencia de la planta
- Función de transferencia del sensor.

3.1.9.1 FUNCIÓN DE TRANSFERENCIA DEL ACTUADOR

El actuador diseñado está conformado por dos circuitos: el circuito de disparo y el circuito de potencia.

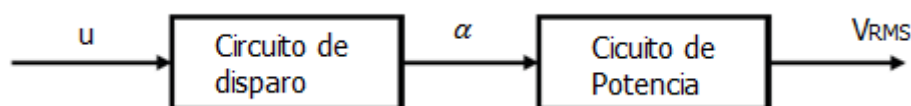


Fig.41. Diagrama de bloques del actuador

De la figura deducimos que:

$$G_{actuador}(s) = G_{c\ disp} * G_{c\ pot}$$

Donde:

$G_{c\ disp}$: Función de transferencia del circuito de disparo

$G_{c\ pot}$: Función de transferencia del circuito de potencia

3.1.9.1.1 FUNCIÓN DE TRANSFERENCIA DEL CIRCUITO DE DISPARO

El ángulo de disparo para el convertidor AC - AC (control de fase directo), la variación del ángulo de disparo, es de: $\alpha: 0^{\circ} - 180^{\circ}$. La respuesta es inmediata.

3.1.9.1.2 FUNCIÓN DE TRANSFERENCIA DEL CIRCUITO DE POTENCIA

El diagrama de bloques del actuador, el circuito de potencia, tiene como señal de entrada el ángulo de disparo (α), de activación del transistor, obteniendo como la señal de salida el voltaje RMS aplicado a la lámpara.

La función de transferencia del circuito de potencia se la puede aproximar a una función de primer orden.

$$G_{c\ pot} = \frac{K_{c\ pot}}{1 + \tau_{c\ pot} s}$$

En el conversor AC – AC, control de fase directo, la forma de onda de voltaje aplicado a la lámpara se muestra en la figura 2.1 y la expresión que relaciona el voltaje RMS y el ángulo de disparo es la siguiente:

$$V_{rms} = V_{max} \sqrt{\frac{\pi - \alpha}{2\pi} + \frac{\sin 2\alpha}{4\pi}}$$

El siguiente programa muestra el comportamiento del circuito eficaz con el ángulo de desfase.

```
% Programa para la simulación del comportamiento del voltaje eficaz
% de salida del circuito de potencia, en función del ángulo de desfase
% el voltaje es medido en voltios y el ángulo de desfase en radianes
% En este programa no se tiene que cargar ningún dato ya que los datos
% se van a generar desde la simulación
% Colocar todo los programas en C:\User\PC\Documents
clear all
close all
clc
% Simulamos los diferentes angulos de disparo que con los que se puede
% trabajar con la etapa de potencia.
% Para este programa, se considera el rango completo de 0 a 180°
alpha = 0:0.001:1;
alpha = pi*alpha;
vref_s = 220; % Voltaje de la alimentación local
% Ecuación que relaciona el voltaje eficaz de salida de la etapa de
% potencia con el angulo de desfase que entra al circuito
vref = vref_s*sqrt((pi-alpha)/pi+sin(2*alpha)/(2*pi));
% Graficamos los resultados de la simulación
plot(180*alpha/pi,vref,'r')
xlabel('angulo de desfase(°)')
ylabel('voltaje eficaz de salida de la etapa de potencia(V)')
title('Comportamiento de la Etapa de Potencia')
```

```
grid on
hold on
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Prototipo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

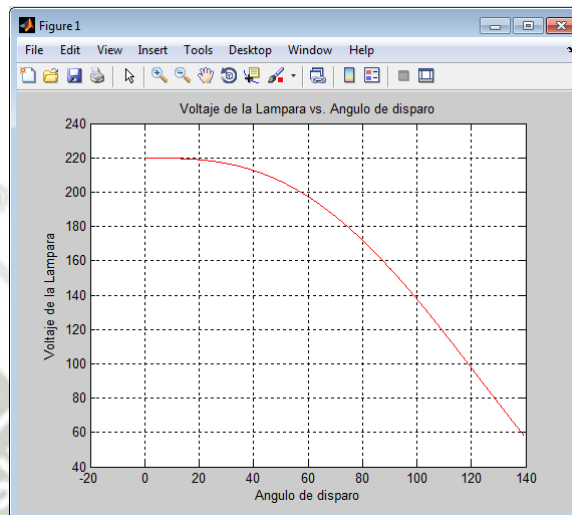


Fig. 42. Comportamiento de Voltaje eficaz de salida del circuito de potencia

En la curva de la figura 41 se toma en cuenta la zona lineal para obtener el valor de la ganancia de la función de transferencia.

Debido a que los circuitos conversores son de característica no lineal, existe una zona donde el comportamiento de la función es lineal, por lo que se realiza la liberalización de la curva aproximando su comportamiento.

La zona de linealidad de la ecuación comprende desde $\alpha_1 = 80^\circ$ hasta $\alpha_2 = 140^\circ$ cuyos valores correspondientes de voltaje RMS son: $V_{RMS1} = 170\text{ V}$ y $V_{RMS2} = 60\text{ V}$.

Aplicamos la herramienta de matlab cftool para la aproximación de la pendiente, la cual sería nuestra ganancia.

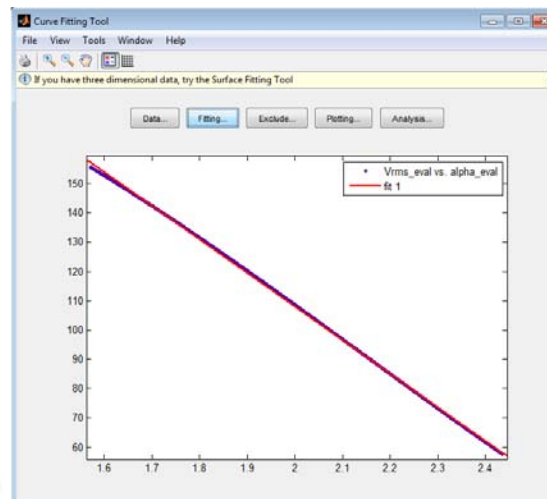


Fig. 43. Uso de cftool para linealización

Obtenemos los siguientes resultados:

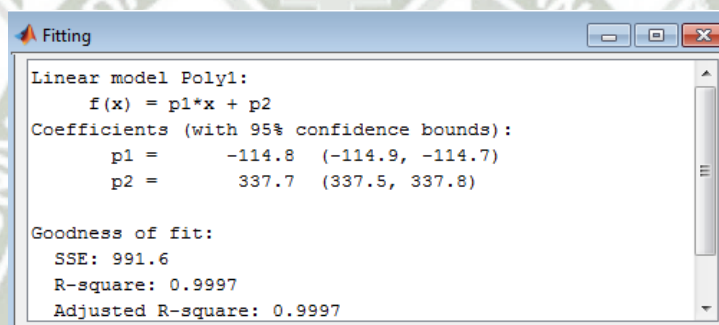


Fig. 44. Resultado de la Ganancia Lineal

Donde :

$p1 = -114.8$ es la ganancia de nuestra función de transferencia.

La expresión está representada bajo condiciones ideales, despreciando la caída de voltaje en el elemento de potencia en conducción y despreciando posibles variaciones de amplitud de voltaje en la red eléctrica.

$\tau_{C\ pot}$: Es la constante de tiempo de la respuesta del voltaje distorsionado al ángulo de disparo que controla el TRIAC. El tiempo de respuesta del

conversor depende íntegramente de la frecuencia de operación del circuito de disparo. Se toma como el valor promedio de la variación del ángulo de disparo para la transformación a segundos se toma el valor del periodo circuito de disparo $T_r = 8.33 \text{ ms}$.

$$\tau_{c \text{ pot}} = \frac{0 + 8.33 \text{ ms}}{2}$$

$$\tau_{c \text{ pot}} = 4.16 \text{ ms}$$

La función de transferencia del circuito de potencia es:

$$G_{c \text{ pot}} = \frac{-114.8}{1 + 4.16 \times 10^{-3} \text{ s}}$$

Modelamos la etapa de potencia como un sistema de primer orden teniendo como entrada el ángulo de disparo; esto, con el fin de simplificar el modelo y, de esa forma, facilitar el diseño del controlador. Sin embargo, en la implementación real, este ángulo de disparo viene a estar dado por una señal cuadrada periódica, con frecuencia igual a 120Hz, con un flanco de subida que se genera después de un intervalo de tiempo generado por el microcontrolador y la salida de la etapa de potencia viene a estar dada como una fracción de la señal de alimentación, la cual es proporcional al tiempo de disparo de la señal cuadrada antes mencionada.

El siguiente programa nos permite ver el comportamiento real de la etapa de potencia.

```
% Programa para graficar el voltaje de salida de la etapa de potencia
%(no solo su voltaje eficaz) para poder observar el comportamiento
% real de la etapa de potencia. De igual forma, se muestra el voltaje
```

```
% eficaz correspondiente a dicha señal para ser comparado con los
% datos experimentales obtenidos
clear all
close all
clc
% Ingresamos los parámetros para la simulacion
Vrms_s = 220; % Voltaje Eficaz de la Alimentacion
Ts = 0.001; % Tiempo de Muestreo, esto para asegurar que la
% actualización del tiempo del flanco de disparo se actualice
% solo despues del cruce por cero
t = 0:Ts:75;
t = t';
n = length(t);
Vrms = 0;
f = 60;
k = 2*pi*f/1000;
beta = k*t;
Vs = sqrt(2)*Vrms_s*sin(beta);
Vo = zeros(n,1);
Vc = zeros(n,1);
N = ceil(t(n)/(2*pi));
% Ingresamos la entrada a la etapa de potencia proveniente del dspic
delta_T = 2;%ms
alpha = k*delta_T;%rad
for i=0:N-1
    Vo = Vo+((beta>=2*pi*i+alpha &
beta<=2*pi*i+pi)+(beta>=2*pi*i+pi+alpha & beta<=2*pi*i+2*pi));
    Vc = Vc+((beta>=2*pi*i+alpha &
beta<=2*pi*i+alpha+0.5*k)+(beta>=2*pi*i+pi+alpha &
beta<=2*pi*i+pi+alpha+0.5*k));
end
Vo = Vo.*Vs;
Vc = 3.33*Vc; % Valor de voltaje permitido por dsPIC : 3.3V
clear i
delta_beta = k*Ts;
for i=1:n
    Vrms = Vrms+ (Vo(i)^2)*delta_beta;
end
Vrms = Vrms/beta(n);
Vrms = sqrt(Vrms);
subplot(211)
plot(t,Vc, 'k')
grid on
xlabel('Tiempo(ms)')
ylabel('Voltaje(V)')
title('Voltaje de Salida del uC')
subplot(212)
```

```
plot(t,Vs, 'r')
grid on
hold on
plot(t,Vo, 'g')
plot(t,Vrms*ones(n,1), 'b')
xlabel('Tiempo(ms)')
ylabel('Voltaje(V)')
title('Voltaje de Potencia')
legend('Alimentacion', 'Salida de Potencia', 'Voltaje RMS de Salida')
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Protoripo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

Mostramos los resultados para diferentes tiempos de disparo. Con un primer ángulo de disparo, observamos su comportamiento:

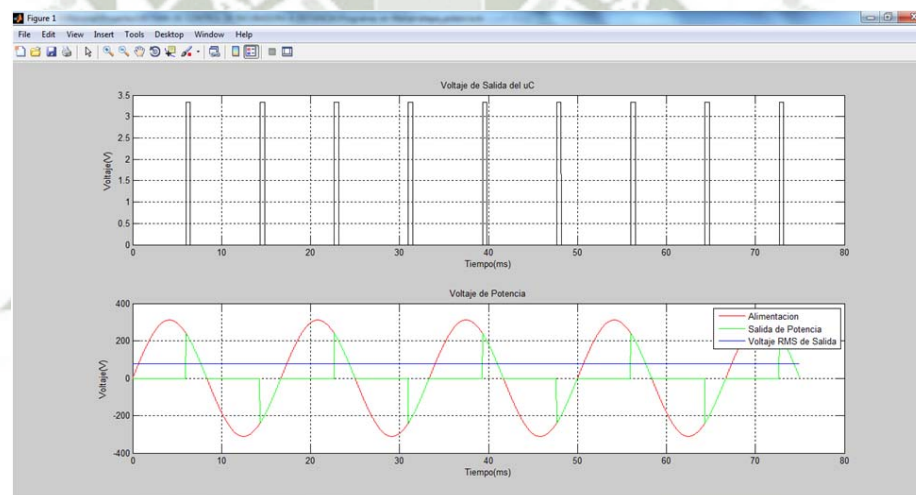


Fig. 45. Simulación de Etapa de Potencia con un primer ángulo de disparo

Con un segundo ángulo de disparo, observamos su comportamiento:

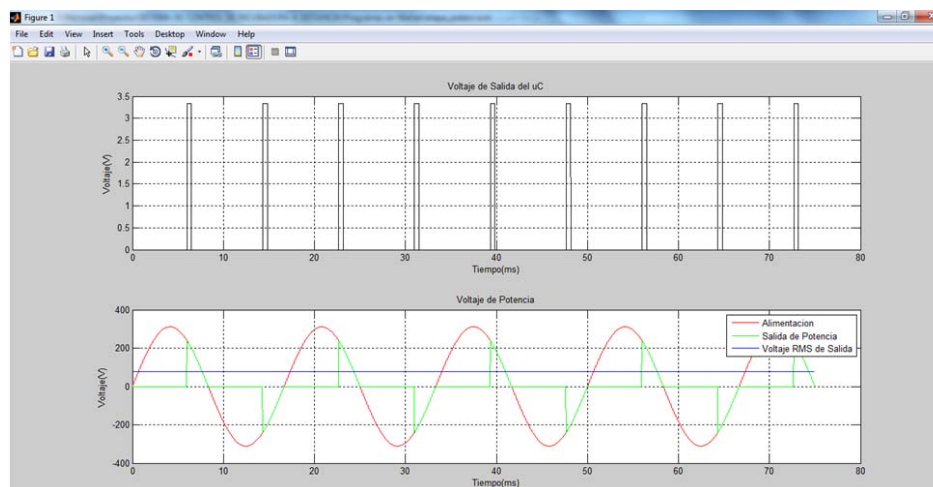


Fig. 46. Simulación de Etapa de Potencia con un segundo ángulo de disparo

Por último un tercer ángulo de disparo

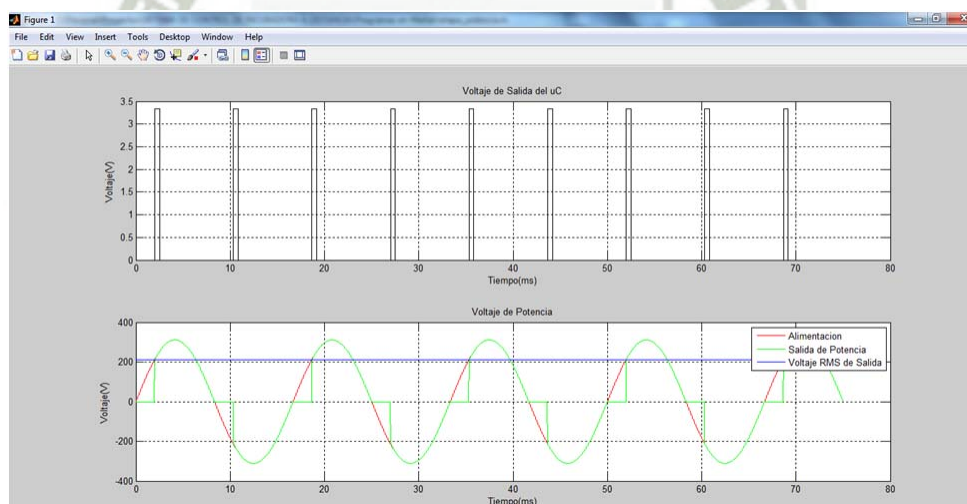


Fig. 47. Simulación de Etapa de Potencia con un tercer ángulo de disparo

De las gráficas, se tiene lo siguiente:

- La curva de color negro es la señal cuadrada que sale del microcontrolador

- La curva de color rojo es el voltaje de alimentación que entra a la etapa de potencia
- La curva de color verde es el voltaje de salida de la etapa de potencia y la curva de color azul es su voltaje eficaz correspondiente.

Entonces la función de transferencia del actuador es:

$$G_{actuador}(s) = 1 * \frac{-114.9}{1 + 4.16 \times 10^{-3} s}$$

$$G_{actuador}(s) = \frac{-114.9}{1 + 4.16 \times 10^{-3} s}$$

% El siguiente programa nos ayuda al modelamiento de la etapa de
% potencia, los tiempos de reacción del MOC3021, se puede
% modelar a esta parte como un amplificador

```
num = -114.9;  
den = 1;  
Gpot = tf(num,den);
```

Para tener un modelamiento más exacto de todo nuestro proceso, se debe de considerar las perturbaciones que afectan a nuestro proceso.

% Asimismo, se considera la función de transferencia residual como
% producto de la liberalización de Taylor (aunque no se haya usado
% explícitamente, se debe tener en cuenta que este modelo es solo
% lineal para un tramo)

```
num = 338;  
den = 1;  
Gpert_pot = tf(num,den);
```

3.1.9.2 FUNCIÓN DE TRANSFERENCIA DE LA PLANTA

La planta está conformada por la lámpara incandescente, la cual convierte la energía eléctrica en energía lumínica y energía calórica; produce cambio de temperatura conforme existe cambio de voltaje RMS aplicado.

La función de transferencia puede ser aproximada a una función de transferencia de primer orden:

$$G_{planta} = \frac{K_{pl}}{1 + \tau_{pl}s}$$

K_{pl} : Ganancia de la planta

τ_{pl} : Constante de tiempo de la planta

Para la identificación de transferencia de nuestra planta utilizaremos la herramienta de matlab ident (identificación de sistemas), el cual nos simplificará el procedimiento de identificación de la función de transferencia de la cámara térmica.

Cargamos los datos del tiempo, voltaje eficaz y temperatura, luego creamos una data object, que es el tipo de objeto con que trabaja el toolbox Ident.

```
load data_tiempo_Vrms_Temp.mat
tiempo = data_tiempo_Vrms_Temp(:,1);
Vrms = data_tiempo_Vrms_Temp(:,2);
Temp = data_tiempo_Vrms_Temp(:,3);
Ts = tiempo(2) - tiempo(1);
data = iddata(Temp,Vrms,Ts); %iddata: Crea un data object
ident
Opening System Identification Tool ..... done.
```

Se abre la ventana siguiente:

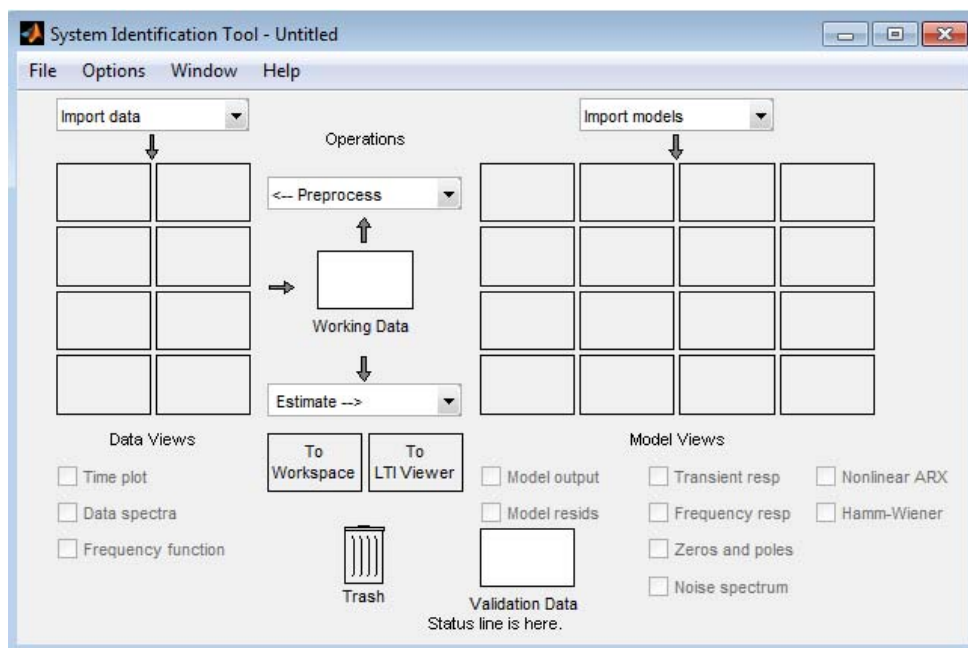


Fig. 48. Ventana de inicio de Toolbox Ident

Cargamos nuestra data que está en función del tiempo, Vrms y temperatura e identificamos el proceso.

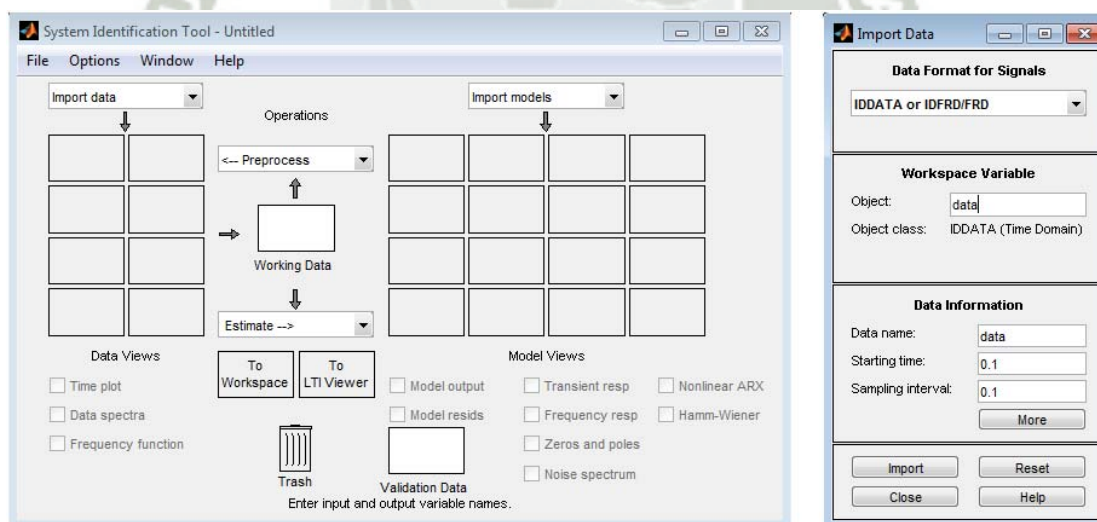


Fig. 49. Ident Importando el data Object para calcular el modelo

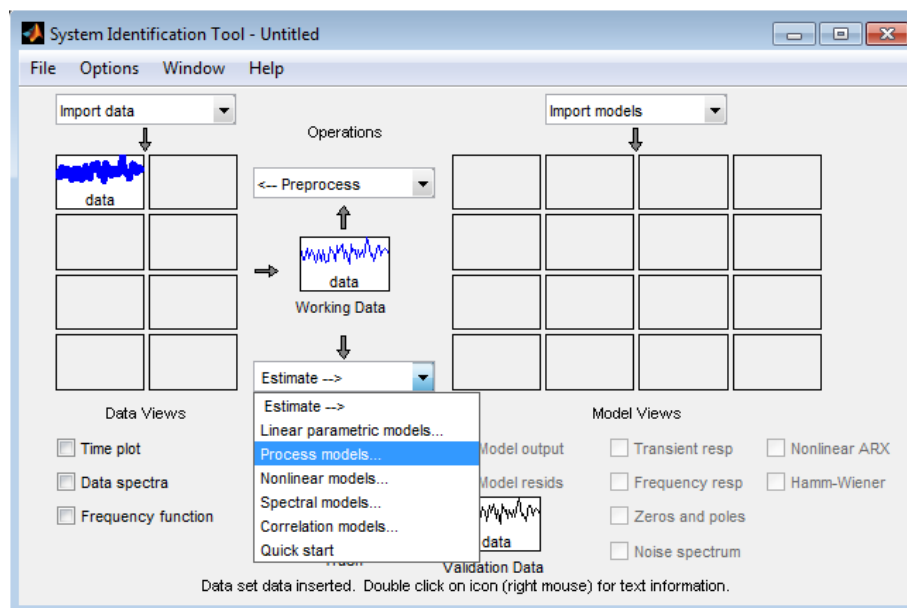


Fig. 50. Ident calculando el Modelo del Proceso

Colocamos las condiciones como queramos que estime la función de transferencia y estimamos

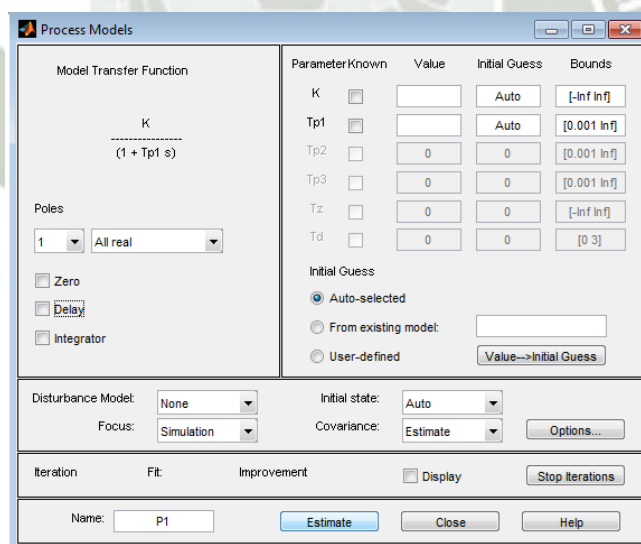


Fig. 51. Ident .Condiciones iniciales para cálculo de Función de Transferencia

La aproximación de la función de transferencia de la planta tiene una precisión de 93.02%.

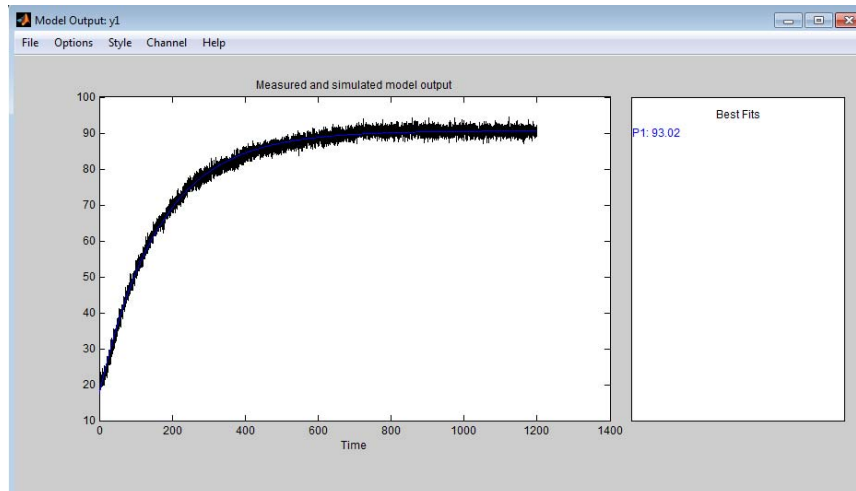


Fig. 52. Aproximación de la Planta con Ident

El toolbox ident identifica automáticamente la función de transferencia de nuestra planta.

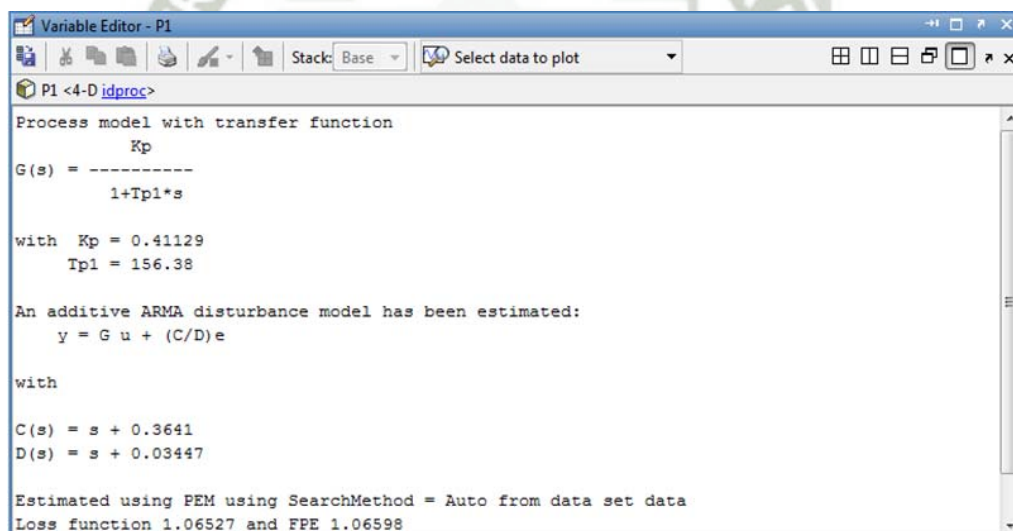


Fig. 53. Valores de Función de Transferencia

La función de transferencia de la planta es :

$$G_{Planta\ Total} = G_{planta} + G_{pert}$$

$$G_{planta} = \frac{0.41129}{1 + 156.38 * s}$$

Para un modelamiento más real se debe considerar la función de transferencia de la planta sumando su perturbación, esto se hace con la representación de espacio estados para ser más preciso ya que tiene condición inicial (Temperatura Ambiente).

