



CCADET

CENTRO DE CIENCIAS APLICADAS Y DESARROLLO
TECNOLÓGICO
UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Controlador de temperatura proporcional integral

José Castillo Hernández
Benjamín Valera Orozco
María Esther Mata Zamora

Agosto 2013

Proyecto interno

Controlador de temperatura proporcional integral

José Castillo Hernández, Benjamín Valera Orozco y María Esther Mata Zamora

Centro de Ciencias Aplicadas y Desarrollo Tecnológico, Universidad Nacional Autónoma de México,
Circuito Exterior S/N, Ciudad Universitaria AP 70-186, C.P. 04510, México D.F. México

Resumen

El control de temperatura es una técnica muy solicitada en diversos procesos de investigación, desarrollo tecnológico y producción. El desarrollo e implementación de la técnica depende principalmente de las características de la planta, del rango de operación y de la resolución requerida. En este informe reportamos el diseño y construcción de un controlador de temperatura del tipo Proporcional Integral, PI, para un horno semi-cerrado con resistencia eléctrica calefactora. El sistema controlador es un circuito electrónico sobre la base de un microcontrolador PIC18F4553 con interface USB hacia una computadora personal. El microcontrolador ejecuta el algoritmo clásico de control PI para impulsar la resistencia del horno mediante un relevador de estado sólido y la temperatura es registrada a través de un canal de conversión Analógico a Digital, AD, su respectivo termopar tipo K y circuito acondicionador de señal. El sistema permite ejecutar la interface con el usuario localmente o en forma remota desde la PC. Los experimentos reportados están dirigidos a caracterizar el horno para determinar sus parámetros de control con el objetivo de alcanzar una respuesta óptima (sin sobrepaso).

Indice

Introducción	3
Definición del problema	3
Entorno actual	3
Descripción del problema a resolver y motivación	4
Relevancia y limitaciones	4
Relación con otras áreas	4
Método	4
Objetivos y resultados esperados	5
Organización del informe	5
1 Sistema electrónico	6
1.1 Descripción del sistema electrónico	6
1.1.1 Etapa de control	6
1.1.2 Impulsor CA	7
1.1.3 Acondicionador de señal	7
1.1.4 Firmware en el microcontrolador	7
1.2 Descripción de la operación local del sistema electrónico	9
1.3 Procedimiento típico para realizar un experimento de control de temperatura de forma local	10
2 Descripción del software de operación	12
2.1 Procedimiento de instalación	12
2.1.1 Requisitos mínimos	12
2.1.2 Instalación del software	12
2.2 Descripción del programa en el ámbito del operador	12
2.3 Descripción del programa en el ámbito del programador	14
2.3.1 Clase CAboutDlg	14
2.3.2 Clase CControlPIApp	15
2.3.3 Clase CControlPIDlg	15
2.3.4 Clases CNTgraphctrl1, COleFont y CPicture	17
2.4 Procedimiento típico para controlar temperatura	17
3 Resultados y discusión	19
3.1 Resultados	20
3.2 Discusión	20
Conclusiones	21
Anexo A Diagramas electrónicos	22
A.1 Etapa de control	22
A.2 Impulsor CA y fuente de alimentación	23
A.3 Acondicionador de señal	24
Anexo B Circuito impreso	25
Anexo C Lista de material	26
Referencias	28

Introducción

Un controlador de temperatura es generalmente un sistema electrónico que permite establecer una temperatura deseada en un ambiente determinado. El sistema registra continuamente la temperatura del ambiente para impulsar los actuadores que permiten variar su temperatura. Es común que estos tipos de sistemas eleven la temperatura del ambiente con respecto al medio ambiente mediante una resistencia eléctrica calefactora. Para el registro de la temperatura se pueden utilizar una gran variedad de transductores como termopares, termistores y dispositivos de estado sólido.

En la implementación de un sistema controlador de temperatura se deben tomar en cuenta las características de la planta y el tipo de respuesta deseada. En la actualidad, muchos sistemas controladores combinan hardware y software para el control de temperatura. Un sistema de este tipo consta de un procesador dedicado a ejecutar algoritmos de control digital e interfaces analógico a digital AD para transductores y digital a analógico DA para actuadores. El esquema general de control se muestra en la figura 1.

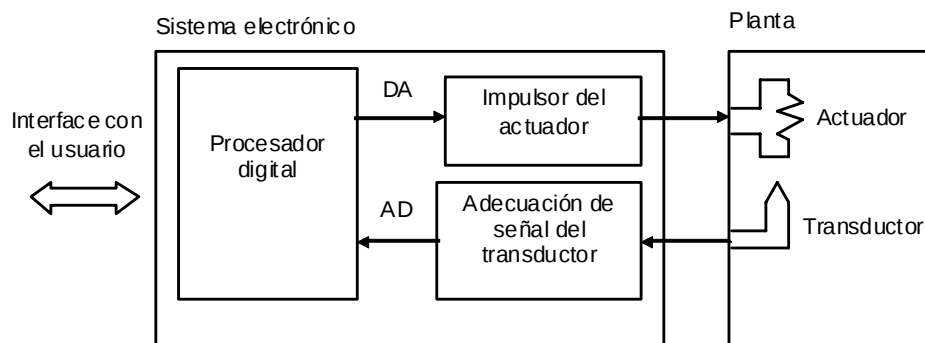


Figura 1. Sistema de control de temperatura.

Definición del problema

En este trabajo se plantea el problema de diseñar y construir un sistema electrónico controlador de temperatura para un horno semi-cerrado con resistencia calefactora del cual se desconocen sus características. Parte del presente trabajo es determinar los parámetros de operación óptimos que nos permitan una respuesta sin sobrepaso y habilitarlo como infraestructura para diversos experimentos en el Laboratorio de Materiales y Nanotecnología.

Entorno actual

En la actualidad existen diversos dispositivos electrónicos comerciales para el control de temperatura desde sistemas simples y económicos hasta controladores de procesos industriales. En este ambiente enorme y complejo abundan las opciones tanto de selección de dispositivos como elección de opciones auxiliares y no resultaría difícil hacer malas o costosas selecciones. Como ejemplo tenemos la abundante información que Omega [Omega, 2013] ofrece a sus clientes.

Por otra parte, los sistemas de control Proporcionales Integrales y Derivativos, PID, han demostrado su robustez y confiabilidad, razón por la cual se utilizan desde 1910 hasta nuestros días [Ang et al, 2005].

Un antecedente al presente trabajo es el control de temperatura reportado en [Montiel et al, 2010]. En este dispositivo, el procesador está formado por una tarjeta de adquisición de datos USB-6211 lo cual eleva su costo y además la tarjeta resulta excesiva para la aplicación.

Descripción del problema a resolver y motivación

La motivación en el presente trabajo es desarrollar una solución específicamente elaborada para un horno en particular, requisitos específicos del cliente y disminuir el costo de materiales con respecto a lo reportado en [Montiel et al, 2010].

Los requisitos del cliente se mencionan a continuación.

- Interface de operación local (despliegue y teclado) y remota (computadora personal del tipo laptop).
- Horno de características desconocidas pero suministrado por el cliente.
- Termopar tipo K.
- Costo económico del material.
- Sobrepaso nulo de temperatura sobre la temperatura deseada.

Relevancia y limitaciones

La principal relevancia del presente proyecto es la creación de infraestructura de desarrollo propio y la experiencia que adquiere el personal participante.

Por el contrario, las principales limitaciones están relacionadas con las limitaciones propias de los algoritmos de control PI y con las limitaciones de los sistemas de control discretos. En primer lugar un control PID es un sistema lineal y esto puede ocasionar oscilaciones en una planta no lineal. Por otra parte la influencia del ruido afecta el cálculo de los términos derivativos e integrales lo que puede ocasionar saturaciones en los niveles de voltaje aplicados a la planta.

Relación con otras áreas

Existe gran relación con las áreas de control discreto, microprocesadores, electrónica analógica, electrónica digital y medición e instrumentación.

Método

El método utilizado consiste en desarrollar un sistema de control discreto PI con base en un microcontrolador PIC18F4553 y circuitos de interface con transductores y actuadores como se muestra en la figura 2.

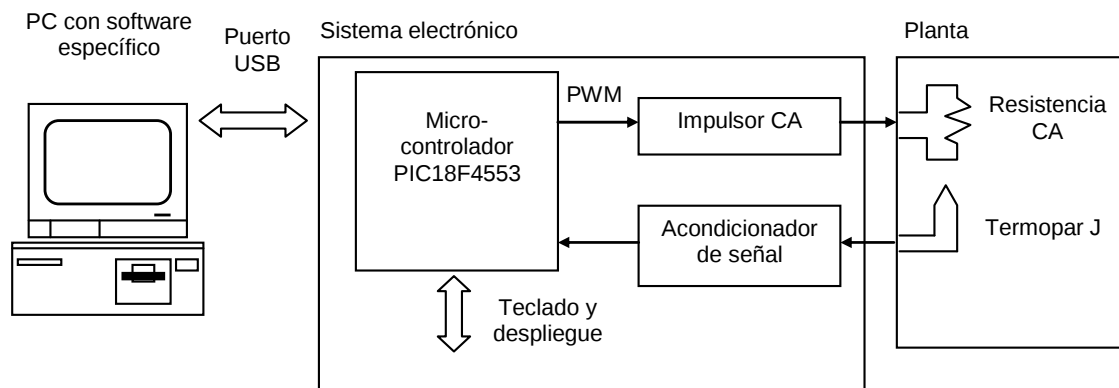


Figura 2. Metodología.

