

**DESARROLLO E IMPLEMENTACIÓN DE UN MODELO EXPERIMENTAL DE
MONITOREO Y CONTROL POSICIÓN DE UN MOTOR DC QUE ESTÁ
INSTALADO EN EL MÓDULO DE MONITOREO Y CONTROL DE
INSTRUMENTACIÓN (M2CI) QUE PERMITA OPERAR Y CONTROLAR LA
PLANTA PILOTO DE CONTROL DE POSICIÓN MEDIANTE ACCESO REMOTO**

ALEXIS MELENDEZ OCAMPO

**UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA
CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
INGENIERIA ELECTRONICA
NEIVA
2017**

**DESARROLLO E IMPLEMENTACIÓN DE UN MODELO EXPERIMENTAL DE
MONITOREO Y CONTROL POSICIÓN DE UN MOTOR DC QUE ESTÁ
INSTALADO EN EL MÓDULO DE MONITOREO Y CONTROL DE
INSTRUMENTACIÓN (M2CI) QUE PERMITA OPERAR Y CONTROLAR LA
PLANTA PILOTO DE CONTROL DE POSICIÓN MEDIANTE ACCESO REMOTO**

ALEXIS MELENDEZ OCAMPO

TRABAJO DE GRADO

PEDRO TORRES SILVA

**Director de tesis
Docente Asistente**

**UNIVERSIDAD NACIONAL ABIERTA Y A DISTANCIA
CIENCIAS BÁSICAS, TECNOLOGÍA E INGENIERÍA
INGENIERIA ELECTRONICA
NEIVA
2017**

Nota de Aceptación

CUATRO PUNTO TRES (4,3)

Ing. Gilma Paola Andrade Trujillo

Presidente del Jurado

Ing. FABIAN BOLIVAR MARIN

Jurado

Neiva, 20 octubre de 2017

Dedico este trabajo a Dios todo poderoso que me ha iluminado el camino para llegar este punto, junto a mi familia y su gran apoyo para culminar esta etapa de mi vida.

AGRADECIMIENTOS

A mi director de proyecto de grado por permitirme que pudiera aplicar el conocimiento adquirido durante lapso de tiempo de mis estudios.

A los tutores que me brindaron su gran apoyo durante la formación académica de mi carrera profesional.

A la Universidad por brindarme la oportunidad de terminar mi ciclo profesional de manera virtual.

CONTENIDO

1. INTRODUCCIÓN	16
2. OBJETIVOS.....	17
2.1 OBJETIVO GENERAL.....	17
2.2 OBJETIVOS ESPECÍFICOS	17
3 PLANTEAMIENTO DEL PROBLEMA.....	18
3.1 DEFINICIÓN DEL PROBLEMA	18
3.2 JUSTIFICACIÓN.....	19
4. MARCO TEÓRICO	20
4.1 DAQ USB 6008 (TAD de National Instruments – NI).....	20
4.2 Servomotor	21
4.3 Tarjeta de control de posición del servo motor.	24
4.4. Potenciómetro lineal.	26
4.5 Labview	27
4.5.1 Panel Frontal	27
4.5.2 Paleta de Controles	28
4.5.3 Controles e Indicadores.....	29
4.5.4 Controles e Indicadores Booleano	30
4.5.5 Controles e Indicadores de Cadena de Caracteres.....	30
4.5.6 Diagrama de Bloques	31
4.5.7 Terminales.....	32
4.5.7 Controles, Indicadores y Constantes.....	33
4.5.9 Nodos de Diagrama de Bloques.....	35
4.5.10 Funciones.....	35
4.5.11 SubVIs.....	35
4.5.12 Express Vis	36
4.5.13 Paleta de Funciones.....	37

5. MATERIALES Y MÉTODOS.....	38
5.1 MATERIALES.....	38
5.2 METODOLOGÍA.....	38
6 DESARROLLO DEL PROYECTO	39
6.1 Análisis del desarrollo del proyecto	58
6.2 Cronograma.....	60
CONCLUSIONES	61
RECOMENDACIONES.....	62
BIBLIOGRAFÍA.....	63
ANEXO A.....	64

LISTA DE TABLAS

	Pág.
Tabla 1. Estado de funcionamiento del puente en H	30
Tabla 2. Tabla de estados.	56

LISTA DE FIGURAS

	Pág.
Figura 1. DAQ USB 6008	20
Figura 2. Servomotor de D.C.	22
Figura 3. Tren de pulsos para control del servo	23
Figura 4. Otra posibilidad de pulsos de control	24
Figura 5. Estructura de un puente H.	25
Figura 6. Los 2 estados básicos del circuito.	25
Figura 7. Gráfica de potenciómetro lineal	27
Figura 8. Ejemplo de un Panel Frontal (1) Ventana de Panel Frontal (2) Barra de Herramientas (3) Paleta de Controles	28
Figura 9. Paleta de Controles	28
Figura 10. Controles e Indicadores Numéricos (1) Botones de Incremento/Reducción (2) Control Numérico (3) Indicador Numérico	30
Figura 11. Controles e Indicadores Booleano	30
Figura 12. Controles e Indicadores de cadena de caracteres	31
Figura 13. Ejemplo de un Diagrama de Bloques y Panel Frontal Correspondiente (1) Terminales de Indicador (2) Cables (3) Nodos (4) Terminales de Control	31
Figura 14. Diagrama de Bloques	32
Figura 15. Panel Frontal del Área de un Triángulo	33

Figura 16. Diagrama de Bloques del Área de un Triángulo con Vista de Terminal de Ícono (1) Controles (2) Indicador (3) Constante	34
Figura 17. Diagrama de Bloques del Área de un Triángulo sin Vista de Terminal de Ícono	35
Figura 18. Paleta de Funciones	37
Figura 19. Diagrama de adquisición de datos	39
Figura 20. Conexión del módulo DAQ USB 6008	40
Figura 21. Conexión del módulo a la PC	40
Figura 22. Asistente de instalación de hardware nuevo	41
Figura 23. Ventana de ubicación de driver	42
Figura 24. Ventana indicando el estado de instalación.	43
Figura 25. Ventana Measurements and Automation Explorer.	44
Figura 26. Measurements and Automation Explorer reconociendo el DAQ 6008	44
Figura 27. Selección de menú Self-Test	45
Figura 28. Autodiagnóstico del DAQ exitoso	45
Figura 29. Abriendo labview	46
Figura 30. Creando un nuevo proyecto en labview	46
Figura 31. Ubicación de las funciones del DAQ	47
Figura 32. Creación de la variable	48
Figura 32. Configuración inicial para leer un voltaje.	48

Figura 33. Especificando el canal de entrada para el voltaje del DAQ	49
Figura 34. Configuración del nombre y de los voltajes límites.	49
Figura 35. Creación de la variable	50
Figura 36. Configuración de la salida digital del DAQ	50
Figura 37. Configuración de la salida digital a través P0.0	51
Figura 38. Configuración del puerto de salida digital	51
Figura 39. Programación de VI para el monitoreo y control posición de un motor dc	52
Figura 40. Diagrama a bloques del diseño de VI del monitoreo y control posición de un motor dc	52
Figura 41. Diagrama de flujo del diseño de VI del monitoreo y control posición de un motor dc	53
Figura 41. Tabla del CI L293D	55
Figura 42. Diagrama a bloques del diseño de VI del monitoreo y control posición de un motor dc	56
Figura 43. Ejecución del VI a 0 grados	56
Figura 44. Ejecución del VI a 90 grados	57
Figura 44. Ejecución del VI a 180 grados	57

LISTA DE ANEXOS

	Pág.
Anexo A. Diagrama eléctrico del control de giro del motor DC	64

GLOSARIO

Bit: un dígito que forma parte del sistema binario. A diferencia del sistema decimal, que utiliza diez dígitos (0, 1, 2, 3, 4, 5, 6, 7, 8 y 9), el sistema binario apela a sólo dos (0 y 1). Un bit, por lo tanto, puede representar a uno de estos dos valores (0 ó 1).

Conversor o convertidor de señal analógica a digital: (Conversor Analógico Digital, CAD; Analog-to-Digital Converter, ADC) es un dispositivo electrónico capaz de convertir una señal analógica, ya sea de tensión o corriente, en una señal digital mediante un cuantificador y codificándose en muchos casos en un código binario en particular.

Driver: es un programa informático que permite al sistema operativo interactuar con un periférico, haciendo una abstracción del hardware y proporcionando una interfaz (posiblemente estandarizada) para utilizar el dispositivo

E/S (entrada/salida): Se utiliza para hablar de la cantidad de entradas y salidas que necesitan para un sistema, o el número de entradas y salidas que un determinado controlador lógico programable puede manejar.

Servomotor: (también llamado servo) es un dispositivo similar a un motor de corriente continua que tiene la capacidad de ubicarse en cualquier posición dentro de su rango de operación, y mantenerse estable en dicha posición

Potenciómetro: es un dispositivo conformado por 2 resistencias en serie, las cuales poseen valores que pueden ser modificados por el usuario

Puente en H: es un circuito electrónico que permite a un motor eléctrico DC girar en ambos sentidos, avance y retroceso

PWM: La modulación por ancho de pulsos (también conocida como PWM, siglas en inglés de pulse-width modulation) de una señal o fuente de energía es una técnica en la que se modifica el ciclo de trabajo de una señal periódica (una

senoidal o una cuadrada, por ejemplo), ya sea para transmitir información a través de un canal de comunicaciones o para controlar la cantidad de energía que se envía a una carga.

USB: Universal Serial Bus (Bus Universal en Serie) es un puerto diseñado para conectar varios periféricos a una computadora.

RESUMEN

Se realiza la implementación del monitoreo y control posición de un motor dc del módulo M2CI, con el objetivo de aprender a manipular la variable del giro de un motor DC con ayuda del DAQ UBS 6008.

Para poder controlar el control de giro del motor de DC fue necesario investigar los dispositivos involucrados para tal fin; se realizó la descarga de programas para poder realizar el correcto funcionamiento y la calibración de los instrumentos. Se descargó el software Labview y NIMax los cuales se complementaron para poder hacer las pruebas respectivas de adquisición y procesamiento de datos, para posteriormente realizar el control del motor DC.

Por medio del programa de Labview y su interfaz gráfica nos ayudó a programar y procesar los datos adquiridos por el DAQ 6008 todas las variables ya sea de forma analógica o digital.

1. INTRODUCCIÓN

El presente trabajo de tesis es para recibir el grado de Ingeniería Electrónica de la Universidad Nacional Abierta y A Distancia (UNAD). En este trabajo se pondrá en práctica los conocimientos estudiados durante a lo largo de la carrera, donde enfatizaremos en la adquisición de datos y su procesamiento de forma digital por medio de un DAQ USB

Hoy en día y gracias a los avances tecnológicos podemos adquirir mas fácil cualquier variable física y procesarla de forma digital según a nuestras necesidades, colocar límites y parámetros operacionales por medio de la programación del Labview permitiendo reducir tiempo en proceso, minimizar la posibilidad de que se generen errores.

En la Universidad Nacional Abierta y a Distancia (UNAD) tiene cuatro módulos de monitoreo y control de instrumentación (M2CI). El módulo M2CI consta del PLC S7-200 y un DAQ USB 6008 y una serie de dispositivos electrónicos que sirven para simular cuatro procesos industriales entre los cuales tenemos (Temperatura, nivel, posición y conteo), los cuales nos ayudaran a desarrollar las habilidades, y competencias.