La representación de espacio estados es la siguiente:

$$G_{planta\ total} = \frac{K_p}{\tau_p s + 1} * V(s) + G_{pert} * U(s)$$

$$G_{planta} = \frac{K_p}{\tau_p s + 1} ==> \frac{K_p}{\tau_p s + 1} = \frac{T(s)}{V(s)}$$

$$==> K_p * V(s) = \tau_p s * T(s) + T(s)$$

Aplicando la Transformada Inversa de Fourier

$$K_p * V(t) = \tau_p * \frac{d}{dt} T(t) + T(t)$$

$$\frac{d}{dt} T(t) = -\frac{1}{\tau_p} * T(t) + \frac{K_p}{\tau_p} * V(t)$$

$$Y = T(t)$$

Si tomamos $X_1 = T(t)$

$$\frac{d}{dt} X_1(t) = -\frac{1}{\tau_p} * X_1(t) + \frac{K_p}{\tau_p} * V(t)$$

$$Y = X_1(t) + 0 * V(t)$$

$$A = -\frac{1}{\tau_p} \quad B = \frac{K_p}{\tau_p} \quad C = 1 \quad D = 0$$

Con estos datos procedo a realizar el modelamiento de la planta

```
% En el modelamiento de la camara se utiliza la representación en
% espacio-estado para este apartado por depender de
% condiciones iniciales (temperatura ambiente)
kp = 0.41129;
tp = 156.38;
A = -1/tp;
B = kp/tp;
C = 1;
D = 0;
Gcam = ss(A,B,C,D);
% Asimismo, se considera la funcion de transferencia residual como
% producto de la linealizacion de taylor (aunque no se haya usado
% explicitamente, se debe tener en cuenta que este modelo es solo
% lineal por tramos)
num = [1 0.3649];
den = [1 0.03486];
Gpert_cam = tf(num,den);
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Prototipo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

3.1.9.3 FUNCIÓN DE TRANSFERENCIA DEL SENSOR

El modelamiento del sensor de temperatura, incluye 3 etapas, la primera consiste en la conversión de Temperatura a mV (con ganancia 0.01 según datasheet), la segunda consiste en un filtro de primer orden (para eliminar ruidos indeseados), la última que consiste en un amplificador de señal.

El siguiente programa nos muestra el comportamiento de nuestro filtro

```
% Programa para la simulacion del filtro de primer orden para el
% sensor lm35; este filtro se determino a partir de las
% recomendaciones del datasheet del integrado se esta colocando
% un filtro bypass para asegurar un mejor desempeño del sensor
clear all
```

```
close all
clc
% Ingresamos los valores de las resistencias y de los condensadores
% del filtro
R1 = 75;
C1 = 1e-6;
R2 = 100e3;
C2 = 0.01e-6;
% Definimos nuestro intervalo de simulacion
t = 0:1e-4:0.008;
t = t';
n = length(t);
Vi = heaviside(t);
Vi = Vi + 0.1*random('Normal',0,1,n,1);
% Graficamos cada resultado
% Primero graficamos nuestra señal de entrada
subplot(411)
plot(t,Vi,'r')
grid on
num = [C1 0];
den = [R1*C1 1];
G_I1_Vi = tf(num,den);
clear num den
num = [R1*C1 1];
num = C2*num;
den = [R2*C2 1];
den = C1*den;
G_I2_I1 = tf(num,den);
clear num den
num = 1;
den = [C2 0];
G_Vo_I2 = tf(num,den);
clear num den
num = 1;
den = [R2*C2 1];
I1 = lsim(G_I1_Vi,Vi,t);
subplot(412)
plot(t,I1,'r')
grid on
I2 = lsim(G_I2_I1,I1,t);
subplot(413)
plot(t,I2,'g')
grid on
Vo = lsim(G_Vo_I2,I2,t);
```

```
subplot(414)
plot(t,Vo,'b')
grid on
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Prototipo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

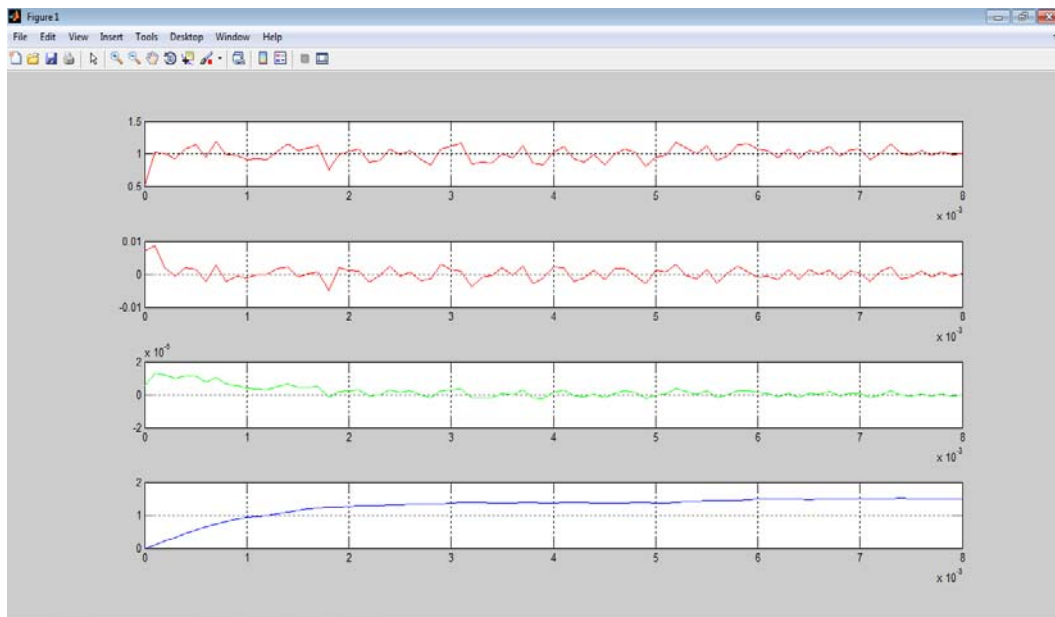


Fig. 54 Comportamiento del filtro de primer orden

Debido a que este sensor no depende de condiciones iniciales, se utilizará la representación en espacio estados la cual nos dará la función de transferencia.

$$G_{\text{sensor}} = \frac{0.05}{1e - 3s + 1}$$

```
% El modelamiento del sensor de temperatura incluye 3 etapas en
% una: La primera consiste en la conversión de Temperatura a mV (con
% ganancia 0.01)la segunda consiste en un filtro de primer orden(para
% eliminar ruidos indeseados)y la última que consiste en un
% amplificador de señal debido a que este sensor no depende de
% condiciones iniciales, se utilizará la representación con función
% de transferencia
k1 = 0.01; % Ganancia del LM35
```

```
R = 100e3; % Resistencia del filtro de primer orden
C = 0.01e-6; % Condensador del filtro de primer orden
k2 = 5; % Ganancia del amplificador operacional
A = -1/(R*C);
B = k1*k2/(R*C);
C = 1;
D = 0;
Gsen = ss(A,B,C,D);
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Prototipo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

3.1.10. DISEÑO DEL CONTROL PID

El control PID presenta distintos tipos de características dependiendo de la conexión de los circuitos del controlador. La configuración más utilizada a nivel industrial es la conexión paralela que se representa en la siguiente ecuación:

$$u(t) = Kp \left(e(t) + \frac{1}{Ti} \int e(t)dt + Td \frac{d}{dt} e(t) \right)$$

La función de transferencia del control G_{CPID}

$$G_{CPID} = \frac{U_{PID}(s)}{E(s)}$$

$$G_{CPID} = Kp \left(1 + \frac{1}{Ti} + Td(s) \right)$$

En donde:

Kp : Constante proporcional.

Ti : Tiempo integral.

Td : Tiempo derivativo

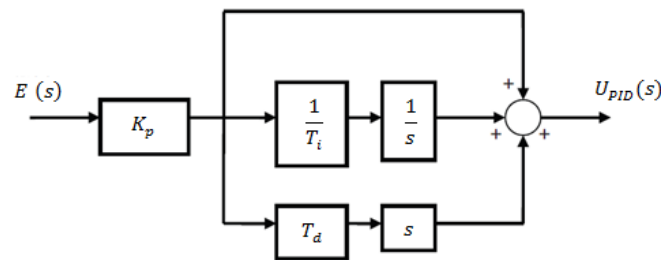


Fig. 55. Diagrama de Bloques del control PID

En el control PID paralelo cuyo diagrama de bloques se muestra en la figura 59, la constante de proporcionalidad afecta el valor de la señal de error produciendo cambios en el comportamiento en la parte integral y derivativa del control.

Desarrollando la función de transferencia del control PID, se tiene:

$$G_{CPID} = Kp + \frac{Kp}{T_i s} + Kp T_d s$$

En donde:

$$\frac{Kp}{T_i} = K_i : \text{Constante Integral}$$

$$Kp T_d = K_d : \text{Constante derivativa}$$

$$G_{CPID} = Kp + \frac{K_i}{s} + K_d s$$

3.1.10.1 SINTONIZACIÓN DE LAS CONSTANTES PID

Con la selección del tipo de control PID, se procede a determinar las constantes del controlador (K_p, T_i, T_d) para tener una referencia en la variación de las constantes, para esto tenemos que simular todo nuestro sistema con simulink, se usa el bloque PID controller el cual nos ayuda a

sintonizar nuestro controlador, por prueba y error. La ventaja que nos da PID controller es que nos ayuda en la sintonización proponiéndonos constantes K_p, T_i, T_d y así tomar referencia, cual es el más cercano en la sintonización.

El método computacional es usado cuando la sintonización por los métodos ya conocidos no se puede aproximar.

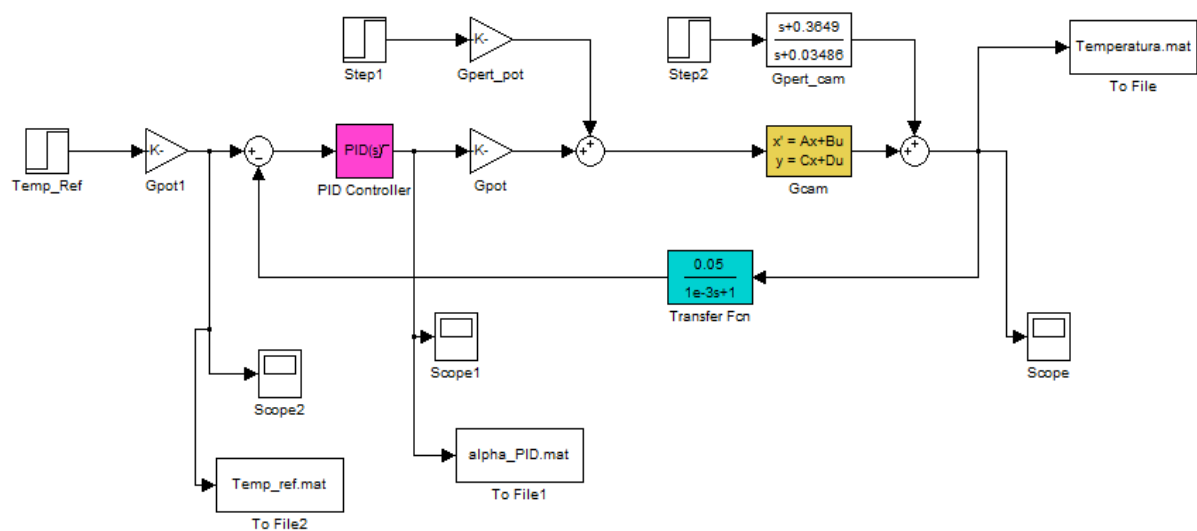


Fig. 56. Sistema de Control de temperatura en simulink (PID continuo)

Configuración del PID controller:

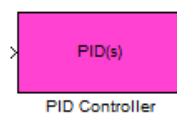


Fig.57. Bloque del PID controller

Al iniciar la sintonización en el PID controller aparece la siguiente ventana, en la cual se muestra que matlab intenta la auto calibración

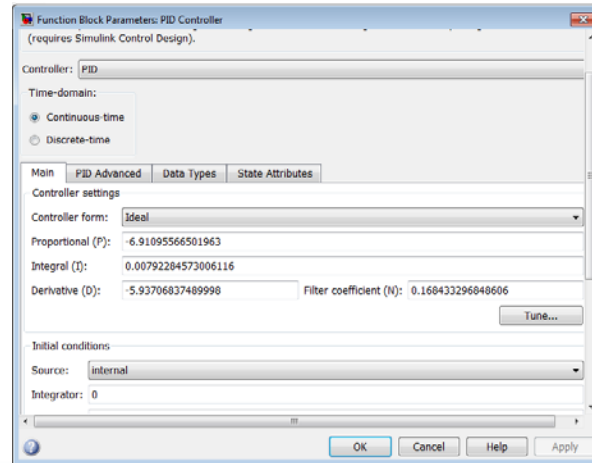


Fig. 58. Autoajuste PID por PID tuner

Presionamos tune y nos muestra la siguiente ventana:

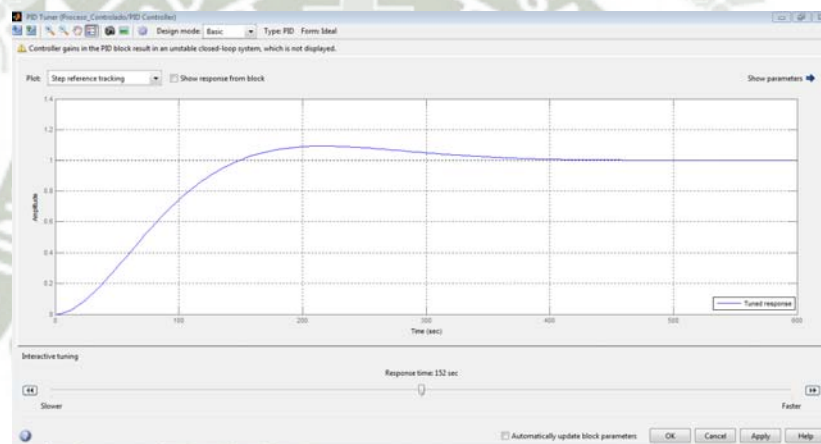


Fig. 59. Ventana de aproximación del sintonizado de PID tuner

En esta ventana nos muestra el comportamiento al cual debemos llevar nuestro controlador, después de realizar varios intentos de sintonización se logró obtener el más óptimo, que tiene los siguientes valores:

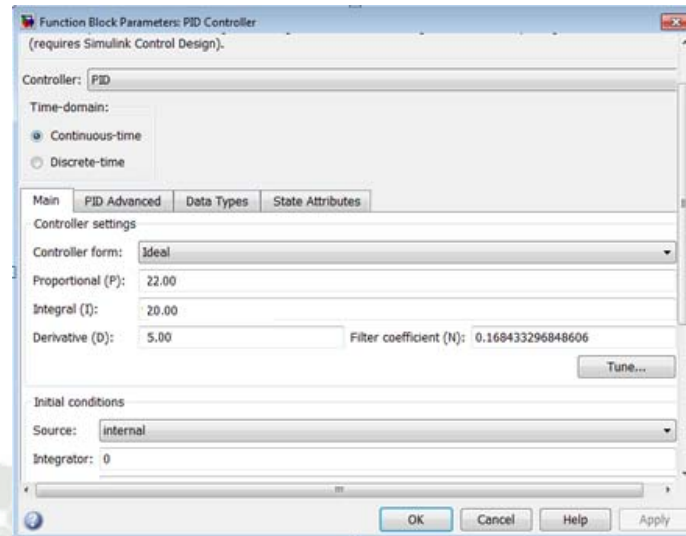


Fig. 60. Constantes del Controlador PID encontrado

Simulamos el proceso y visualizamos el comportamiento en un “Scope”.

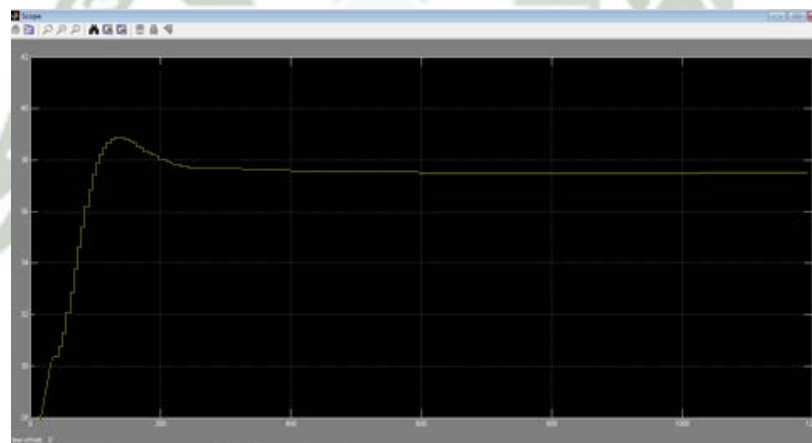


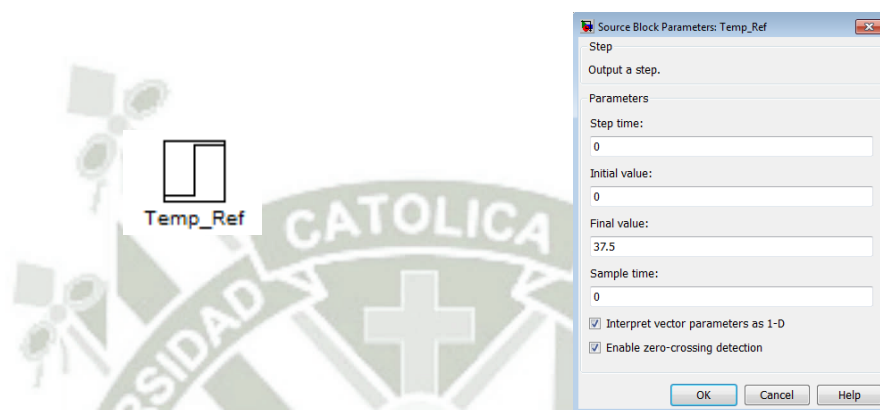
Fig. 61. Comportamiento del controlador PID

Tiene un “overshoot” no tan prolongado (dentro de los rangos permitidos), llevando a la temperatura de la incubadora de una forma suave,

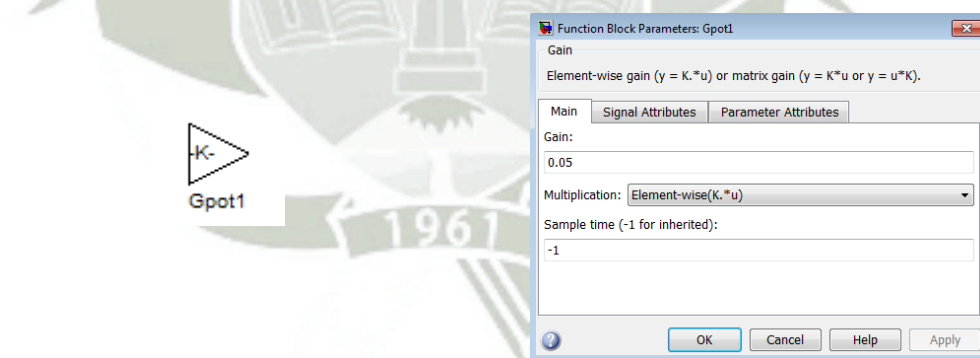
Ahora procedo a explicar todos los bloques que intervinieron en la simulación de nuestro controlador con simulink de matlab, para verificar su

comportamiento , además se desarrolla un programa para visualizar el comportamiento del controlador en forma más clara.

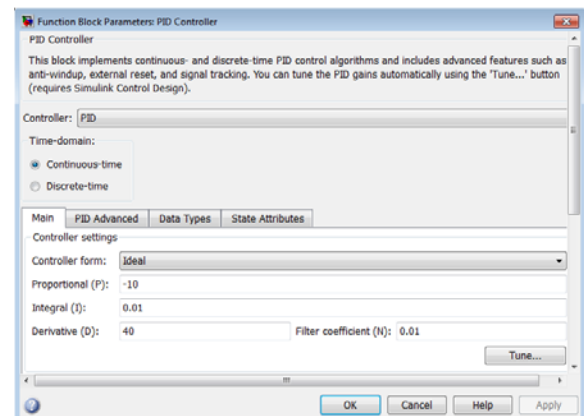
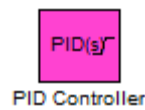
Temp_Ref , este bloque contiene una entrada escalón como valor final 37.5 , que es la temperatura que se requiere en la cámara térmica.



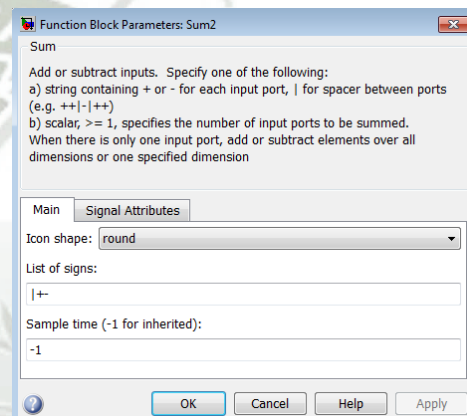
Gpot1 , este bloque contiene una entrada escalón como valor final 37.5 °C , que es la temperatura que se requiere en la cámara térmica.



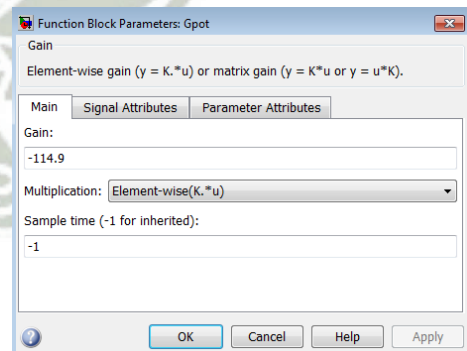
PID Controller, este bloque contiene las constantes de nuestro PID para mantener la temperatura a 37.5°C de la incubadora.



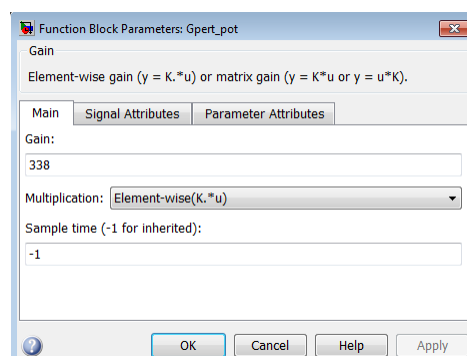
Retroalimentación, bloque que realimenta el sistema.



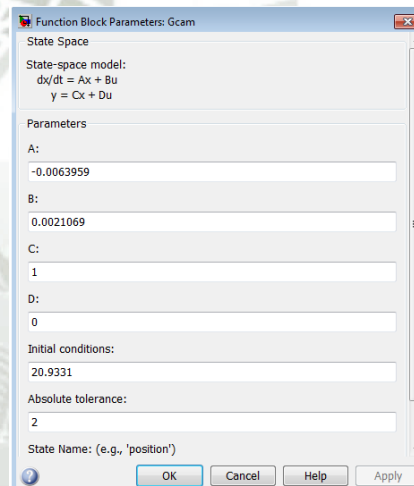
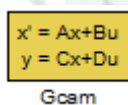
Gpot, bloque que contiene la ganancia de la etapa de potencia.



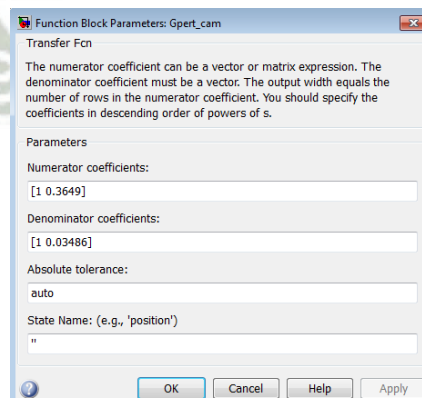
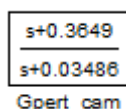
Gpert_pot, bloque que contiene la perturbación de la etapa de potencia.



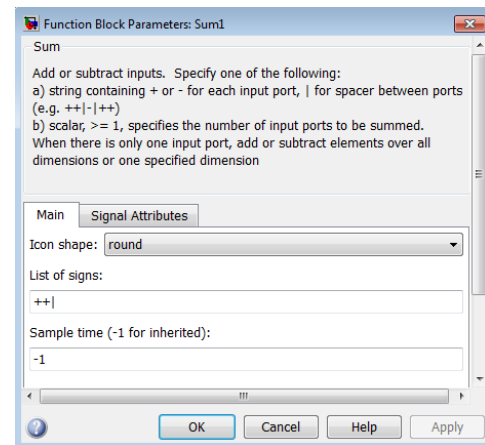
Gcam, bloque que contiene el espacio estado de la planta.



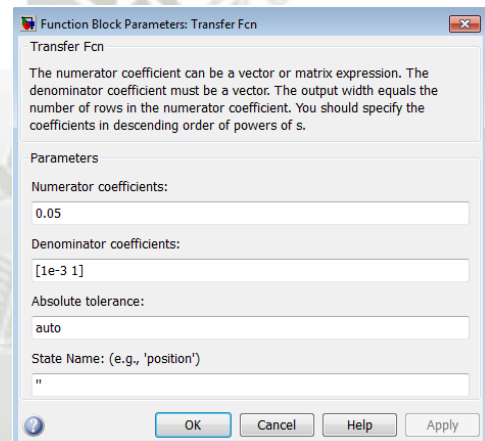
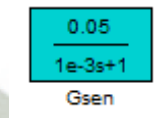
Gpert_cam, bloque que contiene la perturbación del espacio estado de la planta.



Sum, bloque que suma dos entradas.



Gsen, bloque que contiene función de transferencia del sensor.



Se desarrolla un programa para visualizar mejor el comportamiento del controlador, la gráfica superior nos indica el comportamiento del controlador en donde estabiliza de forma eficaz y eficiente la temperatura, y en la gráfica de la parte inferior el comportamiento de la señal del ángulo que da el controlador a la lámpara incandescente y así poder entregar la potencia necesaria a la lámpara.

```
% El siguiente programa nos muestra el comportamiento del
% controlador y el ángulo de disparo.
% El controlador tiene no tiene que sobrepasar el overshoot
% permitido. El ángulo de disparo tiene que estar entre la zona
% lineal, así llegar a la temperatura deseada y de la forma más
% estable posible además el comportamiento debe ser suave.
% Para ejecutar el siguiente programa primero se debe de ejecutar
% el proceso en simulink.
clear all
close all
clc
% Cargamos los datos de la temperatura y del angulo de disparo
% del controlador
load Temperatura.mat
load alpha_PID.mat
t = Temperatura(1,:);
Temp = Temperatura(2,:);
alpha = alpha_PID(2,:);
subplot(211)
plot(t,Temp,'r')
xlabel('tiempo(s)')
ylabel('Temperatura(°C)')
grid on
hold on
subplot(212)
plot(t,180*alpha/pi,'g')
xlabel('tiempo(s)')
ylabel('Angulo de Disparo(°)')
grid on
hold on
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Protoripo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

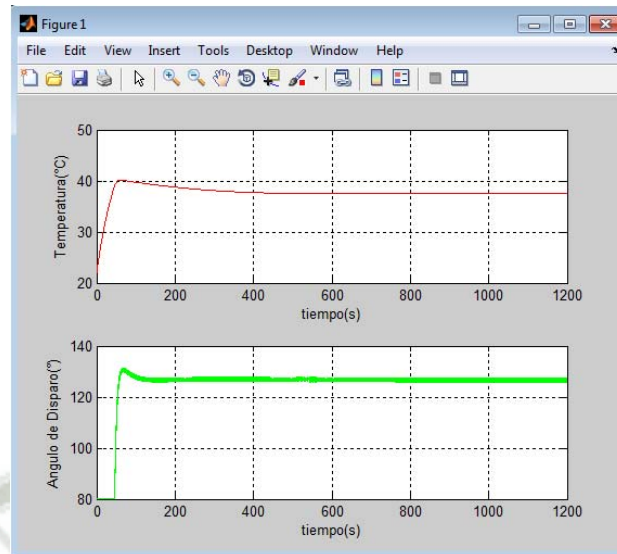


Fig. 62. Comportamiento del controlador en función de la temperatura y el ángulo de disparo

El siguiente programa nos muestra el comportamiento del actuador (cruce por cero) con la cámara térmica.

```
% Programa para graficar las señales de salida del simulink; para este
% caso, dichas señales de salida son:
% Temperatura: Es la temperatura del aire dentro de la camara
% incubadora (°C)
% delta_T: Es el tiempo de espera, contando desde el cruce por cero de
% la señal de alimentación alterna, para generar el flanco de disparo
% para la etapa de potencia cabe resaltar que la variable delta_T solo
% se podra actualizar despues de pasar 8.33ms (tiempo que demora la
% señal de alimentación en cruzar por cero)
% Finalmente, los datos que se cargan en el presente programa fueron
% exportados de la simulación obtenida a partir del simulink.
% Para ejecutar el siguiente programa primero se debe de ejecutar
% el proceso en simulink.
```

```
clear all
close all
clc
load Temperatura.mat
load Temp_ref.mat
load delta_T.mat
load Voltaje_rms.mat
t = Temperatura(1, :);
```

```
Temp = Temperatura(2,:);
Temp_ref = Temp_ref(2,:);
delta_T = delta_T(2,:);
Voltaje_rms = Voltaje_rms(2,:);
subplot(311)
plot(t,Temp,'r')
xlabel('tiempo(s)')
ylabel('Temperatura(°C)')
title('Temperatura del Aire dentro de la Incubadora')
grid on
hold on
plot(t,Temp_ref,'g')
legend('Planta','Referencia')
subplot(312)
plot(t,1e3*delta_T,'g')
xlabel('tiempo(s)')
ylabel('Delta de Tiempo(ms)')
title('Tiempo para el Flanco de Disparo')
grid on
subplot(313)
plot(t,Voltaje_rms,'r')
xlabel('tiempo(s)')
ylabel('Voltaje(V)')
title('Voltaje RMS de salida de la Etapa de Potencia')
grid on
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Prototipo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

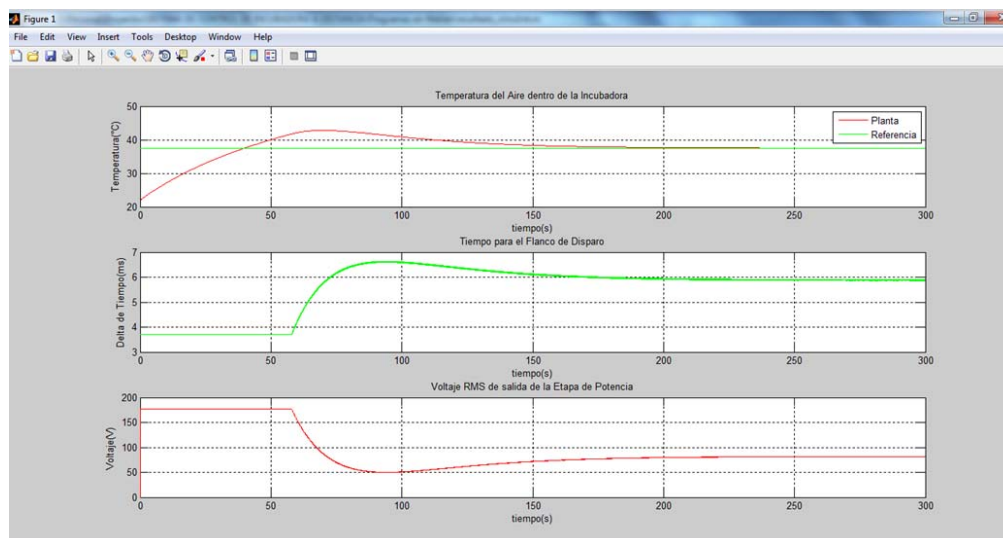


Fig. 63. Simulación del controlador PID continuo

3.1.11 DISEÑO DE LOS CONTROLADORES DIGITALES

3.1.11.1 CONTROL PID A BASE DE MICROCONTROLADOR

Los sistemas de control análogos utilizan señales definidas en un intervalo de tiempo. La gran mayoría de las señales que se desean controlar son de naturaleza análoga, para llevarlas a sistemas de control digitales se procede a la digitalización de las mismas por medio de conversores análogo-digitales en un tiempo discreto, a esto se llama muestreo de señales.

El proceso de muestreo es seguido de un proceso de cuantificación, o sea llevar a la señal muestreada a una representación numérica binaria, de esta manera se procesa el dato numérico por medio de microcontroladores para recibir una respuesta digital, la misma que se la transforma a una señal analógica por medio de conversores de digital a analógico.

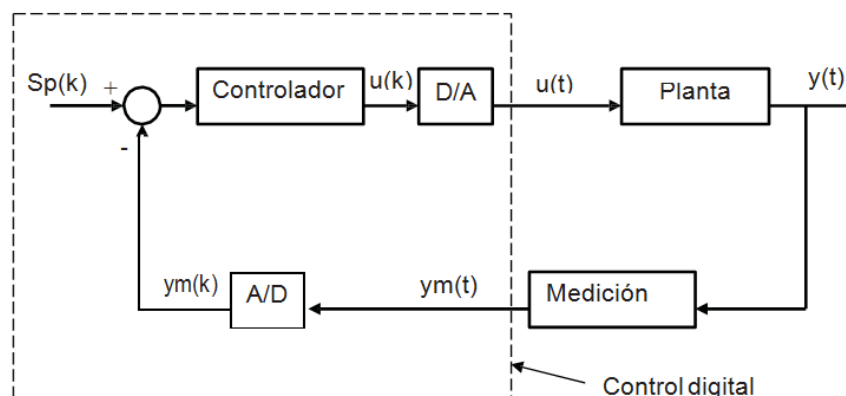


Fig.64. Diagrama de bloques de un sistema control digital

El sistema de control (figura 63) indica que la señal de la variable controlada es acondicionada y luego digitalizada para que el controlador realice la resta en forma digital para generar la señal de error como dato digital, y con el valor de la referencia se la introduce en forma digital.

La señal de control es transformada a una señal analógica para el control de la planta.

Otro tipo de sistema de control digital es la digitalización de la señal de error para que el controlador digital realice la acción de control, pero este método no se toma en cuenta en la presente tesis.

3.1.11.2 DISCRETIZACIÓN DEL CONTROL PID

Para realizar el control digital se utiliza varios métodos de discretización de la función de transferencia del tipo de control para que posteriormente, forme parte del programa del microcontrolador.

Para ingresar la función de transferencia del control PID, se la representa en forma de una ecuación en diferencias, para llegar a la ecuación hay que partir de la función de transferencia del control PID (figura 59).

$$G_{CPID} = \frac{U_{PID}(s)}{E(s)}$$

$$G_{CPID} = K_p \left(1 + \frac{1}{T_i s} \right) + T_d s$$

Para discretizar el control PID, el método más comúnmente utilizado es aplicar en la parte integral del PID el método de TUSTIN y para la parte derivativa el método rectangular hacia atrás.

Se realiza el reemplazo de la parte integral con la siguiente expresión:

$$s = \frac{2(z-1)}{T(z+1)}, \text{ método de TUSTIN}$$

Para la parte derivativa se realiza el reemplazo con la siguiente expresión:

$$s = \frac{z-1}{Tz}, \text{ método rectangular hacia atrás}$$

$$\frac{U_{CPID}(z)}{E(z)} = K_p \left(1 + \frac{1}{T_i \frac{2(z-1)}{T(z+1)}} \right) + T_d \frac{z-1}{Tz}$$

Resolviendo la expresión se tiene:

$$\frac{U_{CPID}(z)}{E(z)} = K_p \left(1 + \frac{T(1+z')}{2T(1-z')} + \frac{T_d}{T} (1-z') \right)$$

$$\frac{U_{CPID}(z)}{E(z)} = K_p \left(1 - \frac{T(1-z')}{2T(1-z')} + \frac{2T}{2T(1-z')} + \frac{T_d}{T} (1-z') \right)$$

$$\frac{U_{CPID}(z)}{E(z)} = K_p \left(1 - \frac{T}{2T_i} + \frac{T}{T_i(1-z')} + \frac{T_d}{T} (1-z') \right)$$

Para obtener la ley de control en dominio del tiempo discreto, se utiliza un método de control discreto en la cual no interviene el periodo (T) para incluirlo al microcontrolador al que se llama control incremental, el cual parte de las siguientes definiciones.

- Control proporcional. Se tiene la siguiente ley de control con su función de transferencia en dominio de z.

$$u[k] = K_p e[k]$$

$$G(z) = K_p$$

- Control integral. Su ley de control y su función de transferencia en el dominio de z es la siguiente.

$$u[k] - u[k-1] = K_i e[k]$$

$$G(z) = \frac{K_i}{1-z'}$$

- Control derivativo. Presenta la siguiente ley de control y su función de transferencia en dominio de z.

$$u[k] = K_d (e[k] - e[k-1])$$

$$G(z) = K_d (1 - z')$$

Aplicando el método denominado control incremental, la función de transferencia del control PID en dominio de (z) se la puede simplificar de la siguiente forma:

$$\frac{U_{CPID}(z)}{E(z)} = (K_p + \frac{K_i}{1-z'} + K_d(1-z'))$$

En donde:

$$K_p = K - \frac{K_i}{2} \quad \text{Constante Proporcional}$$

$$K_i = \frac{KT}{T_i} \quad \text{Constante Integral}$$

$$K_d = \frac{KT}{T_d} \quad \text{Constante Derivativa}$$

$$U_{CPID}(z) = \left(\frac{K_p(1-z') + K_i + K_d(1-z')^2}{(1-z')} \right) E(z)$$

$$(1-z')U_{CPID}(z) = (K_p(1-z') + K_i + K_d(1-z')^2) E(z)$$

$$(1-z')U_{CPID}(z) = (K_d z^2 - (K_p + 2K_d)z' + (K_p + K_i + K_d))E(z)$$

Aplicando la transformada inversa se tiene la ley de control expresada en la siguiente ecuación en diferencias:

$$u[k] - u[k-1] = K_d e[k-2] - (K_p + 2K_d)e[k-1] + (K_d + K_i + K_d)e[k]$$

3.1.11.2.1 COMPORTAMIENTO DE CONTROL DISCRETO PID

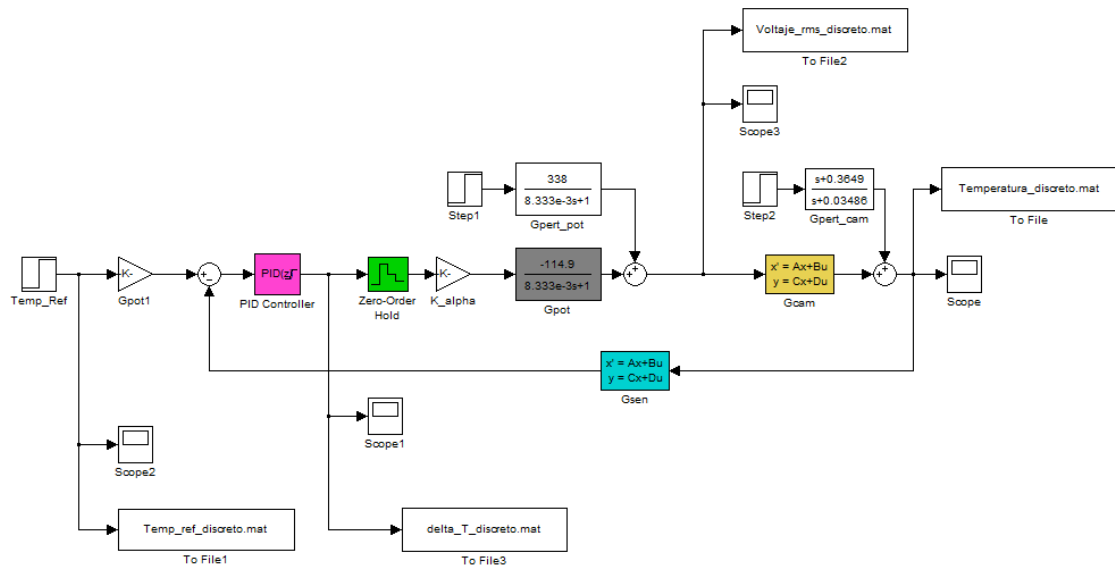


Fig. 65. Sistema de Control de temperatura en simulink (PID discreto)

% Programa para graficar las señales de salida del simulink; para
% este caso, dichas señales de salida son:
% Temperatura: Es la temperatura del aire dentro de la cámara
% incubadora (°C)
% delta_T: Es el tiempo de espera, contando desde el cruce por cero
% de la señal de alimentación alterna, para generar el flanco de
% disparo para la etapa de potencia cabe resaltar que la variable
% delta_T solo se podrá actualizar después de
% pasar 8.33ms (tiempo que demora la señal de alimentación en cruzar
% por cero)
% Finalmente, los datos que se cargan en el presente programa fueron
% exportados de la simulación obtenida a partir del simulink