En la figura 2 un sistema electrónico implementa tres funciones principales:

- El algoritmo de control PI. El microcontrolador genera una señal por modulación del ancho de pulso, PWM la cual se utiliza para impulsar la carga de corriente alterna CA. La señal PWM varia el nivel de potencia CA con lo cual es posible la acción proporcional del algoritmo. La señal de realimentación de temperatura del algoritmo PI es suministrada al microcontrolador a través de una entrada de conversión analógica a digital, AD, de 12 bits. El algoritmo de control entonces genera el nivel de potencia eléctrica que resulta de procesar el nivel de temperatura deseado y la temperatura real del sistema mediante la ecuación en diferencias clásica del controlador PI.
- La interface con el usuario. Se utiliza principalmente para observar el estado real de la planta e ingresar la temperatura deseada. Localmente se utiliza un despliegue y teclado. De forma remota se utiliza una computadora con conexión USB y software PC de desarrollo específico.
- La adecuación de señales. Se requieren dos etapas para adecuar las señales de entrada y salida al microcontrolador. En primer lugar la señal digital de PWM debe ser adecuada en potencia para impulsar la resistencia calefactora en el horno. La técnica utilizada se conoce como modulación por ancho de pulso y requiere de dispositivos de potencia de estado sólido y circuitos acopladores ópticos. En segundo lugar la señal del transductor de temperatura, termopar tipo K, debe cubrir la excursión analógica de la entrada en el microcontrolador y compensar electrónicamente el offset de voltaje conocido como "unión fría". Lo anterior se implementa con circuitos amplificadores operacionales específicos para la conexión de termopares.

El software específico implementa la interface entre el usuario y el sistema controlador a través de un enlace USB emulando una terminal electrónica de transferencia de comandos y datos.

Objetivos y resultados esperados

- Diseñar y construir un sistema electrónico controlador de temperatura para un horno semi-cerrado.
- Obtener sobrepaso nulo en la especificación de la temperatura deseada.
- Desarrollar software para PC para el control y registro continuo de temperatura.

Organización del informe

En el capítulo uno del informe se describe el sistema electrónico desarrollado y la implementación de los circuitos de control, impulsor CA y acondicionador de señal. En el capítulo dos se describe el software desarrollado en Visual C++ en los ámbitos del usuario y del programador. En el capítulo tres se muestran los resultados obtenidos de un experimento de control de temperatura que verifican el desempeño del sistema. Finalmente se mencionan las conclusiones y se plantea el trabajo a futuro. Los anexos contienen los elementos útiles para reproducir el controlador desarrollado y se incluye un CD conteniendo la siguiente estructura de directorios.

```
Informe
CodigoFuente
    PIC
    VisualC
Programa
    Instalacion
    Manejadores
```

1. Sistema electrónico

En este trabajo se desarrolló un sistema electrónico con base en un microcontrolador PIC18F4553, circuitos para el acondicionamiento de la señal de un termopar tipo K y etapa impulsora de potencia de corriente alterna. El esquema general se presenta en la figura 1.1.

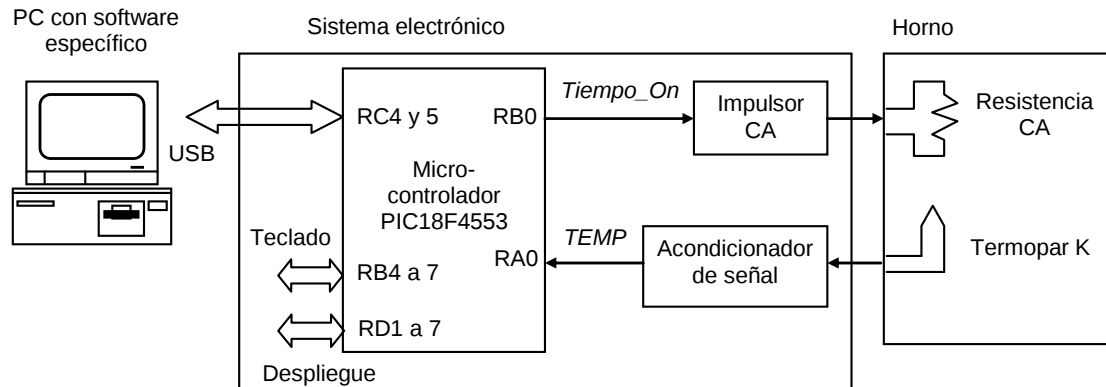


Figura 1.1. Esquema general para el control de temperatura.

El microcontrolador genera una señal por modulación del ancho de pulso, PWM, *Tiempo_On*, la cual se utiliza para impulsar la carga de corriente alterna CA. La señal PWM varía el nivel de potencia CA con lo cual es posible la acción proporcional del algoritmo. La señal de realimentación de temperatura del algoritmo PI, *TEMP*, es suministrada al microcontrolador a través de una entrada de conversión Analógica a Digital, AD, de 12 bits. El algoritmo de control entonces genera el nivel de potencia eléctrica que resulta de procesar el nivel de temperatura deseado y la temperatura real del sistema mediante la ecuación en diferencias clásica del controlador PI. El microcontrolador se encarga de administrar la interface con el usuario a través de la conexión USB o teclado y despliegue.

1.1 Descripción del sistema electrónico

A continuación se describen las etapas del sistema electrónico. Para mayores detalles consulte los anexos A y B.

1.1.1 Etapa de control

La etapa de control está formada por el microcontrolador PIC18F4553 [Microchip, 2007] y sus dispositivos asociados para impulsar su oscilador local, establecer los niveles de referencia en el convertidor AD y entradas y salidas digitales de propósito general. La figura 1.2 muestra la arquitectura básica del microcontrolador para la presente aplicación.

Con referencia a la figura 1.2, localmente el controlador de temperatura utiliza un despliegue en RD1 a RD7 y teclado de 4 botones en RB4 a RB7 para presentar e ingresar parámetros. De forma remota se utiliza una computadora con conexión USB en RC4 y RC5 y software para PC de desarrollo específico.

La señal *TEMP*, proveniente de la etapa de acondicionamiento de señal, es convertida en el canal AD RA0. Adicionalmente se requiere una referencia de voltaje en RA3. La referencia se genera con un circuito integrado LM336 [Texas Instruments, 2005].

La señal de PWM en RB0, *Tiempo_On*, es enviada a la etapa impulsora de CA en donde se acondiciona para aplicar potencia a la carga.

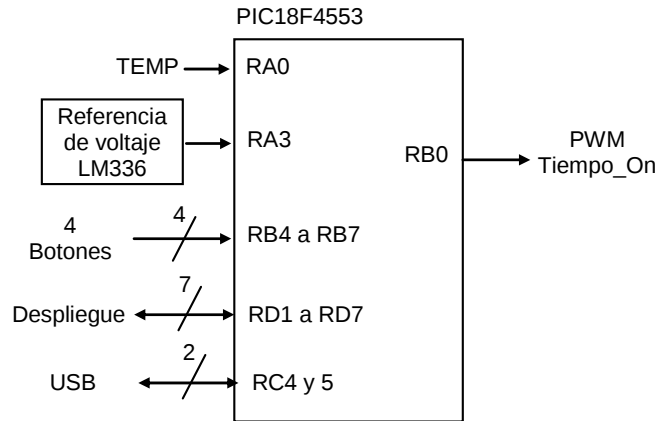


Figura 1.2. Etapa de control.

1.1.2 Impulsor CA

La señal digital Tiempo_On, proveniente de la etapa de control, debe ser adecuada en potencia para impulsar la resistencia calefactora en el horno. La técnica utilizada se conoce como modulación por ancho de pulso y requiere de un dispositivo de potencia de estado sólido, MAC15A8 [On Semiconductor, 2008], y un circuito acoplador óptico, MOC3031 [Fairchild, 2010]. La figura 1.3 muestra la configuración utilizada.

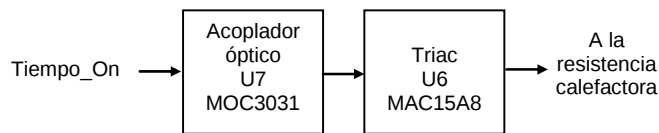


Figura 1.3. Impulsor CA.

1.1.3 Acondicionador de señal

La señal del transductor de temperatura, TEMP, del termopar tipo K, debe cubrir la excursión analógica de la entrada en RA0 y compensar electrónicamente el offset de voltaje conocido como “unión fría”. Lo anterior se implementa con circuitos amplificadores operacionales específicos para la conexión de termopares, AD595 [Analog Devices, 1999]. La configuración empleada utiliza el amplificador diferencial U4, un amplificador inversor, U5A y un filtro paso bajas U5B, como se muestra en la figura 1.4.

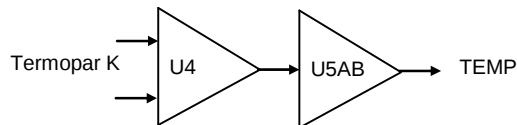


Figura 1.4. Acondicionador de señal.

1.1.4 Firmware en el microcontrolador

El firmware en el microcontrolador básicamente consiste de un programa de terminal USB que espera recibir la notificación del programa en la PC para ejecutar alguno de los comandos establecidos o leer el convertidor digital a analógico en RA0, Temp, y calcular la potencia aplicada a la carga, Tiempo_On, como se muestra en la figura 1.5.

