

“MANUAL DE PRÁCTICAS DE CONTROL AVANZADO UTILIZANDO LA TARJETA DE ADQUISICIÓN DE DATOS NI USB 6008”.

Autor: Isaac Pérez Moreno

Asesor: MC Francisco Ronay López Estrada



Período Agosto 2009- Enero de 2010

Índice

1. INTRODUCCIÓN	5
1.1. INTRODUCCIÓN	5
1.2. OBJETIVOS	6
1.2.1. OBJETIVO GENERAL	6
1.2.2. OBJETIVOS ESPECÍFICOS	6
1.3. JUSTIFICACIÓN	6
2. FUNDAMENTOS TEÓRICOS	7
2.1. Modelos con Variables de Estado	7
2.1.1. Modelo de Estados	7
2.1.2. Función de transferencia de un modelo de estados	8
2.1.3. Modelado de un filtro pasabajas RC	8
2.1.4. Modelado de un sistema mecánico translacional	9
2.1.5. Modelado de un circuito con OPAM's	11
2.2. Respuesta en el Tiempo	12
2.2.1. Sistemas de Primer Orden	12
2.2.2. Sistemas de segundo orden	13
2.3. Respuesta en frecuencia	15
2.3.1. Diagramas de Bode	15
2.4. Acciones de control	17
2.4.1. Control ON-OFF	17
2.4.2. Controladores PID	19
2.5. Métodos de sintonización de Ziegler and Nichols	25
2.5.1. Método de lazo abierto	25
2.5.2. Método de lazo cerrado	26
3. USO DE LA DAQ NI USB 6008	28
3.1. Introducción	28
3.1.1. Adquisición de Datos	28
3.1.2. ¿Qué es la DAQ NI USB 6008?	28
3.1.3. El entorno Labview	30
3.2. Adquisición de Entradas Analógicas y registro de Datos	31
3.2.1. Adquisición y registro de datos de una señal analógica	31
3.2.2. Adquirir entradas analógicas simultáneamente	38
3.2.3. Adquisición de entradas analógicas y generación de reportes en Excel.	38
3.3. Generación de Señales Analógicas	40
3.3.1. Generación de un nivel de voltaje	40
3.4. Adquisición y Generación de E/S Digitales.	43
3.5. Adquisición de Entradas Analógicas y Generación de Salidas Digitales	49
4. PRÁCTICAS PROPUESTAS	50
4.1. PRÁCTICA I: “RESPUESTA TEMPORAL”	50
4.1.1. INTRODUCCIÓN	50
4.1.2. OBJETIVO DE LA PRÁCTICA	50
4.1.3. MATERIALES Y EQUIPO	50
4.1.4. DESARROLLO DE LA PRÁCTICA	50

4.2.	PRÁCTICA II: “RESPUESTA EN FRECUENCIA”	58
4.2.1.	INTRODUCCIÓN	58
4.2.2.	OBJETIVOS	58
4.2.3.	MATERIALES Y EQUIPO	58
4.2.4.	DESARROLLO DE LA PRÁCTICA	58
4.3.	PRÁCTICA III: “CONTROL ON-OFF”	63
4.3.1.	INTRODUCCION	63
4.3.2.	OBJETIVOS	63
4.3.3.	MATERIALES Y EQUIPO	64
4.3.4.	DESARROLLO DE LA PRÁCTICA	64
4.4.	PRÁCTICA IV: “PWM UTILIZANDO LA DAQ NI USB 6008”	74
4.4.1.	INTRODUCCIÓN	74
4.4.2.	OBJETIVO	74
4.4.3.	EQUIPO	74
4.4.4.	DESARROLLO DE LA PRÁCTICA	74
4.5.	PRÁCTICA V. “CARACTERIZACIÓN DE UN SISTEMA TÉRMICO POR EL MÉTODO DE LA CURVA DE REACCIÓN”.	80
4.5.1.	INTRODUCCIÓN	80
4.5.2.	OBJETIVO DE LA PRÁCTICA	80
4.5.3.	MATERIALES Y EQUIPO	80
4.5.4.	DESARROLLO DE LA PRÁCTICA	80
4.6.	PRÁCTICA VI: “CONTROL PROPORCIONAL (P)”	83
4.6.1.	INTRODUCCIÓN	83
4.6.2.	OBJETIVOS	83
4.6.3.	MATERIALES Y EQUIPO	84
4.6.4.	DESARROLLO DE LA PRÁCTICA	84
4.7.	PRÁCTICA VII: “CONTROL PROPORCIONAL-INTEGRAL”	89
4.7.1.	INTRODUCCIÓN	89
4.7.2.	OBJETIVOS	90
4.7.3.	MATERIALES Y EQUIPO	90
4.7.4.	DESARROLLO DE LA PRÁCTICA	90
4.8.	PRÁCTICA VIII: “CONTROL PROPORCIONAL-DERIVATIVO”	95
4.8.1.	INTRODUCCIÓN	95
4.8.2.	OBJETIVOS	95
4.8.3.	MATERIALES Y EQUIPO	95
4.8.4.	DESARROLLO DE LA PRÁCTICA	96
4.9.	PRÁCTICA IX: “CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO”	99
4.9.1.	INTRODUCCIÓN	99
4.9.2.	OBJETIVOS	99
4.9.3.	MATERIALES Y EQUIPO	99
4.9.4.	DESARROLLO DE LA PRÁCTICA	99
5.	CONCLUSIONES	103
6.	ANEXOS	104
6.1.	ANEXO I: HOJAS DE ESPECIFICACIONES DE LOS COMPONENTES	104
6.2.	ANÉXO II: OTROS MODELOS DE DAQ’s	108
6.2.1.	NI USB-6009	108

6.2.2.	NI USB-6210	108
6.2.3.	NI USB-6211	109
6.2.4.	NI PCI-6024E	109
6.3.	ANEXO III: PROGRAMA PRUEBA DE PID	110
6.4.	CONTENIDO DEL INFORME DE PRÁCTICA	111

1. INTRODUCCIÓN

1.1. INTRODUCCIÓN

El “Manual de Prácticas de Control Avanzado utilizando la Tarjeta de Adquisición de Datos NI USB 6008” muestra un enfoque práctico de la utilización de Labview y la adquisición de datos para sistemas de monitoreo y acciones de control. Además, introduce al estudiante a la programación en lenguaje gráfico en caso de que éste desconozca el tema.

La primera parte del manual contiene los conceptos de control tratados en las prácticas a realizar. Se comienza tratando temas ya conocidos como variables de estado, respuesta en el tiempo, respuesta en frecuencia y se llega al momento en que se tratan los temas de acciones básicas de control empleando partiendo de conceptos analógicos para facilidad de comprensión, y así posteriormente realizar la metodología de discretización e implementación en el entorno Labview.

La segunda parte del manual es una guía para el uso de la tarjeta de adquisición de datos por medio de la programación gráfica que ofrece Labview. Es necesario realizar los ejercicios allí descritos ya que son la base para las aplicaciones posteriores planteadas en este manual, dado que en ellas se muestra la forma de adquirir y generar datos con Labview por medio de la DAQ NI USB 6008 de una forma clara y sencilla.

Finalmente, la tercera parte propone prácticas de control a realizarse aplicando los conceptos descritos en la primera sección y entendiendo que el estudiante ha realizado los ejercicios básicos del uso de la tarjeta de adquisición de datos en cuestión. Se comienza con prácticas que permitan al usuario de éste manual seguir familiarizándose con el uso de la DAQ NI USB 6008 y el entorno Labview; posteriormente se plantean prácticas en donde se llevan a cabo aplicaciones de control de lazo cerrado utilizando una variable física real, que van desde un control ON-OFF simple, hasta un control PID y su relación respectiva a un PWM. Básicamente, a lo largo de éste manual proponen prácticas de control de temperatura para un sistema térmico compuesto de una caldera y una parrilla eléctrica. Ésto es con la finalidad de controlar un sistema físico real, de forma similar a la realización de un proceso industrial.

Estado del Arte. En la red existen numerosas prácticas de control PID discreto realizadas con microcontroladores, principalmente para control de velocidad y/o posición. También existen documentos de prácticas planteadas de control PID analógico para sistemas de temperatura, velocidad y posición. Por ejemplo, en la memoria SOMI XV preparado por el Centro de Instrumentos UNAM se propone el desarrollo de un control PID didáctico en base a OPAM's para un sistema compuesto por una parrilla eléctrica y una caldera. Otro caso similar una nota técnica preparada por el Grupo desarrollo Arian, S.A. en donde se plantean diversos conceptos de control PID para el control de temperatura de un horno industrial. Se han escrito también numerosos libros y una infinidad de tutoriales sobre el uso de Labview e incluso existen en menor cantidad documentos en la red sobre la construcción de algoritmos que simulan un control PID.

La National Instruments ha creado foros de ayuda y discusión para aquellos que utilizan sus productos, tales como tarjetas de adquisición de datos, toolkits de Labview, etc. y se enfocan principalmente en proponer respuestas a las dudas de los usuarios respecto al uso de los productos NI. Sin embargo, hay ocasiones en las que se presenta el inconveniente de no tener las suficientes bases o experiencia para comprender alguna de las soluciones que se brindan.

A pesar de la utilidad de la gran cantidad de información existente, en la gran mayoría de los casos los temas de control, adquisición de datos y uso de Labview son abordados de forma aislada. El manual que proponemos aborda los temas no de forma aislada, sino como elementos interrelacionados que constituyen una herramienta de aprendizaje significativa.

De ninguna manera un manual sustituye a un libro de texto de teoría de control o de Labview, dado que en un libro de texto se trata con mayor profundidad cada uno de los temas. Por el contrario con este manual se pretende despertar curiosidad del usuario para profundizar en los temas y consultar diferentes bibliografías que sustenten las prácticas aquí planteadas.

1.2. OBJETIVOS

1.2.1. OBJETIVO GENERAL

Realizar un Manual de Prácticas de Control Avanzado con la tarjeta de adquisición de datos (DAQ) NI USB 6008 utilizando el software Labview para la adquisición de Datos y el software Matlab para las respectivas simulaciones de *c/u* de las prácticas.

1.2.2. OBJETIVOS ESPECÍFICOS

Llevar a cabo la realización de las siguientes prácticas:

- 1) Respuesta Temporal.
- 2) Respuesta en Frecuencia.
- 3) Control ON-OFF.
- 4) Control Proporcional (P).
- 5) Control Proporcional-Integral (PI).
- 6) Control Proporcional-Derivativo (PD).
- 7) Control PID.

1.3. JUSTIFICACIÓN

El control de procesos ha tenido una importante evolución en la actualidad y el área correspondiente a la ingeniería electrónica ha sido ampliamente beneficiada. Ello requiere que los estudiantes de ingeniería electrónica estén lo suficientemente preparados para enfrentar los cambios y desafíos del mundo moderno. Actualmente la adquisición de datos es una poderosa herramienta para el monitoreo y control de procesos y es necesario brindar al estudiante la formación necesaria para dominar ésta herramienta. El presente manual de prácticas es una opción para acercar a los estudiantes al uso de tarjetas de adquisición de datos, aplicando esta herramienta a las diversas estrategias de control que se han aprendido en los diferentes cursos de la materia.

Es común que los cursos de control se enfoquen mayormente a aspectos teóricos y de contenido matemático fuerte, razón por la cual el aspecto práctico tiende a descuidarse. Es por ello que se plantean prácticas partiendo de los conceptos de control analógico y discreto ayudándonos de la adquisición de datos en Labview por medio de la DAQ NI USB 6008, en las cuales el estudiante aplicará los conceptos aprendidos de la materia a un proceso físico real desarrollando una interfaz en el lenguaje gráfico de Labview.

Además, el modo de programación gráfica en Labview para dispositivos de la National Instruments, es genérico, por lo cual, si alguien desea programar otra tarjeta de adquisición de datos la familia de productos de la National Instruments para alguna otra aplicación, estará capacitado para hacerlo con el hecho de haber aprendido a programar la DAQ NI USB 6008, la única diferencia radica en las características específicas de la tarjeta que utilice (vease anexo II).

2. FUNDAMENTOS TEÓRICOS

2.1. Modelos con Variables de Estado

2.1.1. Modelo de Estados

Cuando se tiene un sistema MIMO (Multiple input- Multiple output), tenemos un vector u de entrada, un vector y de salida, y trabajamos con variables de estado x en el dominio del tiempo:

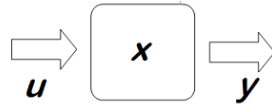


Figura 1: MODELO DE ESTADOS

Representación matemática del modelo de estados:

$$\frac{dx}{dt} = Ax + Bu \dots \text{Ecuación de Estado}$$

$$y = Cx + Du \dots \text{Ecuación de Salida}$$

Las matrices A, B, C, D que forman el modelo de estados del sistema, de donde:

- La matriz A ó matriz dinámica del sistema y está relacionada a la estabilidad del sistema.
- La matriz B llamada matriz de mando y está relacionada a la controlabilidad.
- La matriz C conocida como matriz de observación corresponde a la observabilidad.
- La matriz D se conoce como matriz de acoplamiento directo.

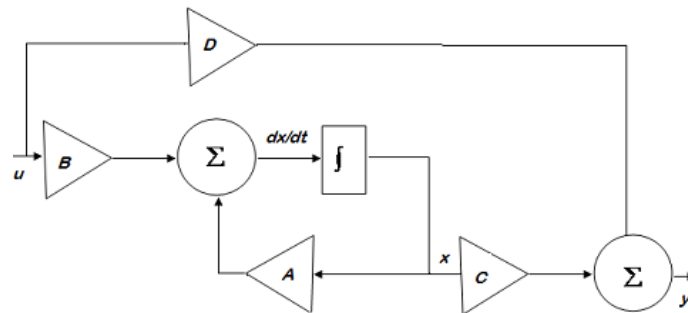


Figura 2: DIAGRAMA A BLOQUES DE UN SISTEMA QUE UTILIZA MODELO DE ESTADOS

Las variables de estado tienen que ser linealmente independientes, ya que son el mínimo de variables necesarias para describir el comportamiento de un sistema. Asimismo, el orden del sistema está dado por el número de variables de estado que éste tenga. A su vez, por cada elemento que almacena energía tendremos una variable de estado. Se recomienda que al elegir una variable de estado, cuando se trate de un circuito eléctrico se elijan voltajes en los capacitores y corrientes en las bobinas; para sistemas mecánicos, se recomiendan las posiciones y velocidades.

2.1.2. Función de transferencia de un modelo de estados

Dadas las ecuaciones de un modelo de estados

$$\frac{dx}{dt} = Ax + Bu \dots\dots \text{Ecuación de Estado}$$

$$y = Cx + Du \dots\dots \text{Ecuación de Salida}$$

se toma la transformada de la Laplace, suponiendo condiciones iniciales en cero, las expresiones quedan de la siguiente forma:

$$sX(s) = AX(s) + BU(s) \dots\dots (1,1)$$

$$Y(s) = CX(s) + DU(s) \dots\dots (1,2)$$

manipulando para despejar $X(s)$

$$(sI - A)X(s) = BU(s) \dots\dots (1,3)$$

obtenemos

$$X(s) = (sI - A)^{-1}BU(s) \dots\dots (1,4)$$

donde I es la matriz identidad.

Sustituyendo la ecuación (1.4) en la (1.2) resulta:

$$Y(s) = [C(sI - A)^{-1}B + D]U(s) \dots\dots (1,5)$$

$$\frac{Y(s)}{U(s)} = [C(sI - A)^{-1}B + D] \dots\dots (1,6)$$

La matriz $[C(sI - A)^{-1}B + D]$ se llama matriz de funciones de transferencia, puesto que relaciona el vector salida $Y(s)$ con el vector de entrada $U(s)$.

2.1.3. Modelado de un filtro pasabajas RC

Consideremos el siguiente circuito pasabajas de primer orden:

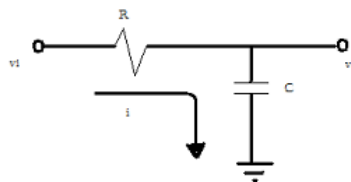


Figura 3: Circuito RC

Por leyes de Kirchooff sabemos que:

$$v_i - v_R - v_o = 0 \dots \dots (1,7)$$

Tenemos que la corriente en el circuito está dada por:

$$i = C \frac{d}{dt} v_c = C \frac{d}{dt} v_o \dots \dots (1,8)$$

pero tambien:

$$i = \frac{v_o - v_i}{R} \dots \dots (1,9)$$

Ahora bien, para este caso tomaremos a $v_i = u$ (entrada), $v_o = x$ (estado), $i = y$ (salida). Para ello usaremos la ecuacion 1.7 para encontrar la ecuación de estado:

$$v_i - v_R - v_o = v_i - Ri - v_o = v_i - RC \frac{d}{dt} v_o - v_o = 0 \dots \dots (1,10)$$

la ecuación que resulta es la siguiente:

$$\frac{d}{dt} v_o = \frac{1}{RC} v_i - \frac{1}{RC} v_o \dots \dots (1,11)$$

La ecuación 1.9 corresponde a la ecuación de salida, ya que la salida de nuestro sistema es la corriente i . Con lo cual nuestro modelo de estados será el siguiente:

$$\left[\frac{d}{dt} v_o \right] = \left[-\frac{1}{RC} \right] [v_o] + \left[\frac{1}{RC} \right] [v_i] \dots \dots (1,12)$$

$$[i] = \left[-\frac{1}{R} \right] [v_o] + \left[\frac{1}{R} \right] [v_i] \dots \dots (1,13)$$

sustituyendo las matrices A, B, C, D en $[C(sI - A)^{-1}B + D]$ y manipulando la expresión, tenemos que la función de transferencia es:

$$\frac{I(s)}{Vi(s)} = \frac{Cs}{RCs + 1} \dots \dots (1,14)$$

2.1.4. Modelado de un sistema mecánico traslacional

Consideremos el siguiente sistema:

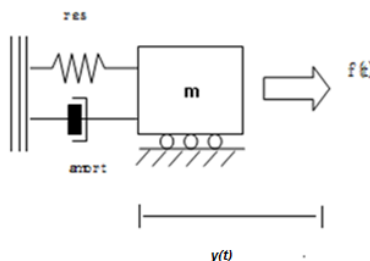


Figura 4: Sistema Mecánico Traslacional

En la figura anterior observamos un objeto de masa m que se desplaza una distancia $y(t)$ debido a una fuerza $f(t)$, dicha masa está sujeta a un resorte y a un amortiguador viscoso. Sabemos que las ecuaciones del resorte y el amortiguador viscoso son las siguientes:

$$f_{res} = ky \dots \dots \dots (1,15)$$

$$f_{amort} = b \frac{dy}{dt} \dots \dots \dots (1,16)$$

Donde k es la constante del resorte y b es la constante del amortiguador viscoso. Ahora utilizando la segunda ley de Newton:

$$\Sigma F = ma = m \frac{d^2 y(t)}{dt^2} \dots \dots \dots (1,17)$$

De donde sabemos que la aceleración es la derivada de la velocidad $\frac{dy(t)}{dt}$. La fuerza $f(t)$ tiene oposición por medio de la f_{res} y f_{amort} , entonces podemos arreglar la ecuación (3) de modo que obtenemos la siguiente expresión:

$$\Sigma F = ma = m \frac{d^2 y(t)}{dt^2} = f(t) - f_{res} - f_{amort} \dots \dots (1,18)$$

$$m \frac{d^2 y}{dt^2} = f(t) - ky(t) - b \frac{d}{dt} y(t) \dots \dots (1,19)$$

Tomaremos como entrada a la fuerza $f(t)$, como salida al desplazamiento $y(t)$, y los estados serán $x_1 = y(t)$, $x_2 = \frac{dy(t)}{dt}$. Teniendo en cuenta lo anterior podemos formar el modelo de estados:

Las ecuaciones de estado son

$$\frac{d}{dt} x_1 = x_2 \dots \dots \dots (1,20)$$

$$\frac{d}{dt} x_2 = \frac{1}{m} f(t) - \frac{k}{m} y(t) - \frac{b}{m} \frac{d}{dt} y(t) \dots \dots (1,21)$$

La ecuación de salida es

$$y(t) = x_1 \dots \dots (1,22)$$

Dado lo anterior, expresamos en forma matricial el modelo de estados

$$\begin{bmatrix} \frac{d}{dt} x_1 \\ \frac{d}{dt} x_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} [f(t)] \dots \dots (1,23)$$

$$[y(t)] = [1 \quad 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [0] [f(t)] \dots \dots (1,24)$$

Ahora, si deseamos hallar la función de transferencia del modelo de estados aplicamos $[C(sI - A)^{-1}B + D]$. Sustituyendo las matrices A, B, C, D en dicha expresión

$$\frac{Y(s)}{F(s)} = \left[\begin{bmatrix} 1 & 0 \end{bmatrix} \left[s \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{b}{m} \end{bmatrix} \right]^{-1} \right] \dots\dots\dots(1,25)$$

Realizando las operaciones con matrices obtenemos

$$\frac{Y(s)}{F(s)} = \frac{1}{ms^2 + bs + k} \dots\dots\dots(1,26)$$

2.1.5. Modelado de un circuito con OPAM's

Consideremos el siguiente circuito y consideremos con entradas a vi ; salida a vo ; estados a los voltajes $vc1$, $vc2$.

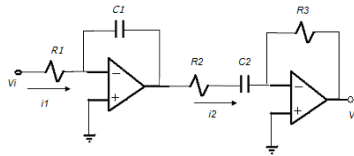


Figura 5: Circuito con OPAM's

Tenemos que la corrientes son $i1$ e $i2$:

$$i1 = \frac{vi}{R1} = c1 \frac{d}{dt} vc1 \dots\dots(1,27)$$

$$i2 = -\frac{vc1 + vc2}{R2} = c2 \frac{d}{dt} vc2 \dots\dots(1,28)$$

Manipulando las expresiones anteriores podemos encontrar las ecuaciones de estado

$$\frac{d}{dt} vc1 = \frac{1}{c1R1} vi \dots\dots(1,29)$$

$$\frac{d}{dt} vc2 = -\frac{1}{R2C2} vc1 - \frac{1}{R2C2} vc2 \dots\dots(1,30)$$

La ecuación de salida está dada por

$$vo = R3 \left[\frac{vc1 + vc2}{R2} \right] \dots\dots(1,31)$$

$$vo = \frac{R3}{R2} vc1 + \frac{R3}{R2} vc2 \dots\dots(1,32)$$

El modelo de estados en forma matricial es el siguiente

$$\begin{bmatrix} \frac{d}{dt} vc1 \\ \frac{d}{dt} vc2 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ -\frac{1}{R2C2} & -\frac{1}{R2C2} \end{bmatrix} \begin{bmatrix} vc1 \\ vc2 \end{bmatrix} + \begin{bmatrix} \frac{1}{R1C1} \\ 0 \end{bmatrix} [vi] \dots\dots\dots(1,23)$$

$$\begin{bmatrix} vo \end{bmatrix} = \begin{bmatrix} \frac{R3}{R2} & \frac{R3}{R2} \end{bmatrix} \begin{bmatrix} vc1 \\ vc2 \end{bmatrix} + \begin{bmatrix} 0 \end{bmatrix} [vi] \dots\dots\dots(1,24)$$

2.2. Respuesta en el Tiempo

Cuando obtenemos el modelo matemático de un sistema podemos emplear diversos métodos para el análisis del desempeño del sistema. Una de las formas más comunes es aplicando señales de entrada específicas y particulares para observar la respuesta del sistema, siendo una de ellas la escalón. Con estas señales de prueba, es posible realizar con facilidad análisis matemáticos y experimentales de sistemas de control, dado que las señales son funciones del tiempo relativamente simples.

La respuesta en el tiempo de un sistema de control consta de dos partes: la respuesta transitoria y la respuesta en estado estable. Por respuesta transitoria nos referimos a la que va del estado inicial al estacionario. Por respuesta en estado estable, nos referimos a la manera en la cual se comporta la salida del sistema conforme el tiempo t tiende infinito.

2.2.1. Sistemas de Primer Orden

La función de transferencia de un sistema de primer orden se puede generalizar como:

$$H(s) = \frac{Y(s)}{U(s)} = \frac{K}{\tau s + 1} \dots\dots(2,1)$$

En la siguiente figura se muestra el diagrama a bloques de un sistema de primer orden:

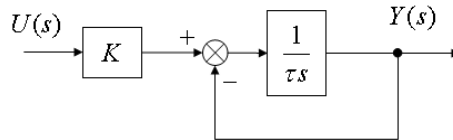


Figura 6: diagrama a bloques de un sistema de primer orden

Donde K se conoce como "ganancia de lazo abierto" y τ como "constante de tiempo". Si se despeja $Y(s)$ de la ecuación (1) se obtiene la siguiente expresión de la salida:

$$Y(s) = \frac{K}{\tau s + 1} U(s) \dots\dots(2,2)$$

Una entrada típica de prueba muy común es la función escalón de magnitud constante M ; es decir $u(t) = \Gamma$. Dado que la transformada de Laplace de $u(t)$ es $U(s) = \frac{\Gamma}{s}$; sustituyendo $U(s)$ en (2.2) se tiene:

$$Y(s) = \frac{K}{\tau s + 1} \left(\frac{\Gamma}{s} \right) = \frac{\frac{K\Gamma}{\tau}}{s(s + \frac{1}{\tau})} \dots\dots(2,3)$$

Por fracciones parciales, de (2.3) se tiene la solución $y(t)$ en términos de la función de entrada $u(t)$; es decir:

$$y(t) = \mathcal{L}^{-1} \left\{ \frac{K\Gamma}{s} - \frac{K\Gamma}{s + \frac{1}{\tau}} \right\} = K\Gamma(1 - e^{-\frac{t}{\tau}}) \dots\dots(2,4)$$

De la ecuación (2.4) se observa que cuando $t = \tau$ entonces $y(t) = K\Gamma(1 - e^{-1}) = K\Gamma(0.6321)$. Esto implica que la respuesta en el tiempo de $y(\tau) = K\Gamma(0.6321)$. Notese también que el valor final de $y(t) = K\Gamma$ sucede aproximadamente cuando $t = 5$; es decir: $y(5\tau) = K\Gamma(1 - e^{-5}) = (0.9932)K\Gamma \approx K\Gamma$.

En la siguiente figura se muestra la gráfica de un ejemplo de respuesta al escalón de un sistema de primer orden:

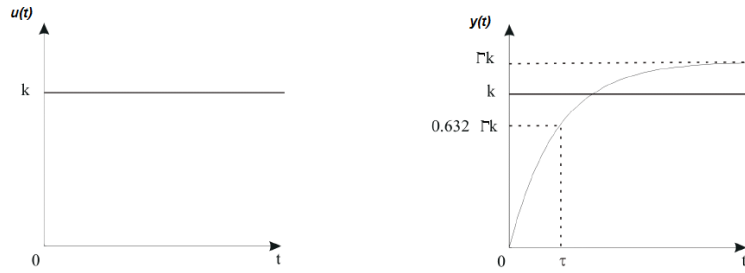


Figura 7: Respuesta de un sistema de primer orden

2.2.2. Sistemas de segundo orden

La función de transferencia de un sistema de segundo orden se puede generalizar de la siguiente manera

$$H(s) = \frac{Y(s)}{U(s)} = \frac{K\omega_N^2}{s^2 + 2\xi\omega_N s + \omega_N^2} \dots\dots(2.5)$$

donde K es la ganancia de lazo abierto, ω_N^2 es la “frecuencia natural no amortiguada” y ξ es el “coeficiente de amortiguamiento” del sistema. El diagrama a bloques de un sistema de segundo orden es el siguiente:

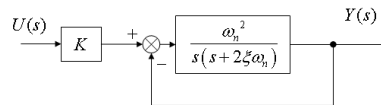


Figura 8: Diagrama a Bloques de un sistema de Segundo Orden

La respuesta en el tiempo se obtiene analizando la solución del sistema (2.6) a una entrada escalon Γ , dando como resultado la ecuación (2.7) para t mayor que 0.

$$H(s) = \frac{Y(s)}{U(s)} = \frac{K\omega_N^2}{s^2 + 2\xi\omega_N s + \omega_N^2} \left(\frac{\Gamma}{s} \right) \dots\dots(2.6)$$

$$y(t) = K\Gamma \left[1 - \frac{e^{-\xi\omega_n t}}{\sqrt{1-\xi^2}} \sin \left(\omega_n \left(\sqrt{1-\xi^2} \right) t + \arctan \frac{\sqrt{1-\xi^2}}{\xi} \right) \right] \dots\dots(2.7)$$

De lo anterior se concluye que el único tipo de respuesta útil para identificar un sistema de segundo orden es cuando el coeficiente de amortiguamiento es mayor que cero y menor que uno. Además, para cada caso del coeficiente de amortiguamiento, las respuestas del sistema tienen diferentes nombres:

- Respuesta no amortiguada (para $\xi = 0$)
- Respuesta subamortiguada (para $0 < \xi < 1$)
- Respuesta amortiguada o críticamente amortiguada (para $\xi = 1$)
- Respuesta sobreamortiguada (para $\xi > 1$).

El efecto de sigma lo podemos observar en la siguiente figura:

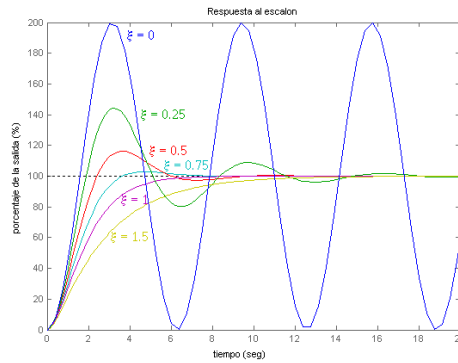


Figura 9: Variación del amortiguamiento

En la figura 10 se observan las características de la respuesta en el tiempo de un sistema de segundo orden.

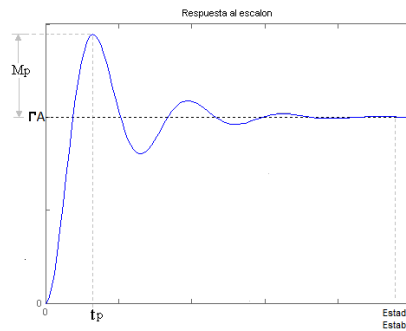


Figura 10: Respuesta al escalón de un Sistema de Segundo Orden

Donde:

- M_p se define como el sobrepulso máximo o también definido como porcentaje de sobrepulso máximo ($\%M_p$).

$$M_p = K\Gamma e^{-\frac{\sqrt{1-\xi^2}}{\xi}\pi} (100\%)$$

- y_p se define como el valor pico de la respuesta del sistema

$$y_p = K\Gamma + K\Gamma e^{-\frac{\sqrt{1-\xi^2}}{\xi}\pi}$$

- t_p se define como el tiempo en el que se alcanza el sobrepulso máximo

$$t_p = \frac{\pi}{\omega_N \sqrt{1-\xi^2}}$$

2.3. Respuesta en frecuencia

2.3.1. Diagramas de Bode

El diagrama de Bode consiste en la representación grafica de la función de transferencia mediante dos curvas, una relativa a la amplitud y la otra a la fase. Las gráficas de Bode Representan el $20\log G$ para que las gráficas aparezcan en dB en función de la frecuencia (o la frecuencia angular) en escala logarítmica. El diagrama de fase de Bode representa la fase de la función de transferencia en función de la frecuencia (o frecuencia angular) en escala logarítmica y se puede dar en grados o en radianes.

