PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
SEDE AMBATO

ESCUELA DE INGENIERIA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TITULO DE INGENIERÍA DE SISTEMAS

"ELABORACION DE MATERIAL DIDÁCTICO PARA LA IMPLEMENTACIÓN DE UN LABORATORIO DE AUTOMATIZACIÓN Y CONTROL DE LA ESCUELA DE INGENIERÍA DE SISTEMAS DE LA PUCESA"

PABLO XAVIER PARRA MOREIRA

DIRECTOR DE LA DISERTACIÓN:

ING. VICTOR CHUNCHA

AMBATO, 2006

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR
SEDE AMBATO

ESCUELA DE INGENIERIA DE SISTEMAS

DISERTACIÓN DE GRADO PREVIA LA OBTENCIÓN DEL TÍTULO DE INGENIERÍA DE SISTEMAS

"ELABORACION DE MATERIAL DIDÁCTICO PARA LA IMPLEMENTACIÓN DE UN LABORATORIO DE AUTOMATIZACIÓN Y CONTROL DE LA ESCUELA DE INGENIERÍA DE SISTEMAS DE LA PUCESA"

DIRECTOR : _____

ING. VICTOR CHUNCHA

PABLO XAVIER PARRA MOREIRA

DEDICATORIA

Dedico este trabajo a las dos personas más importantes de mi vida que siempre han estado en los momentos difíciles, que sin su apoyo desinteresado no lo hubiera logrado Mario y Nancy

Pablo

AGRADECIMIENTO

Un agradecimiento especial para mi Padre Celestial por todos todas las bendiciones recibidas durante toda mi vida.

Mi gratitud a toda mi familia: Mario, Nancy, Eduardo, a mi gordita Gaby, Fernando, Paúl, Ricardo, ya que siempre estuvieron en los momentos difíciles brindándome su apoyo. Pero un gracias de todo corazón a mis padres, por preocuparse y confiar en mi siempre, por estar a mi lado y apoyarme en todo.

A mi director Ing. Víctor Chuncha por haberme brindado su amistad ya que sin su ayuda no hubiera podido acabar este trabajo.

A todos mis profesores que ayudaron a mi formación profesional y me formaron como un hombre de bien

Un reconociendo muy especial para mi Tía CECI así como para Doña Lupita que me brindaron su apoyo y amistad en mi época colegial

Finalmente agradezco a todas las otras personas que me ayudaron a cumplir este objetivo.

Pablo

ÍNDICE DE CONTENIDOS

Dedicatoria…….....I

Agradecimiento……II

Capitulo I

1. Proyecto de Investigación…..……1

1.1. Introducción…..…….…..1

2. Planteamiento del Problema…….……..1

2.1. Problema…….…….…2

2.2. Problematización…….……....2

3. Delimitación……..….2

4. Importancia y justificación……..…...3

4.1. Importancia……..……3

4.2. Justificación…….…4

5. Objetivos……4

5.1. Objetivo General…….…4

5.1. Objetivo Específico…….5

6. Hipótesis…….5

7. Marco teórico…….....5

8. Aspectos Metodológicos……...….5

8.1. Paradigma……..…..5

8.2. Métodos de Investigación……6

9. Diagrama de procedimientos…….7

Capitulo II – Marco Teórico

2.1. Automatización y Control……..8

2.1.1 Introducción…….8

2.1.2 Conceptos y definiciones……..9

2.1.3 Automatización de procesos…….....12

2.1.4 Autómatas programables…….13

2.2. Automatización y Control en el ámbito académico……..16

2.2.1National Instruments……19

2.2.2 Labview……...21

2.2.3 Paquete de instrumento……...34

2.3. Herramientas para la Generación de proyectos Multimedia…….36
2.3.1. Camtasia Recorder……36
2.3.2 Protell99……41
Capitulo III – Desarrollo del Proyecto
3.1. Planificación……..47
3.1.1. Análisis de las guías y talleres……...47
3.2. Diseño de las guías y talleres……...47
3.2.1. Desarrollo de las guías propuestos……48
3.3 Lista de guías y talleres multimedia……48
Capitulo IV – Validación y Verificación de Resultados
4.1. Verificación de la Hipótesis……..50
4.2. Conclusiones…….50
4.3. Recomendaciones……..51
4.4. Validación…….52
A**nexos**
Anexo1-Práctica No 1……52
Anexo2-Práctica No 2……75
Anexo3-Práctica No 3……82
Anexo4-Práctica No 4…….89
Anexo5-Práctica No 5…….96
Anexo6-Práctica No 6……102
Anexo7- Data Sheet materiales……..110

CAPITULO I

1. PROYECTO DE INVESTIGACIÓN

1.1 INTRODUCCIÓN

En la actualidad la Ingeniería de Sistemas no es una disciplina aislada encargada únicamente del desarrollo de software, sino más bien su campo de acción es cada vez más amplio debido a las necesidades de automatización en diferentes áreas.

Nuestro país no se puede quedar relegado con los avances inminentes de la tecnología y cada día la industria requiere automatizar sus procesos productivos. Esto hecho hace que un Ingeniero de Sistemas esté preparado para afrontar estos cambios. Frente a esto la Escuela de Ingeniería de Sistemas de la PUCESA, se ve en la necesidad de facilitar nuevos medios para el aprendizaje de disciplinas de nueva generación, y uno de éstos es un laboratorio de automatización y control.

Este laboratorio será de gran ayuda para la Escuela de Ingeniería de Sistemas de la PUCESA, ya que se va a poder impartir de forma práctica lo concerniente a la automatización de procesos y control. Con éste aprendizaje se lograrán obtener profesionales competitivos y con conocimiento para crear soluciones informáticas en la automatización de industrias ambateñas y del centro del país.

2. PLANTEAMIENTO DEL PROBLEMA

2.1 PROBLEMA

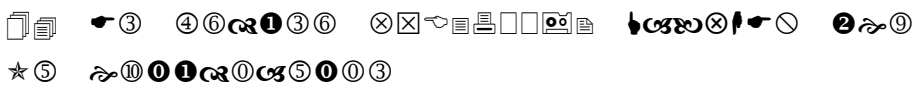
- La carencia de un Laboratorio de Automatización y Control en la Escuela de Ingeniería de Sistemas de la PUCESA en el período 2005-2006.

2.2 PROBLEMATIZACIÓN

- La Escuela de Sistemas de la PUCESA mediante convenios con la empresa Nacional Instrument posee software orientado a la automatización que no se puede aprovecharlo de forma integral.
- No existen los equipos necesarios para que los estudiantes puedan interactuar con el Software y generar nuevo conocimiento.
- La carencia de manuales y guías para que alumnos y profesores puedan realizar sus prácticas y talleres de Automatización y Control.

3. DELIMITACIÓN

Dentro del área de automatización y control, la Escuela de Ingeniería de Sistemas de la PUCESA tiene un convenio con la empresa NATIONAL INSTRUMENTS, la cual ha compartido licencias académicas del software LABVIEW y capacitación en el área a dos de sus docentes. Sin embargo, la necesidad de la Escuela es la implementación de un laboratorio con equipos que puedan ayudar a integrar el software con el hardware para lograr crear talleres de aprendizaje en el campo de la automatización. Para la implementación de éste laboratorio se requieren dos elementos fundamentales:



2. Guías de apoyo para la realización de prácticas y talleres

La implementación de guías y talleres que permitan a profesores y alumnos a manipular el software y los equipos de una forma adecuada y se pueda lograr el aprendizaje en el campo de la Automatización y Control. El material didáctico contendrá las especificaciones, pasos e instrumentos que se utilizarán dentro de 6 prácticas que se podrían llevar a cabo en un semestre. Cada práctica constará con su respectivo material didáctico multimedia.

El Laboratorio con los dos elementos antes detallados se implementará dentro del período septiembre 2005 a enero 2006 para que pueda ser utilizado desde el siguiente semestre.

4- IMPORTANCIA Y JUSTIFICACIÓN

4.1 Importancia.

A nivel Local los estudiantes de la carrera de Sistemas podrán complementar los conocimientos adquiridos en las aulas a través de la práctica, orientando su conocimiento a nuevas disciplinas que un Ingeniero de Sistemas debe conocer para lograr una alta competitividad.

A nivel Provincial será importante que la Escuela cuente con un Laboratorio de Automatización y Control para mejorar los servicios que se brindan a todos los estudiantes de la provincia y de ésta manera puedan implementar soluciones informáticas de automatización en empresas o industrias de la provincia.

A nivel Nacional los profesionales que se formen en la Escuela tendrán un mejor perfil profesional lo que facilitaría su integración laboral en el ámbito nacional y de ésta manera se aporta al crecimiento y desarrollo del país.

4.2 Justificación.

Desde el punto de vista Tecnológico se justifica la implementación de un Laboratorio de Automatización y Control en la PUCESA para aprovechar el convenio existente con Nacional Instruments y para brindar nuevos conocimientos a los estudiantes a través de la praxis.

En cuanto a lo Económico, se justifica ya que Nacional Instruments ofrece licencias de Software completas de sitio para Departamentos, Colegios y Campus. Todas estas licencias tienen instalaciones ilimitadas dentro del Departamento, Facultad o Campus que lo haya adquirido. Los costos del suite de instrumentos son asequibles y se cuenta con los conocimientos y herramientas necesarias para la implementación del Laboratorio.

5. OBJETIVOS

5.1 OBJETIVO GENERAL

- Implementar un Laboratorio de Automatización y Control para la Escuela de Ingeniería de Sistemas de la PUCESA con su respetivo Software, Hardware y guías académicas multimedia.

5.2 OBJETIVOS ESPECIFICOS

- Establecer contacto con la empresa Nacional Instruments para solicitar la versión del paquete de programas académicos compatibles con el suite de instrumentos que se adquieran.
- Diagnosticar y adquirir los equipos, instrumentos y materiales necesarios para formar el Laboratorio de Automatización y Control.
- Desarrollar guías y talleres multimedia para uso de profesores y alumnos de la Escuela de Ingeniería de Sistemas.

6. HIPÓTESIS

Mediante la implementación del Laboratorio de Automatización y Control y el desarrollo de guías y talleres multimedia, los estudiantes de la Escuela de Ingeniería de Sistemas de la PUCESA podrán realizar sus prácticas en la Universidad lo que facilitará su aprendizaje en el campo de la Automatización y Control.

7. MARCO TEÓRICO

Para el desarrollo de este proyecto se trataran los siguientes temas.

- Automatización y Control de procesos
- Autómatas programables
- National Instruments
- LABVIEW

- Herramientas para generar proyectos multimedia

8. ASPECTOS METODOLÓGICOS

8.1.PARADIGMA

En este proyecto de disertación se utilizarán dos paradigmas descritos a continuación.

Racionalista y Pragmático. Racionalista porque el proyecto tiene preconcebido la implementación de programas, instrumentos y guías de apoyo.

Pragmático porque el resultado final es el funcionamiento del laboratorio sin importar el proceso que se utilice.

8.2 MÉTODOS DE INVESTIGACIÓN

Durante el proceso de investigación del presente proyecto de disertación, se utilizará el Método Científico para plantear las bases de la investigación. Así mismo, será una Investigación Experimental en la medida en que se ejecutará experimentación del software y los instrumentos de Automatización y Control.

8.2.1 Tipo de investigación

En este proyecto se llevará a cabo una Investigación de Tipo Bibliográfica, puesto que se realizará primeramente un estudio de las especificaciones y compatibilidad del software y hardware que se adquiera, y una Investigación Experimental porque se probarán y diagnosticaran los instrumentos de automatización y Control adquiridos, antes del desarrollo de las guías y talleres.

8.2.2 Técnicas de investigación

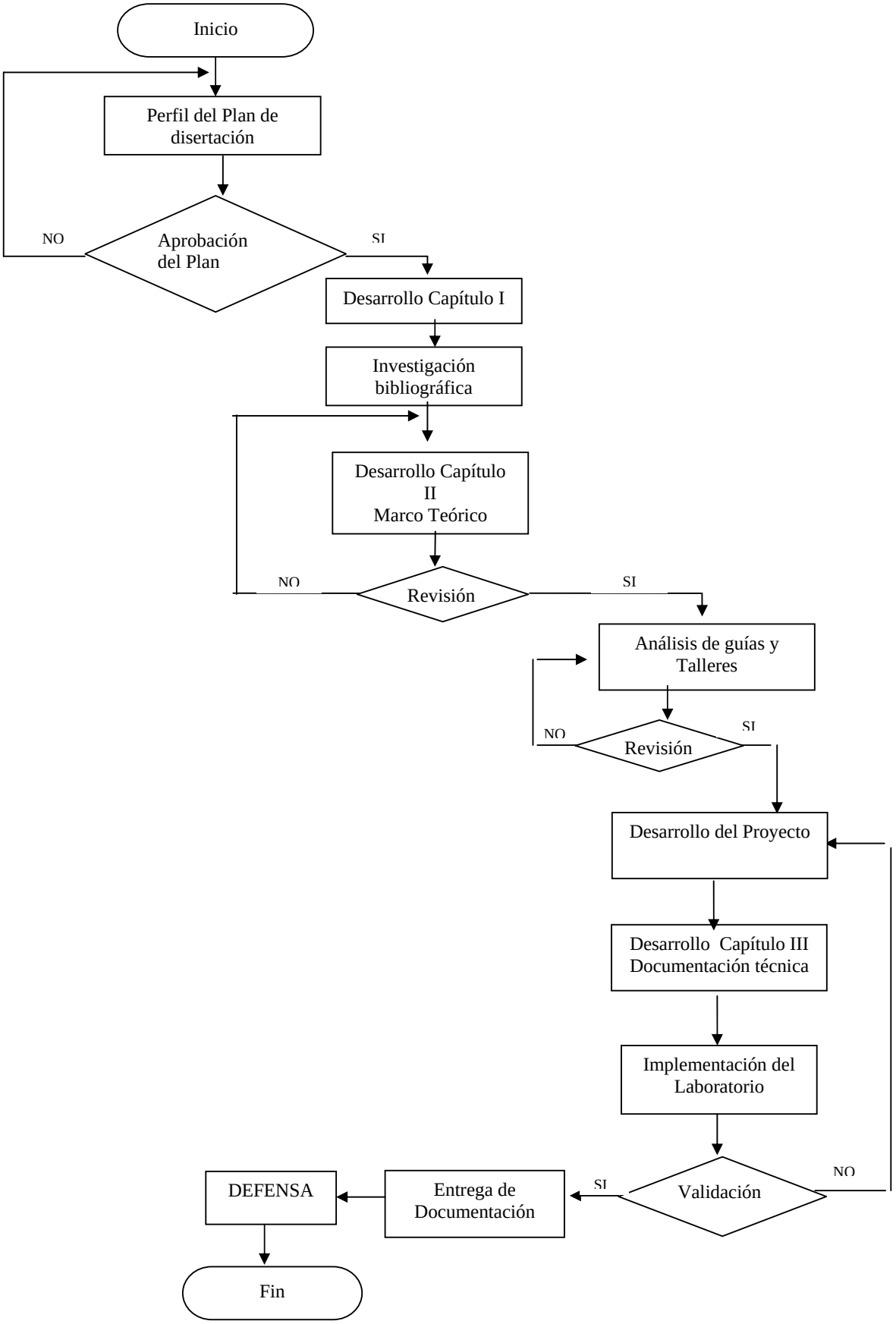
En la presente investigación se requiere extraer información de diferentes fuentes, para esto se realizará búsqueda y recuperación de información en el Internet, así como el uso

de manuales y tutoriales. Lo anterior conlleva al uso de técnicas de fichaje que permita clasificar y organizar las fuentes de consulta.

8.2.3 Nivel de investigación

Descriptivo y explicativo, ya que se realizará una descripción de la tecnología necesaria para la realización de las guías y talleres que utilizarán profesores y estudiantes.

1.1 DIAGRAMA DE PROCEDIMIENTOS



Capitulo II

2. Marco Teórico

2.1. Automatización y Control

2.1.1 Introducción

En los últimos 20 años, en respuesta a las demandas del cliente, los sistemas de control han sobrellevado cambios fundamentales, una verdadera revolución en control. Esta revolución ha sido liderada por una nueva arquitectura en la cual la computadora es el corazón del sistema. La revolución en control ha transformado las aplicaciones de prueba, control y automatización de instrumentos y dispositivos de caja poco integrados y muchas veces incompatibles a sistemas de control y automatización de alto desempeño y altamente integrados. En el centro de esta revolución se encuentra un componente que cada vez se hace más importante, el software.

Los avances en el hardware de las computadoras personales han impulsado mejoras en el desempeño significativas y reducción de costos en sistemas de control comparados con instrumentos tradicionales. También el software, es altamente integrado y productivo que permite que los ingenieros y científicos tomen ventaja de estos beneficios. Un conjunto de herramientas, desde arquitecturas de drivers y conectividad a instrumentos hasta ambientes de desarrollo abiertos y estándares, le brinda a los ingenieros la libertad de crear nuevos sistemas de control personalizables y muy poderosos.

Se puede definir control como “la manipulación indirecta de las magnitudes de un sistema llamado planta a través de otro sistema llamado sistema de control”



2.1.2 Conceptos y definiciones

Automatismo:

El control automático de procesos es parte del progreso industrial desarrollado durante lo que ahora se conoce como la segunda revolución industrial. El uso intensivo de la ciencia de control automático es producto de una evolución que es consecuencia del uso difundido de las técnicas de medición y control .Su estudio intensivo ha contribuido al reconocimiento universal de sus ventajas.

El control automático de procesos se usa fundamentalmente porque reduce el costo de los procesos industriales, lo que compensa con creces la inversión en equipo de control. Además hay muchas ganancias intangibles, como por ejemplo la eliminación de mano de obra pasiva, la cual provoca una demanda equivalente de trabajo especializado. La eliminación de errores es otra contribución positiva del uso del control automático.

Automatización:

Aplicación de sistemas automáticos en la realización de un proceso.
Elementos de un sistema automático Ver figura 1

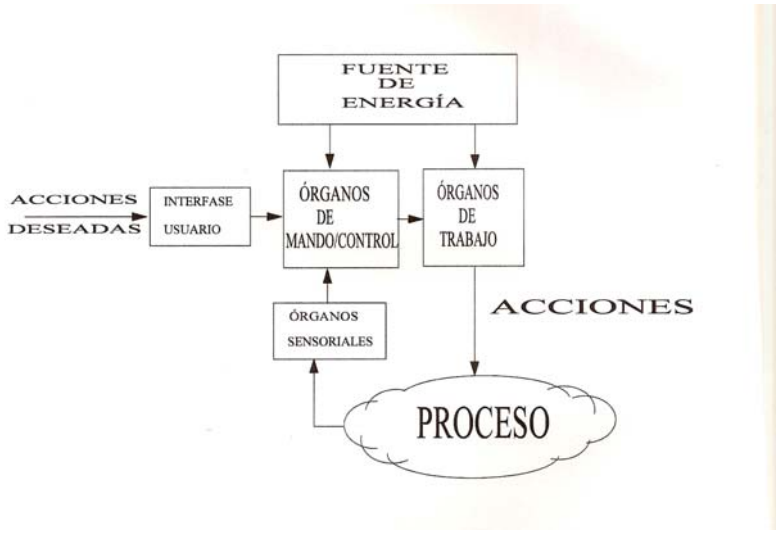


Figura 1 Elementos sistemas Automáticos

Acciones.

Actuación sobre el medio o proceso, con frecuencia son operaciones que se pueden repetir indefinidamente. Suelen ser acciones humanas susceptibles de ser sustituidas por acciones mecánica realizadas por los órganos de trabajo.

Fuentes de Energía.

Las operaciones y movimientos de los sistemas automáticos suponen un gasto energético que ha de ser aportado por un medio externo.
Suele denominarse fuente de potencia a aquélla que suministra energía a los órganos de trabajo que actúan sobre el proceso.
Las funciones propias del sistema automático también necesitan de un soporte energético.

Órganos de Mando/Control.

Representa el sistema que decide cuando realizar las acciones, que acciones realizar, y en su caso, el valor que han de tener algunos de los parámetros que definen una acción o tarea.

Órganos Sensoriales

Son sistemas cuya misión consiste en captar o medir determinados valores o magnitudes durante la realización del proceso. Estos órganos proporcionan información a los órganos de mando para que estos puedan dividir consecuentemente:

Procesos Continuos

Las magnitudes que determinan la evolución del proceso cambian de forma continua en el tiempo.

Procesos Discretos o Discontinuos

Las magnitudes que determinan la evolución del proceso cambian de forma discreta o discontinua y suelen tomar solo determinados valores.

El sistema evoluciona mediante eventos. Estos procesos son también conocidos como procesos de eventos discretos. En los procesos discretos se actúa sobre objetos concretos también llamados elementos discretos.

Procesos por Lotes.

Son procesos discretos en los que intervienen más de un elemento o pieza inicial para ser transformados en un solo producto.

Regulación Automática.

Mecanismos que permiten actuar durante un proceso continúo con el fin de que las magnitudes alcancen un valor determinado.

Cuando este valor se mantiene constante en el tiempo se dice que se está ante un problema de regulación.

Cuando este valor varía en el tiempo se dice que se está ante un problema de servomecanismo. Mando de un sistema de Automatización, Control Secuencial

El concepto de Automatización industrial suele aplicarse al control de procesos discretos. Los órganos de mando reciben información discreta del proceso y proporcionan órdenes discretas sobre los órganos de trabajo.

Los sistemas de mando adquieren una estructura secuencial:

- El proceso se divide en una serie de estados o estadios.
- Cada estado se activa y desactiva de forma secuencial.
- Cada estado activo tiene asociada una serie de acciones.

2.1.3 Automatización de procesos

El modelaje y simulación se ha constituido en una poderosa herramienta para el diseño, análisis y optimización de sistemas y procesos industriales. La disponibilidad de computadores personales cada vez más poderosos, de menor costo y de fácil uso, acompañados por software ó programas de aplicación y lenguajes de programación altamente flexibles, ha permitido la masificación del uso de diferentes técnicas de simulación y control de procesos. Este trabajo se concentra en la aplicación de las técnicas de modelaje, simulación y control de procesos haciendo uso de la tecnología PC y un lenguaje de programación 100% gráfico que permite la construcción de modelos mediante un paradigma de programación con "íconos interconectados por cables gráficos", cuyo código es ejecutado en base al principio de "flujo de datos" por los "cables" que interconectan los diferentes "íconos" o funciones. Combinando este lenguaje de programación 100% gráfico con tarjetas de adquisición y control (entradas/salidas) para el PC, se puede entonces configurar un sistema de simulación y control capaz de manejar señales reales (analógicas y digitales, de entrada y de salida) y ejecutar modelos matemáticos de control de procesos en tiempo-real.

La simulación de sistemas se ha convertido entonces en una poderosa herramienta para la toma de decisiones que nos permite, entre otras cosas:

- Predecir el resultado de las acciones que se tomen sobre el proceso o sistema de control.
- Comprender porqué los eventos ocurren.
- Identificar áreas problemáticas antes de la implantación del sistema.
- Explorar los efectos de las modificaciones.
- Evaluar ideas y su viabilidad, e identificar sus ineficiencias.
- Estimular el pensamiento creativo y entrenar al personal.
- Optimizar los procesos (ahorros de energía, cuellos de botella, mejoras de los rendimientos, etc.).

2.1.4 Autómatas programable

Origen y perspectivas

Los autómatas programables aparecieron en los Estados Unidos de América en los años 1969 – 70, y más particularmente en el sector de la industria del automóvil; fueron empleados en Europa alrededor de dos años más tarde. Su fecha de creación coincide, pues con le comienzo de la era del microprocesador y con la generación de la lógica cableada modular.

El autómata es la primera máquina con lenguaje, es decir, un calculador lógico cuyo juego de instrucciones se orienta hacia los sistemas de evolución secuencial. Hay que apreciar que, cada vez más, la universalidad de los ordenadores tiende a desaparecer, el futuro parece abrirse hacia esta nueva clase de dispositivos: máquina para proceso de señales, para la gestión de bases de datos, etc.

El autómata programable es, pues en este sentido un percusor y constituye para los automatistas un esbozo de la maquina ideal.

La creciente difusión de aplicaciones de la electrónica, la fantástica disminución del precio de los componentes, el nacimiento y el desarrollo de los microprocesadores y, sobretodo, la miniaturización de los circuitos de memoria permiten presagiar una introducción de los autómatas programables, cuyo precio es atractivo incluso para equipos de prestaciones modestas, en una inmensa gama de nuevos campos de aplicación.

El autómata programable satisface las exigencias tanto de procesos continuos como discontinuos. Regula presiones, temperaturas, niveles y caudales así como todas las funciones asociadas de temporización, cadencia, conteo y lógica. También incluye una tarjeta de comunicación adicional, el autómata se transforma en un poderoso satélite dentro de una red de control distribuida.

El autómata programable es un aparato electrónico programable por un usuario programador y destinado a gobernar, dentro de un entorno industrial, máquinas o procesos lógicos secuenciales.
Constitución.- Un autómata programable propiamente dicho está constituido por:

Fuente de alimentación:

Es la encargada de convertir la tensión de la red, 220v corriente alterna, a baja tensión de corriente continua, normalmente a 24v. Siendo esta la tensión de trabajo en los circuitos electrónicos que forma el autómata.