```
clear all
close all
clc
load Temperatura_discreto.mat
load Temp_ref_discreto.mat
load delta_T_discreto.mat
load Voltaje_rms_discreto.mat
Temp = Temperatura_discreto(2,:);
Temp_ref = Temp_ref_discreto(2,:);
delta_T = delta_T_discreto(2,:);
Voltaje_rms = Voltaje_rms_discreto(2,:);
```

```
subplot(311)
plot(Temp, 'r')
xlabel('muestra')
ylabel('Temperatura(°C)')
title('Temperatura del Aire dentro de la Incubadora')
grid on
hold on
plot(Temp_ref, 'g')
legend('Planta', 'Referencia')
subplot(312)
plot(1e3*delta_T, 'g')
xlabel('muestra')
ylabel('Delta de Tiempo(ms)')
title('Tiempo para el Flanco de Disparo')
grid on
subplot(313)
plot(Voltaje_rms, 'r')
xlabel('muestra')
ylabel('Voltaje(V)')
title('Voltaje RMS de salida de la Etapa de Potencia')
grid on
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Prototipo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

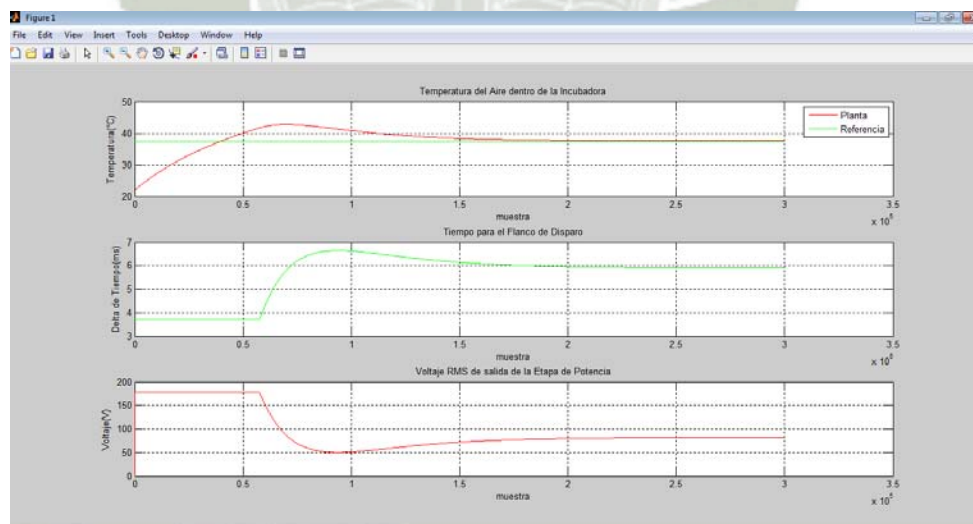
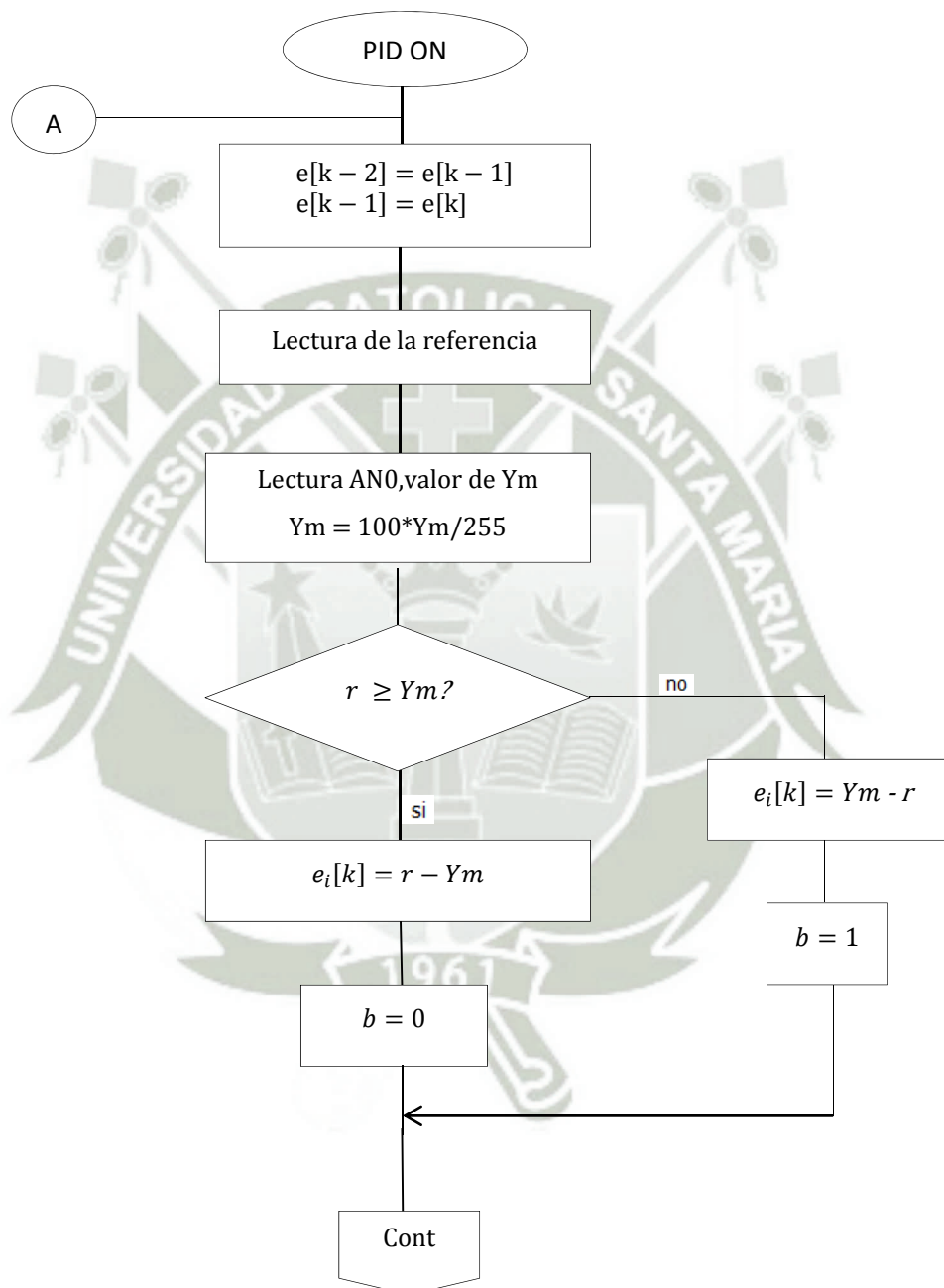
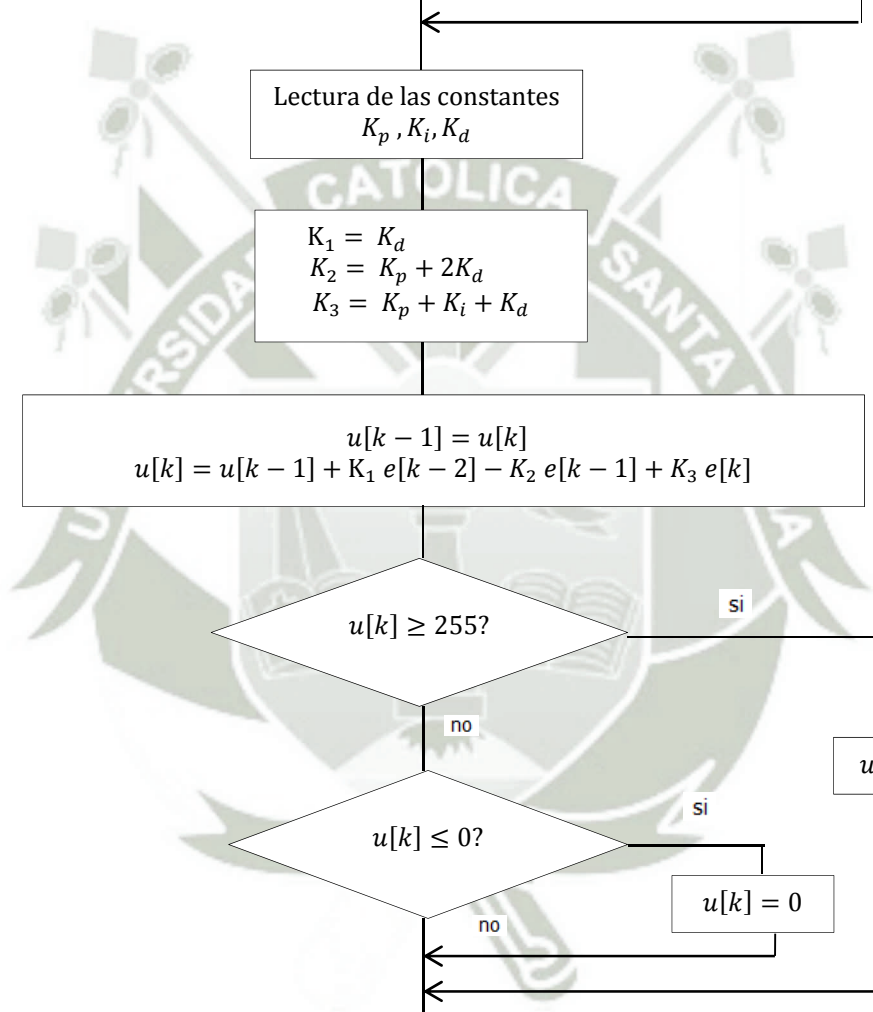


Fig. 66. Simulación del controlador PID discreto

3.1.11.2.2 IMPLEMENTACIÓN DE LA LEY DE CONTROL.

De acuerdo con el diagrama de bloques del sistema de control con el controlador digital, la lectura de la variable controlada de la planta (Y_m) realiza la digitalización a un valor numérico (conversión análogo - digital).





El valor de referencia (S_p) se calibra directamente en forma digital, teniendo otra manera para ingreso de esta variable.

La señal de error se genera restando la variable medida y la referencia en forma digital, y con este valor numérico se aplica la ley de control obtenida por medio de la discretización.

El valor numérico de la variable de control ($u[k]$), realiza la conversión digital-analógica que sirve como la señal de control para la planta.

En vista de los periféricos que se deben implementar para el control digital, se utiliza un microcontrolador que cumpla con las exigencias del sistema de control.

Se escoge ATmega644 del fabricante ATMEL cuyas características más importantes son las siguientes:

- Memoria FLASH: 64KBytes, 16384 Instrucciones.
- Memoria RAM: 2K
- Memoria EEPROM: 256 Bytes.
- Timers 5 x 16-bit 2 x 32-bit
- Modos de Oscilador
 - Externo hasta 48MHz
 - Externo auxiliar, para modulo USB
 - Interno con RC
- Interno programable de 7.37 MHz, 512 kHz

- Pines de entrada/salida digitales: 21.
- Canales ADC: 13 entradas conversores análogo a digital de 10 bits de resolución.
- 2-canales de 16-bit de control de motores PWM.
- Comunicación: USB V2.0, Baja velocidad (1.5 Mb/s) y velocidad Completa (12 Mb/s). Serial USART, SPI, I2C.
- Rango de Temperatura para funcionamiento: -40 Hasta 150°C
- Voltaje de operación: 2.7 V – 5.5 V.

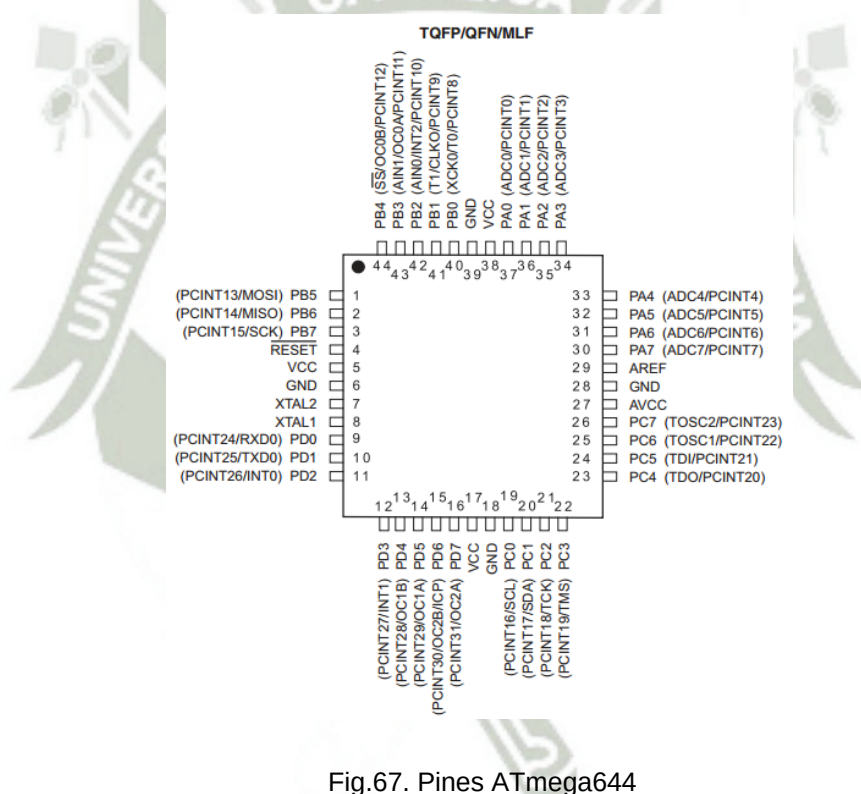


Fig.67. Pines ATmega644

El siguiente programa es que el ejecutará el algoritmo de PID :

```
#include <stdlib.h>
#include <stdint.h>
#include <avr/io.h>
#include <avr/interrupt.h>

#include "TaskApplication.h"
```

```
#include "Kernel/MultiTask.h"
#include "Utils.h"
#include "AppDefinitions.h"
#include "AVRDriver/AVR_UART.h"

void TaskApplication_SetPWM(U16 PulseWidth);
extern brdStatus CreateTimer(U32 *TaskTimer_TimerValue);
extern unsigned char LoadFromEEPROM(short AddressInit, unsigned char
*Buffer, short DataLength);
extern byte SyncIsDetected(void);
extern void SendSyncPulse(void);
extern void UpdateOutputs(void);
extern void UpdateSemaphoreMode(void);
void TaskApplication_Status(void);

U32 TaskApplication_TimeOut;
U16 TaskApplication_CounterSecond;
U16 TaskApplication_Chksum;
byte *TaskApplication_ptrBuffer;

U16 TaskApplication_Aux16;
byte TaskApplication_ActiveController;

U32 TaskApplication_LastSamples[64];
U64 TaskApplication_Promedio;
byte TaskApplication_SampleID;

long PID_Error0;
long PID_Error1;
long PID_Auxiliar;
long PID_Auxiliar2;

UInt16 TaskApplication_SensorValue;
UInt32 TaskApplication_Acumulator;
byte TaskApplication_Counter;
UInt32 FactorK;
UInt32 DeltaADC;

//-----//
//struct_Status GetSemaforesStatus(void);
//-----//

void TaskApplication()
{
    TaskApplication_Init();
    TaskApplication_ActiveController=1;

    TaskApplication_SampleID=0;
    FactorK = 537;
    DeltaADC= 0;

    while(1)
    {
        TaskApplication_Sleep(1000);
        DeltaADC = TaskApplication_SensorValue;
        DeltaADC = DeltaADC*FactorK + 5000;
        AppStatus.Temperature0 = (UInt16)(DeltaADC/100);

        if(AppSetting.Active==0)
```

```

    {
        TaskApplication_SetPWM(0);
        PID_Auxiliar = 0;
        PID_Error0 = 0;
        PID_Error1 = 0;

        AppStatus.PWM_Duty      = 0;
        AppStatus.Feedback      = 0;

        AppStatus.States.Y0 = 0;
        AppStatus.States.Y1 = 0;
        AppStatus.States.Y2 = 0;

        AppStatus.States.Er0= 0;
        AppStatus.States.Er1= 0;
        AppStatus.States.Er2= 0;
    }
    else
    {
        PID_Error0= (long)(AppSetting.SetPoint -
AppStatus.Temperature0);

        PID_Auxiliar = (long)(AppSetting.PID.b0)*PID_Error0;
        PID_Auxiliar += (long)(AppSetting.PID.b1)*PID_Error1;
        PID_Auxiliar = PID_Auxiliar / 100;
        PID_Auxiliar +=
(long)(AppStatus.States.Y1*AppSetting.PID.a0);

        AppStatus.States.Y0 = (UInt16)PID_Auxiliar;

        PID_Error1 = PID_Error0;
        /* Ecuacion en diferencias
        /*
        if(AppStatus.States.Y0 & 0x1000)
            AppStatus.States.Y0 = 0;
        if(AppStatus.States.Y0>0x03FF)
            AppStatus.States.Y0=0x03FF;
        */

        // Actualizar PWM
        TaskApplication_SetPWM(AppStatus.States.Y0);

        //Guarda valores anteriores.
        AppStatus.States.Y1 = AppStatus.States.Y0;
        AppStatus.States.Er2 = AppStatus.States.Er1;
        AppStatus.States.Er1 = AppStatus.States.Er0;
    }
}

void TaskApplication_SetPWM(U16 PulseWidth)
{
    // OCR3A = PulseWidth & 0x03FF;
}

// Inicializacion de parametros
brdStatus TaskApplication_Init(void)
{
    // Inicializacion de variables locales

```

```

TaskApplication_SensorValue = 0;
TaskApplication_Counter = 0;
TaskApplication_Acumulator=0;
TaskApplication_TimeOut = 0;

CreateTimer(&TaskApplication_TimeOut);

TaskApplication_LoadDefault();
TaskApplication_Status();

// Deshabilitacion de los canales 0 y 1 del ADC
DIDR0 = (1<<ADC0D)|(1<<ADC1D);

// Inicializacion de ADCs, ADC inicilizara con el Timer1 overflow
ADMUX = (0 << ADLAR) | (3 << REFS0) | (0 << MUX0);
ADCSRA = (0 << ADEN) | (0 << ADSC) | (1 << ADATE) | (0 << ADIE) |
(5 << ADPS0);
ADCSRB = (0 << ADTS0);
ADCH = 0;
ADCL = 0;

// Inicializacion Timer1

// Inicializacion de PWMs (Timer2)

// Configuracion de interrupciones del ADC
ADCSRA |= (1 << ADIE);

// Configuracion de interrupciones del timer (Periodo de muestreo)

// Habilitacion de la conversion:
ADCSRA |= (1 << ADEN);
ADCSRA |= (1 << ADSC);

return BRD_OK;
}

void TaskApplication_LoadDefault(void)
{
    // Configuracion inicial de parametros
    AppSetting.HardSetting.CommPort.PortNumber = 0;
    AppSetting.HardSetting.CommPort.BaudRate = 9600;
    AppSetting.HardSetting.CommPort.Parity = UART_PARITY_NONE;
    AppSetting.HardSetting.CommPort.StopBits = UART_STOPBIT_ONE;
    AppSetting.HardSetting.CommPort.DataBits = UART_BITS_8;

    AppSetting.Active = 1;
    AppSetting.SetPoint = 3900;

    AppSetting.PID.a0 = 1;
    AppSetting.PID.a1 = 0;
    AppSetting.PID.a2 = 0;
    AppSetting.PID.b0 = 10;
    AppSetting.PID.b1 = 3;
    AppSetting.PID.b2 = 0;
}

void TaskApplication_Status(void)

```

```

{
    // Resetea variables de estado.
    AppStatus.Temperature0= 0;
    AppStatus.PWM_Duty      =      0;
    AppStatus.Feedback      =      0;

    AppStatus.States.Y0 = 0;
    AppStatus.States.Y1 = 0;
    AppStatus.States.Y2 = 0;

    AppStatus.States.Er0= 0;
    AppStatus.States.Er1= 0;
    AppStatus.States.Er2= 0;
}

//-----//
//-----//
void TaskApplication_Sleep(U32 TaskApplication_wait)
{
    TaskApplication_TimeOut = TaskApplication_wait;
    while( TaskApplication_TimeOut > 0)
    {
        DoEvents();
    }
}

//-----//
// Control de interrupciones
ISR (ADC_vect)
{
    TaskApplication_Acumulator += ADC;
    TaskApplication_Counter++;
    if(TaskApplication_Counter>7)
    {
        TaskApplication_SensorValue = TaskApplication_Acumulator
/ 8;
        TaskApplication_Counter = 0;
        TaskApplication_Acumulator = 0;
    }
}

// Para el manejo del LCD
extern brdStatus CreateTimer(U32 *TaskTimer_TimerValue);

U32 TaskLCD_Timer;
U32 TaskLCD_TimerOption;
byte TaskLCD_Step;
byte TaskLCD_Char;
byte TaskLCD_Option;
char TaskLCD_Buffer8[32]="00000000000000000000000000000000";

void TaskLCD(void)
{
    TaskLCD_Init();
    LCD_Cls();
    sleepTaskLCD(100);
    TaskLCD_ShowInitMessage();
}

```

```

while(1)
{
    sleepTaskLCD(200);

    if(TaskLCD_TimerOption==0)
    {
        LCD_Cls();
        switch(TaskLCD_Option)
        {
            case 0:
                LCD_WriteString(0,0,"- Y0:",sizeof("-
Y0:")-1,0);
                LCD_WriteString(0,1,"- Y1:",sizeof("-
Y1:")-1,0);
                Short2StrMSB0(AppStatus.States.Y0,
TaskLCD_Buffer8, 5);

                LCD_WriteString(11,0,TaskLCD_Buffer8,5,0);
                Short2StrMSB0(AppStatus.States.Y1,
TaskLCD_Buffer8, 5);

                LCD_WriteString(11,1,TaskLCD_Buffer8,5,0);
                break;
            case 1:
                LCD_WriteString(0,0,"- Y2:",sizeof("-
Y2:")-1,0);
                LCD_WriteString(0,1,"- Er0:",sizeof("-
Er0:")-1,0);
                Short2StrMSB0(AppStatus.States.Y2,
TaskLCD_Buffer8, 5);

                LCD_WriteString(11,0,TaskLCD_Buffer8,5,0);
                Short2StrMSB0(AppStatus.States.Er0,
TaskLCD_Buffer8, 5);

                LCD_WriteString(11,1,TaskLCD_Buffer8,5,0);
                break;
            case 2:
                LCD_WriteString(0,0,"- Er1:",sizeof("-
Er1:")-1,0);
                LCD_WriteString(0,1,"- Er2:",sizeof("-
Er2:")-1,0);
                Short2StrMSB0(AppStatus.States.Er1,
TaskLCD_Buffer8, 5);

                LCD_WriteString(11,0,TaskLCD_Buffer8,5,0);
                Short2StrMSB0(AppStatus.States.Er2,
TaskLCD_Buffer8, 5);

                LCD_WriteString(11,1,TaskLCD_Buffer8,5,0);
                break;
            case 3:
                LCD_WriteString(0,0,"- a0:",sizeof("-
a0:")-1,0);
                LCD_WriteString(0,1,"- a1:",sizeof("-
a1:")-1,0);
                Short2StrMSB0(AppSetting.PID.a0,
TaskLCD_Buffer8, 5);

```

```

        LCD_WriteString(11,0,TaskLCD_Buffer8,5,0);
        Short2StrMSB0(AppSetting.PID.a1,
TaskLCD_Buffer8, 5);

        LCD_WriteString(11,1,TaskLCD_Buffer8,5,0);
        break;
    case 4:
        LCD_WriteString(0,0,"- a2:",sizeof("-
a2:")-1,0);
        LCD_WriteString(0,1,"- b0:",sizeof("-
b0:")-1,0);
        Short2StrMSB0(AppSetting.PID.a2,
TaskLCD_Buffer8, 5);

        LCD_WriteString(11,0,TaskLCD_Buffer8,5,0);
        Short2StrMSB0(AppSetting.PID.b0,
TaskLCD_Buffer8, 5);

        LCD_WriteString(11,1,TaskLCD_Buffer8,5,0);
        break;
    case 5:
        LCD_WriteString(0,0,"- b1:",sizeof("-
b1:")-1,0);
        LCD_WriteString(0,1,"- b2:",sizeof("-
b2:")-1,0);
        Short2StrMSB0(AppSetting.PID.b1,
TaskLCD_Buffer8, 5);

        LCD_WriteString(11,0,TaskLCD_Buffer8,5,0);
        Short2StrMSB0(AppSetting.PID.b2,
TaskLCD_Buffer8, 5);

        LCD_WriteString(11,1,TaskLCD_Buffer8,5,0);
        break;
    case 6:

        LCD_WriteString(0,0,"SetPoint:",sizeof("SetPoint:")-1,0);
        LCD_WriteString(0,1,"Temp.:",sizeof("Temp.:")-1,0);

        Short2StrMSB0(AppSetting.SetPoint,
TaskLCD_Buffer8, 5);

        LCD_WriteString(11,0,TaskLCD_Buffer8,5,0);
        Short2StrMSB0(AppStatus.Temperature0,
TaskLCD_Buffer8, 5);

        LCD_WriteString(11,1,TaskLCD_Buffer8,5,0);
        break;
    default:
        break;
}

if(++TaskLCD_Option > 6)
    TaskLCD_Option = 0;

TaskLCD_TimerOption=2000;
}

```

```

    }
}

brdStatus TaskLCD_Init(void)
{
    //Inicializacion de variables
    TaskLCD_Timer = 0;
    TaskLCD_TimerOption=0;
    DoEvents();
    CreateTimer(&TaskLCD_Timer);
    CreateTimer(&TaskLCD_TimerOption);

    //Inicializacion de pantalla LCD
    LCD_Init();
    sleepTaskLCD(25);

    TaskLCD_Option=0;

    return BRD_OK;
}

void sleepTaskLCD(U32 TaskLCD_wait)
{
    TaskLCD_Timer = TaskLCD_wait;
    while(TaskLCD_Timer>=1)
        DoEvents();
}

void TaskLCD_ShowInitMessage(void)
{
    LCD_Cls();
    LCD_WriteString(0,0,"Proyecto",sizeof("Proyecto")-1,1);
    LCD_WriteString(0,1,"Incubadora",sizeof("Incubadora")-1,1);
    sleepTaskLCD(1500);
}

% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Prototipo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria

```

La señal de la variable medida (Y_m) se la acondiciona para llegar al nivel de voltaje que maneja el microcontrolador (niveles cmos). Se digitaliza a esta señal por medio del conversor A/D del microcontrolador por el canal (AN0). La resolución de la conversión es de 16 bits.

La ecuación en diferencias toma los valores del error en estados anteriores, el microcontrolador debe actualizar estos valores, la cuales se introducen como constante en la programación del microcontrolador.

El resultado de la ecuación en diferencias es un valor numérico que se encuentra limitado por el número de bits utilizado.

3.2 COMUNICACIÓN GSM

3.2.1 COMUNICACIÓN DE MODULOS

Una vez ya culminado el diseño de nuestro controlador PID e implementado en el microcontrolador procedemos a realizar la etapa de comunicación remota.

Se desarrolló los siguientes programas teniendo en cuenta la programación que da el fabricante:

Programa para las configuraciones iniciales y funciones generales del Fastrack Supreme 20

```
// Configuraciones y funciones generales FASTRACK SUPREME, se hace de  
// acuerdo a su manual de ejecución.  
// Importante: Este archivo se debe de guardar con nombre AUXILIAR.c  
// Declaramos funciones usadas:
```

```
void configurar_parametros(void); // Parámetros iniciales del  
programa.  
void sel_adc(int8 canal); // Seleccionamos canal AD para  
trabajar.
```

```
// Definimos el hardware utilizado:
```

```
#define PULSADOR1 PIN_A6 // Pulsador 1 - Activo en nivel bajo.  
#define PULSADOR2 PIN_B7 // Pulsador 2 - Activo en nivel bajo.
```

```
#define PULSADOR3 PIN_A7 // Pulsador 3 - Activo en nivel bajo.
#define PULSADOR4 PIN_B1 // Pulsador 4 - Activo en nivel bajo.

#define PWRKEY PIN_B6 // Pin PWRKEY FASTRACK.
#define PSSIM300 PIN_B4 // Fuente del FASTRACK.
#define OUT1 PIN_A0 // OUT 1 - Activo a nivel alto.
#define OUT2 PIN_A1 // OUT 2 - Activo a nivel alto.
#define RELAY PIN_A4 // RELE - Activo a nivel alto.

#define OUT1_ON output_high(OUT1); // Encendemos OUT1.
#define OUT1_OFF output_low(OUT1); // Apagamos OUT1.
#define OUT2_ON output_high(OUT2); // Encendemos OUT2.
#define OUT2_OFF output_low(OUT2); // Apagamos OUT2.
#define RELAY_ON output_high(RELAY); // Encendemos RELE.
#define RELAY_OFF output_low(RELAY); // Apagamos RELE.

// Estado de las entradas digitales.
#define ESTADO_PULSADOR1 input(PULSADOR1)
#define ESTADO_PULSADOR2 input(PULSADOR2)
#define ESTADO_PULSADOR3 input(PULSADOR3)
#define ESTADO_PULSADOR4 input(PULSADOR4)

// Estado de las salidas digitales.
#define ESTADO_OUT1 input(OUT1)
#define ESTADO_OUT2 input(OUT2)
#define ESTADO_RELE input(RELAY)

// Macros del módulo FASTRACK.

#define FASTRACK_PWRKEY_ON output_high(PWRKEY); // FASTRACK ON.
#define FASTRACK_PWRKEY_OFF output_low(PWRKEY); // FASTRACK OFF.

#define PSFASTRACK_ON output_high(PSFASTRACK); // Power Supply FASTRACK ON.
#define PSFASTRACK_OFF output_low(PSFASTRACK); // Power Supply FASTRACK OFF.

// desarrollo de funciones de usuario:

void configurar_parametros(void) {

// Configuramos E/S:
set_tris_a (0b11101100); // RA7 a RA0.
set_tris_b (0b10000111); // RB7 a RB0.
// Configuramos el oscilador.
setup_oscillator(OSC_8MHZ);
// Desactivamos las interrupciones.
disable_interrupts(GLOBAL);
disable_interrupts(INT_rda);
// Configuramos el conversor AD.
SETUP_ADC(ADC_CLOCK_DIV_16); // Clock/16.
SETUP_ADC_PORTS(sAN2|sAN3|VSS_VDD); // Canal AD: 2 y 3 ; referencia: 0 - VDD.

}

void sel_adc(int8 canal) {
    set_adc_channel(canal); // Seleccionamos canal analógico que utilizamos.
}
```

```
% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Protoripo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

El siguiente programa para proceder a la inicialización de Modulo GSM

Fastrack

```
// Programa de inicializacion de FASTRACK
// Importante: Este archivo se debe de guardar con nombre AUXILIAR.c
// Declaramos funciones usadas:

void FASTRACK_on(void); // Encendemos y
configuramos el módulo FASTRACK

// desarrollo de las funciones del modem:

void FASTRACK_on(void){
    FASTRACK_OFF // Apagamos la fuente del FASTRACK
    FASTRACK_PWRKEY_OFF // PRWKEY=0;
    delay_ms(1000); // Esperamos 1 seg.
    PSFASTRACK_ON // Encendemos la fuente del FASTRACK
    delay_ms(500); // Esperamos 1/2 seg.
    FASTRACK_PWRKEY_ON // PWRKEY=1;
    delay_ms(1500); // Esperamos 1-1/2 seg.
    FASTRACK_PWRKEY_OFF // PRWKEY=0;
    delay_ms(30000); // Esperamos 40 segundos el Call Ready.
    fputs("AT", FASTRACK);
    delay_ms(10000);
    fprintf(FASTRACK, "AT\n\r"); // Comando de
verificación.
    delay_ms(500); // Esperamos 500ms.
    fprintf(FASTRACK, "ATE0\n\r"); // Eco OFF.
    delay_ms(500); //
    Esperamos 500ms.
    fprintf(FASTRACK, "AT+CMGF=1\n\r"); // Configuramos modo
texto.
    delay_ms(500); //
    Esperamos 500ms.
    fprintf(FASTRACK, "AT+CPMS=\"SM\", \"SM\"\n\r"); // Selecciono la
mememoria SIM para recibir y leer datos.
    delay_ms(500); //
    Esperamos 500ms.
    fprintf(FASTRACK, "AT+CNMI=2,2,0,0,0\n\r"); // El dato, una vez
recibidos son enviados directamente al buffer de la USART.
    delay_ms(500); //
    Esperamos 500ms.
}
}
```

Programa para comunicación del módulo GSM con el microcontrolador.