El siguiente es un procedimiento general para realizar gráficos de Bode a partir de una función de transferencia. Supongamos una función de transferencia de orden n:

$$H(s) = \frac{k(s + B_1)(s + B_2)...(s + B_m)}{(s + A_1)(s + A_2)...(s + A_n)} \dots\dots(3.1)$$

trabajando en el dominio de $j\omega$

$$H(j\omega) = \frac{k(j\omega + B_1)(j\omega + B_2)...(j\omega + B_m)}{(j\omega + A_1)(j\omega + A_2)...(j\omega + A_n)} \dots\dots(3.2)$$

ahora encontrando la magnitud de $|H(j\omega)|$ tenemos

$$|H(j\omega)| = \frac{k\sqrt{(\omega^2 + B_1^2)}\sqrt{(\omega^2 + B_2^2)}\dots\sqrt{(\omega^2 + B_m^2)}}{\sqrt{(\omega^2 + A_1^2)}\sqrt{(\omega^2 + A_2^2)}\dots\sqrt{(\omega^2 + A_n^2)}} \dots\dots(3.3)$$

para tener la magnitud en dB aplicamos propiedades de los logaritmos

$$M_{dB} = 20[\log k + \log\sqrt{(\omega^2 + B_1^2)} + \log\sqrt{(\omega^2 + B_2^2)} + \dots + \log\sqrt{(\omega^2 + B_m^2)} - \log\sqrt{(\omega^2 + A_1^2)} - \log\sqrt{(\omega^2 + A_2^2)} - \dots - \log\sqrt{(\omega^2 + A_n^2)}] \dots\dots(3.4)$$

volviendo más simple la expresión anterior resulta la expresión:

$$M_{dB} = 20\log(k) + 10[\log(\omega^2 + B_1^2) + \log(\omega^2 + B_2^2) + \dots + \log(\omega^2 + B_m^2) - \log(\omega^2 + A_1^2) - \log(\omega^2 + A_2^2) - \dots - \log(\omega^2 + A_n^2)] \dots\dots(3.5)$$

$$M_{dB} = 20\log(k) + 10 \left[\sum_{k=1}^m \log(\omega^2 + B_k^2) - \sum_{l=1}^n \log(\omega^2 + A_l^2) \right] \dots\dots(3.6)$$

para la fase la expresión es

$$\phi = \arctan(0) + \sum_{l=1}^n \arctan\left(\frac{\omega}{A_l}\right) + \sum_{k=1}^m \arctan\left(\frac{\omega}{B_k}\right) \dots\dots(3.7)$$

En la siguiente figura podemos observar un ejemplo de Diagrama de Bode:

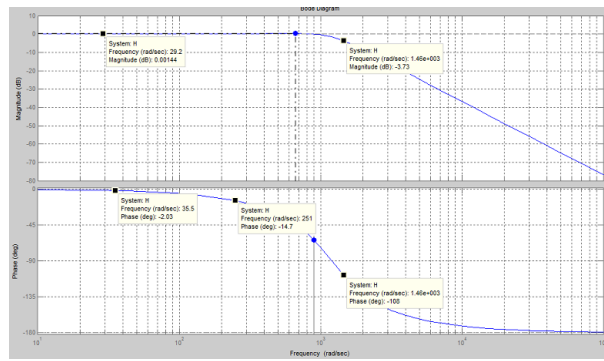


Figura 11: Ejemplo de diagrama de Bode

2.4. Acciones de control

2.4.1. Control ON-OFF

En la siguiente figura se muestra el diagrama a bloques de un control ON-OFF:

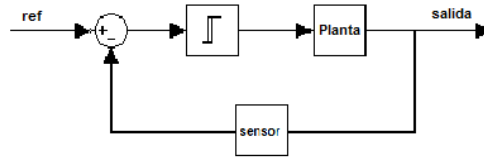


Figura 12: Diagrama a bloques de un control ON-OFF

2.4.1.1 Control ON-OFF de dos posiciones.

La salida del controlador ON-OFF, o de dos posiciones, solo puede cambiar entre dos valores al igual que dos estados de un interruptor. El controlador no tiene la capacidad para producir un valor exacto en la variable controlada para un valor de referencia dado pues el controlador produce una continua desviación del valor de referencia, como se observa en la siguiente figura:

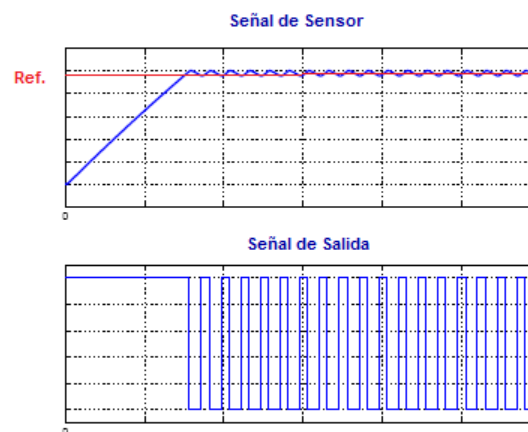


Figura 13: Respuesta del control ON-OFF de dos posiciones

Si deseamos una menor variación en la respuesta del sistema podemos introducir una zona de histéresis.

2.4.1.2 Control ON-OFF con histéresis.

La histéresis está definida como la diferencia entre los tiempos de apagado y encendido del controlador. En este tipo de acción de control vemos que la frecuencia de cambio en la señal de sensor decrece, dado a la zona de histéresis definida por las referencias inferior y superior. Esto provoca que el tamaño de las oscilaciones de la salida aumente, por tanto, tenemos un tiempo de respuesta menor, lo cual repercute en el valor de cambio de la entrada (señal de sensor en este caso) justo como se observa en la siguiente figura:

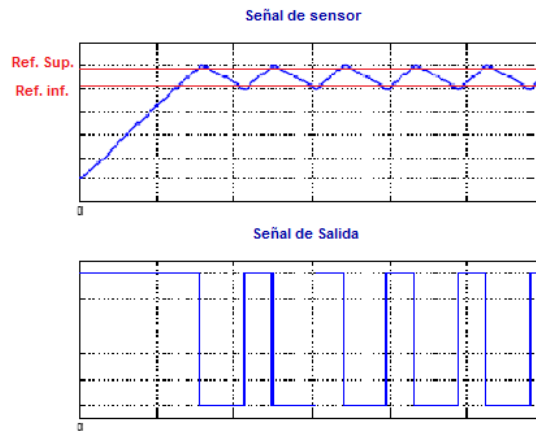


Figura 14: Respuesta de un control ON -OFF con histéresis

Cabe mencionar que entre mayor sea el intervalo de la histéresis este comportamiento será más acentuado, por lo cual, es importante que la zona de histéresis sea cuidadosamente elegido para no tener un sistema demasiado lento en su respuesta.

2.4.2. Controladores PID

La estructura básica de un controlador PID es la siguiente:

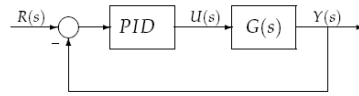


Figura 15: Estructura básica de un controlador PID

donde $G(s)$ es la función de transferencia de la planta a controlar. Los miembros de la familia de controladores PID, incluyen tres acciones: proporcional (P), integral (I) y derivativa (D). Estos controladores son los denominados P, I, PI, PD y PID.

2.4.2.1 Control Proporcional (P)

La acción de control proporcional, da una salida del controlador que es proporcional al error, es decir:

$$u(t) = K_p e(t) \dots (4.1)$$

que descrita desde su función transferencia queda:

$$C_P(s) = K_p \dots (4.2)$$

donde K_p es una ganancia proporcional ajustable. Un controlador proporcional puede controlar cualquier planta estable, pero posee desempeño limitado y error en régimen permanente.

El control proporcional tiene los siguientes efectos sobre el sistema:

- Reduce el tiempo de subida.
- Aumenta el sobreimpulso máximo.
- Disminuye el tiempo de establecimiento.

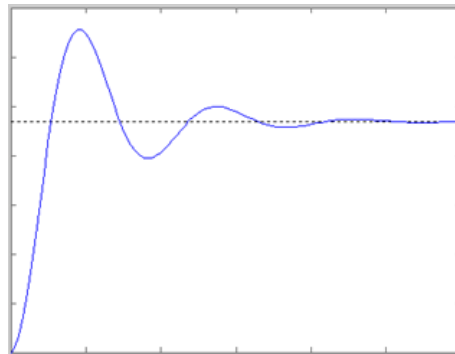


Figura 16: Efectos del control proporcional

Discretización del control proporcional. Para poder diseñar un control digital tenemos que tomar en cuenta la aplicación de la transformada z. En el caso de un control proporcional sabemos que la función de transferencia en el dominio de Laplace es:

$$\frac{U(s)}{E(s)} = K_p \dots (4.3)$$

ahora bien, si aplicamos transformada Z tenemos que el error es

$$\frac{U(z)}{E(z)} = K_p \dots (4.4)$$

De lo anterior observamos que un control proporcional discreto es básicamente el mismo que un control digital en el tiempo continuo, sin embargo, se verá que el proceso de discretización es necesario al desarrollar un sistema de control digital dado que en muchas ocasiones el control proporcional suele combinarse con acciones integrales y/o derivativas.

2.4.2.2 Control Proporcional-Integral (PI)

La acción de control integral da una salida del controlador que es proporcional al error acumulado, lo que implica que es un modo de controlar lento.

La señal de control $u(t)$ tiene un valor diferente de cero cuando la señal de error $e(t)$ es cero. Por lo que se concluye que dada una referencia constante, o perturbaciones, el error en régimen permanente es cero.

$$u(t) = K_i \int_0^t e(\tau) d\tau \dots (4.5)$$

La acción integral tiene los siguientes efectos sobre el sistema:

- Incrementa el tiempo de subida.
- Reduce el sobreimpulso máximo.
- Disminuye el tiempo de establecimiento.
- Mejora el sobreamortiguamiento.

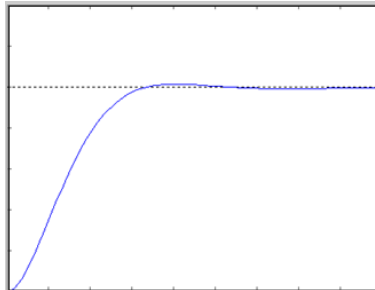


Figura 17: Efectos de la acción integral

La acción de control proporcional-integral, se define mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau \dots (4.6)$$

Donde T_i se denomina tiempo integral y es quien ajusta la acción integral. Dado que la ganancia proporcional K_i es:

$$K_i = \frac{K_p}{T_i} \dots (4.7)$$

Reescribimos la expresión (4.4) de la siguiente manera:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau \dots (4.8)$$

La función de transferencia resulta:

$$C_{PI}(s) = K_p \left(1 + \frac{1}{T_i s} \right) \dots\dots\dots(4.9)$$

Con un control proporcional, es necesario que exista error para tener una acción de control distinta de cero. Con acción integral, un error pequeño positivo siempre nos daría una acción de control creciente, y si fuera negativo la señal de control será decreciente. Este razonamiento sencillo nos muestra que el error en régimen permanente será siempre cero. Muchos controladores industriales tienen solo acción PI. Se puede demostrar que un control PI es adecuado para todos los procesos donde la dinámica es esencialmente de primer orden. Lo que puede demostrarse en forma sencilla, por ejemplo, mediante un ensayo al escalón.

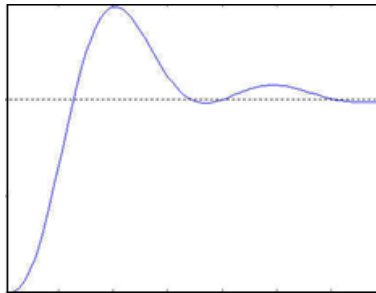


Figura 18: Respuesta de un control PI

Discretización de un control proporcional-integral.

Los sistemas de control no pueden ser diferenciados o integrados de manera exacta, sin embargo, existen métodos de aproximación para realizar las operaciones integrales y diferenciales. Estas estrategias consisten en utilizar los conceptos de aproximación de diversos métodos numéricos trasladándolos al dominio de z.

Método de integración simple. Sabemos que s puede ser descrita en el dominio de z como:

$$s = \frac{z - 1}{Tz} \dots\dots\dots(4.10)$$

Si sustituimos en la ecuación anterior en la función de transferencia de un integrador en el dominio de s tenemos que

$$\frac{U(s)}{E(s)} = \frac{K_i}{s} \implies \frac{U(z)}{E(z)} = \frac{K_i T z}{z - 1} = \frac{K_i T}{1 - z^{-1}} \dots\dots\dots(4.11)$$

manipulando la expresión anterior para encontrar U(z)

$$U(z) - U(z)z^{-1} = E(z)K_i T \implies U(z) = E(z)K_i T + U(z)z^{-1} \dots\dots\dots(4.12)$$

trasladando al dominio del tiempo discreto tendremos que la acción integral queda definida como

$$U[n] = K_i T e[n] + u[n - 1] \dots\dots\dots(4.13)$$

Si observamos la expresión anterior, notaremos que obedece a las características del método de integración de Euler. Ahora bien, si deseamos diseñar un control proporcional-integral digital con este método la expresión a utilizar será la siguiente:

$$U[n] = K_p e[n] + (K_i T e[n] + u[n - 1]) \dots\dots\dots(4.14)$$

Método de integración por transformación bilineal. En la transformación bilineal se queda descrito en el dominio de z como:

$$s = \frac{2}{T} \cdot \frac{z - 1}{z + 1} \dots\dots\dots(4.15)$$

Entonces en el dominio de z la función de transferencia de la acción integral quedaría definida como:

$$\frac{U(z)}{E(z)} = \frac{K_i T}{2} \left(\frac{1 + z^{-1}}{1 - z^{-1}} \right) \dots\dots\dots(4.16)$$

Si manipulamos la expresión y la trasladamos al dominio del tiempo discreto, la expresión para la acción integral es la siguiente:

$$U[n] = \frac{K_i T}{2} (e[n] + e[n - 1]) + u[n - 1] \dots\dots\dots(4.17)$$

De la expresión anterior podemos observar que cumple con las características del método de integración trapezoidal. Si deseamos lograr el diseño de un control proporcional-integral utilizando éste método la expresión resultante es:

$$u[n] = K_p e[n] + \left(\frac{K_i T}{2} (e[n] + e[n - 1]) + u[n - 1] \right) \dots\dots\dots(4.18)$$

2.4.2.3 Control Proporcional-Derivativo (PD)

La acción de control proporcional-derivativa, se define mediante:

$$u(t) = K_p e(t) + K_p T_d \frac{d}{dt} e(t) \dots\dots\dots(4.19)$$

donde T_d es una constante denominada tiempo derivativo. Esta acción tiene carácter de previsión, lo que hace más rápida la acción de control, aunque tiene la desventaja importante que amplifica las señales de ruido y puede provocar saturación en el actuador. La acción de control derivativa nunca se utiliza por sí sola, debido a que sólo es eficaz durante períodos transitorios.

La acción derivativa tiene los siguientes efectos sobre un sistema:

- Reduce el sobreimpulso máximo.
- Disminuye el tiempo de asentamiento.
- Aumenta el ancho de banda.

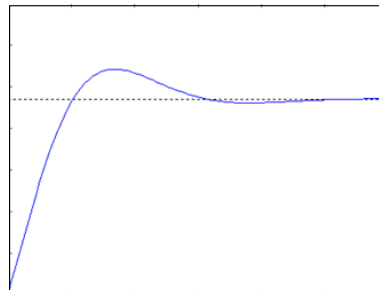


Figura 19: Efecto de la acción derivativa

Dado que la ganancia que la ganancia derivativa está dada por:

$$K_d = K_p T_d \dots\dots\dots(4.20)$$

Reescribimos la ecuación (4.7) de la siguiente manera:

$$u(t) = K_p e(t) + K_d \frac{d}{dt} e(t) \dots \dots \dots (4.21)$$

La función de transferencia de un controlador PD resulta:

$$C_{PD}(s) = K_p + K_d s \dots \dots \dots (4.22)$$

Cuando una acción de control derivativa se agrega a un controlador proporcional, permite obtener un controlador de alta sensibilidad, es decir que responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Aunque el control derivativo no afecta en forma directa al error en estado estacionario, añade amortiguamiento al sistema y, por tanto, permite un valor más grande que la ganancia K , lo cual provoca una mejora en la precisión en estado estable.

Discretización de la acción derivativa.

Método de derivación simple. En el método de derivación simple utilizamos la expresión utilizada para el método de derivación simple, por lo tanto, podemos afirmar que

$$\frac{U(s)}{E(s)} = K_d s \implies \frac{U(z)}{E(z)} = \frac{K_d(1 - z^{-1})}{T} \dots \dots \dots (4.23)$$

Manipulando la expresión y trasladando al dominio del tiempo discreto, la expresión de la acción derivativa es la siguiente:

$$U[n] = K_d \left(\frac{e[n] - e[n - 1]}{T} \right) \dots \dots \dots (4.24)$$

Para un control proporcional-derivativo:

$$U[n] = K_p e[n] + K_d \left(\frac{e[n] - e[n - 1]}{T} \right) \dots \dots \dots (4.25)$$

Método de derivación por transformación bilineal.

Utilizando la expresión de transformación bilineal podemos afirmar que

$$\frac{U(s)}{E(s)} = K_d s \implies \frac{U(z)}{E(z)} = K_d \left(\frac{2}{T} \cdot \frac{1 - z^{-1}}{1 + z^{-1}} \right) \dots \dots \dots (4.26)$$

manipulando la expresión y trasladando al dominio del tiempo discreto, la expresión de la acción derivativa es la siguiente:

$$U[n] = \frac{2K_d}{T} (e[n] - e[n - 1]) + u[n - 1] \dots \dots \dots (4.27)$$

Para un control proporcional-derivativo:

$$U[n] = K_p e[n] + \left(\frac{2K_d}{T} (e[n] - e[n - 1]) + u[n - 1] \right) \dots \dots \dots (4.28)$$

2.4.2.4 CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO (PID)

Ésta acción combinada reúne las ventajas de cada una de las tres acciones de control individuales. La ecuación de un controlador con esta acción combinada se obtiene mediante:

$$u(t) = K_p e(t) + \frac{K_p}{T_i} \int_0^t e(\tau) d\tau + K_p T_d \frac{d}{dt} e(t) \dots \dots \dots (4.29a)$$

ó bien

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \dots \dots \dots (4.29b)$$

La función de transferencia resulta:

$$C_{PID}(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \dots \dots \dots (4.30a)$$

o bien

$$C_{PID}(s) = K_p + \frac{K_i}{s} + K_d s \dots \dots \dots (4.30b)$$

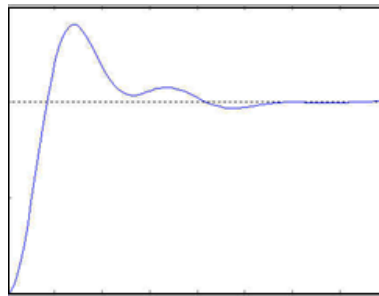


Figura 20: Respuesta de un control PID

Si deseamos discretizar el control pid lo que tenemos que hacer es elegir un método de integración y un método de derivación discretos, y combinar las acciones con el control proporcional, por ejemplo, la siguiente ecuación combina el método de transformación bilineal y el método de derivación simple:

$$u[n] = K_p e[n] + K_i \left(\frac{T}{2} (e[n] + e[n - 1]) + u[n - 1] \right) + K_d \left(\frac{e[n] - e[n - 1]}{T} \right) \dots \dots \dots (4.31)$$

y es así como podemos diseñar un control pid discreto.

2.5. Métodos de sintonización de Ziegler and Nichols

Sabemos que la función de transferencia de un controlador PID está dada por:

$$C_{PID}(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) = K_p + \frac{K_i}{s} + K_d s \dots \dots \dots (5.1)$$

Para determinar los parámetros de un controlador PID Ziegler and Nichols propusieron dos métodos de sintonización basados en estudios experimentales de respuesta al escalón de diferentes sistemas. Estas reglas son usadas para sistemas en los que son conocidas las estructuras de los modelos matemáticos. Los valores obtenidos son aproximados, obteniendo una respuesta estable, aunque puede ser que ciertos parámetros, como por ejemplo la sobre-oscilación, no suelen resultar aceptables, por lo que se requerirá un ajuste fino posterior. Ziegler-Nichols propusieron 2 métodos: en lazo abierto, y en lazo cerrado.

2.5.1. Método de lazo abierto

El primer método (conocido también como método de la curva de reacción) se obtiene a partir de la respuesta a escalón en lazo abierto y a partir de allí el modelo del proceso puede ser aproximado a la siguiente función de transferencia:

$$\frac{Y(s)}{U(s)} = \frac{K e^{-Ls}}{\tau s + 1} \dots \dots \dots (5.2)$$

en la cual los parámetros K , τ y L corresponden con la ganancia, la constante de tiempo y el retardo del proceso y deben ser identificados a partir de dicha respuesta.

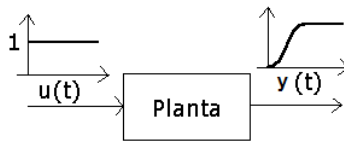


Figura 21: Diagrama de bloques del método de lazo abierto

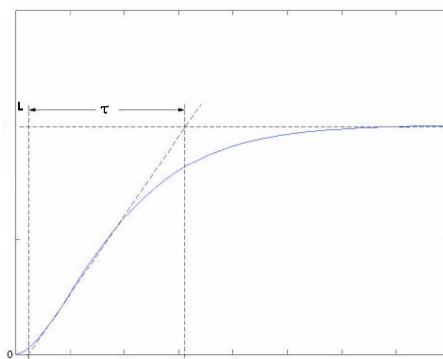


Figura 22: Curva de reacción

Sin embargo, determinar el valor de L resulta complicado, para ello C.L. Smith realizó una aproximación de dos puntos del transitorio: t_{28} y t_{63} . Estos puntos corresponden al momento en el que se alcanzan 28.3% y 63.2% del cambio total. Para estos parámetros dinámicos se emplean las siguientes expresiones:

$$t_{28} = L + \frac{\tau}{3} \dots \dots \dots (5.3)$$

$$t_{63} = L + \tau \dots \dots \dots (5.4)$$

Para encontrar el valor de L y τ simplemente se manipulan algebraicamente las expresiones (5.3) y (5.4)

$$\tau = \frac{3}{2}(t_{63} - t_{28}) \dots \dots \dots (5.5)$$

$$L = t_{63} - \tau \dots \dots \dots (5.6)$$

Una vez identificados los parámetros antes mencionados se fijan los parámetros del controlador de acuerdo a la siguiente tabla, dependiendo del tipo de controlador que se implemente

Tipo de Controlador	K_c	T_i	T_d
P	$\frac{\tau}{L}$	∞	0
PI	$0,9 \frac{\tau}{L}$	$\frac{L}{0,3}$	0
PID	$1,2 \frac{\tau}{L}$	$2L$	$0,5L$

2.5.2. Método de lazo cerrado

Este método (conocido también como método de oscilación continua) solamente puede aplicarse cuando un proceso a lazo cerrado, como el que se muestra en la figura 18, presenta una respuesta que oscile continuamente, como la que se muestra en la figura 19, para un valor específico de su ganancia a lazo abierto, es decir, el sistema a lazo cerrado tiene una ganancia crítica, la cual corresponde con el límite de estabilidad del sistema a lazo cerrado.

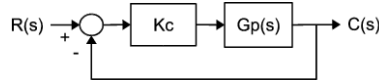


Figura 23: Diagramas a Bloques de Lazo cerrado

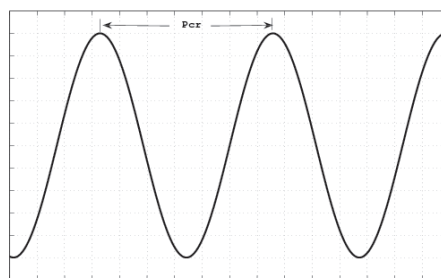


Figura 24: Oscilación continua

Tanto la ganancia crítica (K_{cr}) como el período crítico (P_{cr}) pueden ser determinados en forma experimental, a partir de los cuales los parámetros del controlador pueden fijarse según la tabla que se muestra a continuación.

Tipo de Controlador	K_c	T_i	T_d
P	$0,5K_{cr}$	∞	0
PI	$0,45K_{cr}$	$\frac{P_{cr}}{1,2}$	0
PID	$0,5K_{cr}$	$0,5P_{cr}$	$0,125P_{cr}$

Adicionalmente, si se conoce la función de transferencia del proceso los valores de la ganancia y del periodo críticos podrán ser obtenidos teóricamente utilizando el criterio de estabilidad de Routh-Hurwitz. Al igual que en el primer método, los parámetros determinados a través de la tabla servirán como inicio a partir de los cuales se ajustarán los mismos hasta lograr la respuesta deseada.

3. USO DE LA DAQ NI USB 6008

3.1. Introducción

3.1.1. Adquisición de Datos

La adquisición de datos o adquisición de señales, consiste en la toma de muestras del mundo real (sistema analógico) para generar datos que puedan ser manipulados por un ordenador u otro sistema electrónico (sistema digital). Se requiere una etapa de acondicionamiento, que adecua la señal a niveles compatibles con el elemento que hace la transformación a señal digital. El elemento que hace dicha transformación es el módulo de digitalización o tarjeta de Adquisición de Datos (DAQ). El propósito de adquisición de datos es medir un fenómeno eléctrico y físico como voltaje, corriente, temperatura, presión o sonido. La adquisición de datos basada en PC utiliza una combinación de hardware modular, software de aplicación y una PC para realizar medidas. Mientras cada sistema de adquisición de datos se define por sus requerimientos de aplicación, cada sistema comparte una meta en común de adquirir, analizar y presentar información. Los sistemas de adquisición de datos incorporan señales, sensores, actuadores, acondicionamiento de señales, dispositivos de adquisición de datos y software de aplicación.

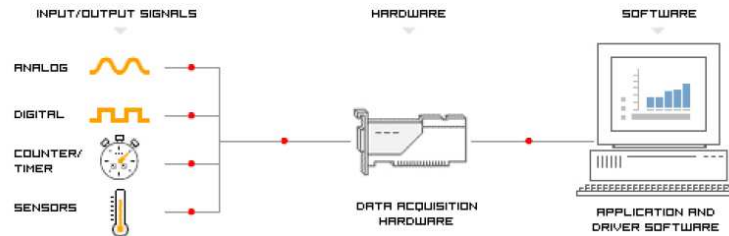


Figura 25: Sistema de adquisición de Datos

Es importante recalcar que de acuerdo al Teorema de Nyquist, al muestrear una señal, la frecuencia de muestreo debe ser mayor que dos veces el ancho de banda de la señal de entrada, para poder reconstruir la señal original de forma exacta a partir de sus muestras. En caso contrario, aparecerá el fenómeno del aliasing que se produce al infra-muestrear. Si la señal sufre aliasing, es imposible recuperar el original. Velocidad de muestreo recomendada:

- 2*frecuencia mayor (medida de frecuencia)
- 10*frecuencia mayor (detalle de la forma de onda)

3.1.2. ¿Qué es la DAQ NI USB 6008?

La DAQ NI USB 6008 es un dispositivo de adquisición de datos de la compañía National Instruments que posee las siguientes características:

- Está provista de 8 Entradas Analógicas para señales de ± 10 Volts con respecto a GND (12 bits, 10 kS/s).
- 2 Salidas Analógicas de 0 a 5 Volts (12 bits a 150 S/s).
- 12 Líneas de Entradas y Salidas Digitales con valores lógicos de 0 a 5 Volts.
- Contador de 32 bits.

Este dispositivo es capaz de ejecutar simultáneamente múltiples tareas interactivas, esto quiere decir que puede adquirir señales analógicas, generar niveles analógicos de salida, y desempeñar Entradas y/o Salidas Digitales. Con esta funcionalidad se pueden desarrollar aplicaciones que van desde el registro de datos, generación de señales, e inclusive, control de lazo cerrado. El dispositivo luce como se muestra en la siguiente figura:

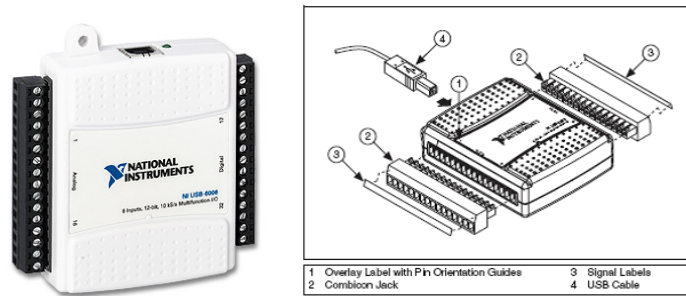


Figura 26: Vista de la DAQ NI USB 6008

La configuración de terminales para las Entradas y Salidas Analógicas se muestra en la siguiente tabla:

Module	Terminal	Signal, Single-Ended Mode	Signal, Differential Mode
	1	GND	GND
	2	AI 0	AI 0+
	3	AI 4	AI 0-
	4	GND	GND
	5	AI 1	AI 1+
	6	AI 5	AI 1-
	7	GND	GND
	8	AI 2	AI 2+
	9	AI 6	AI 2-
	10	GND	GND
	11	AI 3	AI 3+
	12	AI 7	AI 3-
	13	GND	GND
	14	AO 0	AO 0
	15	AO 1	AO 1
	16	GND	GND

Figura 27: Terminales de Entradas y Salidas Analógicas

La configuración de las Entradas y Salidas Digitales se muestra en la siguiente tabla:

Module	Terminal	Signal
	17	P0.0
	18	P0.1
	19	P0.2
	20	P0.3
	21	P0.4
	22	P0.5
	23	P0.6
	24	P0.7
	25	P1.0
	26	P1.1
	27	P1.2
	28	P1.3
	29	PFI 0
	30	+2.5 V
	31	+5 V
	32	GND

Figura 28: Terminales de Entradas y Salidas Digitales

Nota: Es necesario realizar la instalación del software de soporte DAQmx 8.8 que es proporcionado junto con la tarjeta antes de realizar cualquier operación.

3.1.3. El entorno Labview

LabVIEW (Laboratory Virtual Instrument Engineering Workbench) es un lenguaje de programación gráfico para el diseño de sistemas de adquisición de datos, instrumentación y control. Labview permite diseñar interfaces de usuario mediante una consola interactiva basado en software. Labview es a la vez compatible con herramientas de desarrollo similares y puede trabajar con programas de otra área de aplicación, como por ejemplo Matlab. Tiene la ventaja de que permite una fácil integración con hardware, específicamente con tarjetas de medición, adquisición y procesamiento de datos.

La programación gráfica de Labview consta de un panel frontal y un panel de código. En el panel frontal es donde se diseña la interface de usuario y se ubican los controles e indicadores. En el panel de código (diagrama a bloques) se encuentran las funciones. Cada control que se utiliza en la interfaz tiene una representación en el panel de código, igualmente los indicadores necesarios para entregar la información procesada al usuario tienen un icono que los identifica en el panel de código o de programación. Los controles pueden ser booleanos, numéricos, strings, un arreglo matricial de estos o una combinación de los anteriores; y los indicadores pueden ser como para el caso de controles pero pudiéndolos visualizar como tablas, gráficos en 2D o 3D, browser, entre otros.

Cuando se diseñan programas con Labview se está trabajando bajo algo denominado VI, es decir, un instrumento virtual, se pueden crear VI a partir de especificaciones funcionales que se diseñen. Este VI puede utilizarse en cualquier otra aplicación como una subfunción dentro de un programa general. Los VI's se caracterizan por: ser un cuadrado con su respectivo símbolo relacionado con su funcionalidad, tener una interfaz con el usuario, tener entradas con su color de identificación de dato, tener una o varias salidas y por su puesto ser reutilizables.

Labview tiene su mayor aplicación en sistemas de medición, como monitoreo de procesos y aplicaciones de control, un ejemplo de esto pueden ser sistemas de monitoreo en transportación, Laboratorios para clases en universidades, procesos de control industrial. Labview es muy utilizado en procesamiento digital de señales (wavelets, FFT, Total Distorsion Harmonic TDH), procesamiento en tiempo real de aplicaciones biomédicas, manipulación de imágenes y audio, automatización, diseño de filtros digitales, generación de señales, entre otras, etc.

3.2. Adquisición de Entradas Analógicas y registro de Datos

3.2.1. Adquisición y registro de datos de una señal analógica

Antes de comenzar la adquisición de datos comenzaremos conectando la tarjeta NI USB 6008 a un generador de señales y elegiremos:

- Una Terminal de *GND* (para éste caso elegiremos la Terminal 1).
- El canal *ai0* (Terminal 2).

Conectaremos ambas terminales a la salida del generador. En caso de no tener a su disposición un generador de señales puede conectar el canal mencionado de la tarjeta a un potenciómetro, ya que la tarjeta posee internamente un divisor de voltaje que entrega aproximadamente 1.4 Volts cuando el canal no tiene ninguna carga.