El principal objetivo está en el diseño, programación e implementación del monitoreo y control de giro de un motor DC donde se observara proceso mediante una interfaz gráfica y de forma física todo esto por medio del programa Labview.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

- Desarrollar e implementar un modelo experimental de monitoreo y control posición de un motor DC que está instalado en el módulo de Monitoreo y Control de Instrumentación (M2CI) que permita operar y controlar la planta piloto de control de posición mediante acceso remoto.

2.2 OBJETIVOS ESPECÍFICOS

- Identificar los diferentes dispositivos y su funcionamiento de la planta piloto del módulo M2CI referente al monitoreo y control de posición de un motor D.C.
- Diseñar una práctica de laboratorio con el DAQ USB 6008 para el monitoreo y control de posición de un motor D.C.
- Desarrollar un instrumento virtual (VI) en LabView para monitorear y controlar la posición de un motor DC de la planta piloto del módulo M2CI.
- Adecuar las conexiones a las tarjetas desarrolladas en el proyecto PIE 01-15

3 PLANTEAMIENTO DEL PROBLEMA

3.1 DEFINICIÓN DEL PROBLEMA

La Universidad Nacional Abierta y a Distancia (UNAD) posee un módulo de Sistema de monitoreo y control de instrumentación industrial asistido por PC (M2CI), el cual cuenta con cuatro (4) plantas pilotos, dentro de ellas está la planta de posición.

Tal como lo mencionan los ingenieros Rodríguez y Peñarete [1] en su tesis de grado: “La medición y control de movimiento de rotación es una de las aplicaciones más importantes en el control moderno, instrumentación y tecnologías informáticas. También es utilizada en el control de posición, velocidad y aceleración, en una, dos o tres dimensiones, y en muchos sistemas en los que estos parámetros deben ser sensados y/o controlados. Casi todas las máquinas, procesos y sistemas de monitoreo tienen un eje de rotación en algún lugar de su mecanismo. Por esta razón, los transductores de movimiento de rotación son elementos de gran importancia en la ingeniería moderna [2]”

El uso de dicha planta piloto, se debe articular con el proceso formativo que brinda la UNAD a sus estudiantes, en especial a los de los programas académicos de Ingeniería Electrónica y en Ingeniería de Telecomunicaciones. Cuando se requiere de profundizar en sistemas de control, instrumentación y automatización.

La planta de control de posición está instalada de forma que se requiere de realizar de forma manual su cableado físico para poder implementar diferentes prácticas en ella. De otra parte, el módulo posee una tarjeta de adquisición de datos DAQ USB 6008 (TAD de National Instruments – NI). Con la cual se podría usar para la realización de controles, al permitir la toma de datos en tiempo real. Asimismo, puede ser usada como actuador en el momento de realizar el control. Esta tarjeta, es compatible y del mismo fabricante del software LabView, con ello es posible la concepción, el diseño, la implementación y la operación de la planta de control de posición.

No existe de una parte la documentación necesaria para la realización de prácticas en dicha planta, como los desarrollo de los instrumentos virtuales (VI) bajo el software de LabView. De otra parte, se hace necesaria la creación de las

curvas para la calibración de la planta piloto y su posterior adecuación a las diversas prácticas que allí se puedan desarrollar.

3.2 JUSTIFICACIÓN

El desarrollo del proyecto es viable por contar con los recursos de componentes electrónicos como de conocimiento de los estudiantes de los programas de ingeniería electrónica, telecomunicaciones, de las Tecnologías en Automatización electrónica y de menor forma en Ingeniería Industrial. El Modulo M2CI está disponible ya que la Universidad tiene en el laboratorio uno de ellos, y también cuenta con el software disponible.

El proyecto permitirá la integración de esta planta piloto a los desarrollos ya realizado para la virtualización y el uso remoto del módulo.

En un ambiente de educación a distancia, aún más, en ambientes virtuales de aprendizaje, el contar con laboratorios reales que permitan el acceso de forma remota y el uso de simuladores, potencializan los procesos académicos de sus estudiantes. La UNAD es pionera en estos escenarios y ha favorecido la investigación e implementación de desarrollos tecnológicos.

El proyecto, en su componente financiero, no demandará mayores recursos, salvos en dispositivos que comercialmente son muy accesibles a nuestras economías.

4. MARCO TEÓRICO

En el marco conceptual y teórico nos centraremos en dos partes, la primera partes son los dispositivos que compone la práctica de laboratorio que están incluidos dentro del módulo M2CI de la parte de la planta piloto referente al monitoreo y control de la posición de un motor DC, la otra parte se encarga del desarrollo de la práctica para la aplicación de los conceptos adquirido en la teoría. Donde todo nos permitirá que el estudiante pueda fortalecer y verificar los diferentes procesos de aprendizaje y poder profundizar a cabalidad en esta parte del módulo.

Ya que contamos con el modulo y los elementos que lo conforma la práctica se hace más específica el desarrollo de ella, se permite tener acceso más rápido a las practicas se refleja en un ahorro de tiempo y de espacio, se proporciona una guía adecuada del correcto funcionamiento de la planta piloto para tanto para los equipos como las personas involucradas.

La experimentación es una parte fundamental en los procesos formativos e investigativos en las áreas de la ingeniería. La apropiación del conocimiento se plantea en una movilidad del pensamiento entre la interpretación de las teorías y su aplicación a situaciones reales. Es por ello, que en el desarrollo de los procesos de aprendizaje en las ingenierías el componente práctico se hace indispensable. Pero este debe tener múltiples significados [3] desde el estudiante: Que sea un elemento motivador, que sea generador de nuevas técnicas tanto para el diseño de soluciones como para la construcción de modelos experimentales; que de una u otra forma favorezca las denominadas “actitudes científicas” y contribuya al fortalecimiento y comprensión de los conceptos científicos, y, por último; que la práctica a realizar sea coherente con el ejercicio profesional.

4.1 DAQ USB 6008 (TAD de National Instruments – NI).



Figura 1. DAQ USB 6008

Fuente: <http://www.ni.com/es-co/support/model.usb-6008.html>

[4] El USB-6008 brinda funcionalidad DAQ básica para aplicaciones como registro de datos simple, medidas portátiles y experimentos académicos de laboratorio. Es accesible para uso de estudiantes, pero lo suficientemente poderoso para aplicaciones de medida más sofisticadas. Utilice el USB-6008 que incluye el software registrador de datos para empezar a tomar medidas básicas en minutos o prográmelo usando LabVIEW o C y el software de servicios de medida NI-DAQmx Base para un sistema de medida personalizado.

Para cursos suplementarios con experimentos prácticos sobre teoría de simulación, medida y automatización, NI desarrolló un Kit de Estudiante USB-6008 que incluye una copia de LabVIEW Edición de Estudiante. Estos paquetes son exclusivamente para estudiantes, proporcionándoles una herramienta de aprendizaje potente, práctica y a bajo costo. Visite la página de productos académicos de NI para más detalles.

Características del DAQ USB 6008

- 8 entradas analógicas (12 bits, 10 kS/s)
- 2 salidas analógicas estáticas (12 bits); 12 E/S digitales; contador de 32 bits
- Energizado por bus para una mayor movilidad, conectividad de señal integrada
- La versión OEM está disponible
- Compatible con LabVIEW, LabWindows™/CVI y Measurement Studio para Visual Studio .NET
- Considere el USB-6001 para resolución de 14 bits y velocidades de muestreo más altas.

4.2 Servomotor

[5] Los servos son un tipo especial de motor de c.c. que se caracterizan por su capacidad para posicionarse de forma inmediata en cualquier posición dentro de su intervalo de operación. Para ello, el servomotor espera un tren de pulsos que se corresponde con el movimiento a realizar. Están generalmente formados por un amplificador, un motor, un sistema reductor formado por ruedas dentadas y un circuito de realimentación, todo en una misma caja de pequeñas dimensiones. El resultado es un servo de posición con un margen de operación de 180° aproximadamente.

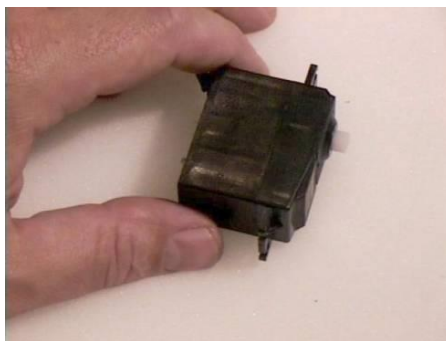


Figura 2. Servomotor de D.C.

Fuente:http://platea.pntic.mec.es/vgonzale/cyr_0204/ctrl_rob/robotica/sistema/motores_servo.htm

Se dice que el servo es un dispositivo con un eje de rendimiento controlado ya que puede ser llevado a posiciones angulares específicas al enviar una señal codificada. Con tal de que exista una señal codificada en la línea de entrada, el servo mantendrá la posición angular del engranaje. Cuando la señal codificada cambia, la posición angular de los piñones cambia.

La modulación por anchura de pulso, PWM (Pulse Width Modulation), es una de los sistemas más empleados para el control de servos. Este sistema consiste en generar una onda cuadrada en la que se varía el tiempo que el pulso está a nivel alto, manteniendo el mismo período (normalmente), con el objetivo de modificar la posición del servo según se desee.

Para la generación de una onda PWM en un microcontrolador, lo más habitual es usar un timer y un comparador (interrupciones asociadas), de modo que el microcontrolador quede libre para realizar otras tareas, y la generación de la señal sea automática y más efectiva. El mecanismo consiste en programar el timer con el ancho del pulso (el período de la señal) y al comparador con el valor de duración del pulso a nivel alto. Cuando se produce una interrupción de overflow del timer, la subrutina de interrupción debe poner la señal PWM a nivel alto y cuando se produzca la interrupción del comparador, ésta debe poner la señal PWM a nivel bajo.

El sistema de control de un servo se limita a indicar en qué posición se debe situar. Esto se lleva a cabo mediante una serie de pulsos tal que la duración del pulso indica el ángulo de giro del motor. Cada servo tiene sus márgenes de operación, que se corresponden con el ancho del pulso máximo y mínimo que el

servo entiende. Los valores más generales se corresponden con pulsos de entre 1 ms y 2 ms de anchura, que dejarían al motor en ambos extremos (0° y 180°). El valor 1.5 ms indicaría la posición central o neutra (90°), mientras que otros valores del pulso lo dejan en posiciones intermedias. Estos valores suelen ser los recomendados, sin embargo, es posible emplear pulsos menores de 1 ms o mayores de 2 ms, pudiéndose conseguir ángulos mayores de 180°. Si se sobrepasan los límites de movimiento del servo, éste comenzará a emitir un zumbido, indicando que se debe cambiar la longitud del pulso. El factor limitante es el tope del potenciómetro y los límites mecánicos constructivos.

El período entre pulso y pulso (tiempo de OFF) no es crítico, e incluso puede ser distinto entre uno y otro pulso. Se suelen emplear valores ~ 20 ms (entre 10 ms y 30 ms). Si el intervalo entre pulso y pulso es inferior al mínimo, puede interferir con la temporización interna del servo, causando un zumbido, y la vibración del eje de salida. Si es mayor que el máximo, entonces el servo pasará a estado dormido entre pulsos. Esto provoca que se mueva con intervalos pequeños.

Es importante destacar que para que un servo se mantenga en la misma posición durante un cierto tiempo, es necesario enviarle continuamente el pulso correspondiente. De este modo, si existe alguna fuerza que le obligue a abandonar esta posición, intentará resistirse. Si se deja de enviar pulsos (o el intervalo entre pulsos es mayor que el máximo) entonces el servo perderá fuerza y dejará de intentar mantener su posición, de modo que cualquier fuerza externa podría desplazarlo.