Unidad Central de Procesos o CPU:

Se encarga de recibir las órdenes del operario por medio de la consola de programación y el módulo de entradas. Posteriormente las procesa para enviar respuestas al módulo de salidas. En su memoria se encuentra residente el programa destinado a controlar el proceso.
Contiene las siguientes partes:
Unidad central o de proceso

- Temporizadores y contadores
- Memoria de programa
- Memoria de datos
- Memoria imagen de entrada
- Memoria de salida

Módulo de entrada:

Es al que se unen los captadores (interruptores, finales de carrera, pulsadores). Cada cierto tiempo el estado de las entradas se transfiere a la memoria imagen de entrada. La información recibida en ella, es enviada al CPU para ser procesada de acuerdo a la programación.

Se pueden diferenciar dos tipos de captadores conectables al módulo de entradas: los *pasivos* y los *activos*.

Los *captadores pasivos* son los que cambian su estado lógico (activado o no activado) por medio de una acción mecánica. Estos son los interruptores, pulsadores, finales de carrera

Los *captadores activos* son dispositivos electrónicos que suministran una tensión al autómata, que es función de una determinada variable.

Módulo de salidas:

Es el encargado de activar y desactivar los actuadores (bobinas de contactores, lámparas, motores pequeños).

La información enviada por las entradas al CPU, una vez procesada, se envía a ala memoria imagen de salidas, de donde se envía a la interfase de salidas para que estas sean activadas y a la vez los actuadores que en ellas están conectados.

Según el tipo de proceso a controlar por el autómata, podemos utilizar diferentes módulos de salidas. Existen tres tipos bien diferenciados:

A relés: son usados en circuitos de corriente continua y corriente alterna. Están basados en la conmutación mecánica, por la bobina del relé, de un contacto eléctrico normalmente abierto.

A triac: se utilizan en circuitos de corriente continua y corriente alterna que necesitan maniobras de conmutación muy rápidas.

A transistores a colector abierto: son utilizados en circuitos que necesiten maniobras de conexión / desconexión muy rápidas. El uso de este tipo de módulos es exclusivo de los circuitos de corriente continua.

Terminal de programación:

El terminal o consola de programación es el que permite comunicar al operario con el sistema. Las funciones básicas de éste son las siguientes:

- Transferencia y modificación de programas.
- Verificación de la programación.
- Información del funcionamiento de los procesos.

Como consolas de programación pueden ser utilizadas las construidas específicamente para el autómata, tipo calculadora o bien un ordenador personal, PC, que soporte un software específicamente diseñado para resolver los problemas de programación y control.

Periféricos:

Los periféricos no intervienen directamente en el funcionamiento del autómata, pero sin embargo facilitan la labor del operario.

Los más utilizados son:

- Grabadoras a cassettes.
- Impresoras.
- Cartuchos de memoria EPROM.
- Visualizadores y paneles de operación OP.
- Memorias EEPROM.

2.2. Automatización y Control en el ámbito académico

Instrumentos Virtuales versus Instrumentos Tradicionales

Los instrumentos autónomos tradicionales, tales como osciloscopios y generadores de ondas, son muy poderosos, caros y diseñados para llevar a cabo una o más tareas específicas definidos por el fabricante. Sin embargo, el usuario por lo general no puede extender o personalizar esas tareas. Las perillas y botones del instrumento, sus circuitos electrónicos y las funciones disponibles para el usuario son todas específicas a la naturaleza del instrumento. Además, deben desarrollarse una tecnología especial y

componentes costosos para construirlos, lo cual los hace muy caros y lentos para adaptarlos.

Debido a que están basados en la PC (Computador Personal), los instrumentos virtuales aprovechan inherentemente los beneficios de la última tecnología de las computadoras personales corrientes. Estos avances en tecnología y rendimiento, que están cerrando rápidamente la brecha entre los instrumentos autónomos y las PCs, incluyen poderosos procesadores, tales como el Pentium 4, sistemas operativos como: Microsoft Windows XP, .NET y el Apple Mac OS X, y tecnologías. Además de incorporar características poderosas, esas plataformas también ofrecen un acceso sencillo a herramientas también poderosas tales como la Internet. Los instrumentos tradicionales también adolecen frecuentemente de falta de portabilidad, en tanto que los instrumentos virtuales que corren en las computadoras portátiles automáticamente incorporan esta naturaleza portátil.

Los ingenieros y científicos cuyas necesidades, aplicaciones y requerimientos varían muy rápidamente, necesitan flexibilidad para crear sus propias soluciones. Se puede adaptar un instrumento virtual a sus necesidades particulares sin necesidad de reemplazar todo el instrumento dado que posee el software de aplicación instalado en la computadora y al amplio rango disponible de hardware para instalar en ella.

Flexibilidad

A excepción de los componentes especializados y los circuitos hallados en los instrumentos tradicionales, la arquitectura general de los instrumentos autónomos es muy similar a la hallada en un instrumento virtual basado en computadora.

Ambos requieren uno o más microprocesadores, puertos de comunicación (por ejemplo: serie y GPIB) y capacidad de mostrar resultados así como también módulos de adquisición de datos. Lo que diferencia uno del otro es su flexibilidad y el hecho que usted puede modificar y adaptar el instrumento a sus necesidades particulares. Un instrumento tradicional podría contener un circuito integrado para llevar a cabo un conjunto particular de instrucciones de procesamiento de datos; en un instrumento virtual estas funciones podrían llevarse a cabo por el programa que corre en el

procesador de la computadora. Se puede fácilmente extender ese conjunto de funciones y estar sólo limitado por la potencia del software que utilice.

Reducción de Costos

Utilizando soluciones basadas en la instrumentación virtual, usted puede reducir los costos de inversión, desarrollo de sistemas y mantenimiento al mismo tiempo que mejora el tiempo de comercialización y la calidad de sus propios productos.

Hardware para Insertar y de Red

Existe una amplia variedad disponible de hardware que usted puede o bien insertar en su computadora o bien acceder a través de una red. Estos dispositivos ofrecen un amplio rango de capacidades de adquisición de datos a un costo significativamente inferior que el correspondiente a dispositivos dedicados. A medida que la tecnología de circuitos integrados avanza y los componentes comunes se vuelven más baratos y poderosos, también lo hacen las placas que ellos utilizan. Junto con estos avances tecnológicos viene un incremento en las velocidades de adquisición de datos, precisión de las mediciones y mejor aislamiento de las señales.

Dependiendo de la aplicación en particular, el hardware que usted elija podría incluir entradas o salidas analógicas, entradas o salidas digitales, contadores, temporizadores, filtros, muestreo simultáneo y capacidades de generación de ondas. La amplia gama de placas y hardware podría incluir cualquiera de esas características o una combinación de ellas.

Instrumento Tradicional	**Instrumento Virtual**
Definido por el fabricante	Definido por el usuario
Hardware es la clave.	Software es la clave
Alto costo/función	Bajo costo/función, variedad de funciones, reusable.
Arquitectura "cerrada"	Arquitectura "abierta".
Lenta incorporación de nuevas tecnología.	Rápida incorporación de nuevas

	tecnologías, gracias a la plataforma PC.
Bajas economías de escala, alto costo de mantenimiento.	Altas economías de escala, bajos costos de mantenimiento.
Funcionalidad específica, con conectividad limitada.	Funcionalidad ilimitada, orientado a aplicaciones, conectividad amplia.

2.2.1National Instruments (NI)

National Instruments es una empresa pionera y líder en la tecnología de la instrumentación virtual, un concepto revolucionario que ha cambiado la forma en que ingenieros y científicos abordan las aplicaciones de medición y automatización.

Aprovechando el poder de la PC y sus tecnologías relacionadas, la instrumentación virtual aumenta la productividad y reduce los costos por medio de software de fácil integración (como el ambiente de desarrollo gráfico LabVIEW de NI) y hardware modular (como los módulos PXI para adquisición de datos, control de instrumentos y visión artificial).

Los clientes de NI incluyen ingenieros, científicos y profesionales técnicos en una amplia gama de industrias. Desde pruebas a grabadoras de DVD hasta investigación de medicamentos avanzados, ellos utilizan el software y hardware de NI para ofrecer un amplio conjunto de productos de manera más rápida y a un menor costo.

INSTRUMENTACION VIRTUAL

Mucho hemos oído hablar sobre la "instrumentación virtual" y sus beneficios. El concepto de instrumentación virtual nace a partir del uso del computador personal como "instrumento" de medición de tales señales como temperatura, presión, caudal, etc.
Es decir, el PC comienza a ser utilizado para realizar mediciones de fenómenos físicos representados en señales de corriente (Ej. 4-20mA) y/o voltaje (Ej. (0-5Vdc). Sin embargo, el concepto de "instrumentación virtual" va más allá de la simple medición de corriente o voltaje, sino que también involucra el procesamiento, análisis, almacenamiento, distribución y despliegue de los datos e información relacionados con

la medición de una o varias señales específicas. Es decir, el instrumento virtual no se conforma con la adquisición de la señal, sino que también involucra la interfaz hombre-máquina, las funciones de análisis y procesamiento de señales, las rutinas de almacenamiento de datos y la comunicación con otros equipos.

Por ejemplo; el osciloscopio tradicional tiene una funcionalidad ya predefinida desde la fábrica donde lo diseñan, producen y ensamblan. Es decir, la funcionalidad de este tipo de instrumento es definida por el fabricante del equipo, y no por el usuario mismo. El término "virtual" nace precisamente a partir del hecho de que cuando se utiliza el PC como "instrumento" es el usuario mismo quién, a través del software, define su funcionalidad y "apariencia" y por ello decimos que "virtualizamos" el instrumento, ya que su funcionalidad puede ser definida una y otra vez por el usuario y no por el fabricante.

El instrumento virtual es definido entonces como una capa de software y hardware que se le agrega a un PC en tal forma que permite a los usuarios interactuar con la computadora como si estuviesen utilizando su propio instrumento electrónico "hecho a la medida".

¿Cómo construir un instrumento virtual?

Para construir un instrumento virtual, sólo se requiere un PC, una tarjeta de adquisición de datos con acondicionamiento de señales (PCMCIA, ISA, XT, PCI, etc.) y el software apropiado, los tres (3) elementos clave en la conformación de un instrumento virtual, teniendo un chasis de acondicionamiento de señales como elemento opcional. Ver Figura 2

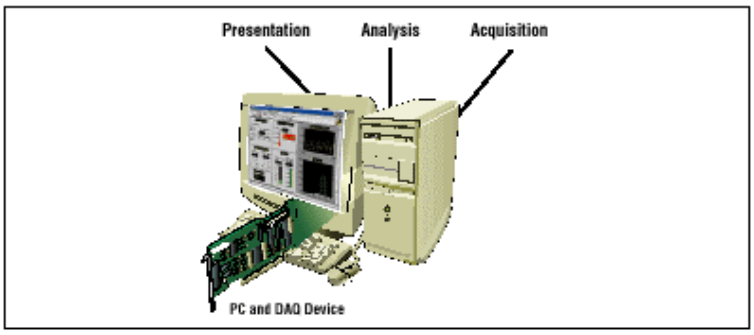


Figura 2 Partes de un instrumento virtual

El "acondicionamiento de señales" es opcional, porque dependiendo de cada señal y/o aplicación, se puede o no requerir amplificación, atenuación, filtraje, aislamiento, etc. de cada señal. Si la señal está en el rango de los +/- 5Vdc y no se requiere de aislamiento o filtraje, la misma puede ser conectada directamente la tarjeta de adquisición de datos.

Con el sistema indicado anteriormente, podríamos construir un osciloscopio "personalizado", con la interfaz gráfica que uno desee, agregándole inclusive más funcionalidad. Sin embargo, este mismo sistema puede también ser utilizado en la medición de temperatura, o en el control de arranque/parada de una bomba centrífuga.

2.2.2 Labview

Por casi 20 años, la popularidad de LabVIEW entre programadores tanto principiantes como experimentados en diferentes aplicaciones de la ingeniería e industria, se puede atribuir a su lenguaje intuitivo de programación gráfica usado para medidas automatizadas y sistemas de control. El lenguaje gráfico de flujo de datos y el enfoque del diagrama de bloques de LabVIEW representan naturalmente el flujo de sus datos y mapea de manera intuitiva los controles de la interfaz del usuario a sus datos; de esta manera usted puede fácilmente ver y modificar sus datos o entradas de control.

Para programadores principiantes, la tecnología de LabVIEW transforma medidas comunes y tareas de automatización en VIs intuitivos de mucho mayor nivel.

Para programadores experimentados, LabVIEW brinda el rendimiento, flexibilidad y compatibilidad de un lenguaje tradicional como C o BASIC. De hecho, el lenguaje completo de programación de LabVIEW posee las mismas características que los lenguajes tradicionales, variables, tipos de datos, ciclos y estructuras secuenciadas así como manejo de error. Y con LabVIEW se puede rehusar código heredado como DLLs o bibliotecas compartidas e integrar con otro software usando ActiveX, TCP y otras tecnologías estándares

Measurement & Automation de National Instruments con LabVIEW

LabVIEW es un lenguaje de Programación Gráfica (Lenguaje G), diseñado por National Instruments (NI) para aplicaciones en instrumentación, automatización, adquisición de datos y control de procesos, aunque sus creadores insisten en que LabVIEW es un lenguaje de programación de aplicación general como lo es Visual Basic, C, Pascal, etc.

Al instalar LabVIEW se crea en el Escritorio de Windows una carpeta llamada Measurement & Automation [M&A], la cual sirve para configurar las Salidas y entradas, tanto Analógicas como Digitales, de los diferentes dispositivos DAQ (data acquisition) reconocidos por National Instruments.

Para concluir con esta breve introducción debe decirse que todas las configuraciones y el manejo que permite hacer la carpeta **Measurement & Automation** también se pueden hacer por medio de programación G, ya que LabVIEW contiene las herramientas necesarias para tal fin.

Adquisición de Datos

Se entiende por *Adquisición de datos* a la acción de medir variables, convertirlas a formato digital, almacenarlas en un computador y procesarlas en cualquier sentido. Este proceso necesita de una "interfase" entre el mundo físico y el computador que se suele denominar como tarjeta de adquisición de datos.

El proceso de adquisición de datos del mundo físico conlleva los siguientes pasos fundamentales:

Utilización de un sensor/transductor adecuado para la variable que se desea medir, el cual permite detectar y convertir la variable física en una señal analógica de voltaje o corriente eléctrica.

1. Acondicionamiento de la señal eléctrica: En esta etapa se resuelven problemas relacionados con la señal obtenida, como son el ruido, la amplitud y/ó potencia de la señal, la no linealidad de la misma, entre otras.
2. Traducción de la señal eléctrica al lenguaje binario, propio del computador: Este proceso se conoce técnicamente como conversión Analógica/Digital (A/D).
3. Almacenamiento de los datos que, en forma digital, podrán ser almacenados en la memoria del computador y desplegados luego en la pantalla o en otro periférico del mismo. Ver Figura 3

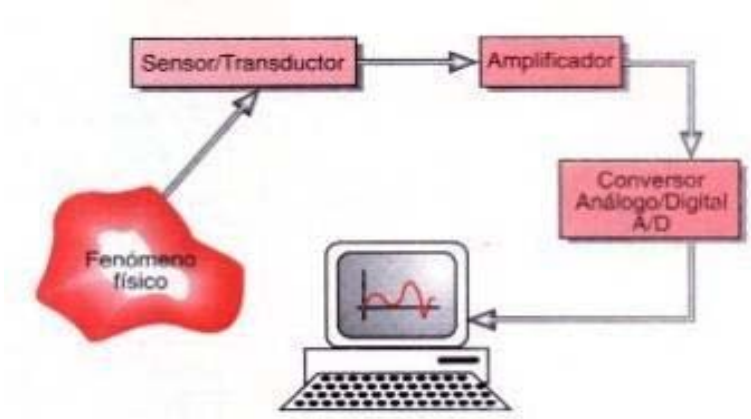


Figura 3 Proceso de adquisición de datos

Para las entradas analógicas existen tres modos diferentes, que se pueden configurar mediante software. Estos modos son:

- Entrada sin referencia con un único terminal (NRSE): Ambas referencias, la de la tarjeta y la del aparato de medición, son independientes, siendo la del medidor la utilizada para la señal de entrada.
- Entrada con referencia con un único terminal (RSE): La referencia de la señal de entrada es conectada físicamente a la referencia interna de la tarjeta.
- Entrada Diferencial (DIFF): Una línea de entrada se conecta a una señal de medida (S1) y otra línea de entrada se conecta a otra señal de medición (S2). Lo que se obtiene es la señal diferencial resultante (S1-S2 ó S2-S1, según se configure).

El Explorador de la Carpeta Measurement & Automation

Al dar doble clic sobre la carpeta [M&A], ubicada en el Escritorio, se abre el Explorador de Windows Figura4 mostrando los objetos del Escritorio en el lado izquierdo y el contenido de dicha carpeta hacia el lado derecho, siendo importante anotar que el Explorador que se abre no es exactamente igual al que se abriría normalmente. Las diferencias más importantes de este Explorador están en el menú **Archivo**, el cual se muestra (Fig8) y explica a continuación.

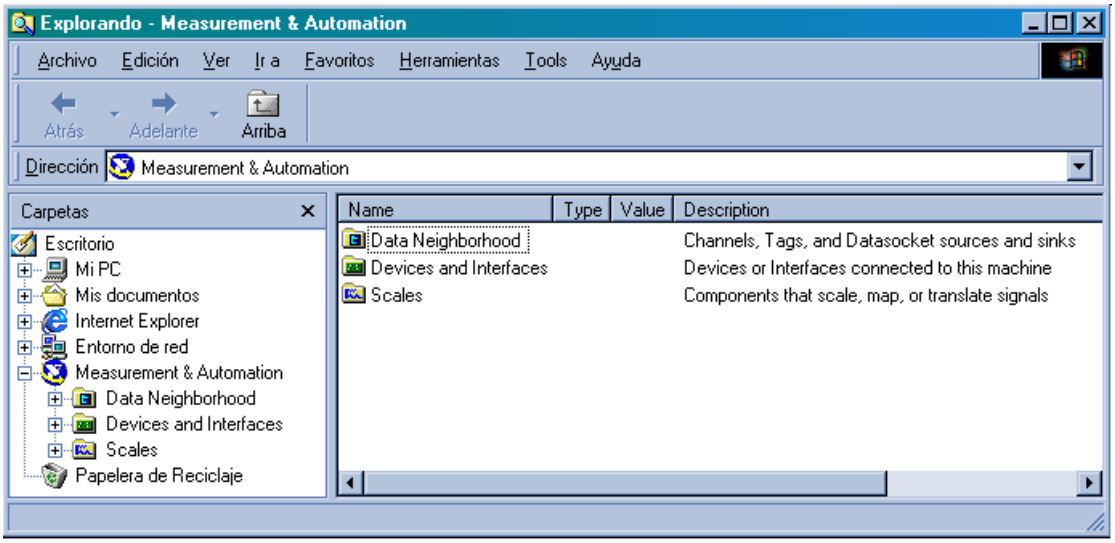


Figura 4.- Explorador de la Carpeta Measurement & Automation

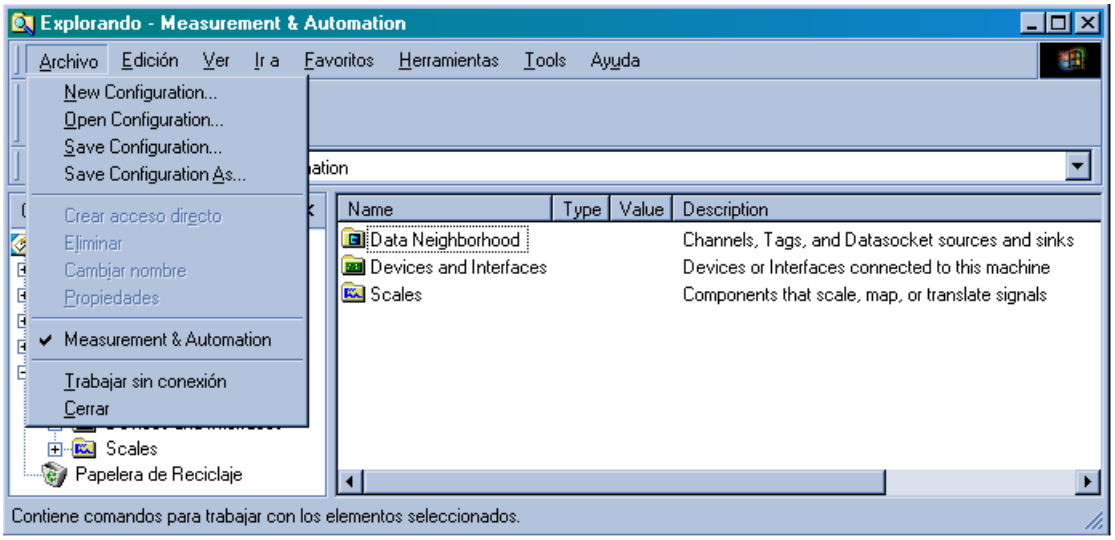


Figura5.- Menú Archivo del Explorador de la Carpeta Measurement & Automation

- **New Configuration**: Crea una nueva configuración de Entradas/Salidas para la tarjeta, la cual será almacenada en un archivo con extensión .daq, que se podrá guardar en cualquier carpeta.

- **Open Configuration**: Abre (carga) una configuración ya existente de la tarjeta. recuerde que las configuraciones se guardan en archivos con extensión .daq.

- **Save Configuration** y **Save Configuration As**: Para salvar configuraciones nuevas o ya existentes.

Contenido de la Carpeta Measurement & Automation

La carpeta en cuestión contiene tres subcarpetas, las cuales son:

- **Data Neighborhood** [DN]: En esta carpeta es donde se crean los atributos y/ó propiedades de los diferentes canales de la tarjeta que van a ser utilizados en una configuración dada.
- **Devices and Interfaces** [DI]: En esta carpeta es donde se pueden mostrar y editar los atributos de las diferentes Tarjetas de Adquisición de Datos que están instalados en el computador. Un atributo que debemos tener muy en cuenta es el Número de Dispositivo (Device Number), el cual nos servirá como identificador a la hora manipular los datos obtenidos de la tarjeta por medio de programación G (LabVIEW).
- **Scales** [S]: Aquí es donde se almacenan y modifican lo relacionado con las escalas que se usarán en las diferentes entradas y/ó salidas analógicas; esto es importante a la hora de visualizar los datos.

Introducción a LABVIEW

Los Programas en LabVIEW son llamados instrumentos virtuales (VIs). Cada VI contiene tres partes principales:

Panel frontal: El usuario interacciona con el VI, en esta ventana se observan los instrumentos virtuales.

Diagrama de bloque: El código que controla el programa.

Icono/Conector Medios para conectar un VI con otros VIs.

El panel frontal es utilizado para interaccionar con el usuario cuando el programa esta corriendo. Usuarios pueden controlar el programa, cambiar entradas, y ver datos actualizados en tiempo real. Recuerde que los controles son usados como entradas ajustando controles de deslizamiento para colocar un valor de alarma, encendiendo o apagando un switch, o parando un programa. Los indicadores son usados como salidas. Termómetros, luces, y otros indicadores indican valores del programa. Esto puede incluir datos, estados de programa y otra información.