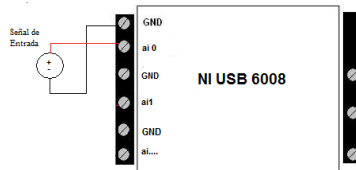


Figura 29: Adquisición de una entrada Analógica

Después creamos un nuevo VI en Labview para construir el programa que necesitamos para realizar la adquisición de datos seleccionando la opción *Blank VI*.

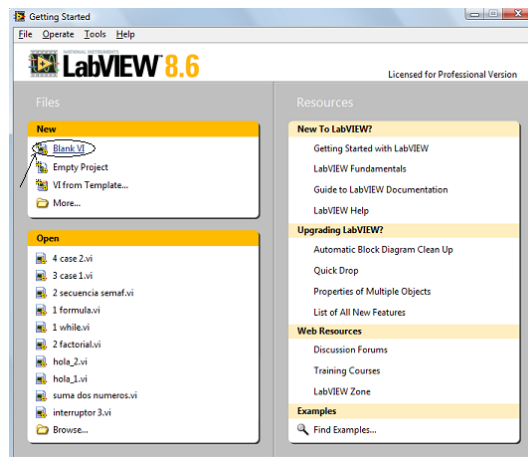


Figura 30: Creación de un Blank VI

Al hacerlo, aparecerá un nuevo panel frontal sin título y un diagrama a bloques en blanco:

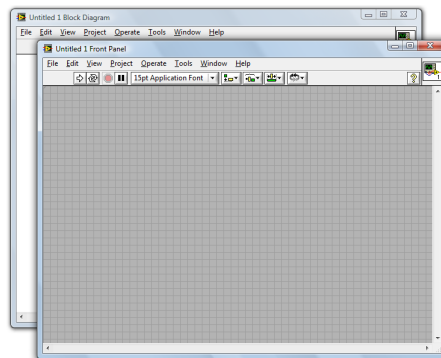


Figura 31: Nuevo Panel Frontal y Nuevo Diagrama a Bloques

El primer paso a realizarse al construir el programa es colocar dos visualizadores de señal *Waveform Graph* en el panel frontal. Con un click derecho aparecerá el menú *Controls*, después elija *Graph Indicators* | *Waveform Graphs*:

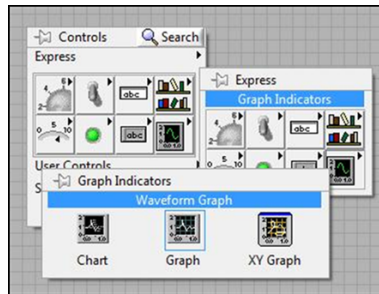


Figura 32: Selección del Waveform Graph

La gráfica del *Waveform Graph* será utilizada para mostrar los datos que están siendo adquiridos por la tarjeta. La gráfica del *Waveform Graph 2* servirá para mostrar los datos del archivo que se generen una vez que se termine la adquisición. El aspecto del panel frontal será el que se muestra en la figura:

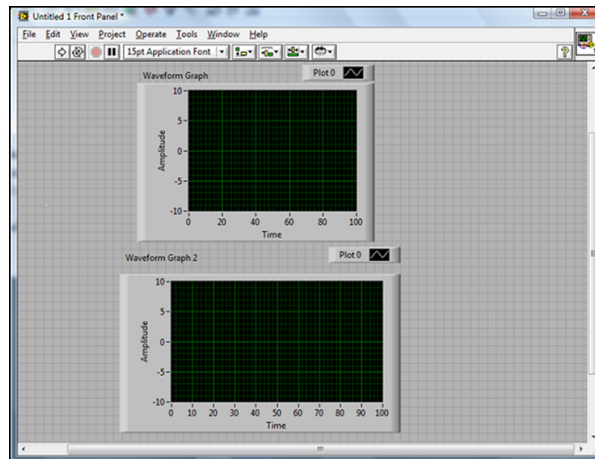


Figura 33: Vista desde el panel frontal

Posteriormente, debemos ir a la ventana donde se encuentra el diagrama de bloques. Una vez en ella pulsamos click derecho para que aparezca el menú *Functions* y llamamos a la función de adquisición de datos seleccionando *Measurement I/O* | *DAQ mx* | *DAQ Assistant*:

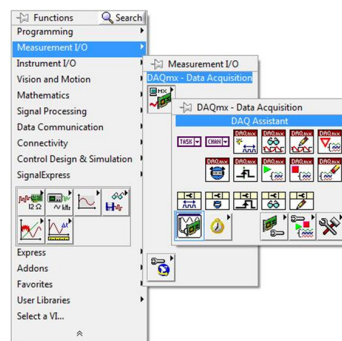


Figura 34: Selección del DAQ Assistant

Luego, pulsando doble click sobre el *DAQ Assistant* para aparecerá el cuadro de diálogo *Create a New Express Task*. Seleccionando *Acquire signals/Analog input/Voltage* indicamos que el tipo de entrada es una señal de voltaje analógico:

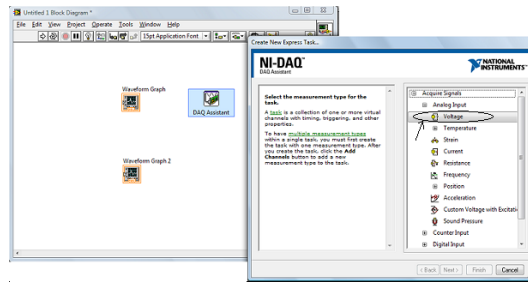


Figura 35: Selección de Entrada Analógica

Después elegimos el canal ai0:

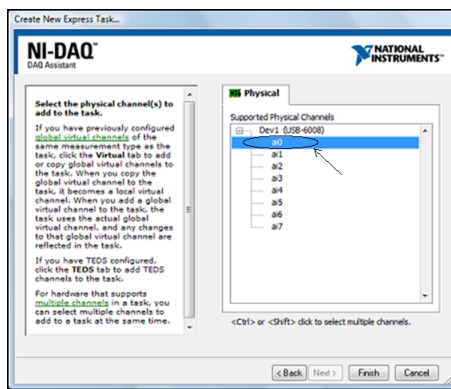


Figura 36: Selección del canal ai0

Una vez elegido el canal, debemos configurar el DAQ Assistant (figura 37) de la siguiente manera:

- Rango de entrada (Signal Input Range): Mín -10 Volts, Máx 10 Volts.
- La configuración de la terminal (Terminal Configuration): *RSE (Referencial Signal Ended)*
- El modo de adquisición (Adquisition Mode): *Continuoss Samples*:

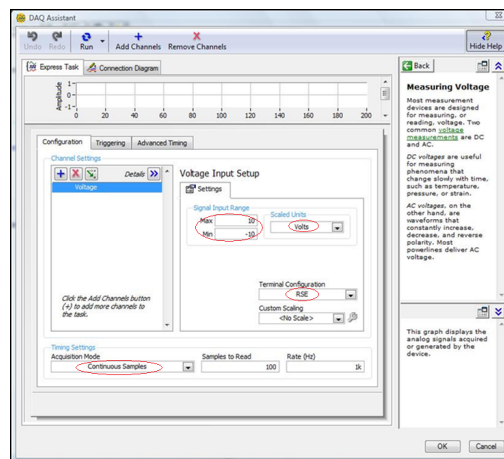


Figura 37: Configuración del DAQ Assistant

Al hacer lo anterior, Labview coloca automáticamente un ciclo while para ejecutar el programa de forma continua. Después debemos conectar los datos del *DAQ assistant* al *Waveform Graph*, tal como se muestra en la siguiente figura:

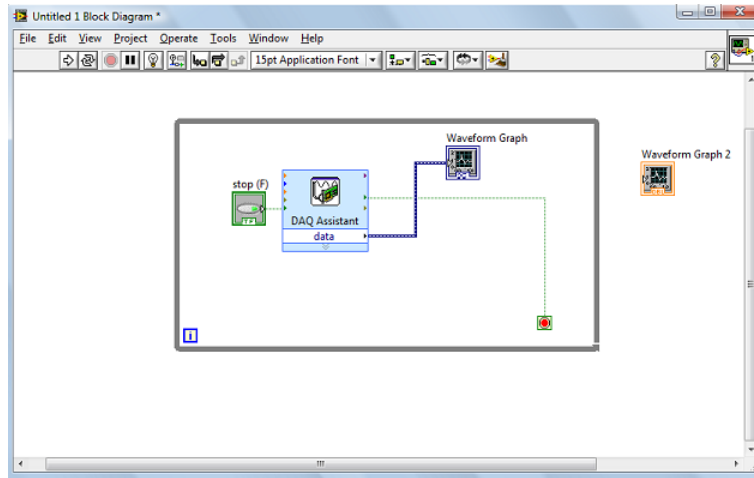


Figura 38: Diagrama a Bloques

Oprimiendo click derecho para que aparezca el menú *function* y seleccionando la ruta *programming | File I/O | Write to measurement File*, encontraremos la función *Write to measurement File* en el diagrama a bloques (Figura 39).

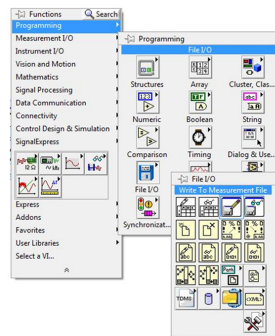


Figura 39: Selección del Write To Measurement File

Con un doble click izquierdo sobre *Write to Measurement File* se mostrará un cuadro de diálogo llamado *Configure Write To Measurement File*, (figura 40) que debe ser configurado seleccionando las opciones:

- *Binary*
- *One header only*
- *One column for channel*

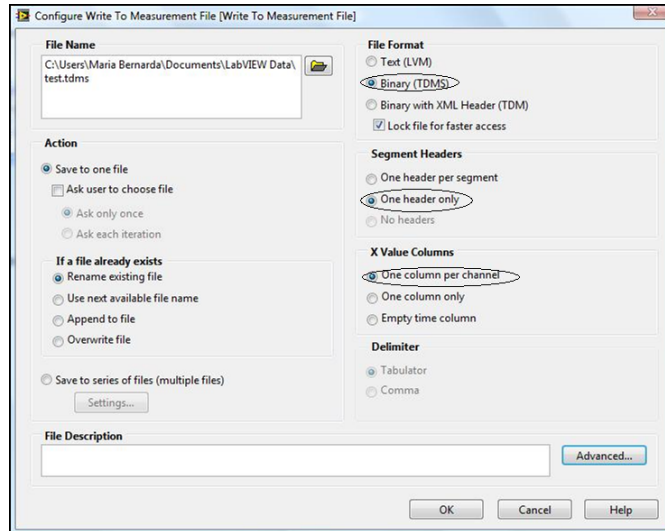


Figura 40: Configurando el Write To Measurement File

Luego, el conexionado deberá ser realizado como se muestra en la figura 41:

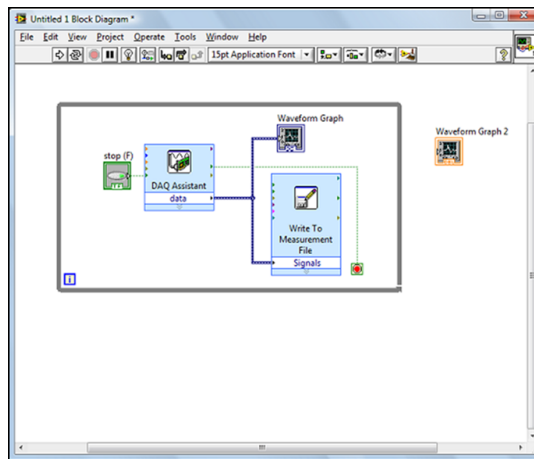


Figura 41: Diagrama Bloques

Ahora, pulsaremos click derecho para que aparezca el menú Functions. Una vez realizado ha de seleccionarse la ruta *programming | File I/O | Read for measurement File*, para que se muestre en el diagrama a bloques la función *Read for measurement File*:

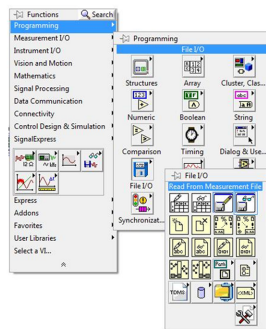


Figura 42: Selección del Read From Measurement File

Dando doble click izquierdo sobre *Read From Measurement File* se mostrará en pantalla un cuadro de diálogo llamado *Configure Read From Measurement File* (Figura 43), que debe de ser configurado seleccionando la opción *binary TDMS*:

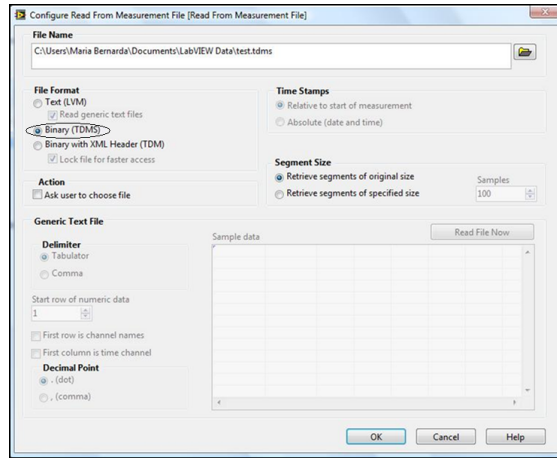


Figura 43: Configuración del Read from Measurement File

Posteriormente regresamos a la ventana del diagrama a bloques y conexionamos como se observa en la figura:

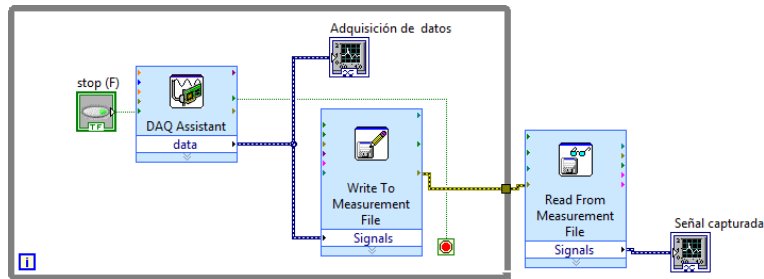


Figura 44: Diagrama a bloques para adquisición de Señales Analógicas y Registro de Datos

La vista del panel frontal es la siguiente:

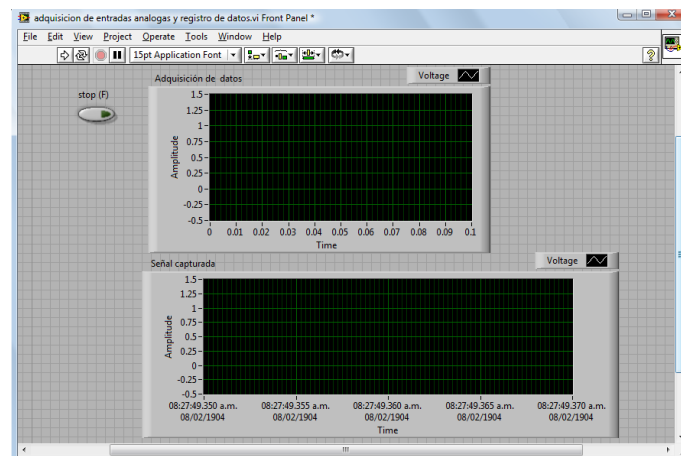


Figura 45: Panel frontal

Para configurar el *waveform graph* pulse click derecho y seleccione la opción *Properties* para que aparezca el siguiente cuadro de diálogo:

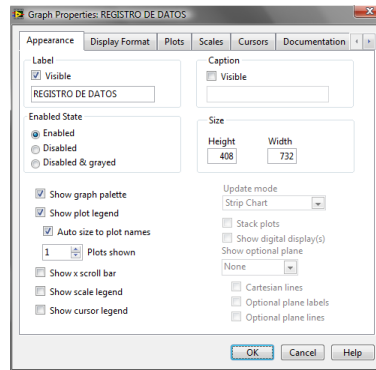
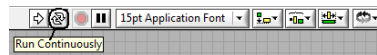


Figura 46: Configuración del Waveform Graph

En el cuadro de diálogo anterior podemos configurar, entre otras opciones, las escalas de tiempo que deseamos utilizar (dado que éstas originalmente en milisegundos y en tiempo absoluto), número de gráficas a visualizar, etc. Para los *Waveform Chart* la configuración se realiza de forma muy similar, dado que el cuadro de diálogo de configuración es prácticamente el mismo.

Ejecución del programa

1. Pulse el ícono *Run Continuously* para ejecutar el programa



2. Luego de ejecutar el programa durante un lapso de tiempo pulse el control boleano *stop* colocado en el panel frontal y observara que en el *Waveform Graph 2* quedarán registrados los datos adquiridos hasta ese momento, y en el *Waveform Graph* seguirán apareciendo los datos registrados en el instante.



3. Si desea dejar de ejecutar el programa pulse el ícono stop que se encuentra junto al ícono run continuously.

Guardando la imagen de la señal adquirida Si se desea guardar la imagen como un archivo, coloque el cursor de mouse sobre el *Waveform Graph* de captura de señal y oprimiendo click derecho aparecerá un submenú, en cual seleccionaremos *Export Simplified Image*. Después aparecerá un cuadro de diálogo y en el seleccionaremos la opción *Save to File* en el formato que se haya preferido.

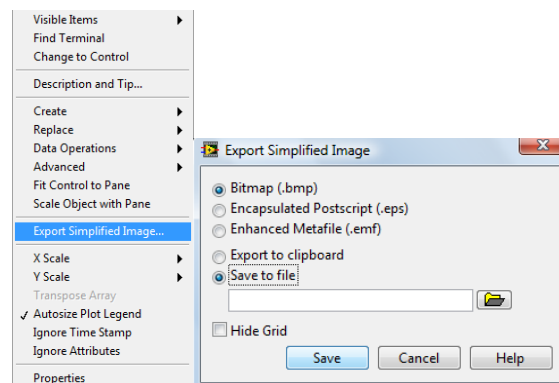


Figura 47: Selección de Export Simplified Image

3.2.2. Adquirir entradas analógicas simultáneamente

En ocasiones, necesitamos adquirir más de una señal analógica de manera simultánea utilizando el *DAQ Assistant*. Dando doble click sobre el ícono del asistente aparecerá el cuadro de diálogo de configuración del asistente. El canal para añadir una señal se agrega pulsando el ícono *Add Channels*, pudiendo agregarse hasta cuatro canales en total si ésto llegara a ser necesario.

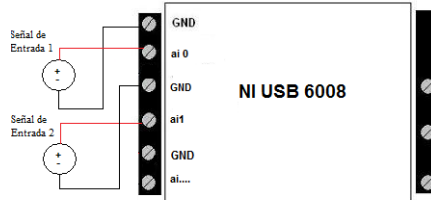


Figura 48: Conexión de la DAQ NI USB 6008 para adquirir mas de una señal de entrada

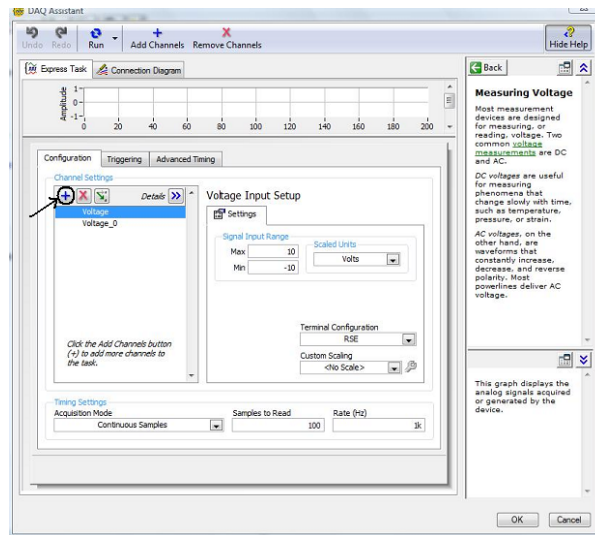


Figura 49: Configuración del DAQ Assistant

3.2.3. Adquisición de entradas analógicas y generación de reportes en Excel.

Existe una alternativa para la adquisición y registro de datos. Ésta consiste en el diseño de un programa mediante el cual generamos reportes en Microsoft Excel, para ello necesitamos contar con el Report Generato Toolkit (y su debida licencia). En la figuras 50 y 51, se muestra un programa (no ahondaremos en la metodología de construcción) de adquisición de señales analógicas y generación de Reportes en Excel. En dicho programa se determina el tiempo en que se generará un reporte y el intervalo de tiempo en que se registrará una medición en el reporte generado. Para fijar los parámetros mencionados lo hacemos desde el panel frontal. El resultado es un reporte generado en un lapso de tiempo determinado, en el cual tenemos una gráfica de la adquisición de datos, y las tablas correspondientes a la medición realizada en los intervalos de tiempo establecido(se visualizan en hora y fecha).

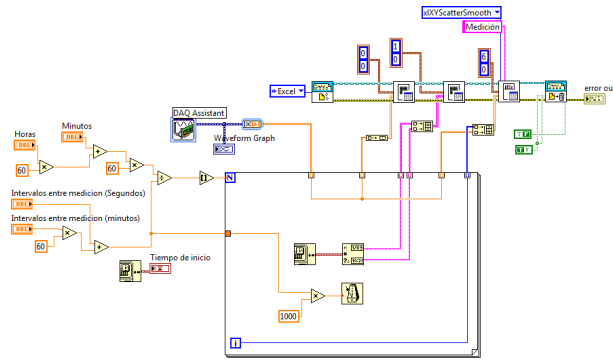


Figura 50: Diagrama a bloques de Generación de Reportes en Excel

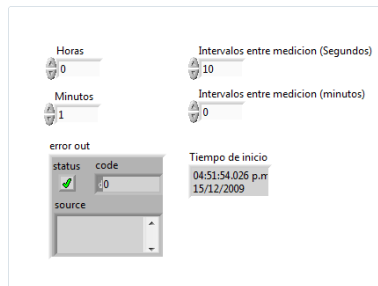


Figura 51: Panel Frontal de Generación de Reportes en Excel

3.3. Generación de Señales Analógicas

3.3.1. Generación de un nivel de voltaje

Para comenzar con la generación de un nivel de voltaje, utilizaremos:

- El canal ao0 (Terminal 14)
- Una de terminal de GND (Terminal 13).

Conectaremos como carga un led en serie con una resistencia de 1k. El led deberá estar en polarización directa, es decir, el lado p debe estar conectada a la Terminal 14 (colocando antes la resistencia) y el lado n debe de ir conectado a la Terminal 13 que corresponde a tierra. Después creamos un nuevo VI y comenzaremos a crear el programa correspondiente en Labview.

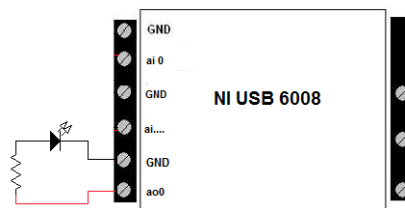


Figura 52: Conexión de la DAQ NI USB 6008 para generar un nivel de voltaje

Una vez que se ha creado un nuevo VI, llamamos a la función *DAQ Assistant* como ya hemos aprendido a hacerlo y debemos seleccionar la opción de generación de señales siguiendo la ruta *Generate signals | Analog output | Voltage*:

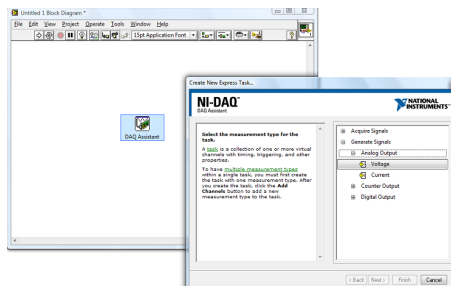


Figura 53: Selección para generar Voltaje de Salida

Después seleccionaremos la salida ao0:

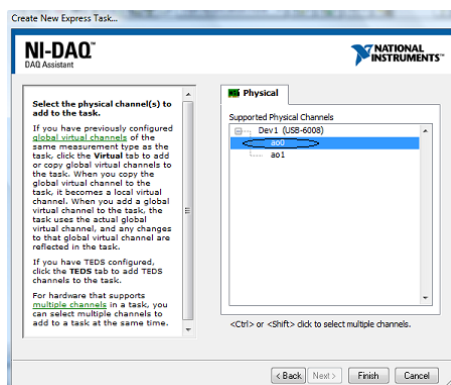


Figura 54: Selección del canal ai0

Ahora configuramos el DAQ Assistant de la siguiente manera:

- Signal Output Range: Mín 0 Volts, Máx 5 Volts.
- Terminal configuration: *RSE*.
- Generation Mode: *1 Sample (On Demand)*.

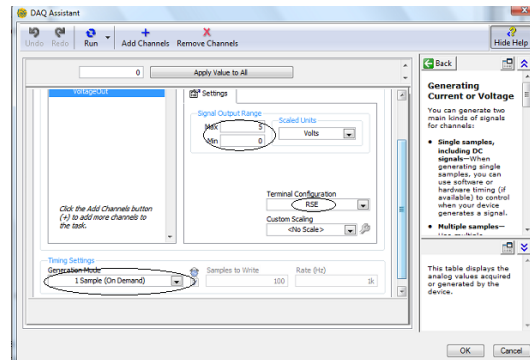


Figura 55: Configuración del DAQ Assistant

Luego seleccionamos *Programming* | *Structures* | *While Loop* y añadimos un botón de paro. Después colocamos una perilla deslizable *Slide* en el panel frontal siguiendo la ruta *Modern* \ *Numeric* \ *Vertical Pointer Slide* como se muestra en la figura:

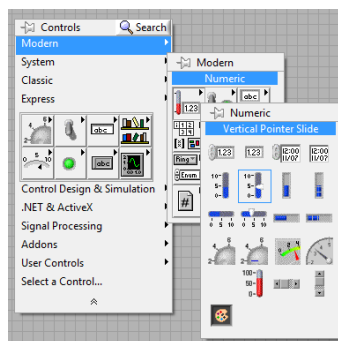


Figura 56: Vertical Pointer Slide

Luego regresamos al programa y conexionamos la perilla con el DAQ assistant:

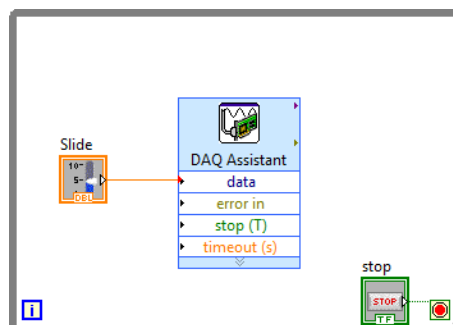


Figura 57: Diagrama a bloques

Regresando al panel frontal la vista es la siguiente:

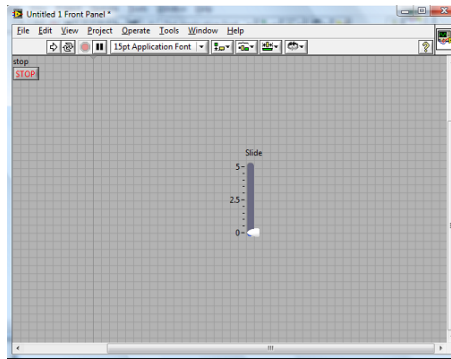


Figura 58: Panel Frontal

Ejecución del programa

1. Pulse el ícono *Run continuously* ejecutar el programa.
2. Coloque el cursor del mouse sobre el *Slide* manteniendo presionado clic izquierdo varíe el valor de 0 a 5. Haciendo ésto observaremos como varía la intensidad de luz del led conectado a la tarjeta debido.
3. Si desea dejar de ejecutar el programa pulse el ícono *Stop* que esta junto al ícono Run continuously.

3.4. Adquisición y Generación de E/S Digitales.

2.4.1 Adquisición de Entradas Digitales

El conexionado de la tarjeta para adquirir una entrada digital es la siguiente:

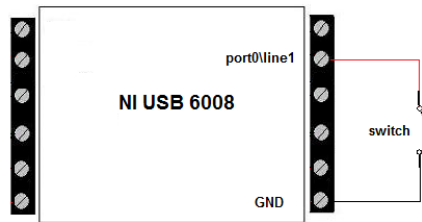


Figura 59: Conexionado de la DAQ NI USB 6008 para adquirir un Entrada Digital

Estando en el panel frontal colocamos un Array siguiendo la ruta Modern\Array, Matrix & Cluster\Array:

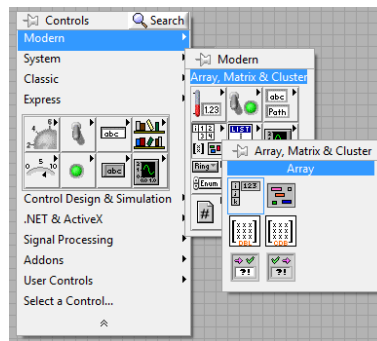


Figura 60: Selección del Array

Después colocamos un Round LED siguiendo la ruta Modern\Boolean\Round LED, esto es para que el pueda existir un flujo de datos de datos de la adquisición hacia el indicador booleano:

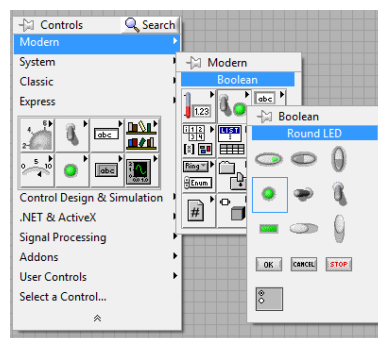


Figura 61: Selección de Round LED

Hecho lo anterior colocamos el indicador booleano dentro del array, la vista en el panel frontal es la siguiente:

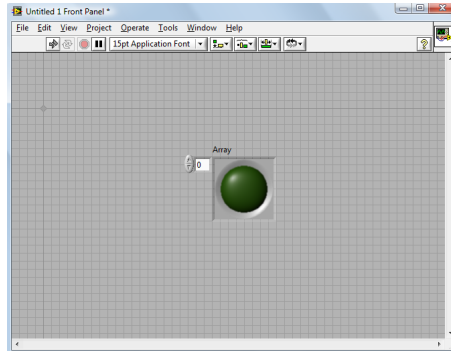


Figura 62: Vista desde el panel frontal

Estando en el diagrama a bloques colocamos un DAQ Assistant y lo configuramos para entradas Digitales siguiendo la ruta Acquire Signals\Digital Input\Line Input.

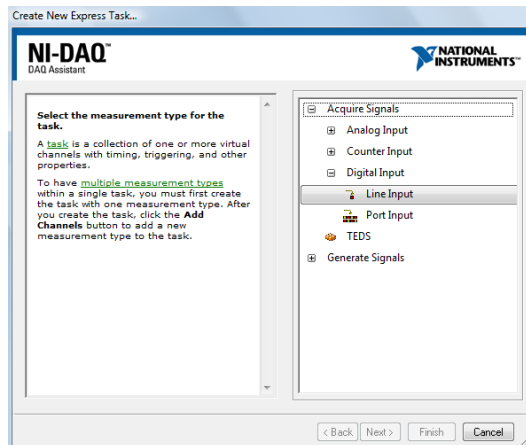


Figura 63: Configuración del Asistente para entradas digitales

Después seleccionamos que terminal usaremos en la adquisición de datos, en éste caso seleccionaremos port0\line1.