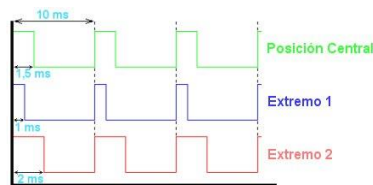


Figura 3. Tren de pulsos para control del servo

Fuente: <http://diymakers.es/control-velocidad-y-sentido-de-motor-dc/>

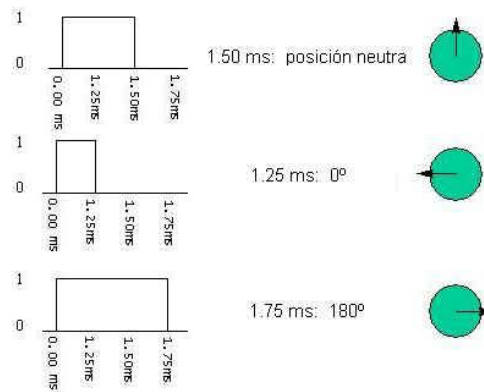


Figura 4. Otra posibilidad de pulsos de control

Fuente: <http://diymakers.es/control-velocidad-y-sentido-de-motor-dc/>

4.3 Tarjeta de control de posición del servo motor.

La tarjeta de control de posición del servo motor está constituida principalmente por un circuito integrado L293 el cual es un es un driver de potencia para motores paso a paso bipolares o un sistema puente en H [6] Es un sistema para controlar el sentido de giro de un motor DC usando cuatro transistores. En la imagen vemos que los transistores se comportan como interruptores y dependiendo que transistores conducen y cuáles no cambia la polarización del motor, y con esto el sentido de giro.

El sistema puente en H [7] se construye con 4 interruptores (mecánicos o mediante transistores). Cuando los interruptores S1 y S4 (figura 5) están cerrados (y S2 y S3 abiertos) se aplica una tensión positiva en el motor, haciéndolo girar en un sentido. Abriendo los interruptores S1 y S4 (y cerrando S2 y S3), el voltaje se invierte, permitiendo el giro en sentido inverso del motor.

Con la nomenclatura que estamos usando, los interruptores S1 y S2 nunca podrán estar cerrados al mismo tiempo, porque esto cortocircuitaría la fuente de tensión. Lo mismo sucede con S3 y S4. Figura 6

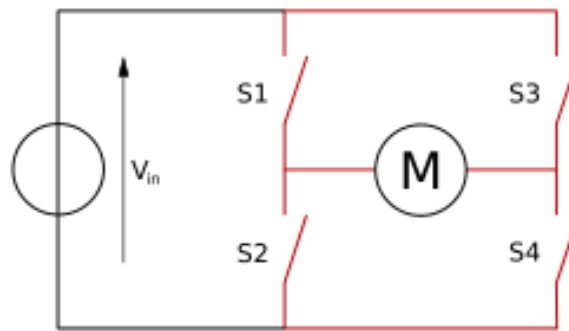


Figura 5. Estructura de un puente H.

Fuente: [https://es.wikipedia.org/wiki/Puente_H_\(electr%C3%B3nica\)](https://es.wikipedia.org/wiki/Puente_H_(electr%C3%B3nica))

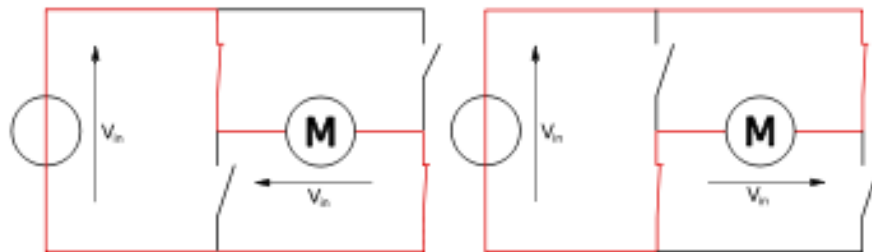


Figura 6. Los 2 estados básicos del circuito.

Fuente: [https://es.wikipedia.org/wiki/Puente_H_\(electr%C3%B3nica\)](https://es.wikipedia.org/wiki/Puente_H_(electr%C3%B3nica))

Como hemos dicho el puente H se usa para invertir el giro de un motor, pero también puede usarse para frenarlo (de manera brusca), al hacer un corto entre las bornes del motor, o incluso puede usarse para permitir que el motor frene bajo su propia inercia, cuando desconectamos el motor de la fuente que lo alimenta. En el siguiente cuadro se resumen las diferentes acciones. Tabla 1.

S1	S2	S3	S4	Resultado
1	0	0	1	El motor gira en <i>avance</i>
0	1	1	0	El motor gira en <i>retroceso</i>
0	0	0	0	El motor se detiene bajo su inercia

1	0	1	0	El motor frena (<i>fast-stop</i>)
0	1	0	1	El motor frena (<i>fast-stop</i>)
1	1	0	0	Corto circuito
0	0	1	1	Corto circuito
1	1	1	1	Corto circuito

Tabla 1. Estado de funcionamiento del puente en H

Fuente: [https://es.wikipedia.org/wiki/Puente_H_\(electr%C3%B3nica\)](https://es.wikipedia.org/wiki/Puente_H_(electr%C3%B3nica))

4.4. Potenciómetro lineal.

[8] Un potenciómetro es un resistor eléctrico con un valor de resistencia variable y generalmente ajustable manualmente. Los potenciómetros utilizan tres terminales y se suelen utilizar en circuitos de poca corriente, para circuitos de mayor corriente se utilizan los reostatos. En muchos dispositivos eléctricos los potenciómetros son los que establecen el nivel de salida. Por ejemplo, en un altavoz el potenciómetro ajusta el volumen; en un televisor o un monitor de ordenador se puede utilizar para controlar el brillo.

El valor de un potenciómetro viene expresado en ohmios (símbolo Ω) como las resistencias, y el valor del potenciómetro siempre es la resistencia máxima que puede llegar a tener. El mínimo lógicamente es cero. Por ejemplo un potenciómetro de $10K\Omega$ puede tener una resistencia variable con valores entre 0Ω y 10.000Ω .

Variación lineal

La resistencia es directamente proporcional al ángulo de giro. Figura 7.

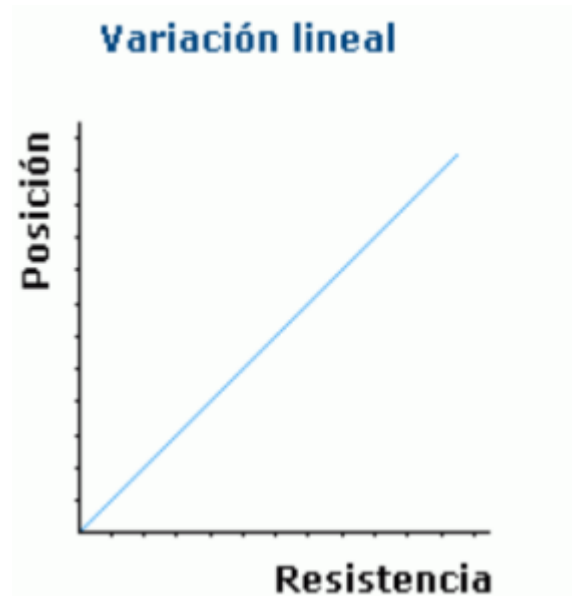


Figura 7. Gráfica de potenciómetro lineal

Fuente: <http://www.ingmecafenix.com/electronica/potenciometro/>

4.5 Labview

[9] Los programas de LabVIEW son llamados instrumentos virtuales o VIs ya que su apariencia y operación generalmente imitan a los instrumentos físicos, como osciloscopios y multímetros. LabVIEW contiene una extensa variedad de herramientas para adquirir, analizar, visualizar y almacenar datos, así como herramientas para ayudarle a solucionar problemas en el código que escriba. Cuando crea un nuevo VI, ve dos ventanas: la ventana del panel frontal y el diagrama de bloques.

4.5.1 Panel Frontal

Cuando abre un VI nuevo o existente, aparece la ventana del panel frontal del VI. La ventana del panel frontal es la interfaz de usuario para el VI. La Figura 8 muestra un ejemplo de una ventana del panel frontal.

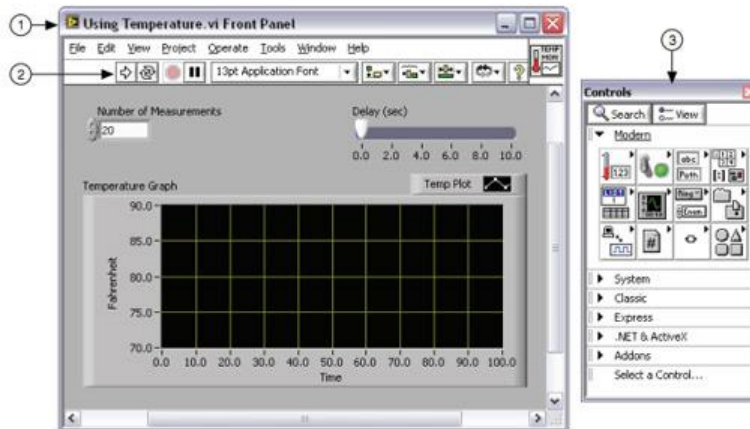


Figura 8. Ejemplo de un Panel Frontal (1) Ventana de Panel Frontal | (2) Barra de Herramientas | (3) Paleta de Controles

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

4.5.2 Paleta de Controles

La paleta de Controles contiene los controles e indicadores que utiliza para crear el panel frontal. Puede tener acceso a la paleta de Controles de la ventana del panel frontal al seleccionar View»Controls Palette o al dar clic con botón derecho en cualquier espacio en blanco en la ventana del panel frontal. La paleta de Controles está dividida en varias categorías; puede exponer algunas o todas estas categorías para cumplir con sus necesidades. La Figura 9 muestra la paleta de Controles con todas las categorías expuestas y la categoría Moderna expandida.

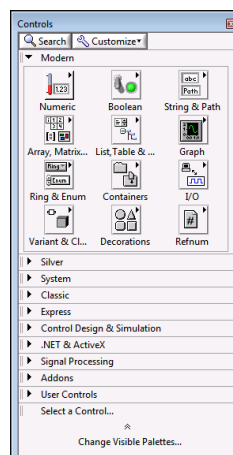


Figura 9. Paleta de Controles

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

Para ver o esconder categorías (subpaletas), haga clic en el botón Customize y seleccione Change Visible Palettes.

4.5.3 Controles e Indicadores

Cada VI tiene un panel frontal que usted puede diseñar como una interfaz de usuario. Usted también puede usar paneles frontales como la manera de transmitir entradas y recibir salidas cuando se llama al VI desde otro diagrama de bloque. Usted crea la interfaz de usuario de un VI al colocar controles e indicadores en el panel frontal de un VI. Cuando interactúa con un panel frontal como una interfaz de usuario, puede modificar los controles para transmitir entradas y ver los resultados en los indicadores. Los controles que definen las entradas y los indicadores muestran las salidas.

Los controles generalmente son perillas, botones, perillas, deslizadores y secuencias. Simulan dispositivos de entrada de instrumentos y suministran datos al diagrama de bloques del VI. Los indicadores generalmente son gráficas, tablas, LEDs y secuencias de estado. Los indicadores simulan dispositivos de salida de instrumentos y muestran los datos que diagrama de bloques adquiere o genera. La Figura 10 tiene dos controles: Number of Measurements y Delay (sec). Tiene un indicador: una gráfica XY llamada Temperature Graph.

El usuario puede cambiar el valor de entrada para los controles Number of Measurements and Delay (sec). El usuario puede ver el valor generado por el VI en el indicador Temperature Graph. El VI genera los valores para los indicadores basados en el código creado en el diagrama de bloques.