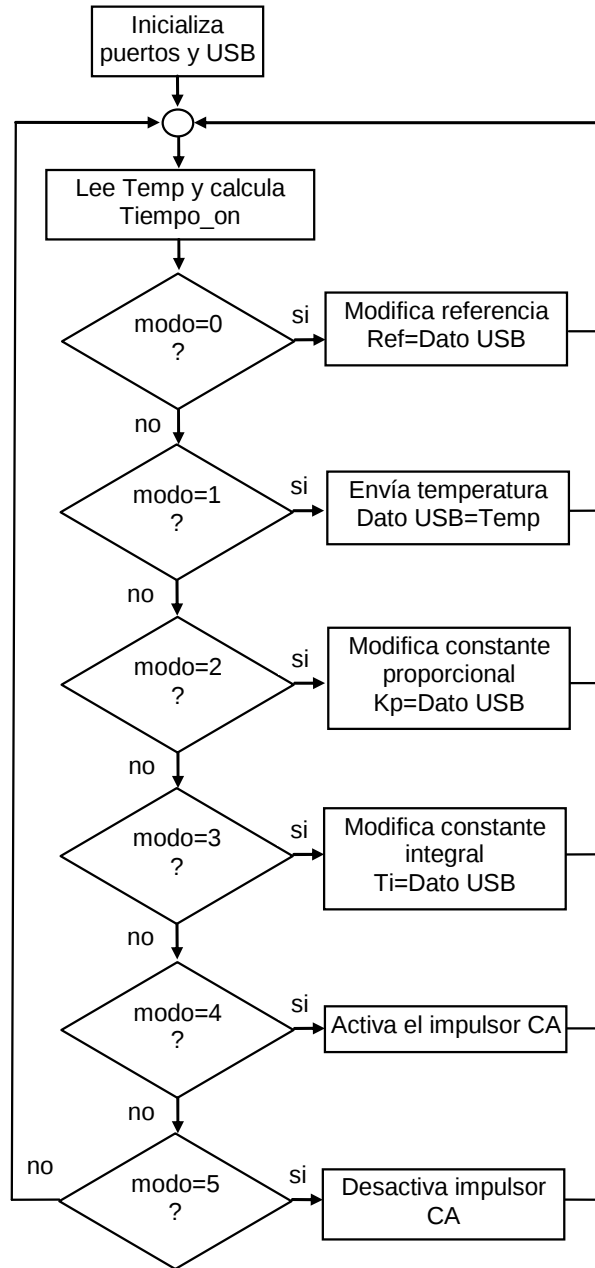


Figura 1.5. Terminal USB.

El algoritmo de control PI que adquiere la temperatura actual, *Temp*, calcula la ecuación en diferencias, *Tiempo_On* y que incluye anti windup [Chakir et al, 2006], se presenta en el diagrama de flujo de la figura 1.6.

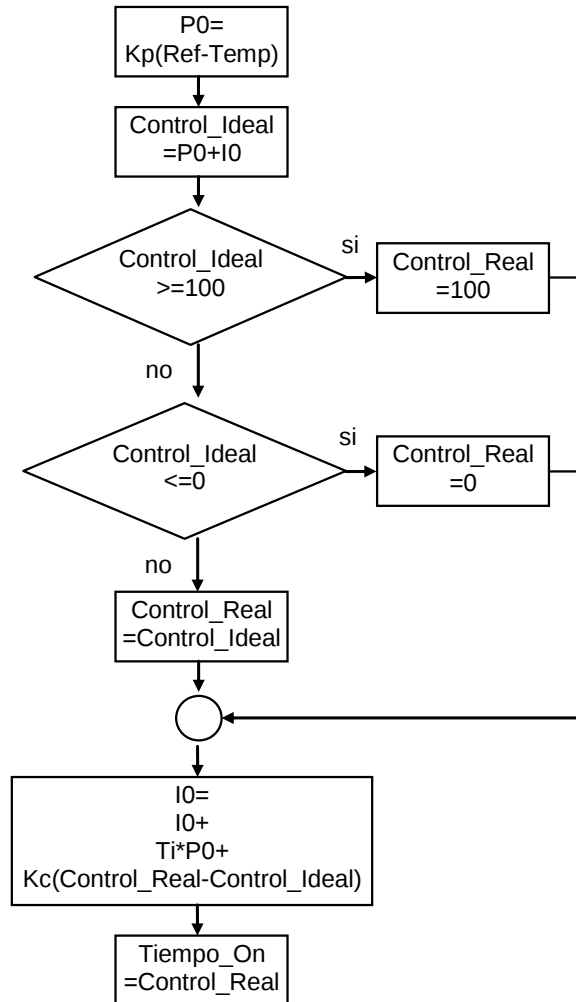


Figura 1.6. Diagrama de flujo del firmware en el PIC18F4553.

En la figura 1.6 $P0$ es la acción proporcional e $I0$ es la acción integral. Los valores de Kp , Ti y Kc son constantes determinadas de forma empírica pero que pueden ser modificadas mediante el enlace USB y su comando respectivo. Los valores de las constantes determinan en gran medida la respuesta de la planta.

El algoritmo fue programado en lenguaje C para microcontroladores de MicroChip [Custom Computer, 2007]. Los detalles de implementación se encuentran en el CD anexo en el directorio CódigoFuente/PIC

1.2 Descripción de la operación local del sistema electrónico

El sistema electrónico para el control de temperatura puede ser operado localmente mediante su teclado y despliegue incorporados o de forma remota desde una PC con software específico. En esta sección se describe la operación local.

El sistema electrónico, operado localmente, tiene tres modos de funcionamiento:

- Modo de Inicio. Aquí se especifica si el sistema electrónico va a funcionar con conexión USB a una computadora con software específico o de manera local mediante su teclado y despliegue.

- **Menú de Opciones.** En este modo de operación el usuario puede ingresar, mediante el teclado, los siguientes parámetros de operación: Temperatura de referencia (Ref), Apagar o encender la etapa impulsora de CA, Constante Proporcional (Kp) y Constante Integral (Ti).
- **Modo de Operación.** En este modo el sistema electrónico realiza el control de temperatura energizando la carga para llevar el horno hacia la temperatura de referencia. En el despliegue se presentan los datos de temperatura de referencia y temperatura actual de la planta en grados centígrados.

La siguiente tabla muestra los diferentes modos de operación y la funcionalidad que se habilita en cada tecla.

Despliegue	Botones 1 2 3 4				Comentarios	Modo de operación
= TEC o USB ? = === TECLADO ===	Teclado	USB		OK	Para configurar seleccione Teclado o USB y presione OK	Inicio
= CONTINUAR = === NO ===		Si	No		Presione Si para continuar con el modo Opciones o No para regresar	
= TEMPERATURA = 120.00	-	+	Menu -	Menu +	Presione + o - para modificar el valor de temperatura de referencia y Menú+ Menú- para desplazarse por las Opciones	Opciones
=== EQUIPO === *** APAGADO ***		On			Presione On para encender el equipo y acceder al modo de Operación	
== Cte Kp ¿ == 2.20	-	+	Menu -	Menu +	Presione + o - para modificar el valor de la constante proporcional y Menú+ Menú- para desplazarse por las Opciones	
== Cte Ti ? == 1600.0	-	+	Menu -	Menu +	Presione + o - para modificar el valor de la constante integral y Menú+ Menú- para desplazarse por las Opciones	
SP = 120.0 ° C T = 21.40 ° C					Presione cualquier tecla para regresar al menú de Opciones y apagar la etapa impulsora de CA	Operación

Tabla 1.1. Modos de operación en el sistema electrónico

1.3 Procedimiento típico para realizar un experimento de control de temperatura de forma local

Para el siguiente ejemplo de control de temperatura se asume lo siguiente:

- Temperatura de referencia 120°C.
- Constantes del controlador por omisión ya que son las apropiadas para el horno utilizado.
- Horno semi-cerrado de asbesto.
- Termopar K suministrado con el equipo.

1. Realice las conexiones de la figura 1.7.

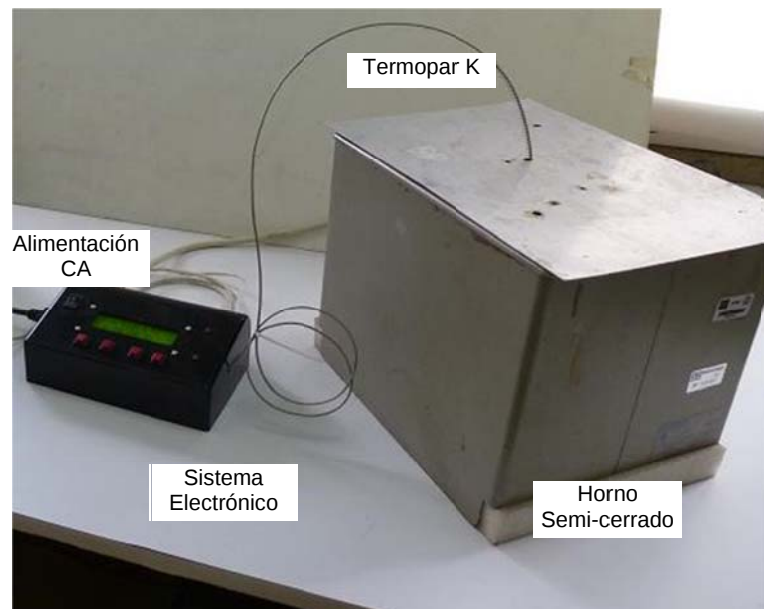


Figura 1.7. Conexiones para el control de temperatura en modo local.