```
// Inicialización del GSM councicacion del dsPIC.
// Microcontrolador utilizado: ATmega 644.

// Definimos el microcontrolador utilizado y las librerias utilizadas.
#include <stdlib.h>
#include <stdint.h>
#include <avr/io.h>
#include <avr/interrupt.h>
#include "AUXILIAR.c"
#include "FASTRACK.c"

// Conversor de 10 bits con justificación a la derecha.
#define ADC=10
// Configuramos velocidad de operación.
#define delay(clock=4000000) // Clock: 4Mhz.

// Configuramos fusibles de programación.
#define fuses INTRC_IO // Oscilador interno.
#define fuses NOWDT // No utilizamos el perro guardian.
#define fuses NOPROTECT // Memoria de programa no protegida.
#define fuses PUT // Power-up timer activado.
#define fuses NOMCLR // NO Master clear reset.
#define fuses NOBROWNOUT // Reset por bajo voltaje desactivado.
#define fuses NOLVP // Programación con bajo voltaje desactivado.
#define fuses NOCPD // Memoria EEPROM no protegida.
#define fuses NOWRT // NO habilitar escritura en memoria FLASH.
#define fuses NODEBUG // No incluir código para debug.

// Configuramos los puertos RS232 utilizados.
#define use_rs232(baud=9600, xmit=PIN_B5,rcv=PIN_B2,errors,stream=FASTRACK)
// RS232 - FASTRACK
#define use_rs232(baud=9600, xmit=PIN_B3,rcv=PIN_B0,errors,stream=DEBUG)
// RS232 - DEBUG.

// Utilizamos el modo de acceso rápido a los puertos.
#define use_fast_io(A)
#define USE_fast_io(B)

// Incluimos librerías utilizadas.

#include "AUXILIAR.c" // Funciones generales de configuración.
#include "FASTRACK.c" // Funciones de control del módulo FASTRACK.

void main() {

    configurar_parametros(); // Configuramos parámetros principales.

    FASTRACK_on(); // Encendemos el módulo GSM FASTRACK.

    while(true){ }
}

% Autor : Miguel Angel Ramirez Lazo
% Tesis : Control de Temperatura para Protoripo de Incubadora
% controlada a distancia
% Universidad Catolica de Santa Maria
```

3.3 INTERFAZ GRAFICA DE CONTROL POR COMPUTADORA

El diseño de la interfaz gráfica para el monitoreo de las señales del controlador, permite realizar un análisis de las repuestas de las señales provenientes del controlador

La visualización del comportamiento del sistema se realiza en un gráfico permitiendo tener mejor apreciación de la respuesta de la planta y de la señal de control.

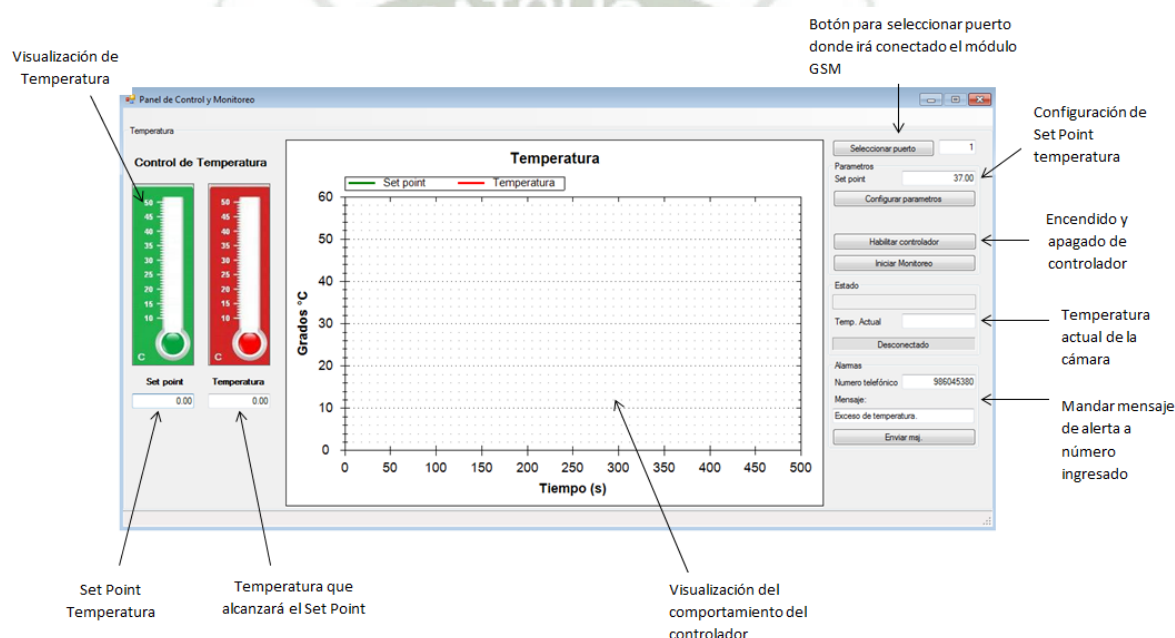


Fig.68. Interfaz gráfica del prototipo de incubadora

La interfaz gráfica se divide en tres bloques la parte izquierda consta de dos termómetros el de color verde, el cual será la temperatura a la cual la cámara será llevada, el termómetro rojo es el que nos indicará la temperatura actual y tendrá como tope la temperatura set point.

La parte central consta de un gráfico en el cual se ve el comportamiento de nuestro controlador en tiempo actual.

La última parte, la que se encuentre en el lado derecho, nos muestra en la configuración de parámetros, la configuración de la temperatura del set point, además se podrá encender o apagar el controlador , se verá la temperatura actual de la cámara y también se implementó un sistema de alarma para poder mandar un mensaje de alerta a cualquier numero tecleado por computadora.

La resolución del gráfico que maneja C# es de 1024 pasos, por lo que grafica la misma cantidad de puntos antes de que la pantalla comience a desplazarse y perder información.

Dependiendo del tipo de planta y de la respuesta que presente el sistema de control que se utiliza, para la obtención de la respuesta en estado transitorio y estable, al programa se incluye una rutina de retardo para el muestreo de la adquisición de datos de las señales monitoreadas.

El valor de la referencia es un valor ingresado por usuario dentro del rango de operación, nuestro controlador se estabilizará lo más rápido posible, tal como se halló en los cálculos y en la sintonización de éste.

La Programación del interfaz es la siguiente:

```
// Programacion para la configuracion de la interfaz para el monitoreo  
de temperatura del modulo de controlador  
using System;
```

```
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Text;
using System.Windows.Forms;

using System.Threading;
using ZedGraph;

namespace TCM02
{
    public partial class APP_MAIN : Form
    {
        bool m_EnableMonitoring;
        string m_DataRead;
        int xValue;

        struct _Control
        {
            public int SetPoint;
            public int Kp;
            public int Ki;
            public int Kd;
            public int Enable;
        }

        struct _Status
        {
            public int SetPoint;
            public int Kp;
            public int Ki;
            public int Kd;
            public int Controlar;
            public int Temperature;
        }

        _Control m_PID_Ctrl;
        _Status m_PID_Sts;

        double[] x = new double[500];
        double[] y = new double[500];
        double[] z = new double[500];

        public APP_MAIN()
        {
            InitializeComponent();
        }

        private void APP_MAIN_Load(object sender, EventArgs e)
        {
            m_EnableMonitoring = false;
            m_DataRead = "";
            APP_TIMER.Enabled = m_EnableMonitoring;
            Plot();

            try
            {

```

```

        APP_COMPORT.PortName = "COM" +
APP_TXT_COM.Text.Trim();
        APP_COMPORT.Open();

    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

private void APP_TIMER_Tick(object sender, EventArgs e)
{
    string ReceivedData;
    int StartPoint;
    int EndPoint=0;
    DateTime CurrentDate;
    APP_TIMER.Enabled = false;
    Plot();
    APP_TIMER.Enabled = m_EnableMonitoring;
}

private void button1_Click(object sender, EventArgs e)
{
    APP_COMPORT.Write("ATD94405151\r");
    APP_COMPORT.RtsEnable = true;
    Thread.Sleep(50);
    APP_COMPORT.DtrEnable = false;
}

private void APP_BTN_SET_COM_Click(object sender, EventArgs e)
{
    try
    {
        if (APP_COMPORT.IsOpen == true)
            APP_COMPORT.Close();
        APP_COMPORT.PortName = "COM" +
APP_TXT_COM.Text.Trim();
        APP_COMPORT.Open();
        MessageBox.Show(APP_COMPORT.PortName + " Selected.");
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}

private void APP_BTN_ENABLE_MONITORING_Click(object sender,
EventArgs e)
{
    if (m_EnableMonitoring == true)
    {
        m_EnableMonitoring = false;
        APP_BTN_ENABLE_MONITORING.Text = "START MONITORING";
        APP_BTN_SET_COM.Enabled = true;
        APP_TXT_COM.Enabled = true;
        APP_TIMER.Enabled = m_EnableMonitoring;
    }
    else

```

```

        {
            m_EnableMonitoring = true;
            APP_BTN_ENABLE_MONITORING.Text = "STOP MONITORING";
            APP_BTN_SET_COM.Enabled = false;
            APP_TXT_COM.Enabled = false;
            APP_TIMER.Enabled = m_EnableMonitoring;
        }
    }

    private void APP_BTN_STARTCONTROLLER_Click(object sender,
    EventArgs e)
    {
        xValue = 0;
        Plot();
        string message;
        m_PID_Ctrl.Enable = 1;
        m_PID_Ctrl.SetPoint = (int)(100 *
    Convert.ToDouble(APP_TXT_SETPPOINT.Text.Trim()));
        m_PID_Ctrl.Kp = (int)(100 *
    Convert.ToDouble(APP_TXT_KP.Text.Trim()));
        m_PID_Ctrl.Ki = (int)(100 *
    Convert.ToDouble(APP_TXT_KI.Text.Trim()));
        m_PID_Ctrl.Kd = (int)(100 *
    Convert.ToDouble(APP_TXT_KD.Text.Trim()));

        message = "$$$";
        message += string.Format("{0:0}", m_PID_Ctrl.Enable);
        message += string.Format("{0:0000}", m_PID_Ctrl.SetPoint);
        message += string.Format("{0:000}", m_PID_Ctrl.Kp);
        message += string.Format("{0:000}", m_PID_Ctrl.Ki);
        message += string.Format("{0:000}", m_PID_Ctrl.Kd);
        message += string.Format("{0:0000}", 0);

        APP_BTN_SETPARAM.Enabled = false;
        SendSMS("988319429", message);
        APP_BTN_SETPARAM.Enabled = true;
    }

    private void APP_BTN_GETPARAM_Click(object sender, EventArgs
    e)
    {
        string[] ReceivedPackets = new string[0];
        APP_BTN_GETPARAM.Enabled = false;
        DeleteSMS(); // Elimina mensajes recibidos
        ReceivedPackets = CheckSMSList(); // Lee nuevos mensajes;

        int Counter = ReceivedPackets.Length;
        if (Counter>0)
        {
            for (int pX = 0; pX < Counter; pX++)
            {
                m_PID_Sts.Enable =
    Convert.ToInt32(ReceivedPackets[pX].Substring(3, 1));
                m_PID_Sts.SetPoint =
    Convert.ToInt32(ReceivedPackets[pX].Substring(4, 4));
                m_PID_Sts.Kp =
    Convert.ToInt32(ReceivedPackets[pX].Substring(8, 3));
                m_PID_Sts.Ki =
    Convert.ToInt32(ReceivedPackets[pX].Substring(11, 3));
            }
        }
    }

```

```

        m_PID_Sts.Kd =
Convert.ToInt32(ReceivedPackets[pX].Substring(14, 3));
        m_PID_Sts.Temperature =
Convert.ToInt32(ReceivedPackets[pX].Substring(17, 4));

        APP_TXT_TEMPERATURE.Text =
m_PID_Sts.Temperature.ToString();
    }
    }
    APP_BTN_GETPARAM.Enabled = true;
}

e)
private void APP_BTN_SETPARAM_Click(object sender, EventArgs
{
    // Configuración de parametros
    string message;
    m_PID_Ctrl.Enable = 0;
    m_PID_Ctrl.SetPoint = (int)(100 *
Convert.ToDouble(APP_TXT_SETPOINT.Text.Trim()));
    m_PID_Ctrl.Kp = (int)(100 *
Convert.ToDouble(APP_TXT_KP.Text.Trim()));
    m_PID_Ctrl.Ki = (int)(100 *
Convert.ToDouble(APP_TXT_KI.Text.Trim()));
    m_PID_Ctrl.Kd = (int)(100 *
Convert.ToDouble(APP_TXT_KD.Text.Trim()));

    message = "$$$";
    message += string.Format("{0:0}", m_PID_Ctrl.Enable);
    message += string.Format("{0:0000}", m_PID_Ctrl.SetPoint);
    message += string.Format("{0:000}", m_PID_Ctrl.Kp);
    message += string.Format("{0:000}", m_PID_Ctrl.Ki);
    message += string.Format("{0:000}", m_PID_Ctrl.Kd);
    message += string.Format("{0:0000}", 0);

    APP_BTN_SETPARAM.Enabled = false;
    SendSMS("988319429", message);
    APP_BTN_SETPARAM.Enabled = true;
}

private void SendSMS(string PhoneNumber, string Message)
{
    string AT_Command;
    APP_COMPORT.RtsEnable = true;
    APP_COMPORT.DtrEnable = false;
    AT_Command = "AT+CMGS=\"";
    AT_Command += PhoneNumber + "\" + '\r';
    APP_COMPORT.Write(AT_Command);
    Thread.Sleep(50);
    AT_Command = Message;
    APP_COMPORT.Write(AT_Command);
    Thread.Sleep(50);
    AT_Command = "";
    AT_Command += (char)26;
    APP_COMPORT.Write(AT_Command);
    Thread.Sleep(50);
    APP_COMPORT.ReadExisting();
}

```

```
private string[] CheckSMSList()
{
    string[] MessageList = new string[50];
    string[] MessageListResp = new string[0];

    string AT_Command;
    APP_COMPORT.RtsEnable = true;
    APP_COMPORT.DtrEnable = false;
    AT_Command = "AT+CMGL=\"REC UNREAD\" + '\r';//graba datos
    de mensajes
    APP_COMPORT.Write(AT_Command);
    Thread.Sleep(100);
    m_DataRead = "";
    while (APP_COMPORT.BytesToRead > 0)
    {
        m_DataRead += APP_COMPORT.ReadExisting();
        Thread.Sleep(50);
    }

    int StartIndex = 0;
    int Counter = 0;
    StartIndex = m_DataRead.IndexOf("$$$");
    while (StartIndex >= 0)
    {
        MessageList[Counter] =
m_DataRead.Substring(StartIndex, 21);
        m_DataRead = m_DataRead.Substring(StartIndex - 1 +
21);

        StartIndex = m_DataRead.IndexOf("$$$");
        Counter++;
    }

    MessageListResp= new string[Counter];

    while (Counter > 0)
    {
        Counter--;
        MessageListResp[Counter] = MessageList[Counter];
    }

    return MessageListResp;
}

private string ReadSMS(int MessageID)
{
    string Message = "";
    //

    return Message;
}

private void DeleteSMS()
{
    string AT_Command;
    APP_COMPORT.RtsEnable = true;
    APP_COMPORT.DtrEnable = false;
    AT_Command = "AT+CMGD=1,2" + '\r';
    APP_COMPORT.Write(AT_Command);
    Thread.Sleep(50);
}
```

```

        m_DataRead = APP_COMPORT.ReadExisting();
        Thread.Sleep(50);
        m_DataRead = "";
        while (APP_COMPORT.BytesToRead > 0)
        {
            m_DataRead += APP_COMPORT.ReadExisting();
            Thread.Sleep(50);
        }
    }

    private void Plot()
    {
        x[xValue] = xValue;
        y[xValue] = m_PID_Ctrl.SetPoint;
        z[xValue] = m_PID_Sts.Temperature;
        xValue++;
        if (xValue >= x.Length)
        {
            xValue = 0;
        }

        // Para eliminar los graficos actuales cada ves que se
        // reinicia
        APP_VIEWER.GraphPane.CurveList.Clear();

        // Graphane: Panel que contiene los datos de la gráfica.
        GraphPane myPane = APP_VIEWER.GraphPane;

        myPane.XAxis.Scale.Min = 0;
        myPane.XAxis.Scale.Max = 500;
        myPane.YAxis.Scale.Min = 0;
        myPane.YAxis.Scale.Max = 60;
        myPane.XAxis.Scale.MinorStep = 50;
        myPane.XAxis.Scale.MajorStep = 50;

        myPane.YAxis.MajorGrid.IsVisible = true;
        myPane.YAxis.MinorGrid.IsVisible = true;

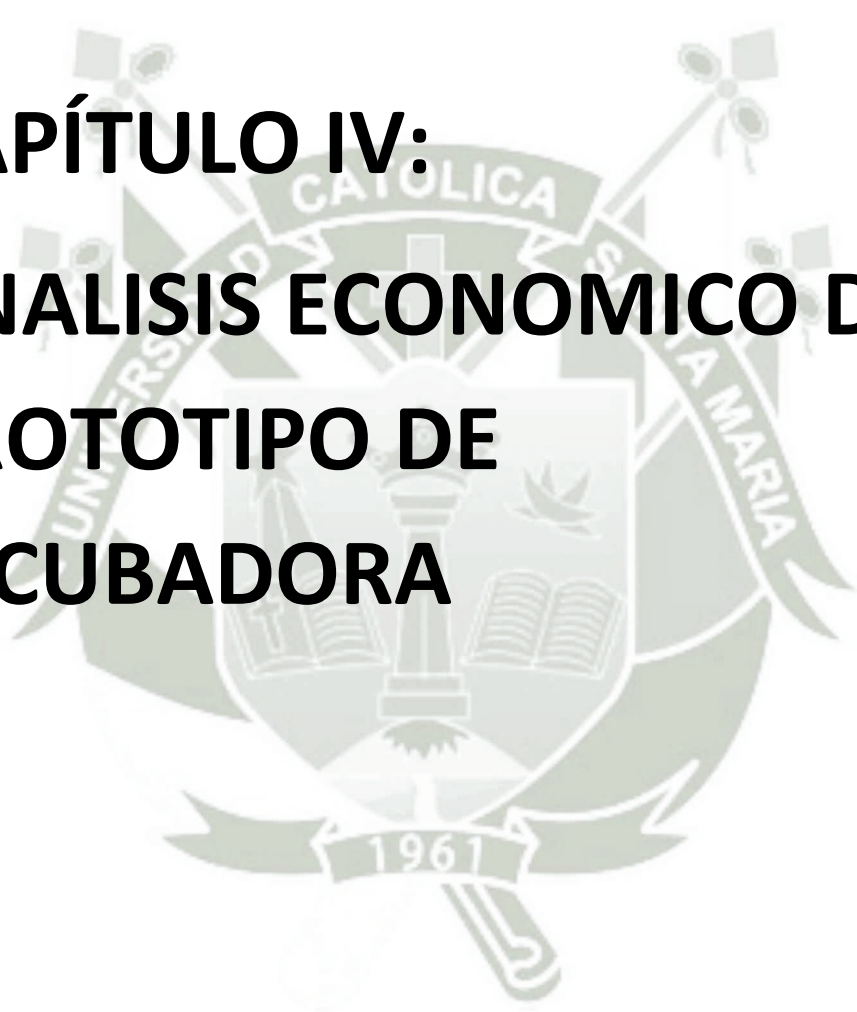
        // PointPairList: contiene los datos para el trazado de
        // los ejes X y Y
        PointPairList spl1 = new PointPairList(x, y);
        PointPairList spl2 = new PointPairList(x, z);

        // Añade el color de curvas al gráfico
        LineItem myCurve1 = myPane.AddCurve("Set point value",
spl1, Color.Red, SymbolType.None);
        LineItem myCurve2 = myPane.AddCurve("Temperature value",
spl2, Color.Blue, SymbolType.None);

        myCurve1.Line.Width = 3.0F;
        myCurve2.Line.Width = 3.0F;
        myPane.Title.Text = "Range time temperature";