Cada control o indicador del panel frontal tiene una terminal correspondiente en el diagrama de bloques. Cuando un VI se ejecuta, los valores de los controles fluyen a través del diagrama de bloques, en donde estos son usados en las funciones del diagrama, y los resultados son pasados a otras funciones o indicadores

El panel frontal es la interfase del usuario con el VI. Se construye el panel frontal con controles e indicadores, que son las entradas y salidas que interactúan con las terminales del VI, respectivamente. Los controles son botones, botones de empuje, marcadores y otros componentes de entradas. Los indicadores son las graficas, luces y otros dispositivos. Los controles simulan instrumentos de entradas de equipos y suministra datos al diagrama de bloques del VI. Los indicadores simulan salidas de instrumentos y suministra datos que el diagrama de bloques adquiere o genera. Ver figura 6

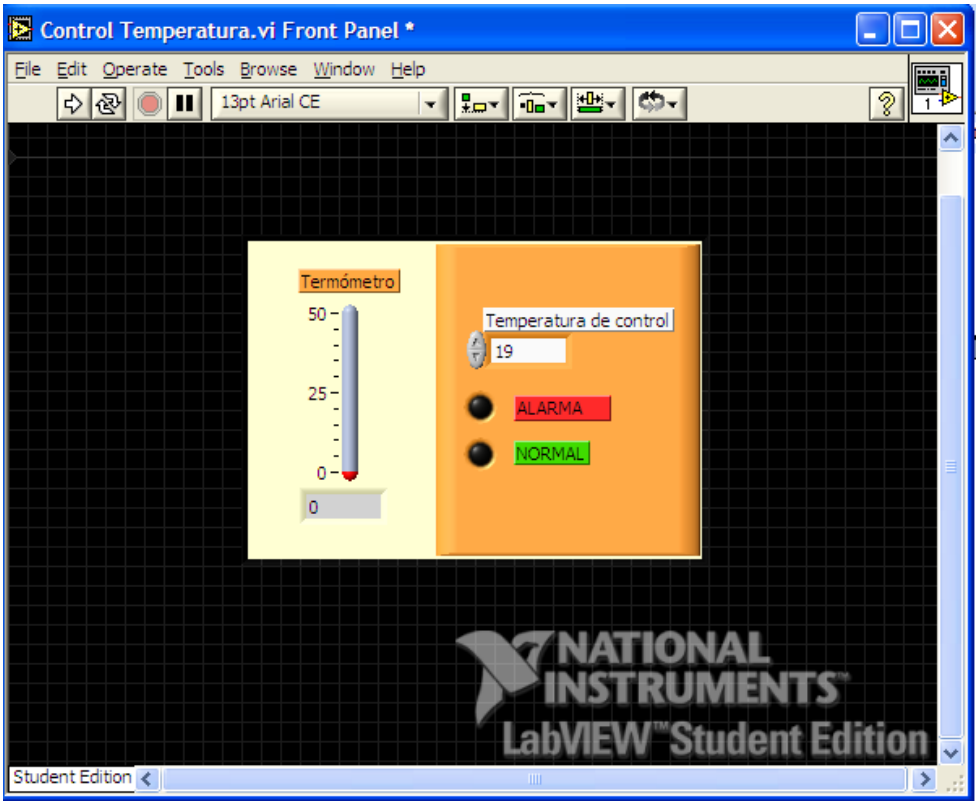


Figura 6 Panel frontal

El diagrama de bloque contiene el código fuente grafico. Los objetos del panel frontal aparecen como terminales en el diagrama de bloque. Adicionalmente, el diagrama de bloque contiene funciones y estructuras incorporadas en las bibliotecas de LabVIEW VI. Los cables conectan cada uno de los nodos en el diagrama de bloques, incluyendo controles e indicadores de terminal, funciones y estructuras. Ver Figura 7

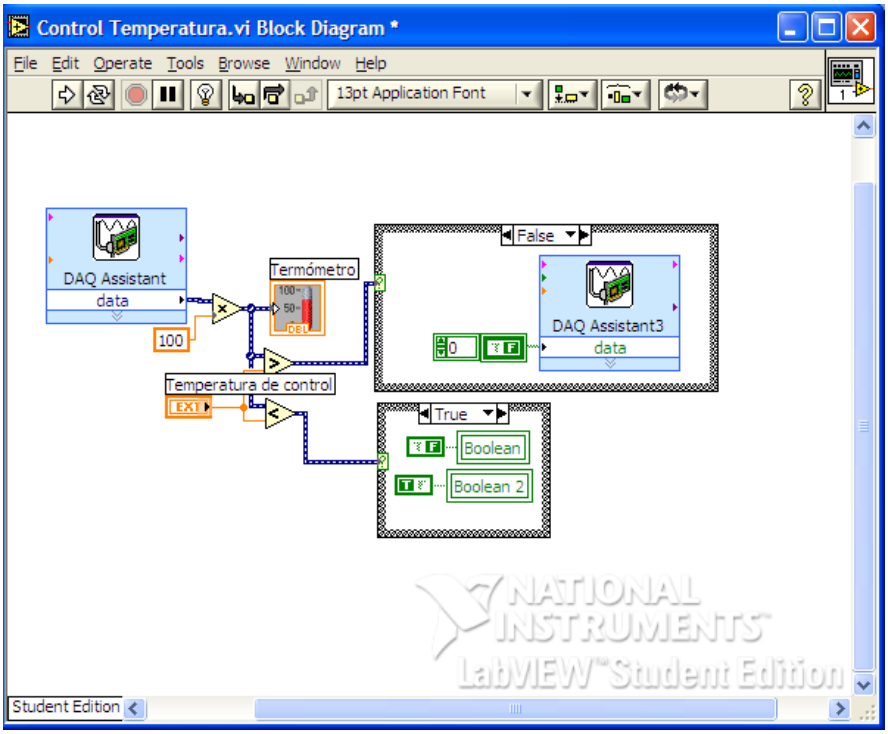


Figura 7 Diagrama de Bloque

La paleta de controles (**Controls**) se usa para colocar los controles e indicadores en el panel frontal. La paleta de controles esta disponible solamente en el panel frontal. Seleccione **Window»Show Controls palette** o haga clic derecho en el espacio de trabajo en el panel frontal para desplegar la paleta de controles. También puede desplegar la paleta de controles haciendo un clic derecho en una área abierta del panel frontal. Para desaparecer la paleta de controles presione el botón en la parte superior izquierda de la paleta. Ver Figura8



Figura 8 Ventana de control

La paleta de funciones (**Functions**), sirve para construir un diagrama de bloque. La paleta de funciones esta disponible solamente en el diagrama de bloque. Seleccione **Window»Show Functions Palette** o haga un clic derecho en el espacio de trabajo del diagrama de bloque para desplegar la paleta de funciones. Se puede desplegar la paleta de funciones dando un clic derecho en una área abierta del diagrama de bloques. Para desaparecer la paleta de funciones presione el botón en la parte superior izquierda de la

paleta. En esta barra vamos a encontrar todos los comandos indispensables para programar y realizar nuestros VIs Ver Figura 9

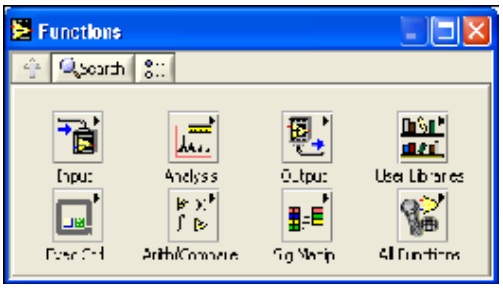


Figura 9 Ventana de funciones

Si la herramienta de selección automática esta habilitada y se mueve el cursor sobre un objeto en el panel frontal o en el diagrama de bloque, LabVIEW automáticamente selecciona la herramienta correspondiente de la paleta de controles. Cambie a la herramienta de selección automática tecleando el botón de Selección de Herramienta Automática en la paleta de Herramientas.

Utilice la herramienta de operación para cambiar valores o para seleccionar texto de un control. Utilice la herramienta de posicionamiento para seleccionar, mover o redimensionar objetos. La herramienta de posicionamiento cambia las formas cuando se mueve sobre una esquina de un objeto reajustable. Utilice la herramienta de etiquetado para editar textos y crear etiquetas libres. La herramienta de etiquetado se convierte en cursor cuando usted crea etiquetas libres. Utilice la herramienta de cableado para cablear objetos juntos en el diagrama de bloque. Ver Figura 10



Figura 10 Ventana de herramientas

Barra de herramientas de estado



 Botón de Ejecución

 Botón de ejecución continua

 Cancelación de ejecución

 Botón de pausa/continuación

 Configuración de texto

 Alineamiento de objetos

 Distribución de objetos

 Reordenamiento

 Redimensionamiento de objetos en el panel frontal

Pulse el botón de ejecución (**Run**) para ejecutar el VI. Mientras el VI se esta ejecutando, el botón de ejecución aparecerá con una flecha negra si es que el VI es un VI principal, lo que significa que no ha sido llamado por otro VI y por lo tanto este no es un subVI.

Pulse el botón de ejecución continua (**Continuous Run**) para ejecutar el VI hasta que el botón de cancelación de ejecución o de pausa sea presionado. Usted también puede pulsar este botón nuevamente para deshabilitar la ejecución continua. Mientras el VI se esta ejecutando, El botón de cancelación de ejecución (**Abort Execution**) aparece.
Presione este botón para detener el VI inmediatamente.
Nota: Evite usar el botón de cancelación de ejecución (**Abort Execution**) para detener el VI. Ya sea deje que el VI complete su flujo de datos o diseñe un método para detener el VI programáticamente. Al hacer esto el VI se encuentra en un estado conocido. Por ejemplo, coloque un botón en el panel frontal que detenga el VI cuando sea presionado. Pulse el botón de pausa (**Pause**) para detener momentáneamente la ejecución de un VI. Cuando usted presiona el botón de Pausa, LabVIEW señala la posición donde usted detuvo la ejecución en el diagrama de bloque. Pulse el botón de pausa nuevamente para que el VI continué ejecutándose.

Seleccione el menú **Configuración de Textos (Text Settings)** para cambiar el tipo de fuente (letra) del VI, incluyendo el tamaño, estilo y color.

Seleccione el menú **Alineamiento de Objetos (Align Objects)** para alinear objetos con respecto a los ejes, incluyendo eje vertical, superior, izquierdo, etc.

Seleccione el menú **Distribución de Objetos (Distribute Objects)** para espaciar objetos uniformemente, incluyendo espacios vacíos, compresiones, etc.

Seleccione el **Redimensionamiento de Objetos (Resize Objects)** para cambiar el tamaño de las ventanas

Seleccione el menú **Reordenamiento (Reorder)** cuando tenga objetos superpuestos unos con otros y usted quiera definir cual esta enfrente de quien y cual atrás. Seleccione uno de los objetos con la herramienta de posicionamiento y seleccione entre **Move Forward** (mover hacia adelante), **Move Backward** (mover hacia atrás), **Move to Front** (mover al frente) y **Move to Back** (mover hacia atrás).

Ciclos

Los ciclos Mientras (While) y Para (For) están localizados en la paleta **Functions»Structures.** El ciclo Para difiere del ciclo Mientras en que el ciclo Para, ejecuta una serie (set) de veces predeterminada. El ciclo Mientras deja de ejecutar el sub. diagrama solo si el valor en la condición terminal existe.

Ciclo Mientras

Similar al ciclo Haga (Do) o al ciclo Repita-hasta (Repeat-Until) en lenguajes de programación basados en texto, un ciclo Mientras, mostrado Figura 11, ejecuta un sub diagrama hasta que la condición sea cumplida. El ciclo Mientras ejecuta el sub diagrama hasta que la terminal dependiente, recibe un valor Booleano específico. El comportamiento incumplido y la apariencia de la Terminal dependiente (valor por defecto) es **Continue if True** (continue si es Verdadero), mostrado a la izquierda. Cuando una terminal dependiente es **Continue if True**, el ciclo Mientras ejecuta su sub. diagrama hasta que la terminal dependiente recibe un valor FALSO. La terminal de

iteración (una terminal de salida),. El conteo de iteraciones siempre empieza en cero.

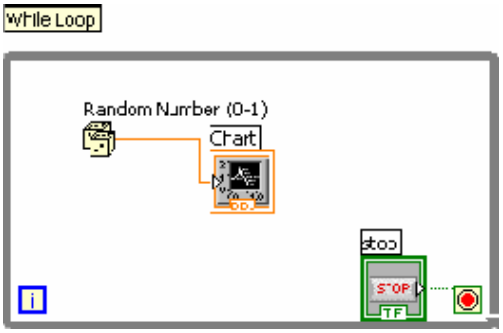


Figura 11 Ciclo While

Ciclo Para

Un ciclo Para, mostrado en la figura 12, ejecuta una serie varias veces. El valor en la terminal de conteo (una terminal de entrada) representada por la N, indica cuantas veces repetir el sub. diagrama. La terminal de iteración (una entrada de salida), mostrada a la izquierda, contiene el numero de iteraciones completas. El conteo de iteraciones siempre empieza en cero Ver Figura12

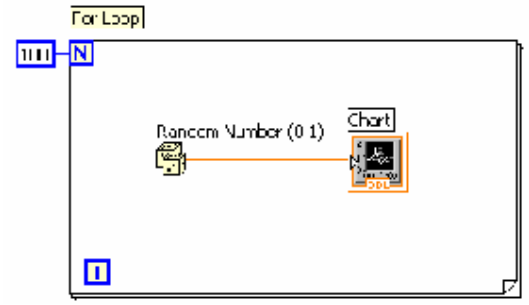
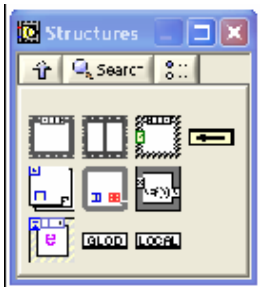


Figura 12 ciclo For Loop

Usando Ciclos

Coloque ciclos en su diagrama seleccionándolos desde la paleta de Estructuras de la paleta de Funciones. Cuando este seleccionado, el cursor del ratón se convierte en un puntero especial que se usa para encerrar la sección del código que desea repetir. Dar clic al botón del ratón para definir la esquina superior-izquierda, dar clic otra vez al botón del ratón en la esquina inferior izquierda, y la frontera del ciclo Mientras es creada alrededor del código seleccionado. Arrastre nodos adicionales adentro del ciclo mientras si es necesario. Ver figuras 13 y 14



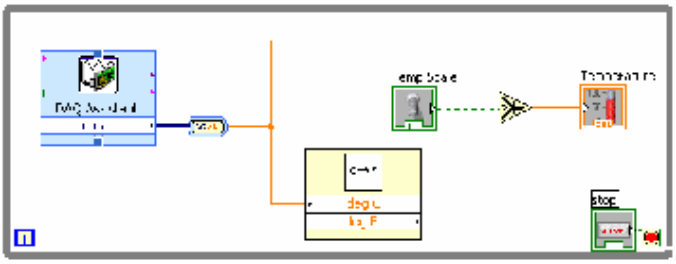


Figura 13 selección de estructuras Figura 14 encerrar código a ser repetido

Abrir y ejecutar instrumentos virtuales Ver Figura 15

1. Seleccione **Start » Programs » National Instruments » LabVIEW 7.0 » LabVIEW** para lanzar LabVIEW. Aparece la ventana de dialogo de **LabVIEW**.

2. Seleccione **Find Examples** del menú de ayuda (Help). La ventana de dialogo que aparece da la lista y enlaces a todos los VIs de ejemplo de LabVIEW.

3. Usted puede buscar ejemplo por categorías, o puede utilizar palabras claves en su búsqueda. Haga clic en la viñeta de búsqueda (**Search**) para abrir el buscador de palabras claves.

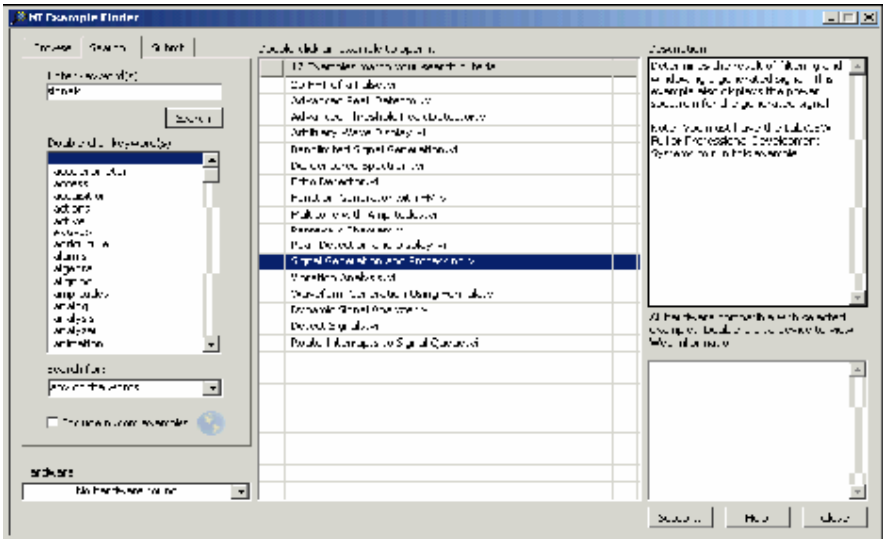


Figura 15 Muestra pantalla ed abrir documentos existentes

Creando un VI

Cuando usted crea un objeto en el panel frontal, una terminal es creada en el diagrama de bloques. Estas terminales le dan acceso a los objetos del panel frontal del código creado mediante el diagrama de bloque. Cada terminal contiene información útil referente al objeto al cual corresponde en el panel frontal. Por ejemplo, el color y los símbolos proporcionan el tipo de dato.
Números de punto flotante y de doble-precisión, son representados con terminales anaranjadas y las letras DBL. Las terminales booleanas son verdes y son representadas por las letras TF.

En general, las terminales anaranjadas deben unirse (cablearse) con las terminales anaranjadas, verdes con verdes, y así sucesivamente. Esta no es una regla que no se puede romper; por ejemplo LabVIEW permitirá al usuario conectar una terminal azul (valor entero) a una terminal anaranjada (valor fraccional). Pero en la mayoría de casos, busque mejor una igualdad en colores.

Los controles tienen una flecha en el lado derecho y tienen un borde grueso. Los indicadores tienen una flecha en el lado izquierdo y un borde fino. Reglas lógicas pueden ser aplicadas al conectar en LabVIEW: Cada cable debe tener una (pero solo una) fuente (o control), y cada cable puede tener varios destinos (o indicadores).

Además de los terminales del panel frontal, el diagrama de bloques contiene funciones. Cada función puede tener múltiples terminales de entradas y salidas. La conexión de estas terminales es una parte muy importante de la programación en LabVIEW.

La herramienta para conectar o de cableado es utilizada para conectarse a los nodos de las funciones. Cuando usted apunte con la herramienta de cableado, apunte con el extremo del cable que cuelga del carrete. Aquí es donde el cable será colocado.

Mientras usted mueve la herramienta de cableado sobre las funciones, observe la viñeta amarilla que aparece. Esto le dirá el nombre de la terminal al que se esta conectando.

Mientras mueva la herramienta de cableado encima de una terminal, esta va a mostrar información. Esto le ayudara a identificar donde se va a unir el cable. Para mas ayuda con los terminales, haga clic derecho en la función y seleccione **Visible Items o Objetos Visibles>>Terminals o Terminales.** Un dibujo de la función será colocada atrás para revelar las terminales de la conexión. Note los colores - estos corresponden a los tipos de datos utilizados por los terminales del panel frontal.

Para ayuda adicional, seleccione **Help >> Show Context Help**, o presione **CTRL+H.** Esto mostrara la ventana de ayuda en contexto. A medida que uno mueva el raton (mouse) sobre la función, esta ventana le mostrara la función, terminales, y una breve descripción

2.2.3 Paquete de instrumento

NI USB 6009

Proporciona la conectividad de ocho canales de entrada analógica (AI) , dos canales de salidas analógico (AO); y 12 canales de entrada/salida (DIO) digitales , y un contador del 32-bits Fig16

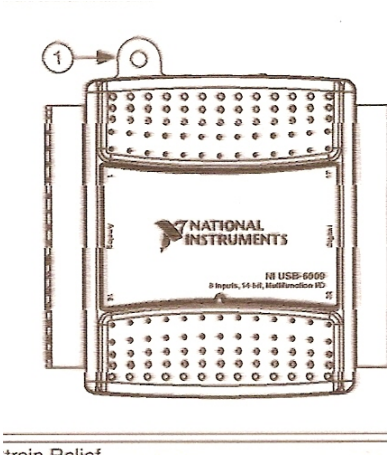


Fig16

Guía de Seguridad

No opere el dispositivo en una atmósfera explosiva, o donde pueda existir gases inflamables o humo. Si tiene que trabajar en este ambiente, construya una caja de seguridad herméticamente cerrada; para proteger las partes internas del dispositivo de algún incendio producido por los circuitos internos del mismo

Para limpiar el dispositivo utilizar una tela seca; y no ubicar donde este pueda tener contacto con el agua

No exceda el número máximo de conexiones del dispositivo

No realizar cableado mientras este conectado a la señal eléctrica ya que se bloquea para protección del mismo

Recuerde que el rango de voltaje que trabaja este dispositivo sea para entrada o salida de señal es de 5 V, tome precaución ya que si alimenta a mas voltaje puede ocasionar un corto circuito

Cuadro de ubicación de cada uno de los puertos analógicos Ver Figura 17

Table 3. Analog Terminal Assignments

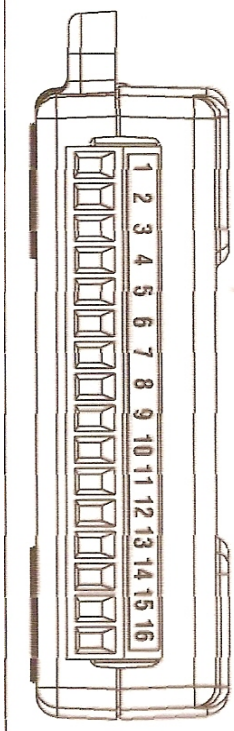
Module	Terminal	Signal, Single-Ended Mode	Signal, Differential Mode
	1	GND	GND
	2	AI 0	AI 0+
	3	AI 4	AI 0–
	4	GND	GND
	5	AI 1	AI 1+
	6	AI 5	AI 1–
	7	GND	GND
	8	AI 2	AI 2+
	9	AI 6	AI 2–
	10	GND	GND
	11	AI 3	AI 3+
	12	AI 7	AI 3–
	13	GND	GND
	14	AO 0	AO 0
	15	AO 1	AO 1
	16	GND	GND

Figura 17 Cuadro de ubicación de canales analógicos USB 6009

Cuadro de canales digitales E/S Ver Figura18

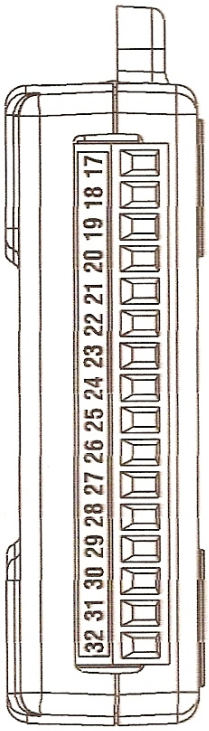
Module	Terminal	Signal
	17	P0.0
	18	P0.1
	19	P0.2
	20	P0.3
	21	P0.4
	22	P0.5
	23	P0 6
	24	P0.7
	25	P1.0
	26	P1.1
	27	P1.2
	28	P1.3
	29	PFI 0
	30	+2.5 V
	31	+5 V
	32	GND

Fig18 Cuadro de señales digitales

2.3. Herramientas para la Generación de proyectos Multimedia

2.3.1. ROGRAMA CAMTASIA RECORDER

La finalidad de este programa es generar archivos .AVI de animación en los que captura partes de la pantalla o su totalidad con todas las acciones que en ella se realizan. Ideal para generar ayudas o explicar procesos en el trabajo con programas informáticos

La ventana principal del programa Camtasia son las que se muestran en las Figuras 19 y 20

Dependerá de las opciones que elija en el menú "View". O también sobre el icono: 



Figura 19 Ventana Compacta

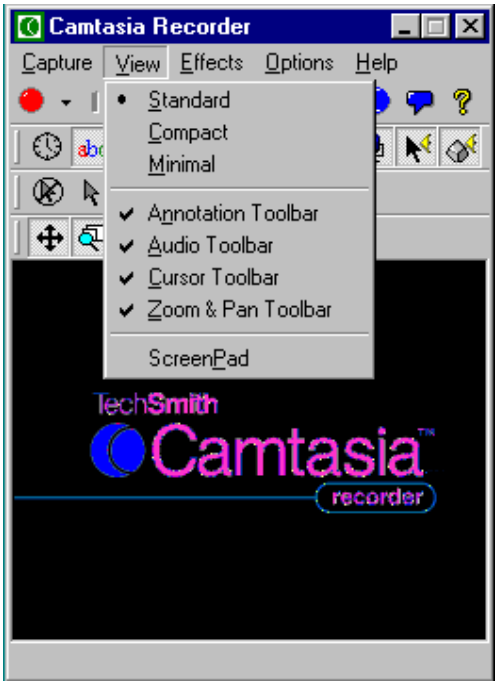


Figura 20 Ventana Estándar

Grabar:

Al abrir el programa se puede seleccionar

- Comenzar un nuevo proyecto grabando la pantalla Ver Figura 21

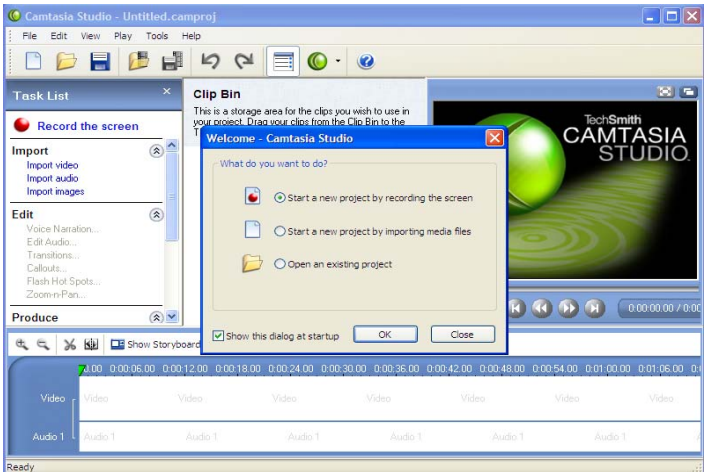


Figura 21 Ventana de inicio nuevo proyecto grabando la pantalla

Una vez elegido un nuevo proyecto se mostrara el asistente de grabación, el cual nos ayudara a configurar el tamaño del área de la pantalla a ser grabada; dependiendo del trabajo a realizar se puede elegir Ver Figura 22