2. Encienda el sistema electrónico y espere a que se deshabilite el letrero de bienvenida.
3. Presione el botón 4 para aceptar el modo de operación con teclado.
4. Presione el botón 2 para continuar con el modo de Opciones.
5. Presione el botón 4 para ingresar a la opción que permitirá iniciar el control de temperatura.
6. Presione el botón 2 para encender la etapa impulsora CA e iniciar el control de temperatura.
7. Observe el despliegue durante algunos minutos para verificar que la temperatura T se incrementa hasta alcanzar su referencia SP como se muestra en la figura 1.8



Figura 1.8. Control de temperatura en ejecución.

2. Descripción del software de operación

El sistema controlador de temperatura puede ser operado de manera local, mediante su despliegue y teclado, o de manera remota a través de una computadora conectada mediante la interface USB. El software de operación permite registrar la temperatura de la planta, especificar la temperatura deseada y modificar otros parámetros del algoritmo de control. En esta sección se describe el software de operación desarrollado con Visual C++ y la MFC [Kruglinski et al, 1998]. Se abordan dos enfoques en la descripción. En primer lugar, una descripción en el ámbito del operador como manual de referencia. En segundo lugar, en el ámbito del programador para futuras modificaciones o depuraciones. Además se proporciona una guía para la instalación del software y un procedimiento típico para realizar un experimento de control de temperatura.

2.1 Procedimiento de instalación

Para instalar el software en una computadora anfitriona se requiere el CD anexo al presente informe.

2.1.1 Requisitos mínimos

- Procesador Core 2 Duo
- 4 GB de RAM
- 1 GB de espacio libre en disco duro
- Monitor 1600X900 pixeles de resolución y 24 bits
- Windows Vista Home Premium de 64 bits
- Puerto USB disponible

2.1.2 Instalación del software

1. Copiar el contenido del directorio Programa/Instalacion del CD anexo a un directorio en el disco duro de la computadora anfitriona, por ejemplo, C:\ControlPI.
2. Conectar el cable USB entre la computadora y el sistema electrónico. Energizar el sistema electrónico.
3. En este momento Windows indicará la presencia de nuevo hardware encontrado. Indicar que el software controlador se instalará desde el directorio Programa/Manejadores en la unidad de disco.
4. Verifique que el controlador ha sido correctamente instalado desde el Panel de control de Windows. El controlador se deberá desplegar como "Custom USB Devices/UNAM".
5. Con el "Control de Cuentas de Usuario de Windows" deshabilitado y desde el directorio C:\ControlPI, ejecute en la línea de comandos el siguiente comando:

```
regsvr32 ntgraph.ocx
```

6. Recibirá la notificación que el archivo ocx, o control Activex ha sido correctamente instalado. De otra manera regrese al paso número 5 y verifique el estado del control de cuentas de usuario.
7. Para verificar la correcta operación del software y del sistema electrónico, desde el explorador de archivos de Windows ejecute el archivo C:\ControlPI/ControlPI.exe. Presione el botón "Lee Temp" para verificar que la temperatura en la planta se aproxima a la temperatura ambiente.

2.2 Descripción del programa en el ámbito del operador

El programa ControlPI.exe despliega una pantalla como la mostrada en la figura 2.1.

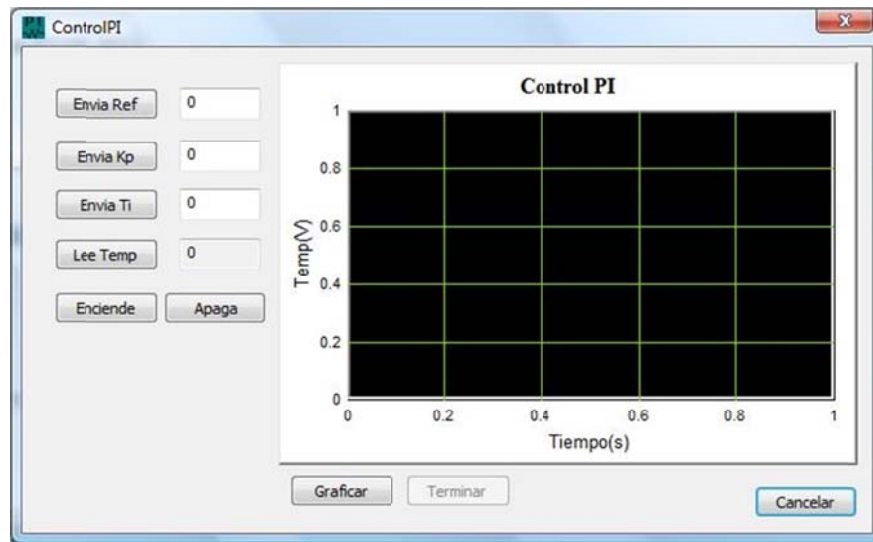


Figura 2.1. Pantalla de operación.

La pantalla de operación contiene los siguientes elementos de control:

- El botón “Envia Ref”. Utilizado para enviar al sistema electrónico la temperatura deseada en grados centígrados. El valor enviado se introduce sobre la caja de texto a la derecha de éste botón.
- El botón “Envia Kp”. Utilizado para enviar al sistema electrónico la constante proporcional. El valor enviado se introduce sobre la caja de texto a la derecha de éste botón.
- El botón “Envia Ti”. Utilizado para enviar al sistema electrónico la constante integral. El valor enviado se introduce sobre la caja de texto a la derecha de éste botón.
- El botón “Lee Temp”. Utilizado para enviar al sistema electrónico la instrucción para que regrese la temperatura de la planta. El valor recibido se despliega sobre la caja de texto a la derecha de éste botón.
- El botón “Enciende”. Utilizado para iniciar el proceso de control de temperatura y ajuste a la temperatura deseada introducida por “Envia Ref”.
- El botón “Apaga”. Utilizado para detener el proceso de control de temperatura y desenergizar la resistencia del horno.
- El botón “Graficar”. Utilizado para graficar la temperatura de la planta contra el tiempo. La temperatura se grafica en el panel “Control PI”. Al presionar el botón se despliega una caja de diálogo en donde se introduce el nombre del archivo para respaldar los datos. Una vez aceptado el nombre del archivo inicia el proceso de control, graficación, escritura en archivo y se habilita el botón “Terminar”.
- El botón “Terminar”. Utilizado para abortar el proceso iniciado al presionar el botón “Graficar”.
- El botón “Cancelar”. Utilizado para salir de la aplicación.
- El despliegue “Control PI”. Utilizado para desplegar los datos de temperatura durante el proceso de control.

2.3 Descripción del programa en el ámbito del programador

El programa “ControlPI” es una aplicación del tipo caja de diálogo desarrollado sobre la base de la MFC en lenguaje Visual C++ con Visual Studio como entorno de desarrollo [Kruglinski et al, 1998]. El proyecto se organiza en las seis clases que se muestran en la figura 2.2.

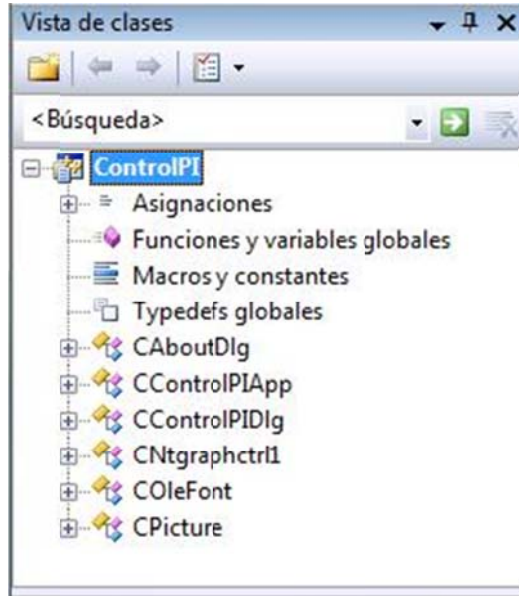


Figura 2.2. Clases del proyecto “ControlPI”.

Para compilar adecuadamente el programa se requiere que el archivo mpusbapi.dll resida en la carpeta del código fuente del proyecto. El archivo es de libre distribución por parte de Microchip [JVR, 2013]. En forma adicional, la sección de inicialización del programa debe cargar la librería DLL como se describe en la siguiente sección.

Las clases se describen a continuación. El código fuente completo se encuentra en el CD anexo en el directorioCodigoFuente/VisualC

2.3.1 Clase CAboutDlg

La clase CAboutDlg se crea por omisión en el ambiente de desarrollo y se utiliza para desplegar la caja de diálogo “Acerca de”. Sus métodos se muestran en la figura 2.3 y su descripción a continuación.

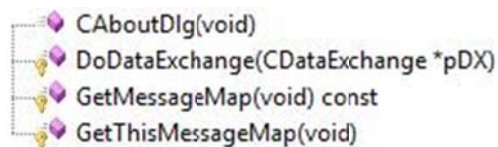


Figura 2.3. Clase CAboutDlg.

CAboutDlg(); Constructor insertado por el ambiente de desarrollo. Por omisión no contiene código alguno.