        // Añade las tres funciones solo para comprobar si
        // estamos seguro que se refresca la trama
        APP_VIEWER.AxisChange();
        APP_VIEWER.Invalidate();
        APP_VIEWER.Refresh();
    }
}

```



CAPÍTULO IV: ANÁLISIS ECONOMICO DEL PROTOTIPO DE INCUBADORA

CAPITULO IV: ANÁLISIS ECONÓMICO DEL PROTOTIPO DE INCUBADORA

Se detalla el presupuesto elaborado para el “DISEÑO E IMPLEMENTACION DE UN CONTROLADOR PID DE TEMPERATURA CONTROLADO A DISTANCIA PARA UN PROTOTIPO DE INCUBADORA “.

Para el análisis económico se considera el costo de los dispositivos electrónicos que intervienen para la implementación, los costos de mano de obra, las herramientas empleadas, y costos adicionales.

El costo del módulo de temperatura se divide en dos etapas detalladas a continuación:

4.1 COSTOS DE LOS COMPONENTES PARA LA TARJETA

La distribución de todos los elementos y de los materiales que componen sólo las tarjetas del módulo de temperatura.

Los precios de los elementos de cada tarjeta incluido IGV y el costo de cada tarjeta se describen a continuación:

Tarjeta Control	Adaptador USB - Serial	TARJETA 1 (TARJETA MODEM PARA PC)				
		ITEM	DESCRIPCIÓN	CANTIDAD	PRECIO UNITARIO	PRECIO TOTAL
		1	Microcontrolador ATmega644	1	16	16
		2	Conector USB Macho	1	5	5
		3	Cristal de 20MHz	3	1.5	4.5
		4	Condensador de 22pF	10	0.1	1
		5	Condensador de 100nF	10	0.1	1
		6	Resistencia de 10k/0.25W	10	0.1	1
		7	Led rojo de 5mm	3	0.3	0.9
		8	Led verde de 5mm	3	0.3	0.9
		9	Espadín en 90° Hilera de 20	2	1.5	3
	Interfaz Trans-receptora Inalámbrico	10	Módulo GSM Fastrack Supreme 20 *	2	270	540
		11	Regulador Ajustable LM317T	3	1.2	3.6
		12	Opam KA358	4	3.0	12
		13	Zócalo para Opam	4	1.0	4.0
		14	Sensor de Temperatura LM35	4	5.0	20
		15	Condensador de 100nF	10	0.1	1.0
		16	Potenciómetro de Precisión de 10K	2	1.5	3.0

	17	Resistencia de 2k/0.25W a 1%	10	0.1	1.0
	18	Resistencia de 200/0.25W a 1%	10	0.1	1.0
	19	Cable Multiconductor Apantallado de 6 hilos	5	3.0	15
	20	Driver de Voltaje 74ls244	5	1.2	6.0
	21	Photo Triac IL410	3	3	9.0
	22	Triac BT136	3	2	6.0
	23	Condensador de 0.33uF	6	0.3	1.8
	Sub Total				656.9
	IGV (18 %)				118.2
	TOTAL				775.1

4.1.1 COSTO DE LA CAMARA TERMICA

El costo de la cámara térmica, comprende todo el material usado en la cámara, la cual se desea mantener la temperatura a un valor constante, el costo incluye el corte y pegado del acrílico.

CAMARA TERMICA (INCUBADORA)				
ITEM	DESCRIPCIÓN	CANTIDAD	PRECIO UNITARIO	PRECIO TOTAL
1	Caja de caja madera Medidas: 0.40 m x 0.35 m x 0.35 m	1	150	150
2	Plancha de acrílico Medidas: 0.25mx0.20m	1	15	15
			TOTAL	165

4.1.2 COSTO DE INGENIERIA

El costo de Ingeniera incluye todo el tiempo de investigación (2 horas diarias por 24 días), pruebas previas al enrutado de placas electrónicas (2 horas diarias por 10 días), tiempo de diseño de placas (4 horas diarias por 2 días), programación de microcontroladores (4 horas diarias por 3 días), simulaciones en computadora (2 horas diarias por 3 días).

COSTO DE INGENIERIA				
ITEM	DESCRIPCIÓN	HORAS	PRECIO UNITARIO	PRECIO TOTAL
1	Tiempo de Investigación	48	10	480
2	Pruebas previas al enrutado de placas electrónicas	20	10	200
3	Tiempo de diseño de placas	8	10	80

4	Programación de microcontrolador	12	10	120
5	Simulaciones en computadora	6	10	60
			TOTAL	940

La siguiente tabla describe todos aquellos costos externos para la implementación, el costo será referencial, se tomó preguntando en diversas casas de fabricación de placas electrónicas, varían un poco con respecto a soldadura⁶ también al acabado y tamaño que el cliente desee:

COSTOS ADICIONALES				
Tarjeta de Control	DESCRIPCIÓN	CANTIDAD	PRECIO UNITARIO	PRECIO TOTAL
	Confección	1	100	100
	Soldadura	1	50	50
			TOTAL	150

Los elementos del prototipo son de bajo costo y fáciles de encontrar en el mercado interno, a excepción del módulo de transmisión GSM FASTRACK SUPREME 20; éste tiene un valor elevado con respecto a los demás componentes, además este tiene que ser importado, pero para realizar esta tesis fueron proporcionado por la universidad.

Además de los materiales empleados tendremos que incluir el costo de papel, impresiones, anillados que van a ser usados en nuestra investigación y presentación:

⁶ En el proceso de soldadura se debe tomar en cuenta , el precio del ácido ferroso , de la baquelita , el ruteo de pistas , pomada de soldar , el estaño , pistola de soldar o cautín, esponja metálica,thinner , etc.

COSTO DE MATERIALES PARA INVESTIGACION				
ITEM	DESCRIPCIÓN	CANTIDAD	PRECIO UNITARIO	PRECIO TOTAL
1	Papel Bond	2	14	28
2	Impresiones	700	0.08	56
3	Anillados	3	2.5	7.5
TOTAL				91.5

4.2 COSTO TOTAL DEL PROTOTIPO DE INCUBADORA CONTROLADA REMOTAMENTE

El costo total será la suma de todos los componentes electrónicos que se usan para la fabricación del proyecto, de los materiales para la investigación, la cámara térmica, las confecciones de las tarjetas (diseño e implementación) así también como la ingeniería

COSTO DE MATERIALES PARA INVESTIGACION			
ITEM	DESCRIPCIÓN	CANTIDAD	TOTAL
1	Costos de los componentes para Tarjeta	1	775.1
2	Costos de la cámara térmica	1	165
3	Costos de Ingeniería	1	940
4	Costos Adicionales	1	340
5	Costos de Materiales para la Investigación	1	81.50
TOTAL			2241.6

CAPÍTULO V: RESULTADOS DEL PROYECTO



CAPITULO V: RESULTADOS DE PROYECTO

En este capítulo se detalla los resultados del proyecto, las pruebas antes de ponerlo en marcha y además imágenes de la cámara térmica a controlar.

5.1 PUESTA EN FUNCIONAMIENTO Y PRUEBAS

Primero se realizan las pruebas en modo manual:

- 1) Probar la comunicación entre PC y modem utilizando comandos AT+ por medio del Hyperterminal.
- 2) Después de lograr la conexión, el programa en Visual responde muy bien a la recepción de datos, logrando así los datos en la pantalla y enviar información al microcontrolador.
- 3) El microcontrolador recibe los datos y realiza la acción de control de temperatura. Recibe los datos del sensor y lo imprime al puerto serial para que la aplicación en la computadora muestre la temperatura.

5.1.1 PUESTA EN FUNCIONAMIENTO

A continuación se mostrará las fotos del prototipo:



Fig.69. Cámara Térmica

El interior constará de dos focos como se señaló en los primeros capítulos y además



Fig.70. Interior de Cámara Térmica

En el lado derecho la ubicación del ventilador para una circulación de calor uniforme en toda la cámara



Fig.71. Posición del Ventilador

Puesta en marcha de nuestra planta a lazo abierto , el tiempo aproximado para que llegue a los 37.5°C es de 10 minutos, además la cámara térmica es hermética y la circulación de aire es a 2V del ventilador para que éste no enfríe el interior.

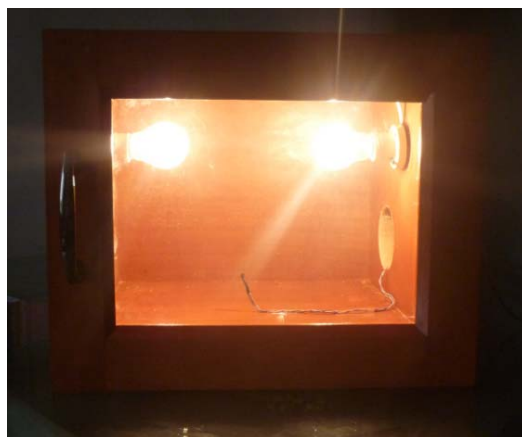


Fig 72. Cámara térmica a lazo abierto

5.1.2 PRUEBA FINAL

Esta prueba se realiza bajo condiciones ambientales similares a la temperatura que aproximamos en el diseño por computadora (20°C).

- Temperatura ambiente en la prueba: $T_{\text{AMB}}=18^{\circ}\text{C}$.
- Voltaje de la red: $V_{\text{red}} = 220 \text{ V}$.
- Temperatura del Set point: $T_{\text{SP}} = 37.5^{\circ}\text{C}$.
- Tiempo de la prueba: 15 min.
- Envío de datos: Via GSM.
- Tipo de control: Automático

Nº Prueba	Temperatura ($^{\circ}\text{C}$)	Tiempo (seg)
1	0	0
2	20.3	130
3	22.1	148
4	24.9	190
5	30.2	250
6	34.4	300
7	39.2	540
8	37.9	610

9	37.5	660
10	37.3	720
11	37.3	780
12	37.4	840
13	37.3	900
14	37.4	1000

Tabla 6. Datos obtenidos del controlador en prueba Final

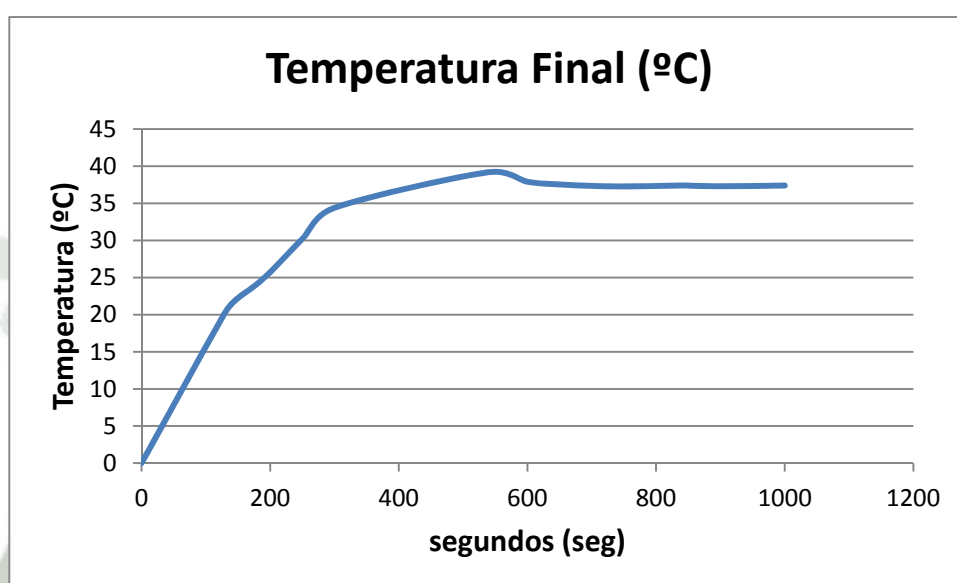


Fig.73. Comportamiento del controlador en prueba Final

5.1.3 COMPARACIÓN DE TEMPERATURAS

Las temperaturas que se tomarán en comparación son las realizadas en la prueba final y aquellas que se obtuvieron por simulación en computadora previa a la etapa de implementación del controlador.

Nº Prueba	Tiempo (seg)	Temperatura Final del Prototipo (°C)	Temperatura Diseño por computadora (°C)
1	0	0	0
2	130	20.3	14.8
3	148	22.1	16.9

4	190	24.9	22.9
5	250	30.2	27
6	300	34.4	32.2
7	540	39.2	41.1
8	610	37.9	40.2
9	660	37.5	40
10	720	37.3	39
11	780	37.3	37.7
12	840	37.4	37.5
13	900	37.3	37.5
14	1000	37.4	37.5

Tabla 7. Comparación de temperaturas de control implementado vs. Controlador diseñado por computador

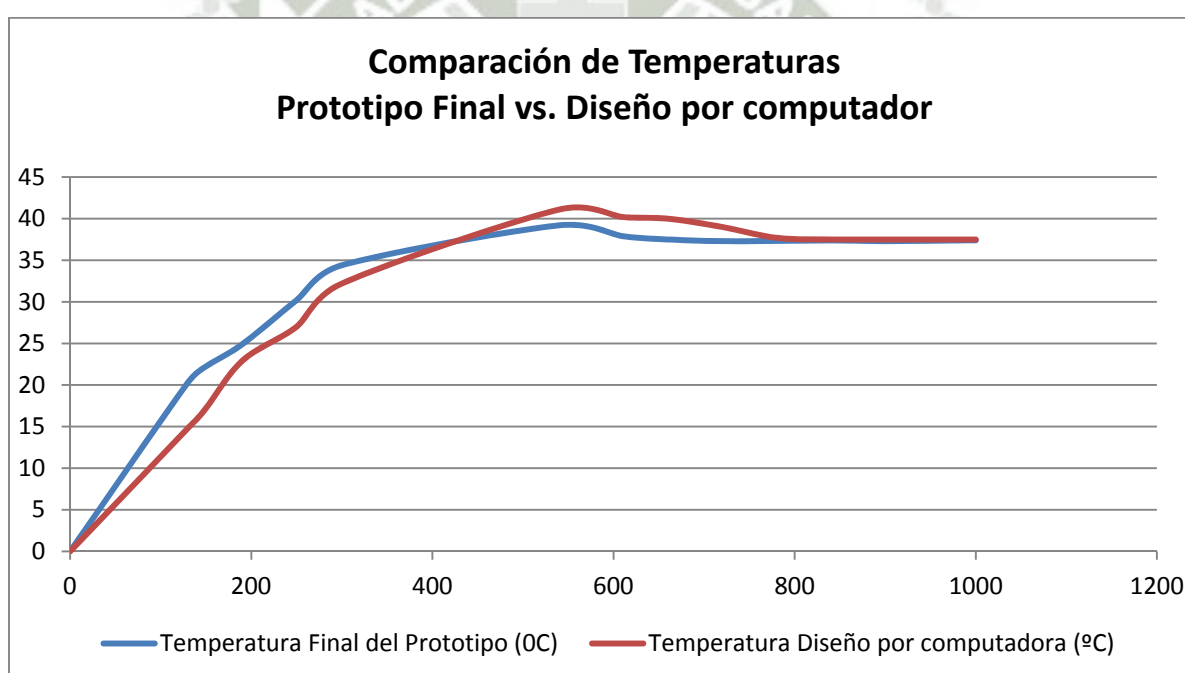


Fig.74. Comparación de temperaturas de control implementado vs. Controlador diseñado por computador

El diseño del controlador que se ha implementado en el prototipo es de comportamiento muy parecido a lo propuesto por el diseño en el computador, esto se esperaba, debido al uso de herramientas computacionales como

matlab que brinda la posibilidad de las simulaciones, lo cual reduce el tiempo en el diseño de controladores.



CONCLUSIONES Y RECOMENDACIONES



CONCLUSIONES

- Al diseñar, construir e implementar el prototipo, se concluye que el prototipo de incubación, sí controla la temperatura a distancia para crear una condición ambiental de temperatura óptima para la cámara térmica.
- Se usó un algoritmo PID ya que este algoritmo es el más usado en comparación con otras teorías de control aplicadas a la industria, además su sobre-impulso está dentro del 25% permitido y su comportamiento es el más estable posible dentro de todo el proceso de incubación.
- El envío de los datos por la red GSM es la más estable posible, y por medio de esta se puede controlar procesos telemétricos.
- Se concluye también que se puede emplear otro tipo de actuador como pueden ser resistencia, infrarrojos, etc. de mayor tiempo de funcionamiento.
- El uso de lenguajes de programación de alto nivel (Matlab, Visual Basic, Labview, Compiladores de programación para microcontrolador, etc.) facilita ampliamente la implementación de controlador PID, u otro tipo de esquema del controlador sobre dispositivos microcontrolados, abriendo así una alternativa de desarrollo de sistemas de control. Éstos lenguajes permiten fácilmente adicionar periféricos para ingreso de datos y supervisión de variables de estado del sistema del controlador como también se reduce el tiempo de sintonización del PID al emplear simulaciones en PC.

RECOMENDACIONES

- El diseño y la construcción de una incubadora a un costo más razonable abastecerían al mercado con una incubadora acorde con las necesidades de los pequeños avicultores, cumpliendo de esta manera el papel de la universidad para con la sociedad, dando así una solución real a los muy diversos problemas que se presentan en la industria.
- Se espera que este proyecto de investigación sea la motivación para que se sigan desarrollando este tipo de trabajos dentro de la universidad enfocados hacia el sector avícola con el fin de aportar cada día más herramientas para el desarrollo del sector en el país.
- Si se decide implementar este prototipo en la industria, es recomendable que se haga la cámara térmica de acero inoxidable y los sensores de medición serían termistores.
- Con el desarrollo de este tipo de proyectos en los que uno como estudiante trata de dar soluciones a problemas reales de la vida cotidiana, demuestra que la ingeniería mecatrónica abarca los más diversos campos de aplicación general, en donde es posible ofrecer excelentes soluciones económicamente viables.
- Una vez comprobada la comunicación del control de temperatura PID, es recomendable emplear tecnología gprs y contratar un paquete de datos para poder controlar un número ilimitado de incubadora a la vez.

REFERENCIAS BIBLIOGRAFICAS



REFERENCIAS BIBLIOGRAFICAS

- KUO, Benjamín, SISTEMAS AUTOMÁTICOS DE CONTROL, 2da Edición, SA. DE CV México.
- OGATA, Katsuhiko, Ingeniería de Control Moderna, 4da edición, Prentice Hall, España 2003.
- AMÉSTEGUI, Mauricio. APUNTES DE CONTROL PID. Universidad Mayor de San Andrés (Bolivia), 2001.
http://www.alumnos.usm.cl/~ignacio.morande/descargas/apuntes_de_control_pid.pdf consultado al 20/10/2012
- ANGULO Usategui José, Microcontroladores PIC, 4ta Ed, Mc Graw Hill, 2007.
- LEWIS, Paul, YANG, Chang, Sistemas de Control en Ingeniería, 3ra edición, Prentice Hall, 2009.
- CREUS, Antonio Sole, Instrumentación Industrial, 7ta edición, Alfaomega Marcombo, Barcelona España 2006.
- Manual de VISUAL C# 2005,
http://www.inele.ufro.cl/apuntes/Manuales/Manual%20Empezando%20_Con%20_VisualC#.pdf consultado al 31/10/12
- KUO, Benjamín, SISTEMAS AUTOMÁTICOS DE CONTROL, 2da Edición, SA. DE CV México.
- ANGULO Usategui José, dsPIC. Diseño Práctico de Aplicaciones, 4ta Ed, Mc Graw Hill, 2009.

- NISE, Norman S., “Sistemas de Control para Ingeniería”. 3ra edición. CECSA, 2002
- OGATA, Katsuhiko, “Sistemas de control en tiempo discreto”. Pearson Educación, 1996.
- ZADEH, Lofti. A., “From circuit theory to system theory”. In Proc. IRE 50, pages 856–865, 1962.
- RIVERA, D.E. y M.W. Braun., “System Identification for Process Applications”. Arizona State University. 2005.
- LENNART, Ljung, “System Identification: Theory for the User”. Prentice-Hall, 1987.
- LENNART, Ljung, Torkel Glad. “Modeling of Dynamic Systems”. Prentice-Hall, 1994.
- BOTERO, Héctor y Ramírez, José. “IDENTIFICACIÓN DE SISTEMAS DE EXCITACIÓN: ANÁLISIS DETALLADO DE METODOLOGÍA Y RESULTADOS”. Artículo académico, Dyna, Año 75, Nro. 156, pp. 65-77. Medellín, Noviembre de 2008. ISSN 0012-7353
- LENART, Ljung, “Matlab User’s Guide: System Identification Toolbox”. Prentice Hall, 1988
- DORF, Richard, “Sistemas de control moderno”, 10ma edición, Pearson-Prentice Hall, 2005
- RASHID, Muhammad H., “Electrónica de Potencia. Circuitos, dispositivos y aplicaciones ”. 2da edición 1995. México: Prentice Hall Hispanoamerica, S.A.

- BENAVENT Garcia, J. Manuel. y A. Abellán, E. Figueres.
“Electrónica de Potencia. Teoría y Aplicaciones”. Prentice Hall, 1999.
- <http://www.arqhys.com/construccion/madera-caracteristicas.html>
- DIAZ Tapia. Nelson. Monografía. “Transferencia de calor”.
<http://www.monografias.com/trabajos15/transf-calor/transf-calor.shtml> fecha
de consulta: Noviembre del 2012.

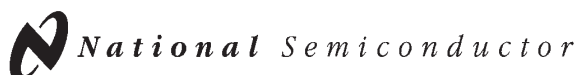




ANEXOS

ANEXO 1





December 1994

LM35/LM35A/LM35C/LM35CA/LM35D

Precision Centigrade Temperature Sensors

General Description

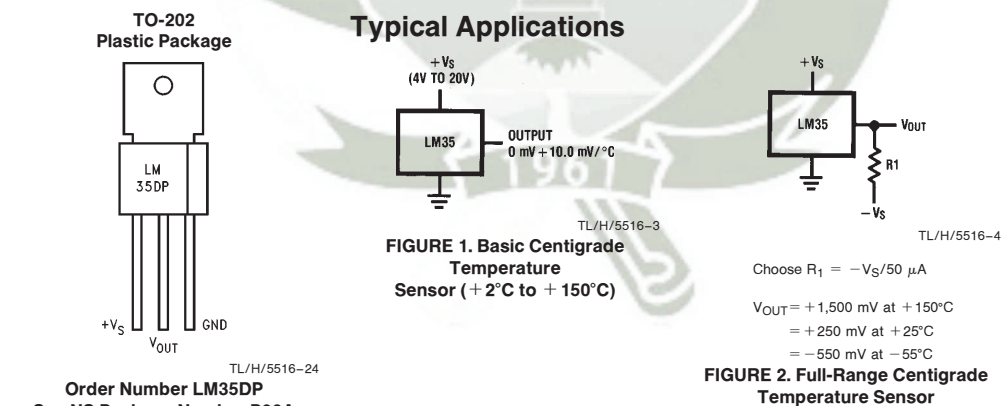
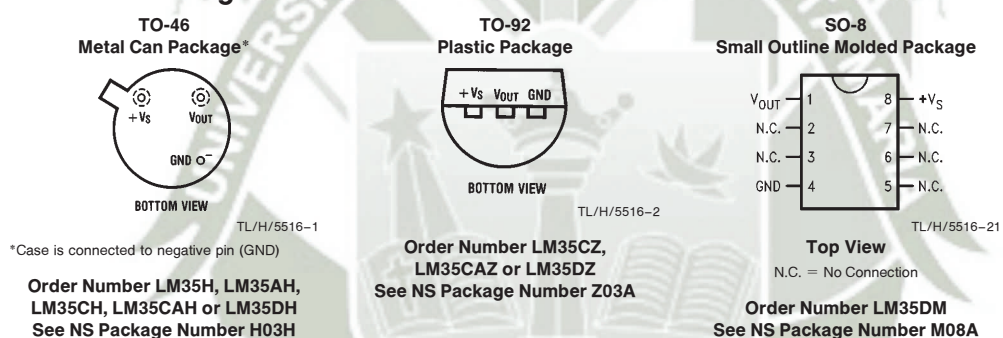
The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling. The LM35 does not require any external calibration or trimming to provide typical accuracies of $\pm 1/4^{\circ}\text{C}$ at room temperature and $\pm 3/4^{\circ}\text{C}$ over a full -55 to $+150^{\circ}\text{C}$ temperature range. Low cost is assured by trimming and calibration at the wafer level. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only $60\text{ }\mu\text{A}$ from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a -55°C to $+150^{\circ}\text{C}$ temperature range, while the LM35C is rated for a -40°C to $+110^{\circ}\text{C}$ range (-10°C with improved accuracy). The LM35 series is

available packaged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-202 package.

Features

- Calibrated directly in ° Celsius (Centigrade)
- Linear $+10.0\text{ mV}/^{\circ}\text{C}$ scale factor
- 0.5°C accuracy guaranteeable (at $+25^{\circ}\text{C}$)
- Rated for full -55°C to $+150^{\circ}\text{C}$ range
- Suitable for remote applications
- Low cost due to wafer-level trimming
- Operates from 4 to 30 volts
- Less than $60\text{ }\mu\text{A}$ current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only $\pm 1/4^{\circ}\text{C}$ typical
- Low impedance output, $0.1\text{ }\Omega$ for 1 mA load

Connection Diagrams



TRI-STATE® is a registered trademark of National Semiconductor Corporation.

©1995 National Semiconductor Corporation TL/H/5516

RRD-B30M75/Printed in U. S. A.

LM35/LM35A/LM35C/LM35CA/LM35D
Precision Centigrade Temperature Sensors

Electrical Characteristics (Note 1) (Note 6) (Continued)

Parameter	Conditions	LM35			LM35C, LM35D			Units (Max.)
		Typical	Tested Limit (Note 4)	Design Limit (Note 5)	Typical	Tested Limit (Note 4)	Design Limit (Note 5)	
Accuracy, LM35, LM35C (Note 7)	$T_A = +25^{\circ}\text{C}$	± 0.4	± 1.0		± 0.4	± 1.0		$^{\circ}\text{C}$
	$T_A = -10^{\circ}\text{C}$	± 0.5			± 0.5		± 1.5	$^{\circ}\text{C}$
	$T_A = T_{\text{MAX}}$	± 0.8	± 1.5		± 0.8		± 1.5	$^{\circ}\text{C}$
	$T_A = T_{\text{MIN}}$	± 0.8		± 1.5	± 0.8		± 2.0	$^{\circ}\text{C}$
Accuracy, LM35D (Note 7)	$T_A = +25^{\circ}\text{C}$				± 0.6	± 1.5		$^{\circ}\text{C}$
	$T_A = T_{\text{MAX}}$				± 0.9		± 2.0	$^{\circ}\text{C}$
	$T_A = T_{\text{MIN}}$				± 0.9		± 2.0	$^{\circ}\text{C}$