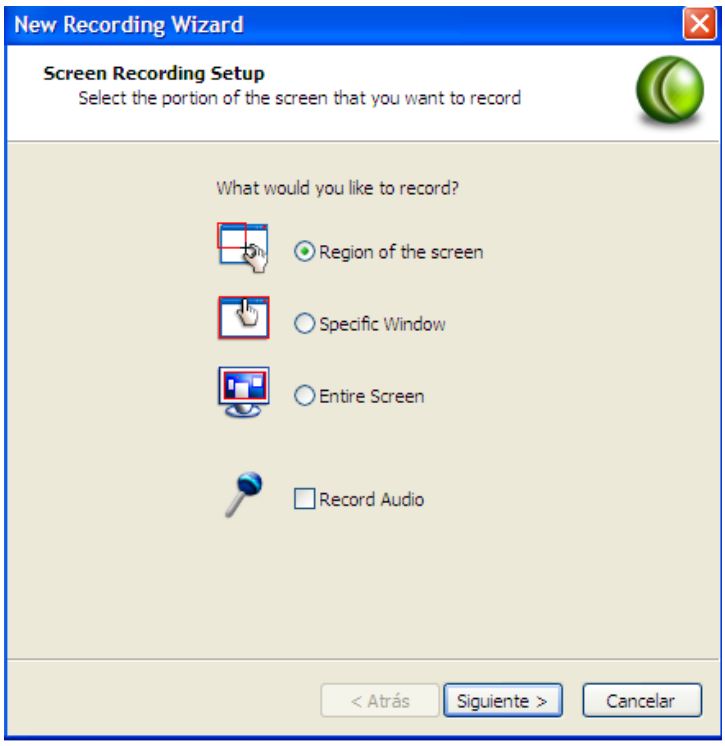


Figura 22 Asistente de nueva grabación

- Región de la pantalla; se puede dar las dimensiones dependiendo de la necesidad del proyecto Ver Figura 23

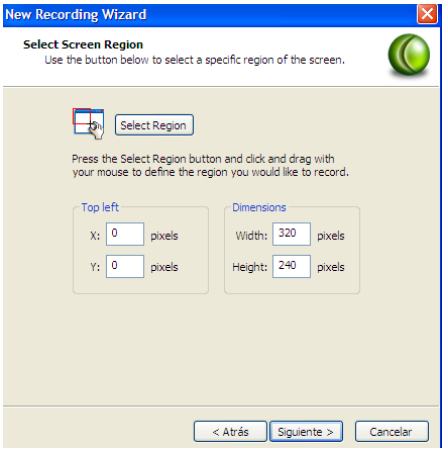


Figura 23 Dimensión del área de pantalla a grabar

- Ventana especifica, se puede elegir una ventana específica para grabar el nuevo proyecto, la diferencia de la anterior radica en que divide a la ventana principal de un programa en ejecución en sub. ventanas. Ver Figura 24

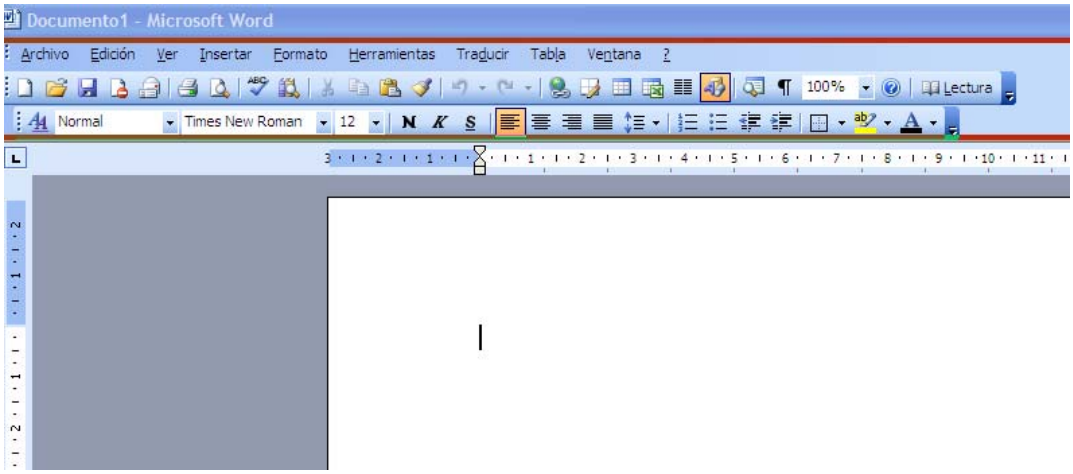


Figura 24 seleccionar una ventana específica

Pantalla completa, esta acción permite grabar las acciones realizadas en toda la pantalla, como se muestra en la figura 25 no se puede especificar el tamaño de la pantalla

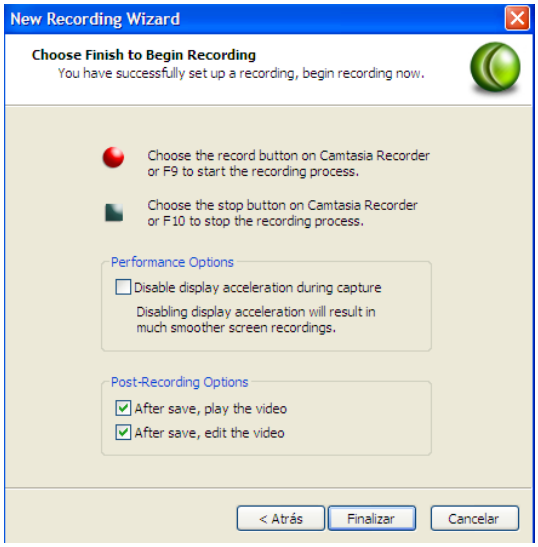


Figura 25 Grabar toda la pantalla

Todos los proyectos realizados en este programa se les pueden incluir audio, es necesario configurar el recurso de entrada tal como se muestra en la figura 26

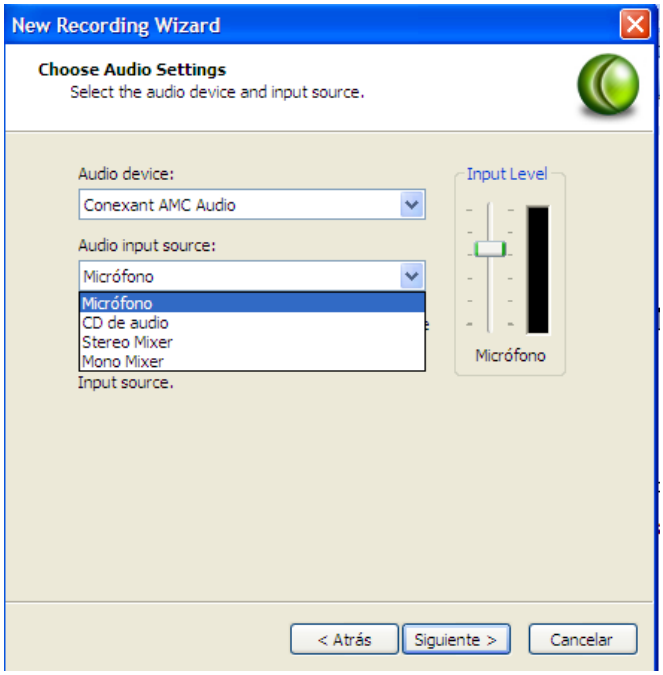


Figura 26 Elegir escena de audio

Otra ventaja que brinda el programa Camtasia es que se puede editar archivos multimedia con los siguientes formatos avi, bmp, mpg, mepg, gip, jpg.

Una vez que se ha delimitado el área que se va a grabar, además de especificar si se va a incluir audio. Se debe comenzar la grabación de dos formas presionando la tecla F9 o el siguiente botón que se muestra en la figura 27



Figura 27 Botón inicio grabación

Si se desea parar la grabación se puede presionar la tecla F10 o el siguiente botón Ver figura 28

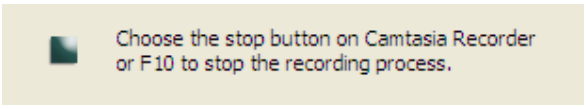


Figura 28 Botón de parada

2.3.2PROTEL 99

Protel 99 SE.

Existen diversas aplicaciones orientadas al desarrollo de circuitos eléctricos y electrónicos, la mayoría de estos permiten realizar los diagramas eléctricos de conexiones, diseñar tarjetas electrónicas y muy pocos permiten realizar la simulación de los circuitos diagramados.

Protel 99 SE, se encuentra dentro estas escasas aplicaciones que permiten realizar todas estas labores permitiendo además generar vistas en tres dimensiones de las tarjetas elaboras por los usuarios de este producto. Cuenta también, con una amplia librería de elementos dentro de la cual se ubican los elementos electrónicos de las más importantes empresas distribuidoras de estos dispositivos, convirtiéndose en una poderosa herramienta para el desarrollo de circuitos electrónicos. Ver Figura 29

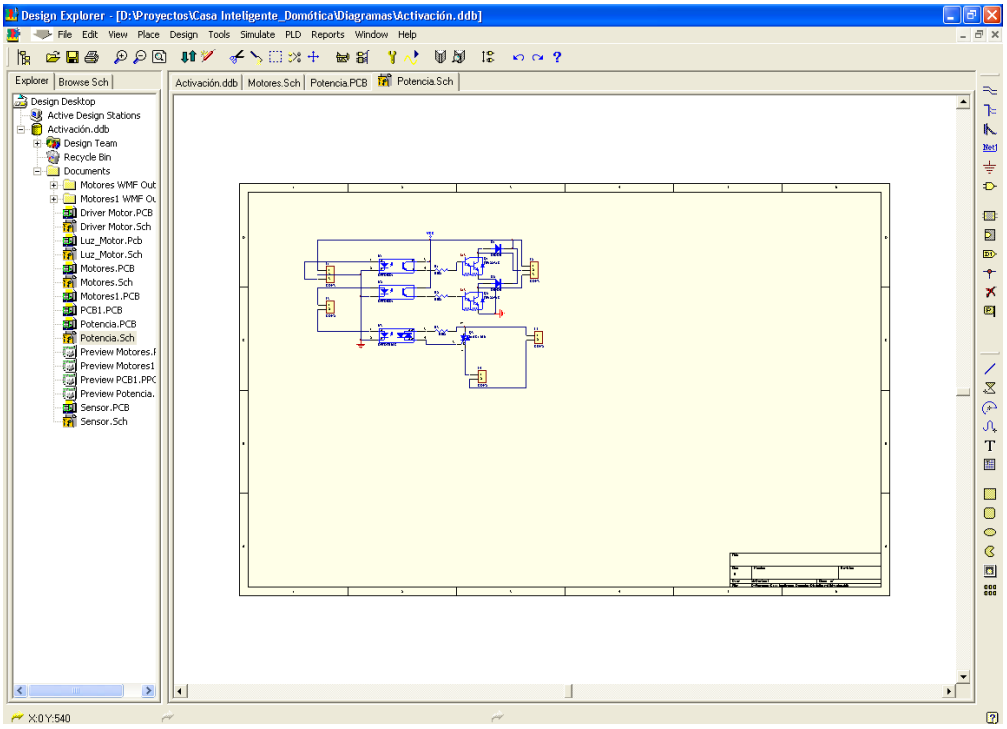


Figura 29 circuito diseñado en Protel99

La ventana contiene el área destinada a las barra de herramientas, el área de navegación, diagramación que a continuación se detallan.

Barra de herramientas estándar.- Consta de varias opciones como archivo, que permite la creación de nuevos documentos, guardar documentos, imprimirlos, importarlos o exportarlos. Edición, que incluye funciones para modificar el documento tales como deshacer y rehacer cambios, cortar, pegar, copiar, seleccionar, mover, buscar o reemplazar elementos en el documento. Ver, con funciones como manejo del zoom, actualizar el documento, elegir las barras de herramientas que se muestran. Colocar, las funciones que ofrece este menú están destinadas a la colocación de nuevos elementos, conectar unos elementos con otros, dibujar figuras geométricas y editar símbolos.

Diseño, permite la creación de la tarjeta electrónica del documento o diagrama de conexiones actual, además permite abrir o crear librerías, nombrar elementos o líneas de conexión. Herramientas, aquí se puede escoger tareas para buscar componentes en las librerías, nombrar automáticamente a los componentes de un documento, seleccionar los elementos correspondientes para crear la tarjeta electrónica o escoger las preferencias (configuración, de unidades, número de lados de pistas en la tarjeta electrónica, tamaño

del área de desarrollo, entre otras) del documento actual. Simulación, esta función permite simular el circuito diseñado, permite ubicar puntos de prueba para señales de voltaje o corriente y seleccionar fuentes de alimentación. PLD, esta función permite compilar y configurar el código de simulación para que sea transportable.

Reportes, elabora una lista detallada de los elementos y sus nombres respectivos, muestra la generación de archivos que se crean durante el desarrollo del documento tales como errores, puntos de prueba en el documento entre los más importantes. Ventana, permite escoger la visualización de las ventanas de los documentos abiertos así como también navegar entre las ventanas que se encuentran abiertas.

Ayuda, aquí se muestran la ayuda de todas las herramientas y opciones que Protel 99 ofrece para el desarrollo de los diferentes circuitos eléctricos y electrónicos que el usuario desee elaborar. La barra de herramientas descrita con sus opciones se muestra a continuación Ver Figura 30

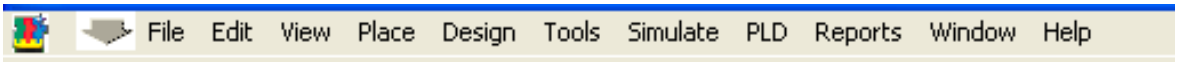


Figura 30 Barra de herramientas programa Protel

Barra de herramientas principal, en ella se encuentra atajos a las funciones más comunes que Protel 99 permite realizar y que se encuentra alojadas en las los diferentes popups de las barra de herramientas estándar. Ver Figura 31



Figura 31 Barra de herramientas principal

El área de trabajo dispone de dos secciones una continuación de la otra y que tiene que ver directamente con la diagramación de los circuitos que intervienen en el documento creado. La primera sección corresponde al navegador del documento, el mismo que permite desplazarse por todos los documentos creados durante el desarrollo del proyecto y que además sirve para seleccionar la librería de componentes que intervienen en el diagrama de conexiones, cuando un elemento es seleccionado la figura correspondiente al elemento se indica mediante un gráfico en aparte inferior de esta área, de esta manera el usuario puede identificar si el elemento escogido es el correcto o no, los gráficos mostrados de los elementos están estandarizados tanto en la simbología (forma de

nombrarlos) como en su representación (gráfica) de tal manera que no difieren de uno u otro programa empleado para el desarrollo de circuitos. Además los elementos de cada una de las librerías son susceptibles de edición y cambio pero no es recomendable elegir esta opción, para superar cualquier defecto o no coincidencia en el elemento, lo más aconsejable es la creación de un nuevo con las características específicas para este que también se puede realizar desde la opción de edición de esta sección que se muestra en la figura 32



Figura 32 Navegador del documento

Por último se dispone de la sección de trabajo en donde se realizan todas las conexiones respectivas entre los elementos que se han seleccionado de las librerías que dispone esta aplicación, en esta sección se pueden mover y ordenar los diferentes diagramas realizados, cambiar las conexiones ubicar texto y otras opciones más para que las conexiones realizas se muestren de la mejor manera, también se pueden reemplazar conexiones, elementos o ubicar nuevos componentes desde la barra de herramientas ubicada en el costado derecho del área de trabajo, esta paleta tiene atajos a las funciones mencionadas en la barra estándar que facilitan la edición del documento. Ver figura 33

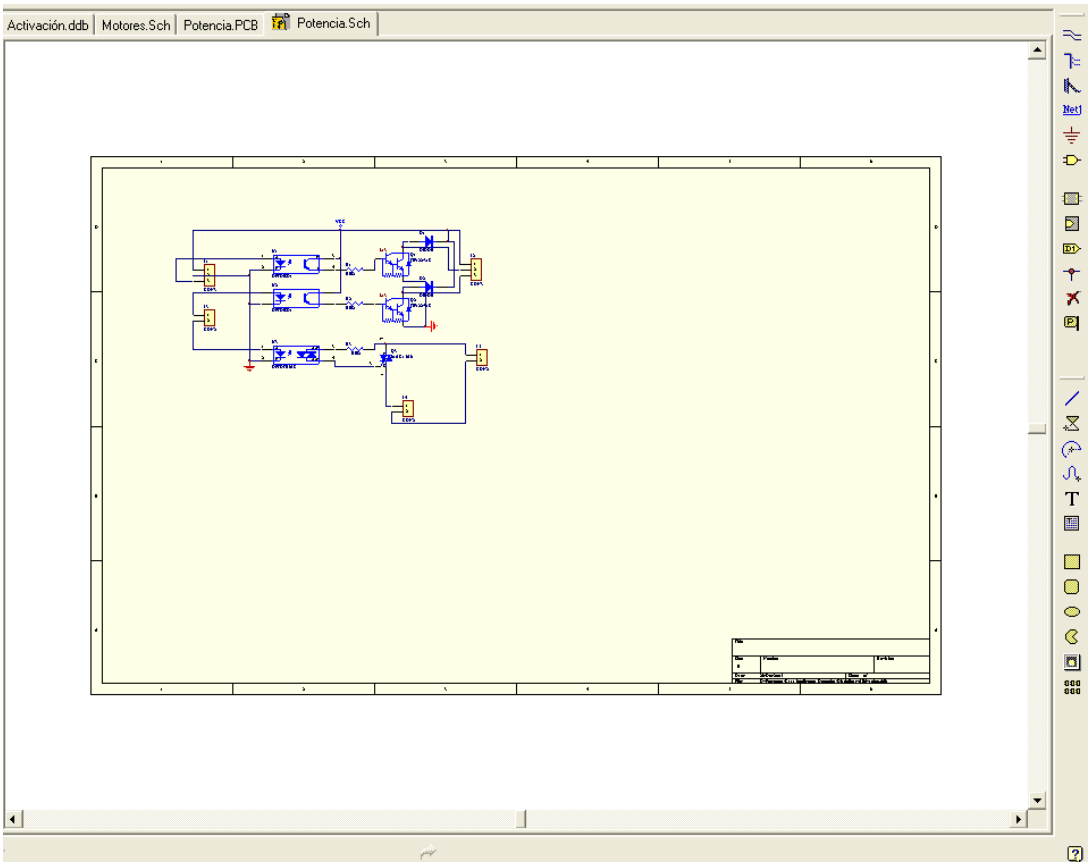


Figura 33 Área de trabajo

Este paquete de diagramación, simulación y diseño de tarjetas electrónica no se encuentra muy difundido, sin embargo ofrece mayores beneficios que los existentes en el mercado y que son de uso más común. La gran limitante para que este paquete no sea tan utilizado es el costo de la licencia del programa, sin embargo para el desarrollo de los circuitos de la presente aplicación se utilizó una versión demo la misma que funciona con todas las opciones que Protel 99 SE, por un periodo de treinta días.

CAPITULO III

3.1. Planificación

3.1.1. Análisis de las guías y talleres

El trabajo realizado en la presente disertación consta de seis prácticas las cuales se detallan a continuación:

Práctica No 1: Control de intensidad; en esta práctica se controla el encendido de una de una lámpara por medio de entrada de datos digitales (sonidos).

Práctica No 2 Control de temperatura; el objetivo principal es controlar la temperatura ambiente para encender un ventilador y así mantener la temperatura ambiente

Práctica No 3 Control de frecuencia se va ha controlar la velocidad de rotación de un motor DC

Práctica No 4 Contador automático; el objetivo principal es el de encender un motor para poder mover, luego de haber contado doce botellas

Práctica No 5 Detector de incendio; controlara una cierta área de trabajo para encender una alarma de sonido si se esta produciendo un encendió

Práctica No 6 Seguridad mediante pulsadores con un display controlara el ingreso de una clave para poder abrir un dispositivo eléctrico

3.2. Diseño de las guías

- Las guías constan de archivo multimedia sobre el uso del Proto Board para armar la placa electrónica
- Archivo realizado en Power Point acerca de cómo se hace Circuitos Impresos
- Archivo multimedia de la programación en LabVIEW así como en Protel 99 de cada una de las prácticas
- Data Sheet de cada instrumento electrónico usado

3.2.2. Desarrollo de las guías

- Práctica No 1 Control de Intensidad ver Anexo1
- Práctica No 2 Control de Temperatura ver Anexo2
- Práctica No Contador Automático ver Anexo3
- Práctica No 4 Control de frecuencia de un motor DC ver Anexo4
- Práctica No 5 Detector de Incendio ver Anexo5
- Práctica No 6 Teclado y Display ver Anexo6

3.3 Lista de guías multimedia

Multimedia sobre el uso del Proto Board para armar la placa electrónica
Placa Proto board

Programas realizados en LabVIEW. **Documentos .AVI**

- Control de Intensidad
- Control de temperatura
- Contador
- Control de de frecuencia
- Detector de incendio
- Teclado y Display

Programas realizados en Protel 99. **Documentos .AVI**

- Control de Intensidad
- Control de temperatura y contador
- Detector de incendio
- Teclado y Display
- Variador de frecuencia

Programación de LabVIEW de cada Proyecto

- Control de Intensidad.vi
- Control de Temperatura.vi
- Dectector de Incendio.vi
- Teclado y Display.vi
- Variador de frecuencia.vi
- Contador.vi

Programación en Protel 99

- Control de Intensidad electrónica.SCH
- Control de Temperatura electrónica.SCH
- Detector de Incendio electrónica.SCH
- Teclado y Display electrónica.SCH

Materiales

- Modulo USB 6009
- Placa electrónica de cada practica
- LabVIEW 7.1 Versión estudiantil

CAPITULO IV

4. VALIDACIÓN Y VERIFICACIÓN DE RESULTADOS

4.1 Verificación de la Hipótesis.

En la presente disertación y para poder verificar la hipótesis se utilizo el modus Ponendo Ponens de lógica proporcional que dice: “Dada una proposición condicional y la afirmación del antecedente, puede concluirse la afirmación del consecuente”

$$P \longrightarrow Q$$

$$\frac{P}{Q}$$

La hipótesis planteada en este proyecto es “Mediante la implementación del Laboratorio de Automatización y Control y el desarrollo de guías, los estudiantes de la Escuela de Ingeniería de Sistemas de la PUCESA podrán realizar sus prácticas en la Universidad lo que facilitará su aprendizaje en el campo de la Automatización y Control”. Donde:

P: Mediante la implementación del Laboratorio de Automatización y Control y el desarrollo de guías (Premisa Verdadera).

Q: Facilitará su aprendizaje en el campo de la Automatización y Control (Premisa Verdadera)

Entonces

P: Mediante la implementación del Laboratorio Automatización y Control y el desarrollo de guías $\longrightarrow$ **Q:** Facilitará su aprendizaje en el de campo de la Automatización y Control

P: Mediante la implementación del Laboratorio Automatización y Control y el desarrollo de guías

---

Q: Facilitará su aprendizaje en el de campo de la Automatización y Control

Con lo que queda demostrada la hipótesis del proyecto mediante este teorema

4.2 Concluciones

Una vez finalizado el proyecto se concluye:

La adquisición del Modulo USB 6009 como también del software LabView versión estudiantil fue necesaria para lograr las prácticas planteadas

Los estudiantes podrán practicar y desarrollar los conocimientos adquiridos en electro logia

El desarrollo de proyectos en LabView de automatización es más fácil debido a su lenguaje de programación visual, fácil conexión con dispositivos externos; que realizarlos en PLC´s

La creación de VI´s cumple con las necesidades del usuario y a la medida de la empresa, ya que los instrumentos tradicionales vienen configurados de fábrica y son más costosos

4.3 Recomendaciones:

- Realizar nuevos proyectos en donde se pongan en uso el Modulo USB 6009 como también los conocimientos en electro logia
- Adquirir mas libros de electrónica y automatización para la biblioteca
- Incluir en el pensún de electro logia la construcción de VI´s, que son la tendencia para el futuro ya que se pueden construir de a cuerdo a las necesidades del usuario y no a la del constructor
- Adquirir mejores productos de la Nacional Instrument que faciliten un mejor análisis de automatización industrial
- Revisar los archivos multimedia para lograr el éxito en las prácticas
- Conectar las tarjetas electrónicas a un nivel de voltaje máximo de 12 V
- Cuando se encuentre trabajando el Modulo USB 6009 no desconectarlo ya que puede sufrir una avería interna
- Mantener en un lugar fresco y sin humedad las tarjetas electrónicas así como el modulo USB6009

ANEXO No 1

Práctica No 1

Tema: Control de intensidad de una Lámpara

Objetivo

Realizar un proyecto en LabView que interactué con el modulo USB 6009 y un circuito electrónico que sirva de entrada digital y salida analógica para poder así controlar la intensidad luminosa de un foco de 4.8 V

Descripción:

Se controlara la intensidad luminosa de un foco por medio del uso de dos puertos (entrada/salida) del USB 6009, puede ser manual o automática. Los datos de entrada los va a receptar por medio de señales digitales (ruido por aplausos) los cuales tendrán el siguiente significado:

- 1 aplauso media intensidad del foco.
- 2 aplausos máxima intensidad
- 3 aplausos apagar el foco

Se va a utilizar un canales digital y además un canal analógico del modulo USB

Entrada canal P0

Salida canal p17

Materiales:

Se necesitaran los siguientes materiales electrónicos:

Para el control de voltaje de entrada y salida al USB 6009:

- 1 regulador de voltaje de 5v
- 1 regulador de voltaje de 12v
- 2 diodos
- 3 capacitores de 1uF

  Receptor de Señal
- 1 amplificador operacional
- 1 potenciómetro
- 3 resistencias.
- 1 micrófono.
- 1amplificador operacional LW358

- 1 transistor
- 1 foco de 4.8 V
- 2 conectores de 2 pines

Paso 1

Realizar la placa electrónico ver diagrama de conexión figura I1. Se debe colocar los instrumentos electrónicos según el diagrama además tener en cuenta la ubicación de los pines correspondientes ya que algunos circuitos como los diodos (D1, D2,D3), reguladores de voltaje (U1, U2), amplificador operacional (U4), potenciómetro(Q1); ya que estos elementos tienen polaridad y cada pin tiene una función especifica.