DoDataExchange(CDataExchange* pDX); Insertado por omisión por el ambiente de desarrollo. Agrega la compatibilidad para el intercambio de datos, DDX, y la validación de datos, DDV.

GetMessageMap(void); Habilita la estructura de manejo de mapas de mensajes. Insertado por el ambiente de desarrollo. Por omisión esta clase no maneja mapas de mensajes.

GetThisMessageMap(void); Habilita la estructura de manejo de mapas de mensajes. Insertado por el ambiente de desarrollo. Por omisión esta clase no maneja mapas de mensajes.

2.3.2 Clase CControlPIApp

La clase CControlPIApp se crea por omisión en el ambiente de desarrollo y se utiliza para controlar la ejecución de la aplicación. Sus métodos se muestran en la figura 2.4 y su descripción a continuación.

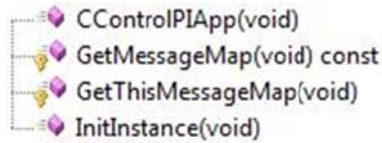


Figura 2.4. Clase CControlPIApp.

CControlPIApp(); Constructor insertado por el ambiente de desarrollo. Por omisión no contiene código alguno.

GetMessageMap(void); Habilita la estructura de manejo de mapas de mensajes. Insertado por el ambiente de desarrollo. Por omisión esta clase no maneja mapas de mensajes.

GetThisMessageMap(void); Habilita la estructura de manejo de mapas de mensajes. Insertado por el ambiente de desarrollo. Por omisión esta clase no maneja mapas de mensajes.

InitInstance(); Habilita el tipo de controles que utiliza la aplicación. Aunque no se implementa código alguno. Se puede utilizar para la inicialización estándar en los archivos de registro y las acciones a ejecutar cuando se presionan los botones “OK” y “Cancel”.

2.3.3 Clase CControlPIDlg

La clase CControlPIDlg se crea por omisión en el ambiente de desarrollo y se utiliza para implementar las funciones específicas del control de temperatura. Sus métodos se muestran en la figura 2.5 y su descripción a continuación.

CControlPIDlg(CWnd* pParent = NULL); Constructor estándar de la aplicación. Se utiliza para inicializar variables globales.

DoDataExchange(CDataExchange* pDX); Permite el intercambio de datos entre la caja de diálogo y los métodos de la clase.

GetMessageMap(void); Declara el mapa de mensajes de la aplicación de manera que se puedan atender los diversos eventos que se generan con la interacción del usuario o con los mensajes de Windows.

GetThisMessageMap(void); Declara el mapa de mensajes de la aplicación de manera que se puedan atender los diversos eventos que se generan con la interacción del usuario o con los mensajes de Windows.

OnBnEnviaApaga(); Es el método que atiende el evento por presionar el botón “Apaga”. Realiza el llamado al método PICEnviaApaga.

OnBnEnviaEnciende(); Es el método que atiende el evento por presionar el botón “Enciende”. Realiza el llamado al método PICEnviaEnciende.

OnBnEnviaKp(); Es el método que atiende el evento por presionar el botón “Envia Kp”. Extrae el valor de la constante de proporcionalidad de la aplicación y realiza el llamado a la función PICEnviaKp.

OnBnEnviaReferencia(); Es el método que atiende el evento por presionar el botón “Envia Ref”. Extrae el valor de la temperatura deseada de la aplicación y realiza el llamado a la función PICEnviaRef.

OnBnEnviaTi(); Es el método que atiende el evento por presionar el botón “Envia Ti”. Extrae el valor de la constante integral de la aplicación y realiza el llamado a la función PICEnviaRef.

OnBnGraficar(); Inicia las variables para graficar automáticamente a partir de un temporizador. Despliega la caja de diálogo para obtener el nombre de archivo en donde se respaldaran los datos de temperatura contra tiempo. Si se presiona el botón “Guardar”, se almacena el primer dato en el archivo y se despliega el primer valor en la gráfica, es decir, la temperatura en el tiempo cero. Se inicia el temporizador a 1 segundo.

OnBnLeeTemp(); Realiza el llamado a las funciones PICEnviaTemp y PICResultado16 para desplegar en la aplicación el dato de temperatura actual de la planta.

OnBnTerminar(); Desactiva el temporizador y cierra el archivo de respaldo de datos.

OnInitDialog(); Realiza la inicialización estándar de la aplicación. Carga la DLL mpushapi.dll en memoria y obtiene la dirección de sus funciones. Inicia el despliegue gráfico.

OnPaint(); Aunque no se dibuja en la aplicación, establece el contexto para dibujar sobre la aplicación. Realiza la inicialización estándar del icono: Centra el icono en el rectángulo del cliente y lo dibuja.

OnQueryDragIcon(); El sistema llama a esta función para obtener el cursor que se muestra mientras el usuario arrastra la ventana minimizada.

OnSysCommand(UINT nID, LPARAM lParam); Despliega la caja de diálogo “Acerca de...”.

OnTimer(UINT_PTR nIDEvent); Es el método del temporizador que se ejecuta cada segundo. Realiza el llamado a las funciones PICEnviaTemp y PICResultado16 para obtener la temperatura actual de la planta. Almacena la temperatura en el archivo de respaldo y la inserta en la gráfica.

PICEnviaApaga(); Envía el código de entrada para apagar la resistencia al sistema electrónico por medio de la función USBSendPacket.

PICEnviaEnciende(); Envía el código de entrada para encender la resistencia al sistema electrónico por medio de la función USBSendPacket.

PICEnviaKp(UINT LB, UINT HB); Envía el código de entrada y constante de proporcionalidad al sistema electrónico por medio de la función USBSendPacket.

PICEnviaRef(UINT LB, UINT HB); Envía el código de entrada y temperatura deseada al sistema electrónico por medio de la función USBSendPacket.

PICEnviaTemp(); Envía el código de entrada para encuestar la temperatura actual de la planta al sistema electrónico por medio de la función USBSendPacket.

PICEnviaTi(UINT LB, UINT HB); Envía el código de entrada y constante integral al sistema electrónico por medio de la función USBSendPacket.

PICResultado16(void); Recibe un paquete de dos bytes del sistema electrónico por medio de la función USBReceivePacket y lo convierte a valor entero.

USBClosePipes(void); Cierra los tubos de comunicación USB.

USBOpenPipes(void); Abre los tubos de comunicación USB.

USBReceivePacket(BYTE* ReceiveData, DWORD* ReceiveLength); Llama a la función USBOpenPipes. Obtiene información del tubo de comunicación y llama a la función USBClosePipes.

USBSendPacket(byte* SendData, DWORD SendLength); Llama a la función USBOpenPipes. Escribe información en el tubo de comunicación y llama a la función USBClosePipes.

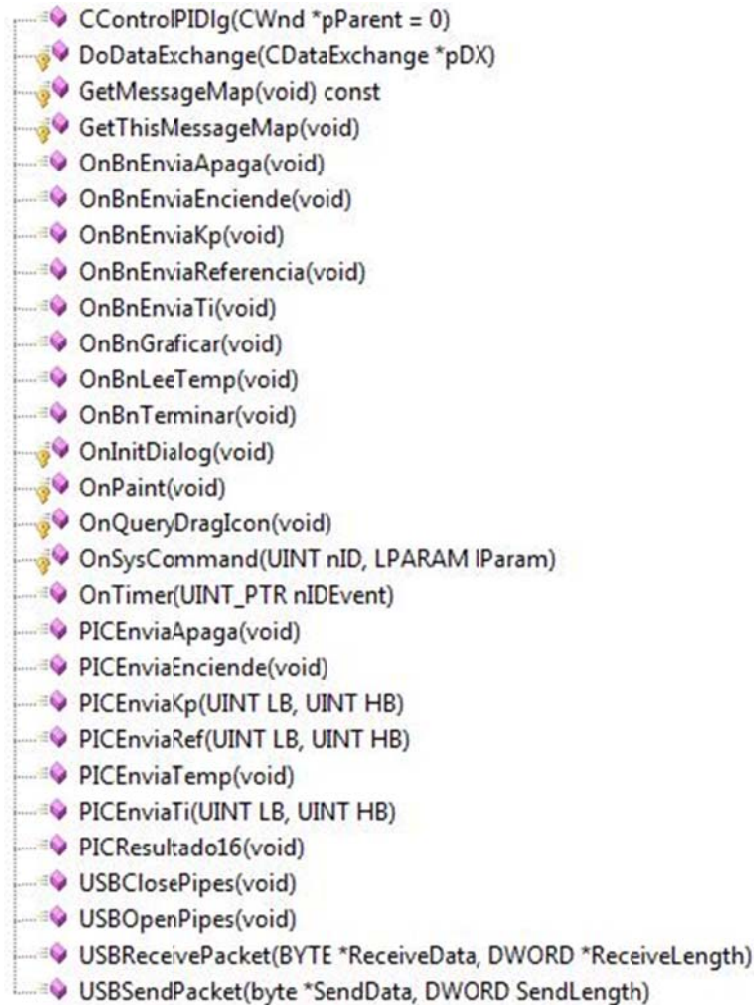


Figura 2.5. Clase CControlPIDlg.

2.3.4 Clases CNTgraphctrl1, COleFont y CPicture

Estas clases son generadas automáticamente por el ambiente de desarrollo cuando se inserta en el proyecto el control activex ntgraph.ocx.

2.4 Procedimiento típico para controlar temperatura

- Realizar las conexiones que se muestran en la figura 2.6.