Cada control o indicador tiene un tipo de datos asociado con él. Por ejemplo, el slide horizontal Delay (sec) es un tipo de datos numérico. Los tipos de datos utilizados con mayor frecuencia son numéricos, Booleano y cadena de caracteres.

Controles e Indicadores Numéricos

El tipo de datos numérico pueden representar números de varios tipos como un entero o real. Los dos objetos numéricos comunes son el control numérico y el indicador numérico, como se muestra en la Figura 10. Los objetos como medidores y perillas también representan datos numéricos.

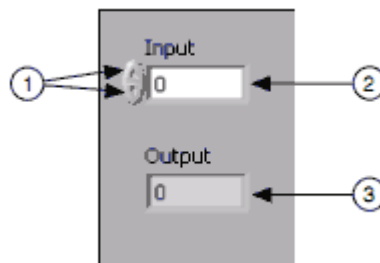


Figura 10. Controles e Indicadores Numéricos (1) Botones de Incremento/Reducción | (2) Control Numérico | (3) Indicador Numérico

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

Para proporcionar o cambiar valores en un control numérico, haga clic en el botón de incremento y reducción o haga doble clic en el número, introduzca un nuevo número y presione la tecla <Enter>.

4.5.4 Controles e Indicadores Booleano

El tipo de datos Booleano representa datos que solamente tienen dos estados posibles, como TRUE y FALSE u ON y OFF. Use los controles e indicadores Booleano para proporcionar y visualizar valores Booleano. Los objetos Booleano simulan interruptores, botones y LEDs. El interruptor de encendido vertical y los objetos LED redondos se muestran en la Figura 11.

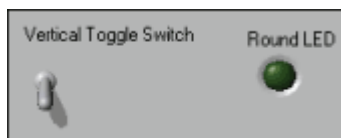


Figura 11. Controles e Indicadores Booleano

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

4.5.5 Controles e Indicadores de Cadena de Caracteres

El tipo de datos cadena de caracteres es una secuencia de caracteres ASCII. Use controles en cadena para recibir texto desde el usuario como una contraseña o nombre de usuario. Use indicadores en cadena para mostrar texto al usuario. Los objetos en cadena más comunes son tablas y cuadros de texto, como se muestra en la Figura 12.



Figura 12. Controles e Indicadores de cadena de caracteres

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

4.5.6 Diagrama de Bloques

Los objetos del diagrama de bloques incluyen terminales, subVIs, funciones, constantes, estructuras y cables, los cuales transfieren datos junto con otros objetos del diagrama de bloques.

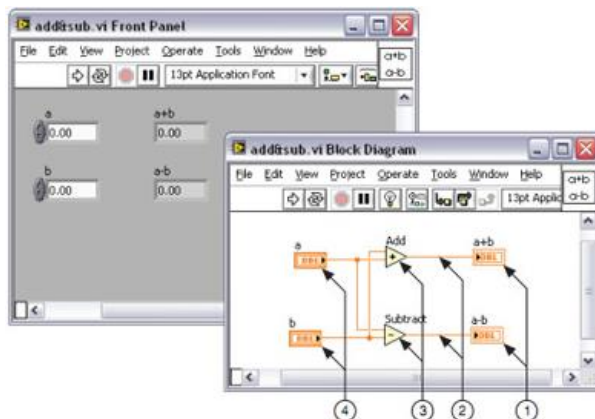


Figura 13. Ejemplo de un Diagrama de Bloques y Panel Frontal Correspondiente
 (1) Terminales de Indicador | (2) Cables | (3) Nodos | (4) Terminales de Control

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

Después de que crea la ventana del panel frontal, añada código usando representaciones gráficas de funciones para controlar los objetos del panel frontal. La ventana del diagrama de bloques contiene este código de fuente gráfica.

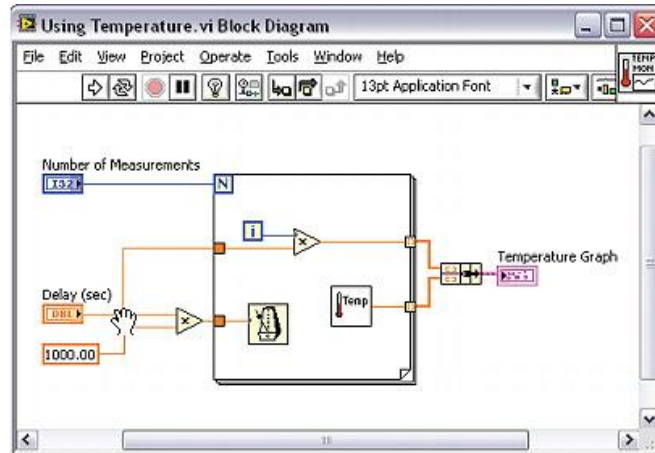


Figura 14. Diagrama de Bloques

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

4.5.7 Terminales

Los objetos en la ventana del panel frontal aparecen como terminales en el diagrama de bloques. Las terminales son puertos de entrada y salida que intercambian información entre el panel frontal y diagrama de bloques. Son análogos a parámetros y constantes en lenguajes de programación basados en texto. Los tipos de terminales incluyen terminales de control o indicador y terminales de nodo. Las terminales de control e indicador pertenecen a los controles e indicadores del panel frontal. Los puntos de datos que usted proporciona en los controles del panel frontal (a y b en el panel frontal anterior) pasan al diagrama de bloques a través de las terminales de control. Entonces los puntos de datos ingresan las funciones de Suma y Resta. Cuando las funciones de Suma y Resta terminan sus cálculos, producen nuevos valores de datos. Los valores de datos van a las terminales de indicador, donde actualizan los indicadores del panel frontal (a+b y a-b en el panel frontal anterior).

4.5.7 Controles, Indicadores y Constantes

Los controles, indicadores y constantes se comportan como entradas y salidas del algoritmo del diagrama de bloques. Considere la implementación del algoritmo para el área de un triángulo:

$$\text{Área} = .5 * \text{Base} * \text{Altura}$$

En este algoritmo, Base y Altura son entradas y Área es una salida, como se muestra en la Figura 15.

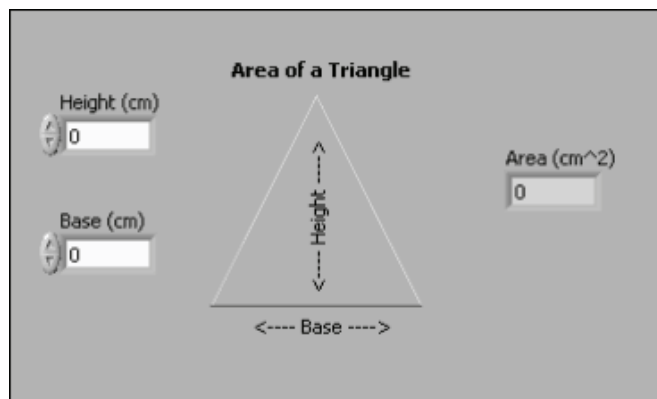


Figura 15. Panel Frontal del Área de un Triángulo

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

El usuario no cambiará o tendrá acceso a la constante 0.5, así que no aparecerá en el panel frontal a menos que se incluya como documentación del algoritmo. La Figura 16 muestra una posible implementación de este algoritmo en un diagrama de bloques de LabVIEW. Este diagrama de bloques tiene cuatro terminales diferentes creadas por dos controles, una constante y un indicador.

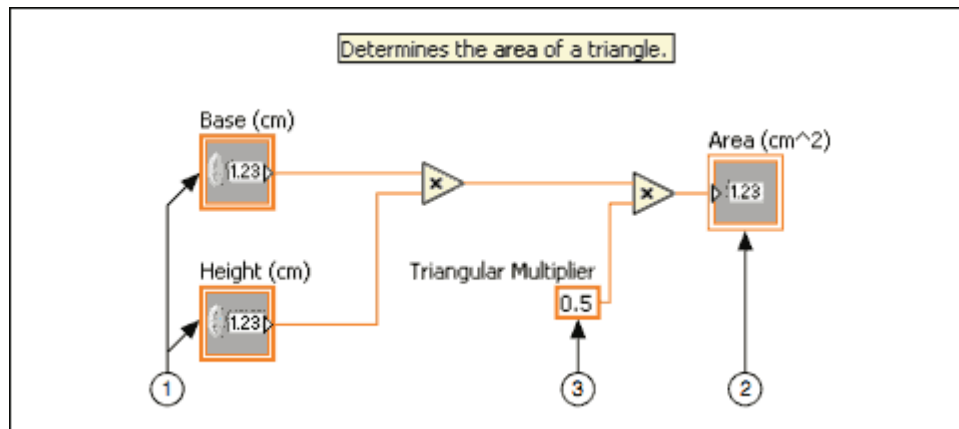


Figura 16. Diagrama de Bloques del Área de un Triángulo con Vista de Terminal de Ícono (1) Controles | (2) Indicador | (3) Constante

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

Note que las terminales del diagrama de bloques Base (cm) y Altura (cm) tienen una apariencia diferente de la terminal Área (cm²). Existen dos características distintivas entre un control y un indicador en el diagrama de bloques. La primera es una flecha en la terminal que indica la dirección del flujo de datos. Los controles tienen flechas que muestran los datos que salen de la terminal, mientras que el indicador tiene una flecha que muestra los datos que entran a la terminal. La segunda característica distintiva es el borde alrededor de la terminal. Los controles tienen un borde grueso y los indicadores tienen un borde delgado.

Puede ver terminales con o sin vista de íconos. La Figura 17 muestra el mismo diagrama de bloques sin usar la vista de ícono de las terminales; sin embargo existen las mismas características distintivas entre los controles y los indicadores.

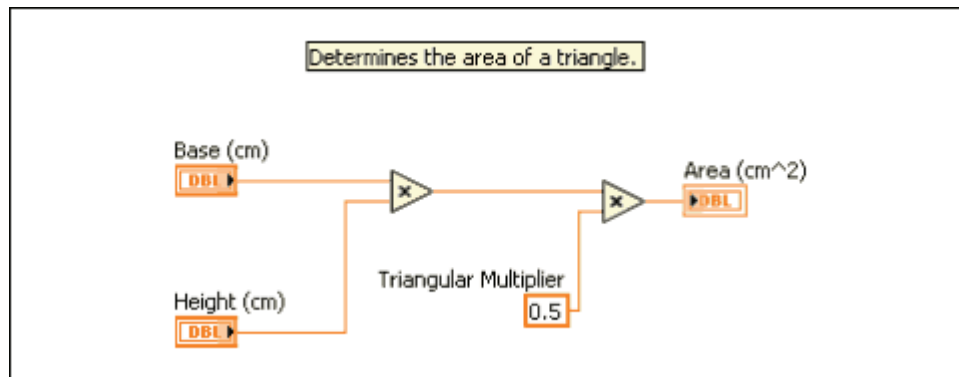


Figura 17. Diagrama de Bloques del Área de un Triángulo sin Vista de Terminal de Ícono

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

4.5.9 Nodos de Diagrama de Bloques

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

4.5.10 Funciones

Las funciones son los elementos de operación fundamentales de LabVIEW. Las funciones de Suma y Resta en la Figura 15 son nodos de función. Las funciones no tienen ventanas del panel frontal o ventanas del diagrama de bloques pero no tienen paneles conectores. Al hacer doble clic en la función solamente selecciona la función. Una función tiene un fondo amarillo pálido en su ícono.

4.5.11 SubVIs

Después de desarrollar un VI, usted puede usarlo en otro VI. Un VI llamado desde el diagrama de bloques de otro VI es llamado un subVI. Usted puede reutilizar un subVI en otros VIs. Para crear un subVI, necesita desarrollar un panel conector y crear un ícono.