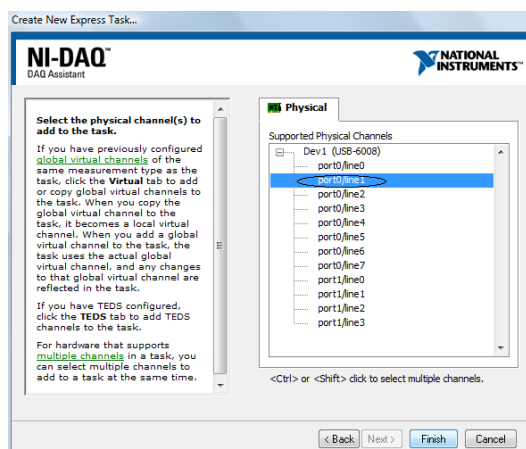


Figura 64: Selección del port0\line1

Hecho lo anterior configuramos el DAQ Assistant seleccionando las opciones:

- Invert Line
- 1 Sample on Demand

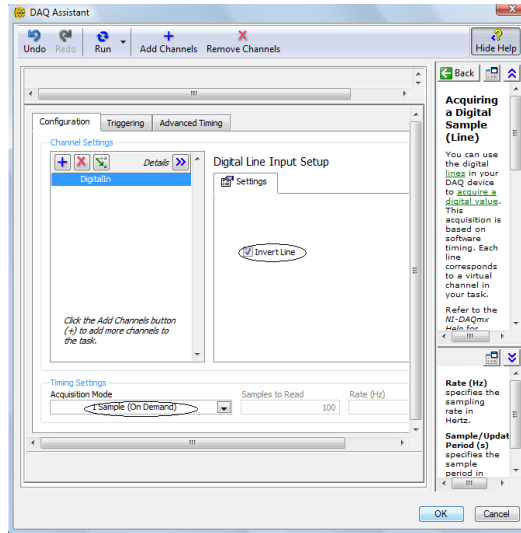


Figura 65: Configuración del Asistente

Estando en el diagrama a bloques colocamos un ciclo while para que el programa se ejecute continuamente, no olvidemos crear un control stop para el ciclo:

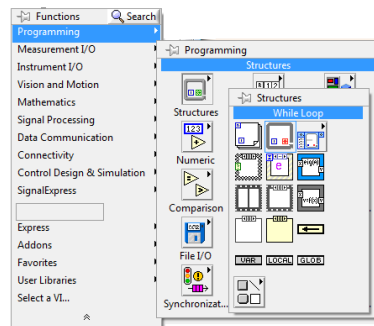


Figura 66: Selección de While Loop

El conexionado del diagrama a bloques es el siguiente:

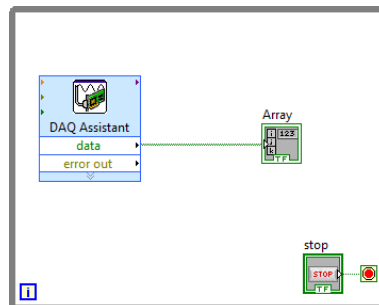


Figura 67: Diagrama a bloques para adquirir Entradas Digitales

Ejecución del programa

1. Ejecute el programa pulsando el botón Run Continuosly
2. Cierre el interruptor que está conectado a la DAQ NI USB 6008 y observe como enciende el indicador booleano en el panel frontal. Cuando el interruptor esté abierto el indicador booleano se apagará.

2.4.2 Generación de Salidas Digitales

El conexionado de la DAQ NI USB 6008 que utilizaremos es el siguiente:

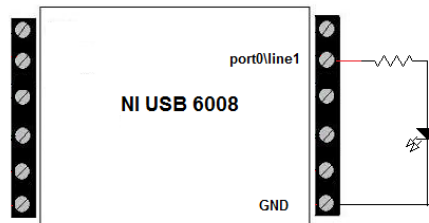


Figura 68: Conexionado de la DAQ NI USB 6008 para generar una Salida Digital

En el panel frontal coloque un Vertical Toggle Switch siguiendo la ruta Modern\Boolean\Vertical Toggle Switch y coloque dentro de un Array, como se hizo anteriormente para el indicador booleano:

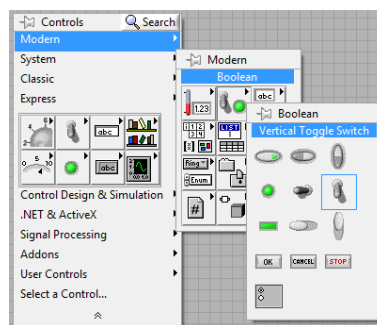


Figura 69: Selección del Vertical Toggle Switch

Coloque un DAQ Assistant en el diagrama a bloques y configúrelo seleccionando Generate Signals\Digital Output\Line Output

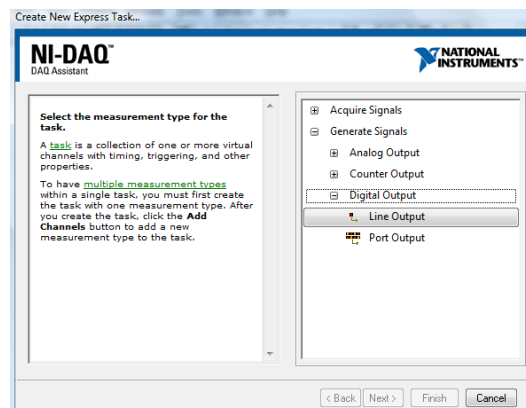


Figura 70: Configuración para Salidas Digitales

Después seleccione línea digital port0\line1:

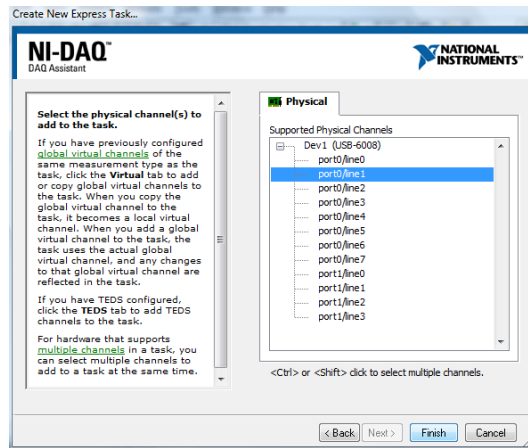


Figura 71: Selección del port0\line1

Hecho lo anterior configure el DAQ Assistant para el modo de adquisición 1 Sample on Demand:

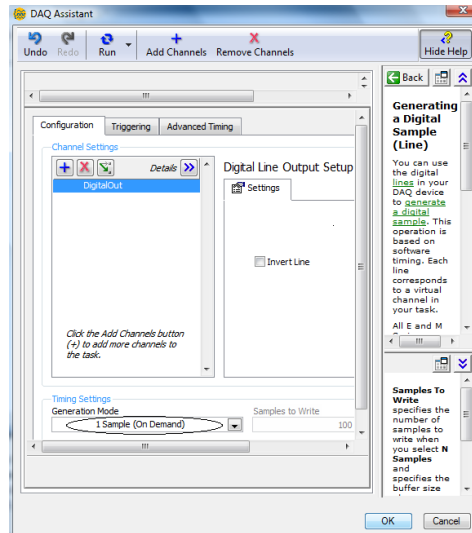


Figura 72: Configuración del DAQ Assistant para salidas digitales

En el diagrama a bloques coloque un ciclo while para que el programa se ejecute continuamente, y conecte los elementos de la siguiente manera:

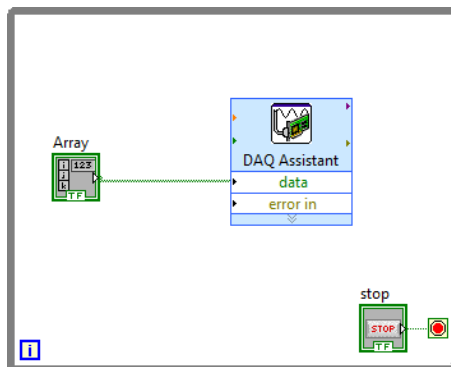


Figura 73: Diagrama a bloques para generar Salidas Digitales

La vista en el panel frontal es la siguiente:

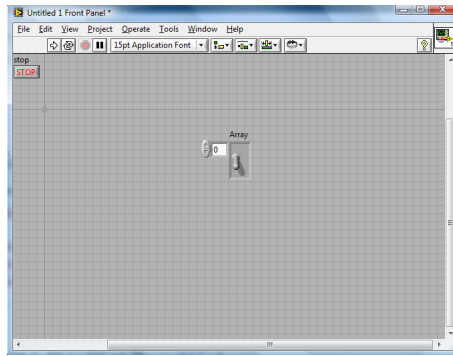


Figura 74: Vista del panel frontal para Generar Salidas digitales

Al ejecutar el programa observamos que cuando switchamos el control boleano desde el panel frontal, el led conectado a la tarjeta se encenderá o apagará. De ésta manera podemos generar señales digitales utilizando la DAQ NI USB 6008.

3.5. Adquisición de Entradas Analógicas y Generación de Salidas Digitales

Dado que ahora conocemos la forma de adquirir entradas y salidas tanto digitales como analógicas cabe mencionar que también es factible utilizar entradas analógicas y salidas digitales simultáneamente. En el siguiente diagrama a bloques se muestra un programa que compara una señal analógica por medio de un comparador *less?* con una referencia y entrega datos de tipo digital para generar salidas digitales a través del DAQ Assistant2 mostrado en el diagrama de bloques del programa.

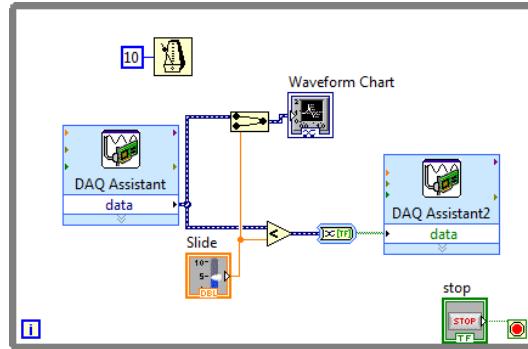


Figura 75: Diagrama de bloques

El conexionado de la tarjeta será utilizando la terminal *ai0* y su respectiva *GND* para realizar la adquisición de datos siguiendo el procedimiento que se mostró en la sección 2.2.1, y utilizando el *port0/line1* como salida digital conectado a un diodo LED como se mostró en la sección 2.4.2. El Asistente utilizado para adquisición de datos será configurado para entradas RSE. En el diagrama a bloques anterior cabe mencionar que ambos DAQ Assistant utilizado están configurados en el modo de adquisición 1 Sample on Demand para que el programa se ejecute correctamente.

La vista desde el panel frontal es la siguiente.

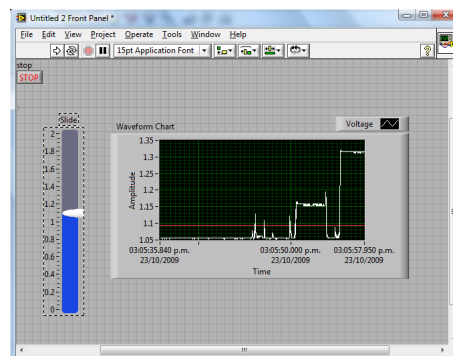


Figura 76: Panel Frontal

4. PRÁCTICAS PROPUESTAS

4.1. PRÁCTICA I: “RESPUESTA TEMPORAL”

4.1.1. INTRODUCCIÓN

La presente práctica trata de la respuesta en el tiempo de sistemas físicos. Se incluyen como sistemas a modelar: Un sistema mecánico traslacional, un filtro pasabajas RC y un filtro pasabajas de Sallen and Key. Se encontrarán modelos de estados y se simularán en Simulink de Matlab. Después deberá llevarse a cabo la implementación física de los filtros pasabajas mencionados para registrar las señales de entrada y salida utilizando la NI USB 6008 en el entorno Labview (el sistema mecánico únicamente será modelado y simulado). Es conveniente leer las Secciones 2.1 y 2.2 de éste manual antes de comenzar con la realización de la práctica.

4.1.2. OBJETIVO DE LA PRÁCTICA

Objetivo general:

Aprender a modelar sistemas físicos utilizando variables de estado y estudiar su respuesta en el tiempo aplicando entrada escalón y diversos tipos de señales de entrada, realizando simulaciones en Simulink de Matlab y utilizando la tarjeta de adquisición de datos NI USB 6008 en el entorno labview.

Objetivos específicos:

1. Hallar el modelo de estados y simular la respuesta al escalón de un sistema mecánico, un filtro pasabajas RC y un filtro Sallen and Key.
2. Implementar el filtro RC y el filtro Sallen and Key para obtener su respuesta al escalón utilizando la tarjeta NI USB 6008.
3. Simular la respuesta en el tiempo de cada uno de los sistemas físicos mencionados a las formas de onda senoidal y cuadrada.
4. Obtener la respuesta a la onda senoidal y a una onda cuadrada de los filtros RC y Sallen and Key implementados.

4.1.3. MATERIALES Y EQUIPO

Materiales

Resistencias, Capacitores, opam 741.

Equipo

1 osciloscopio, 1 Generador de funciones, Fuente Simétrica de dc, Fuente Simple de dc, 1 PC con puertos USB, DAQ NI USB 6008.

4.1.4. DESARROLLO DE LA PRÁCTICA

PRÁCTICA Ia: RESPUESTA DE UN SISTEMA AL ESCALÓN.

1) Simulación de un sistema mecánico traslacional en Simulink.

Considere el siguiente sistema mecánico y realice lo que a continuación se indica:

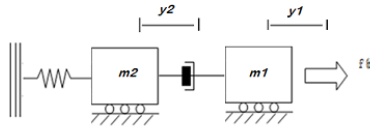


Figura 77: Sistema mecánico

Donde la constante de resorte $k=1$; el coeficiente de amortiguación viscosa $b=1$; la masa $m1=1$; la masa $m2=1$;

a) Hallar el modelo de estados (A,B,C,D) tomando $f(t)=1$ como entrada y el desplazamiento $y2$ como salida.

b) Encuentre la respuesta al escalón del sistema realizando el diagrama de bloques correspondiente en Simulink siguiendo los siguientes pasos:

Paso 1. Desde Matlab ingresar a Simulink library Browser para crear un nuevo modelo.

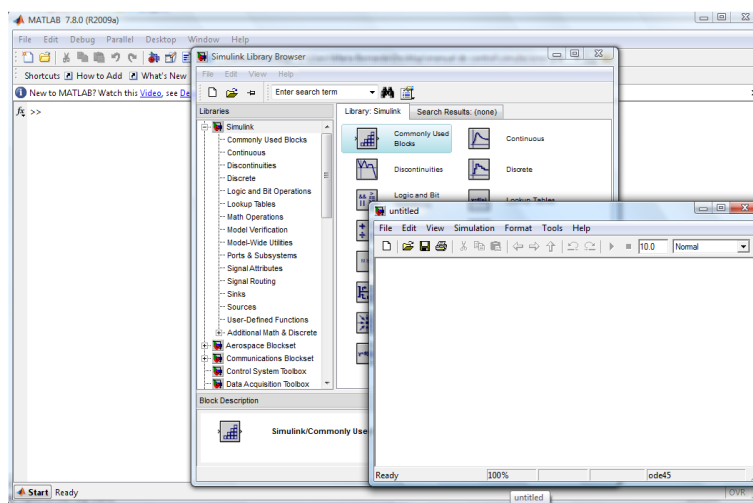


Figura 78: Nuevo modelo

Paso 2. Seleccione la ruta *Simulink* | *Sources* | *Step* y arraste el ícono *Step* a la ventana en blanco.

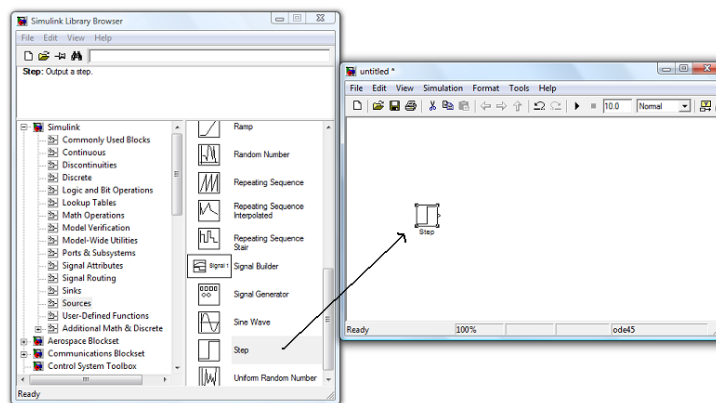


Figura 79: Colocando la herramienta escalón en la ventana en blanco

Paso 3. Del mismo modo seleccione la ruta *Simulink* | *continuous* | *State Space* y arrastre la herramienta *State Space* a la ventana en donde se encuentra el diagrama a bloques.

Paso 4. Coloque ahora la herramienta Scope que se encuentra en *Simulink* | *Sinks* | *Scope* y arrastre la herramienta a la ventana donde se encuentra el diagrama a bloques. Después coloque un multi-

plexor *Mux* en el diagrama a bloques que se encuentra en *Simulink* | *Commonly Used Blocks* | *Mux* y conexione todos los elementos colocados en el diagrama a bloques como se muestra a continuación:

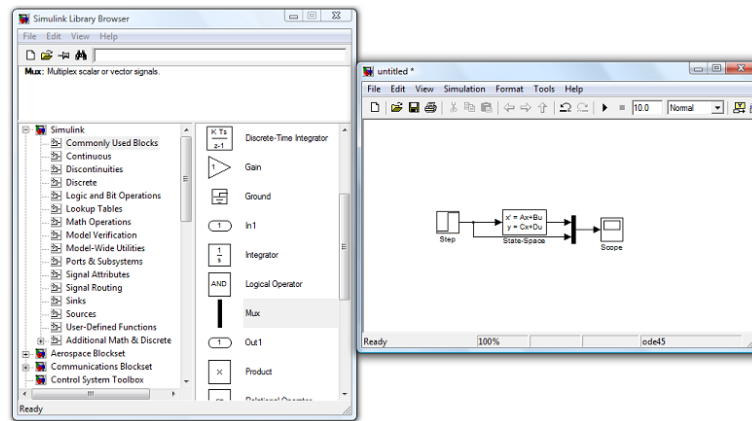


Figura 80: Diagrama a bloques para encontrar la respuesta al escalón

Paso 5. Dé doble click sobre *State-Space* y aparecerá un cuadro de diálogo el cual deberá ser configurado de acuerdo al modelo de estados obtenido. Si el modelo de estados es de segundo orden o más los datos ingresados deberán ser vectores o matrices. Dicho cuadro de diálogo tiene el siguiente aspecto:

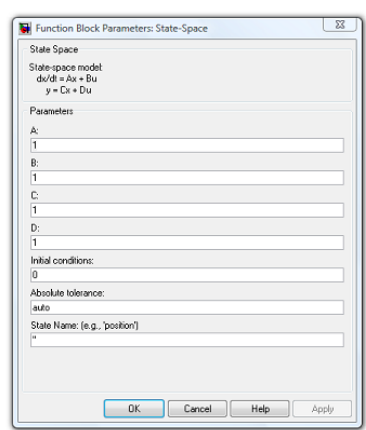


Figura 81: cuadro de diálogo de configuración de modelo de estados

Paso 6. Ahora ejecute el programa pulsando el ícono *start simulación* y de doble click en *Scope* para visualizar los resultados.

Paso 7. Observe los siguientes parámetros en la gráfica obtenida:

- u = Valor de la entrada escalón;
- y_p = Valor máximo de la salida del sistema;
- $y_{(t=\infty)}$ = Valor de la salida en estado estacionario;
- e_{ss} = Error en estado estacionario;
- t_p = tiempo correspondiente para y_p ;
- t_i = tiempo correspondiente a la entrada u .

Paso 8. Registre los resultados en la siguiente tabla.

Cuadro 1: Tabla resumen de Datos

<i>Sistema Mecánico</i>	u	y_p	$y(t=\infty)$	t_p	t_i	$t_p - t_i$	$e_{ss} = 1 - \frac{y_{\infty}}{u}$
SIMULINK							

2) Simulación de un Filtro pasabajas RC en Simulink.

Considere el siguiente circuito y realice lo que a continuación se indica:

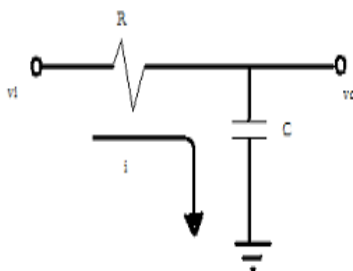


Figura 82: Filtro RC pasabajas

Donde $C=100\mu$ y $R=5.6k$.

- Hallar el modelo de estados tomando como salida el voltaje en el capacitor.
- Hallar su función de transferencia en base al modelo de estados.
- Realizar el diagrama a bloques del sistema utilizando la herramienta *Transfer Fcn* que se encuentra siguiendo la ruta *Simulink | continuous | Transfer Fcn*. Dando doble click sobre ésta herramienta aparecerá un cuadro de dialogo llamado *Function block Parameters: Transfer Fcn*, que deberá ser configurado de acuerdo a la función de transferencia.

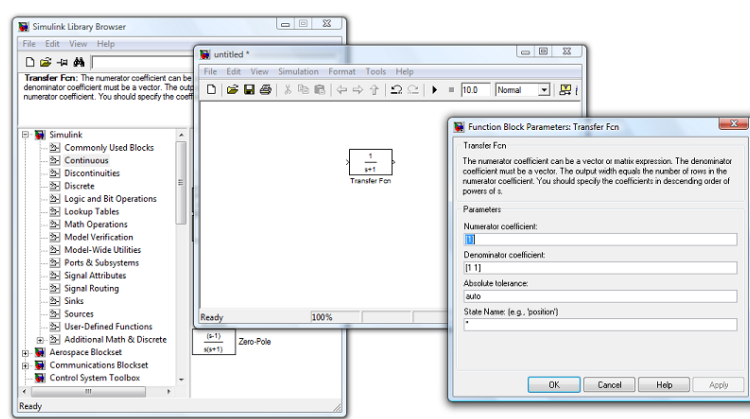


Figura 83: Uso de la herramienta de función de transferencia

Sabiendo lo anterior conéctese la entrada escalón *step* con la función de transferencia y el visualizador *Scope*. Recuerde usar el multiplexor *Mux* para visualizar la señal proveniente del *step* y la señal de salida del *Transfer Fcn*.

- De la señal de salida observe y registre los siguientes parámetros:

- v_i = Valor de la entrada escalón;
- $v_c = v_{5\tau}$ = Valor de la salida en estado estacionario;

- e_{ss} = *Error en estado estable;*
- $t_c = t_{5\tau}$ = *tiempo correspondiente a $v_{5\tau}$;*
- t_i = *tiempo correspondiente al momento de la entrada v_i ;*

3) Simulación de un filtro pasabajas Sallen and key.

El siguiente circuito corresponde a un filtro pasabajas de segundo orden Sallen and Key.

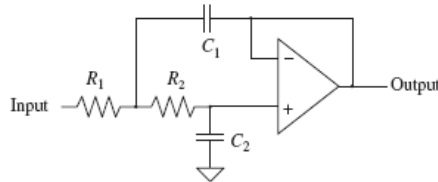


Figura 84: Filtro Sallen and Key

La función de transferencia del circuito es

$$T_{sk} = \frac{\omega_N^2}{s^2 + 2\zeta\omega_N s + \omega_N^2}$$

donde

$$\omega_N^2 = \frac{1}{R1 * C1 * R2 * C2}$$

y

$$2\zeta\omega_N = \frac{R1 + R2}{R1 * R2 * C1}$$

En esta ocasión se trabajará con un filtro cuya función de transferencia es

$$T_{sk} = \frac{100}{s^2 + 20s + 100}$$

para ello los valores de resistencias y capacitores a utilizarse son $C1=100\mu$; $C2=100\mu$; $R1=1k$; $R2=1k$. Sabiendo lo anterior:

- a) Hallar la respuesta al escalón utilizando Simulink tal como se hizo para el filtro pasabajas RC.
- b) De las gráfica obtenidas observe y registre los siguientes parámetros:

- v_i = *Valor de la entrada escalón;*
- v_p = *Valor máximo de la salida del sistema;*
- $v_{(t=\infty)}$ = *Valor de la salida del sistema en estado estable;*
- e_{ss} = *Error en estado estable;*
- t_p = *tiempo correspondiente para v_p ;*
- t_i = *tiempo correspondiente para v_i ;*

4) Respuesta al escalón de un Filtro RC pasabajas utilizando la DAQ NI USB 6008.

En el capítulo 2 se mostró cómo adquirir señales analógicas y como enviar niveles de voltaje por medio de la NI USB 6008, en ésta sección aplicaremos lo aprendido en ese capítulo para realizar las tareas que sean requeridas. Se recomienda corroborar los resultados que sean obtenidos con el osciloscopio.

Sugerencia: Para generar el escalón alimente el circuito con un nivel de voltaje (preferentemente de 5V) utilizando un switch para generar el cambio de nivel en la entrada del circuito.

Paso 1. Implemente en protoboard el circuito pasabajas RC, y conéctelo con la NI USB 6008 como se muestra en el siguiente diagrama:

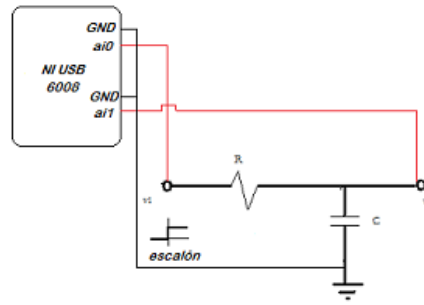


Figura 85: Pasabajas RC conectado a la NI USB 6008

donde: $R=5.6\text{ k}\Omega$, $C=100\mu\text{F}$.

Paso 2. Crear un VI en Labview que adquiera dos señales analógicas y pueda visualizarlas en labview y registre los datos siguiendo los procedimientos de las secciones 2.2.2 y 2.2.3. El diagrama a bloques sugerido para el programa es el siguiente:

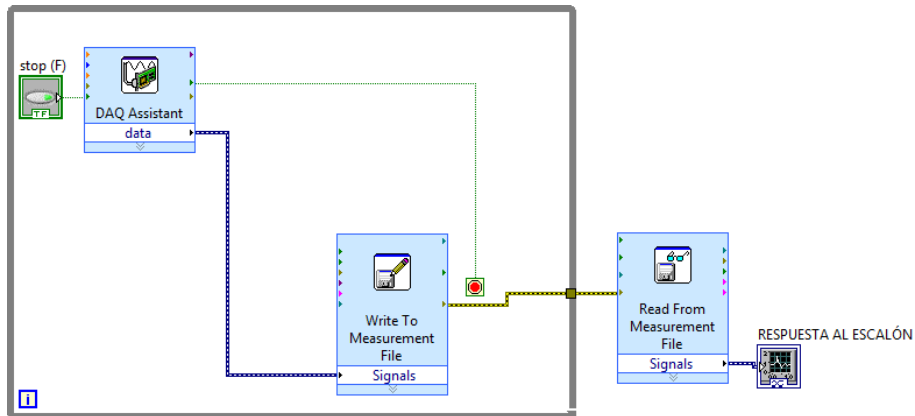


Figura 86: Diagrama a bloques en labview

Procedimiento experimental

1. Ejecute el programa desde el panel frontal.
2. Aplicar la entrada escalón de 5 V.
3. Pulse el control booleano *Stop* colocado en el panel frontal para registrar la señal adquirida y observe la respuesta al escalón del circuito RC.
4. Los resultados obtenidos en la Práctica Ia deben ser registrados en las tablas que a continuación se presentan:

Cuadro 2: Tabla resumen de Datos

Filtro Pasabajas RC	v_i	v_c	t_c	t_i	$5\tau = t_c - t_i$	$e_{ss} = 1 - \frac{v_{\infty}}{v_i}$
SIMULINK						
DAQ NI USB 6008						

5) Respuesta al escalón de un filtro pasabajas Sallen and Key utilizando la DAQ NI USB 6008.

Paso 1. Implemente en protoboard el filtro de Sallen and Key y conéctelo con la NI USB 6008 como se muestra en el siguiente diagrama:

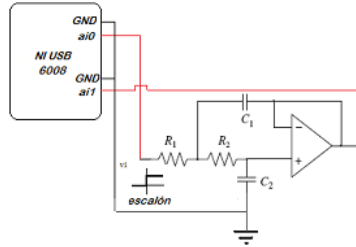


Figura 87: Conexión del Sallen and Key con la NI USB 6008

Donde:

$C_1=100\mu$; $C_2=100\mu$; $R_1=1k$; $R_2=1k$;

Paso 2. Repita el procedimiento utilizado para el filtro pasabajas RC para obtener la respuesta al escalón y registre los resultados en la siguiente tabla:

Cuadro 3: Tabla Resumen de Datos

Filtro Sallen and Key	v_i	v_p	$v(t=\infty)$	t_p	t_i	$t_p - t_i$	$e_{ss} = 1 - \frac{v_{\infty}}{v_i}$
SIMULINK							
DAQ NI USB 6008							

PRÁCTICA Ib: RESPUESTA DE UN SISTEMA A OTRAS FORMAS DE ONDA

1) Simulación en Simulink

Para el filtro RC y el filtro Sallen and Key aplique a la entrada del sistema una onda senoidal en lugar del escalón. Observe la señal en el momento de la frecuencia de corte y el comportamiento general del sistema ante esas formas de onda. Repita el procedimiento para una onda cuadrada aplicada a la entrada del sistema. La onda cuadrada se encuentra en la ruta *Simulink* | *Sources* | *Pulse Generator*; la onda senoidal en la ruta *Simulink* | *Sources* | *Sine Wave*.

2) Respuesta de un sistema utilizando la DAQ NI USB 6008

Aplique una onda senoidal en lugar del escalón para el filtro pasabajas RC y para el filtro pasabajas Sallen and Key, observe la respuesta de la señal y también observe cuando ésta llega a su frecuencia de corte. Repita el procedimiento aplicando una onda cuadrada a la entrada del circuito. Los resultados obtenidos en la Práctica Ib deben ser registrados en las tablas que a continuación se presentan:

Los resultados obtenidos deberán ser registrados en las siguientes tablas

Cuadro 4: Tabla resumen de Datos

Filtro Pasabajas RC	f_c	v_i	v_o
CALCULADO			
SIMULADO			
DAQ NI USB 6008			

Cuadro 5: Tabla resumen de Datos

Filtro Sallen and Key	f_c	v_i	v_o
SIMULADO			
DAQ NI USB 6008			

4.2. PRÁCTICA II: “RESPUESTA EN FRECUENCIA”

4.2.1. INTRODUCCIÓN

En la práctica anterior analizamos la respuesta en el tiempo de un filtro RC pasabajas y un filtro Sallen and Key pasabajas. En esta ocasión analizaremos la respuesta en frecuencia de dichos filtros, haciendo breves ajustes a los valores de las funciones de transferencia para nuevas frecuencias de corte. Utilizaremos Matlab para realizar las simulaciones pertinentes y la DAQ NI USB 6008 para analizar los datos reales de los circuitos que han de implementarse. Antes de comenzar con la práctica se recomienda leer la Sección 2.3 que trata el tema de Respuesta en Frecuencia.

4.2.2. OBJETIVOS

Objetivo general:

Analizar la respuesta en frecuencia de un filtro RC pasabajas y un filtro de Sallen and Key pasabajas, simulando en Matlab y analizando los datos reales de dichos filtros con la DAQ NI USB 6008.

Objetivos específicos:

1. Simular en Matlab la respuesta en frecuencia de un filtro pasabajas RC y un filtro Sallen and Key pasabajas, para realizar su diagrama de Bode y su diagrama de Fase.
2. Realizar la adquisición de datos de dichos filtros con la tarjeta NI USB 6008, para poder realizar los diagramas de bode y los diagrama de fase correspondientes a cada filtro.

4.2.3. MATERIALES Y EQUIPO

Materiales

Resistencias, Capacitores, opam 741.

Equipo

1 osciloscopio, 1 Generador de funciones, Fuente Simétrica de dc, Fuente Simple de dc, 1 PC con puertos USB, DAQ NI USB 6008.