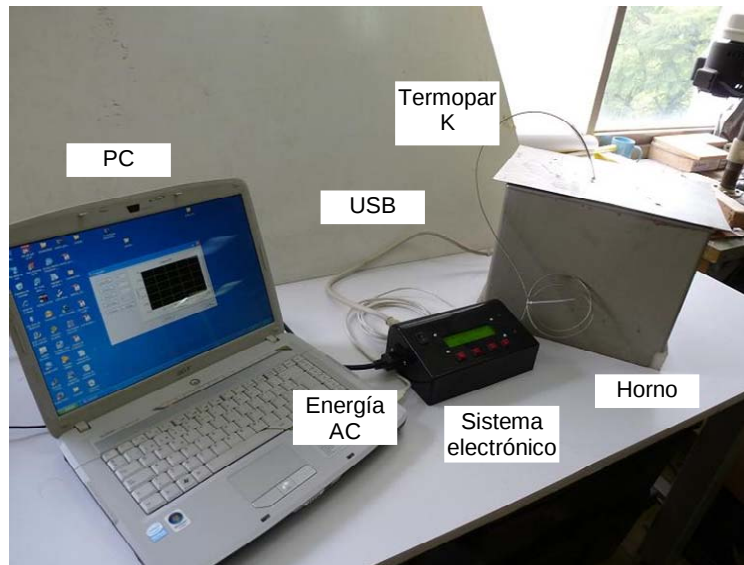


Figura 2.6. Conexiones para el controlador de temperatura.

- Energizar los dispositivos
- En la computadora, ejecutar el archivo C:\ControlPI\ControlPI.exe y presionar el botón “Lee Temp” de manera que se observe una temperatura cercana a la del ambiente. De esta manera se verifica la conexión correcta de los componentes.
- Intrudicir la temperatura deseada o setpoint en la caja de texto al lado derecho del botón “Envia Ref” y presionar dicho botón. De preferencia no modificar las constantes proporcional o integral.
- Presionar el botón “Enciende” para iniciar el proceso de control de temperatura.
- Presionar el botón “Graficar” para ingresar el nombre de archivo para almacenar y mostrar los datos de temperatura contra tiempo.
- El proceso de control de temperatura se puede abortar en cualquier momento al presionar el botón “Apaga”.
- El proceso de graficación y almacenamiento de temperatura se puede abortar en cualquier momento al presionar el botón “Terminar”

3. Resultados y discusión

Como resultados se presentan el sistema electrónico para el control de temperatura y la interface gráfica para computadora personal que se muestran en las figuras 4.1 a 4.3.



Figura 4.1. Sistema para el control de temperatura.

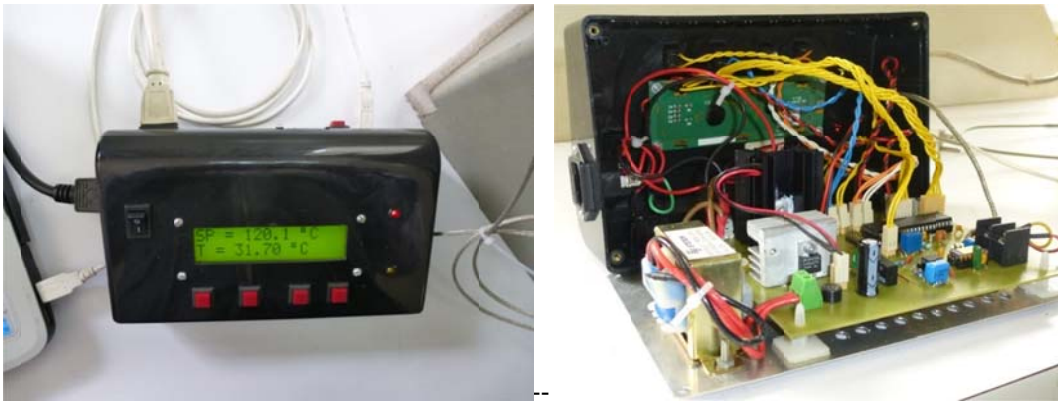


Figura 4.2. Sistema electrónico de control de temperatura.

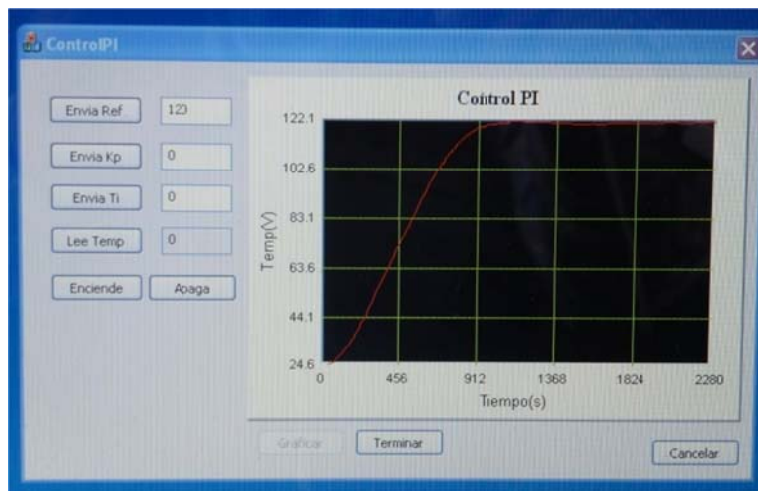


Figura 4.3. Interface gráfica para computadora personal.

3.1 Resultados

Adicionalmente se desarrolló una prueba para el control de temperatura en un horno semi-cerrado. La temperatura de referencia fue 120°C y el archivo de datos se muestra graficado en la figura 4.4. La curva 1 en la gráfica representa la respuesta de la planta bajo un control proporcional con un claro sobretiro en su respuesta. Las curvas 2 y 3 ilustran la repuesta del sistema bajo el control PI diseñado. Claramente se observa que el sistema mantiene una temperatura constante y estable con una variación menor de 3°C alrededor de la temperatura de referencia y que la curva de calentamiento durante el proceso transitorio no presenta sobrepaso.

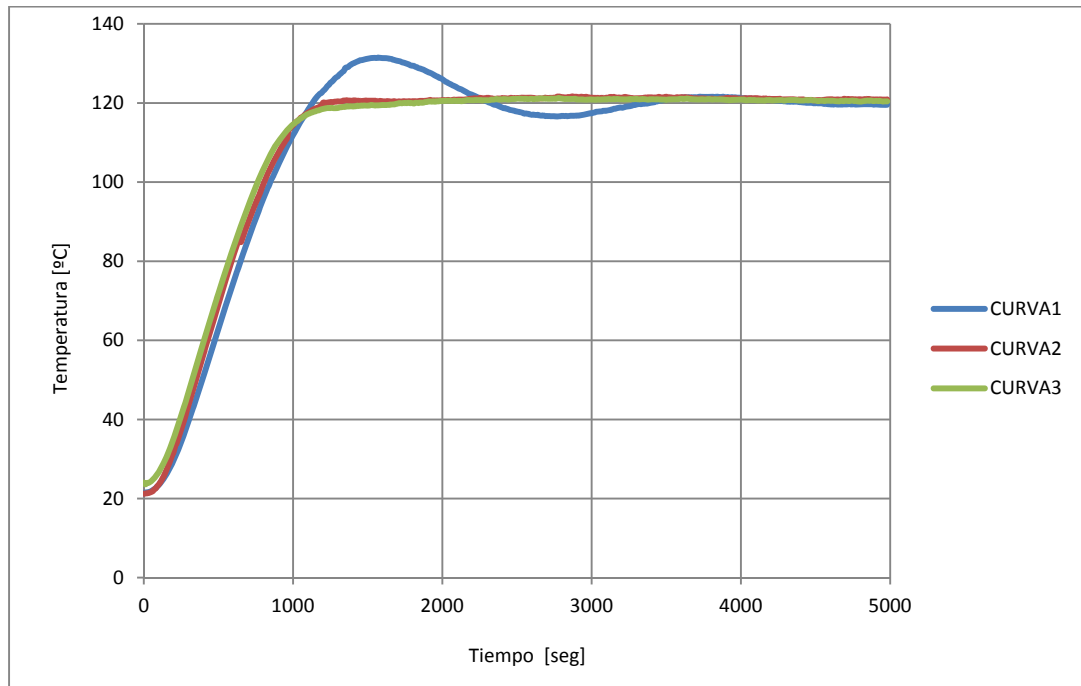


Figura 4.4. Gráficas de temperatura contra tiempo para una referencia de 120°C.

3.2 Discusión

El sistema electrónico desarrollado constituye una herramienta útil para el control de temperatura en un horno semi-cerrado. No obstante, éste puede ser utilizado en diferentes hornos y su ajuste preciso para estas diferentes condiciones de operación se realiza mediante el ajuste de las constantes de proporcionalidad en el algoritmo de control. Lo anterior constituye un problema ya que el ajuste no es un proceso sencillo. Sin embargo, el ajuste se puede llevar a cabo heurísticamente o bien utilizando el método bien conocido de Ziegler-Nichols [Williams, 2013].

Los resultados obtenidos a partir del experimento para el control de temperatura nos indican la confiabilidad del sistema. El estado estacionario del sistema se consiguió en aproximadamente 2000s (media hora). En este sentido, un trabajo a futuro puede ir encaminado a reducir este tiempo de estabilización.