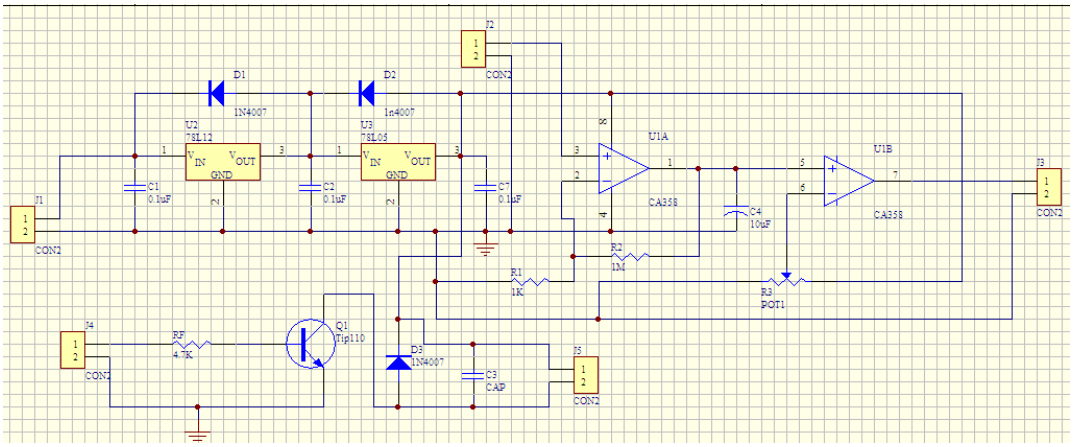


Figura I1 se muestra de conexión circuito intensidad de una lámpara

Paso 2

Programación en LABVIEW

Abrir el programa Labview

- 1.-Botón inicio
- 2.-Todos los programa
- 3.- Clic National Instrument LabView Ver Figura I2

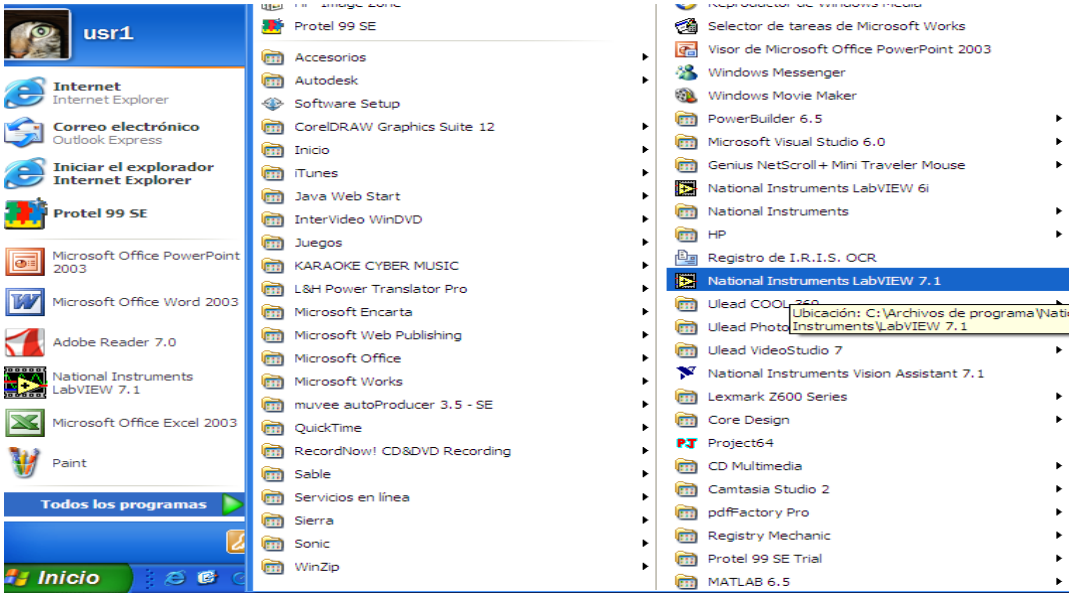


Figura I2 Inicio trabajo en LabVIEW

A continuación aparecerá la siguiente ventana en donde se puede:

- Abrir un archivo existente.
- Crear un nuevo VI
- Crear un nuevo VI con el Data Acquisitione with NI-DAQmx.vi
- Seleccionar Nuevo Blank VI Ver Figura I3

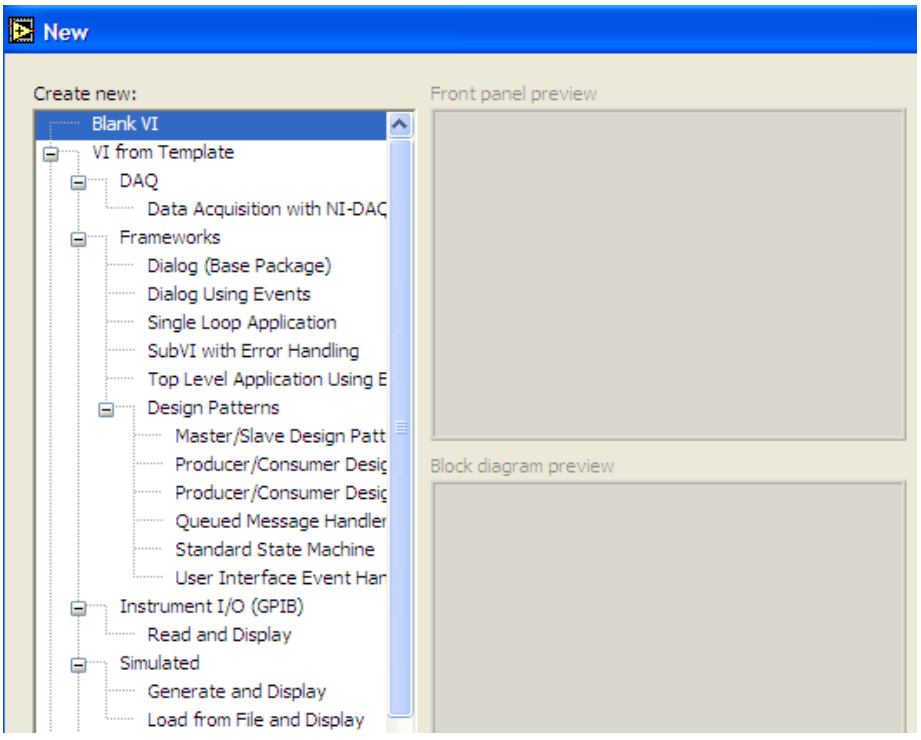


Figura I3 Creación de un nuevo VI

Una vez completados estos pasos se mostrara el Panel frontal así como el Diagrama de bloque de la nueva aplicación. Ver Figura I4

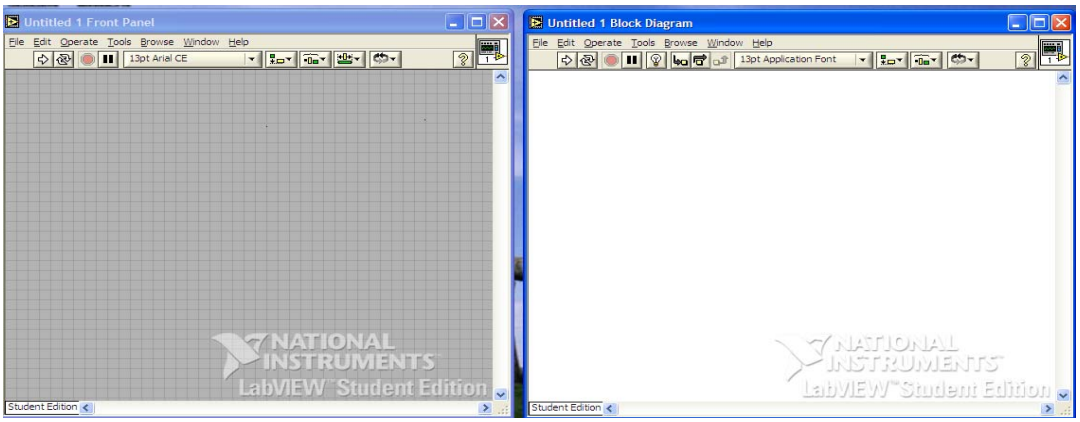


Figura I4 Panel Frontal, Diagrama de Bloque

En el panel frontal insertar un indicador numérico

1. Clic del Mouse en el panel frontal
2. En la ventana de Controls elegir Num Ctrls.
3. Insertar Num Ctrl.
4. Para cambiar su Label doble clic sobre el indicador colocar de Label Dato.

Se puede ver que en el diagrama de bloque ya se muestra el indicador numerico

Ver Figuras I5 y I6

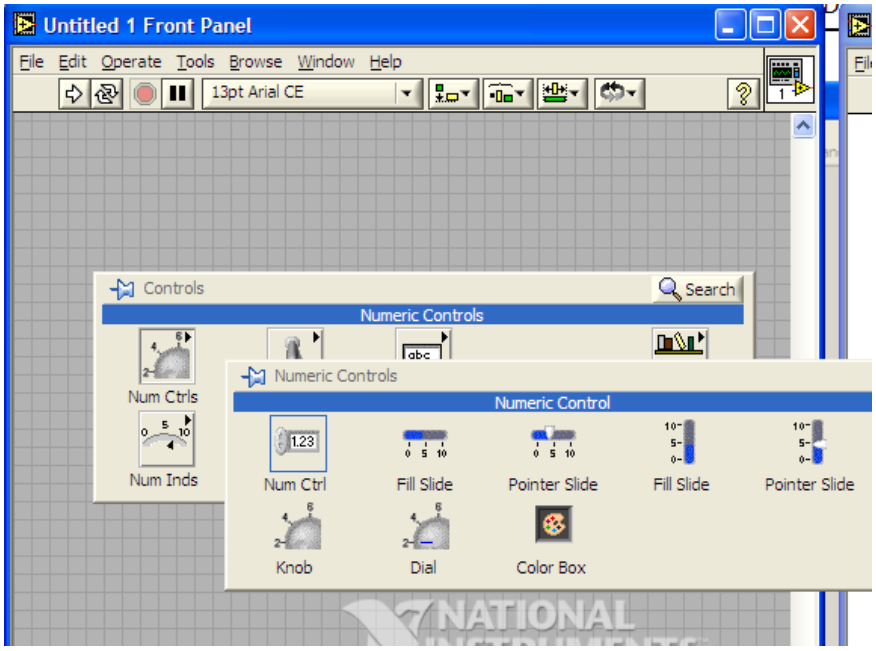


Figura I5 Ventana de control Numeric Control

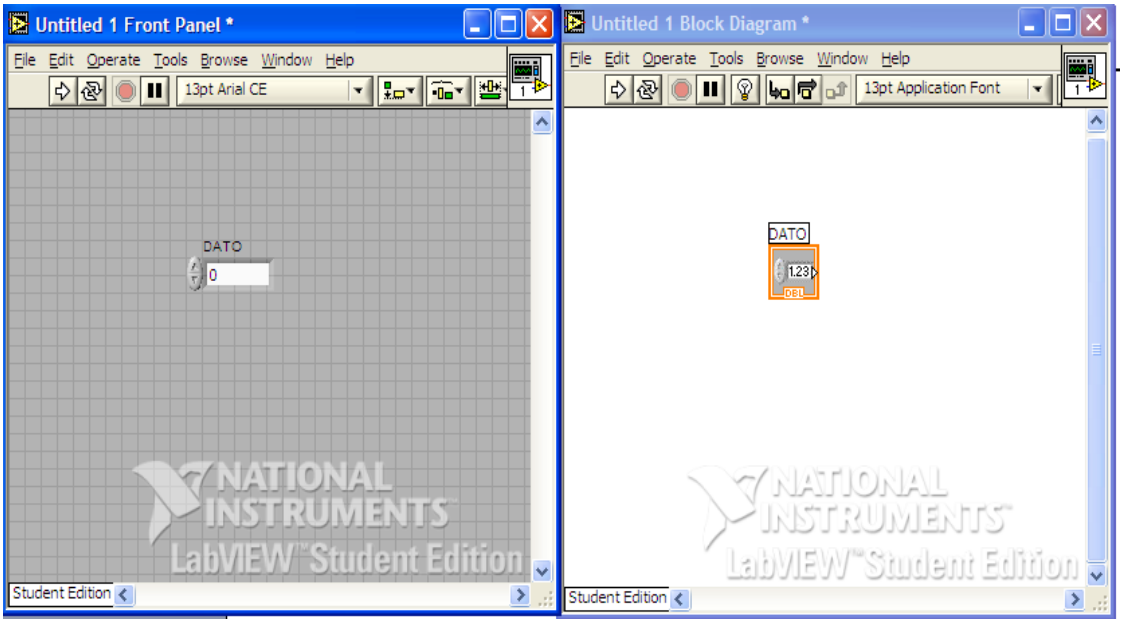


Figura I6 Vista de um indicador numérico panel frontal como diagrama de bloque

Este indicador servirá de referencia para tener como rangos máximo y mínimos al momento de encender la lámpara, ya que esta se encenderá un su mínima expresión cuando le reciba de voltaje 1.6 V y máxima intensidad cuando sea 2.15 V. En la ventana de propiedades del indicador numérico cambiar los rangos mínimos y máximos del indicador.

- Clic sobre el indicador
- Elegir Properties.
- Data ranger
- Escribir en:
- Default Value 1.6 Minimun: 1.6 Maximun 2.16

Ver Figuras I7 y I8

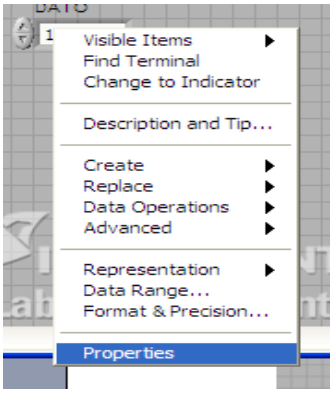


Figura I7 Ventada de configuración del indicador

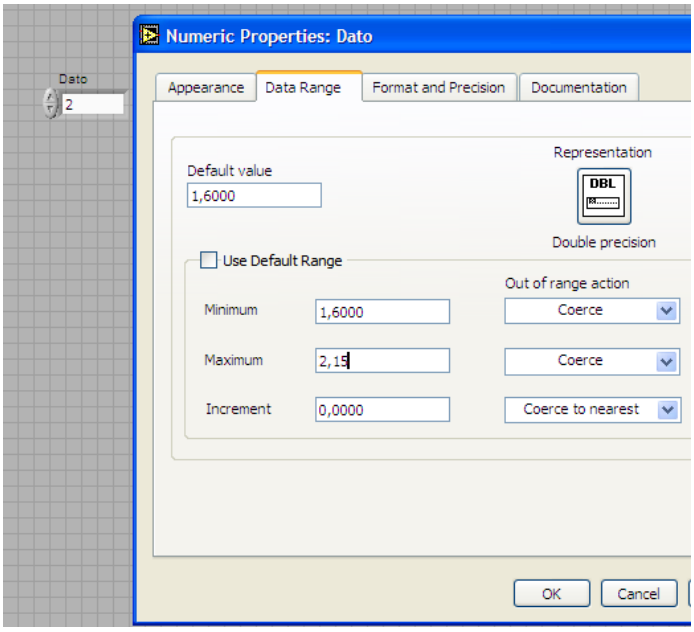


Figura I8 ventana de propiedades del indicador numérico

Se procede a calcular el factor de conversión de iluminación de la lámpara hasta que sea el 100%, para esto realizar los siguientes pasos:

1. En el diagrama de bloque clic se mostrará la ventana de Funciones
2. Elegir: All Funtions
3. Seleccionar Numeric
4. Extraer la función de resta Ver Figuras I9 y I10

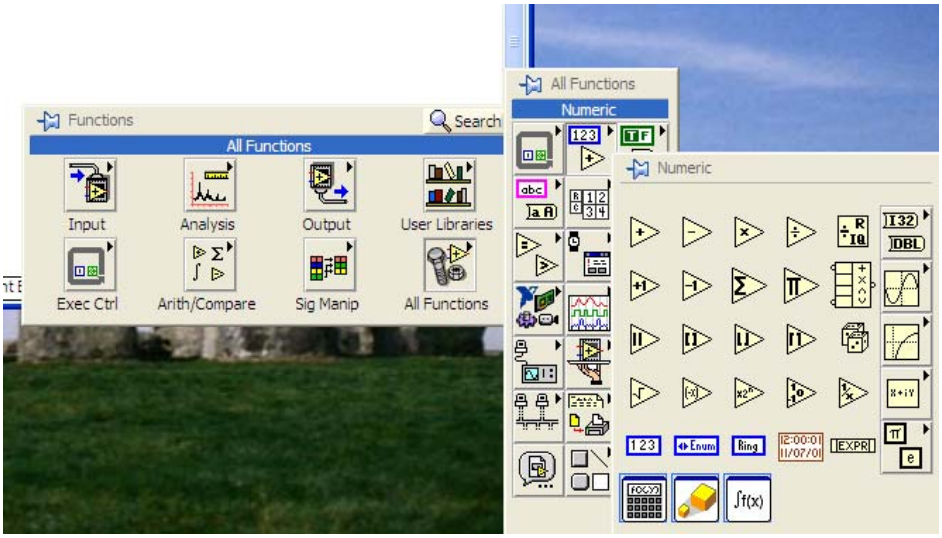


Figura I9 Ventana de funciones, función Numeric

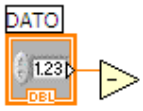


Figura I10 Vista del comando resta en el diagrama de bloque

- Este comando posee dos entradas y una salida
- La primera entrada conectar al indicador numérico en la otra vamos a crear una constante de 1,6.
- Clic en la segunda entrada del indicador (-)
- Elegir Constan asignar el valor de 1,6 Ver Figuras I11-I12

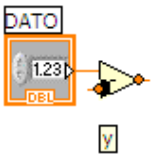


Figura I11 Conectando las entradas de la función resta a la variable dato

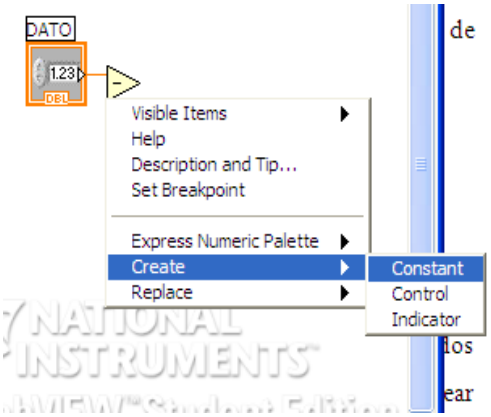


Figura I12 crear una constante e la función resta

Siguiendo estos pasos tendremos el siguiente diagrama Figura I13

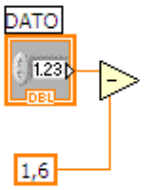


Figura I13 Se muestra a la variable dato menos la constante 1,6

Se a proceder a calcular el porcentaje de variación de voltaje que va a ser suministrado a nuestra lámpara de prueba; para esto necesitamos al resultado obtenido

multiplicarlo por 100 y a este dividirle para 0,55 que es un porcentaje obtenido de restar los limites tanto superior (2,15) como inferior (1,6); una ves obtenido este resultado se mostrará en el panel frontal por medio de un indicador numérico

- En el diagrama de bloque clic; se mostrará la ventana de Funciones
- Elegir All Funtions
- Seleccionar Numeric
- Extraer la función de multiplicación Ver figuras I14, I15

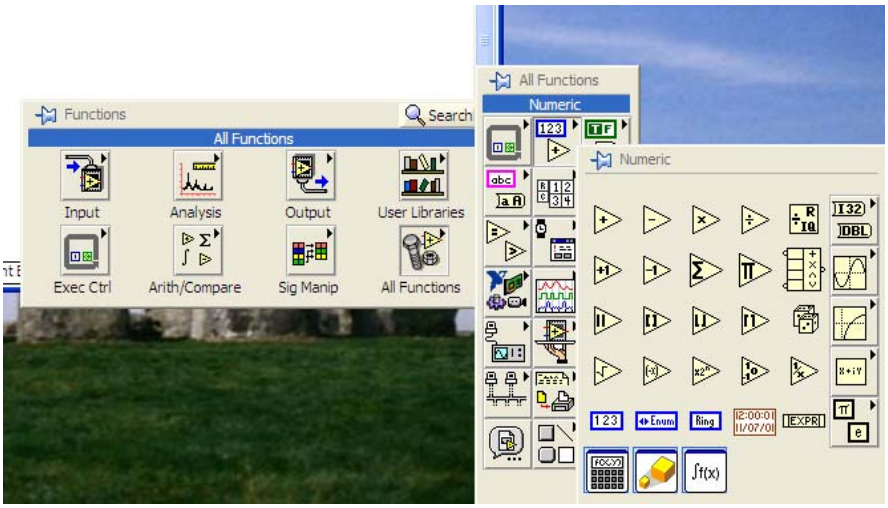


Figura I14 Ventana de funciones; funciones Numeric

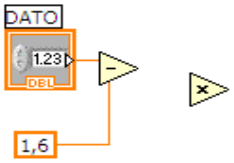


Figura 15 Insertar la función multiplicación en el diagrama de bloque

Esta función posee dos entradas y una salida la primera conectar, a la salida de la función resta; en la otra vamos a crear una constante de 100.