Nonlinearity (Note 8)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.3		± 0.5	± 0.2		± 0.5	$^{\circ}\text{C}$
Sensor Gain (Average Slope)	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	$+ 10.0$	$+ 9.8, + 10.2$		$+ 10.0$		$+ 9.8, + 10.2$	mV/ $^{\circ}\text{C}$
Load Regulation (Note 3) $0 \leq I_L \leq 1 \text{ mA}$	$T_A = +25^{\circ}\text{C}$	± 0.4	± 2.0		± 0.4	± 2.0		mV/mA
	$T_{\text{MIN}} \leq T_A \leq T_{\text{MAX}}$	± 0.5		± 5.0	± 0.5		± 5.0	mV/mA
Line Regulation (Note 3)	$T_A = +25^{\circ}\text{C}$	± 0.01	± 0.1		± 0.01	± 0.1		mV/V
	$4\text{V} \leq V_S \leq 30\text{V}$	± 0.02		± 0.2	± 0.02		± 0.2	mV/V
Quiescent Current (Note 9)	$V_S = +5\text{V}, +25^{\circ}\text{C}$	56	80		56	80		μA
	$V_S = +5\text{V}$	105		158	91		138	μA
	$V_S = +30\text{V}, +25^{\circ}\text{C}$	56.2	82		56.2	82		μA
	$V_S = +30\text{V}$	105.5		161	91.5		141	μA
Change of Quiescent Current (Note 3)	$4\text{V} \leq V_S \leq 30\text{V}, +25^{\circ}\text{C}$	0.2	2.0		0.2	2.0		μA
	$4\text{V} \leq V_S \leq 30\text{V}$	0.5		3.0	0.5		3.0	μA
Temperature Coefficient of Quiescent Current		$+ 0.39$		$+ 0.7$	$+ 0.39$		$+ 0.7$	$\mu\text{A}/^{\circ}\text{C}$
Minimum Temperature for Rated Accuracy	In circuit of Figure 1, $I_L = 0$	$+ 1.5$		$+ 2.0$	$+ 1.5$		$+ 2.0$	$^{\circ}\text{C}$
Long Term Stability	$T_J = T_{\text{MAX}}$, for 1000 hours	± 0.08			± 0.08			$^{\circ}\text{C}$

Note 3: Regulation is measured at constant junction temperature, using pulse testing with a low duty cycle. Changes in output due to heating effects can be computed by multiplying the internal dissipation by the thermal resistance.

Note 4: Tested Limits are guaranteed and 100% tested in production.

Note 5: Design Limits are guaranteed (but not 100% production tested) over the indicated temperature and supply voltage ranges. These limits are not used to calculate outgoing quality levels.

Note 6: Specifications in **boldface** apply over the full rated temperature range.

Note 7: Accuracy is defined as the error between the output voltage and $10\text{mV}/^{\circ}\text{C}$ times the device's case temperature, at specified conditions of voltage, current, and temperature (expressed in $^{\circ}\text{C}$).

Note 8: Nonlinearity is defined as the deviation of the output-voltage-versus-temperature curve from the best-fit straight line, over the device's rated temperature range.

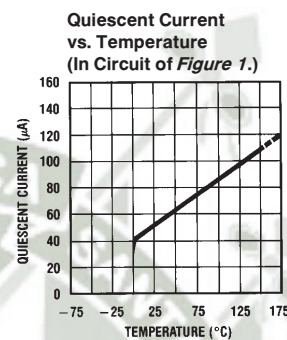
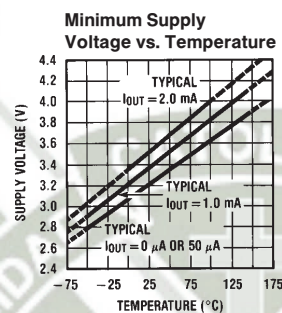
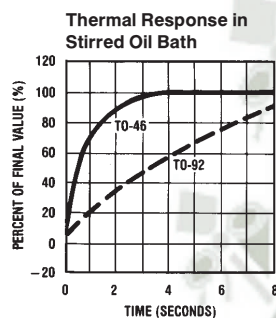
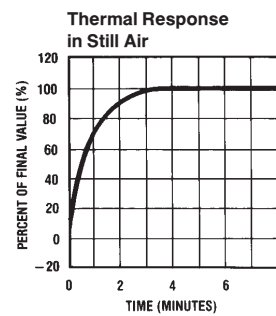
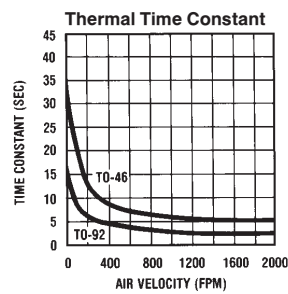
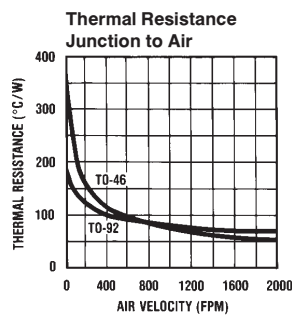
Note 9: Quiescent current is defined in the circuit of Figure 1.

Note 10: Absolute Maximum Ratings indicate limits beyond which damage to the device may occur. DC and AC electrical specifications do not apply when operating the device beyond its rated operating conditions. See Note 1.

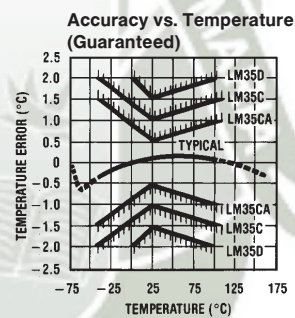
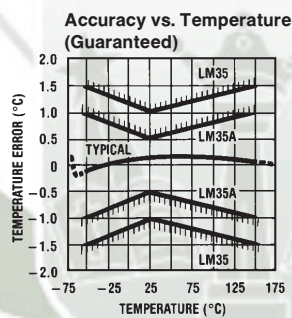
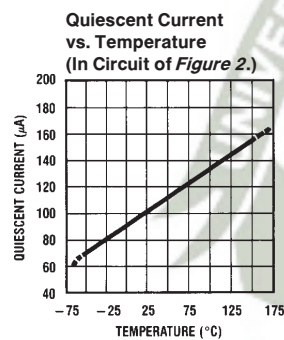
Note 11: Human body model, 100 pF discharged through a 1.5 k Ω resistor.

Note 12: See AN-450 "Surface Mounting Methods and Their Effect on Product Reliability" or the section titled "Surface Mount" found in a current National Semiconductor Linear Data Book for other methods of soldering surface mount devices.

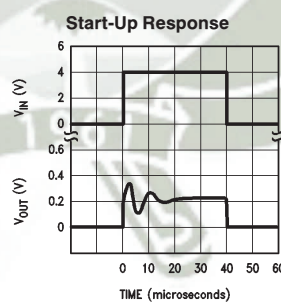
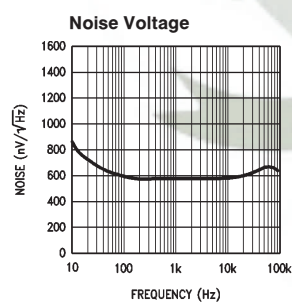
Typical Performance Characteristics



TL/H/5516-17



TL/H/5516-18



TL/H/5516-22

Typical Applications (Continued)

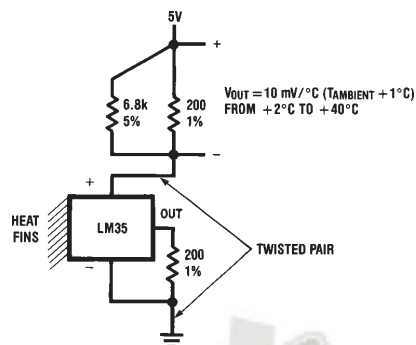


FIGURE 5. Two-Wire Remote Temperature Sensor (Grounded Sensor)

TL/H/5516-5

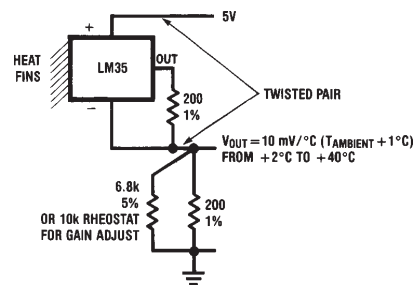


FIGURE 6. Two-Wire Remote Temperature Sensor (Output Referred to Ground)

TL/H/5516-6

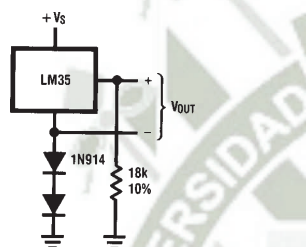


FIGURE 7. Temperature Sensor, Single Supply, -55°C to +150°C

TL/H/5516-7

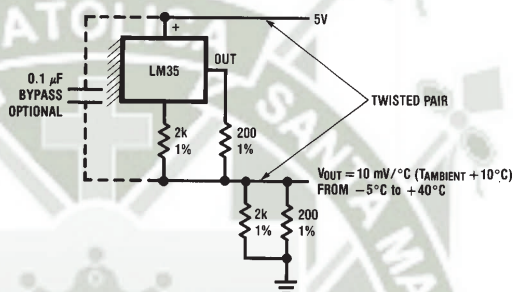


FIGURE 8. Two-Wire Remote Temperature Sensor (Output Referred to Ground)

TL/H/5516-8

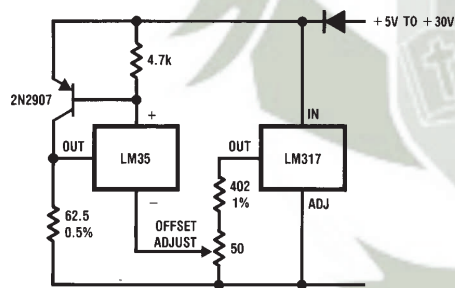


FIGURE 9. 4-To-20 mA Current Source (0°C to +100°C)

TL/H/5516-9

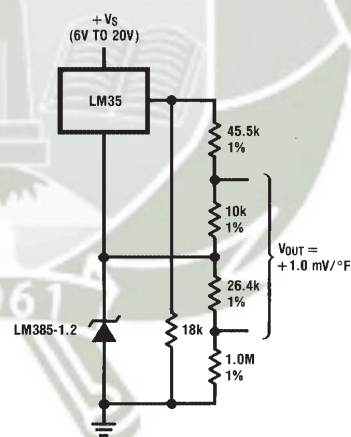


FIGURE 10. Fahrenheit Thermometer

TL/H/5516-10

ANEXO 2



Features

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller
- Advanced RISC Architecture
 - 131 Powerful Instructions – Most Single-clock Cycle Execution
 - 32 × 8 General Purpose Working Registers
 - Fully Static Operation
 - Up to 20 MIPS Throughput at 20MHz
- High Endurance Non-volatile Memory segments
 - 64 Kbytes of In-System Self-programmable Flash program memory
 - 2 Kbytes EEPROM
 - 4 Kbytes Internal SRAM
 - Write/Erase cycles: 10,000 Flash/100,000 EEPROM⁽¹⁾⁽³⁾
 - Data retention: 20 years at 85°C/100 years at 25°C⁽²⁾⁽³⁾
 - Optional Boot Code Section with Independent Lock Bits
 - In-System Programming by On-chip Boot Program
 - True Read-While-Write Operation
 - Programming Lock for Software Security
- JTAG (IEEE std. 1149.1 Compliant) Interface
 - Boundary-scan Capabilities According to the JTAG Standard
 - Extensive On-chip Debug Support
 - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface
- Peripheral Features
 - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
 - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
 - Real Time Counter with Separate Oscillator
 - Six PWM Channels
 - 8-channel, 10-bit ADC
 - Differential mode with selectable gain at 1x, 10x or 200x
 - Byte-oriented Two-wire Serial Interface
 - One Programmable Serial USART
 - Master/Slave SPI Serial Interface
 - Programmable Watchdog Timer with Separate On-chip Oscillator
 - On-chip Analog Comparator
 - Interrupt and Wake-up on Pin Change
- Special Microcontroller Features
 - Power-on Reset and Programmable Brown-out Detection
 - Internal Calibrated RC Oscillator
 - External and Internal Interrupt Sources
 - Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby
- I/O and Packages
 - 32 Programmable I/O Lines
 - 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF
- Speed Grades
 - ATmega644V: 0 - 4MHz @ 1.8V - 5.5V, 0 - 10MHz @ 2.7V - 5.5V
 - ATmega644: 0 - 10MHz @ 2.7V - 5.5V, 0 - 20MHz @ 4.5V - 5.5V
- Power Consumption at 1MHz, 3V, 25°C
 - Active: 240µA @ 1.8V, 1MHz
 - Power-down Mode: 0.1µA @ 1.8V

Notes: 1. Worst case temperature. Guaranteed after last write cycle.
2. Failure rate less than 1 ppm.
3. Characterized through accelerated tests.



**8-bit Atmel
Microcontroller
with 64K Bytes
In-System
Programmable
Flash**

ATmega644/V

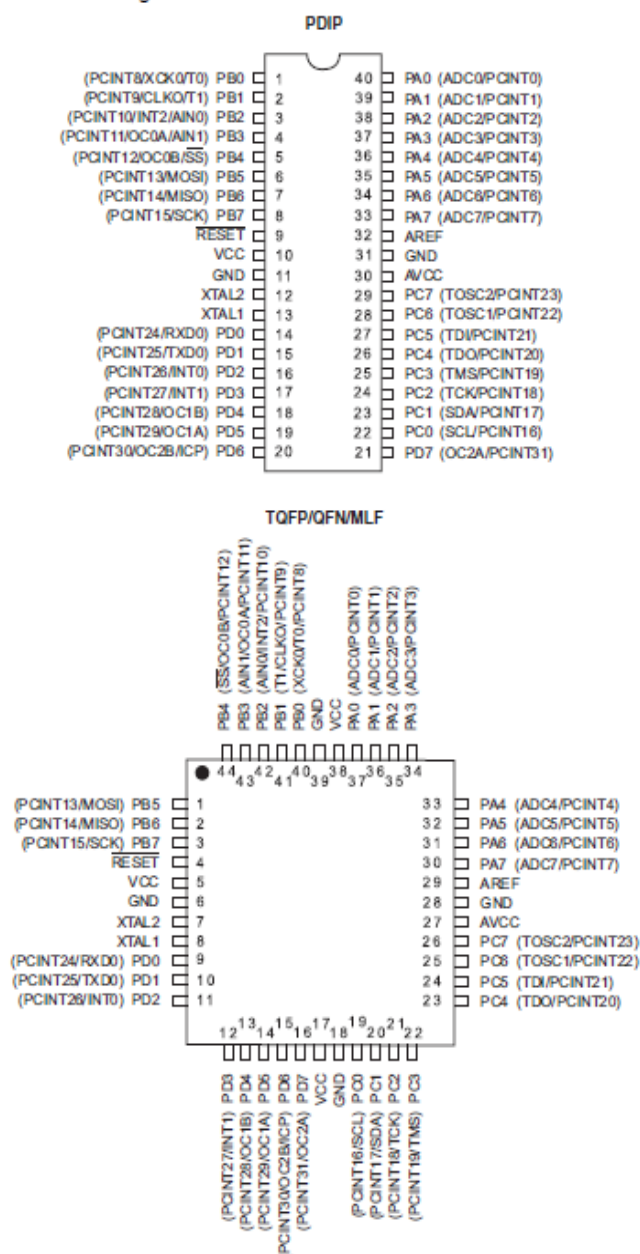
2593O-AVR-02/12





1. Pin Configurations

Figure 1-1. Pinout ATmega644



Note: The large center pad underneath the QFN/MLF package should be soldered to ground on the board to ensure good mechanical stability.

ANEXO 3



Fastrack Supreme User Guide

WA_DEV_Fastrk_UGD_001

Plug and Play Wireless CPU®**FASTRACK Supreme
User Guide**

Reference: WA_DEV_Fastrk_UGD_001

Revision: 001e

Date: 5 june, 2007



Supports Open AT® embedded ANSI C applications

Fastrack Supreme User Guide WA_DEV_Fastrk_UGD_001

Document History

Revision	Date	List of revisions	
001	9 February, 07	First Issue	
001a	23 February, 2007	Update DC cable GPIO mapping, add AutoShutDown	
001b	21 May, 07	Add detail of IES, RTC and serial port autoshtutdown	
001c	1 Jun, 07	Change to Quad Band	
001d	4 Jun, 07	Update label/packaging photo	
001e	5 Jun, 07	Comment	



Fastrack Supreme User Guide

General Presentation

3 General Presentation

3.1 Description

The FASTRACK Supreme description is given in the Figure 4 below.

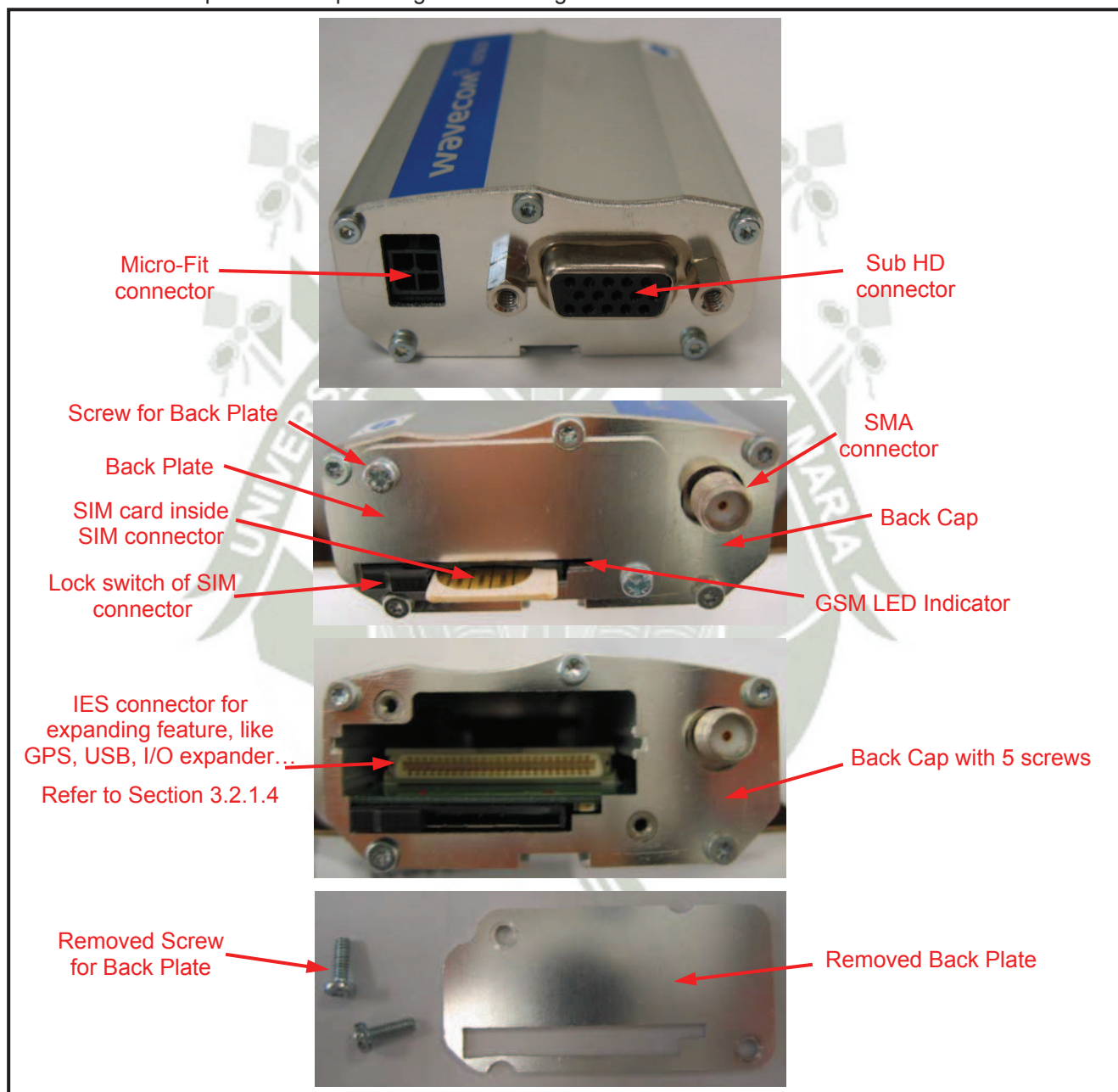


Figure 4: FASTRACK Supreme general description

Fastrack Supreme User Guide

General Presentation



Figure 8: Sub HD 15-pin connector

Table 2: Sub HD 15-pin connector description

Pin #	Signal (CCITT / EIA)	I/O	I/O type	Description	Comment
1	CDCD/CT109	O	STANDARD RS232	RS232 Data Carrier Detect	
2	CTXD/CT103	I	STANDARD RS232	RS232 Transmit serial data	
3	BOOT	I	CMOS	Boot	This signal must not be connected. Its use is strictly reserved to Wavecom or competent retailers.
4	CMIC2P	I	Analog	Microphone positive line	
5	CMIC2N	I	Analog	Microphone negative line	
6	CRXD/CT104	O	STANDARD RS232	RS232 Receive serial data	
7	CDSR/CT107	O	STANDARD RS232	RS232 Data Set Ready	
8	CDTR/CT108-2	I	STANDARD RS232	RS232 Data Terminal Ready	
9	GND	-	GND	Ground	

Fastrack Supreme User Guide

General Presentation

Pin #	Signal (CCITT / EIA)	I/O	I/O type	Description	Comment
10	CSPK2P	O	Analog	Speaker positive line	
11	CCTS/CT106	O	STANDARD RS232	RS232 Clear To Send	
12	CRTS/CT105	I	STANDARD RS232	RS232 Request To Send	
13	CRI/CT125	O	STANDARD RS232	RS232 Ring Indicator	
14	RESET	I/O	Schmitt	Supreme Plug & Play reset	Active low
15	CSPK2N	O	Analog	Speaker negative line	

3.2.1.4 IES Connector

The IES connector is a 50 pins board-to-board connector for expanding application features like GPS, USB, I/O expander... Currently there are already 3 IESM boards available for customer to expand the FASTRACK Supreme features immediately. They are:

- IESM-GPS+USB+I/O
- IESM-GPS+USB
- IESM-USB+I/O

For detail, please refer to Document in Section 1.1.5.

Fastrack Supreme User Guide

Features and Services

4.2.5 SIM Card Lock Feature

The FASTRACK Supreme has now implemented a SIM connector having a carrier with lock. This helps ensuring the user to have proper SIM card insertion and locked before proper use of GSM network.

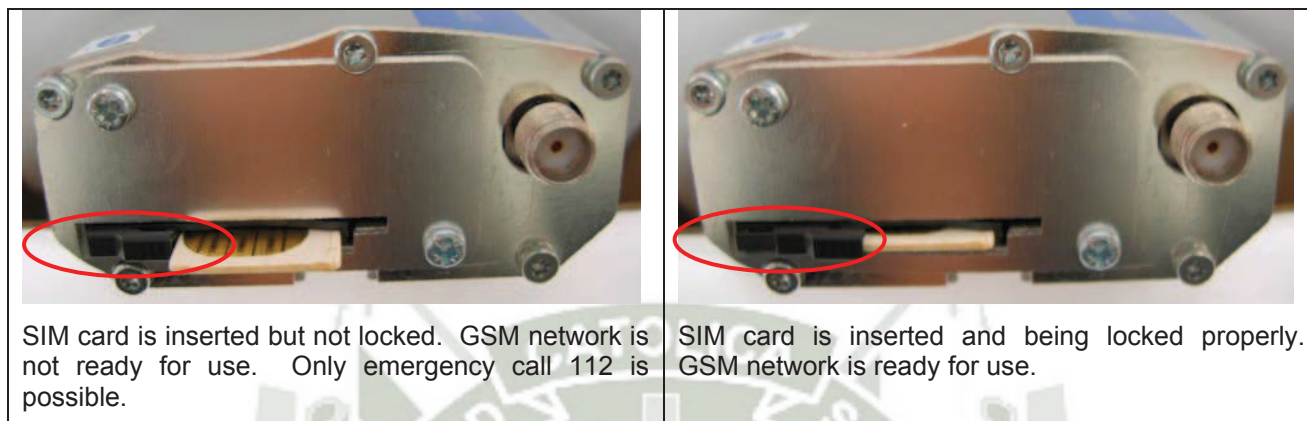


Figure 11: SIM card lock feature

Fastrack Supreme User Guide

Using the FASTRACK Supreme Plug & Play

5.1.3 Check the communication with the FASTRACK Supreme

To check the communication with the FASTRACK Supreme, do the following operations:

- Connect the RS232 link between the DTE (port COM) and the FASTRACK Supreme (DCE).
- Configure the RS232 port of the DTE as follows:
 - Bits per second: **115.200 bps**,
 - Data bits: **8**,
 - Parity: **None**,
 - Stop bits: **1**,
 - Flow control: **hardware**.
- Using a communication software such as a HyperTerminal, enter the **AT+J** command. The response of the FASTRACK Supreme must be **OK** displayed in the HyperTerminal window.
- If the communication cannot be established with the FASTRACK Supreme, do the following:
 - Check the RS232 connection between the DTE and the FASTRACK Supreme (DCE),
 - Check the configuration of the port COM used on the DTE.
- Example of AT commands which can be used after getting started the FASTRACK Supreme:
 - **AT+CGMI**: FASTRACK Supreme answer is "WAVECOM MODEM" when serial link is OK.
 - **AT+CPIN=xxxx**: to enter a PIN code xxxx (if activated).
 - **AT+CSQ**: to verify the received signal strength.
 - **AT+CREG?**: to verify the registration of the FASTRACK Supreme Plug & Play on the network.
 - **ATD<phone number>;**: to initiate a voice call.
 - **ATH**: to hang up (end of call).

For further information on these AT commands and their associated parameters, refer to "AT Commands Interface Guide" [6].

5.1.4 Reset the FASTRACK Supreme

To reset the FASTRACK Supreme, a hardware reset signal is available on pin 14 of the Sub HD 15-pin connector (RESET).

The FASTRACK Supreme reset is carried out when this pin is low for at least 200 μ s.

Warning This signal has to be considered as an emergency reset only. For further details on the FASTRACK Supreme reset, refer to Section 7.7.

Fastrack Supreme User Guide

Using the FASTRACK Supreme Plug & Play

5.9 Verify the FASTRACK Supreme Network Registration

1. Make sure a valid SIM card has been previously inserted and locked in the FASTRACK Supreme SIM card holder.
2. Using a communication software such as a HyperTerminal, enter the following AT commands:

- a. **AT+CPIN=xxxx** to enter PIN code xxxx.
- b. **AT+WMBS?** To check the current band setting in the Plug & Play
- c. **AT+WMBS=<Band> [,<param>]** To switch band/mode when needed
- d. **AT+CREG?** . To ascertain the registration status.

The format of the returned response is as follows:

+CREG: <mode>,<stat> with:

- **<mode>** = unsolicited registration message configuration,
- **<stat>** = registration state.

3. Verify the state of registration according the returned value given in the table below.

Table 10: Values of network registration

Returned Value (*) <mode>,<stat>	Network registration
+CREG: 0,0	No (not registered)
+CREG: 0,1	Yes (registered, home network)
+CREG: 0,5	Yes (registered, roaming)

(*)For further information on the other returned values and their meaning, refer to "AT Commands Interface Guide" [6].

Fastrack Supreme User Guide

Using the FASTRACK Supreme Plug & Play

If the FASTRACK Supreme is not registered, perform the following procedure:

- Check the connection between the FASTRACK Supreme and the antenna.
- Verify the signal strength to determine the received signal strength (refer to Section 5.5).

Note: For information on AT command relating to the network registration in GPRS mode, and in particular: CGREG, CGCLASS, CGATT, refer to "AT Commands Interface Guide" [6].

5.10 Main AT Commands for the Plug & Play

The table below lists the main AT commands required for starting the Plug & Play.

For other AT commands available or further information on the AT commands, refer to "AT Commands Interface Guide" [6].

Table 11: Main usual AT commands for the Plug & Play

Description	AT commands	FASTRACK Supreme Plug & Play response	Comment
Check for selected band(s)	AT+WMBS?	+WMBS:<Band>,<ResetFlag> OK	Current selected band mode is return
Band(s) switch	AT+WMBS=<Band>	OK	Band switch is accepted, Plug & Play has to be reset for change to be effective
	AT+WMBS=<Band>,0	OK	Band switch is accepted, Plug & Play has to be reset for change to be effective
	AT+WMBS=<Band>,1	OK	Band switch is accepted and GSMS stack restarted
	AT+WMBS=<Band>	+CME ERROR: 3	Band not allowed
Enter PIN Code	AT+CPIN=xxxx (xxxx = PIN code)	OK	PIN Code accepted.
		+CME ERROR: 16	Incorrect PIN Code (with +CMEE = 1 mode) (1*)
		+CME ERROR: 3	PIN code already entered (with +CMEE = 1 mode) (1*)
Network registration checking	AT+CREG?	+CREG: 0,1	FASTRACK Supreme Plug & Play registered on the network.
		+CREG: 0,2	FASTRACK Supreme Plug & Play not registered on the network, registration attempt.

ANEXO 4

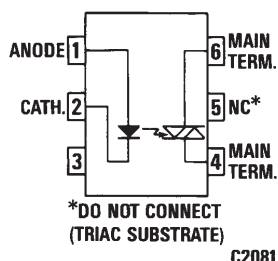
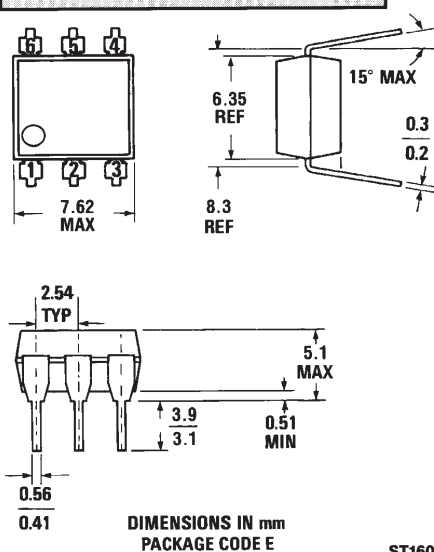




NON-ZERO-CROSSING TRIACS

MOC3020 MOC3021 MOC3022 MOC3023

PACKAGE DIMENSIONS



Equivalent Circuit

DESCRIPTION

The MOC3020, MOC3021, MOC3022 and MOC3023 are optically isolated triac driver devices. These devices contain a GaAs infrared emitting diode and a light activated silicon bilateral switch, which functions like a triac. This is designed for interfacing between electronic controls and power triacs to control resistive and inductive loads for 240 VAC operations.

FEATURES

- Excellent I_r stability—IR emitting diode has low degradation
- High isolation voltage—minimum 7500 VAC peak
- Underwriters Laboratory (UL) recognized—File #E90700

APPLICATIONS

- European applications for 240 VAC
- Triac driver
- Industrial controls
- Traffic lights
- Vending machines
- Motor control
- Solid state relay

ABSOLUTE MAXIMUM RATINGS

TOTAL PACKAGE

Storage temperature	−55°C to 150°C
Operating temperature	−40°C to 100°C
Lead temperature	
(soldering, 10 sec)	260°C

INPUT DIODE

Forward DC current	50 mA
Reverse voltage	3 V
Peak forward current	
(1 μ s pulse, 300 pps)	3.0 A
Power dissipation (25°C ambient)	100 mW
Derate linearly (above 25°C ambient)	1.33 mW/°C

OUTPUT DRIVER

Off-state output terminal voltage	400 Volts
On-state RMS current	$T_a=25^\circ\text{C}$ 100 mA
(Full cycle, 50 to 60 Hz)	$T_a=70^\circ\text{C}$ 50 mA
Peak nonrepetitive surge current	1.2 A
(PW=10 ms, DC=10%)	
Total power dissipation (25°C ambient)	300 mW
Derate above 25°C	4.0 mW/°C



NON-ZERO-CROSSING TRIACS

ELECTRO-OPTICAL CHARACTERISTICS (25°C Temperature Unless Otherwise Specified)

INDIVIDUAL COMPONENT CHARACTERISTICS

CHARACTERISTIC	SYMBOL	MIN.	TYP.	MAX.	UNITS	TEST CONDITIONS
INPUT DIODE						
Forward voltage	V_F		1.2	1.50	V	$I_F = 10 \text{ mA}$
Junction capacitance	C_J		50		pF	$V_F = 0 \text{ V}$, $f = 1 \text{ MHz}$
Reverse leakage current	I_R			100	μA	$V_R = 3.0 \text{ V}$
OUTPUT DETECTOR						
Peak blocking current, either direction	I_{DRM}	—	10	100	nA	$V_{DRM} = 400 \text{ V}$, Note 1
Peak on-state voltage, either direction	V_{TM}	—	2.5	3.0	Volts	$I_{TM} = 100 \text{ mA Peak}$