4.2.4. DESARROLLO DE LA PRÁCTICA

1) SIMULACIÓN EN MATLAB DE LA RESPUESTA EN FRECUENCIA DE UN SISTEMA MECÁNICO TRASLACIONAL

Considere el sistema mecánico traslacional utilizado en la Práctica Ia. En base a ello, realice lo que a continuación se indica:

- a) Encuentre la función de transferencia del modelo de estados de la práctica Ia y realice la gráfica de Bode y el diagrama de fase dada la función de transferencia encontrada. El código de Matlab a seguir para encontrar la función de transferencia del modelo de estados es el siguiente:

```
>> syms s %declaramos variable simbólica
>> k=1; b=1; m1=1; m2=1; % asignamos valores a los parámetros del sistema
>> A=[0,0,1,0; 0,0,0,1; 0,0,-b/m1, b/m1;0,-k/m2,b/m2,-b/m2] % declaramos ma-
trix A
A =
0 0 1 0
0 0 0 1
0 0 -1 1
0 -1 1 -1
>> B=[0;0;-1/m1;0] %declaramos matriz B
B =
```

```

0
0
-1
0
>> C=[0,1,0,0] %declaramos matriz C
C =
0 1 0 0
>> D=[0] %declaramos matriz D
D =
0
>> %Una vez declaradas las matrices aplicamos expresion para encontrar función
de transferencia.
>> simplify(C*inv(s*eye(4)-A)*B+D) % expresion para hallar función de transfer-
enciA; eye (4)corresponde a la matriz identidad de orden 4.
ans =
-1/(s^3+2*s^2+s+1)

```

b) Encuentre su gráfica de bode y su diagrama de fase utilizando Matlab en base a la función de transferencia encontrada.

```

>> % Hallada la función de transferencia del sistema procedemos al código para
hallar diagramas de bode y de fase
>> H=tf(-1,[1,2,1,1]) %configuramos como función de transferencia
Transfer function:
-1
-----
s^3 + 2 s^2 + s + 1
>> bode(H) %Comando para realizar trazas de Bode y de phase.

```

2) SIMULACIÓN EN MATLAB DE LA RESPUESTA EN FRECUENCIA DE UN FILTRO RC PASABAJAS

Considere el circuito RC utilizado en la práctica Ia, ya que conocemos la función de transferencia encuentre su gráfica de Bode y su diagrama de fase siguiendo los procedimientos del ejercicio anterior.

3) SIMULACIÓN EN MATLAB DE LA RESPUESTA EN FRECUENCIA DE UN FILTRO SALLEN AND KEY PASABAJAS

En la práctica Ia se trabajó con un filtro Sallen and Key pasabajas, ahora, haremos algunas modificaciones a los valores del filtro para ello realice lo que a continuación se menciona:

a) Diseñe un filtro Sallen and Key que tenga un valor de $\omega_N = 1200$, recuerde que la función de transferencia está dada por:

$$T_{sk} = \frac{\omega_N^2}{s^2 + 2\xi\omega_N s + \omega_N^2}$$

de donde tenemos que:

$$\omega_N^2 = \frac{1}{R1 * C1 * R2 * C2}$$

$$2\xi\omega_N = \frac{R1 + R2}{R1 * R2 * C1}$$

Sugerencia: Elija un valor de $R1 + R2 = 1k$, $C1 = 1\mu$.

b) Realice el diagrama de Bode y el diagrama de fase utilizando Matlab en base a la función de transferencia que resulte.

4) RESPUESTA EN FRECUENCIA DE UN FILTRO RC UTILIZANDO LA DAQ NI USB 6008

Realizar los procedimientos que a continuación se indican:

Paso1. Realice el conexionado con la DAQ NI USB 6008 con el circuito RC pasabajas tal como se hizo en la práctica Ia y aplique una onda senoidal de 1 Vp en la entrada del circuito.

Paso2. Construya un VI en Labview para realizar la adquisición de la señal de entrada y la señal de salida.

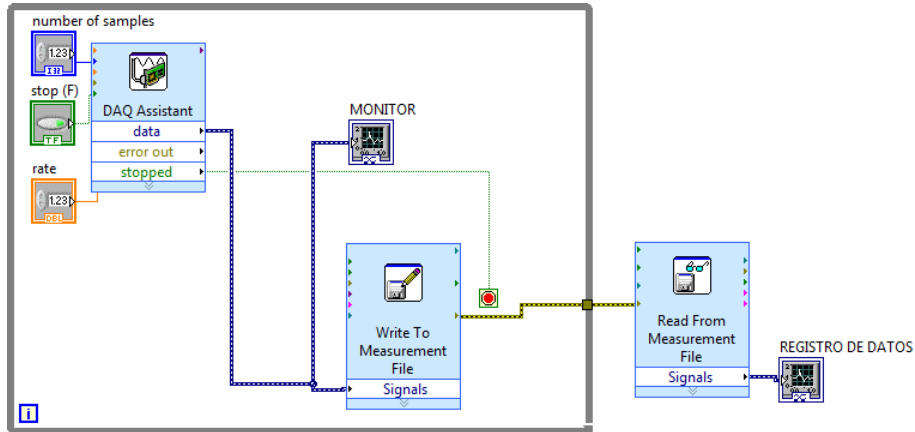


Figura 88: Diagrama a bloques para la adquisición de datos

Puede crear controles para *rate* y para *number of samples*, con el fin de tener un muestreo adecuado de la señal variando dichos parámetros desde el panel frontal para que tengan los valores adecuados. Esto se hace colocando el cursor del mouse sobre las terminales del elemento de interés (para este caso el DAQ Assistant). Después oprima clic derecho sobre los indicadores del DAQ Asistent en el diagrama de bloques y elija la ruta create | control del submenú que aparece.

Paso 3. Ejecute el programa para la adquisición de datos y varíe la frecuencia desde el valor más pequeño posible hasta valores (rango de mV) posteriores a la frecuencia de corte del filtro. Hágalo de manera gradual en pequeños intervalos de tiempo para poder registrar los valores del voltaje de salida y de la frecuencia.

Paso 4. Hecho lo anterior pulse el control boleano *stop* en el panel frontal para el DAQ Assistant para poder visualizar en el *waveform graph* de Registro de datos como fue la variación de la señal de salida con conforme cambiaba el valor de la frecuencia.

Paso 5. Llene la siguiente tabla con los valores registrados y realice la gráfica de bode correspondiente:

Cuadro 6: *Tabla de Vo vs Frecuencia*

Frecuencia (Hz)	Vo (Volts)	$20\log \frac{v_o}{v_i}$

Paso 6. Con los resultados anteriores realizaremos un ejercicio de graficación en Labview para simular señales dibujadas punto a punto. Para ello utilizaremos la función *Simulate Arbitrary signal* siguiendo la ruta *Signal Processing | Waveform Generation | Simulate Arbitrary Signal*:

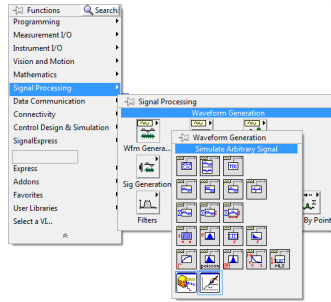


Figura 89: Selección Arbitrary Signal

Ahora realice un diagrama de bloques utilizando un visualizador *Waveform Chart*:

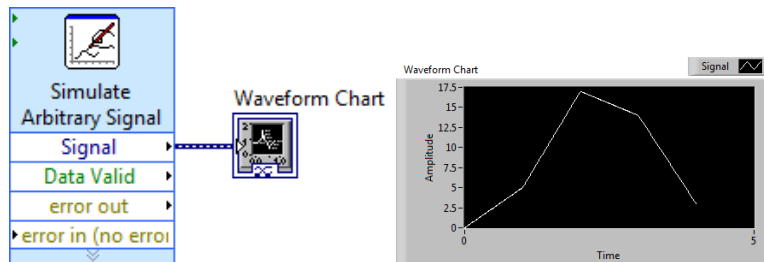


Figura 90: Vista del Diagrama de Bloques y Vista del Panel frontal

Luego, definimos los puntos la señal con los datos de la práctica realizada:

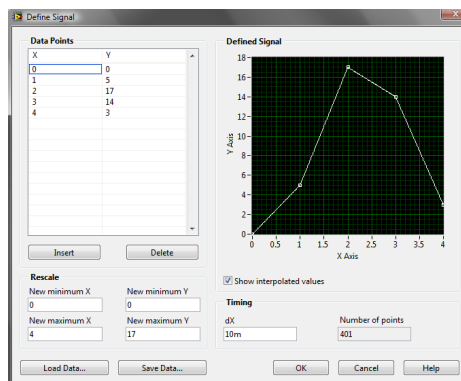


Figura 91: Definiendo Señal

Ahora, configure la señal como se muestra en la figura:

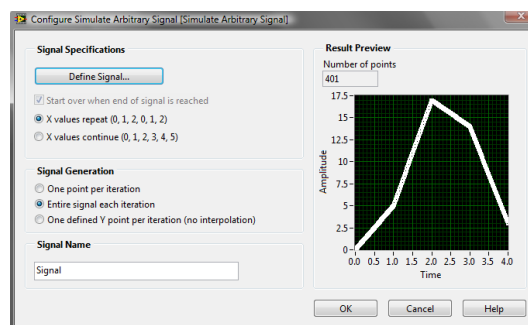


Figura 92: Configurando la Señal

Después ejecute el programa desde el panel frontal y exporte la gráfica obtenida para el gráfico de Bode.

5) RESPUESTA EN FRECUENCIA DE UN FILTRO SALLEN AND KEY UTILIZANDO LA DAQ NI USB 6008

Realice los procedimientos que se indican a continuación:

Paso 1. Conecte el filtro Sallen and Key como se indica en la Práctica Ia utilizando los valores para la simulación en Matlab de la presente práctica y aplique una onda senoidal de 1 Vp a la entrada del circuito.

Paso 2. Repita los procedimientos realizados con el filtro pasabajas RC para el filtro Sallen and Key para observar su respuesta en frecuencia.

4.3. PRÁCTICA III: “CONTROL ON-OFF”

4.3.1. INTRODUCCION

En las prácticas anteriores analizamos la respuesta en el tiempo y la respuesta en frecuencia de diversos sistemas. Sin embargo, no se realizó ninguna acción de control sobre los sistemas que fueron estudiados. En la práctica que se llevará a cabo se implementará un control ON-OFF para el sistema térmico descrito en la siguiente figura.

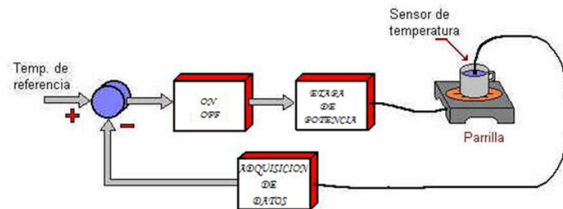


Figura 93: CONTROL ON-OFF

El sistema consiste del controlador ON-OFF digital programado en Labview, usando la DAQ NI USB 6008 para realizar la adquisición de datos y la acción de control de lazo cerrado, un sensor LM35 y una etapa de potencia para activar la parrilla eléctrica. La etapa de potencia utilizada se compone de un optoaislador MOC 3011 y un triac MAC15, la cual, dependiente de un nivel alto (5V) o un nivel bajo(0V) activará o desactivará la parrilla eléctrica según sea el caso.

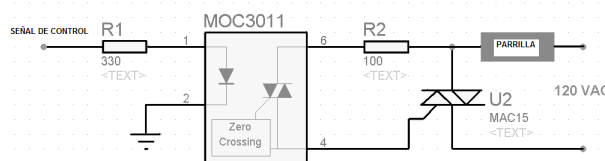


Figura 94: etapa de potencia sugerida

El modo de control ON/OFF según lo expuesto en la Sección 2.4.1. es lo más elemental y en el caso de un sistema térmico consiste en activar el mando de calentamiento cuando la temperatura está por debajo de la temperatura deseada (Set Point) y luego desactivarlo cuando la temperatura esté por arriba. Inevitablemente debido a la inercia térmica del sistema la temperatura estará continuamente fluctuando alrededor del Set Point. La inercia térmica es consecuencia del retardo en la propagación del calor en el recipiente contenedor de agua.

4.3.2. OBJETIVOS

Objetivos Generales:

Diseñar un control ON-OFF de temperatura para el sistema térmico descrito en la Sección 4.3.1 de la práctica presente, utilizando Labview para la interfaz de usuario y la DAQ NI USB 6008 para la adquisición de datos.

Objetivos específicos:

- Diseñar un instrumento virtual en Labview que cumpla la función de control ON-OFF simple.
- Diseñar un instrumento virtual en Labview que cumpla la función de control ON-OFF con histéresis.

4.3.3. MATERIALES Y EQUIPO

Materiales

1 Sensor de Temperatura LM35, 1 Optoaislador MOC 3011, 1 Triac MAC 15, 1 Recipiente con agua.

Equipo

Fuente Simple de dc, PC para la adquisición de datos con la DAQ NI USB 6008, 1 Parrilla Eléctrica.

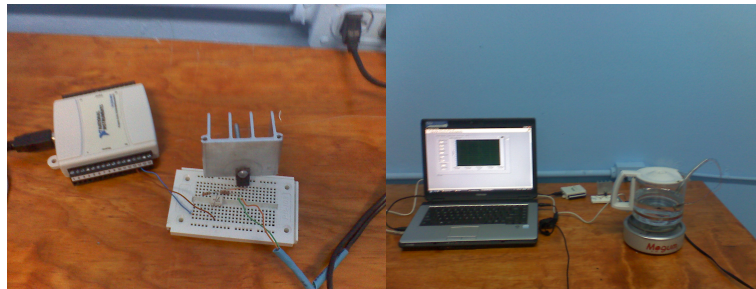


Figura 95: Materiales y equipo utilizados

4.3.4. DESARROLLO DE LA PRÁCTICA

PRACTICA IIIa: Desarrollo de un VI monitor de temperatura.

1) Desarrollo del VI Monitor de Temperatura utilizando la DAQ NI USB 6008

Acondicione el sensor de temperatura de manera que pueda utilizarlo como una sonda para medir la temperatura del agua contenida en el recipiente y conecte el LM35 a la tarjeta de adquisición de datos como se muestra en la siguiente figura:

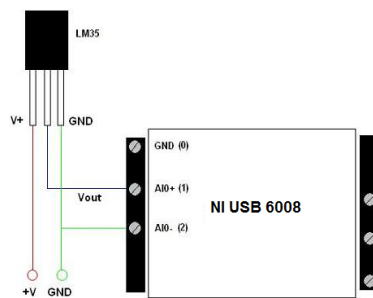


Figura 96: Conexión del LM35 con la DAQ NI USB 6008

Una vez conectado el sensor de temperatura procedemos a construir un VI en Labview de adquisición de datos para monitorear la temperatura que entrega el sensor.

Construcción del programa

El siguiente diagrama de bloques en Labview corresponde al monitor de temperatura que será utilizado en la adquisición de datos con la DAQ NI USB 6008:

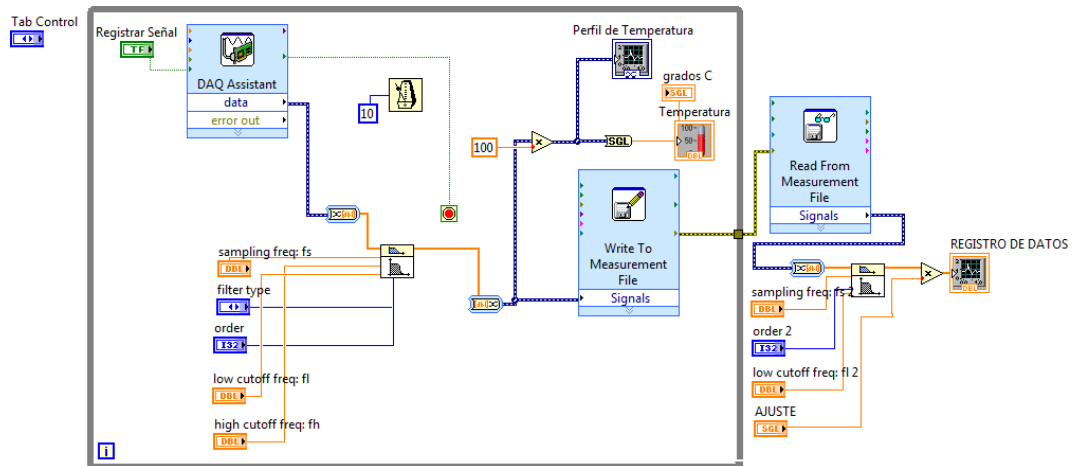


Figura 97: Diagrama a bloques del Monitor de Temperatura

Para visualizar los controles creados de la forma en que se muestran en el diagrama a bloques, pulse clic derecho de cada uno de los elementos y seleccione la opción *View as Icon* del submenú que aparece.

- Paso 1. Después de crear un nuevo Blank VI coloque en el panel frontal dos *Waveform Graph*.
- Paso 2. Estando en el panel frontal pulse clic derecho para que aparezca el menú *Controls*. Seleccione la ruta *Modern | Numeric | Thermometer* para colocar un termómetro, como se muestra en la siguiente figura:

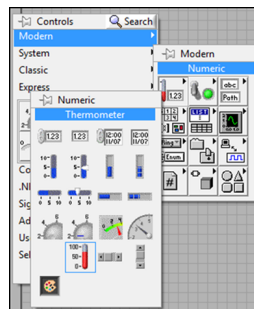


Figura 98: Selección del Termómetro

- Paso 3. En el diagrama a bloques coloque un DAQ Assistant, seleccione el canal ai0 y configúrelo para entradas diferenciales. Después coloque un *Write to Measurement File* y también un *Read From Measurement File* los cuales deben ser configurados de la forma en que se ha hecho en las prácticas anteriores y en la sección 2.2.

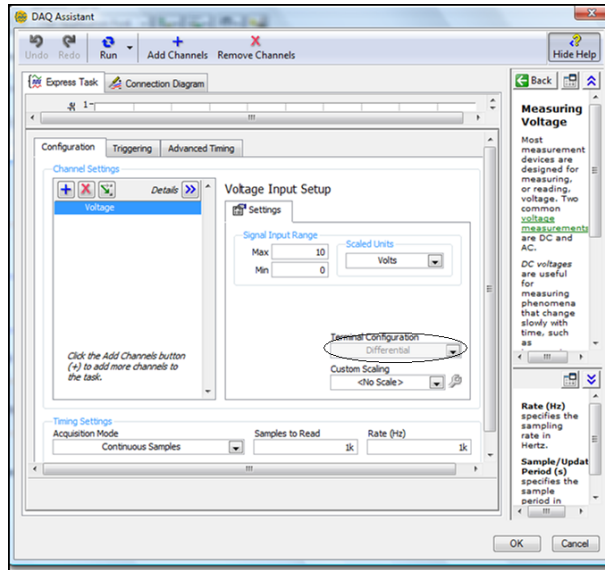


Figura 99: Configuración del DAQ Assistant para entradas diferenciales

Paso 4. En el diagrama a Bloques coloque dos filtros Butterworth pulsando clic izquierdo y seleccionando la ruta *Functions | Signal Processing | Filters | Butterworth Filter.vi*.

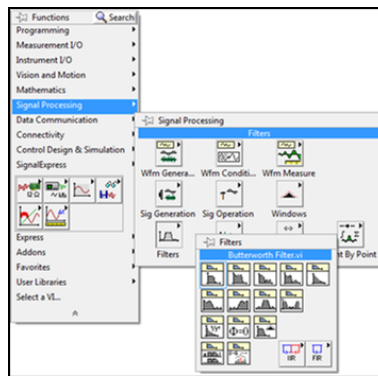


Figura 100: Selección del Filtro Butterwoth

Paso 5. Coloque dos multiplicadores en el diagrama a bloques pulsando clic derecho y seleccionando la ruta *Functions | Programming | Numeric | Multiply*. Después pulse clic derecho sobre el multiplicador y seleccione *Create | Constant* del submenú que aparece.

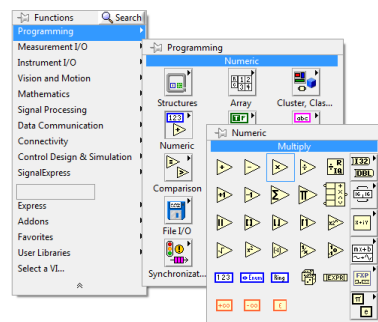


Figura 101: Selección del multiplicador

Paso 6. En el diagrama a bloques añada al programa un convertidor para tipo de datos de precisión flotante singular siguiendo la ruta *Functions | Programming | Numeric | Conversion | To Single Precision Float*.

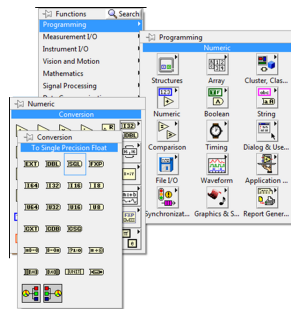


Figura 102: Selección del To Single Precision Float

Paso 7. Coloque un *Wait Until ms Multiple* siguiendo la ruta *Programming | Timing | Wait Until ms Multiple*.

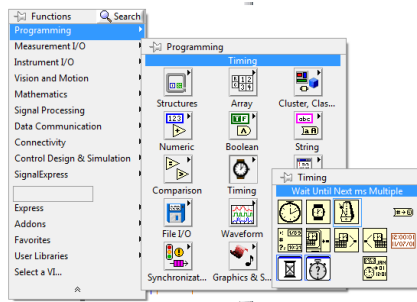


Figura 103: Selección del Wait Until ms Multiple

Paso 8. En el panel frontal coloque un contenedor *System Tab Control* pulsando clic derecho y seleccionando la ruta *Control | System | System Tab control*. Coloque también un *System Recessed Frame* seleccionando la ruta *System | Recessed Frame*.

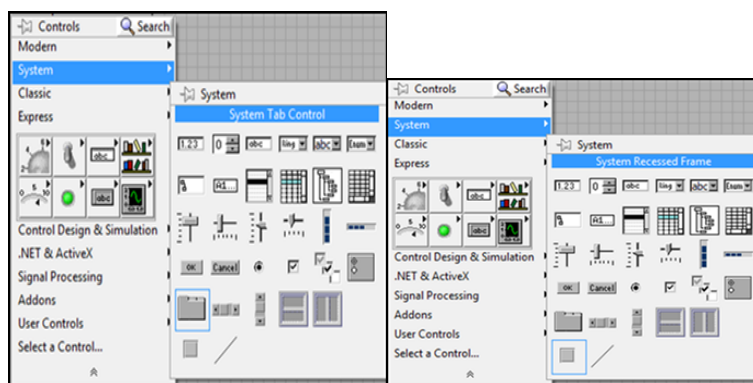


Figura 104: Selección del System Tab Control y System Recessed Frame

Paso 9. Hecho lo anterior coloque los componentes que se encuentran en el panel frontal de manera que la vista sea la siguiente:

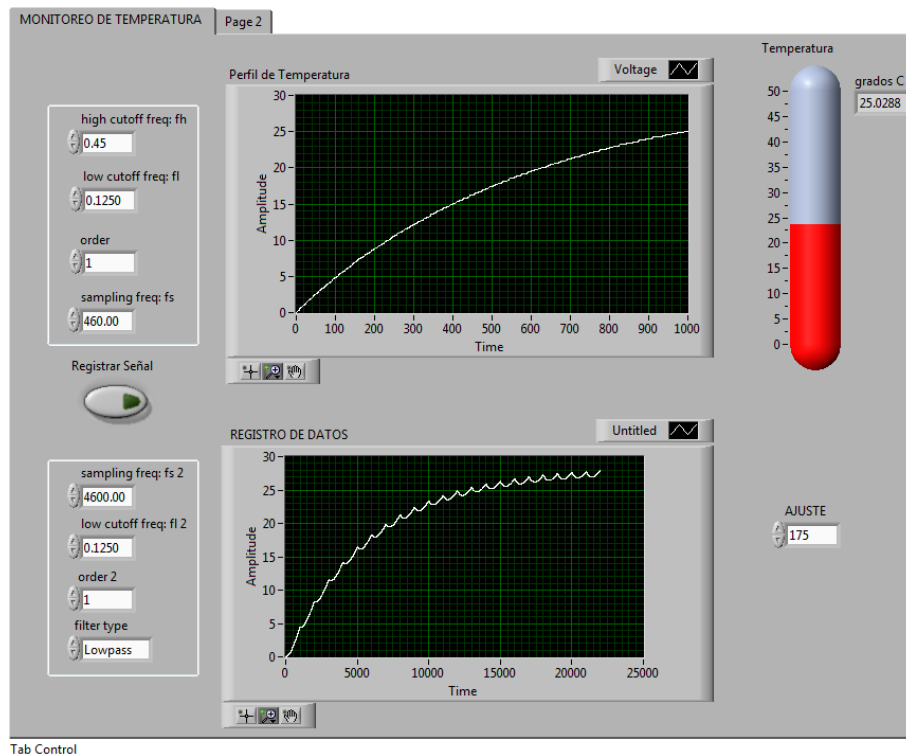


Figura 105: Panel frontal del Programa de Monitoreo de Temperatura

Ahora el programa está listo para ser ejecutado.

Ejecución del programa:

1. Ejecute el programa desde el panel frontal.
2. Fije los parámetros *Sampling freq.* y *order* del Perfil de temperatura de manera que las lecturas coincidan con los voltajes que entrega el LM35 (verificar con un voltímetro)
3. Fije los parámetros *Sampling freq.*, *order* y AJUSTE ahora para el *Waveform Graph* correspondiente al *REGISTRO DE DATOS* de manera que las lecturas coincidan con el perfil de temperatura.
4. Oprima el control boleano *Registrar Señal*. Si los valores obtenidos en el *REGISTRO DE DATOS* no coinciden con los valores de perfil de temperatura vuelva a ajustar los parámetros correspondientes.

PRÁCTICA IIIb: Control ON-OFF de Temperatura.

1) Realización del Control ON-OFF en Labview utilizando la DAQ NI USB 6008.

Conexione la DAQ NI USB 6008 como a continuación se indica:

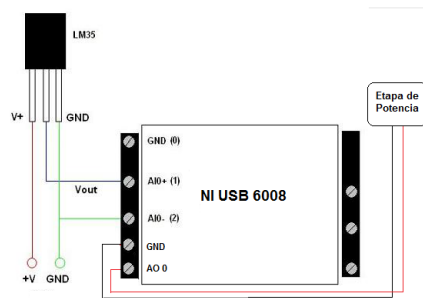


Figura 106: conexión de la DAQ NI USB 6008

Construcción del programa

Al programa anteriormente realizado para monitorear la temperatura de nuestro sistema térmico lo acondicionaremos para que cumpla la función de un control on-off con los datos obtenidos a través de la DAQ NI USB 6008. El diagrama de bloques del Labview es el siguiente:

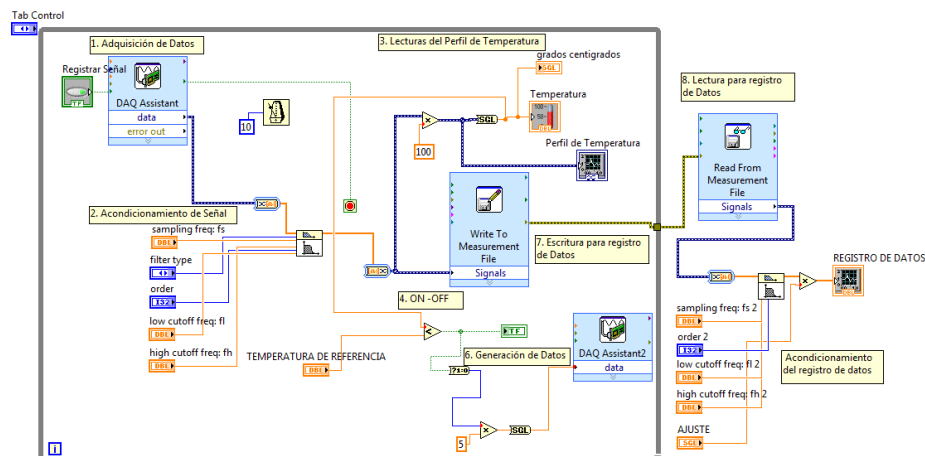


Figura 107: Diagrama a Bloques en Labview del control on-off de temperatura

Paso 1. Coloque otro DAQ Assistant para utilizarlo en la generación de niveles de voltaje de salida y configúrelo como se hizo en la Sección 2.3

Paso 2. Coloque un comparador siguiendo la ruta pulsando clic derecho y siguiendo la ruta *Functions | Programming | Comparison | Less?* y añada un indicador en la salida del mismo.

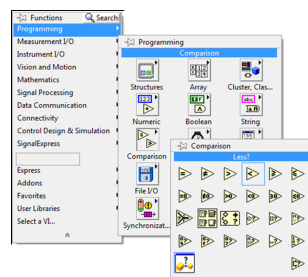


Figura 108: Selección del less?

Paso 3. En el panel frontal agregue un control deslizante pulsando click derecho y siguiendo la ruta *Controls* | *Classic Numeric* | *Vertical Fill Slide*.

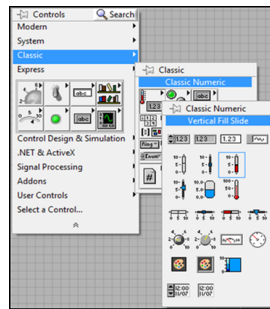


Figura 109: Selección del Vertical Fill Slide

Paso 4. En el diagrama a bloques añada al programa un convertidor de datos booleano a numérico siguiendo la ruta *Functions* | *Programming* | *Numeric* | *Conversion* | *Boolean To (0,1)*

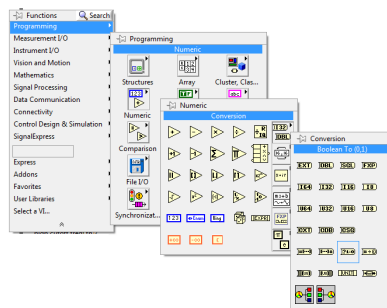


Figura 110: Selección de Boolean To (0,1)

El panel frontal resultante deberá tener el siguiente aspecto:

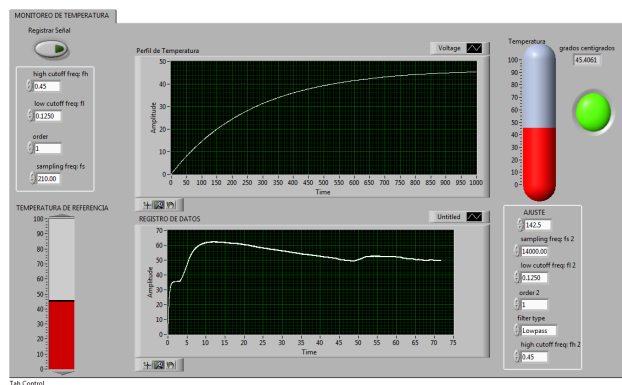


Figura 111: Panel frontal del control on-off de temperatura

Funcionamiento del programa:

Por medio del DAQ Assistant adquirimos la señal que entrega el sensor LM35. Esta señal es acondicionada por medio del filtro Butterworth para eliminar tendencias no deseadas. Una vez que la señal ha sido acondicionada será amplificada 100 veces para poder visualizarla en °C en el panel frontal, recordemos que el sensor LM35 entrega 10mV/°C. Luego, la señal es comparada con la referencia para determinar un valor booleano de falso o verdadero, el cual es convertido a un dato numérico para la generación de un nivel de voltaje por medio del DAQ Assistant 2.

Procedimiento experimental

Paso 1. Ejecute el programa desde el panel frontal.

Paso 2. Fije la referencia de temperatura.

Paso 3. Ajuste los controles para la frecuencia de muestreo de los filtros Butterworth para una correcta lectura del perfil de temperatura

Paso 4. Ajuste los controles para el registro de datos de manera que estos coincidan con la lectura del perfil de temperatura.

Paso 5. Conecte la parrilla eléctrica para observar el comportamiento del sistema y registre la señal de temperatura después de haber transcurrido un lapso de tiempo suficiente para que se observe el cambio de estado de la parrilla. Registre los datos en una tabla de temperatura contra tiempo de acuerdo a los resultados obtenidos, recuerde exportar los resultados obtenidos en la adquisición de datos.

Cuadro 7: Tabla resumen de datos

Tiempo (seg)	Temperatura	Estado

PRÁCTICA IIIc. Control ON-OFF con histéresis

Una de las formas de realizar un control ON-OFF con histéresis es utilizando un elemento de memoria para tener una zona muerta. Una forma sencilla de hacerlo es por medio de un flip flop que conste de un arreglo de compuertas lógicas. En labview es factible utilizar compuertas lógicas, tan solo debemos de seguir la ruta Programming\Boolean\... y seleccionar la compuerta o elemento que mas nos convenga. En el siguiente diagrama a bloques se muestra un control ON-OFF con histéresis implementado en labview. Se deja como ejercicio localizar los elementos que hayan sido agregados respecto al programa del control ON-OFF de la Práctica IIIb.