Un nodo de subVI corresponde a una llamada de subrutina en lenguajes de programación basados en texto. El nodo no es subVI, solamente una instrucción de llamada de subrutina en un programa que no es la propia subrutina. Un diagrama de bloques que contiene varios nodos de subVI idénticos llama al mismo subVI varias veces.

Los indicadores y controles de subVI reciben datos y regresan datos al diagrama de bloques del VI que hace el llamado. Cuando hace doble clic en un subVI en el diagrama de bloques, aparece la ventana del panel frontal. El panel frontal incluye controles e indicadores. El diagrama de bloques incluye cables, íconos, funciones, subVIs probables u otros objetos de LabVIEW.

Cada VI muestra un ícono en la esquina superior derecha de la ventana del panel frontal y la ventana del diagrama de bloques. Un ejemplo del ícono predeterminado se muestra aquí. Un ícono es una representación gráfica de un VI. El ícono puede contener texto e imágenes. Si usa un VI como un subVI, el ícono identifica al subVI en el diagrama de bloques del VI. El ícono predeterminado contiene un número que indica cuántos nuevos VI abrió después de iniciar LabVIEW.

Para usar un VI como un subVI, necesita construir un panel conector, como se muestra arriba. El panel conector es un conjunto de terminales en el ícono que corresponde a los controles e indicadores de ese VI, similares a la lista de parámetros de una función llamada en lenguajes de programación basados en texto. Obtenga acceso al panel conector al dar clic con botón derecho en el ícono en la parte superior derecha de la ventana del panel frontal. Usted puede tener acceso al panel conector desde el ícono en la ventana del diagrama de bloques. Un ícono de subVI tiene un fondo blanco en su ícono.

4.5.12 Express Vis

Los Express VIs son nodos que requieren cableado mínimo ya que usted los configura con ventanas de diálogo. Use Express VIs para tareas de medidas comunes. Consulte el tema de Express VIs de la Ayuda de LabVIEW para más información. Aparecen en el diagrama de bloques como nodos expandibles con iconos rodeados por un campo azul.

4.5.13 Paleta de Funciones

La paleta de Funciones contiene los VIs, funciones y constantes que usted utiliza para crear el diagrama de bloques. Usted tiene acceso a la paleta de Funciones del diagrama de bloques al seleccionar View»Functions Palette. La paleta de Funciones está dividida en varias categorías; usted puede mostrar y esconder categorías para cumplir con sus necesidades. La Figura 20 muestra la paleta de Funciones con todas las categorías expuestas y la categoría de Programación expandida.

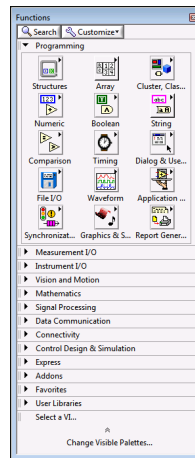


Figura 18. Paleta de Funciones

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/environment>

5. MATERIALES Y MÉTODOS

5.1 MATERIALES

RECURSO	DESCRIPCIÓN	PRESUPUESTO
Equipo Humano	Estudiante de la UNAD	0
Equipos y Software	Módulo M2CI, Software LabView y computador.	0
Viajes y Salidas de Campo	N/A	
Materiales y suministros	N/A	
Bibliografía	N/A	
TOTAL		

5.2 METODOLOGÍA

En el desarrollo del proyecto se usara un modelo experimental, donde se realiza a prueba y error, partiendo de los conocimientos previos de los diferentes dispositivos involucrados en la planta piloto del módulo M2CI para poder desarrollar la práctica del monitorio y control del motor de DC.

Existen prácticas realizadas en el módulo lo que nos garantiza el correcto funcionamiento y de los dispositivos que se involucraran la práctica, donde se realiza la implementación de un programa bajo la plataforma de Labview, por medio de la DAQ USB 6008 que recopilara los datos y nos mostrara también el resultado del programa desarrollado.

6 DESARROLLO DEL PROYECTO

El propósito de la adquisición de datos es medir un fenómeno eléctrico o físico en este caso un voltaje, donde se realiza por medio de diferentes dispositivos electrónicos que utiliza una combinación de hardware, software y un elemento para medir. Donde la adquisición de datos está definida para una aplicación específica, cada uno de las señales adquiridas requiere un análisis y una presentación de la información. Todo este proceso incorpora señales, sensores, motores, tarjeta de adquisición de datos y de un programa.

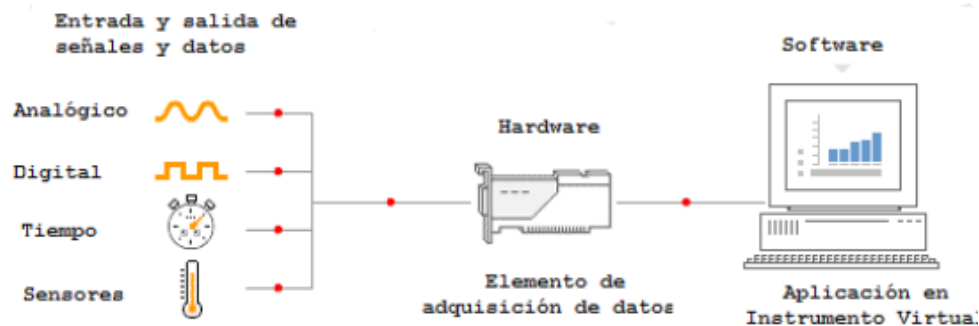


Figura 19. Diagrama de adquisición de datos

Fuente: <https://instrumentacionmecanica.files.wordpress.com/2011/12/prc3a1ctica-11-adquisicic3b3n-y-anc3a1lisis-de-datos-en-labview.pdf>

Para familiarizarnos con la adquisición de datos usamos el DAQ USB 6008 de la national instruments, que nos hará todo el proceso de procesamiento de datos de forma digital, y el paquete de labview nos ayuda a verlo de forma más amigable y también poder programar el dispositivo para que controle cada una de las variables tanto análogas como digitales conectadas a él.

Conectando nuestro dispositivo DAQ USB 6008 a la tarjeta de control de giro de un motor DC que está en el módulo, cual controla un motor DC de 5V DC, él cuenta con una configuración de un Puente H, que nos facilita el control de giro del motor ya sea hacia la derecha o izquierda.

Inicialmente la comunicación entre el DAQ USB 6008 y el PC equipado con el software NIMAX, debe hacerse a través de uno de los puertos USB. Se utiliza para

ello un cable apropiado de terminales USB B y USB A, como se muestra en la figura 20.



Figura 20. Conexión del módulo DAQ USB 6008

Fuente: <https://es.slideshare.net/FelixRamirezRamrezMo/trabajando-con-la-ni-usb>

Para asegurar la conexión entre el DAQ USB 6008 y el PC estén conectados como se mostró en la figura 20. Inicialmente la conexión del PC al DAQ se hace a través del puerto Usb, donde el instala el controlador del nuevo dispositivo a la computadora.

A continuación se describen en detalle los pasos necesarios para poder comunicarse con el DAQ.

Paso 1:

Asegúrese de que la computadora tenga instalado LabView. Si no es así, instálelo y también el NI DAQ mx, conecte la tarjeta DAQ al cable USB y este a algún puerto USB de la computadora. Aparecerá un globo de diálogo en la barra de tareas como se muestra en la siguiente figura 21:



Figura 21. Conexión del módulo a la PC

Fuente: Melendez, Alexis

Paso 2:

Posteriormente aparecerá el asistente de instalación de hardware nuevo. Seleccione la opción que evita que busque en Windows Update el controlador de la tarjeta y presione el botón de “siguiente” como se muestra en la siguiente figura 22:



Figura 22. Asistente de instalación de hardware nuevo

Fuente: Melendez, Alexis

Paso 3:

Luego aparecerá la ventana de ubicación de driver. En este caso seleccione instalar el programa automáticamente y de click en “Siguiente” como se muestra en la siguiente figura 23:



Figura 23. Ventana de ubicación de driver

Fuente: Melendez, Alexis

Paso 3:

Aparecerá una ventana indicando que la instalación se está llevando a cabo y luego confirmará que la instalación ha sido exitosa como se muestra en la siguiente figura 24:

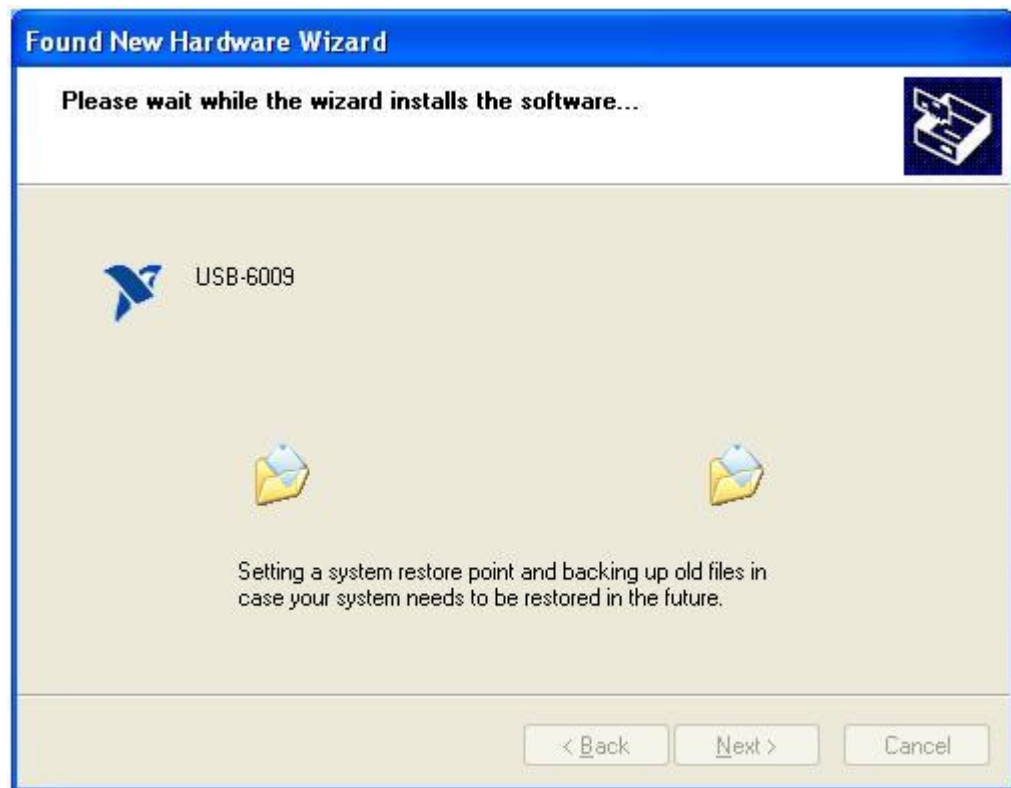


Figura 24. Ventana indicando el estado de instalación.

Fuente: Melendez, Alexis

Paso 4:

Ahora comprobamos la instalación del DAQ USB 6008 en la computadora, Ejecute el programa Measurements and Automation Explorer (NI MAX) que se encuentra en Menú Inicio >> Todos los programas >> National Instruments. Como se muestra en la siguiente figura 25:



Figura 25. Ventana Measurements and Automation Explorer.

Fuente: Melendez, Alexis

Paso 5:

Cuando inicie el programa, utilice la ventana de configuración para acceder al DAQ (My system > Devices and Interfaces > NI USB-6008) y seleccione el dispositivo. Si es el único elemento conectado a la computadora aparecerá como NI USB-6008: "Dev1", como se muestra en la figura 26.