Note 1. Test voltage must be applied within dv/dt rating.

TRANSFER CHARACTERISTICS

DC CHARACTERISTICS	SYMBOL	MIN.	TYP.	MAX.	UNITS	TEST CONDITIONS
LED trigger current (current required to latch output)	MOC3020 I_{FT}	—	—	30	mA	Main terminal voltage = 3.0 V, $R_L = 150\Omega$
	MOC3021 I_{FT}	—	—	15	mA	
	MOC3022 I_{FT}	—	—	10	mA	
	MOC3023 I_{FT}	—	—	5	mA	
Holding current	I_H	—	100	—	μA	Either direction

TRANSFER CHARACTERISTICS

CHARACTERISTICS	SYMBOL	MIN.	TYP.	MAX.	UNITS	TEST CONDITIONS
dv/dt RATING						
Critical rate of rise of off-state voltage	dv/dt	—	12	—	V/ μs	Static dv/dt, $T_A = 85^\circ\text{C}$ (see Fig. 3)
Critical rate of rise of commutating voltage	dv/dt	—	0.2	—	V/ μs	Commutating dv/dt, $I_{LOAD} = 15 \text{ mA}$ (see Fig. 4)

ISOLATION CHARACTERISTICS

CHARACTERISTICS	SYMBOL	MIN.	TYP.	MAX.	UNITS	TEST CONDITIONS
Isolation voltage	V_{ISO}	5300			V_{ACRMS}	$I_{IO} \leq 1 \mu\text{A}$, 1 Minute
	V_{ISO}	7500			V_{ACPEAK}	$I_{IO} \leq 1 \mu\text{A}$, 1 Minute
Isolation resistance	R_{ISO}	10^{11}			ohms	$V_{IO} = 500 \text{ VDC}$
Isolation capacitance	C_{ISO}		0.5		pF	$f = 1 \text{ MHz}$

Note 1: Ratings apply to either polarity of pin 6 — referenced to pin 4. Voltages must be applied within dv/dt rating.

ANEXO 5



**Meet or Exceed TIA/EIA-232-F and ITU
Recommendation V.28**

Operate With Single 5-V Power Supply

Operate Up to 120 kbit/s

Two Drivers and Two Receivers

Low Supply Current . . . 8 mA Typical

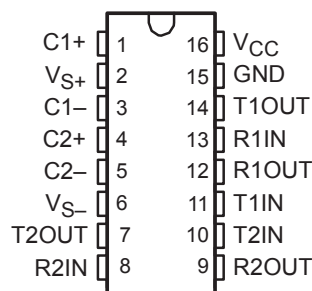
**Designed to be Interchangeable With
Maxim MAX232**

**ESD Protection Exceeds JESD 22
– 2000-V Human-Body Model (A114-A)**

Applications

**TIA/EIA-232-F
Battery-Powered Systems
Terminals
Modems
Computers**

**MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)**



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ± 30 -V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

T _A	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube	MAX232N	MAX232N
		Tube	MAX232D	MAX232
	SOIC (D)	Tape and reel	MAX232DR	
		Tube	MAX232DW	MAX232
	SOIC (DW)	Tape and reel	MAX232DWR	
	SOP (NS)	Tape and reel	MAX232NSR	MAX232
–40°C to 85°C	PDIP (N)	Tube	MAX232IN	MAX232IN
		Tube	MAX232ID	MAX232I
	SOIC (D)	Tape and reel	MAX232IDR	
		Tube	MAX232IDW	MAX232I
	SOIC (DW)	Tape and reel	MAX232IDWR	
		Tube	MAX232IDWR	MAX232I

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.



LinASIC is a trademark of Texas Instruments.

PRODUCTION DATA information is current as of publication date.

testing of all parameters.

Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include



**TEXAS
INSTRUMENTS**

POST OFFICE BOX 655303 DALLAS, TEXAS 75265

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)[†]

Input supply voltage range, V_{CC} (see Note 1)	–0.3 V to 6 V
Positive output supply voltage range, V_{S+}	$V_{CC} - 0.3$ V to 15 V
Negative output supply voltage range, V_{S-}	–0.3 V to –15 V
Input voltage range, V_I : Driver	–0.3 V to $V_{CC} + 0.3$ V
Receiver	±30 V
Output voltage range, V_O : T1OUT, T2OUT	$V_{S-} - 0.3$ V to $V_{S+} + 0.3$ V
R1OUT, R2OUT	–0.3 V to $V_{CC} + 0.3$ V
Short-circuit duration: T1OUT, T2OUT	Unlimited
Package thermal impedance, θ_{JA} (see Note 2): D package	73°C/W
DW package	57°C/W
N package	67°C/W
NS package	64°C/W
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds	260°C
Storage temperature range, T_{stg}	–65°C to 150°C

[†] Stresses beyond those listed under “absolute maximum ratings” may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under “recommended operating conditions” is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: All voltage values are with respect to network ground terminal.

2. The package thermal impedance is calculated in accordance with JESD 51-7.

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IH}	High-level input voltage (T1IN, T2IN)	2			V
V_{IL}	Low-level input voltage (T1IN, T2IN)			0.8	V
R1IN, R2IN	Receiver input voltage			±30	V
T_A	Operating free-air temperature	MAX232	0	70	°C
		MAX232I	–40	85	

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Note 3 and Figure 4)

PARAMETER	TEST CONDITIONS	MIN	TYP [‡]	MAX	UNIT
I_{CC} Supply current	$V_{CC} = 5.5$ V, All outputs open, $T_A = 25^\circ\text{C}$		8	10	mA

[‡] All typical values are at $V_{CC} = 5$ V and $T_A = 25^\circ\text{C}$.

NOTE 3: Test conditions are C1–C4 = 1 μF at $V_{CC} = 5$ V $\pm$ 0.5 V.


**TEXAS
INSTRUMENTS**

POST OFFICE BOX 655303 DALLAS, TEXAS 75265

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047I – FEBRUARY 1989 – REVISED OCTOBER 2002

DRIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 3)

PARAMETER		TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V _{OH}	High-level output voltage	T1OUT, T2OUT R _L = 3 kΩ to GND	5	7		V
V _{OL}	Low-level output voltage‡	T1OUT, T2OUT R _L = 3 kΩ to GND		–7	–5	V
r _o	Output resistance	T1OUT, T2OUT V _{S+} = V _{S–} = 0, V _O = ±2 V	300			Ω
I _{OS} §	Short-circuit output current	T1OUT, T2OUT V _{CC} = 5.5 V, V _O = 0		±10		mA
I _{IS}	Short-circuit input current	T1IN, T2IN V _I = 0			200	∞A

† All typical values are at V_{CC} = 5 V, T_A = 25°C.

‡ The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

§ Not more than one output should be shorted at a time.

NOTE 3: Test conditions are C1–C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 3)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
SR	Driver slew rate	R _L = 3 kΩ to 7 kΩ, See Figure 2			30	V/μs
SR(t)	Driver transition region slew rate	See Figure 3		3		V/μs
	Data rate	One TOUT switching		120		kbit/s

NOTE 3: Test conditions are C1–C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

RECEIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 3)

PARAMETER		TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V _{OH}	High-level output voltage	R1OUT, R2OUT I _{OH} = –1 mA	3.5			V
V _{OL}	Low-level output voltage‡	R1OUT, R2OUT I _{OL} = 3.2 mA			0.4	V
V _{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN V _{CC} = 5 V, T _A = 25°C		1.7	2.4	V
V _{IT–}	Receiver negative-going input threshold voltage	R1IN, R2IN V _{CC} = 5 V, T _A = 25°C	0.8	1.2		V
V _{hys}	Input hysteresis voltage	R1IN, R2IN V _{CC} = 5 V	0.2	0.5	1	V
r _i	Receiver input resistance	R1IN, R2IN V _{CC} = 5, T _A = 25°C	3	5	7	kΩ

† All typical values are at V_{CC} = 5 V, T_A = 25°C.

‡ The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

NOTE 3: Test conditions are C1–C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 3 and Figure 1)

PARAMETER		TYP	UNIT
t _{PLH(R)}	Receiver propagation delay time, low- to high-level output	500	ns
t _{PHL(R)}	Receiver propagation delay time, high- to low-level output	500	ns

NOTE 3: Test conditions are C1–C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.


POST OFFICE BOX 655303 DALLAS, TEXAS 75265

ANEXO 6





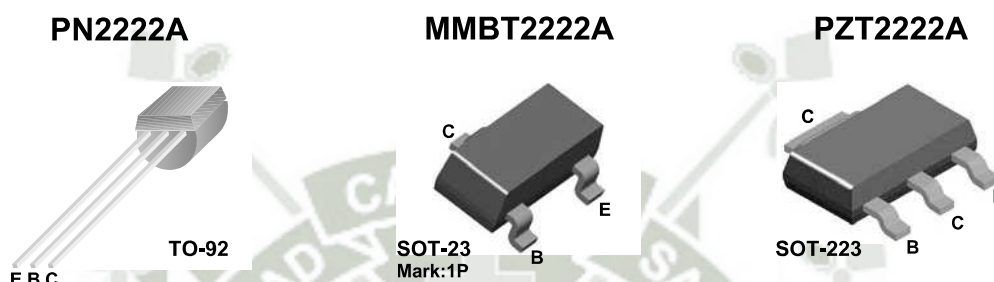
August 2010

PN2222A / MMBT2222A / PZT2222A

NPN General Purpose Amplifier

Features

- This device is for use as a medium power amplifier and switch requiring collector currents up to 500mA.
- Sourced from process 19.



Absolute Maximum Ratings * $T_a = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Value	Units
V_{CEO}	Collector-Emitter Voltage	40	V
V_{CBO}	Collector-Base Voltage	75	V
V_{EBO}	Emitter-Base Voltage	6.0	V
I_C	Collector Current	1.0	A
T_{STG}	Operating and Storage Junction Temperature Range	- 55 ~ 150	$^\circ\text{C}$

* This ratings are limiting values above which the serviceability of any semiconductor device may be impaired.

NOTES:

- 1) These rating are based on a maximum junction temperature of 150 degrees C.
- 2) These are steady limits. The factory should be consulted on applications involving pulsed or low duty cycle operations.

Thermal Characteristics $T_a = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Max.			Units
		PN2222A	*MMBT2222A	**PZT2222A	
P_D	Total Device Dissipation Derate above 25°C	625 5.0	350 2.8	1,000 8.0	mW mW/ $^\circ\text{C}$
$R_{\theta JC}$	Thermal Resistance, Junction to Case	83.3			$^\circ\text{C}/\text{W}$
$R_{\theta JA}$	Thermal Resistance, Junction to Ambient	200	357	125	$^\circ\text{C}/\text{W}$

* Device mounted on FR-4 PCB $1.6'' \times 1.6'' \times 0.06''$.

** Device mounted on FR-4 PCB $36\text{mm} \times 18\text{mm} \times 1.5\text{mm}$; mounting pad for the collector lead min. 6cm^2 .

PN2222A / MMBT2222A / PZT2222A — NPN General Purpose Amplifier

Electrical Characteristics $T_a = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Test Condition	Min.	Max.	Units
Off Characteristics					
$BV_{(BR)CEO}$	Collector-Emitter Breakdown Voltage *	$I_C = 10\text{mA}, I_B = 0$	40		V
$BV_{(BR)CBO}$	Collector-Base Breakdown Voltage	$I_C = 10\mu\text{A}, I_E = 0$	75		V
$BV_{(BR)EBO}$	Emitter-Base Breakdown Voltage	$I_E = 10\mu\text{A}, I_C = 0$	6.0		V
I_{CEX}	Collector Cutoff Current	$V_{CE} = 60\text{V}, V_{EB(off)} = 3.0\text{V}$		10	nA
I_{CBO}	Collector Cutoff Current	$V_{CB} = 60\text{V}, I_E = 0$ $V_{CB} = 60\text{V}, I_E = 0, T_a = 125^\circ\text{C}$		0.01 10	μA μA
I_{EBO}	Emitter Cutoff Current	$V_{EB} = 3.0\text{V}, I_C = 0$		10	nA
I_{BL}	Base Cutoff Current	$V_{CE} = 60\text{V}, V_{EB(off)} = 3.0\text{V}$		20	nA
On Characteristics					
h_{FE}	DC Current Gain	$I_C = 0.1\text{mA}, V_{CE} = 10\text{V}$ $I_C = 1.0\text{mA}, V_{CE} = 10\text{V}$ $I_C = 10\text{mA}, V_{CE} = 10\text{V}$ $I_C = 10\text{mA}, V_{CE} = 10\text{V}, T_a = -55^\circ\text{C}$ $I_C = 150\text{mA}, V_{CE} = 10\text{V}^*$ $I_C = 150\text{mA}, V_{CE} = 1\text{V}^*$ $I_C = 500\text{mA}, V_{CE} = 10\text{V}^*$	35 50 75 35 100 50 40	300	
$V_{CE(sat)}$	Collector-Emitter Saturation Voltage *	$I_C = 150\text{mA}, I_B = 15\text{mA}$ $I_C = 500\text{mA}, I_B = 50\text{mA}$		0.3 1.0	V V
$V_{BE(sat)}$	Base-Emitter Saturation Voltage *	$I_C = 150\text{mA}, I_B = 15\text{mA}$ $I_C = 500\text{mA}, I_B = 50\text{mA}$	0.6	1.2 2.0	V V
Small Signal Characteristics					
f_T	Current Gain Bandwidth Product	$I_C = 20\text{mA}, V_{CE} = 20\text{V}, f = 100\text{MHz}$	300		MHz
C_{obo}	Output Capacitance	$V_{CB} = 10\text{V}, I_E = 0, f = 1\text{MHz}$		8.0	pF
C_{ibo}	Input Capacitance	$V_{EB} = 0.5\text{V}, I_C = 0, f = 1\text{MHz}$		25	pF
$rb'C_C$	Collector Base Time Constant	$I_C = 20\text{mA}, V_{CB} = 20\text{V}, f = 31.8\text{MHz}$		150	pS
NF	Noise Figure	$I_C = 100\mu\text{A}, V_{CE} = 10\text{V},$ $R_S = 1.0\text{K}\Omega, f = 1.0\text{KHz}$		4.0	dB
$Re(h_{ie})$	Real Part of Common-Emitter High Frequency Input Impedance	$I_C = 20\text{mA}, V_{CE} = 20\text{V}, f = 300\text{MHz}$		60	Ω
Switching Characteristics					
t_d	Delay Time	$V_{CC} = 30\text{V}, V_{EB(off)} = 0.5\text{V},$ $I_C = 150\text{mA}, I_{B1} = 15\text{mA}$		10	ns
t_r	Rise Time			25	ns
t_s	Storage Time	$V_{CC} = 30\text{V}, I_C = 150\text{mA},$ $I_{B1} = I_{B2} = 15\text{mA}$		225	ns
t_f	Fall Time			60	ns

* Pulse Test: Pulse Width $\leq 300\mu\text{s}$, Duty Cycle $\leq 2.0\%$

ANEXO 7





BT136-600D

4Q Triac

Rev. 03 — 1 April 2011

Product data sheet

1. Product profile

1.1 General description

Planar passivated very sensitive gate four quadrant triac in a SOT78 plastic package intended for use in general purpose bidirectional switching and phase control applications, where high sensitivity is required in all four quadrants. This very sensitive gate "series D" triac is intended to be interfaced directly to microcontrollers, logic integrated circuits and other low power gate trigger circuits.

1.2 Features and benefits

- Direct triggering from low power drivers and logic ICs
- High blocking voltage capability
- Low holding current for low current loads and lowest EMI at commutation
- Planar passivated for voltage ruggedness and reliability
- Triggering in all four quadrants
- Very sensitive gate

1.3 Applications

- General purpose motor control
- General purpose switching

1.4 Quick reference data

Table 1. Quick reference data

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
V_{ORM}	repetitive peak off-state voltage		-	-	600	V
I_{TSM}	non-repetitive peak on-state current	full sine wave; $T_{J(int)} = 25\text{ }^{\circ}\text{C}$; $t_p = 20\text{ ms}$; see Figure 4 ; see Figure 5	-	-	25	A
$I_{T(RMS)}$	RMS on-state current	full sine wave; $T_{mb} \leq 107\text{ }^{\circ}\text{C}$; see Figure 1 ; see Figure 2 ; see Figure 3	-	-	4	A



NXP Semiconductors

BT136-600D

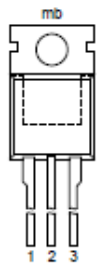
4Q Triac

Table 1. Quick reference data ...continued

Symbol	Parameter	Conditions	Min	Typ	Max	Unit
Static characteristics						
I_{GT}	gate trigger current	$V_D = 12\text{ V}$; $I_T = 0.1\text{ A}$; T2+ G+; $T_J = 25\text{ }^\circ\text{C}$; see Figure 7	-	2	5	mA
		$V_D = 12\text{ V}$; $I_T = 0.1\text{ A}$; T2+ G-; $T_J = 25\text{ }^\circ\text{C}$; see Figure 7	-	2.5	5	mA
		$V_D = 12\text{ V}$; $I_T = 0.1\text{ A}$; T2- G-; $T_J = 25\text{ }^\circ\text{C}$; see Figure 7	-	2.5	5	mA
		$V_D = 12\text{ V}$; $I_T = 0.1\text{ A}$; T2- G+; $T_J = 25\text{ }^\circ\text{C}$; see Figure 7	-	5	10	mA
I_H	holding current	$V_D = 12\text{ V}$; $T_J = 25\text{ }^\circ\text{C}$; see Figure 9	-	1.2	10	mA

2. Pinning information

Table 2. Pinning information

Pin	Symbol	Description	Simplified outline	Graphic symbol
1	T1	main terminal 1		
2	T2	main terminal 2		
3	G	gate		
mb	T2	mounting base; main terminal 2		

SOT78 (TO-220AB)

3. Ordering information

Table 3. Ordering information

Type number	Package		Version
	Name	Description	
BT136-600D	TO-220AB	plastic single-ended package; heatsink mounted; 1 mounting hole; 3-lead TO-220AB	SOT78
BT136-600D/DG	TO-220AB	plastic single-ended package; heatsink mounted; 1 mounting hole; 3-lead TO-220AB	SOT78

4. Limiting values

Table 4. Limiting values

In accordance with the Absolute Maximum Rating System (IEC 60134).

Symbol	Parameter	Conditions	Min	Max	Unit
V_{DRM}	repetitive peak off-state voltage		-	600	V
$I_{\text{T(RMS)}}$	RMS on-state current	full sine wave; $T_{\text{mb}} \leq 107\text{ }^{\circ}\text{C}$; see Figure 1; see Figure 2; see Figure 3	-	4	A
I_{TSM}	non-repetitive peak on-state current	full sine wave; $T_{\text{J(Init)}} = 25\text{ }^{\circ}\text{C}$; $t_{\text{p}} = 20\text{ ms}$; see Figure 4; see Figure 5	-	25	A
		full sine wave; $T_{\text{J(Init)}} = 25\text{ }^{\circ}\text{C}$; $t_{\text{p}} = 16.7\text{ ms}$	-	27	A
I^2t	I^2t for fusing	$t_{\text{p}} = 10\text{ ms}$; sine-wave pulse	-	3.1	A^2s
di_{T}/dt	rate of rise of on-state current	$I_{\text{T}} = 6\text{ A}$; $I_{\text{G}} = 0.2\text{ A}$; $dI_{\text{G}}/dt = 0.2\text{ A}/\mu\text{s}$; $T_2 + \text{G}+$	-	50	$\text{A}/\mu\text{s}$
		$I_{\text{T}} = 6\text{ A}$; $I_{\text{G}} = 0.2\text{ A}$; $dI_{\text{G}}/dt = 0.2\text{ A}/\mu\text{s}$; $T_2 + \text{G}-$	-	50	$\text{A}/\mu\text{s}$
		$I_{\text{T}} = 6\text{ A}$; $I_{\text{G}} = 0.2\text{ A}$; $dI_{\text{G}}/dt = 0.2\text{ A}/\mu\text{s}$; $T_2 - \text{G}-$	-	50	$\text{A}/\mu\text{s}$
		$I_{\text{T}} = 6\text{ A}$; $I_{\text{G}} = 0.2\text{ A}$; $dI_{\text{G}}/dt = 0.2\text{ A}/\mu\text{s}$; $T_2 - \text{G}+$	-	10	$\text{A}/\mu\text{s}$
I_{GM}	peak gate current		-	2	A
V_{GM}	peak gate voltage		-	5	V
P_{GM}	peak gate power		-	5	W
$P_{\text{G(AV)}}$	average gate power	over any 20 ms period	-	0.5	W
T_{stg}	storage temperature		-40	150	$^{\circ}\text{C}$
T_{J}	junction temperature		-	125	$^{\circ}\text{C}$

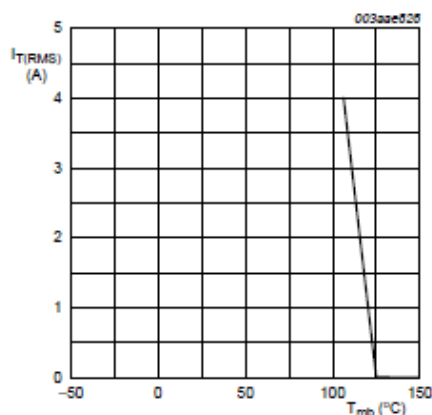


Fig 1. RMS on-state current as a function of mounting base temperature; maximum values

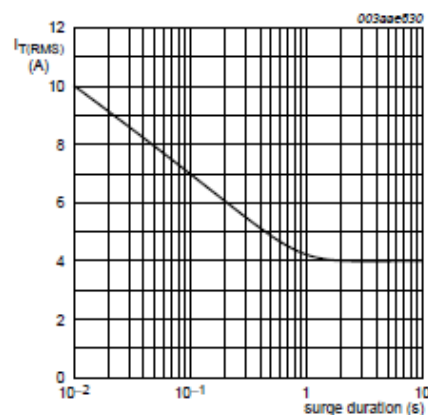


Fig 2. RMS on-state current as a function of surge duration; maximum values

ANEXO 8

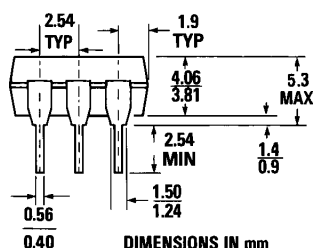
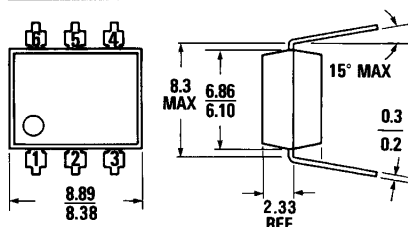




PHOTOTRANSISTOR OPTOCOUPLEDERS

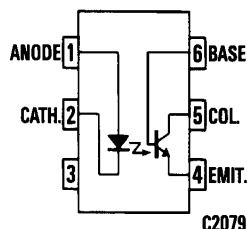
4N25 4N27
4N26 4N28

PACKAGE DIMENSIONS



DIMENSIONS IN mm
PACKAGE CODE K

ST1603A



Equivalent Circuit

DESCRIPTION

The 4N25, 4N26, 4N27, and 4N28 series of optocouplers have an NPN silicon planar phototransistor optically coupled to a gallium arsenide diode.

FEATURES & APPLICATIONS

- AC line/digital logic isolator
- Digital logic/digital logic isolator
- Telephone/telegraph line receiver
- Twisted pair line receiver
- High frequency power supply feedback control
- Relay contact monitor
- Power supply monitor
- Small package size and low cost
- Excellent frequency response
- UL recognized—File E90700

ABSOLUTE MAXIMUM RATINGS

TOTAL PACKAGE

- *Storage temperature -55°C to 150°C
- *Operating temperature at junction -55°C to 100°C
- *Lead temperature (soldering, 10 sec) 260°C