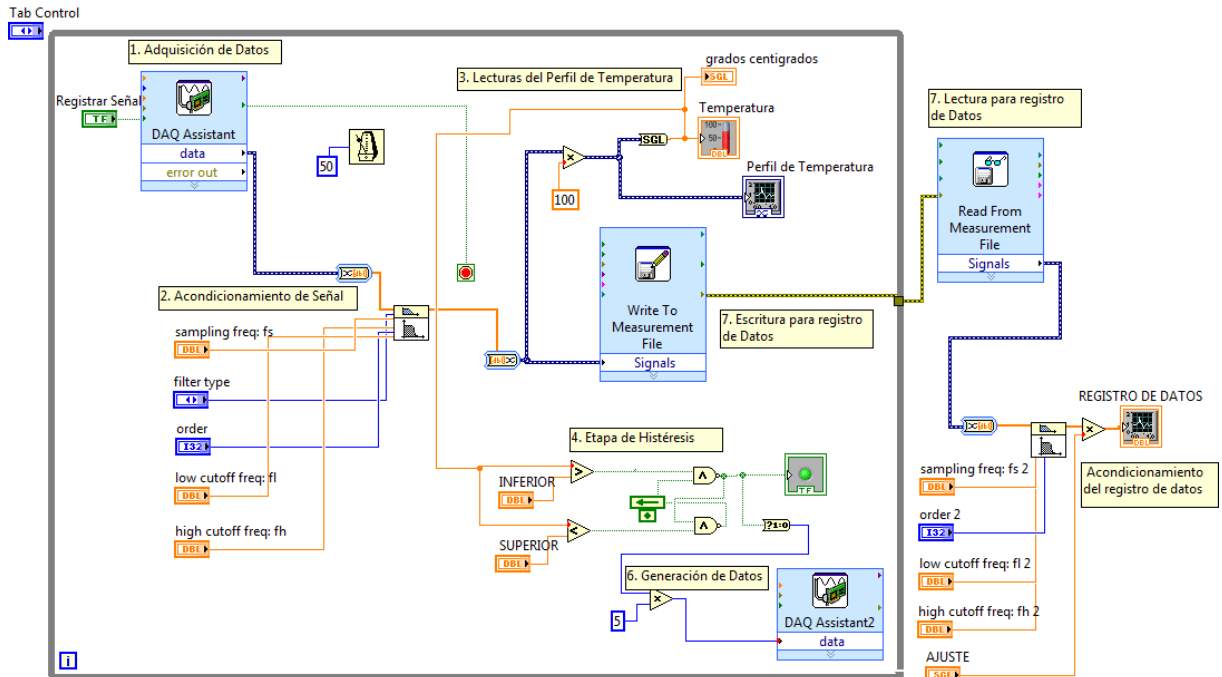


Figura 112: Diagrama a bloques de Control ON-OFF con histéresis

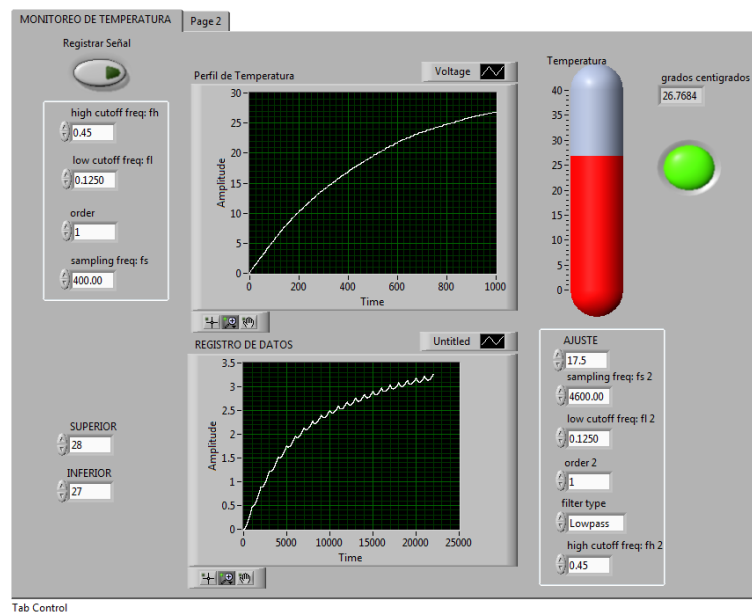


Figura 113: Panel frontal de Control ON-OFF con histéresis

Procedimiento experimental

Paso 1. Fije las referencias inferior y superior.

Paso 2. Ajuste los parámetros del filtro buterrworth para acondicionar las lecturas del perfil de temperatura.

Paso 3. Ajuste los parámetros para el registro de datos de manera que sean los mismos que la lectura del perfil de temperatura.

Paso 4. Encienda la parrilla y observe el comportamiento debido a la histéresis en el control de temperatura.

Cuadro 8: Tabla resumen de datos

Tiempo	Temperatura	Estado

4.4. PRÁCTICA IV: “PWM UTILIZANDO LA DAQ NI USB 6008”

4.4.1. INTRODUCCIÓN

La presenta práctica sobre PWM (Siglas en Inglés de Modulación por ancho de Pulso) ya que es una parte fundamental para aplicaciones en donde se utiliza el control proporcional. Los parámetros fundamentales del PWM son el periodo (T) y el ciclo de trabajo (D). El ciclo de trabajo indica el tiempo que la función vale uno respecto al tiempo total (el periodo). La figura siguiente muestra tres ciclos de trabajo distintos:

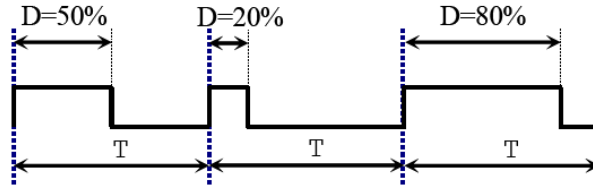


Figura 114: Ciclos de trabajo distintos para una salida PWM

Podemos observar que el periodo del PWM se mantiene constante, y lo que cambia es el tiempo en que la señal se mantiene a uno respecto al periodo total. El PWM puede realizarse apartir de diversas estrategias de diseño, entre las cuales podemos mencionar la modulación por medio de la generación de un tren de pulsos, por medio de comparadores utilizando amplificadores operacionales o por medio de microcontroladores pic, etc. Sin embargo, la modulación por ancho de pulso puede llevarse a acabo utilizando la herramienta de programación Labview gracias a las aplicaciones existentes de adquisición y generación de datos. En nuestro caso utilizaremos la DAQ NI USB 6008 ya que cuenta con salidas digitales, mediante las cuales podemos generar un tren de pulsos que será utilizado para realizar el PWM.

4.4.2. OBJETIVO

Desarrollar una interfaz en Labview que nos permita realizar la modulación por ancho de pulso, siendo capaces de controlar como mínimo los parámetros de Tiempo de Ciclo y Ciclo de Trabajo, para generar una salida PWM por medio de la DAQ NI USB 6008 que sea utilizable para aplicaciones posteriores.

4.4.3. EQUIPO

DAQ NI USB 6008, 1 Osciloscopio.

4.4.4. DESARROLLO DE LA PRÁCTICA

1) Simulación de PWM en LABVIEW

Antes de realizar los programas de PWM utilizando los puertos de salida de la DAQ NI USB 6008, es conveniente realizar el programa que realice el cálculo de parámetros y genere el algoritmo PWM que se ejecutará posteriormente para la generación de datos. Para ello realizaremos el conexionado del siguiente diagrama a bloques:

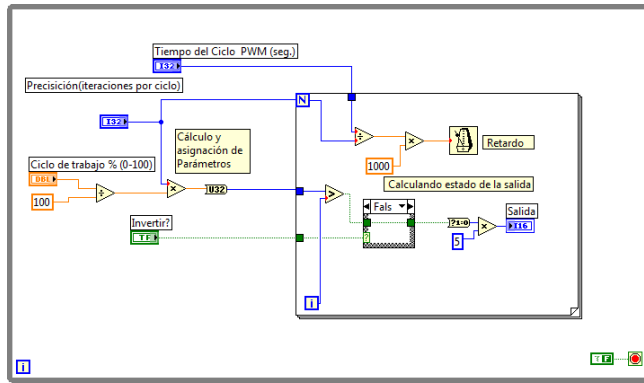
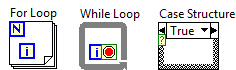


Figura 115: Programa PWM

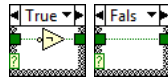
Construcción del programa

Para construir el diagrama de bloques anterior siga los siguientes pasos:

Paso 1. Colocar las estructuras *For Loop*, *While Loop* y *Case* como se muestra en el diagrama a bloques que se encuentran en la ruta *Functions | Programming | Estructures | ...*



Paso 2. Dentro de la estructura *Case* coloque un inversor para el caso verdadero; para el caso falso únicamente conecte los puntos que se encuentran en el margen de la estructura.



Paso 3. Ubique los demás componentes utilizados y conéxionalos como se muestra en el diagrama de bloques.



Paso 4. Asegurese que de crear los controles, constantes e indicadores necesarios de manera que el panel frontal tenga la siguiente vista:

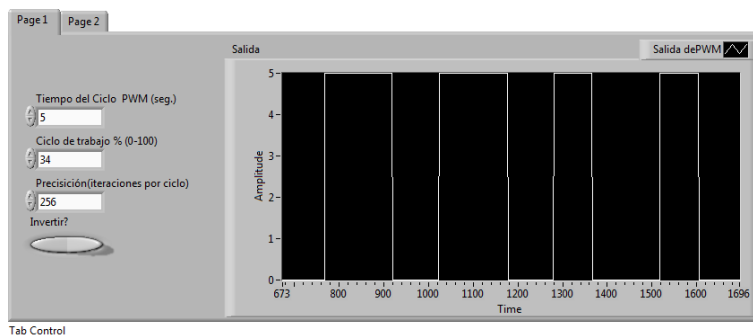


Figura 116: Vista del panel frontal

Paso 5. Ejecute el programa y varíe cada uno de los parámetros para observar como varía la salida del PWM.

Ejecución del programa

Para ejecutar el programa y realizar la acción PWM realice lo siguiente:

1. Establezca el tiempo del ciclo PWM.
2. Establezca el ciclo de trabajo.
3. Fije la precisión (iteraciones por ciclo).
4. Si desea invertir la salida pulse el control booleano Invertir?.

2) Realización del PWM utilizando una salida analógica de la DAQ NI USB 6008 mediante el DAQ Assistant utilizando salidas analógicas.

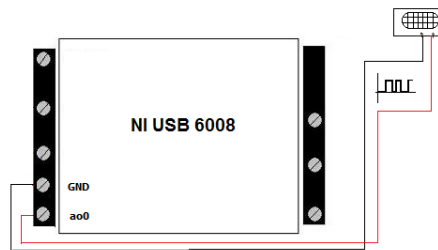


Figura 117: Diagrama de conexionado del pwm con salida analógica

Al programa de la simulación anterior le añadiremos el DAQ Assistant para configurarlo para utilizar una de las salidas analógicas de la DAQ NI USB 6008. El diagrama a bloques es el siguiente:

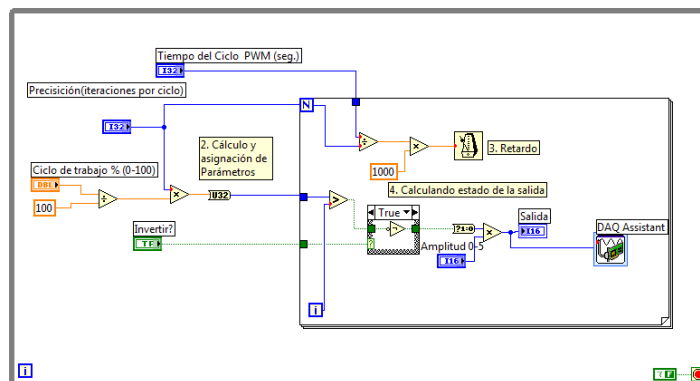


Figura 118: PWM con el DAQ Assistant

Lo único distinto en el panel frontal será la aparición de un control de amplitud.

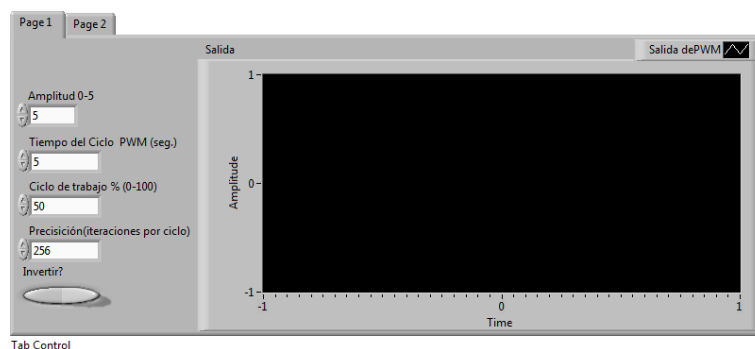


Figura 119: Vista del panel frontal

El modo de ejecución del programa es el mismo que para el programa anterior

3) Realización del PWM utilizando una línea digital de salida con la DAQ NI USB 6008

El conexionado de la tarjeta NI USB 6008 para ésta práctica es el siguiente:

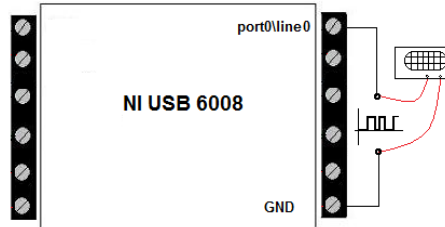


Figura 120: Conexionado para verificar PWM

Realice el diagrama a bloques del VI para PWM que se muestra en la siguiente figura.

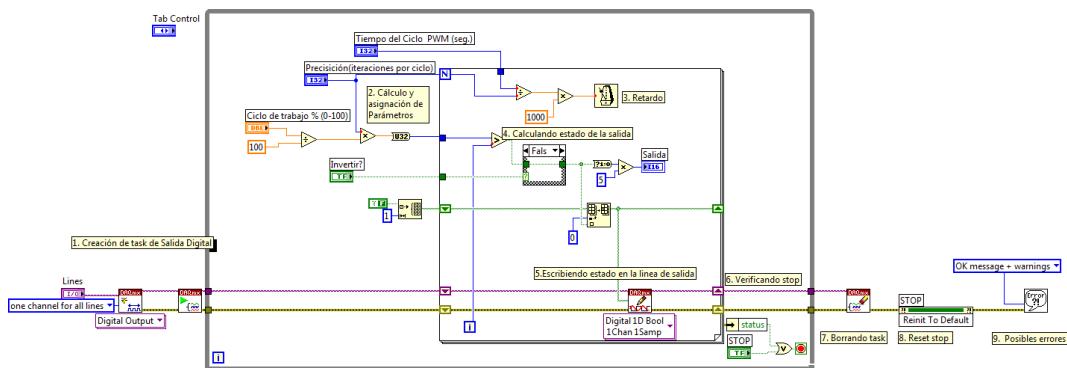


Figura 121: Diagrama a bloques del VI PWM

En el diagrama de bloques se observa que respecto al programa de PWM realizado anteriormente, se incluye una línea digital de salida para la generación de PWM utilizando la DAQ NI USB 6008. Para ello es necesario utilizar la paleta de adquisición de datos, la cual podemos ubicar siguiendo la ruta *Measurement I/O DAQmx-Data Acquisition* como se muestra en la siguiente figura:

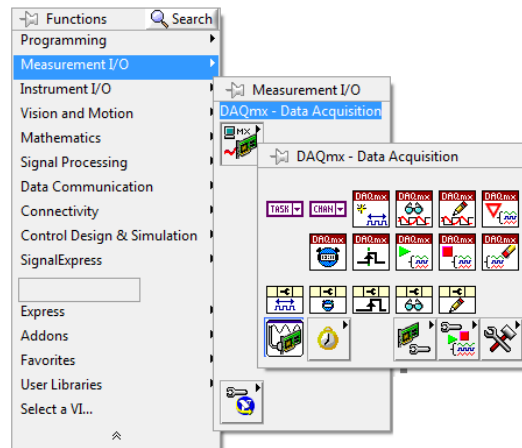
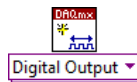


Figura 122: Ubicación de la paleta de adquisición de datos

Paso 1. Crear una línea de Salida Digital como se muestra en el diagrama a bloques utilizando las siguientes funciones de la paleta de adquisición de datos:

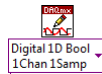
- *DAQmx Create Channel* para crear un canal de salida.



- *DAQmx Start task* para comenzar la tarea de generación de la salida digital



- *DAQmx Write* para escribir los datos de la señal pwm en la línea de salida digital



- *DAQmx Clear Task* para limpiar los datos de la señal de salida digital.



Recuerde crear las constantes, indicadores y controles pertinentes tal y como se muestran en el diagrama a bloques.

Paso 2. Coloque los siguientes elementos para reemplazar los datos de salida booleana del inversor dentro de la estructura case por arrays (para que puedan ser escritos en la línea de salida):

- *Replace Array Subset*, ubicado en *Functions | Programming | Array | Replace Array Subset*.



- *Initialize Array*, ubicado en *Functions | Programming | Array | Initialize Array*.



Es indispensable guardar en el ciclo for las salidas y entradas de estos elementos mediante el uso del *Add Shift Register*, el cual se ubica pulsando clic derecho sobre los puntos de conexión del ciclo y seleccionando la opción *Add Shift Register* del submenú que aparece.

Paso 5. Una vez realizado lo anterior asegúrese que el panel frontal tenga el siguiente aspecto:

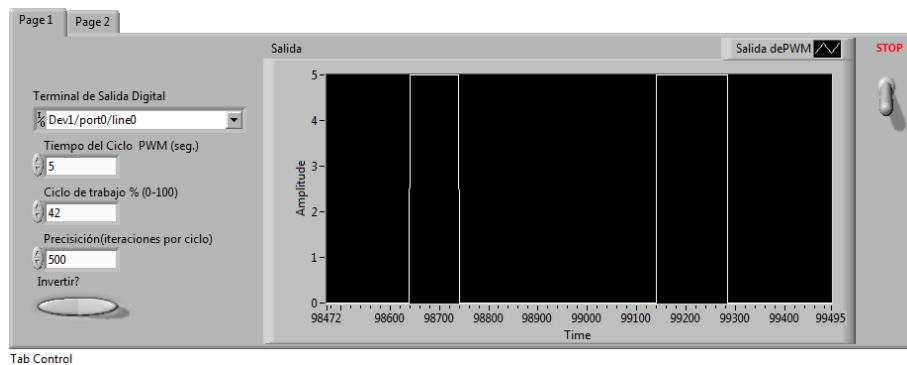


Figura 123: Panel frontal PWM

Ejecución del programa

Para ejecutar el programa y realizar la acción PWM realice lo siguiente:

1. Establezca el tiempo del ciclo PWM.
2. Seleccione la terminal de salida digital de la tarjeta port0\line0.
3. Establezca el ciclo de trabajo.
4. Fije la precisión (iteraciones por ciclo)
5. Si desea invertir la señal pulse el control booleano Invertir?
6. Posteriormente, compruebe que la salida visualizada en el *Waveform Chart* coincida con la salida que se observa en el osciloscopio y/o realice los ajustes necesarios.

4.5. PRÁCTICA V. “CARACTERIZACIÓN DE UN SISTEMA TÉRMICO POR EL MÉTODO DE LA CURVA DE REACCIÓN”.

4.5.1. INTRODUCCIÓN

En la presente práctica se realizará la caracterización del sistema térmico descrito en la práctica III, para ello utilizaremos el programa de monitoreo de temperatura construido anteriormente y el programa de PWM. Para caracterizar el sistema se escogió el método de la curva de reacción, ya que el sistema térmico descrito presenta características aproximadas a las de un sistema de primer orden con tiempo muerto como el dado por la siguiente expresión:

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

donde K es la constante de ganancia del proceso, τ es la constante de tiempo y L es el tiempo de retardo. El método de la curva de reacción consiste de los siguientes pasos:

- Permitir que el proceso alcance estado estacionario.
- Introducir un cambio tipo escalón en la variable de entrada.
- Recolectar datos de la entrada y respuesta de salida hasta que el proceso nuevamente alcance estado estacionario.
- Realizar el calculo gráfico de los parámetros a partir de la curva de reacción.

4.5.2. OBJETIVO DE LA PRÁCTICA

Caracterizar el sistema térmico por el método de curva de reacción para la utilización posterior de los parámetros obtenidos en la sintonización de un control P,PI, PD y PID.

4.5.3. MATERIALES Y EQUIPO

Materiales

1 Sensor de Temperatura LM35, 1 Optoaislador MOC 3011, 1 Triac MAC 15.

Equipo

Fuente Simple de dc, PC para la adquisición de datos con la DAQ NI USB 6008, 1 Parrilla Eléctrica.

4.5.4. DESARROLLO DE LA PRÁCTICA

Conexión de la DAQ NI USB 6008

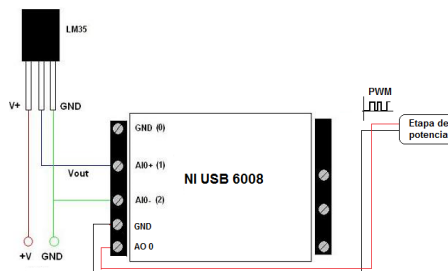


Figura 124: Conexión de la DAQ NI USB 6008

Construcción del programa en Labview

El siguiente diagrama a bloques corresponde al programa en Labview para el programa que utilizaremos en la caracterización del sistema térmico utilizado.

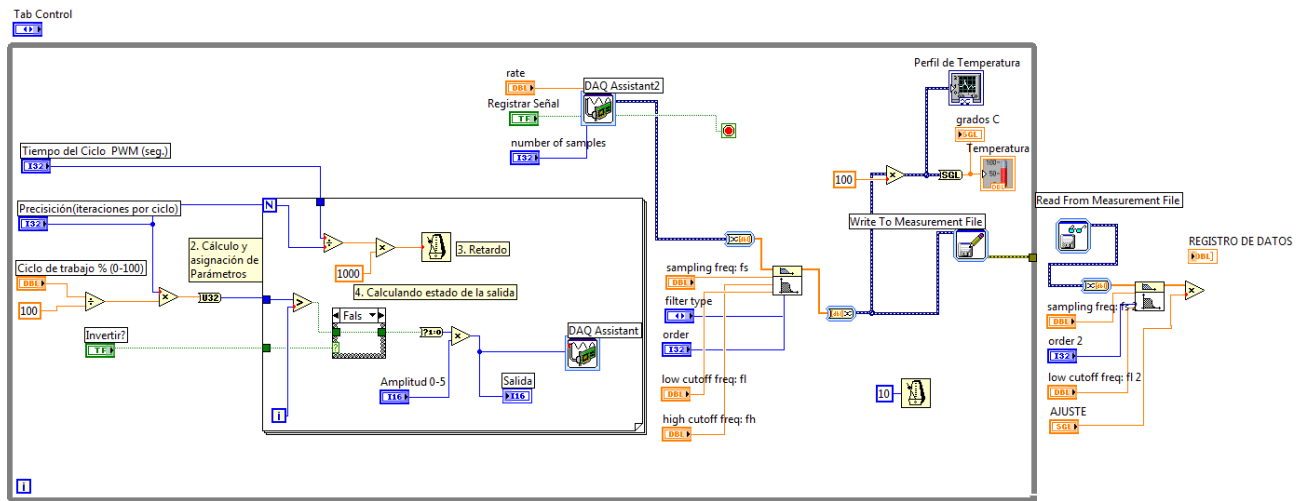


Figura 125: Diagrama de Bloques

El programa construido consta de una etapa de monitoreo y una etapa de PWM. El panel frontal consta de dos páginas en el mismo contenedor. Si seleccionamos la primera pestaña veremos el perfil de temperatura y Registro de Datos.

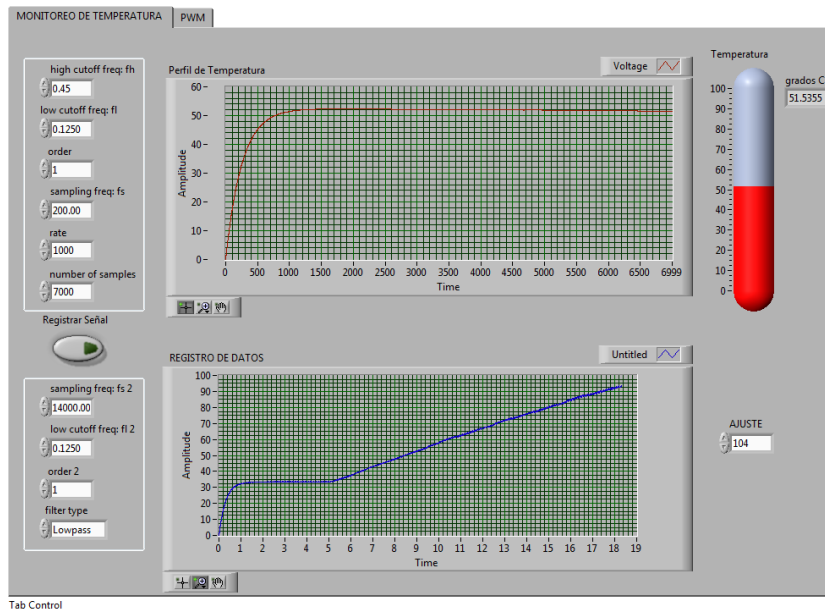


Figura 126: Panel frontal. MONITOREO DE TEMPERATURA

Si seleccionamos la pestaña PWM, veremos los controles y un visualizador para el comportamiento del PWM:

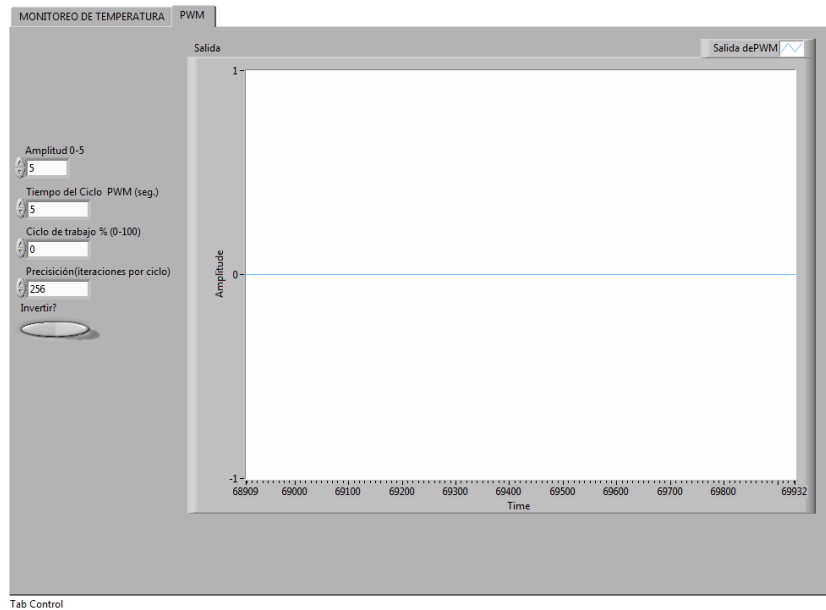


Figura 127: Panel frontal. PWM

Procedimiento experimental

Paso 1. Ejecute el programa para monitoreo de la temperatura.

Paso 2. Ajuste los parámetros del filtro Butterworth para el perfil de Temperatura

Paso 3. Ajuste los parámetros del filtro Butterworth para el registro de datos de manera que las lecturas coincidan con las del Perfil de Temperatura.

Paso 4. Deje de ejecutar el programa para iniciar una nueva lectura con las herramientas de medición ajustadas.

Paso 5. Configure el PWM para un 20% y espere a que la lectura del sensor quede en estado estacionario.

Paso 6. Inmediatamente después conecte la parrilla eléctrica y configure el PWM para un 100% de manera que el agua hierva para que el sistema alcance el estado estacionario de nueva cuenta. Pulse el botón de registro de señal y observe los siguientes parámetros para el nuevo estado estacionario.

- t_{28} = tiempo correspondiente al 28.3% del cambio producido por una entrada escalón.
- t_{63} = tiempo correspondiente al 63.2% del cambio producido por una entrada escalón.
- y_{ss} = Valor final o valor en estado estacionario.
- y_{63} = 63.2% del cambio total esperado.
- y_{28} = 28.3% del cambio total esperado.

Nota: Recuerde que los parámetros anteriores se toman a partir de que se aplica la nueva entrada tipo escalón al conectar la parrilla.

Paso 7. Teniendo los datos anteriores encuentre la función de transferencia aproximada del sistema.

4.6. PRÁCTICA VI: “CONTROL PROPORCIONAL (P)”

4.6.1. INTRODUCCIÓN

La práctica presente trata de la implementación de un Control P en Labview para el sistema térmico correspondiente a la Práctica III utilizando la DAQ NI USB 6008. El sistema deberá ser sintonizado utilizando de acuerdo al método de la curva de reacción mostrado en la sección 2.5. La salida del control proporcional irá a un PWM para controlar el ciclo de trabajo de la parrilla eléctrica de acuerdo al tiempo de ciclo que sea establecido. Recordemos que un control proporcional genera una salida proporcional al error, es decir, dependiendo la cantidad de error será la acción de control ejercida. El sistema térmico a controlar es el siguiente:

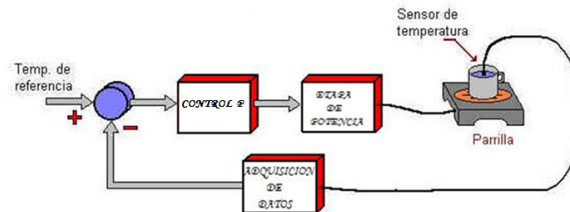


Figura 128: Sistema Térmico para Control Proporcional

El PWM controlado tiene la función de generar un tiempo proporcional de activación y desactivación de la parrilla eléctrica de acuerdo a la base de tiempo que sea fijada. De esta manera la temperatura del agua irá alcanzando el Set Point dependiendo del tiempo proporcional generado por el PWM. Suponiendo que tenemos una base de tiempo de 5 seg (que es un tiempo adecuado para un sistema como el nuestro) el encendido y apagado de la parrilla quedaría descrito por la siguiente tabla:

Cuadro 9: Tiempo proporcional

PWM	Apagado	Encendido
20 %	4 seg	1 seg
40 %	3 seg	2 seg
60 %	2 seg	3 seg
80 %	1 seg	4 seg
100 %	0 seg	5 seg

Para fijar la ganancia proporcional utilice los valores de la sintonización por el método de la curva de reacción obtenidos en la práctica anterior.

4.6.2. OBJETIVOS

Objetivo General

Desarrollar un control proporcional en Labview utilizando la DAQ NI USB 6008 para el sistema térmico descrito en la práctica anterior.

Objetivos Específicos

- Realizar la simulación en Simulink de Matlab.
- Implementar el control proporcional como un VI en Labview.

4.6.3. MATERIALES Y EQUIPO

Materiales

1 Sensor de Temperatura LM35, 1 Optoaislador MOC 3011, 1 Triac MAC 15, 1 Recipiente con agua.

Equipo

Fuente Simple de dc, PC para la adquisición de datos con la DAQ NI USB 6008, 1 Parrilla Eléctrica.

4.6.4. DESARROLLO DE LA PRÁCTICA

1) Simulación de un control P en Simulink de Matlab.

Construya en Simulink de Matlab el siguiente diagrama a bloques:

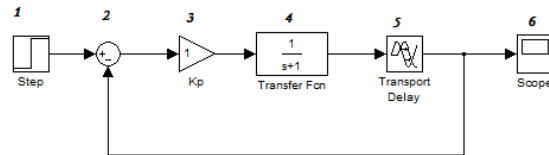


Figura 129: Diagrama a bloques PID en el dominio de Laplace

La ubicación de los componentes es la siguiente:

1. Simulink library browser \ Sources \ Step.
2. Simulink library browser \ Commonly used blocks \ Sum.
3. Simulink library browser \ Commonly used blocks \ Gain.
4. Simulink library browser \ Continuos \ Transfer Function.
5. Simulink library browser \ Continuos \ Transport Delay.
6. Simulink library browser \ Sinks \ Scope.