Conclusiones

Se desarrolló un sistema electrónico y software de operación para el control de temperatura de un horno semi-cerrado para caracterizar materiales en atmósfera controlada. El equipo desarrollado demostró su utilidad al realizar un experimento de control de temperatura y presentar resultados conforme a los objetivos planteados.

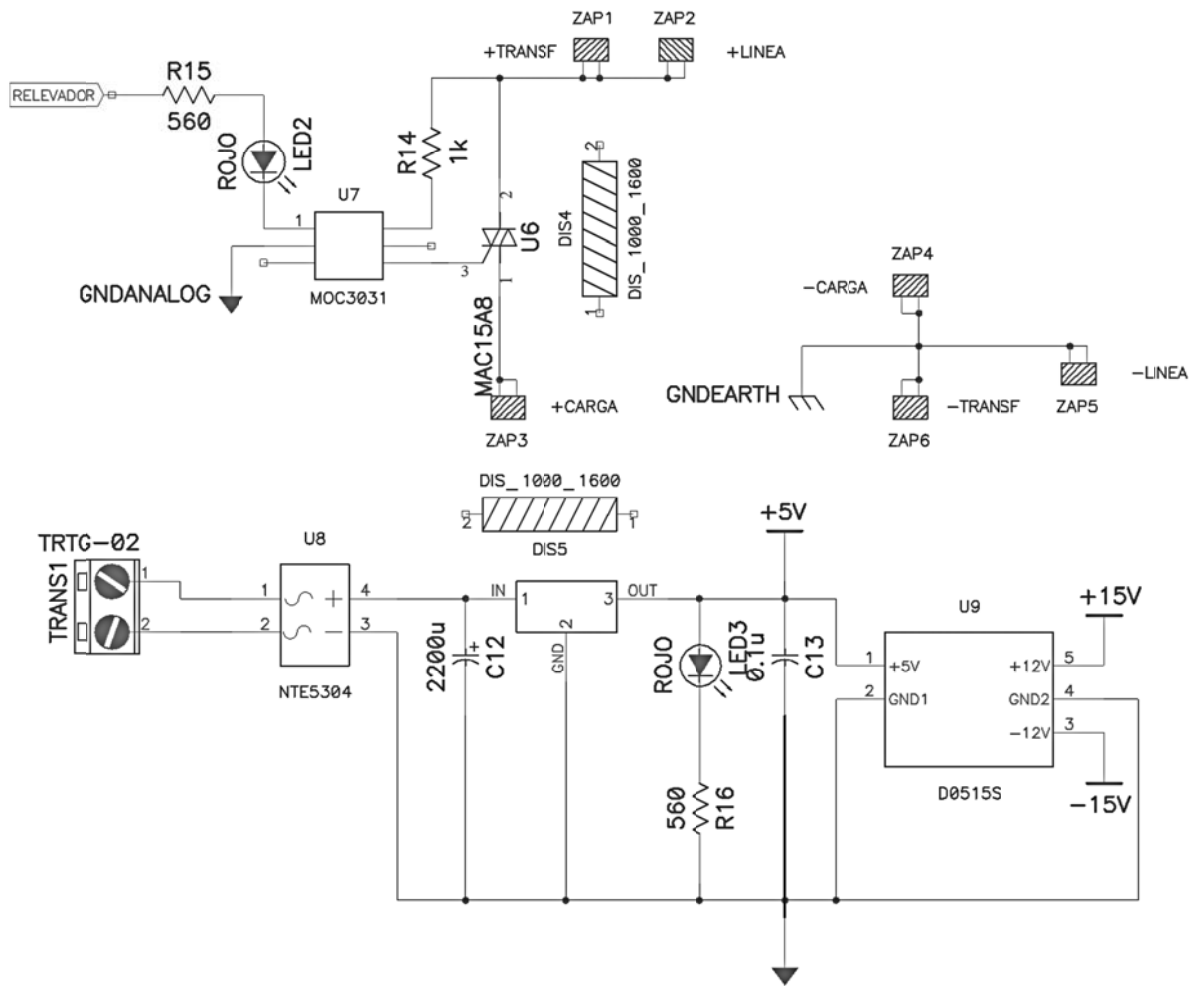
Por otro lado, el control desarrollado no limita su uso a un tipo de planta, en nuestro caso a un horno en particular ya que se desarrollaron diversos experimentos con diversos hornos como por ejemplo parrillas eléctricas expuestas al aire libre.

El sistema es de fácil manejo y permite su operación tanto de forma local a través de un teclado y despliegue alfanumérico o de forma remota mediante una PC con software específico