- *Total package power dissipation at 25°C ambient (LED plus detector) 250 mW
- *Derate linearly from 25°C 3.3 mW/°C

INPUT DIODE

- *Forward DC current continuous 80 mA
- *Reverse voltage 3.0 V
- *Peak forward current
(300 μ s, 2% duty cycle) 3.0 A
- *Power dissipation at 25°C ambient 150 mW
- *Derate linearly from 25°C 2.0 mW/°C

OUTPUT TRANSISTOR

- *Collector emitter voltage (BV_{CE0}) 30 V
- *Collector base voltage (BV_{CBO}) 70 V
- *Emitter collector voltage (BV_{ECO}) 7 V
- *Power dissipation at 25°C ambient 150 mW
- *Derate linearly from 25°C 2.0 mW/°C

*Indicates JEDEC Registered Data.



PHOTOTRANSISTOR OPTOCOUPERS

ELECTRO-OPTICAL CHARACTERISTICS

(25°C Free Air Temperature Unless Otherwise Specified)

INDIVIDUAL COMPONENT CHARACTERISTICS

CHARACTERISTICS	SYMBOL	MIN.	TYP.	GUAR. MAX.	UNITS	TEST CONDITIONS
INPUT DIODE						
*Forward voltage	V_f		1.20	1.50	V	$I_f = 10 \text{ mA}$
Capacitance	C		150		pF	$V_f = 0 \text{ V}$, $f = 1 \text{ MHz}$
*Reverse leakage current			.05	100	μA	$V_R = 3.0 \text{ V}$, $R_L = 1.0 \text{ M}\Omega$
DETECTOR						
DC forward current gain	h_{FE}		250			$V_{CE} = 5 \text{ V}$, $I_C = 500 \mu\text{A}$
*Collector to emitter breakdown voltage	BV_{CEO}	30	65		V	$I_C = 1.0 \text{ mA}$, $I_B = 0$
*Collector to base breakdown voltage	BV_{CBO}	70	165		V	$I_C = 100 \mu\text{A}$, $I_E = 0$
*Emitter to collector breakdown voltage	BV_{ECO}	7	14		V	$I_E = 100 \mu\text{A}$, $I_B = 0$
*Collector to emitter leakage current (4N25, 4N26, 4N27)	I_{CEO}		3.5	50	nA	$V_{CE} = 10 \text{ V}$ Base Open
*Collector to emitter leakage current (4N28)				100	nA	
*Collector to base leakage current	I_{CBO}		0.1	20	nA	$V_{CB} = 10 \text{ V}$ Emitter Open

TRANSFER CHARACTERISTICS

DC CHARACTERISTICS	SYMBOL	MIN.	TYP.	GUAR. MAX.	UNITS	TEST CONDITIONS
*Collector output current (a) (4N25, 4N26) (4N27, 4N28)	I_C	2.0 1.0	5.0 3.0	— —	mA	$V_{CE} = 10 \text{ V}$, $I_f = 10 \text{ mA}$, $I_B = 0$
*Collector-emitter saturation	$V_{CE(SAT)}$		0.2	0.5	V	$I_C = 2.0 \text{ mA}$, $I_f = 50 \text{ mA}$

TRANSFER CHARACTERISTICS

AC CHARACTERISTICS	SYMBOL	TYP.	UNITS	TEST CONDITIONS
Non-saturated Collector Delay time	t_d	0.5	μS	$R_L = 100 \Omega$, $I_C = 2 \text{ mA}$, $V_{CC} = 10 \text{ V}$ (Fig. 10 and 11)
Rise time	t_r	2.5	μS	
Fall time	t_f	2.6	μS	
Non-saturated Collector Delay time	t_d	2.0	μS	$R_L = 1\text{k}\Omega$, $I_C = 2 \text{ mA}$, $V_{CC} = 10 \text{ V}$ (Fig. 10 and 11)
Rise time	t_r	15	μS	
Fall time	t_f	15	μS	

*Indicates JEDEC Registered Data.

(a) Pulse Test: Pulse Width = 300 μs , Duty Cycle $\leq 2.0\%$

(b) For this test LED pins 1 and 2 are common and Phototransistor pins 4, 5 and 6 are common.

(c) If adjusted to yield $I_C = 2 \text{ mA}$ and $I_f = 0.7 \text{ mA RMS}$; Bandwidth referenced to 10 kHz.



PHOTOTRANSISTOR OPTOCOUPLERS

ELECTRO-OPTICAL CHARACTERISTICS

(25°C Free Air Temperature Unless Otherwise Specified) (Cont'd)

TRANSFER CHARACTERISTICS (Cont'd)

AC CHARACTERISTICS	SYMBOL	MIN.	TYP.	GUAR. MAX.	UNITS	TEST CONDITIONS
Saturated t_{on} (from 5 V to 0.8 V)	t_{on} (SAT)		5		μs	$R_L=2k\Omega$, $I_F=15$ mA, $V_{CC}=5$ V
t_{off} (from SAT to 2.0 V)	t_{off} (SAT)		25		μs	$R_B=Open$ (Fig. 10)
Saturated t_{on} (from 5 V to 0.8 V)	t_{on} (SAT)		5		μs	$R_L=2k\Omega$, $I_F=20$ mA, $V_{CC}=5$ V
t_{off} (from SAT to 2.0 V)	t_{off} (SAT)		18		μs	$R_B=100k\Omega$ (Fig. 10)
Non-saturated Base—Collector photo diode Rise time	t_r		175		ns	$R_L=1k\Omega$, $V_{CB}=10$ V
Fall time	t_f		175		ns	
Isolation voltage (b) (4N25, 4N26, 4N27, 4N28) *(4N26, 4N27) *(4N28)	V_{ISO}	5300 1500 500	— — —	— — —	V V V	$I_{FO} \leq 1 \mu A$ RMS, $t=1$ minute Peak Peak
Isolation resistance (b)			10^{11}		Ω	$V=500$ VDC
Isolation capacitance (b)			1.3		pF	$V=0$, $f=1.0$ MHz
Bandwidth (c) (also see note 2)	B_w		300		kHz	$I_C=2.0$ mA, $R_L=100 \Omega$ (Fig. 12)

*Indicates JEDEC Registered Data.

(a) Pulse Test: Pulse Width=300 μs , Duty Cycle $\leq 2.0\%$

(b) For this test LED pins 1 and 2 are common and Phototransistor pins 4, 5 and 6 are common.

(c) If adjusted to yield $I_C=2$ mA and $i_C=0.7$ mA RMS; Bandwidth referenced to 10 kHz.

TYPICAL ELECTRO-OPTICAL CHARACTERISTIC CURVES

(25°C Free Air Temperature Unless Otherwise Specified)

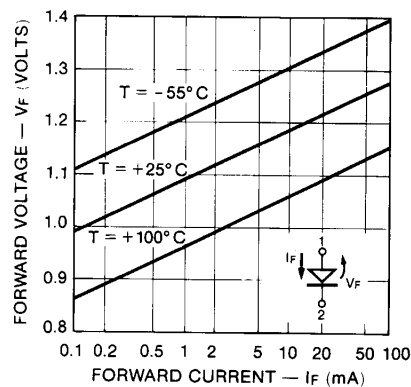


Fig. 1. Forward Voltage vs. Current

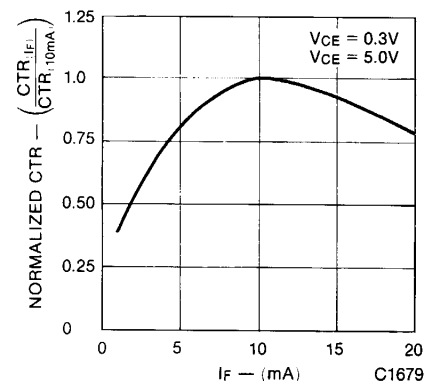


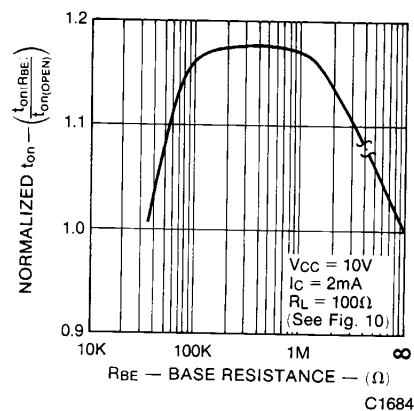
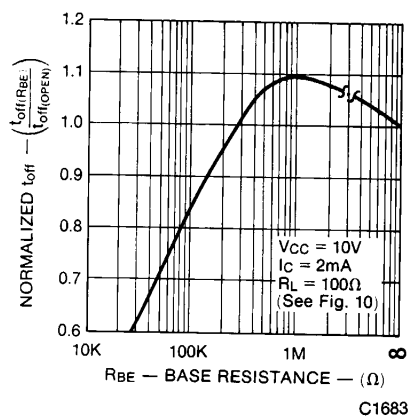
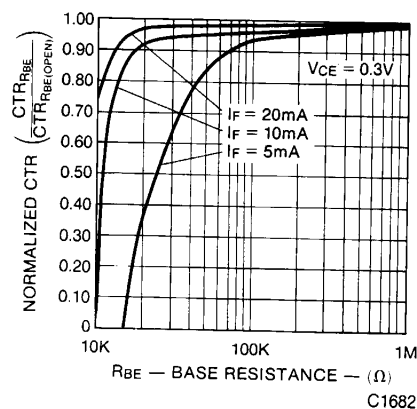
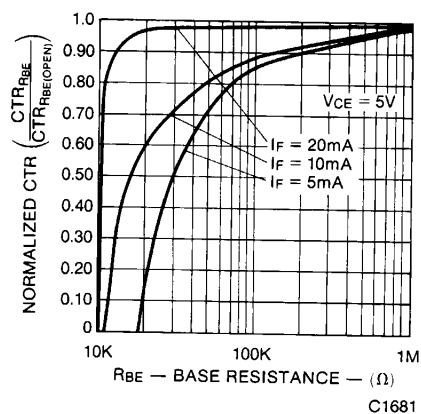
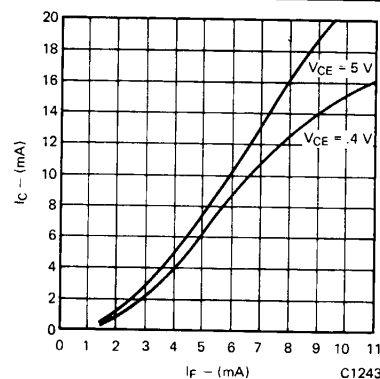
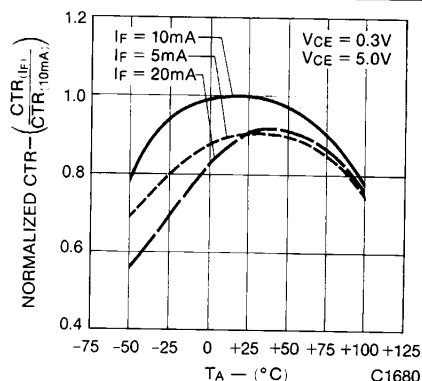
Fig. 2. Normalized CTR vs. Forward Current



PHOTOTRANSISTOR OPTOCOUPERS

TYPICAL ELECTRO-OPTICAL CHARACTERISTIC CURVES

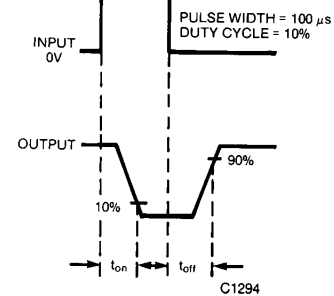
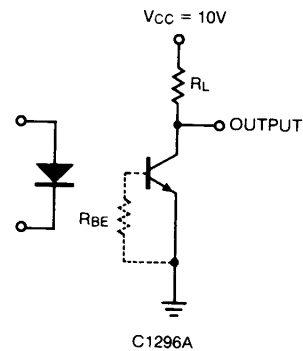
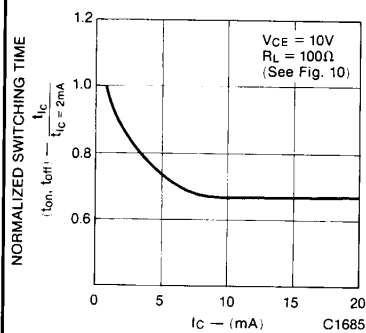
(25°C Free Air Temperature Unless Otherwise Specified) (Cont'd)



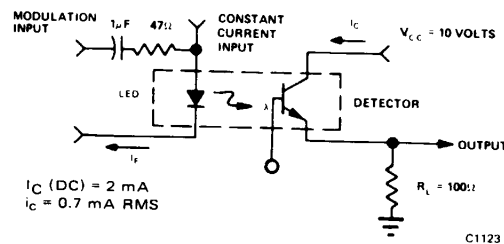


PHOTOTRANSISTOR OPTOCOUPERS

TYPICAL ELECTRO-OPTICAL CHARACTERISTIC CURVES (25°C Free Air Temperature Unless Otherwise Specified) (Cont'd)



OPERATING SCHEMATICS



NOTES

1. The current transfer ratio (I_C/I_F) is the ratio of the detector collector current to the LED input current with V_{CE} at 10 volts.
2. The frequency at which i_c is 3dB down from the 10 kHz value.
3. Rise time (t_r) is the time required for the collector current to increase from 10% of its final value to 90%.
Fall time (t_f) is the time required for the collector current to decrease from 90% of its initial value to 10%.

ANEXO 9



1N4001-1N4007



1N4001 - 1N4007

Features

- Low forward voltage drop.
- High surge current capability.



DO-41

COLOR BAND DENOTES CATHODE

General Purpose Rectifiers (Glass Passivated)

Absolute Maximum Ratings*

$T_A = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Value							Units
		4001	4002	4003	4004	4005	4006	4007	
V_{RRM}	Peak Repetitive Reverse Voltage	50	100	200	400	600	800	1000	V
$I_{F(AV)}$	Average Rectified Forward Current, .375 " lead length @ $T_A = 75^{\circ}\text{C}$	1.0							A
I_{FSM}	Non-repetitive Peak Forward Surge Current 8.3 ms Single Half-Sine-Wave	30							A
T_{stg}	Storage Temperature Range	-55 to +175							$^{\circ}\text{C}$
T_J	Operating Junction Temperature	-55 to +175							$^{\circ}\text{C}$

*These ratings are limiting values above which the serviceability of any semiconductor device may be impaired.

Thermal Characteristics

Symbol	Parameter	Value	Units
P_D	Power Dissipation	3.0	W
$R_{\theta JA}$	Thermal Resistance, Junction to Ambient	50	$^\circ\text{C}/\text{W}$

Electrical Characteristics

$T_A = 25^\circ\text{C}$ unless otherwise noted

Symbol	Parameter	Device							Units
		4001	4002	4003	4004	4005	4006	4007	
V_f	Forward Voltage @ 1.0 A	1.1							V
I_{rr}	Maximum Full Load Reverse Current, Full Cycle $T_A = 75^{\circ}\text{C}$	30							μA
I_R	Reverse Current @ rated V_R $T_A = 25^{\circ}\text{C}$	5.0							μA
	$T_A = 100^{\circ}\text{C}$	500							μA
C_T	Total Capacitance $V_R = 4.0\text{ V}$, $f = 1.0\text{ MHz}$	15							pF

General Purpose Rectifiers (Glass Passivated) (continued)

Typical Characteristics

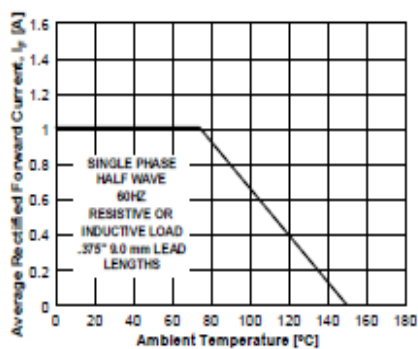


Figure 1. Forward Current Derating Curve

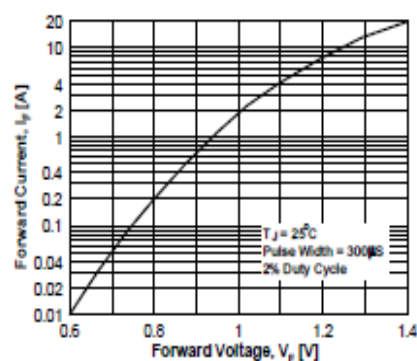


Figure 2. Forward Voltage Characteristics

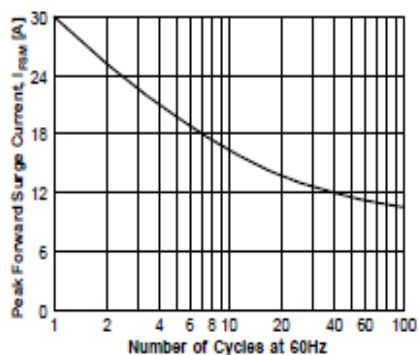


Figure 3. Non-Repetitive Surge Current

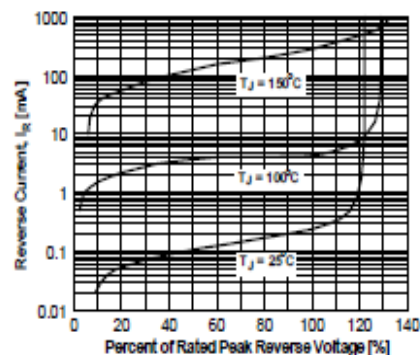
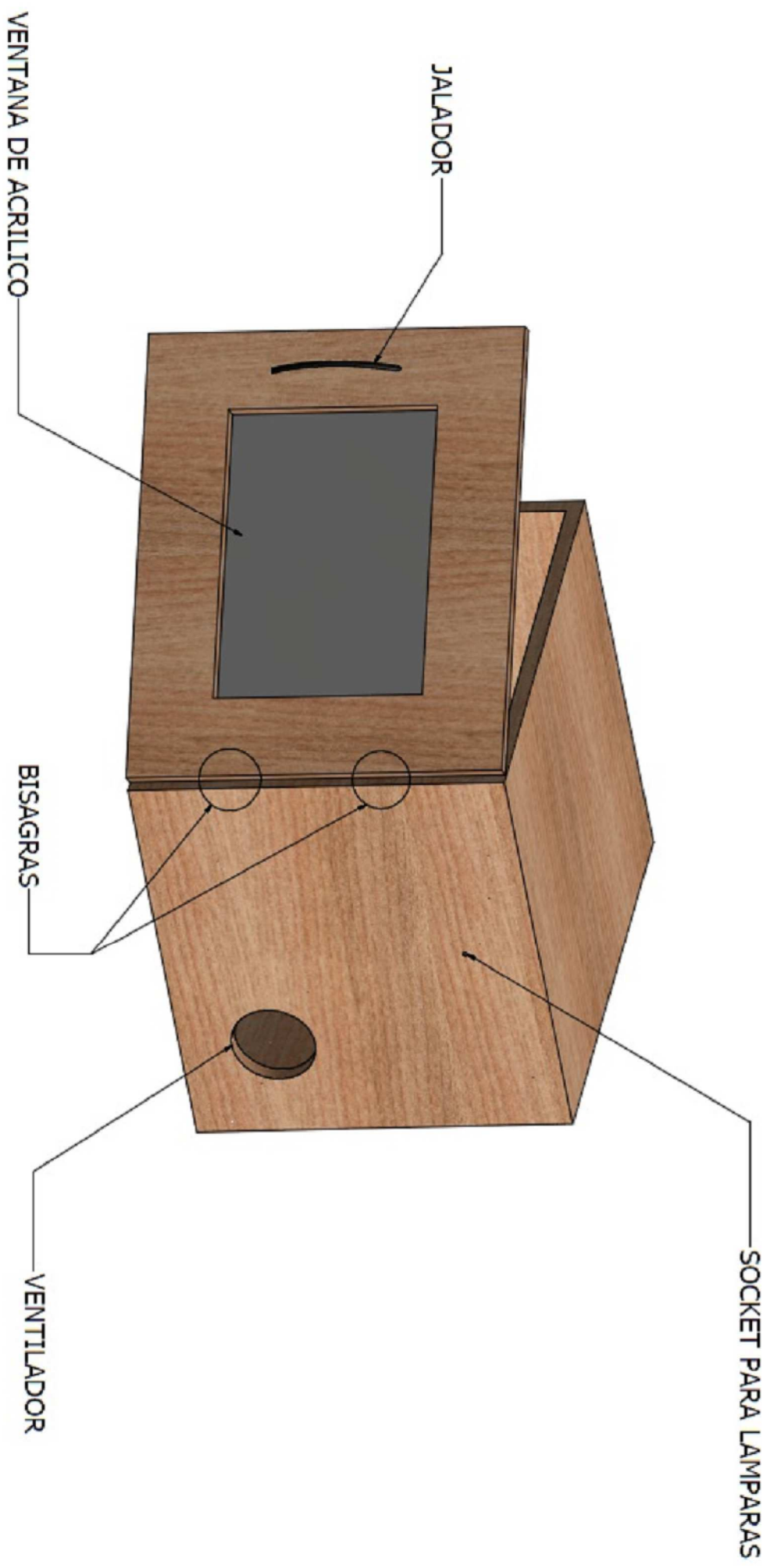


Figure 4. Reverse Current vs Reverse Voltage



VISTA ISOMETRICA

MATERIAL DE PROTOTIPO : Madera
ESPESOR : 20 mm
MATERIAL DE VENTANA : Acrilico
ESPESOR : 4 mm
TODAS LAS MEDIDAS EN MILIMETROS (mm)



UNIVERSIDAD CATOLICA DE SANTA MARIA

Tesis : Diseño e implementación de un controlador PID Plano N°: 01

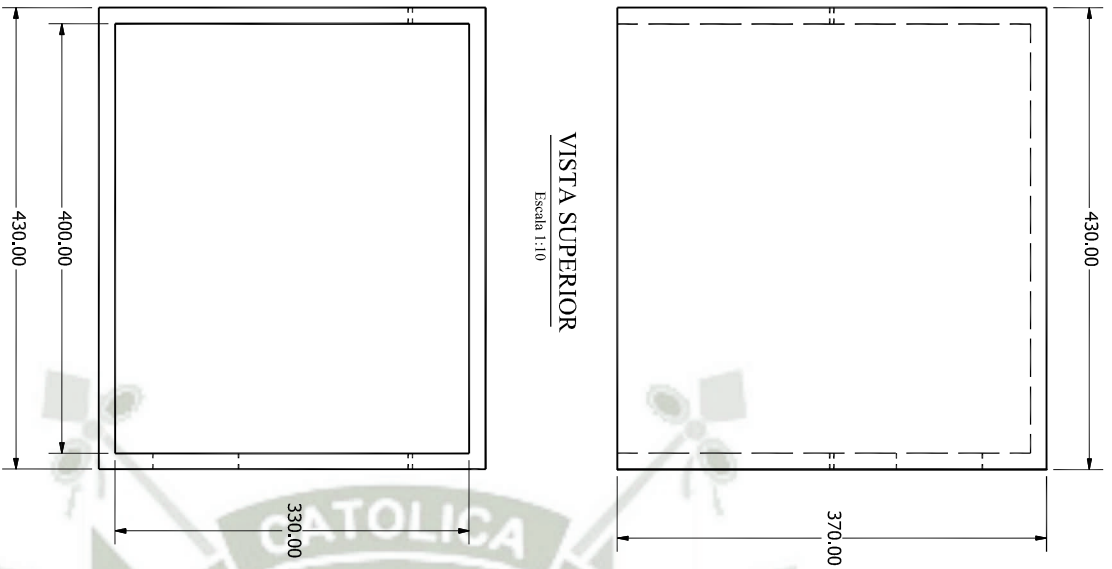
PPIMEM

CAMARA TERMICA

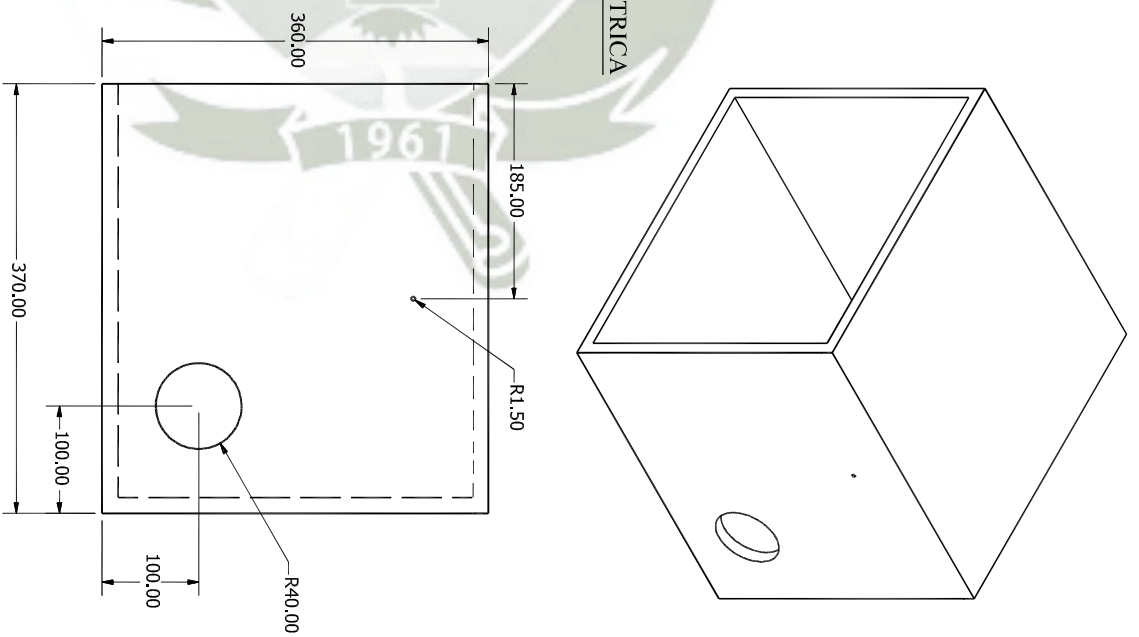
Fecha: 18/09/13

Diseñado por : Miguel Angel Ramirez Lazo

MATERIAL : Madera
ESPESOR : 20 mm
TODAS LAS MEDIDAS EN MILIMETROS (mm)



VISTA FRONTAL
Escala 1:10



UNIVERSIDAD CATOLICA DE SANTA MARIA

Tesis : Diseño e implementación de un controlador PID Plano N° : 02

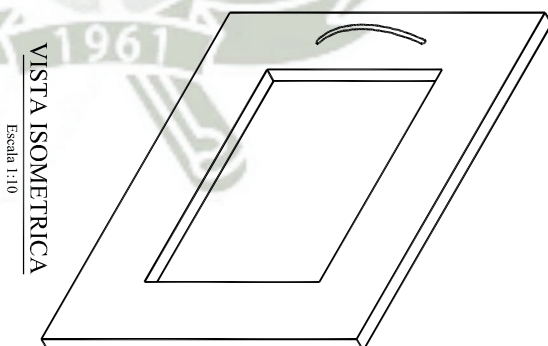
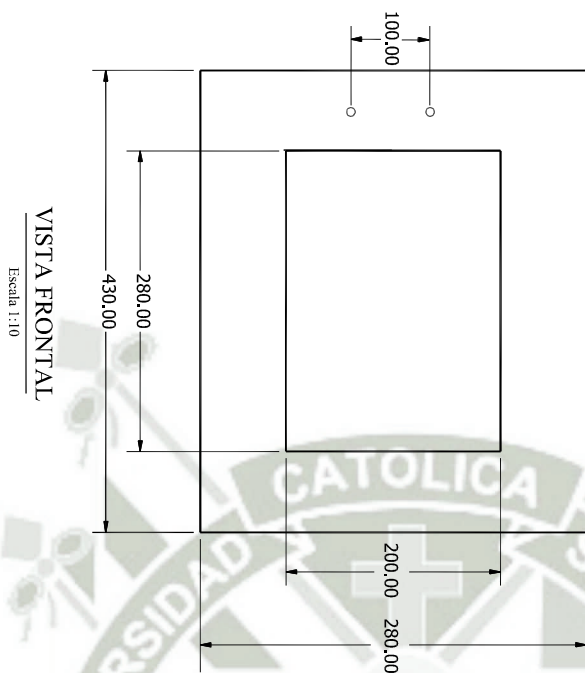
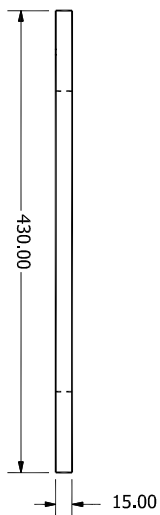
PPIMEM

CUERPO DE CAMARA TERMICA

Diseñado por : Miguel Angel Ramirez Lazo

Fecha: 18/09/13

MATERIAL : Madera
ESPESOR : 20 mm
TODAS LAS MEDIDAS EN MILIMETROS (mm)



UNIVERSIDAD CATOLICA DE SANTA MARIA

Tesis : Diseño e implementación de un controlador PID

Plano N° : 03

PPIMEM

TAPA DE CAMARA TERMICA

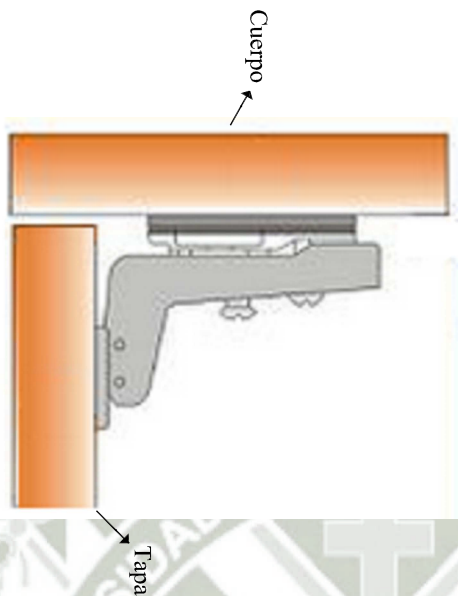
Fecha: 18/09/13

Diseñado por : Miguel Angel Ramirez Lazo



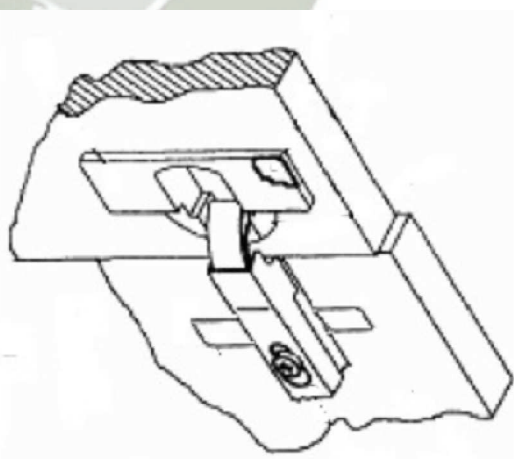
VISTA ISOMETRICA

Escala 1:2



VISTA SUPERIOR DE INSTALACION

Escala 1:10



DETALLE DE INSTALACION

Escala 1:10



UNIVERSIDAD CATOLICA DE SANTA MARIA

Tesis : Diseño e implementación de un controlador PID

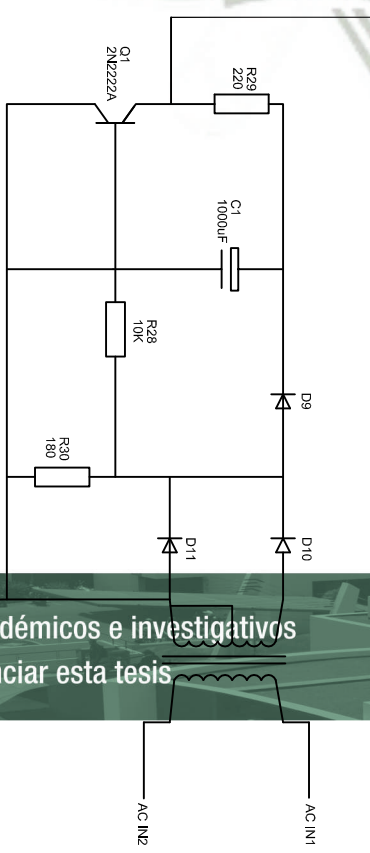
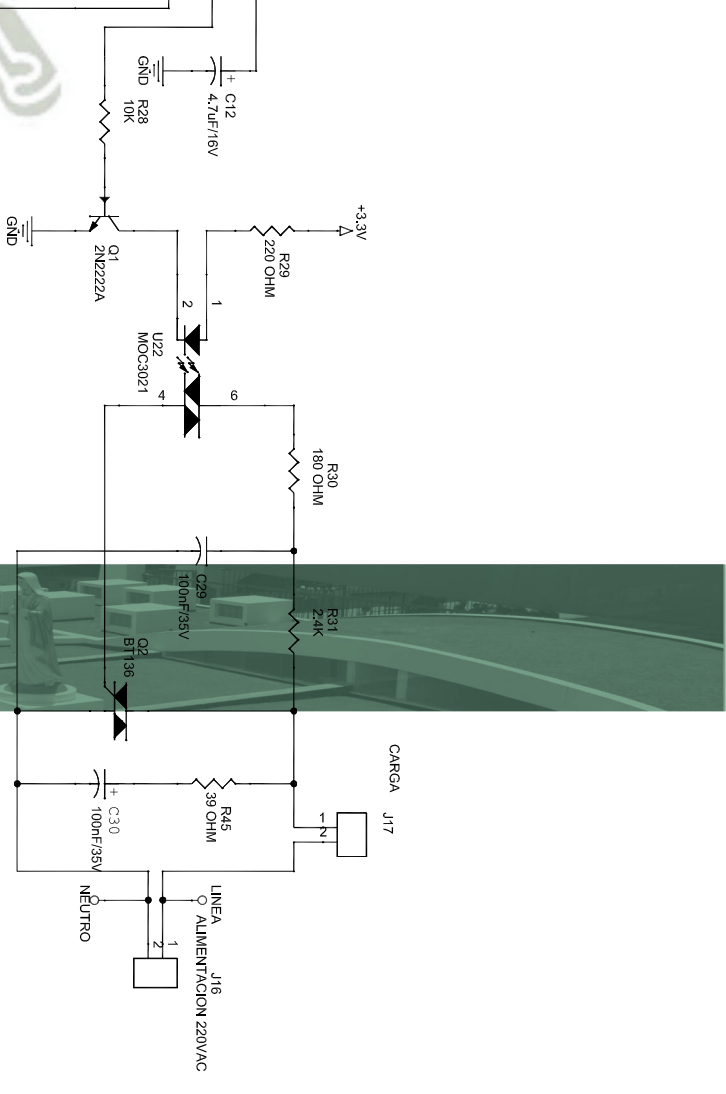
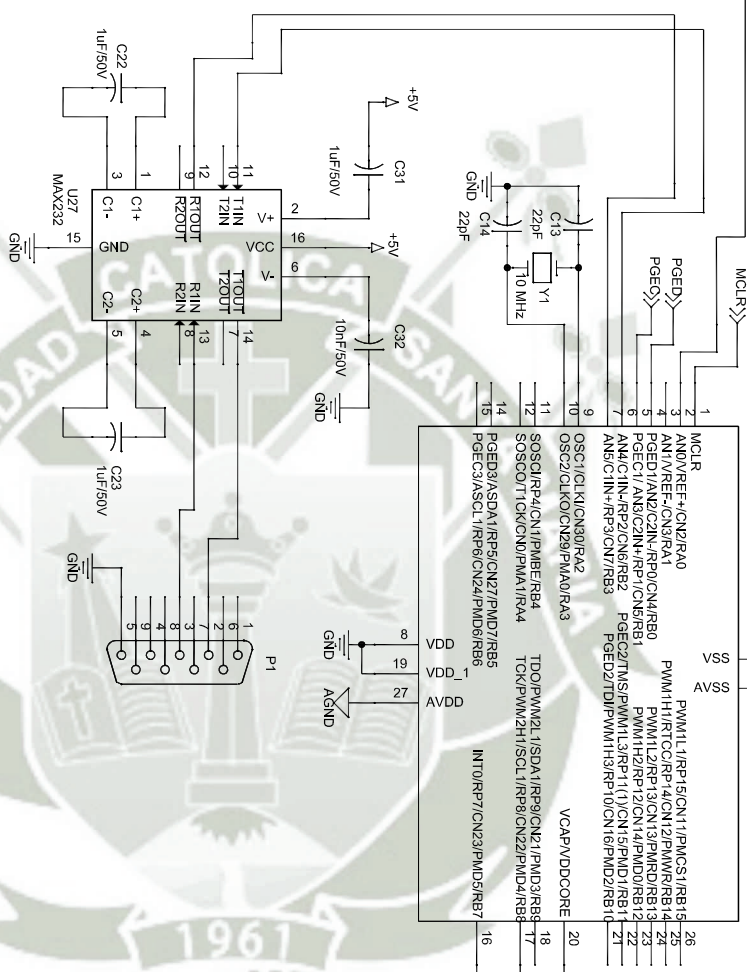
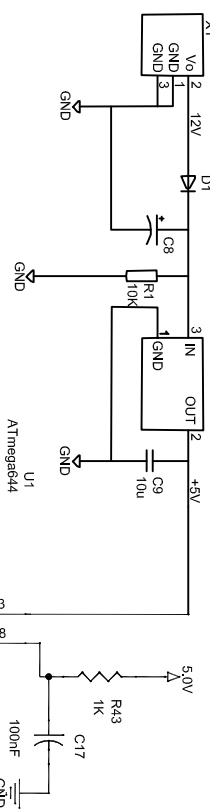
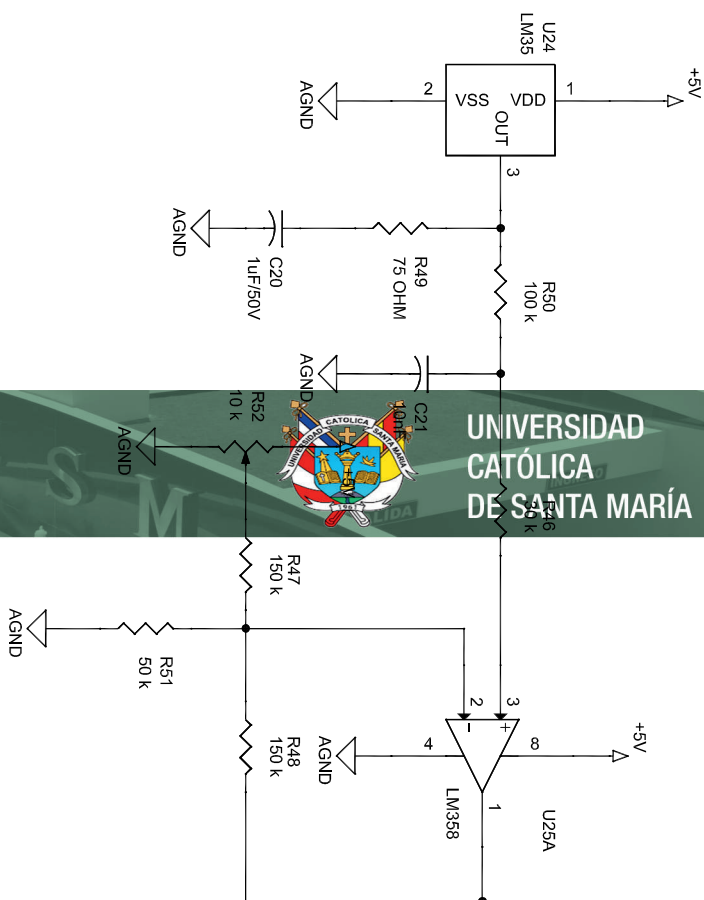
Plano N°. 04

PPIMEM

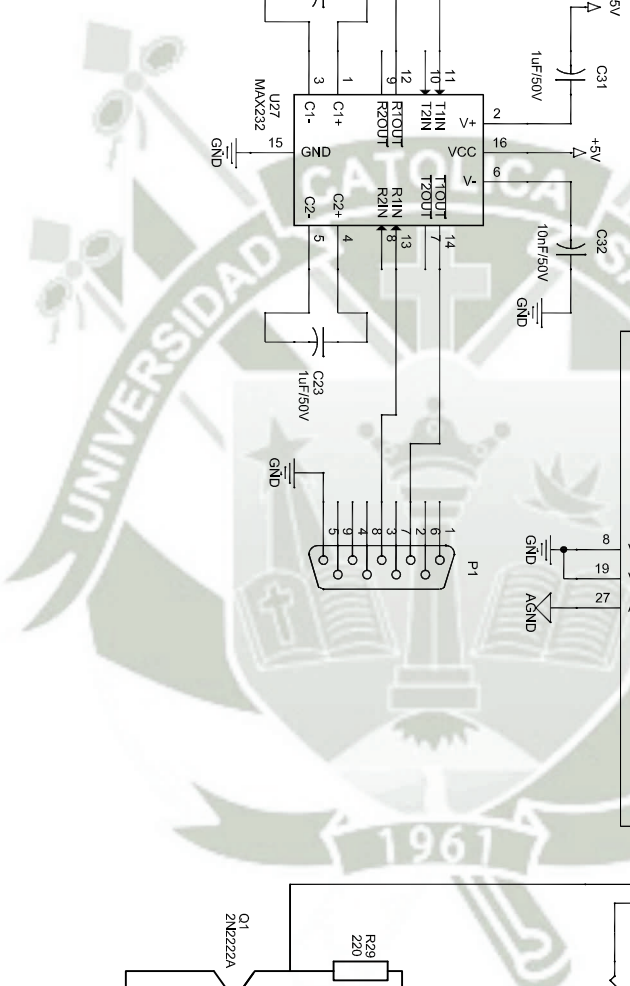
BISAGRA, DETALLE DE INSTALACION

Fecha: 10/05/13

Diseñado por : Miguel Angel Ramirez Lazo



**REPOSITORIO DE
TESIS UCSM**



Publicación autorizada con fines académicos e investigativos
En su investigación no olvide referenciar esta tesis

UNIVERSIDAD CATOLICA DE SANTAMARIA

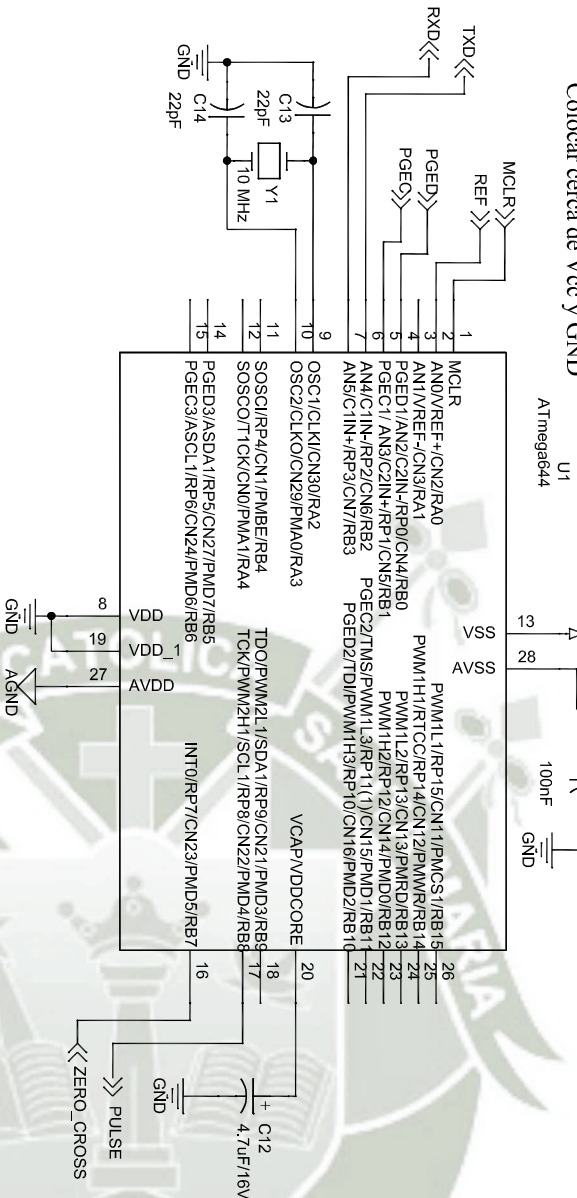
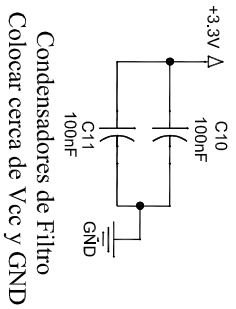
Tesis : Diseño e implementación de un controlador PID Plano N°: 05

CONFIGURACION PLACA DE CONTROL Fecha: 18/09/13

PRIMUM

Diseñado por : Miguel Angel Ramirez Lazo

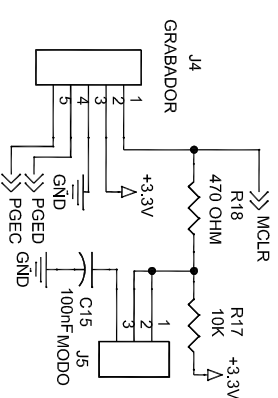




CIRCUITO MICROCONTROLADOR

Este circuito se encarga de colocar todos los componentes requeridos por el microcontrolador Atmega644 para poder trabajar adecuadamente. Los condensadores R43, C17, C10 y C11 se colocan para poder reducir el ruido (por recomendación del datasheet del microcontrolador); además, considerar colocar estos componentes lo más cercano posible a la alimentación de voltaje correspondiente. Asimismo, se considera una tierra analógica (AGND) y a una tierra digital (GND) diferenciadas y unidas únicamente en un solo punto para contribuir al mismo fin mencionado; tener en cuenta que dicha unión nunca se debe de realizar utilizando cables externos.

LISTA DE MATERIALES	
C10	100nF
C11	100nF
C17	100nF
C12	4.7uF/35v
C13	22pF
C14	1700
C15	100nF
Y1	10 MHz
R43	1k
R18	470 OHM
R17	10K



CIRCUITO PARA EL GRABADOR

Este circuito se usa como interfaz eléctrica para el grabador PICKIT2 utilizado en el presente proyecto (dicho esquemático se obtuvo a partir del datasheet del microcontrolador). Se debe resaltar que el PICKIT2 es el encargado de grabar el programa de control .HEX.



UNIVERSIDAD CATOLICA DE SANTA MARIA

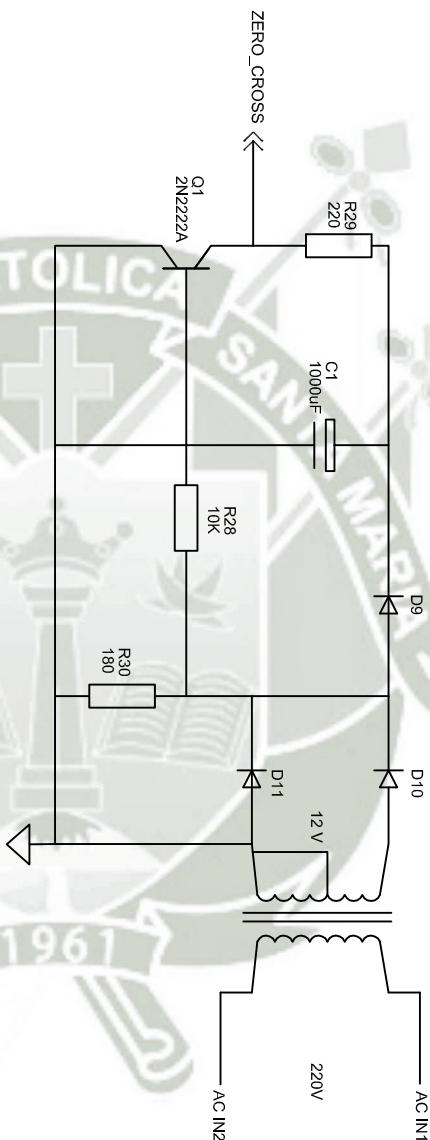
Tesis : Diseño e implementación de un controlador PID Plano N°: 06

CONFIGURACION ATmega644 Fecha: 18/09/13

PPIMEM

Diseñado por : Miguel Angel Ramirez Lazo

LISTA DE MATERIALES				
R28	10K	B9	1N4007	
R29	220 OHM	D10	1N4007	
R30	180 OHM	D11	1N4007	
R31	2.4K	D12	1N4007	
R45	39 OHM	R32	4.7k	
C29	100nF/350V	R33	18k	
C30	100nF/350V	U23	AN25	
Q1	2N2222A	U17	BORNERA DE DOS PINES PARA	
U22	MOC3021		CONECTAR EL FOCO DE 100W	
Q2	BT136	U5	BORNERA DE DOS PINES PARA	
R17	10K		CONECTAR LA ALIMENTACION DE 220 VAC	



CIRCUITO DE DETECCION DE CRUCE POR CERO

Se encarga de indicar al microcontrolador cuando es que la señal de alimentación AC de 220V cruza por cero. Esto con el fin de que el microcontrolador espere el tiempo suficiente para generar el flanco de disparo para la etapa de potencia. Este circuito podría tambien haberse implementado con un opam pero la desventaja es que no ofrece un aislamiento electrico lo cual pone en riesgo al microcontrolador

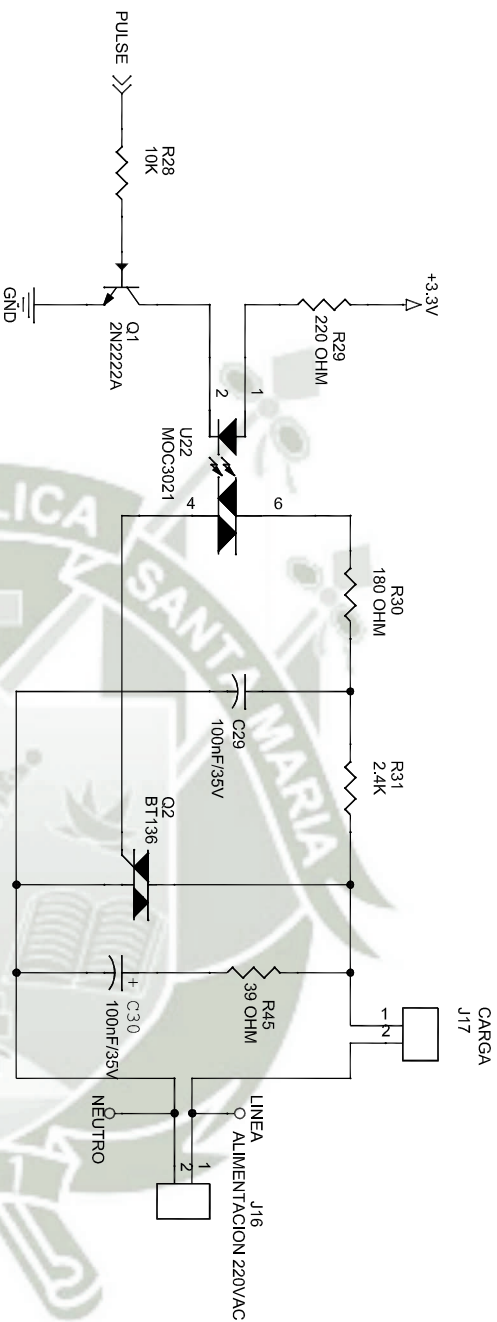
UNIVERSIDAD CATOLICA DE SANTA MARIA

Tesis : Diseño e implementación de un controlador PID Plano N°:07

PPIMEM

CIRCUITO DETECCION CRUCE POR CERO
Diseñado por : Miguel Angel Ramirez Lazo
Fecha: 18/09/13





CIRCUITO DE POTENCIA

Se encarga de realizar la conversión de la señal de tipo pulsante proveniente del microcontrolador para realizar el control de fase de la señal de voltaje alterno de 220VAC (Vendría a ser básicamente un convertidor DC-AC). Se debe resaltar que la señal proveniente del microcontrolador no se conecta directamente al optotriac ya que la corriente que requiere esta es demasiado alta (alrededor de 20 mA). a comparación de la corriente del Atmega (alrededor de 4 mA). Los valores de las resistencias y condensadores para el triac y el optotriac se obtuvieron a partir del datasheet del MOC3021; de forma analoga, se calculo las resistencias del transistor a partir del datasheet del 2N2222A

LISTA DE MATERIALES

R28	10K	B9	1N4007
R29	220 OHM	D10	1N4007
R30	180 OHM	D11	1N4007
R31	2.4K	D12	1N4007
R45	39 OHM	R32	4.7k
C29	100nF/350V	R33	18k
C30	100nF/350V	U23	AN25
Q1	2N2222A	J17	BORNERA DE DOS PINES PARA CONECTAR EL FOCO DE 100W
U22	MOC3021	J5	BORNERA DE DOS PINES PARA CONECTAR LA ALIMENTACION DE 220 VAC
Q2	BT136		
R17	10K		

UNIVERSIDAD CATOLICA DE SANTA MARIA

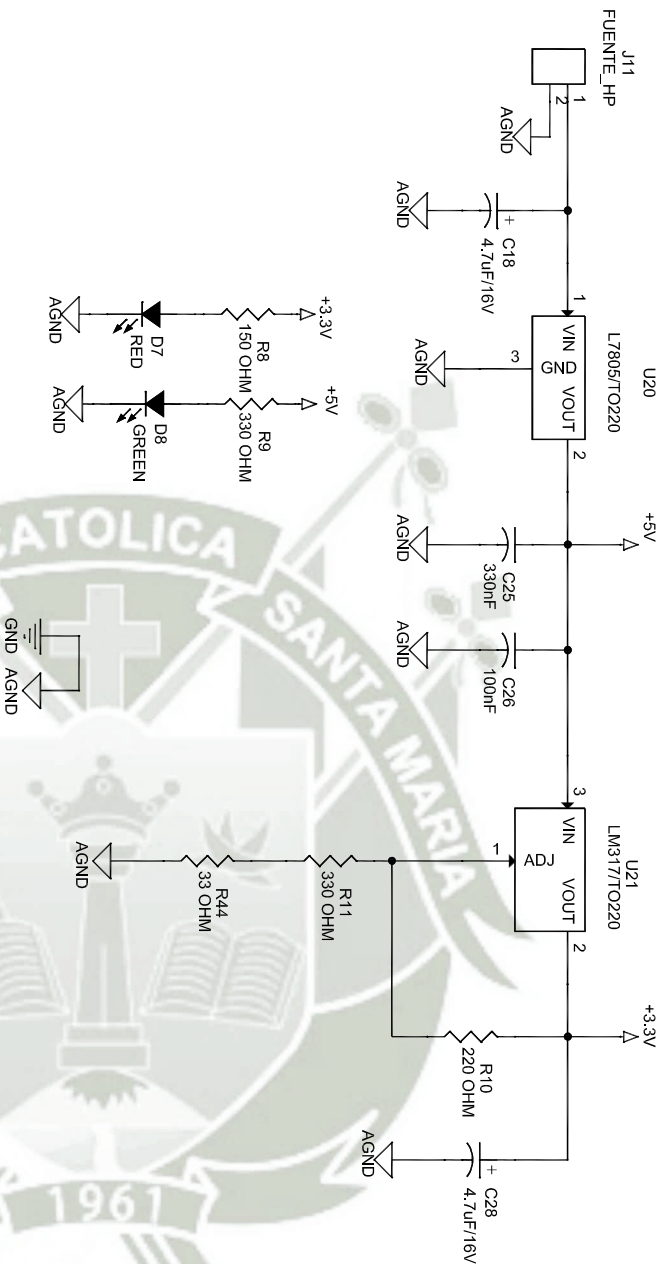
Tesis : Diseño e implementación de un controlador PID Plano N°. 08

PPIMEM

ETAPA DE POTENCIA Fecha: 18/09/13

Diseñado por : Miguel Angel Ramirez Lazo





LISTA DE MATERIALES		
C18	4.7uF/35v	U11
C25	330nF	(AQUÍ SE CONECTA LA ENTRADA DE ALIMENTACION DC >= 9V)
C26	100nF	
C28	4.7uF/35v	D7
R8	150 OHM	INDICADOR DE PRESENCIA DE VOLTAGE DE 3.3 V
R9	330 OHM	
R11	330 OHM	D8
R44	33 OHM	INDICADOR DE PRESENCIA DE VOLTAGE DE 5 V
R10	220 OHM	

CIRCUITO DE FUENTE DE ALIMENTACION

Es la encargada de proporcionar los voltajes de alimentación tanto al microcontrolador como a sus periféricos. Los Voltajes de Salida de dicho circuito son 3.3v (indicado por el led de color rojo) y 5v (indicado por el led de color verde). El valor de las resistencias y los condensadores se determino a partir de los datasheets de los reguladores L7805 y el LM317 mientras que los valores de las resistencias para los leds se seleccionaron considerando que por estos pasan a lo mucho 10mA. Finalmente, como se menciono anteriormente, el plano de tierra analogico (AGND) y el plano de tierra digital (GND) se unen en un unico punto (de preferencia el más cercano a la fuente).

UNIVERSIDAD CATOLICA DE SANTA MARIA

Tesis : Diseño e implementación de un controlador PID Plano N°. 09

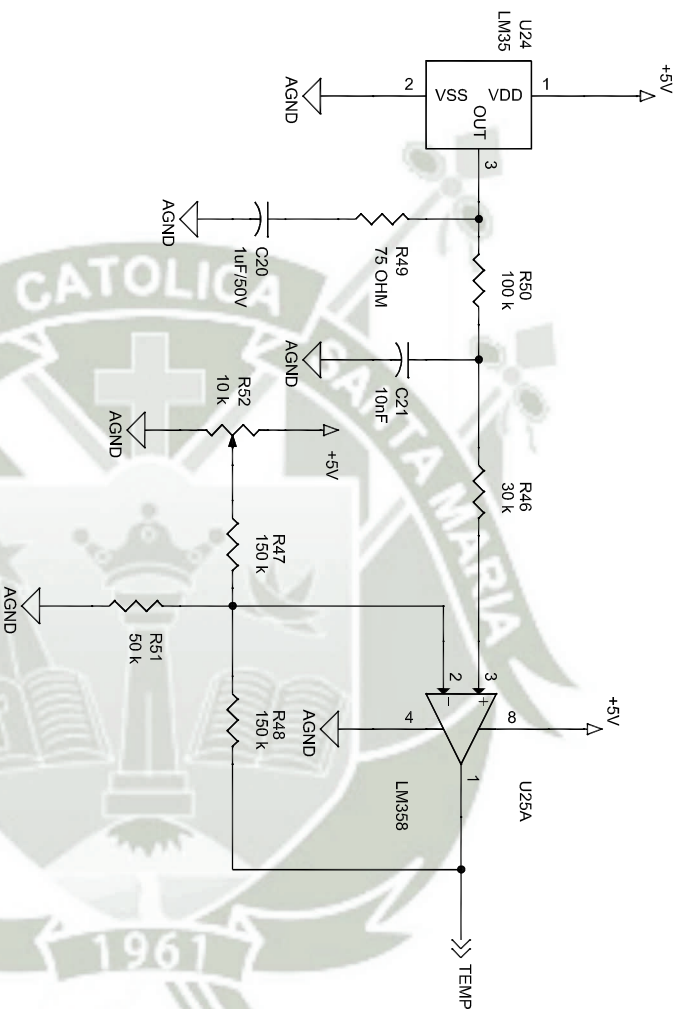
CIRCUITO FUENTE DE ALIMENTACION Fecha: 18/09/13

PPIMEM

Diseñado por : Miguel Angel Ramirez Lazo



LISTA DE MATERIALES				
U24	LM35	R49	75 OHM	
R46	30 K	C20	1uF/50V	
R47	150K	R50	100 K	
R48	150 K	C21	10nF	
R51	50 K	R52	10 K	
U25	LM35			



CIRCUITO PARA SENSADO DE TEMPERATURA

Este circuito se divide en 3 partes. La primera parte, compuesta por el LM35, se encarga de hacer la conversión de temperatura a voltaje con una ganancia de 0.01 °C/mV. Seguidamente, viene la etapa de filtrado compuesta por un circuito R-C en serie y en bypass (Los valores de estos se obtuvieron del datasheet del LM35). Finalmente, viene la etapa de amplificación compuesta por el LM358 que se encarga de elevar el nivel de voltaje que sale del filtro con una ganancia de 5V/V; además, cuenta con un potenciómetro de referencia para poder eliminar cualquier offset que estuviera presente.

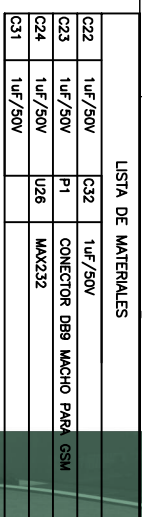
UNIVERSIDAD CATOLICA DE SANTA MARIA

Tesis : Diseño e implementación de un controlador PID Plano N°: 10

PPIMEM

CIRCUITO SENSADO DE TEMPERATURA Fecha: 18/09/13

Diseñado por : Miguel Angel Ramirez Lazo



Este circuito se encarga de elevar los niveles de voltaje del módulo de comunicación serial del microcontrolador a los niveles de voltaje requeridos por la interfaz serial RS-232 (medio físico con el que trabaja el GSM) valiéndose para este fin del integrado MAX-232 y conectándose al GSM mediante el conector DB9 de dicho circuito para hacer la comunicación inalámbrica respectiva. Cabe resaltar que los valores de los condensadores se colocaron de acuerdo a lo recomendado por el datasheet del integrado MAX232; asimismo, el condensador C24 se debe colocar lo más cerca posible de la alimentación de 5V.



Tesis : Diseño e implementación de un controlador PID Plano N°: 11

PRIMUM

INTERFAZ DE COMUNICACION

Diseñado por : Miguel Angel Ramirez Lazo