2) Construcción del programa en Labview para la acción de control P.

Paso 1. Realice el siguiente diagrama para la acción proporcional en Labview:

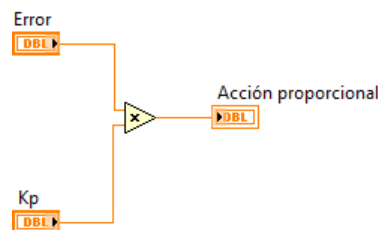


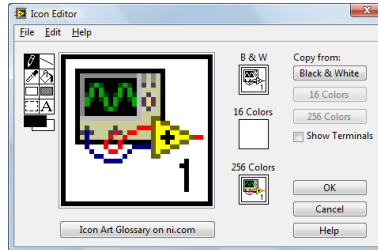
Figura 130: Acción proporcional

Paso 2. Del diagrama a bloques anterior edite un ícono para que pueda utilizarlo posteriormente como un sub VI para aplicaciones posteriores con el siguiente procedimiento:

Pulse doble clic izquierdo sobre el ícono ubicado en la parte superior derecha de la ventana de labview



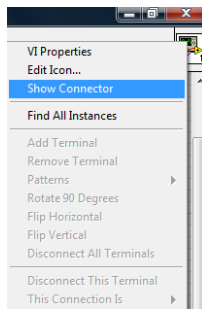
Hecho lo anterior se abrirá un editor de íconos:



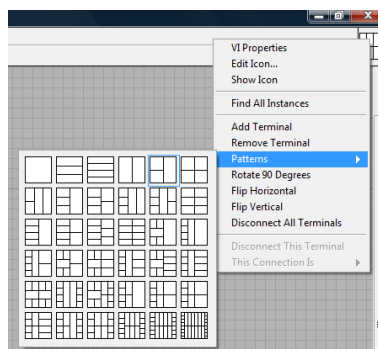
Edite el ícono de manera que tenga el siguiente aspecto:



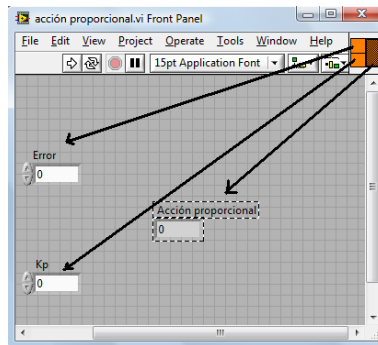
En el panel frontal pulse clic derecho en el ícono que se muestra y seleccione la opción show conector.



Luego pulse clic derecho de nuevo y seleccione la opción patterns dentro del mismo submenú, y seleccione la configuración de acuerdo a las terminales necesarias (3 en este caso).



Después, asigne los controles e indicadores a cada una de las terminales, pulsando secuencialmente el conector y el icono del panel frontal que se desea asignar a dicho conector:



Guarde los cambios realizados para poder utilizar el VI creado en posteriores aplicaciones en la ruta o dirección que crea conveniente.

Paso 3. Realice el siguiente diagrama a bloques que incluye PWM a la salida de control proporcional.

Nota: A manera de sugerencia, compruebe el correcto funcionamiento del algoritmo de control proporcional de manera similar al procedimiento realizado en el Anexo III.

3) Construyendo el VI de control de temperatura con la DAQ NI USB 6008.

El diagrama de bloques que corresponde al control P con salida PWM es el siguiente:

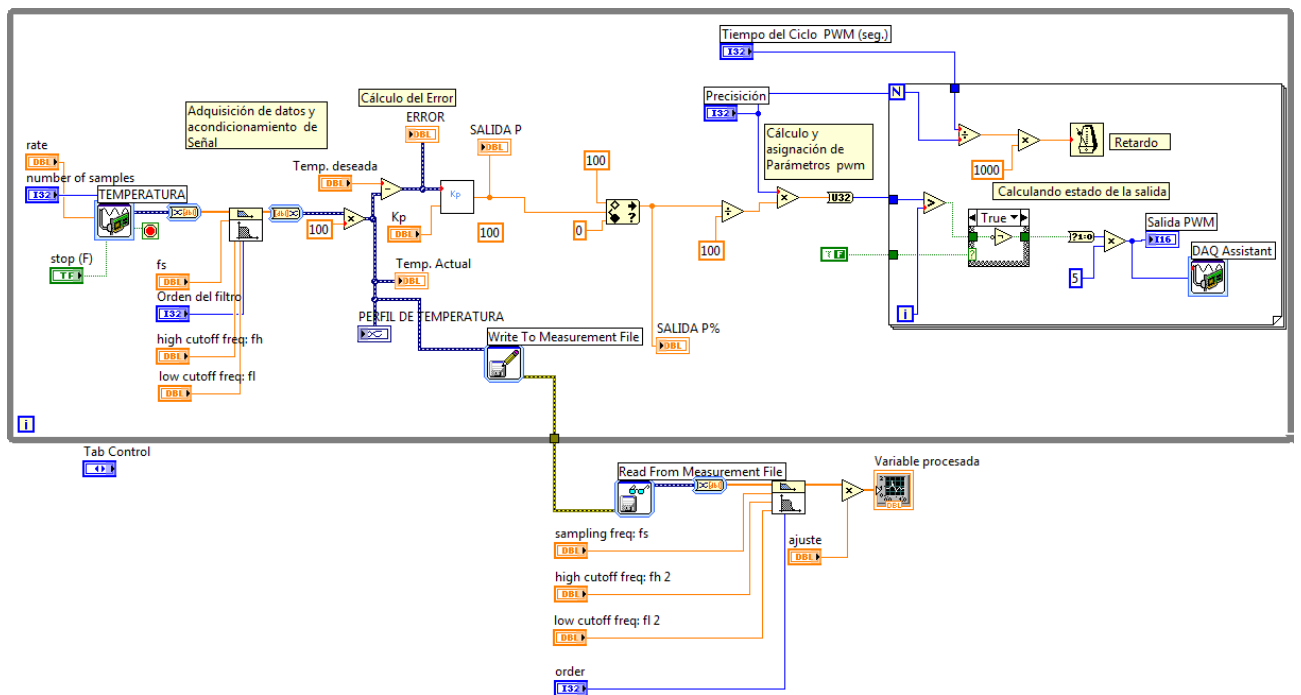


Figura 131: Diagrama a bloques Control P de Temperatura

El panel frontal tiene cuatro páginas distintas en un mismo contenedor, que pueden ser seleccionados por medio de las pestañas de encabezado de cada uno de ellos. El panel de perfil de temperatura para el control proporcional es el siguiente:

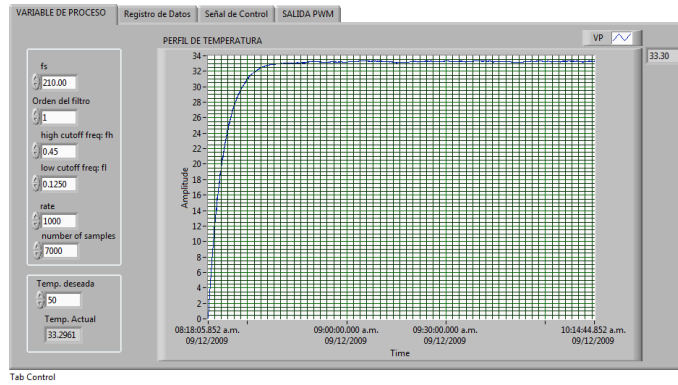


Figura 132: Panel frontal pestaña de perfil de temperatura

Seleccionando la pestaña de Registro de Datos veremos en el panel frontal lo que a continuación se muestra:

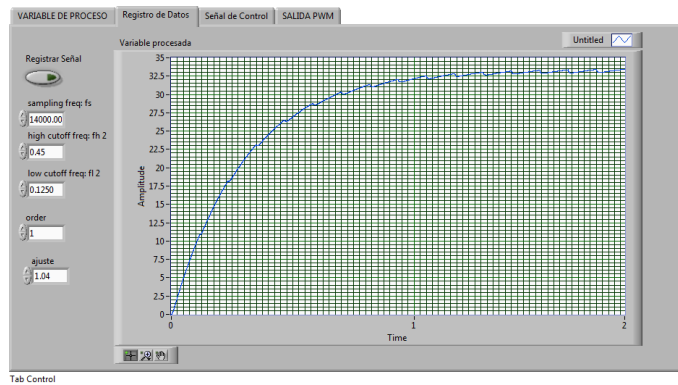


Figura 133: Panel frontal. Pestaña de Registro de Señal

Para la señal de control:

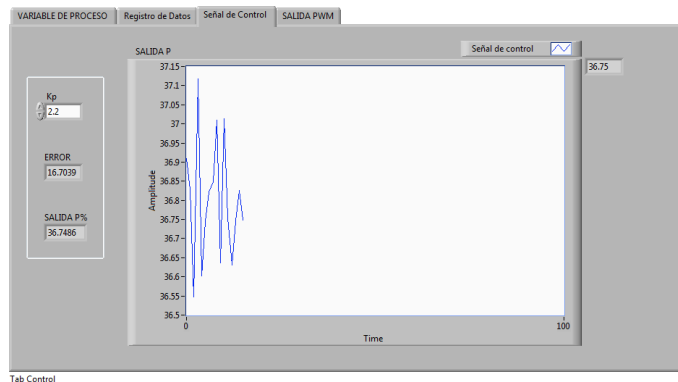


Figura 134: Panel frontal. Pestaña de Señal de Control

El comportamiento del PWM puede ser observado en la página correspondiente, tal y como se muestra en la siguiente figura:

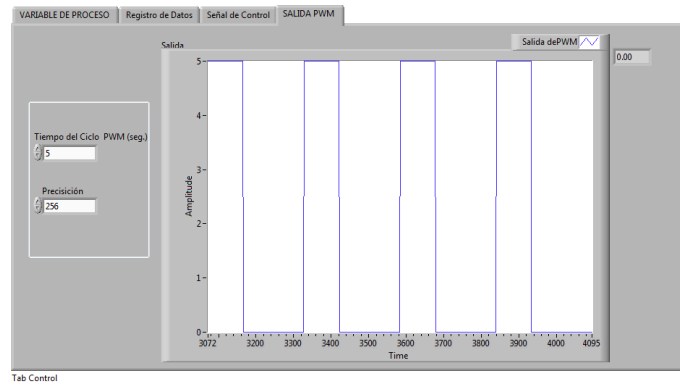


Figura 135: Panel frontal. Comportamiento del PWM

Procedimiento experimental

1. Ajuste los parámetros para el monitoreo del perfil de temperatura y fije la referencia de temperatura deseada (sugerentemente 50 °C)
2. Ajuste los parámetros para el registro de datos de manera que las lecturas coincidan con el perfil de temperatura.
3. Ajuste K_p de acuerdo a la sintonización realizada en la práctica V.
4. Ajuste los parámetros de PWM, sugerentemente para un tiempo de ciclo de 5 seg. y una precisión de 256.
5. Ejecute el programa para observar el comportamiento del sistema ante el control proporcional y espere a que el sistema alcance el estado estacionario, después identifique los siguientes parámetros (no olvide registrar los datos).
 - E_{ss} = Error en estado estacionario
 - t_{ss} = Tiempo en el que alcanza el estado estacionario
 - M_p =Máximo Sobreimpulso
 - t_p = Tiempo de sobreimpulso máximo
6. Los resultados obtenidos deberán ser registrados en la siguiente tabla

Cuadro 10: Tabla resumen de datos

Control P	K_p	M_p	t_p	E_{ss}	t_{ss}
Simulación en Simulink					
Datos Experimentales					

4.7. PRÁCTICA VII: “CONTROL PROPORCIONAL-INTEGRAL”

4.7.1. INTRODUCCIÓN

Anteriormente implementamos un control proporcional y observamos que siempre existirá un error estacionario cuando éste alcanza el estado estable. La forma efectiva de solucionar el problema del error estacionario es agregando al control proporcional la acción integral. Como se menciona en la Sección 2.4.2.2 al control proporcional se le suma la acción integral, que opera tomando en cuenta la magnitud del error y el tiempo que este ha permanecido.

La acción integral es de carácter acumulativo y continua creciendo hasta que el error desaparece. Esta acción se acumula para cada vez que el error es medido, esto quiere decir que cuando el error es 0 la acción integral conserva el valor anterior. En el caso de tener un error negativo la acción integral decrece hasta que el error desaparece. La siguiente tabla muestra un ejemplo de corrección del error y el comportamiento de la acción integral para un sistema de térmico con un Set Point de 200 °C que mide el error cada segundo.

Cuadro 11: Ejemplo de comportamiento de la acción integral

Tiempo	Temperatura	AI	AI acumulada
0	195 °C	5	5
1	195 °C	5	10
2...	195 °C	5	15
...5	195 °C	5	25
6	196 °C	4	29
7...	196 °C	4	33
...11	197 °C	3	48
12	197 °C	3	51
...14	198 °C	2	57
15	198 °C	2	59
xx	200°C	0	63

La tabla anterior es puramente ilustrativa, sin embargo, en un sistema real la acción integral tiene ese comportamiento y se acumula hasta que el error desaparece. Para un control PI la ganancia integral K_i debe tener un valor adecuado, ya que si tiene un valor excesivo el sistema tendrá mucha fuerza y aunque alcanzará velozmente el Set Point el sistema seguirá de largo por efectos de la inercia térmica y habrá un aumento excesivo en la potencia de salida. Si se sintoniza un control PI para valores de K_i demasiado pequeños la acción integral (con error negativo) será en sentido contrario, la AI acumulada irá disminuyendo rápidamente con la magnitud del error. Como consecuencia habrá una excesiva disminución de la potencia de salida y la temperatura probablemente baje del Set Point, entrando así el sistema en un ciclo oscilatorio. En la práctica normalmente K_i deberá ser grande solo en sistemas que reaccionan rápidamente (por ejemplo controles de velocidad de motores) y pequeño para sistemas lentos con mucha inercia (Por ejemplo hornos).

En la presente práctica diseñaremos un control proporcional-integral para el sistema térmico descrito en la prácticas anteriores, utilizando el método de aproximación por transformación bilineal. Controlaremos el sistema térmico utilizado en la práctica anterior, relacionando la salida PI(%) con el ciclo de trabajo de un PWM (tal y como se hizo en la práctica anterior). Recordemos que de acuerdo a lo expuesto en la Sección 1.5.2 el control proporcional-integral correspondiente la expresión correspondiente en el dominio del tiempo discreto para un control PI por aproximación bilineal es:

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

Utilizando la expresión anterior desarrollaremos el código de programa en Labview. Visto como un algoritmo de control el diagrama de bloques para la acción integral es el siguiente:

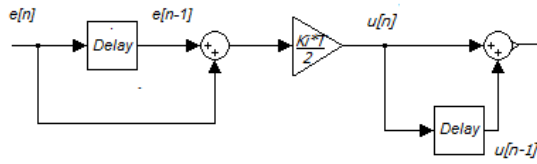


Figura 136: Diagrama de bloques del algoritmo de acción integral

Los parámetros del control PI serán ajustados de acuerdo a los valores obtenidos es la práctica de sintonización por el método de la curva de reacción para un control PI obtenidos .

4.7.2. OBJETIVOS

Objetivo General

Desarrollar un control proporcional-integral en Labview utilizando la DAQ NI USB 6008 para el sistema térmico descrito en la sección 4.7.1.

Objetivos Específicos

- Realizar la simulación en Simulink de Matlab.
- Implementar el control proporcional-integral como un VI en Labview.

4.7.3. MATERIALES Y EQUIPO

Materiales

1 Sensor de Temperatura LM35, 1 Optoaislador MOC 3011, 1 Triac MAC 15, 1 Recipiente con agua.

Equipo

Fuente Simple de dc, PC para la adquisición de datos con la DAQ NI USB 6008, 1 Parrilla Eléctrica.

4.7.4. DESARROLLO DE LA PRÁCTICA

1) Simulación de un control PI en Simulink de Matlab.

Implemente el siguiente diagrama a bloques en simulink para realizar la simulación de Control Integral:

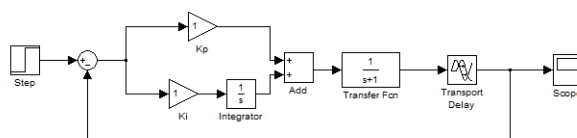


Figura 137: Simulación de control PI

2) Construcción de un VI en Labview que realice la acción de control PI.

Dado que el control proporcional integral requiere un elemento de tiempo dt, crearemos un sub VI que genere una base de tiempo:

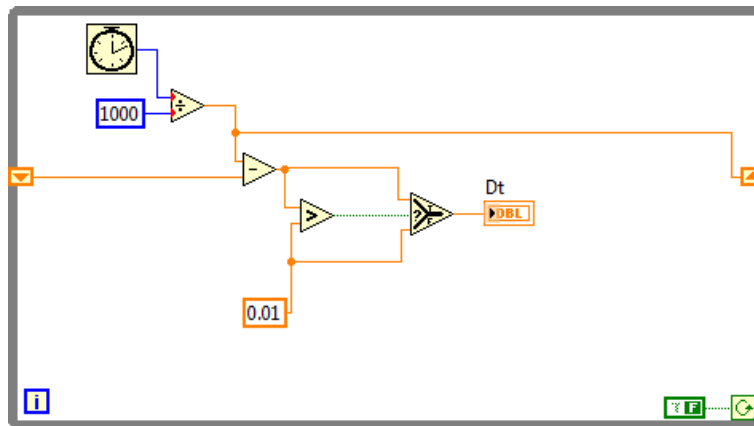


Figura 138: Elemento diferencial de tiempo

La mayoría de los elementos colocados en el VI ya han sido utilizados, sin embargo, notará algunos elementos que hasta el momento no han sido utilizados. La ubicación de dichos elementos es la siguiente:

- Programming\Timing\Tick count (ms)

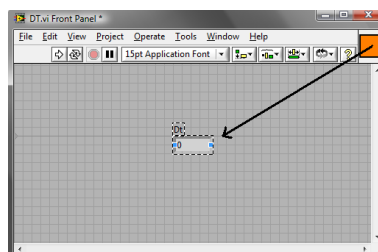


- Programming\Comparison>Select



Edite un ícono para el sub VI realizado con una terminal de salida correspondiente al indicador de salida Dt. Asigne el indicador a la terminal en el panel frontal siguiendo el procedimiento para el VI de acción proporcional.

La asignación es la siguiente :



El ícono deberá tener el siguiente aspecto:



Construya el siguiente diagrama de bloques que corresponde al diseño de la acción integral:

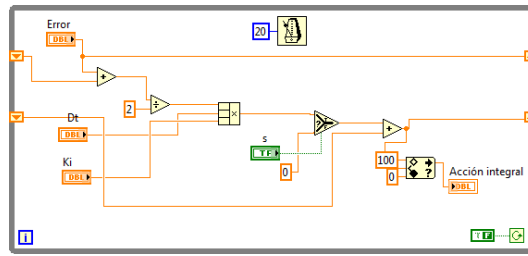


Figura 139: Acción Integral en Labview

La ubicación de los elementos no utilizados en programas anteriores y que se utilizan para la acción integral es la siguiente:

- Programming\Numeric\Compound Arithmetic

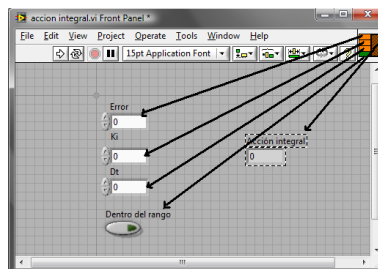


originalmente aparecerá en modo suma, para cambiarlo a modo de multiplicación pulse click derecho sobre el elemento y seleccione la opción change mode.

- Programming\Comparisson\In Range and Coerce



Este elemento servirá para normalizar la Acción integral, en el asignaremos un límite superior y un límite inferior. Después, edite un ícono para la acción integral. La asignación de controles e indicadores para las terminales del sub VI es la siguiente:



El ícono resultante tendrá el siguiente aspecto:



Nota: A manera de sugerencia, compruebe el correcto funcionamiento del algoritmo de control PI de manera similar al procedimiento realizado en el Anexo III.

Al construir la acción integral en Labview es recomendable que tenga un retardo adecuado, dado que si el tiempo de respuesta de la acción integral es mucho mas veloz que el del sistema que nos proponemos controlar la acción integral crecerá demasiado rápido y habrá una saturación pronta en el programa. Como precaución y en caso de que fuese necesario, se coloca en el selector un control booleano que puede ser utilizado como un botón de reseteo para indicar que la acción integral está dentro del rango de trabajo, éste control puede ser dependiente de la normalización del control

de temperatura (mediante un Shift Register) que se diseñará posteriormente, o simplemente se asigna una constante verdadera para que siempre se ejecute la acción integral. Al normalizar la acción integral se recomienda que el límite inferior un valor negativo no muy grande, y que el límite superior no sea mayor a 100. Una vez construido el sub VI de la acción integral, elabore el diagrama a bloques correspondiente al diseño del control PI incluyendo PWM:

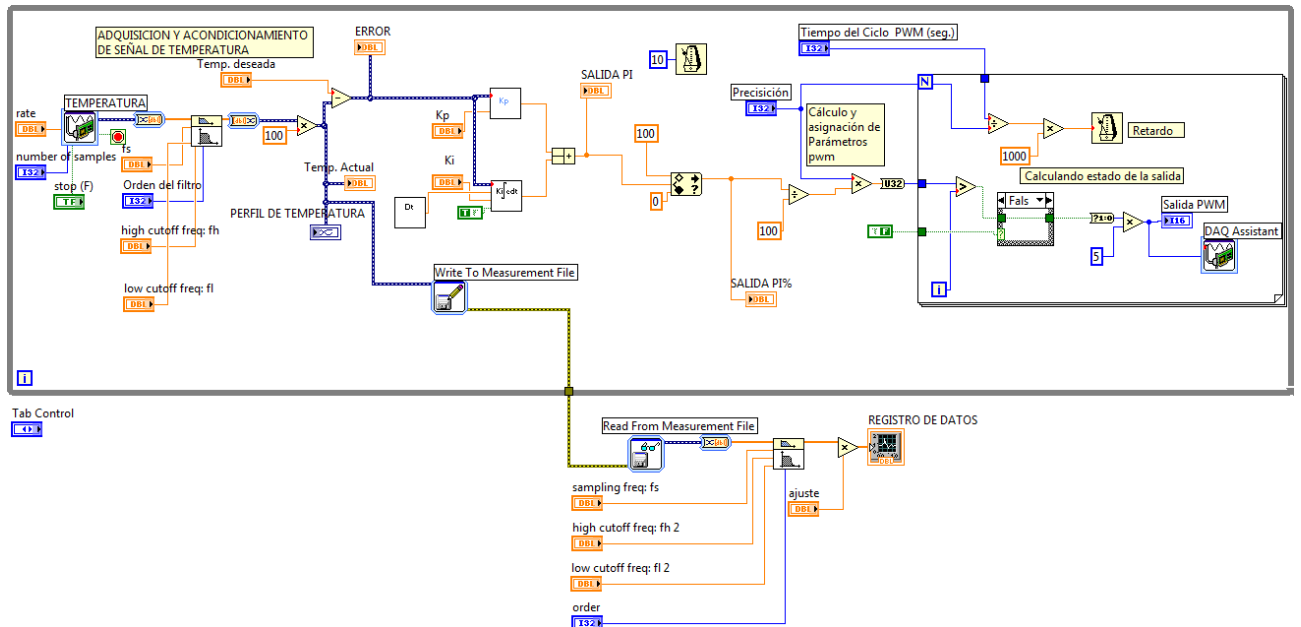


Figura 140: Diagrama a bloques de Control PI de temperatura

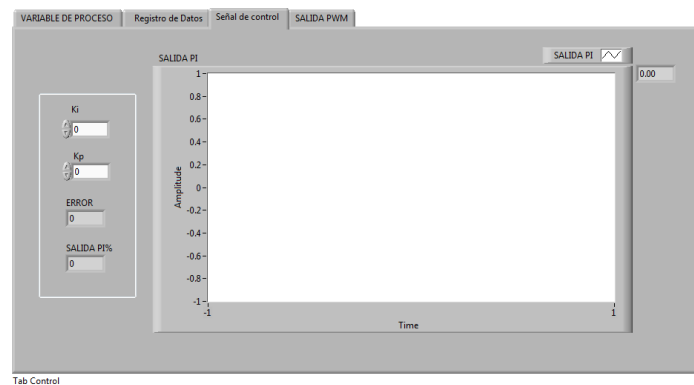


Figura 141: Panel frontal de control PI de temperatura

Según lo mostrado en la figura anterior el panel frontal resultante tiene la siguiente modificación en la tercera pestaña respecto al programa de control proporcional

Procedimiento experimental

1. Ajuste los parámetros para el monitoreo del perfil de temperatura y fije la referencia de temperatura deseada (sugerentemente 50 °C)
2. Ajuste los parámetros para el registro de datos de manera que las lecturas coincidan con el perfil de temperatura.
3. Ajuste K_p y K_i de acuerdo a la sintonización realizada en la práctica V.
4. Ajuste los parámetros de PWM, sugerentemente para un tiempo de ciclo de 5 seg. y una precisión de 256.

5. Ejecute el programa para observar el comportamiento del sistema ante el control proporcional y espere a que el sistema alcance el estado estacionario, después identifique los siguientes parámetros (no olvide registrar los datos).

- Ess= Error en estado estacionario
- tss= Tiempo en el sistema que alcanza el estado estacionario
- Mp=Máximo Sobreimpulso
- tp= Tiempo de sobreimpulso máximo

6. Los resultados obtenidos deberán ser registrados en la siguiente tabla

Cuadro 12: Tabla resumen de datos

Control PI	Kp	Ki	Mp	tp	Ess	tss
Simulación en Simulink						
Datos Experimentales						

4.8. PRÁCTICA VIII: “CONTROL PROPORCIONAL-DERIVATIVO”

4.8.1. INTRODUCCIÓN

Según lo expuesto en la Sección 2.4.2 un control proporcional derivativo es uno proporcional al que se le agrega la capacidad de considerar también la velocidad de la variable de proceso en el tiempo. La acción derivativa considera la razón de cambio de la variable de proceso, con lo cual podemos decir que la acción derivativa es también proporcional a la diferencial del error.

Para entender el efecto de la acción derivativa, consideremos como ejemplo un sistema térmico (horno, parrilla, etc). Si la temperatura del Sistema esta por debajo del Set Point, pero subiendo muy rápidamente(velocidad positiva) y se va a pasar de largo el SP, entonces el control se adelanta y disminuye la potencia de los calefactores considerando una constante derivativa. Al revés si la temperatura es mayor que el Set Point y si el control estima que la temperatura baja muy rápido y se va pasar para abajo del SP, entonces le coloca algo de potencia a la salida para ir frenando el descenso brusco.

En la presente práctica diseñaremos un control proporcional-derivativo utilizando el método de aproximación por el método de derivación simple. Controlaremos el sistema térmico utilizado en la práctica anterior, relacionando la salida PD(%) con el ciclo de trabajo de un PWM (tal y como se hizo en la práctica anterior). Recordemos que de acuerdo a lo expuesto en la Sección 2.4.2 el control proporcional-derivativo correspondiente la expresión correspondiente en el dominio del tiempo discreto para un control PD por el método de derivación simple es:

$$U[n] = K_p e[n] + K_d \left(\frac{e[n] - e[n-1]}{T} \right)$$

Utilizando la expresión anterior desarrollaremos el código de programa en Labview. Visto como diagrama a bloques el algoritmo de la acción derivativa es el siguiente:

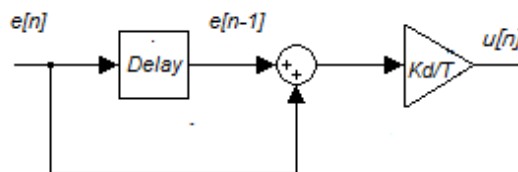


Figura 142: Diagrama de bloques algoritmo de acción derivativa

Los parámetros del control PD serán ajustadas de acuerdo al método de la curva de reacción.

4.8.2. OBJETIVOS

Objetivo General

Desarrollar un control proporcional-derivativo en Labview utilizando la DAQ NI USB 6008 para el sistema térmico descrito en las prácticas anteriores.

Objetivos Específicos

- Realizar la simulación en Simulink de Matlab.
- Implementar el control proporcional-derivativo como un VI en Labview.

4.8.3. MATERIALES Y EQUIPO

Materiales

1 Sensor de Temperatura LM35, 1 Optoaislador MOC 3011, 1 Triac MAC 15, 1 Recipiente con agua.

Equipo

Fuente Simple de dc, PC para la adquisición de datos con la DAQ NI USB 6008, 1 Parrilla Eléctrica.

4.8.4. DESARROLLO DE LA PRÁCTICA

1) Simulación de un control PD en Simulink de Matlab.

Diagrama a bloques:

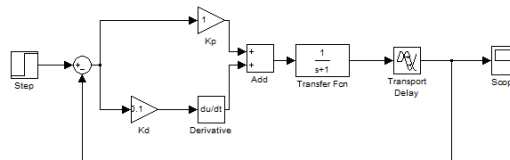


Figura 143: Diagrama a bloques de Control PD en Simulink

2) Construcción de un VI en Labview que realice la acción de control PD.

Diseño de la acción Derivativa:

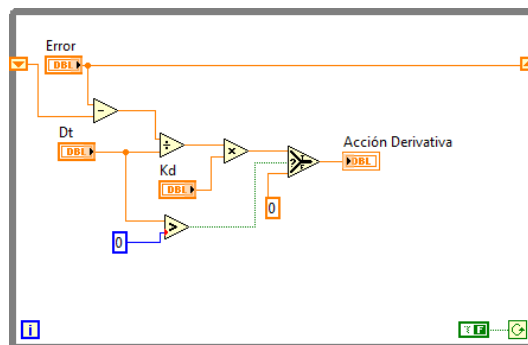
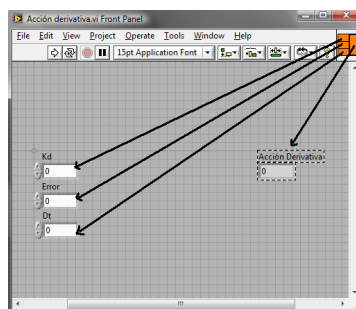


Figura 144: Acción derivativa en Labview

Edite un ícono para la acción derivativa. La asignación de controles e indicadores para las terminales del sub VI es la siguiente



Ícono resultante:



Nota: A manera de sugerencia, compruebe el correcto funcionamiento del algoritmo de control PD de manera similar al procedimiento realizado en el Anexo III.