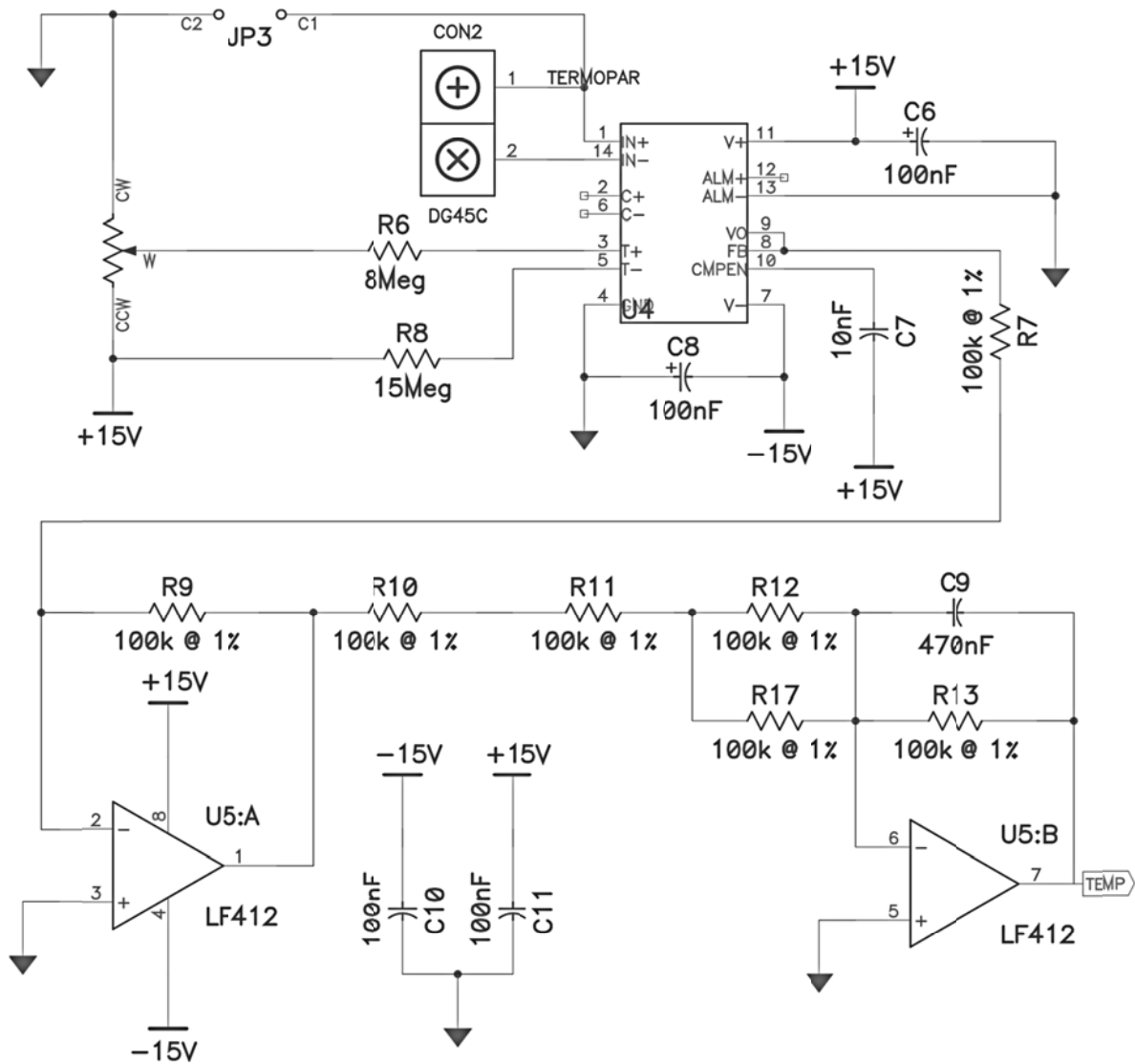
A.1 Etapa de control



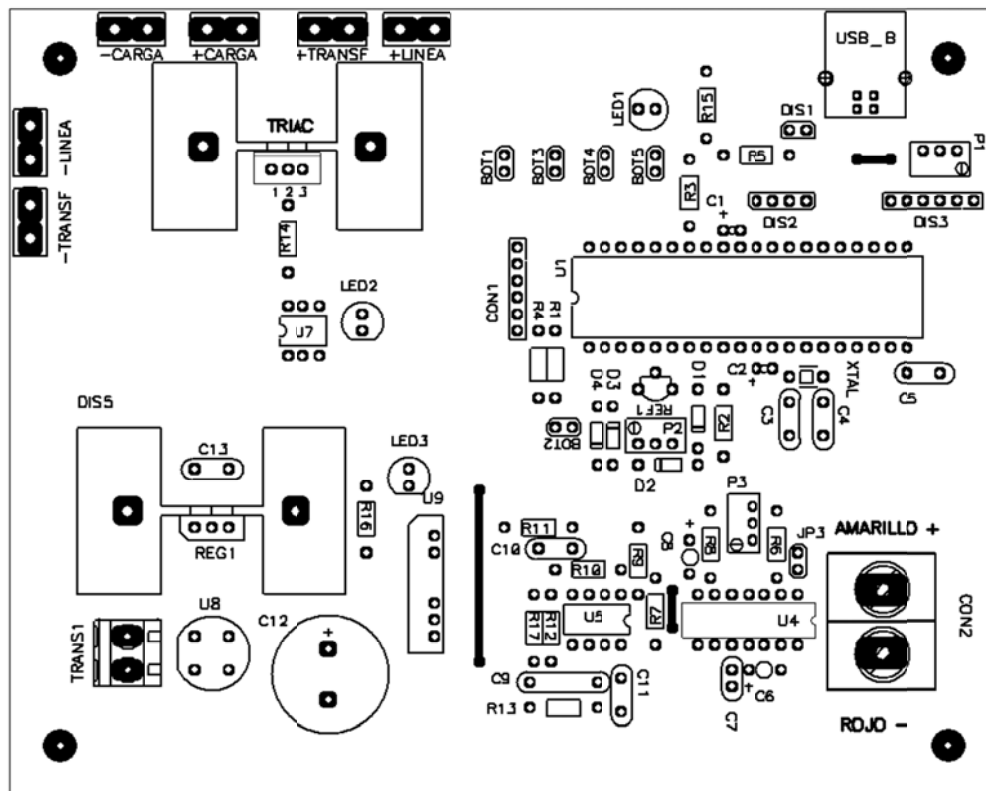
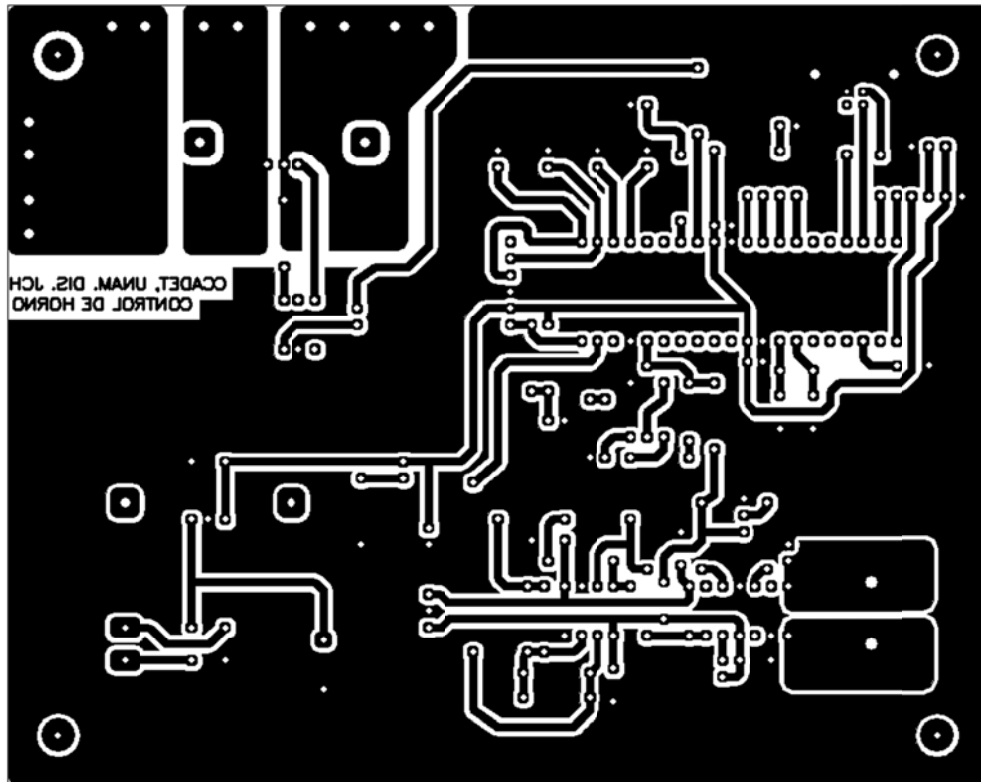
A.2 Impulsor CA y fuente de alimentación



A.3 Acondicionador de señal



Anexo B Circuito Impreso



Anexo C Lista de material

ID	Descripción
D1	Diodo de alta velocidad 1N4448
D2	Diodo de alta velocidad 1N4448
D3	Diodo de alta velocidad 1N4448
D4	Diodo de alta velocidad 1N4448
C1	Capacitor cerámico 100nF
C2	Capacitor cerámico 100nF
C3	Capacitor cerámico 22pF
C4	Capacitor cerámico 22pF
C5	Capacitor cerámico 470nF
C6	Capacitor cerámico 100nF
C7	Capacitor cerámico 10nF
C8	Capacitor cerámico 100nF
C9	Capacitor cerámico 470nF
C10	Capacitor cerámico 100nF
C11	Capacitor cerámico 100nF
C12	Capacitor electrolítico 2200uF
C13	Capacitor cerámico 0.1u
CON2	Conector DG45C
DIS4	Conector DIS_1000_1600
DIS5	Conector DIS_1000_1600
DIS1	Conector EIS2
DIS2	Conector EIS4
CON1	Conector EIS6
DIS3	Conector EIS6
BOT1	Conector y push botton
BOT2	Conector y push botton
BOT3	Conector y push botton
BOT4	Conector y push botton
BOT5	Conector y push botton
JP3	Conector jumper
LED1	Led rojo
LED2	Led rojo
LED3	Led rojo
REG1	Circuito integrado LM7805
REF1	Circuito integrado LM336 5V
U1	Circuito integrado PIC16F4553
U2	Cristal de cuarzo
U3	Conector USB_B
U4	Circuito integrado AD595AD
U5	Circuito integrado LF412
U6	Circuito integrado MAC15A8
U7	Circuito integrado MOC3031
U8	Circuito integrado NTE5304
U9	Circuito integrado D0515S
R1	Resistencia de carbón 10k
R2	Resistencia de carbón 3.3k
R3	Resistencia de carbón 560
R4	Resistencia de carbón 470
R5	Resistencia de carbón 33
R6	Resistencia de carbón 8Meg
R7	Resistencia de precisión 100k @ 1%

R8	Resistencia de carbón 15Meg
R9	Resistencia de precisión 100k @ 1%
R10	Resistencia de precisión 100k @ 1%
R11	Resistencia de precisión 100k @ 1%
R12	Resistencia de precisión 100k @ 1%
R13	Resistencia de precisión 100k @ 1%
R14	Resistencia de carbón 1k
R15	Resistencia de carbón 560
R16	Resistencia de carbón 560
R17	Resistencia de precisión 100k @ 1%
P1	Resistencia variable tipo trimpot 1k
P2	Resistencia variable tipo trimpot 10k
P3	Resistencia variable tipo trimpot 100k
TRANS1	Conector TRTG-02
ZAP1	Zócalo para medición
ZAP2	Zócalo para medición
ZAP3	Zócalo para medición
ZAP4	Zócalo para medición
ZAP5	Zócalo para medición
ZAP6	Zócalo para medición
Otro	Despliegue AND491
Otro	Caja para proyecto
Otro	Termopar K

Referencias

- [1]. Analog Devices. (1999) *Monolithic Thermocouple Amplifiers with Cold Junction Compensation*. Analog Devices USA.
- [2]. Ang KH, Chong G, Li Y. (2005) PID Control Systems Analysis Design and Technology. *IEEE Trans. Control Systems Technology*, vol 13, no 4, pp. 559-576.
- [3]. Chakir E, Alaoui El, Ayad H, Doubabi S. (2006) Fuzzy Anti-Windup Schemes for PID Controllers. *International Journal of Applied Engineering Research*. vol 1, no 3, pp. 295-306.
- [4]. Custom Computer Services. (2007) *PCD C Compiler Reference Manual*. Custom Computer Services USA.
- [5]. Fairchild Semiconductor. (2010) *6-Pin DIP Zero-Cross Optoisolators Triac Driver Output*. Fairchild USA.
- [6]. JVR. (2013) "PIC Calculadora USB" en *Mucho trasto*. [On line]. Available at: <http://www.muchotrasto.com/PICCalculadoraUSBC++.php>. [Accessed on Sept 12, 2013].
- [7]. Kruglinski DJ, Shepherd G, Wingo S. (1998) *Programming Microsoft Visual C++*. Microsoft Press USA.
- [8]. Microchip. (2007) *PIC18F2458/2553/4458/4553 Data Sheet*. Microchip USA.
- [9]. Montiel MH, Mata Z, Valera B, Lopez R. (2010) *Sistema de medición del campo crítico de propagación y temperatura de Curie en materiales ferromagnéticos: Informe Técnico*. CCADET UNAM.
- [10]. Omega. (2013) "Temperature products" en *Omega*. [On line]. Available at: <http://www.omega.com/temperature/tsc.html> [Accessed on Sept 12, 2013].
- [11]. On Semiconductor. (2008) *MAC 15 Series*. Semiconductor Components Industries USA.
- [12]. Texas Instruments. (2005) *LM136-2.5/LM236-2.5/LM336-2.5V Reference Diode*. Texas Instruments USA.
- [13]. Williams CDH. (2013) "Tuning a PID Temperature Controller" en *University of Exeter Physics and Astronomy*. [On line]. Available at: <http://newton.ex.ac.uk/teaching/CDHW/Feedback/Setup-PID.html> [Accessed on Sept 12, 2013].