- Clic en la terminal de la función multiplicación
- Elegir Constan asignar el valor de 100 Ver Figuras I16-I17

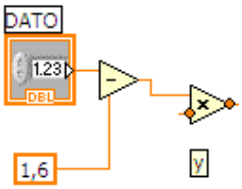


Figura I16 Muestra la unión (cableado) de la salida de la función menos a la entrada de la función por

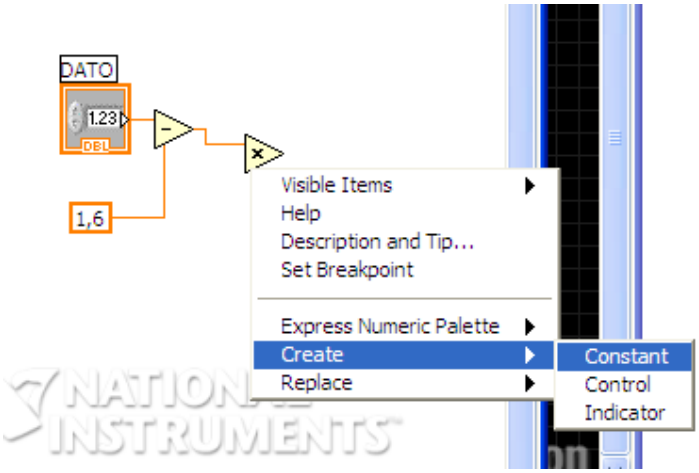


Figura I17 Crear una constante a la función multiplicar

- En el diagrama de bloque clic se mostrara ventana de Funciones
- Elegir: All Funtions
- Seleccionar Numeric
- Extraer la función de división Ver Figuras I18

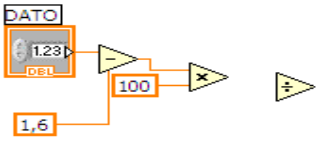


Figura I18 Insertar la función división en el diagrama de bloque

Esta función posee dos entradas y una salida la primera conectar, ala salida de la función multiplicación ; en la otra vamos a crear una constante de 0,55

- Clic en la función dividir (/)
- Elegir Constan asignar el valor de 0,55 Ver Figuras I19-I20

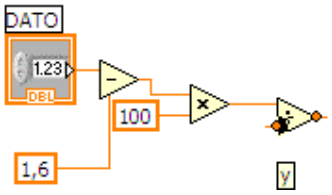


Figura I19 Muestra la unión (cableado) de la salida de la función multiplicación a la entrada de la función dividir

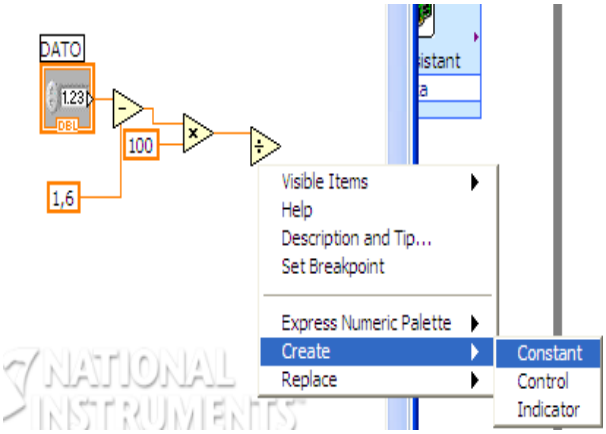


Figura I20 Crear una constante a la función dividir

Una vez completado estos pasos vamos a tener el porcentaje de corriente suministrado a la lámpara; para poder visualizar proceder a realizar en el panel de control a insertar un indicador numérico para este unir con el resultado anteriormente obtenido

1. Clic panel de control en la barra de control
2. Elegir Num Inds
3. Seleccionar Meter VerFigura I21

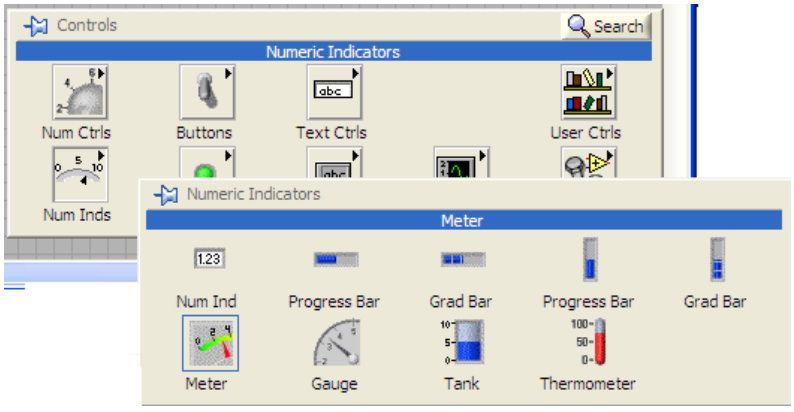


Figura I21 Barra de controles, Numeric Indicators

En el diagrama de bloque unir el resultado anteriormente calculado con el indicador numérico que se visualiza. Figura I22

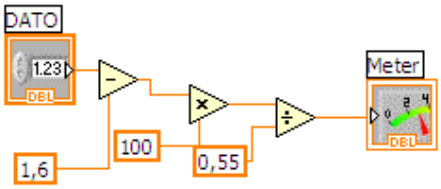


Figura I22 Diagrama de bloque programación porcentaje de intensidad

Para cambiar la escala del indicador ubicar el rango mas alto

Dar clic y escribir el rango máximo; como calculamos el porcentaje ubicar el 100;

Para cambiar el label dar clic Meter y cambiar por Porcentaje Ver Figura I23

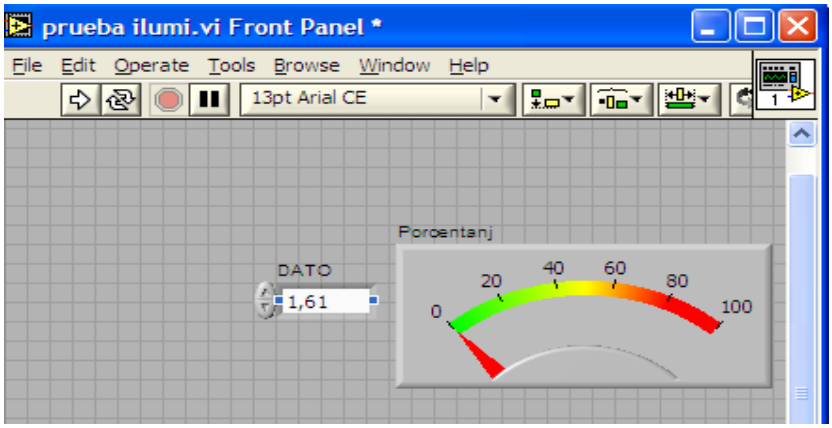


Figura I23 panel frontal vista del indicador gráfico

Insertar un DAQ assistent que será el encargado de controlar la señal de salida hacia el transistor.

- Clic en el diagrama de bloque
- En la barra de funciones seleccionar Output Ver figura I24

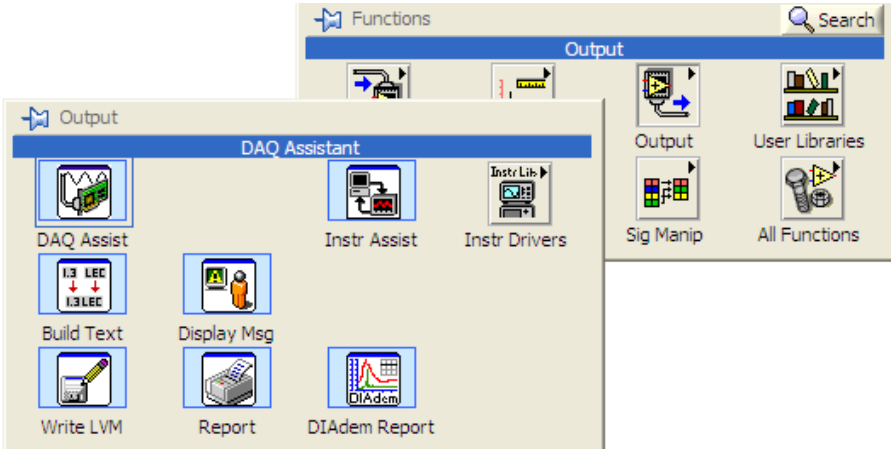


Figura I24 Barra de funciones, Output

- Seccionar Analog Output Ver Figura I25

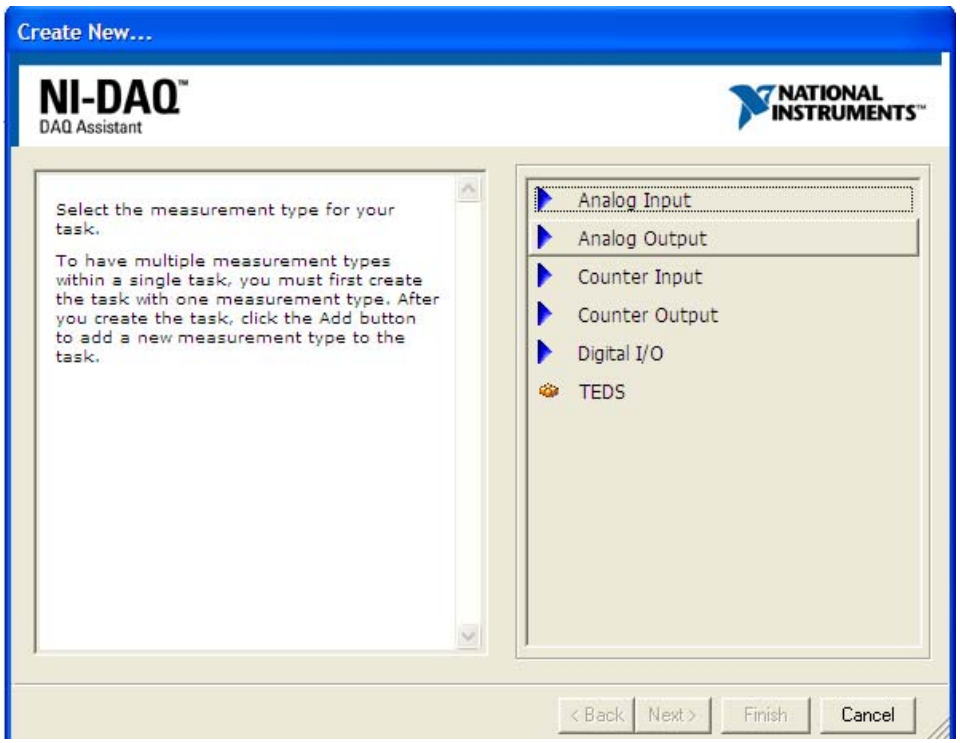


Figura I25 Ventana para elegir el tipo de entrada o salida

Una vez que se ha realizado este ingreso manual procedemos a mejorar la presentación insertando dos botones los cuales nos servirán para incrementar y disminuir el voltaje suministrado para lo cual desarrollar los siguientes pasos.

- Clic en el panel frontal
- Elegir Buttons
- Seleccionar Push Button (dos veces ya que necesitamos dos botones) Ver Figura I26

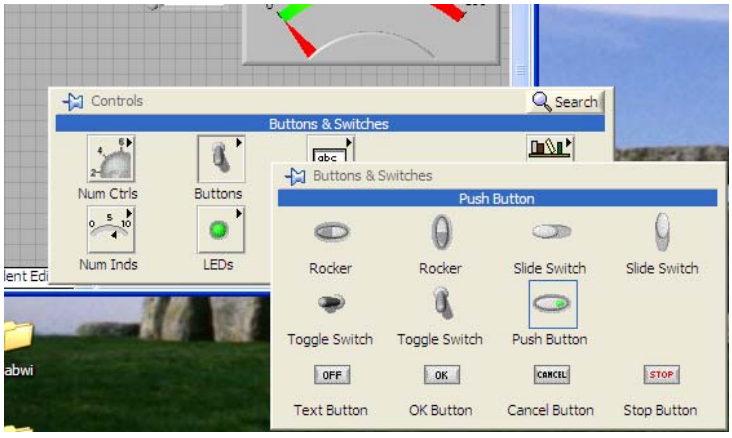


Figura I26 Insertar Push Button en el panel frontal

Cambiamos el Label de cada botón

- Clic sobre cada botón en el panel frontal y escribir incrementar en el primero y disminuir en el segundo Ver Figura 27

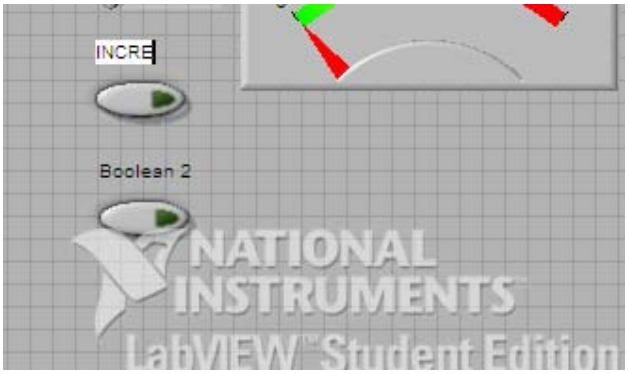


Figura I27 cambiar la presentación para una mejor interfaz

Una vez que se ha completado estos pasos el panel frontal y el diagrama de bloque quedara como se muestra a continuación Figura I28

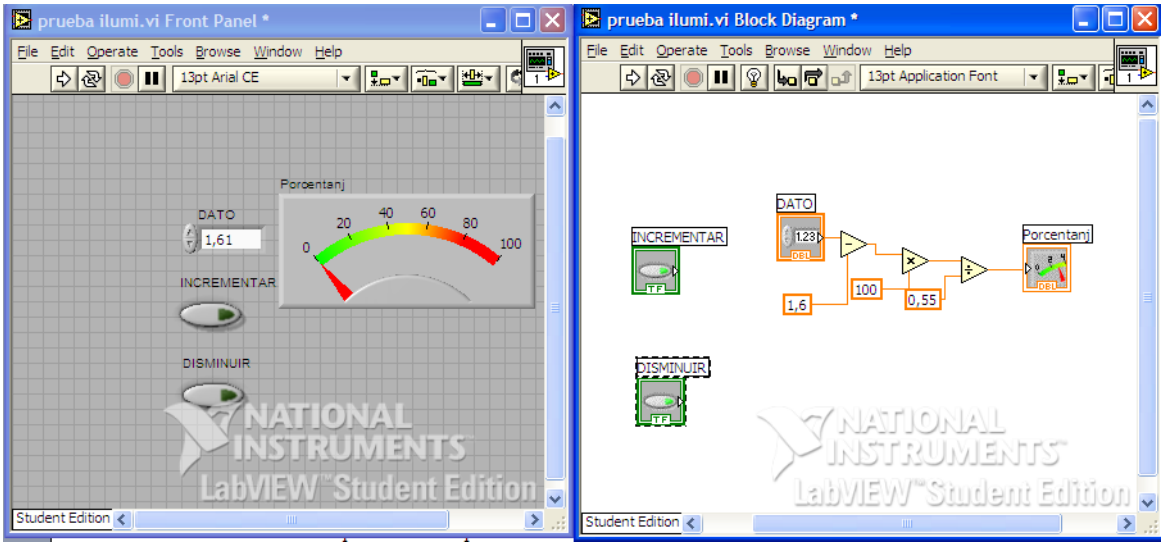


Figura I28 vista preliminar del proyecto

Programar el botón incrementar para así aumentar manualmente; en donde la variable dato va a ir aumentando en un incremento de 0,01 hasta llegar al limite superior que es de 2,14 en donde la variable de salida será el dato aumentado, además insertaremos un tiempo que servirá de espera un tiempo limitado para visualizar mejor el incremento de voltaje; desarrollar los siguientes pasos,

- Crear una variable local del DATO
- Clic derecho sobre dato
- Elegir Create
- Seleccionar Local Variable Ver Figura I29

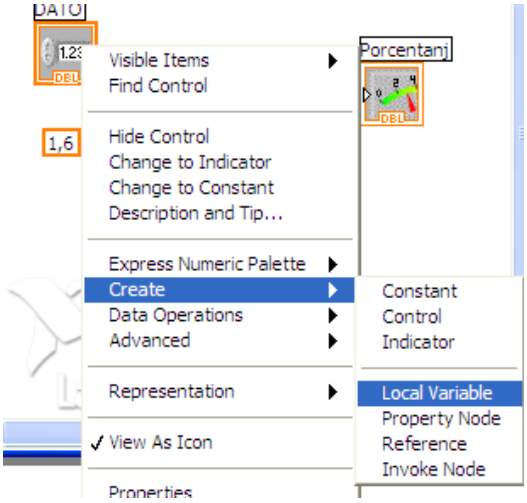


Figura I29 Crear una variable local

- Clic sobre el diagrama de bloque
- Seleccionar All Funtions
- Numerics
- Seleccionar la función sumar (+) Ver Figura I30

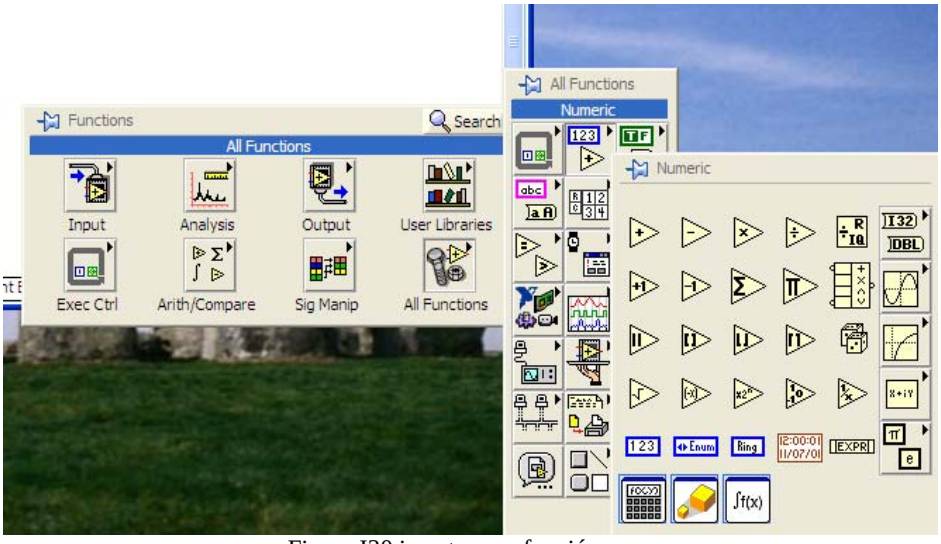


Figura I30 insertar una función suma

- En la barra de funciones seleccionar All Funtions
- Elegir Comparison
- Seleccionar la función menor o igual que, así como la función de comparación Ver Figura I31

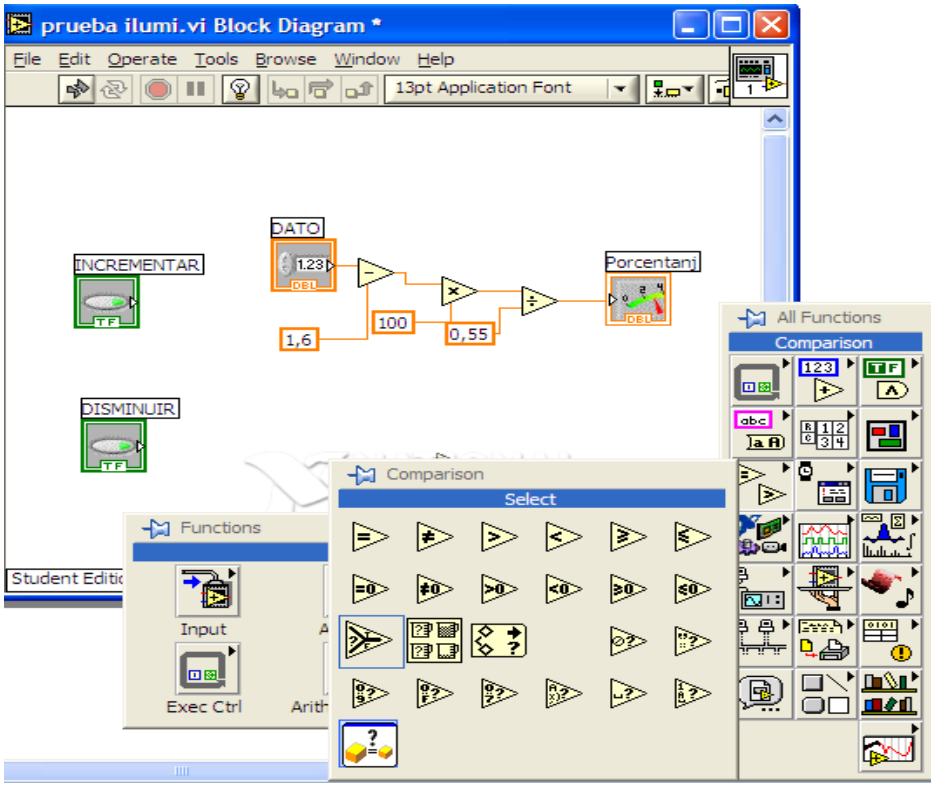


Figura I31 Insertar funciones de comparación

- Insertare el Timer
- Clic en la barra de funciones elegir Timer & Dialog
- Seleccionar Wait(ms) Ver Figura I32

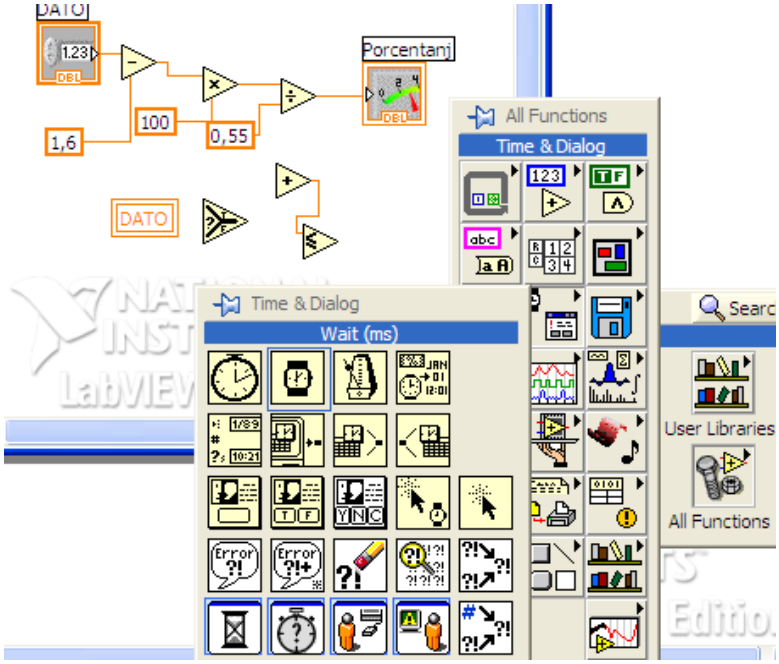


Figura I32 Insertar una función de espera Wait (ms)

Para poder saber si el botón de incrementar ha sido presionado necesitamos un estructura de comparación Case en la parte verdadera se va a proceder a programar el flujo de información con los elementos anteriormente seleccionados

16. En la barra de funciones
17. Elegir All Funtions
18. Seleccionar Structures Ver Figura I33

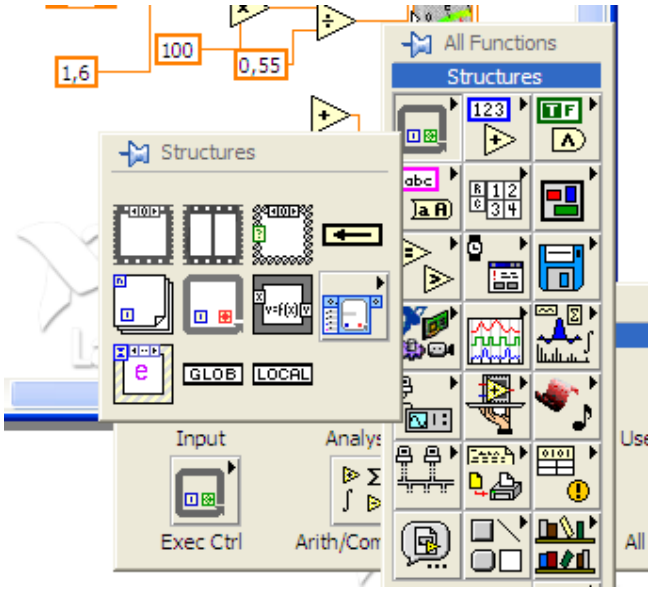


Figura I33 Insertar Una estructura de comparación Case

Siguiendo el flujo de datos nos quedara de la siguiente manera una ves programado el Botón de incremento Figura I34

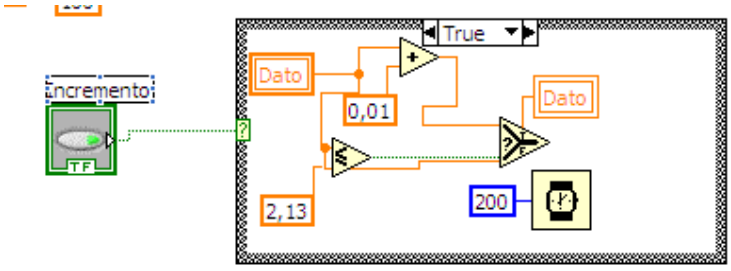


Figura I34 programación del botón incremento

En el botón disminuir realizar los pasos que se realizo para el incremento, con los siguientes cambios que en lugar de sumar tenemos que restar así como el limite de comparación será el menor como también el signo de mayor que Figura I35

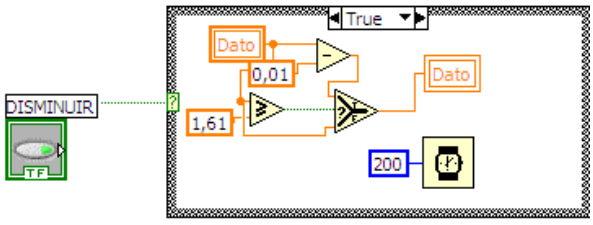


Figura I35 Programación del botón disminuir

Falta programar la parte automática, vamos a colocar un botón de selección para poder elegir sea manual o automática la cual va a estar dentro de un ciclo de selección Case en la parte verdadera se programara la parte automático en la parte falsa lo anteriormente desarrollado. Donde se tiene que incluir el asistente de tarjeta de datos para poder recibir las señales digitales que servirán de ordenes para el encendido