Realice el siguiente diagrama de bloques para el control de temperatura PD:

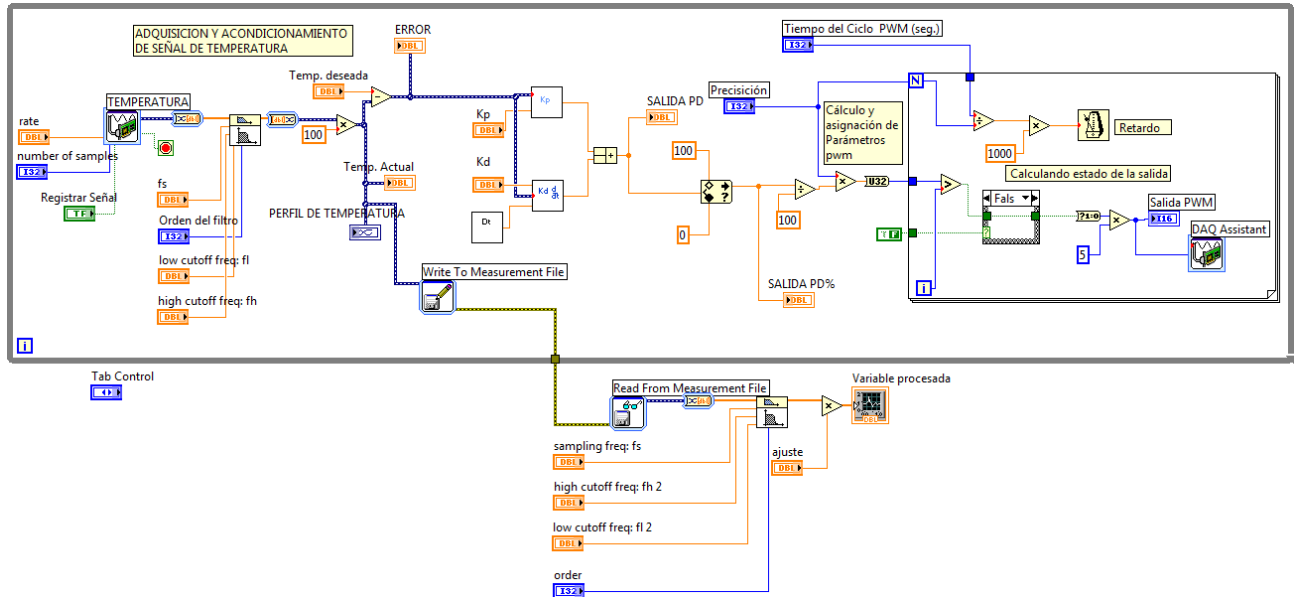


Figura 145: Diagrama de bloques de control PD de temperatura

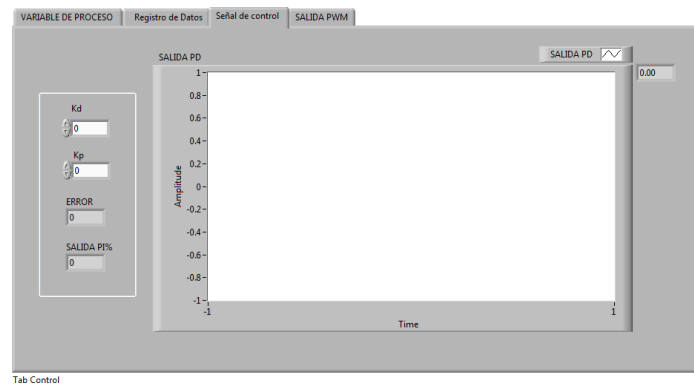


Figura 146: Panel frontal de control PD de Temperatura

En la figura anterior se muestra la pestaña correspondiente a la señal de control desde el panel frontal.

Procedimiento experimental

1. Ajuste los parámetros para el monitoreo del perfil de temperatura y fije la referencia de temperatura deseada (sugerentemente 50 °C)
2. Ajuste los parámetros para el registro de datos de manera que las lecturas coincidan con el perfil de temperatura.
3. Ajuste K_p y K_d de acuerdo a la sintonización realizada en la práctica V.
4. Ajuste los parámetros de PWM, sugerentemente para un tiempo de ciclo de 5 seg. y una precisión de 256.

5. Ejecute el programa para observar el comportamiento del sistema ante el control proporcional y espere a que el sistema alcance el estado estacionario, después identifique los siguientes parámetros (no olvide registrar los datos).

- Ess= Error en estado estacionario
- tss= Tiempo en el sistema que alcanza el estado estacionario
- Mp=Máximo Sobreimpulso
- tp= Tiempo de sobreimpulso máximo

6. Los resultados obtenidos deberán ser registrados en la siguiente tabla

Cuadro 13: Tabla resumen de datos

Control PI	Kp	Kd	Mp	tp	Ess	tss
Simulación en Simulink						
Datos Experimentales						

4.9. PRÁCTICA IX: “CONTROL PROPORCIONAL-INTEGRAL-DERIVATIVO”

4.9.1. INTRODUCCIÓN

En la presente práctica desarrollaremos un control PID utilizando una combinación de los métodos de integración y derivación discretos. Para la acción integral utilizaremos el método de transformación bilineal y para la acción derivativa utilizaremos el método de derivación simple. La expresión a utilizar para el diseño del control PID en Labview es la siguiente:

$$u[n] = K_p e[n] + K_i \left(\frac{T}{2} (e[n] + e[n-1]) + u[n-1] \right) + K_d \left(\frac{e[n] - e[n-1]}{T} \right)$$

La sintonización del PID será de acuerdo al método de la curva de reacción de Ziegler y Nichols.

4.9.2. OBJETIVOS

Objetivo General

Desarrollar un control proporcional-integral-derivativo en Labview utilizando la DAQ NI USB 6008 para el sistema térmico descrito en las prácticas anteriores.

Objetivos Específicos

- Realizar la simulación en Simulink de Matlab.
- Implementar el control proporcional-integral-derivativo como un VI en Labview.

4.9.3. MATERIALES Y EQUIPO

Materiales

1 Sensor de Temperatura LM35, 1 Optoaislador MOC 3011, 1 Triac MAC 15, 1 Recipiente con agua.

Equipo

Fuente Simple de dc, PC para la adquisición de datos con la DAQ NI USB 6008, 1 Parrilla Eléctrica.

4.9.4. DESARROLLO DE LA PRÁCTICA

1) Simulación de un control PID en Simulink de Matlab.

Diagrama a bloques:

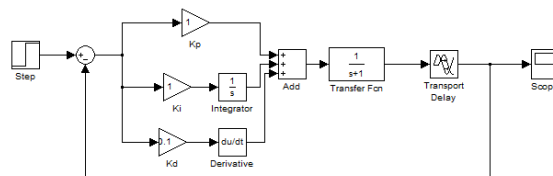


Figura 147: Control PID en Simulink

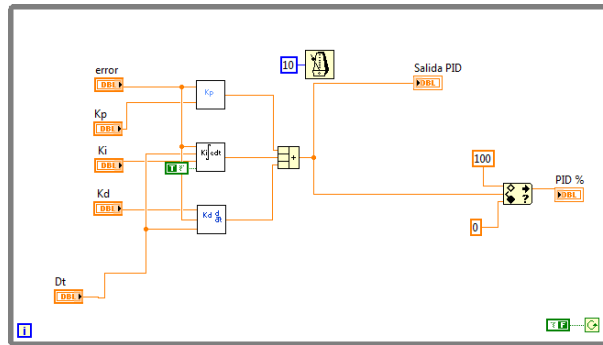
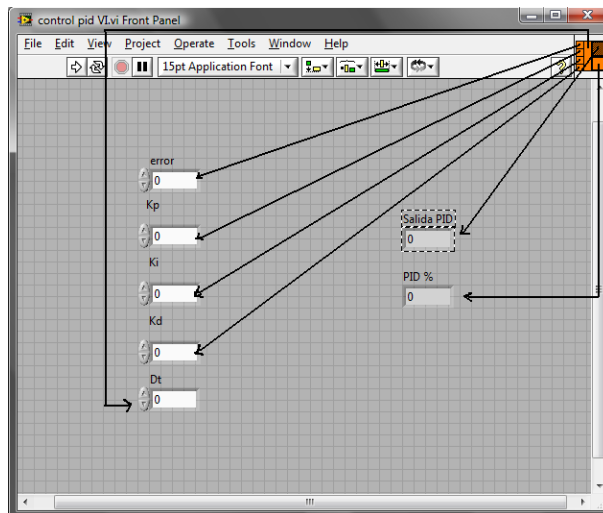


Figura 148: Control PID en Labview

2) Construcción de un VI en Labview que realice la acción de control PID.

La asignación de controles e indicadores al editar el ícono correspondiente es la siguiente:



El aspecto del ícono deberá ser el siguiente:



Nota: A manera de sugerencia, compruebe el correcto funcionamiento del algoritmo de control PID de como se muestra en el Anexo III.

Realice el siguiente diagrama a bloques para el control PID de temperatura:

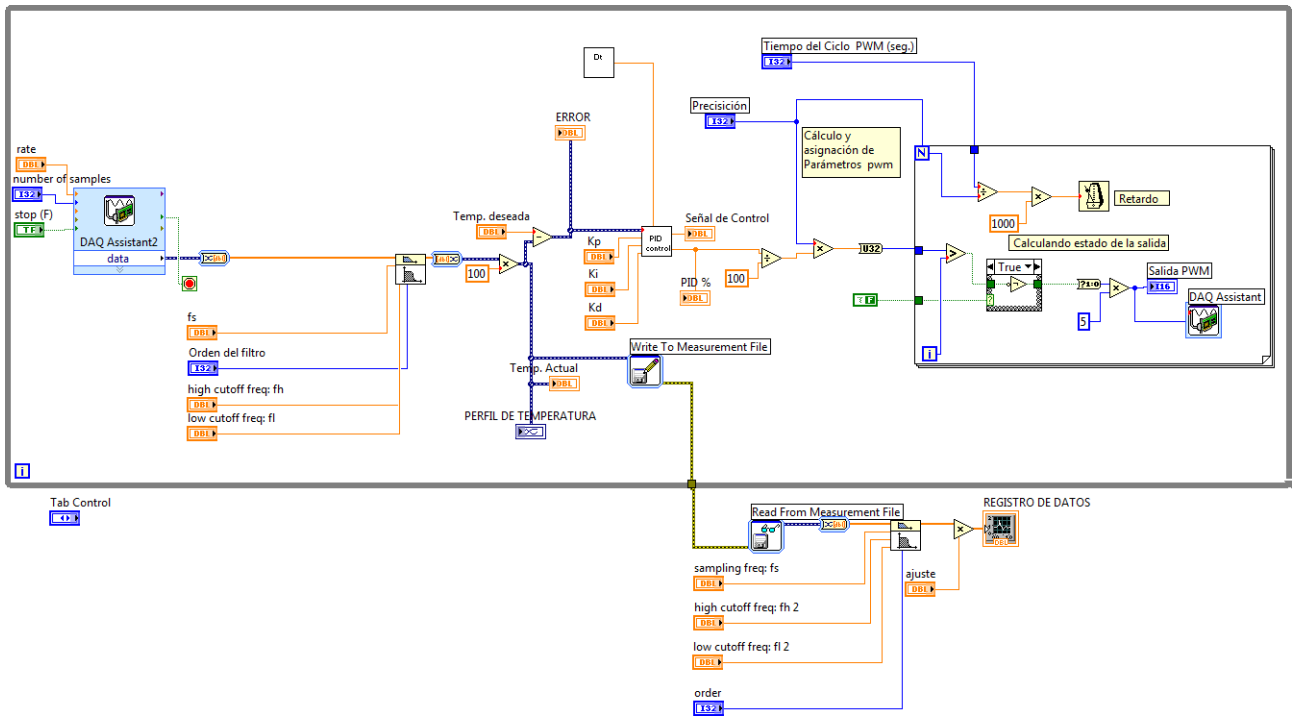


Figura 149: Control PID de temperatura

El panel frontal resultante es el siguiente:

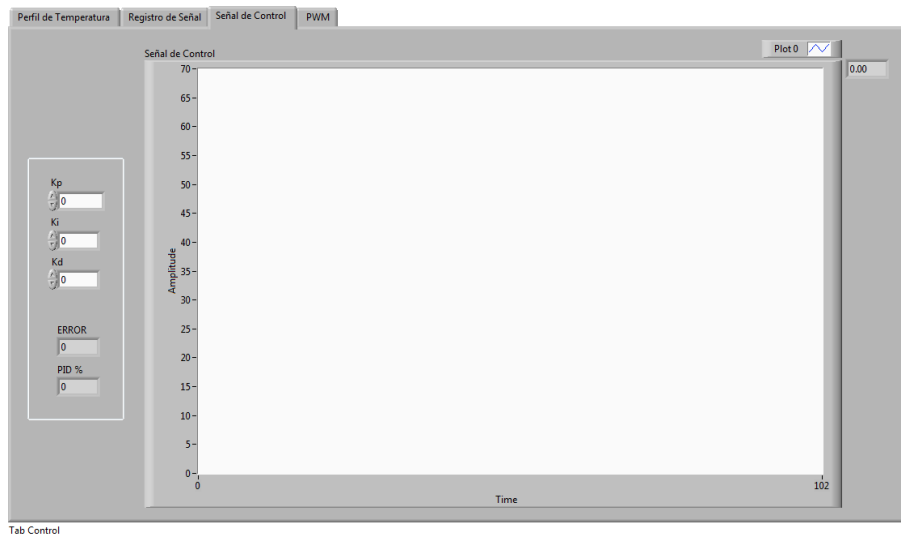


Figura 150: Panel frontal de control PID de temperatura

Procedimiento experimental

1. Ajuste los parámetro para el monitoreo del perfil de temperatura y fije la referencia de temperatura deseada (sugerentemente 50 °C)
2. Ajuste los parámetros para el registro de datos de manera que las lecturas coincidan con el perfil de temperatura.
3. Ajuste K_p , K_i y K_d de acuerdo a la sintonización realizada en la práctica V.
4. Ajuste los parámetros de PWM, sugerentemente para un tiempo de ciclo de 5 seg. y una precisión de 256.
5. Ejecute el programa para observar el comportamiento del sistema ante el control proporcional y espere a que el sistema alcance el estado estacionario, después identifique los siguientes parámetros (no olvide registrar los datos).
 - E_{ss} = Error en estado estacionario
 - t_{ss} = Tiempo en el sistema que alcanza el estado estacionario
 - M_p =Máximo Sobreimpulso
 - t_p = Tiempo de sobreimpulso máximo
6. Los resultados obtenidos deberán ser registrados en la siguiente tabla

Cuadro 14: Tabla resumen de datos

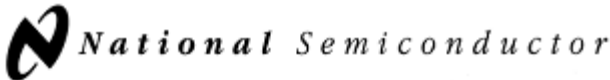
Control PI	K_p	K_i	K_d	M_p	t_p	E_{ss}	t_{ss}
Simulación en Simulink							
Datos Experimentales							

5. CONCLUSIONES

A lo largo del desarrollo de cada una de las prácticas nos percatamos de utilidad y funcionalidad que brinda el software Labview y la adquisición de Datos. También nos percatamos de la dificultad al controlar un sistema físico real, dado que tienen que hacerse consideraciones como tiempo de respuesta del sistema, acondicionamiento de la señal de la variable de proceso (dependiendo del tipo de sensor que se maneje), sin mencionar que debe de hallarse un modelo matemático adecuado que nos permita aproximar adecuadamente el sistema, y a su vez, hacer los ajustes necesarios. Algo que es importante mencionar, es la desaparición de circuitos electrónicos físicos, dado que el único circuito que se implemento fue una etapa de potencia en base a un optoaislador y un triac; pero a la vez, en ningún momento se dejaron de aplicar los conceptos básicos de ingeniería electrónica. Recordemos que en las prácticas realizadas de control de lazo cerrado se reafirmaron los conceptos expuestos en los Fundamentos Teóricos. Cabe recalcar que la DAQ NI USB 6008 es una tarjeta útil mayormente para aplicaciones didácticas dada su multifuncionalidad, sin embargo, para aplicaciones de mayores exigencias (ó a nivel industrial) existen otras tarjetas de adquisición con mayor capacidad de muestreo, y con mas cantidad de entradas y salidas que pueden ser programadas siguiendo los principios de este manual (adaptándonos a las características de la tarjeta y la aplicación que se desee).

6. ANEXOS

6.1. ANEXO I: HOJAS DE ESPECIFICACIONES DE LOS COMPONENTES

November 2000

LM35

Precision Centigrade Temperature Sensors

General Description

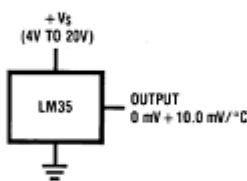
The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in ° Kelvin, as the user is not required to subtract a large constant 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 $60\ \mu\text{A}$ 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° to $+150^\circ\text{C}$ temperature range, while the LM35C is rated for a -40° to $+110^\circ\text{C}$ range (-10° with improved accuracy). The LM35 series is available pack-

aged 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-220 package.

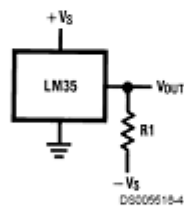
Features

- 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° 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 $60\ \mu\text{A}$ current drain
- Low self-heating, 0.08°C in still air
- Nonlinearity only $\pm 1/4^\circ\text{C}$ typical
- Low impedance output, $0.1\ \Omega$ for 1 mA load

Typical Applications



DS000510-3



DS000518-4

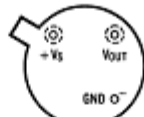
Choose $R_1 = -V_S/50\ \mu\text{A}$
 $V_{\text{OUT}} = +1,500\ \text{mV}$ at $+150^\circ\text{C}$
 $= +250\ \text{mV}$ at $+25^\circ\text{C}$
 $= -550\ \text{mV}$ at -55°C

FIGURE 1. Basic Centigrade Temperature Sensor
($+2^\circ\text{C}$ to $+150^\circ\text{C}$)

FIGURE 2. Full-Range Centigrade Temperature Sensor

Connection Diagrams

**TO-46
Metal Can Package***



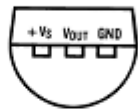
BOTTOM VIEW
DS000510-1

*Case is connected to negative pin (GND)

Order Number LM35H, LM35AH, LM35CH, LM35CAH or LM35DH

See NS Package Number H03H

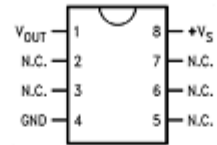
**TO-92
Plastic Package**



BOTTOM VIEW
DS000519-2

Order Number LM35CZ, LM35CAZ or LM35DZ
See NS Package Number Z03A

**SO-8
Small Outline Molded Package**

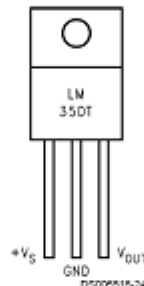


DS000516-21

N.C. = No Connection

Top View
Order Number LM35DM
See NS Package Number M08A

**TO-220
Plastic Package***



DS000515-24

*Tab is connected to the negative pin (GND).

Note: The LM35DT pinout is different than the discontinued LM35DP.

Order Number LM35DT
See NS Package Number TA03F

MAC15 Series

Preferred Device

Triacs

Silicon Bidirectional Thyristors

Designed primarily for full-wave ac control applications, such as solid-state relays, motor controls, heating controls and power supplies; or wherever full-wave silicon gate controlled solid-state devices are needed. Triac type thyristors switch from a blocking to a conducting state for either polarity of applied main terminal voltage with positive or negative gate triggering.

- Blocking Voltage to 800 Volts
- All Diffused and Glass Passivated Junctions for Greater Parameter Uniformity and Stability
- Small, Rugged, Thermowatt Construction for Low Thermal Resistance, High Heat Dissipation and Durability
- Gate Triggering Guaranteed in Three Modes (MAC15 Series) or Four Modes (MAC15A Series)
- Device Marking: Logo, Device Type, e.g., MAC15A6, Date Code

MAXIMUM RATINGS (T_J = 25°C unless otherwise noted)

Rating	Symbol	Value	Unit
Peak Repetitive Off-State Voltage ⁽¹⁾ (T _J = -40 to +125°C, Sine Wave 50 to 60 Hz, Gate Open) MAC15A6 MAC15-8, MAC15A8 MAC15-10, MAC15A10	V _{DRM} , V _{RRM}	400 600 800	Volts
Peak Gate Voltage (Pulse Width ≤ 1.0 μsec; T _C = 90°C)	V _{GM}	10	Volts
On-State Current RMS Full Cycle Sine Wave 50 to 60 Hz (T _C = +90°C)	I _{T(RMS)}	15	A
Circuit Fusing Consideration (t = 8.3 ms)	I ² t	93	A ² s
Peak Non-repetitive Surge Current (One Full Cycle Sine Wave, 60 Hz, T _C = +80°C) Preceded and followed by rated current	I _{TSM}	150	A
Peak Gate Power (T _C = +80°C, Pulse Width = 1.0 μs)	P _{GM}	20	Watts
Average Gate Power (T _C = +80°C, t = 8.3 ms)	P _{G(AV)}	0.5	Watts
Peak Gate Current (Pulse Width ≤ 1.0 μsec; T _C = 90°C)	I _{GM}	2.0	A
Operating Junction Temperature Range	T _J	-40 to +125	°C
Storage Temperature Range	T _{stg}	-40 to +150	°C

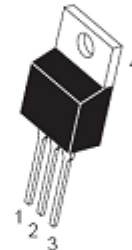
(1) V_{DRM} and V_{RRM} for all types can be applied on a continuous basis. Blocking voltages shall not be tested with a constant current source such that the voltage ratings of the devices are exceeded.



ON Semiconductor

<http://onsemi.com>

TRIACS
15 AMPERES RMS
400 thru 800 VOLTS



TO-220AB
CASE 221A
STYLE 4

PIN ASSIGNMENT

1	Main Terminal 1
2	Main Terminal 2
3	Gate
4	Main Terminal 2

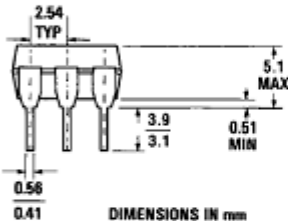
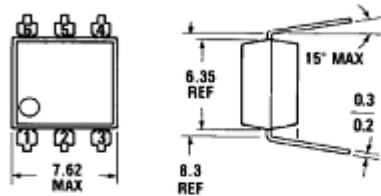
ORDERING INFORMATION

Device	Package	Shipping
MAC15-8	TO220AB	500/Box
MAC15-10	TO220AB	500/Box
MAC15A6	TO220AB	500/Box
MAC15A8	TO220AB	500/Box
MAC15A10	TO220AB	500/Box

Preferred devices are recommended choices for future use and best overall value.

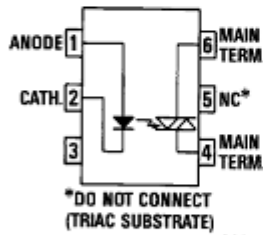
**MOC3009 MOC3010
MOC3011 MOC3012**

PACKAGE DIMENSIONS



DIMENSIONS IN mm
PACKAGE CODE E

ST1603-02



*DO NOT CONNECT
(TRIAC SUBSTRATE)

C2081

Equivalent Circuit

DESCRIPTION

The MOC3009, MOC3010, MOC3011 and MOC3012 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 series is designed for interfacing between electronic controls and power triacs to control resistive and inductive loads for 120 VAC operations.

FEATURES

- Low input current required (typically 5mA—MOC3011)
- High isolation voltage—minimum 7500 VAC peak
- Underwriters Laboratory (UL) recognized—File E90700

APPLICATIONS

- 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
Withstand test voltage ...	7500 VAC Peak (50-60 Hz)

INPUT DIODE

Forward DC current	50 mA
Reverse voltage	3 V
Peak forward current (1 μs pulse, 300 pps)	3.0 A
Power dissipation (25°C ambient)	100 mW
Derate linearly (above 25°C)	1.33 mW/°C

OUTPUT DRIVER

Off-state output terminal voltage	250 volts
On-state RMS current $T_x=25^\circ\text{C}$	100 mA
(Full cycle, 50 to 60 Hz) $T_x=70^\circ\text{C}$	50 mA
Peak nonrepetitive surge current	1.2 A
(PW=10 ms, DC=10%)	
Total power dissipation @ $T_x=25^\circ\text{C}$	300 mW
Derate above 25°C	4.0 mW/°C

6.2. ANEXO II: OTROS MODELOS DE DAQ's

6.2.1. NI USB-6009

La DAQ NI USB 6009 presenta una configuración de terminales idéntica a la DAQ NI USB 6008. Sin embargo su capacidad de muestro es mucho mayor por lo que puede ser utilizada para aplicaciones de mayor exigencia.



- Adquisición de Datos Multifunción por USB de 14 bits a 48 kS/s.
- 8 canales de entrada analógica con resolución de 14 bits, 12 líneas de E/S digital, 2 salidas analógicas, 1 contador.
- Para mayor rendimiento considere el NI USB-6210 y el NI USB-6211.
- Diseño energizado por bus para mayor comodidad y portabilidad.
- Software controlador para Windows, Mac OS X, Linux y Pocket PC.
- Software de NI-DAQmx.

6.2.2. NI USB-6210



- DAQ Multifunción Energizado por Bus USB de 16 bits a 250 kS/s
- 16 entradas analógicas (16 bits), razón de muestreo de un solo canal a 250 kS/s
- 4 entradas digitales, 4 salidas digitales, contadores de 32 bits
- Energizado por bus USB para una mayor movilidad, conectividad de señal integrada
- USB plug-and-play para detección y cautomáticas.
- Compatible con LabVIEW, LabWindows/CVI y Measurement Studio para Visual Studio .NET
- Incluye el software de NI-DAQmx y software interactivo NI LabVIEW SignalExpress para registro de datos.

6.2.3. NI USB-6211



- Multifunción Energizado por Bus USB de 16 bits a 250 kS/s
- 16 entradas analógicas (16 bits), razón de muestreo de un solo canal a 250 kS/s
- 2 salidas analógicas (16 bits a 250 kS/s), 4 entradas digitales, 4 salidas digitales, contadores de 32 bits
- Energizado por bus USB para una mayor movilidad, conectividad de señal integrada
- USB plug-and-play para detección y configuración automáticas
- Compatible con LabVIEW, LabWindows/CVI y Measurement Studio para Visual Studio .NET
- Incluye el software de NI-DAQmx y software interactivo NI LabVIEW SignalExpress para registro de datos

6.2.4. NI PCI-6024E



- 16 Entradas Analógicas DAQ Multifunción de 12 bits a 200 kS/s
- Dos salidas analógicas de 12 bits; 8 líneas de E/S digitales;
- dos contadores de 24 bits # Software controlador NI-DAQmx incluido y servicios de medida adicionales
- Integración superior con LabVIEW, LabWindows/CVI, y Measurement Studio para VB y VS .NET
- Certificado trazable de calibración expedido por el NIST y más de 70 opciones de acondicionamiento de señal

6.3. ANEXO III: PROGRAMA PRUEBA DE PID

El correcto funcionamiento de los algoritmos desarrollados para las aplicaciones de control PID puede ser comprobado mediante el siguiente procedimiento:

1) Conecte la DAQ NI USB realimentando una entrada analógica con una salida analógica como se muestra en la siguiente figura:

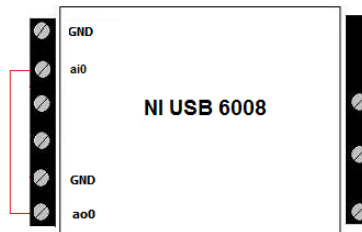


Figura 151: Conexión de la DAQ NI USB 6008

2) Construya el siguiente diagrama de bloques :

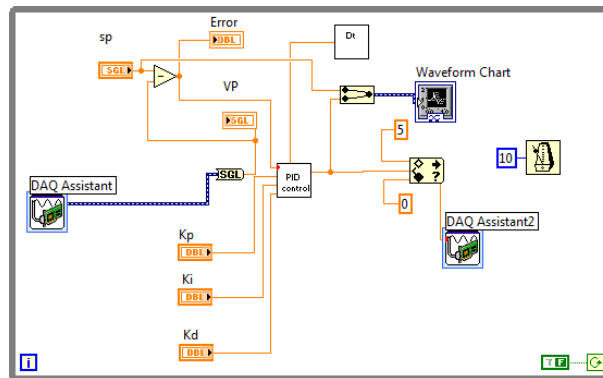


Figura 152: Diagrama de Bloques programa prueba de PID

Los tiempos de retardo para todos los sub VI's que incluye el VI de PID se sugiere sean de 100 ms (solo para realizar la prueba del algoritmo).

3) Fije las ganancias desde el panel frontal (sugerentemente $K_p=0.1$, $K_i=0.1$, $K_d=0.001$):

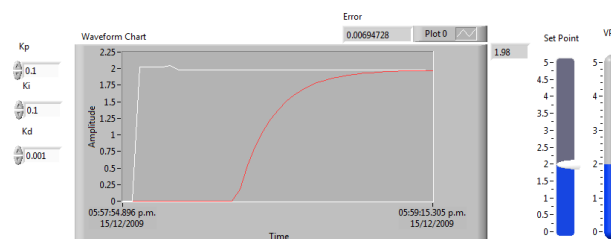


Figura 153: Panel frontal prueba PID

Cuando el programa es ejecutado se observa que el valor de VP alcanzará el valor del Set Point. Cambie las referencias en varias ocasiones y observará que siempre ocurrirá lo mismo.

6.4. CONTENIDO DEL INFORME DE PRÁCTICA

El informe de práctica es un documento en cual hacemos constar la comprensión cabal y total de los experimentos realizados. En el tenemos la oportunidad de expresar de forma concisa y clara los motivos, resultados y conclusiones de cada uno de los experimentos realizados. Es importante recalcar que el informe de práctica debe ser útil no solamente para el catedrático de la materia, sino para la mayor cantidad de lectores posibles.

1. Encabezamiento del informe

Título: Debe ser específico e informativo y, en lo posible, agudo y provocador. Con él debemos dar una idea clara del tema estudiado.

Autoría: Contiene los nombres de los autores, incluyendo alguna vía de comunicación con los mismos, por ejemplo, dirección electrónica, teléfono, dirección postal, etc.

Resumen: El resumen debe dar un adelanto de lo que se leerá en el cuerpo del mismo, con la mayor brevedad posible. Aquí debemos indicar concisamente el tema del trabajo, referirnos sucintamente a la metodología seguida y destacar los resultados más importantes obtenidos.

2. Cuerpo del informe

Introducción: En esta sección introducimos al lector al tema de estudio. Es recomendable incluir los motivos por los cuales se realizó el experimento y un breve marco teórico que sea de ayuda para la comprensión del tema.

Métodología experimental: Al referirnos a la metodología experimental tenemos que considerar los siguientes aspectos:

- Descripción del Sistema: Es útil utilizar diagramas de bloques con una breve descripción de los elementos utilizados en el ó los experimentos.
- Simulación, Programas, Circuitos Implementados: Aquí mostramos los diagramas de bloques de la simulación realizada, los programas en Labview construidos, y los diagramas esquemáticos de los circuitos utilizados. Para cada uno de ellos conviene realizar una breve y concisa descripción de su funcionamiento.

Resultados Obtenidos: En los resultados consideraremos dos renglones.

- Resultados de Simulación: Aquí presentamos los resultados de la Simulación, recordemos que la simulación arroja resultados bajo condiciones ideales.
- Resultados Experimentales: Aquí mostramos los resultados obtenidos en la realización de los experimentos. Es importante recordar que los resultados deberán ser presentados en forma de gráficos, tablas resumen de datos (las planteadas en las prácticas) y en algunos casos modelos matemáticos obtenidos.

3. Conclusiones: En esta sección tenemos que comentar objetivamente qué hemos aprendido del experimento realizado, y sintetizar las consecuencias e implicaciones que encontramos asociadas a nuestros resultados. Podemos decir que un buen informe es aquel que demuestra el mayor número de conclusiones (correctas) alcanzadas a partir de los datos obtenidos.

4. Referencias: Las referencias bibliográficas se ordenan al final del informe. Deben contener el nombre de los autores de las publicaciones (artículos en revistas o libros) citados en el texto, el título de los trabajos, el nombre de la revista o editorial que los publicó; además, se deben incluir los datos que ayuden a la identificación de los mismos: volumen donde están incluidos, capítulo, página, fecha de publicación, etc.

Referencias

- [Ellis, G. (2004)] Ellis, George (2004). *Control System Design Guide* (3rd. Ed.). Elsevier Academic Press.
- [Manuel L., A. M. (2002)] Manuel Lázaro, Antoni Manuel. *LABVIEW 7.1 Programación Gráfica para el control de instrumentación*. Madrid: Editorial Thomson.
- [somi XV] M. Bañuelos S., J. Castillo H., G. Rayo L. S. Quintana T., R. Damián Z., J. Pérez S. *CONTROLADOR PID DE TEMPERATURA DE TIPO DIDACTICO*. Lab. de Electrónica. Centro de Instrumentos, UNAM.
- [N.S., Nise (2002)] Nise, Norman S. (2002). *Sistemas de Control para ingeniería*. Grupo Editorial Patria.
- [Ogata, K. (1998)] Ogata, Kathushiko (1998). *Ingeniería de Control Moderna* (3ra. Ed.). Editorial Prentice-Hall.
- [Arian nt-10] Grupo de desarrollo, ARIAN S.A. *Nota técnica 10: Control PID, una revelación tutorial de los enigmas*.
- [Guide 6008/6009] *User Guide and Specifications 6008/6009*.