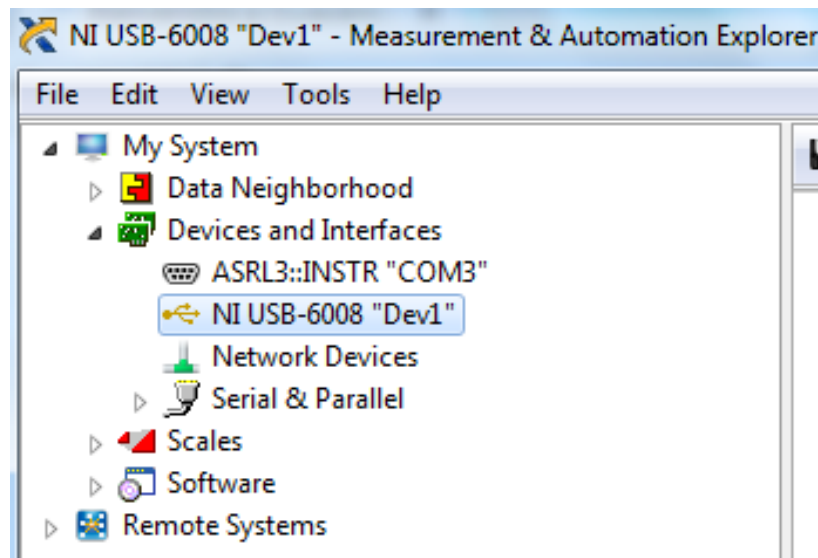


Figura 26. Measurements and Automation Explorer reconociendo el DAQ 6008

Fuente: Melendez, Alexis

Paso 6:

Al seleccionar el dispositivo, aparecerán nuevos menus, como se muestra en la figura 27. Se selecciona el menú **self-test**

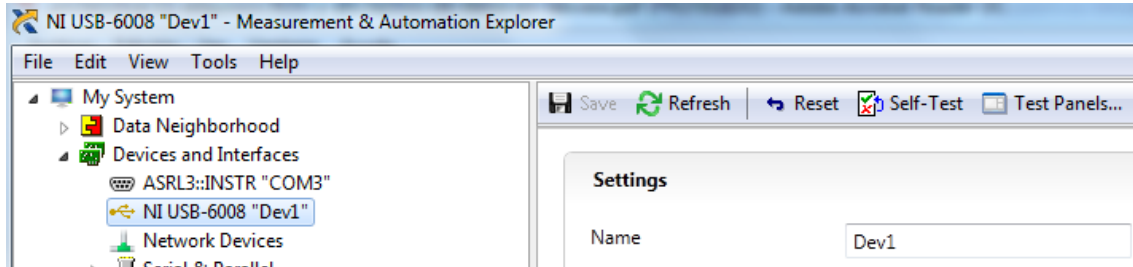


Figura 27. Selección de menú Self-Test

Fuente: Melendez, Alexis

Paso 7:

Si el DAQ no tiene problemas de comunicación con el PC le señalara con una ventana de verificación exitosa el mostrara en la siguiente figura 28.

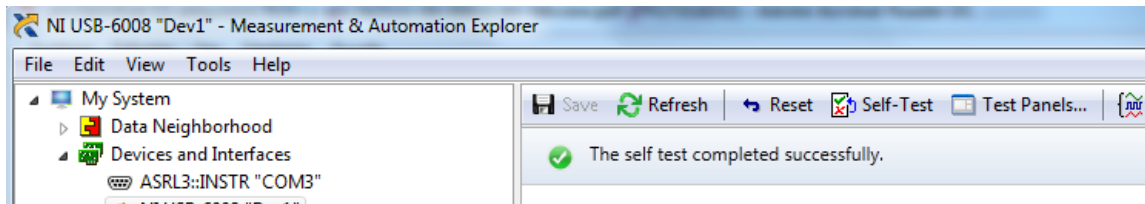


Figura 28. Autodiagnóstico del DAQ exitoso

Fuente: Melendez, Alexis

Paso 8:

Una vez que haya pasado el auto diagnóstico “Self Test” se procedemos a abrir LabView para hacer adquisición de datos y generación de señales que se

encuentra en Menú Inicio > todos los programas > National Instruments > LabView.

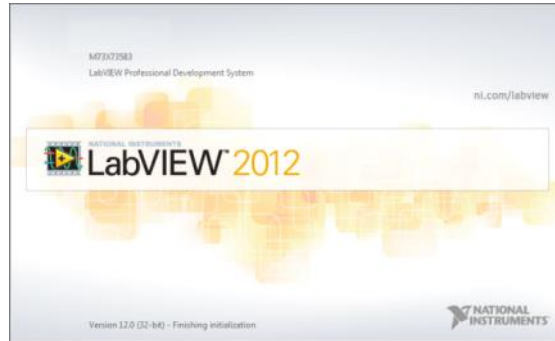


Figura 29. Abriendo labview

Fuente: Melendez, Alexis

Paso 9:

Ahora generamos un nuevo proyecto dentro de labview seleccionado la opción creative project como se muestra en la figura 30.



Figura 30. Creando un nuevo proyecto en labview

Fuente: Melendez, Alexis

Paso 10:

Al tener creado un nuevo VI nos dirigimos a buscar si esta la opción de DAQmx, donde está ubicado en (Functions Measurement I/O DAQmx-Data Acquisition; alojadas todas la funciones para el DAQ se encuentran en este submenú. Como se ilustra en la siguiente figura 31.

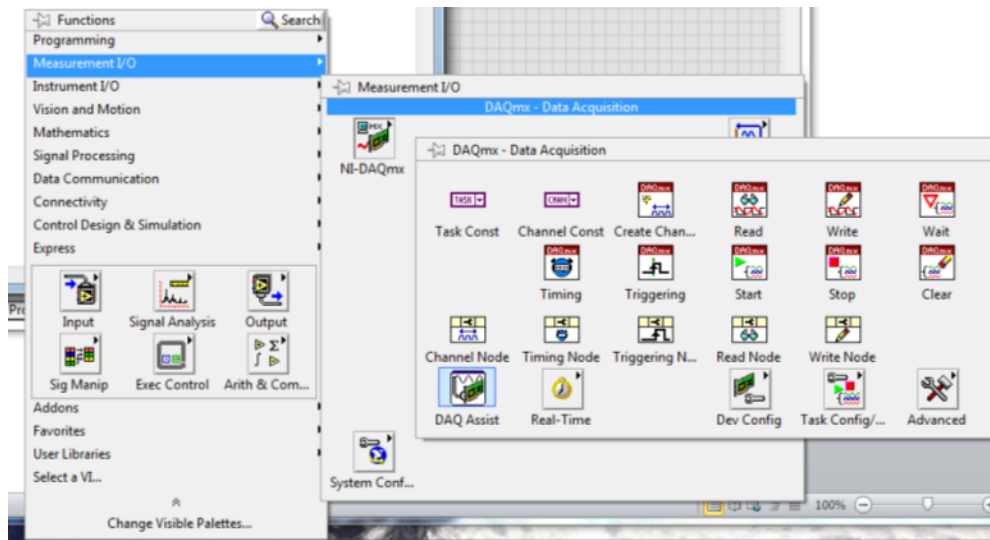


Figura 31. Ubicación de las funciones del DAQ

Fuente: Melendez, Alexis

En este punto podemos garantizar que tenemos bien configurado y funcionando nuestro DAQ USB 6008 a nuestra PC y que nos va adquirir y enviar datos a nuestro dispositivo y lo podemos apreciar dentro de cualquier VI que generemos

Para poder trabajar el DAQ dentro de un VI hay que saber cómo configurarlo en este caso se usó de forma análoga y de forma digital el cual se va explicar cómo configurarlo de cada manera.

Se usó la opción dentro del DAQmx esta la opción de DAQ Asistent el cual se selecciona y nos dará la siguiente opción. Como como se ilustra en la figura 32

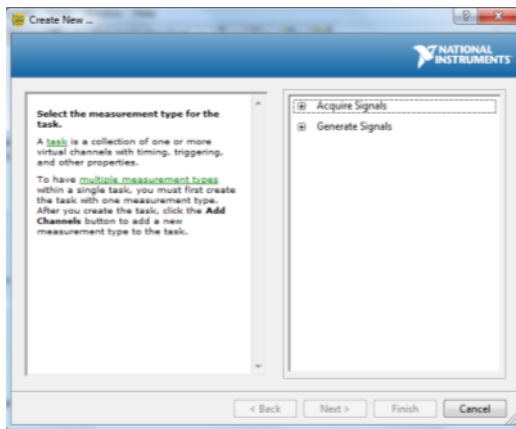


Figura 32. Creación de la variable

Fuente: Melendez, Alexis

Configura la tarea para adquirir voltaje (Acquire Signals / Analog Input / Voltage). Como ilustra en la figura 33.

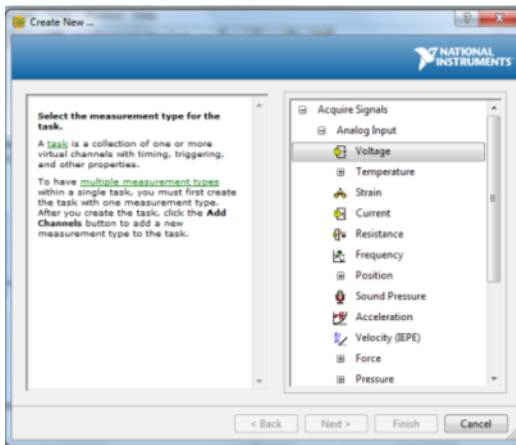


Figura 32. Configuración inicial para leer un voltaje.

Fuente: Melendez, Alexis

Selecciona la entrada ai0 y se presiona el botón Finish. Como ilustra en la figura 33

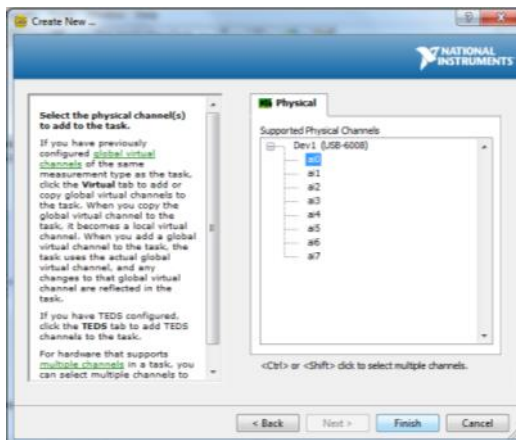


Figura 33. Especificando el canal de entrada para el voltaje del DAQ

Fuente: Melendez, Alexis

En la siguiente ventana se realiza la configuración, sobre la etiqueta, los límites asignados de voltaje donde el valor máximo es de 5 y el valor mínimo de 0 y luego se oprime OK para terminar el proceso. Como ilustra en la figura 34.