- En el panel de control insertar un OK Botton Ver Figura I36

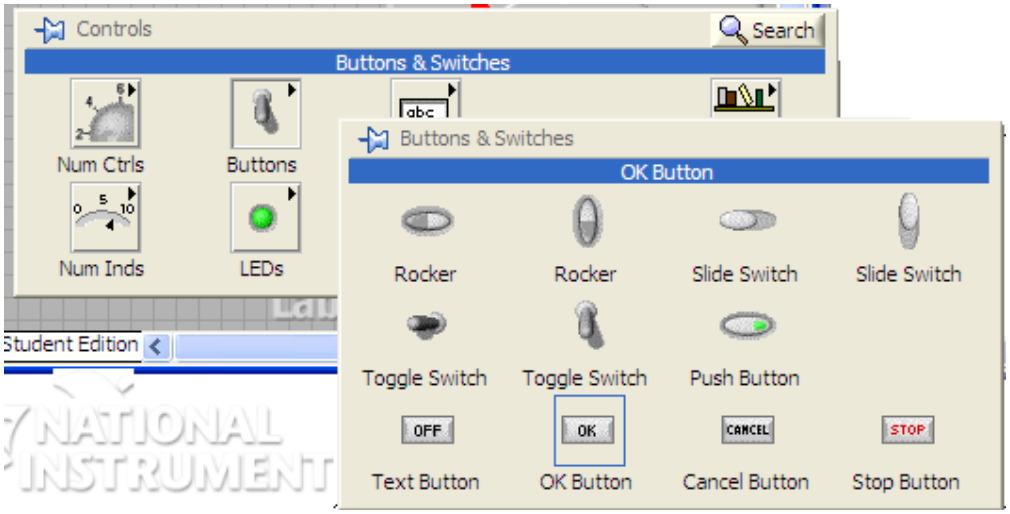


Figura I36 insertar un botón para controlar el estado manual o automático

- Proceder a modificar el label del botón así como los indicadores
- Clic sobre el botón
- Seleccionar Properties
- En el Label escribir Modo
- Seleccionar Multiple Strings Ver Figura I37
    - On text escribir Automático
    - Off Text escribir Manual

Estos pasos permiten una mejor interfaz con el usuario ya que puede ver el estado del botón, una vez que se este ejecutando el proyecto

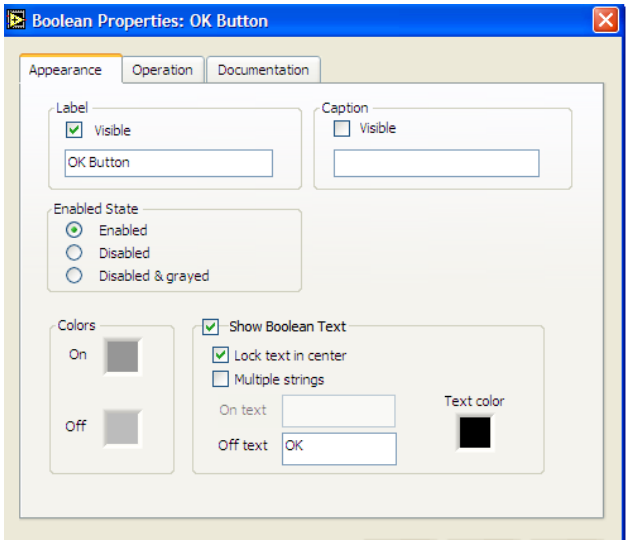


Figura I37 Ventana de propiedades del Push Button

En el diagrama de bloque en la paleta de funciones:

- Elegir Input
- Seleccionar DAQ Assist que nos ayudara a configurar el canal de entrada del dato sea esta digital o analógica Ver Figura I38

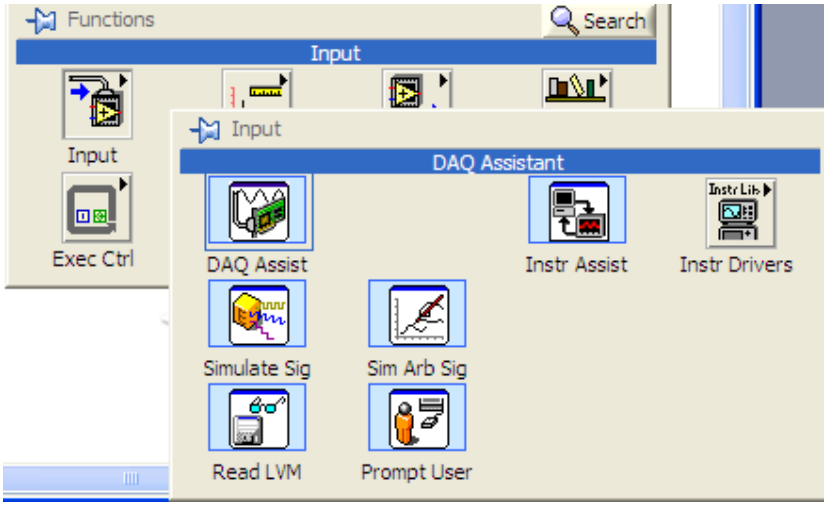


Figura I38 Ventana de lãs funciones de entrada

Se elegir el tipo de entrada dependiendo de la aplicación a desarrollar

- Selecionar Digital I/O Ver Figura 39

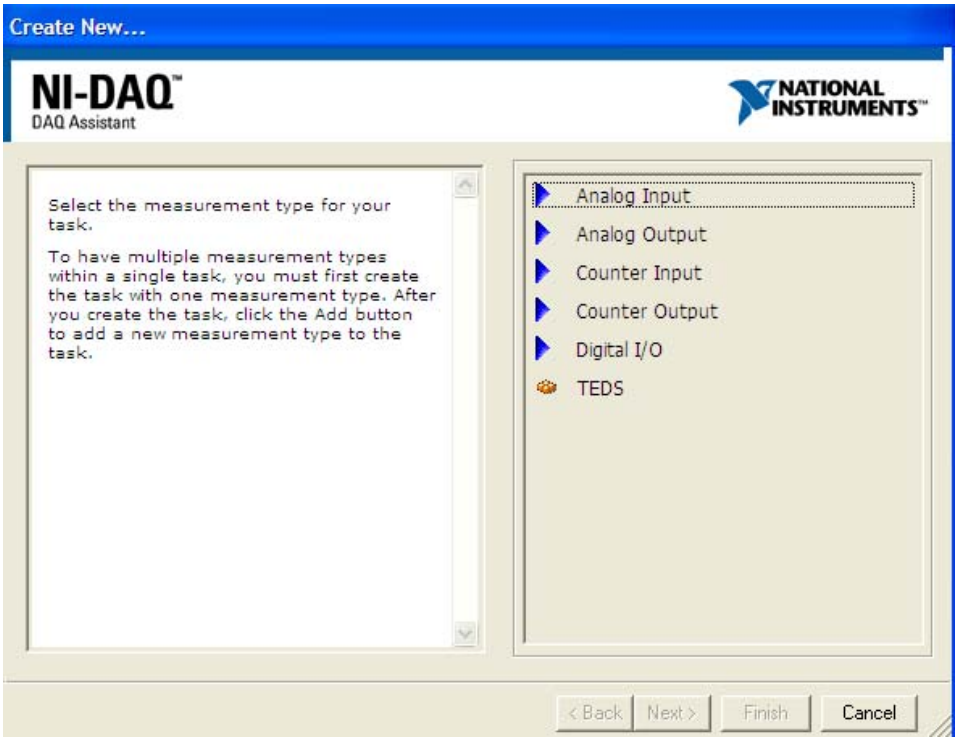


Figura 39 Ventana de selección de entrada

- Elegir Line Input Figura I40

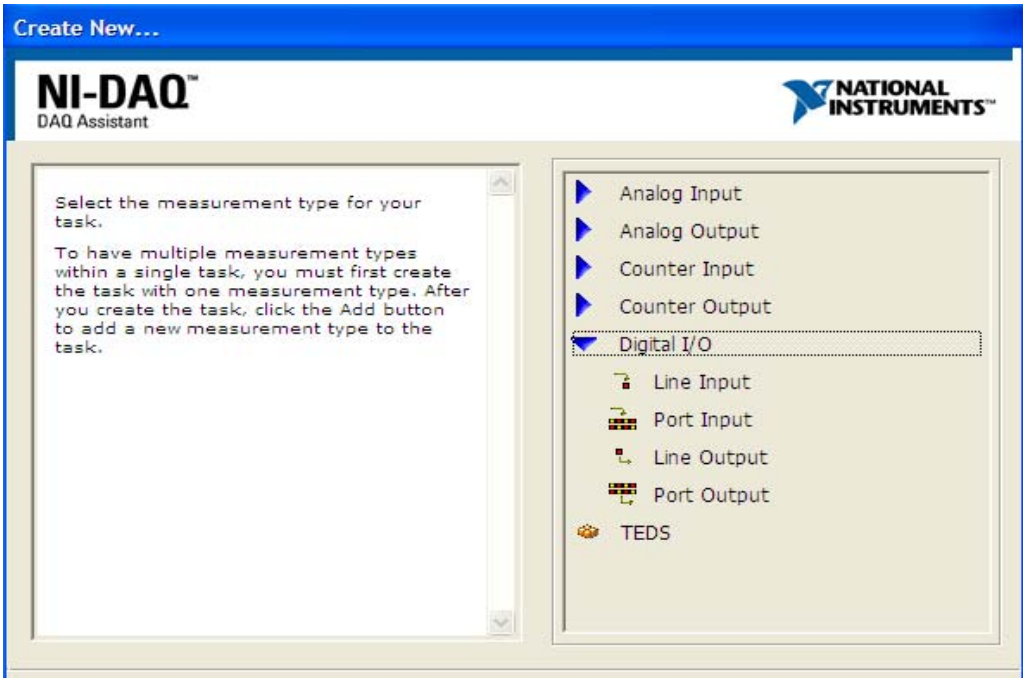


Figura I40 ventana de configuración de los puertos de entada

- Seleccionar port0/line0 Ver Figura I41

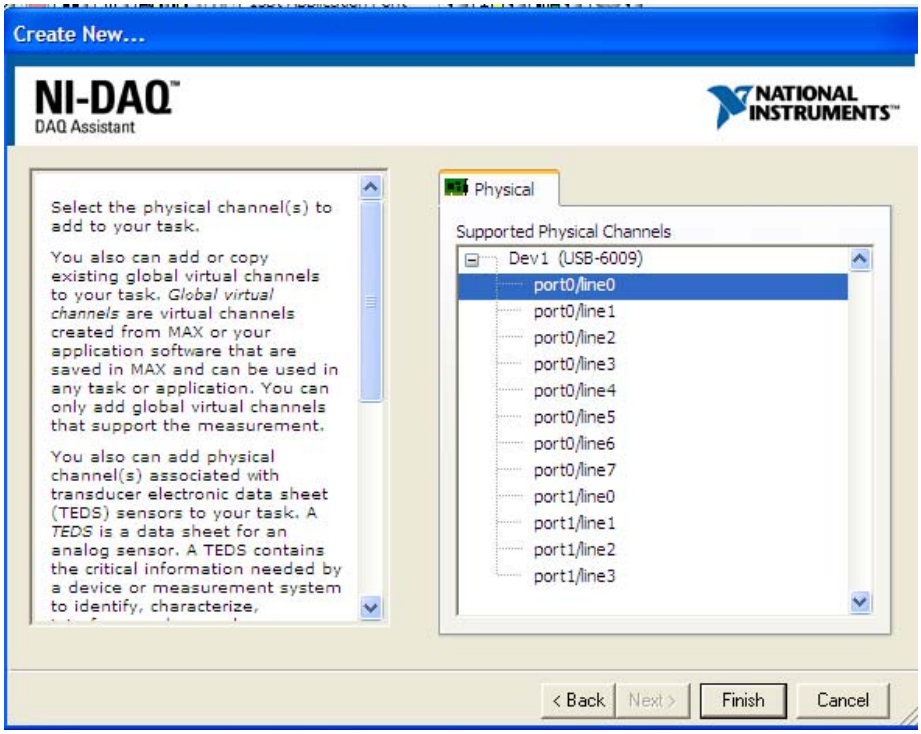


Figura I41 ventana de configuración del puerto a ser utilizado

Se mostrara la siguiente ventana donde se puede hacer una test de funcionamiento y además si se elige el icono  se puede ver el canal utilizado para dicha práctica ver Figura I42

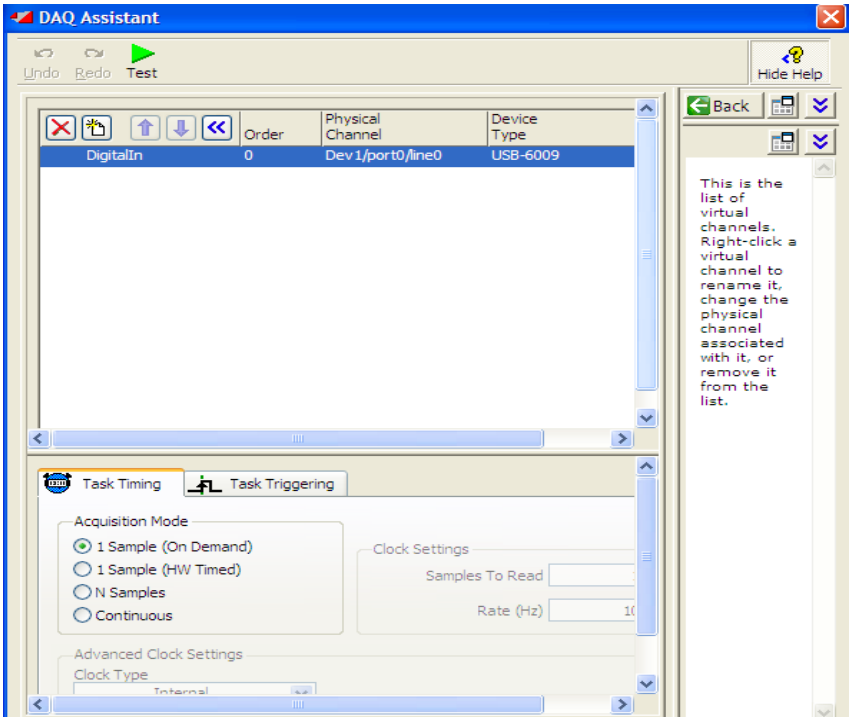


Figura I42 ventana de configuración de la señal a utilizar

Crear un array para almacenar los datos que ingresen por nuestro adquisidor y asi poder comparar ya que el adquisidor de datos trabaja con 8 bits y solo se necesita uno de ellos

- En la barra de funciones
- Elegir Array
- Seleccionar Index Array Ver Figura I43

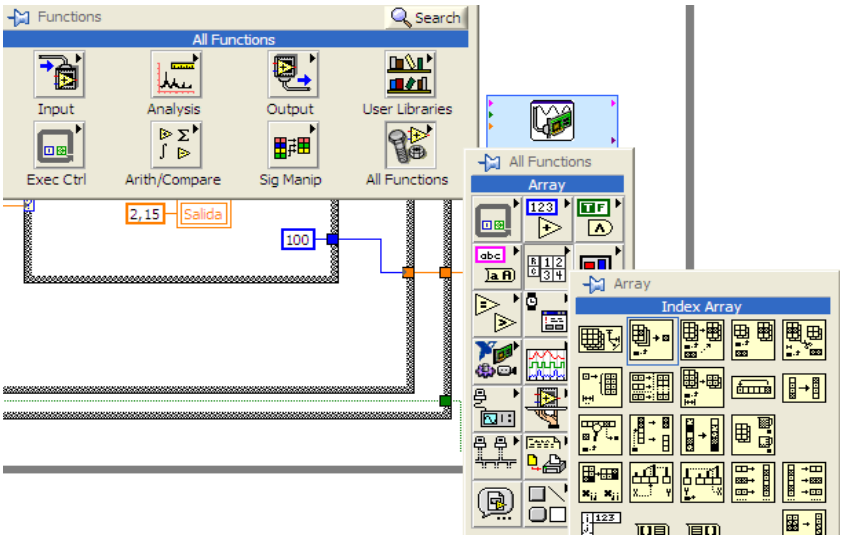


Figura I43 Seleccionar un arreglo de la barra de funciones

- Seleccionar en la barra de funciones en el botón Strcture
- Seleccionar Case Structure y colocar en el diagrama de bloque Ver Figura I44

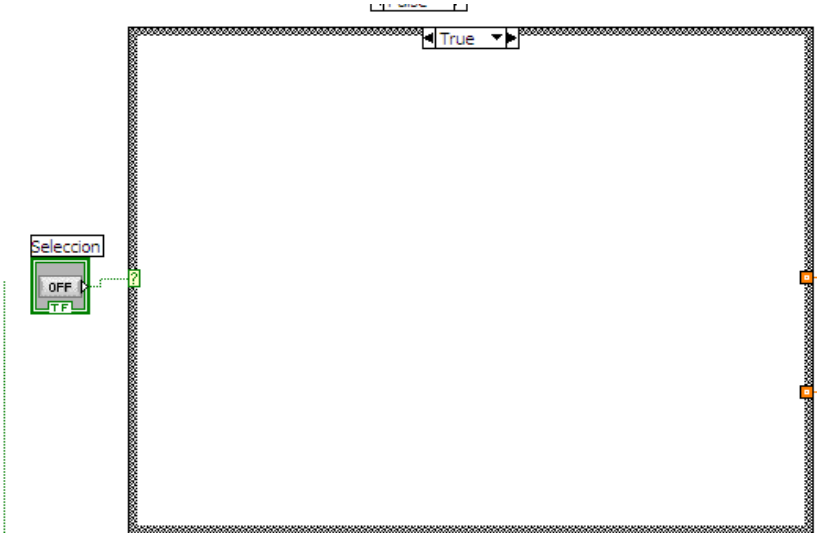


Figura I44 diagrama de bloque insertar un Case Structure

- Crear una variable de control; la cual controlara los tres estados del encendido clic en la barra de control del panel frontal elegir
- Nun Ind
- Seleccionar Num Ind Figura I 45

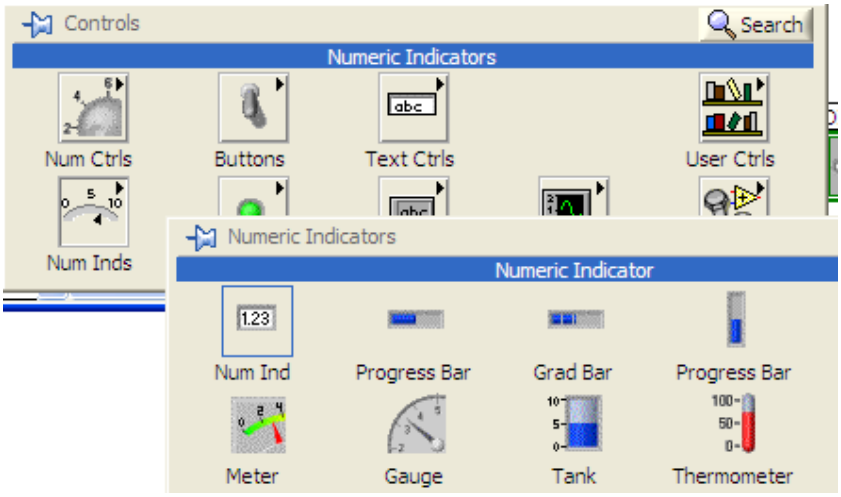


Figura I45 Insertar una variable de control

A este control crear una variable local para poder aumentar

- Clic sobre el indicador
- Create Local Variable Figura I 46

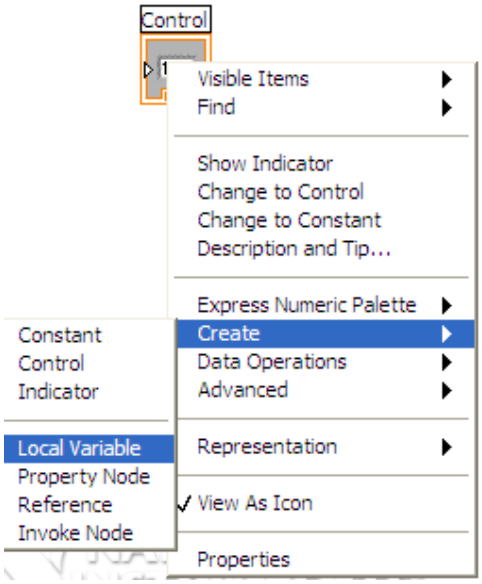


Figura I 46 Ventana de crear una varible de local

Como se puede ver la variable local es una variable de entrada; para cambiar su orden clic derecho seleccionar Change to Read Ver Figura I47

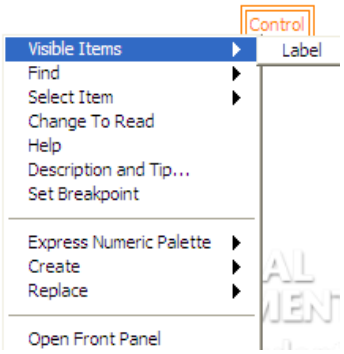


Figura I47 el orden de una variable

En el caso verdadero procedemos a unir los elementos como a continuación se muestra Ver Figura I48

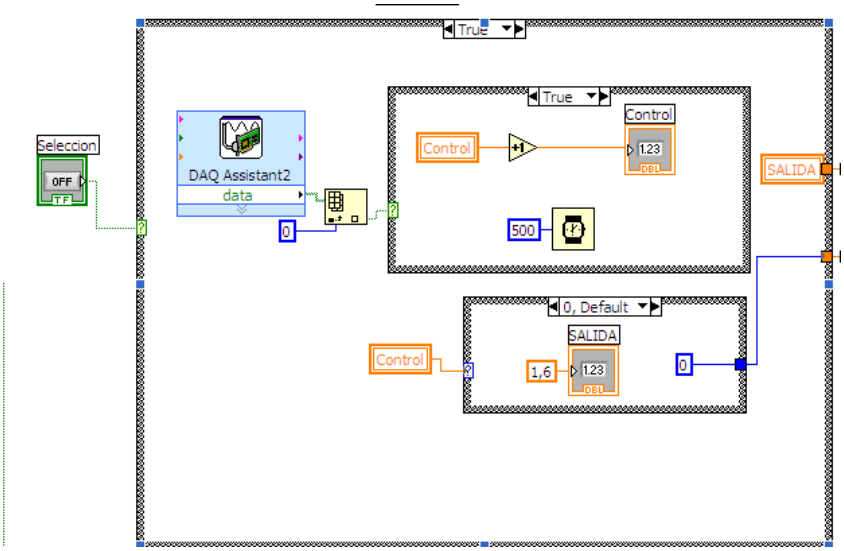


Figura I48 programación de la parte automática del programa

En la parte falsa copiar lo que anteriormente se realizo para la parte manual. Presionar la tecla ctrl.+clic derecho el Mouse, proceder a seleccionar las partes a ser copiado

Ver Figura I49

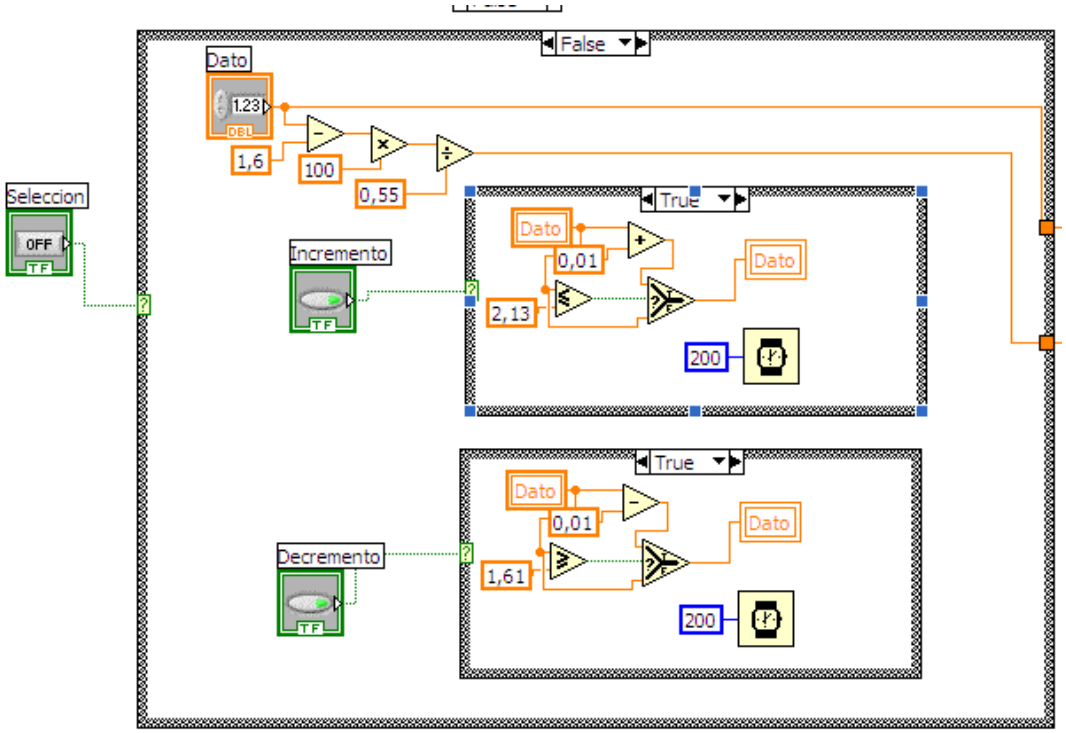


Figura I49 programación de la parte manual

Para ejecutar la aplicación se tiene que dar clic en la barra de herramientas del programa LabVIEW Ver Figuira I50



Figura I50 botones de ejecución del programa

Una vez terminado la programación, se tiene que arreglar el panel frontal para que el usuario solo pueda ver lo necesario; así como modificar el panel de control con colores. Este paso lo podemos realizar con la ayuda de Tools Palette