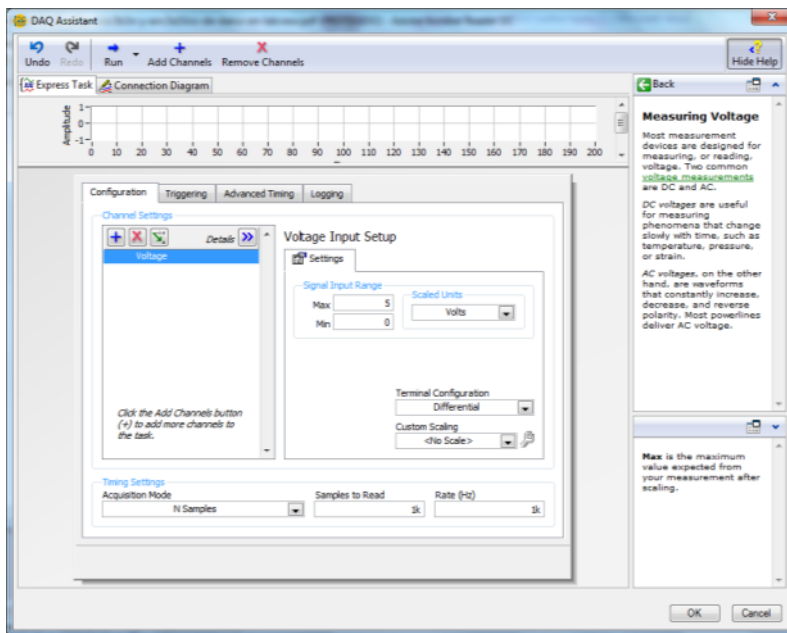


Figura 34. Configuración del nombre y de los voltajes límites.

Fuente: Melendez, Alexis

Realizamos la tarea para otro DAQ Asistent, pero para generar las señales digitales, Se usó la opción dentro del DAQmx está la opción de DAQ Asistent el cual se selecciona y nos dará la siguiente opción. Como como se ilustra en la figura 35

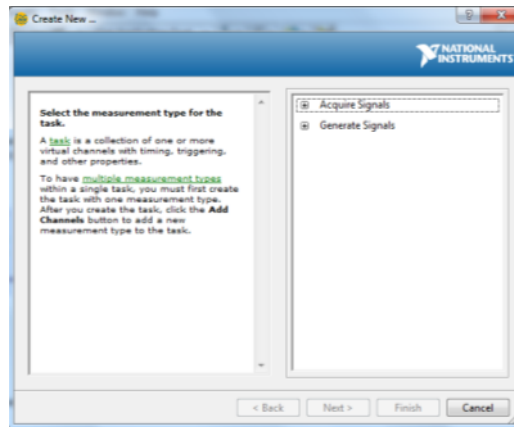


Figura 35. Creación de la variable

Fuente: Meléndez, Alexis

Configura la tarea para generar la señal de voltaje (Generate Signals / Digital Output / Line Output). Como ilustra en la figura 36.

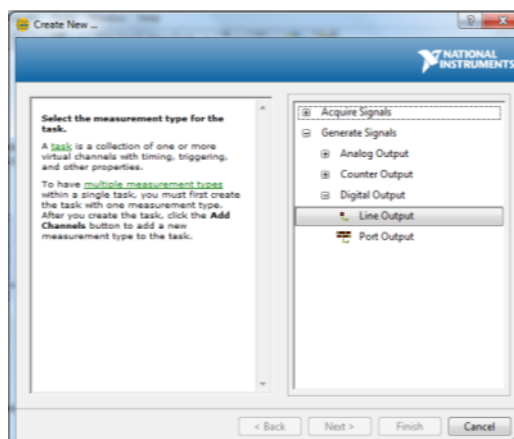


Figura 36. Configuración de la salida digital del DAQ

Fuente: Meléndez, Alexis

Se selecciona el canal deseado en este caso port0/Line0, como ilustra en la figura 37.

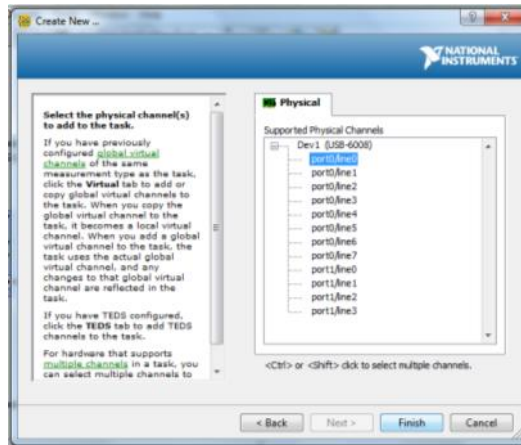


Figura 37. Configuración de la salida digital a través P0.0

Fuente: Meléndez, Alexis

En la siguiente ventana se realiza la configuración del puerto digital, se etiqueta, también si se desea agregar una nueva salida digital, y por ultimo le damos Ok para terminar el proceso. Como ilustra en la figura 38.

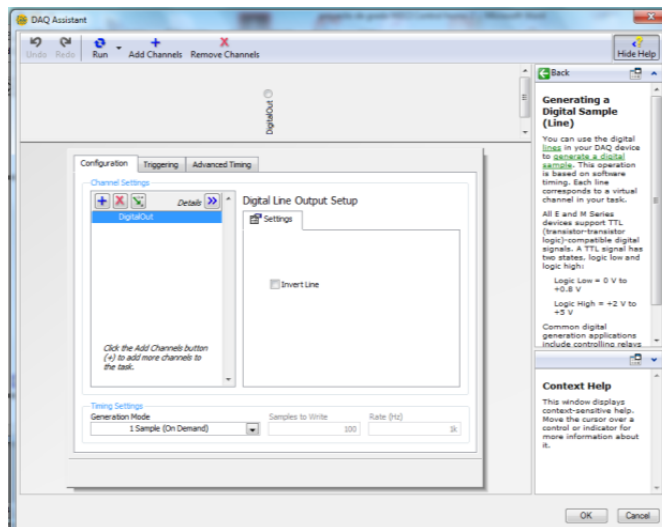


Figura 38. Configuración del puerto de salida digital

Fuente: Meléndez, Alexis

Después de realizar la comprobación de nuestro modulo tiene conectividad con la PC en nuestro caso se hizo el diseño del control por medio de labview el monitoreo y control posición de un motor dc que está en el módulo M2CI el cual se manipulo para que se programara el giro tanto a la izquierda como a la derecha y la mantuviera el grado dado ya sea manual o de forma digital, bajo la creación de un VI, como se ilustra en la figura 39.

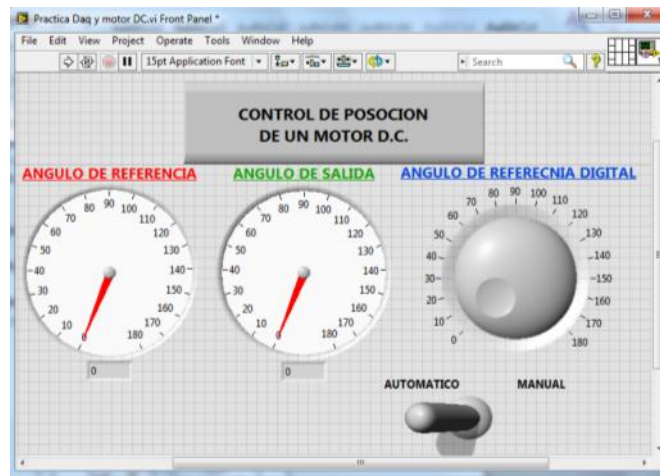


Figura 39. Programación de VI para el monitoreo y control posición de un motor dc

Fuente: Meléndez, Alexis

Ahora se abordara el diseño del diagrama a bloques que controla el VI que fue creación de mía y se explicara cómo fue diseñado este diagrama (figura 40) acompañado de un diagrama de flujo, se muestra en la figura 41.

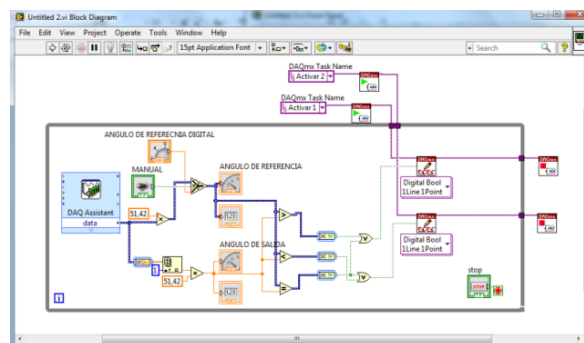


Figura 40. Diagrama a bloques del diseño de VI del monitoreo y control posición de un motor dc.

Fuente: Meléndez, Alexis

Se realizó la configuración del módulo DAQ por el bloque de DAQ assintant para recopilar los datos análogos de las entradas AI2 y AI3, por medio de una conexión múltiple en donde se lleva a un arreglo array para extraer el uno de los datos adquiridos y luego por medio de un reloj análogo y un display se muestra el valor del Angulo de salida, El cuenta con un interruptor que ayuda a seleccionar la entrada de dato del Angulo de referencia de forma análoga o digital que se escoge por medio de una compuerta con condición de verdad o falso; donde la opción de falso va conectada una perilla digital que está dentro del VI y la opción de Falso va conectada al valor que proviene del DAQ assintant, lo valores son mostrados un reloj análogo y un display.

Luego se realiza la comparación de los datos por medio de tres operaciones mayor que hace que se active una salida digital del DAQ en este caso se realiza por el bloque de programación de una salida digital del pin P1.1 y este activa el motor para un sentido, con la condición de menor que hace que se active una salida digital del DAQ en este caso se realiza por el bloque de programación de una salida digital del pin P1.2, igual por último el igual que se encarga de activar las dos salidas digitales del DAQ el P1.1 y P1.2 y frena el motor para que no siga girando más.

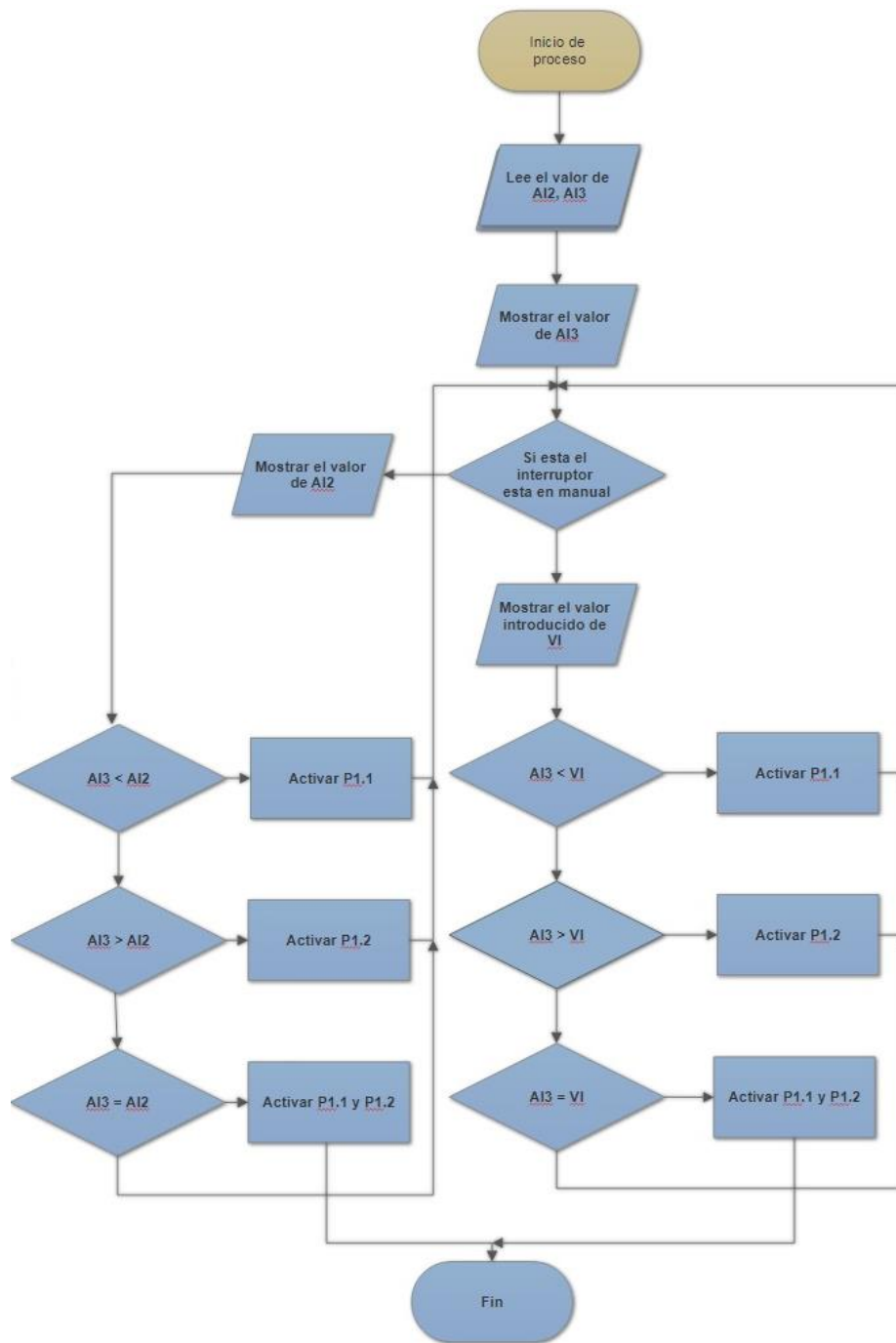


Figura 41. Diagrama de flujo del diseño de VI del monitoreo y control posición de un motor dc.

Fuente: Meléndez, Alexis

Para realizar el control el monitoreo y control de giro del motor de DC, se hizo con una configuración llamada Puente H, cual tiene una configuración que nos ayuda para el control de giro del motor sin causar daños o cortos en los equipos involucrados, la parte de monitoreo por medio de un par de potenciómetros lineales ubicados uno en el motor de DC y el otro de forma independiente para hacer su control manual.

Para realizar el control de giro del motor de DC se realzo por medio de unas comparaciones, obtenidas de unos valores adquiridos por medio del DAQ por sus entradas análogas conectadas a los potenciómetros, por medio del VI creado se pueden apreciar los valores en ángulos tanto el de referencia como el de salida.

Cuenta con un interruptor en el VI si desea controlar el ángulo de referencia de forma física o de forma digital apreci ando de forma didáctica y amigable los valores gracias a la interface gráfica que cuenta el software Labview.

Para poder realizar la selección de esta condición en la programación del VI se realizado por medio de una compuerta de condición de Falso o verdadero la cual se interpola de forma manual o de forma automática, así nos da en control sobre eso. Luego por medio del valor adquirido por las entradas análogas del DAQ se realiza la comparación del ángulo de referencia contra el ángulo de salida, donde se implementó tres condiciones su es mayor, si es menor y si es igual.

Si el valor del ángulo de referencia es mayor que le ángulo de salida el motor gira hacia la derecha, si el valor del ángulo de referencia es menor que le ángulo de salida el motor gira hacia la izquierda, y la última condición si es igual los dos ángulos el para el proceso de giro del motor indiferente si va hacia la izquierda o hacia la derecha.

Para poder activar el motor el sentido de giro hay que cumplir con una tabla de verdad que posee el circuito integrado L293D. Como se ilustra la figura 41. Y son activas por las salidas de los compradores de mayor qué y menor que; nos arroga esta misma tabla. Para la activación se hace por medio de la configuración de las salidas digitales que posee el DAQ para tal fin y llegue al a tarjeta de control del módulo M2CI de control de giro de un motor DC.

EN	1A	2A	FUNCCION
H	L	H	Gira a la derecha
H	H	L	Gira a la izquierda
H	L	L	Parada rápida del motor
H	H	H	Parada rápida del motor
L	X	X	Parada rápida del motor

L = low, H = high, X = don't care

Figura 42. Tabla del CI L293D

Fuente: <http://aprendiendo.laconeccion.com/l293d>

Para el parado del motor se hizo por medio de la implementación de una compuerta OR antes de las salidas digitales del DAQ que hace colocar un uno par que pare el motor cumpliendo con su objetivo principal de frenado de giro. Como se ilustra en la figura 41

A = Respuesta de la comparación mayor que o menor que.

B = Respuesta de la comparación igual.

X = Salida para la activación o frenado del motor de DC

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

Tabla 2. Tabla de estados.

Fuente: Meléndez, Alexis

Al comparar la respuesta de nuestra tabla 1 con la tabla del CI L293D estamos cumpliendo con las condiciones de activación de giro del motor y de frenado del motor de DC.

A continuación presento el diseño final del vi de la práctica del módulo M2CI de la parte de monitoreo y control posición de un motor dc que se muestra en la figura 42.

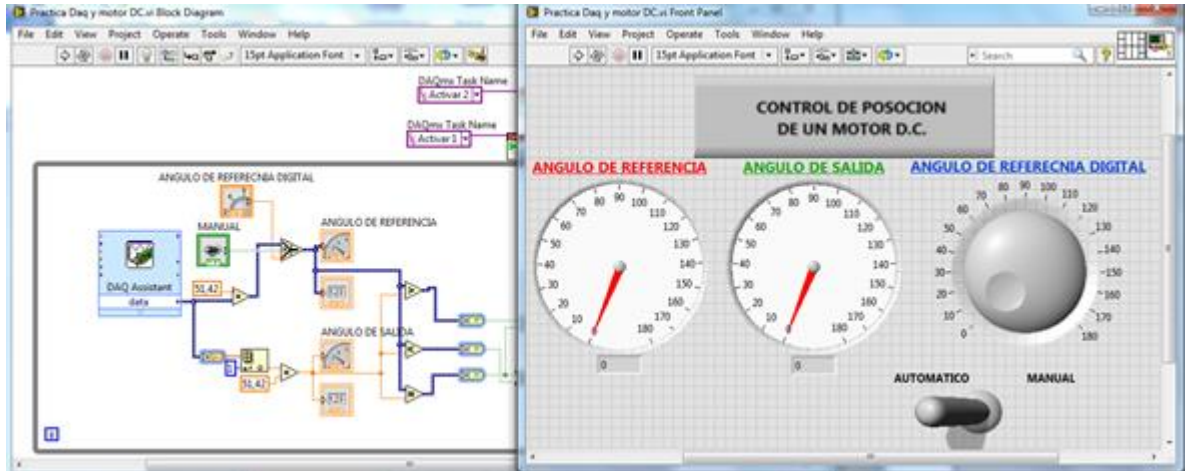


Figura 43. Diagrama a bloques del diseño de VI del monitoreo y control posición de un motor dc

Fuente: Meléndez, Alexis

Se realizó la práctica introduciendo varios valores de ángulo de referencia que son de 0 grados que se aprecia en la figura 43, 90 grados que se aprecia en la figura 44, 180 grados que se aprecia en la figura 45.



Figura 44. Ejecución del VI a 0 grados

Fuente: Meléndez, Alexis



Figura 45. Ejecución del VI a 90 grados

Fuente: Meléndez, Alexis



Figura 46. Ejecución del VI a 180 grados

Fuente: Meléndez, Alexis

6.1 Análisis del desarrollo del proyecto

Para el correcto funcionamiento del módulo se hizo como primer paso una revisión de los elementos involucrados de la parte de modulo en general, luego se hizo un seguimiento del cableado que posee el módulo M2CI para que opere correctamente con el DAQ; se realizó una investigación que software o complementos necesarios para que el pc lo reconozca y se procedió a la

instalación del NIMax y que sea compatible con la versión del Labview 2012, nos diera la opción de DAQ assistant dentro del el, y poder realizar el VI.

Se realizó la conexión, configuración y prueba del DAQ 6008 que trae el software NIMax, luego se procedió a la verificación de la opción habilitada dentro del Labview, la cual ya se cuenta dentro de él. Luego se realizó a prueba y error el correcto funcionamiento de las entradas y salidas del DAQ USB dentro del labview con ayuda de asistente que trae el software y el internet.

Se realizó el cableado que trae el módulo M2CI con ayuda del libro del módulo la parte correspondiente al a tarjeta de monitoreo y control posición de un motor dc al DAQ USB 6008, se programó un VI para que recoja y procese la información adquirida por las variables análogas y también las salidas digitales.

Teniendo la configuración correcta del DAQ desde el labview, a prueba y error se realizó el programa final que controla el sentido de giro del motor que simula el cierre y la apertura de una válvula donde se pude hacer el control de forma manual o controlada de forma digital.

6.2 Cronograma

ACTIVIDAD	SEM 1	SEM 2	SEM 3	SEM 4	SEM 5	SEM 6	SEM 7	SEM 8	SEM 9	SEM 10	SEM 11	SEM 12
Análisis del proyecto	X											
Reconocimiento del módulo		X										
Investigación de los dispositivos			X									
Cableado de la planta piloto				X	X							
Instalación de LabView					X							
Pruebas de funcionamiento						X	X					
Construcción de modelo de práctica								X				
Creación práctica en Labview									X			
Elaboración de informe							X	X	X	X	X	
Sustentación												X

CONCLUSIONES

- Se realizó la correcta Identificación de la estructura y funcionamiento del módulo M2CI de las componentes relacionados con el monitoreo y control de un motor de DC.
- Se programó el módulo DAQ por medio del software de Labview donde se Incorporó un diseño gráfico amigable para el usuario y fácil de entender del funcionamiento del módulo.
- Se recurrió a los datasheet de los dispositivos para saber que pines se conectaban, de qué forma y que voltajes son los permitidos para su correcto funcionamiento.
- Se realizó la correcta instalación de los drivers y paquetes de software para el correcto funcionamiento del DAQ USB 6008, con el labview instalado en el PC donde se controla el modulo M2CI.

RECOMENDACIONES

- Se recomienda para una mayor comprensión y entendimiento del módulo M2CI cuente con mayor información de los dispositivos que trae y los diseños esquemáticos eléctricos sean más detallados.
- Se recomienda que cuente con los paquetes de software necesarios para operar todo lo referente a los dispositivos involucrados en el módulo M2CI.
- Se recomienda que el modulo cuente con conexiones eléctricas ya predeterminadas para evitar en algún corto y un error de manipulación del usuario.

BIBLIOGRAFÍA

- [1] Rodríguez, P.; y Peñarete, R. (2011) MEDICIÓN Y CONTROL DE POSICIÓN ANGULAR EMPLEANDO SYNCHRO – RESOLVER.
- [2] Boyes, Geoffrey S. (1980) Synchro and Resolver Conversion, Memory Devices.
- [3] HODSON, D. (1994) Hacia un enfoque más crítico del trabajo de laboratorio. ENSEÑANZA DE LAS CIENCIAS, 1994, 12.(3), 299-313
- [4] <http://sine.ni.com/nips/cds/view/p/lang/es/nid/201986>
- [5] http://platea.pntic.mec.es/vgonzale/cyr_0204/ctrl_rob/robotica/sistema/motores_servo.htm
- [6] <http://diymakers.es/control-velocidad-y-sentido-de-motor-dc/>
- [7] [https://es.wikipedia.org/wiki/Puente_H_\(electr%C3%B3nica\)](https://es.wikipedia.org/wiki/Puente_H_(electr%C3%B3nica))
- [8] <http://www.ingmecafenix.com/electronica/potenciometro/>
- [9] <http://www.ni.com/getting-started/labview-basics/esa/environment>
- [10] <https://instrumentacionmecanica.files.wordpress.com/2011/12/prc3a1ctica-11-adquisicic3b3n-y-anc3a1lisis-de-datos-en-labview.pdf>
- [11] <https://es.slideshare.net/FelixRamirezRamrezMo/trabajando-con-la-ni-usb>

ANEXO A

Diagrama eléctrico del control de giro del motor DC