Finalmente el proyecto quedara de la siguiente forma Ver Figura I51

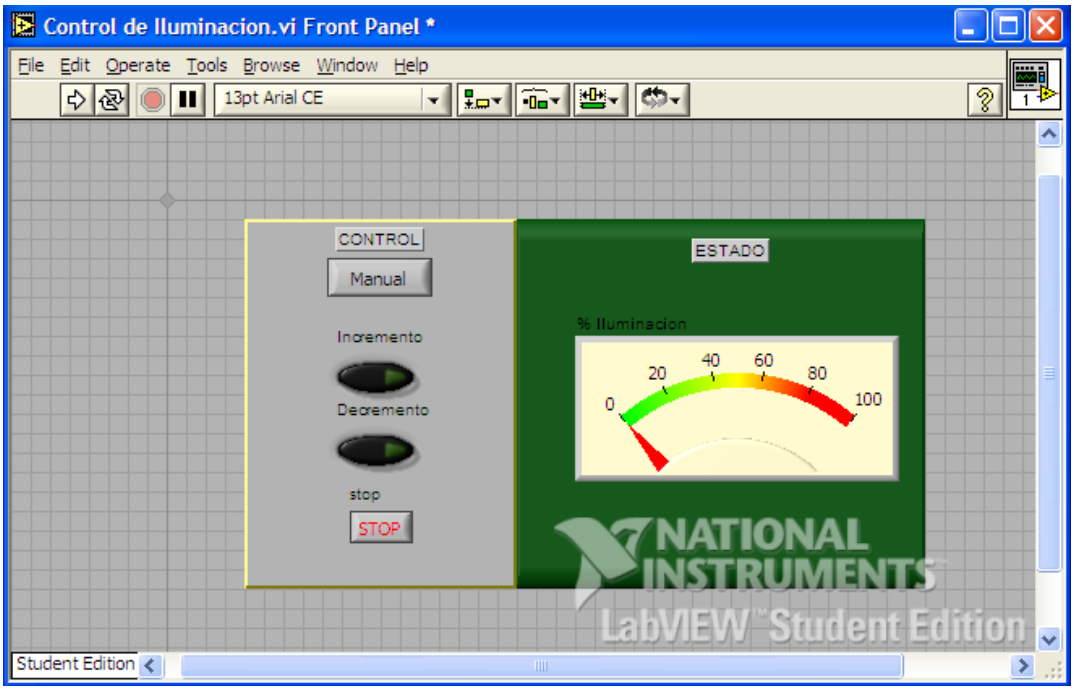


Figura I 51 Panel Frontal programa terminado

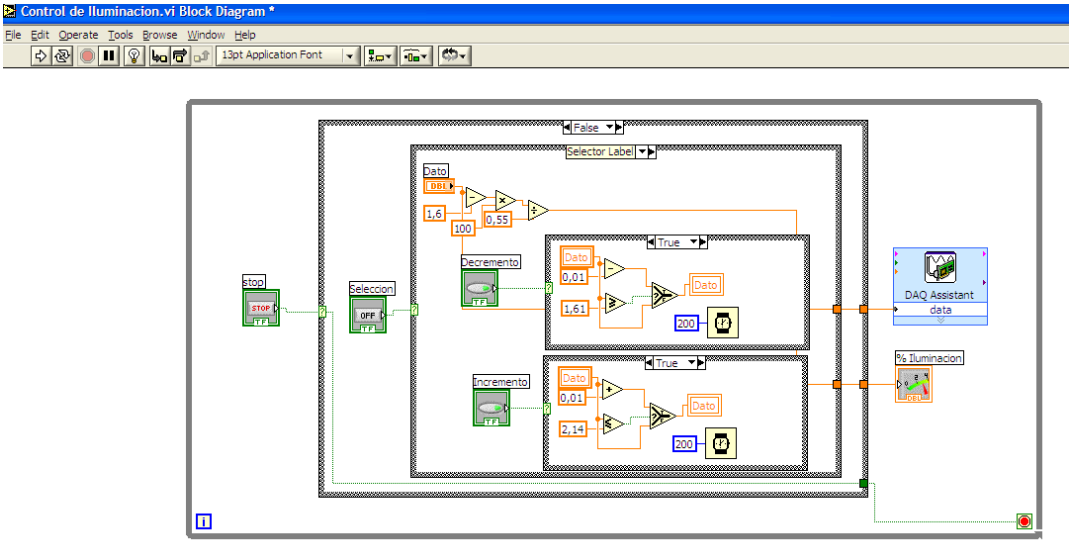


Figura I52 Diagrama de Bloque programa terminado

Ver archivo multimedia Control Intensidad.avi para mayor ayuda

ANEXO 2

Práctica No2

Tema: Control de temperatura

Objetivo:

Realizar un proyecto en LabView que interactué con el modulo USB 6009 y un circuito electrónico que sirva de entrada analógica y salida digital para poder así encender un ventilador.

Descripción:

Se manipulara el control de encendido y apagado de un ventilado; por el puerto de salida digital del modulo USB6009, una vez que se haya censado la temperatura a la que se encuentra por medio de un termómetro digital (LM35); usando una entrada analógica del mismo

Vamos a utilizar un canal digital y además un canal analógico del modulo USB

Entrada analógica puerto P0

Salida digital puerto p18

Se necesitaran los siguientes materiales para realizar la placa electrónica la cual además de poseer el termómetro digital y para aprovechar al máximo del amplificador operacional incluiremos la fotoresistencia que servirá como el contador en la práctica siguiente

Materiales:

Regulador 7805
Regulador 7812
Diodos 1N4007
Condensadores
0,1uF
LM358
LM35
Condensador 22uF
Resistencia 10K
Fotoresistencia
Condensador 10uF
Potenciometro 10K
Borneras 1x2
Espadines 1x3
Resistencias 4,7K

Transistores 2N3904

Ventilador 12V

Motor 12V

Base Dip8

Baquelita

Paso 1

Realizar la placa electrónico ver diagrama de conexión figura T1. Se debe colocar los instrumentos electrónicos según el diagrama; además tener en cuenta los pines del termómetro. Para mayor información de las hojas técnicas de los instrumentos electrónicos ver anexo7

- Primero referencia
- Segundo señal
- Tercero alimentación.

Conectar al Modulo Adquisidor de Datos la placa electrónica esta tiene que estar conectada a la fuente así como el ventilador y los pines de los puertos respectivos, para guía ver en la placa los números que corresponden:

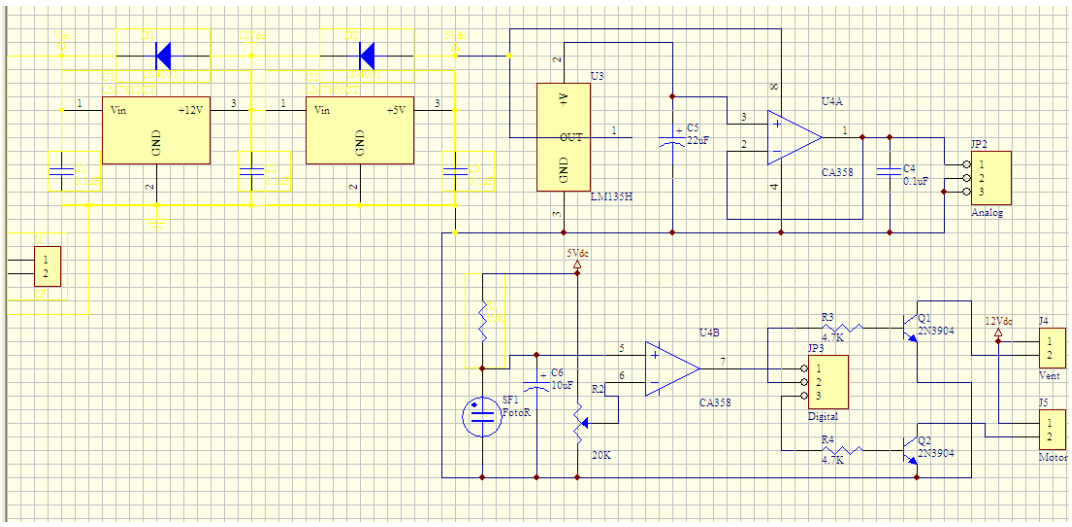


Figura T1 diagrama de conexión de la placa electrónica

Paso 2

Programación en LABVIEW

Crear la señal de entrada; para lo cual elegir un DAQ Assisten en el cual debe especificarse que es: de entrada, señal analógica, de voltaje, en el puerto a0. en la

configuración cambiar la entrada diferencial, el rango máximos 5V y mínimo 0V de voltaje Figura T2

- En LabView crear un VI en Blanco
- En el diagrama de bloque
- Barra de Funciones seleccionar Input
- Elegir DAQ Assistent
- Seleccionar el tipo de canal
    - Analog Input
- Seleccionar el tipo de medida que se va a utilizar en nuestra práctica
    - Voltage
- Añadir el canal físico para nuestra tarea
    - ai0
- configurar el canal de la tarea para lo cual cambiar lo siguiente en la ventana de la figura T2:
- Input Range
    - Max 5
    - Min 0
- Terminal Configuration
    - RSE

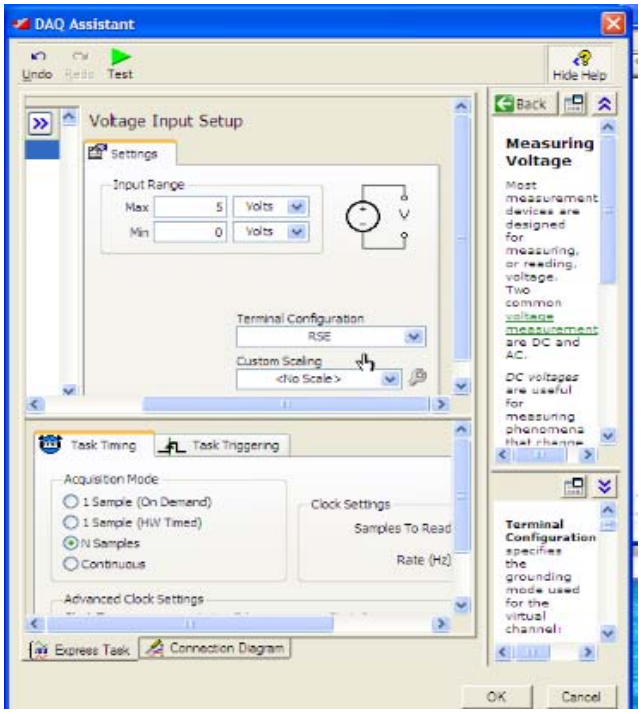


Figura T2 Configuración del canal de entrada análogo

La señal que se va a recibir son los datos de entrada y vamos visualizarlo en un termómetro; a los datos recibidos multiplicar por 100; debido a que el termómetro envía 10 mV por cada grado centígrado Figura T3

- En el diagrama de bloque insertar la función de multiplicación
- Crear una constante de 100
- En el panel frontal insertar un indicador de termómetro

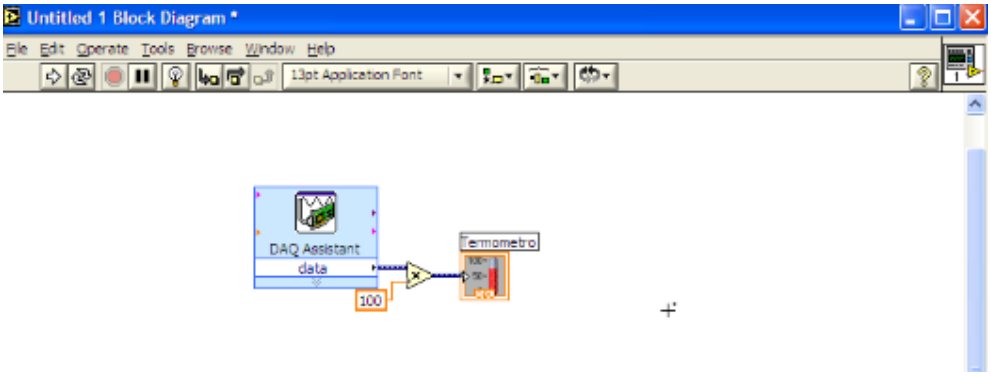


Figura T3 Programación para visualizar la temperatura obtenida por el sensor

Insertar en el panel frontal 2 Led que serán los indicadores para visualizar si se encuentra prendido o apagado el ventilador.

En la barra de controles seleccionar:

LEDs

Round Led

Una vez obtenida la señal de entrada se a compararla con el límite permitido para así encender el ventilador para esto realizar los siguientes pasos:

En el Panel Frontal insertar un indicador numérico; el mismo que se llamara control y es para manipular el limite permitido.

Seleccionar la función de comparación Mayor que 

Seleccionar la Strcture Case

En la parte verdadera se procede a encender el ventilador

Insertar un DAQ Assistent

En el canal de la tarea de salida seleccionar:

Digital I/O

En el canal a utilizar

Line Output

En puerto a utilizar en la tarea

Port0/line0

El indicador de encendido crear una variable la que debe estar en Verdadero

El indicador de apagado crear una variable la que debe estar en Falso

En la parte falsa se va a reinicializar las variables así como el puerto, para lo que el asistente tiene que ocupar el mismo puerto

Insertar un DAQ Assistent

- Digital I/O
- Line Output
- Port0/line0
- A los indicadores de encendido y apagado crear variables locales y trasladarlas a la parte falsa y programarlos de siguiente forma:

  El indicador de encendido crear una variable la que debe estar en Falsa

  El indicador de apagado crear una variable la que debe estar en Verdadera

Se puede ver el diagrama de bloque del proyecto; programada la parte verdadera y falsa Figuras T4-T5

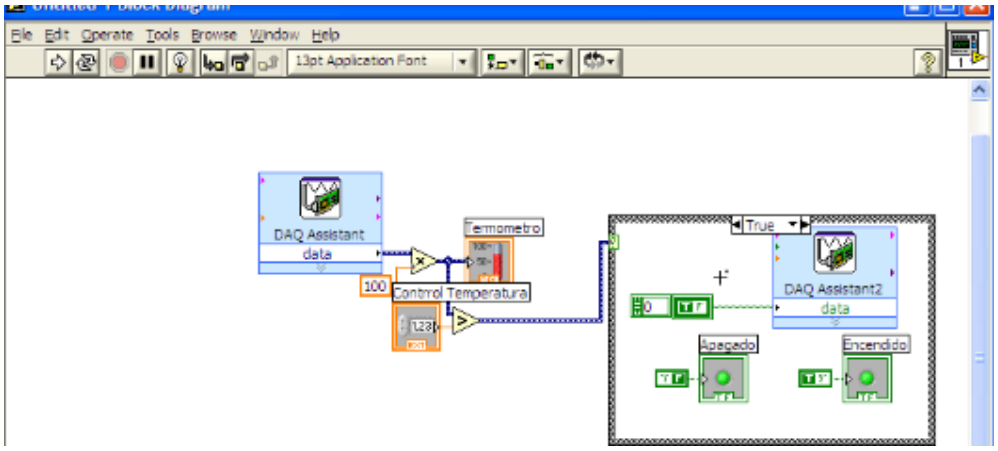


Figura T4 Programación del proyecto en caso de que la temperatura sea mayo al límite permitido

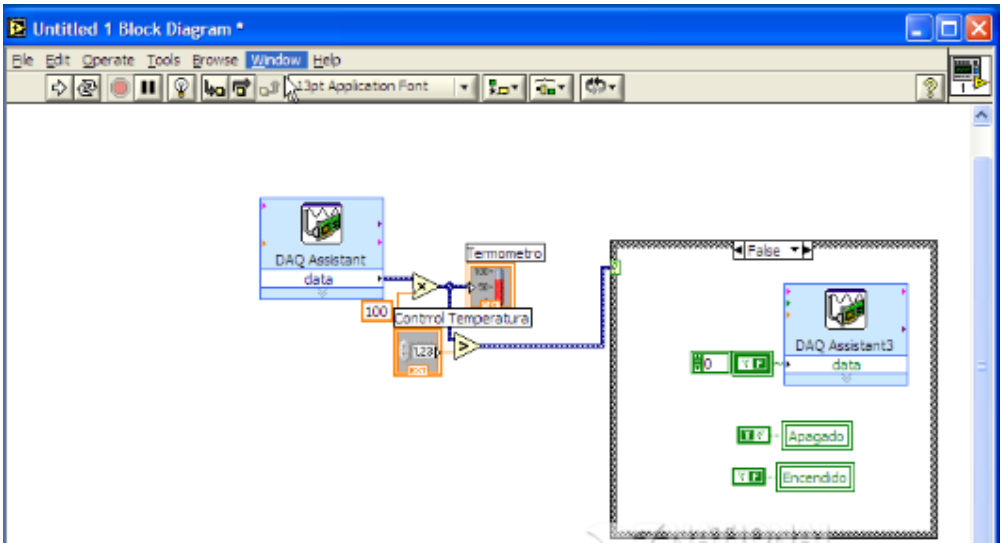


Figura T5 Programación del proyecto en caso de que la temperatura sea menor al límite permitido

Para que el programa se ejecute hasta que el usuario desee terminar se procede a incorporar un lazo While Loop así como un botón STOP que será unido al botón de parada del lazo ver figura T7

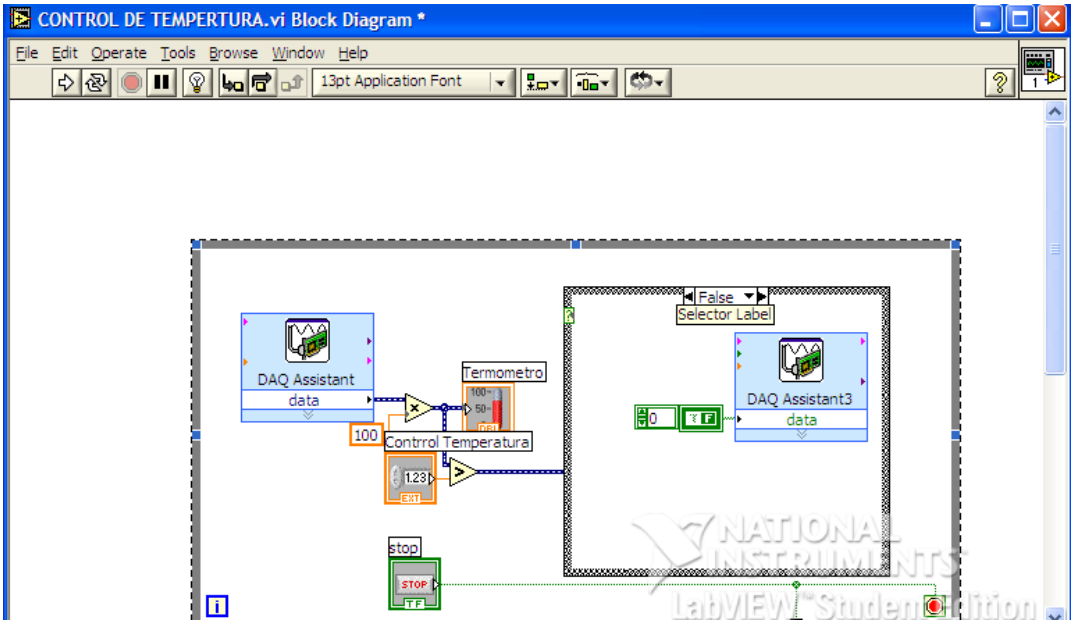


Figura T7 Lazo cíclico de ejecución

Al ejecutar el programa en este paso se puede ver la simulación de un termómetro así como el funcionamiento de los indicadores visuales Figura T8

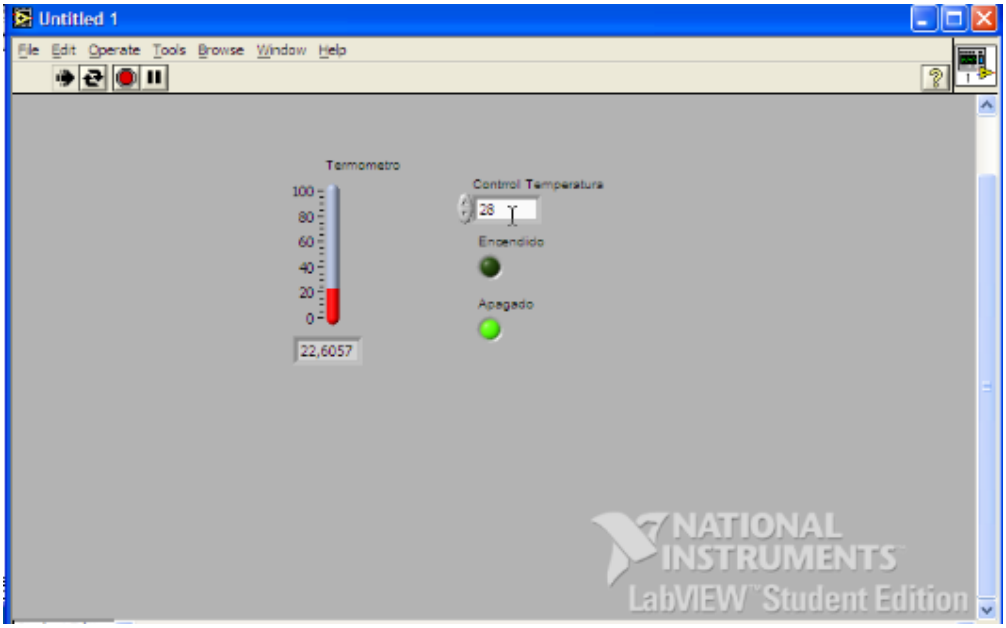


Figura T8 Panel frontal del proyecto en ejecución

Ver archivos multimedia:

- temperatura.AVI
- temperatura y contador protel.AVI

Anexo 3

Práctica No3

Tema: Contador automático.

Objetivo:

Realizar un proyecto en LabView que interactué con el modulo USB 6009 y un circuito electrónico que sirva de entrada contador y salida digital para poder así encender un motor.

Descripción:

Se manipulara el control de encendido y apagado de un motor; por el puerto de salida digital del modulo USB6009, una vez que se haya recibido el pulso por medio de la fotorresistencia ya que varia su resistencia de acuerdo a la intensidad de luz

Vamos a utilizar un canal del contador de entrada y además un canal digital de salida del modulo USB

Entrada de contador de pulso	puerto ctr0
Salida digital	puerto port0/line1

Se usara la placa electrónica del sensor de temperatura ya que en esa placa contan dor prácticas a la vez ya que utilizamos un amplificador operacional

Paso 1

Realizar la placa electrónico ver diagrama de conexión figura C1. Se debe colocar los instrumentos electrónicos según el diagrama. Para mayor información de las hojas técnicas de los instrumentos electrónicos ver anexo7

Conectar al Modulo Adquisidor de Datos la placa electrónica esta tiene que estar conectada a la fuente así como el motor y los pines de los puertos respectivos, para guía ver en la placa los números que corresponden:

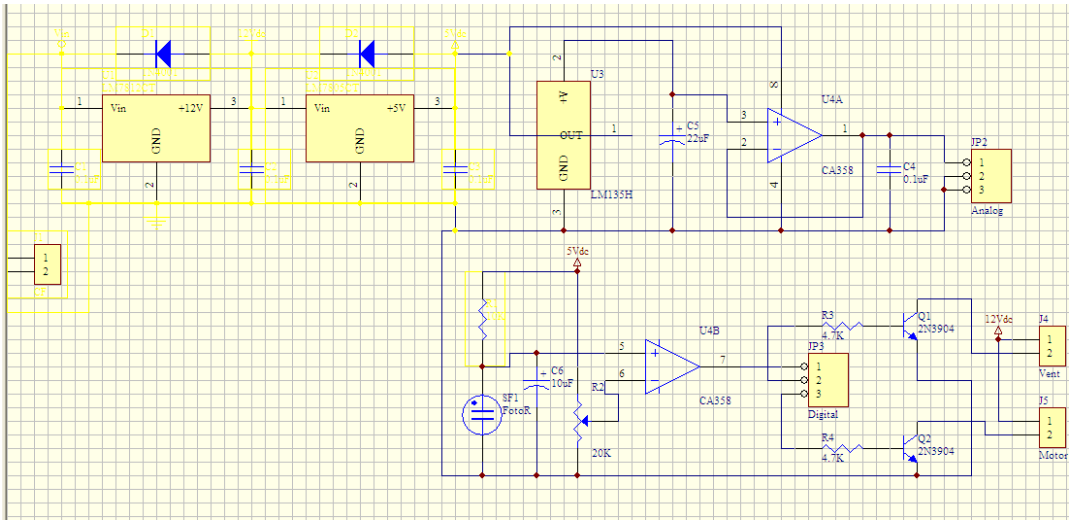


Figura C1 diagrama de conexión de la placa electrónica

Paso 2

Programación en LABVIEW

Crear la señal de entrada; para lo cual elegir un DAQ Assisten en el cual debe especificarse que es: de entrada, contador de pulso, en el puerto ctr0. Ver Figura C2

- En LabView crear un VI en Blanco
- En el diagrama de bloque
- Barra de Funciones seleccionar Input
- Elegir DAQ Assistent
- Seleccionar el tipo de canal
  - Counter Input
- Seleccionar el tipo de medida que se va a utilizar en nuestra práctica
  - Edged input
- Añadir el canal físico para nuestra tarea
  - ctr0

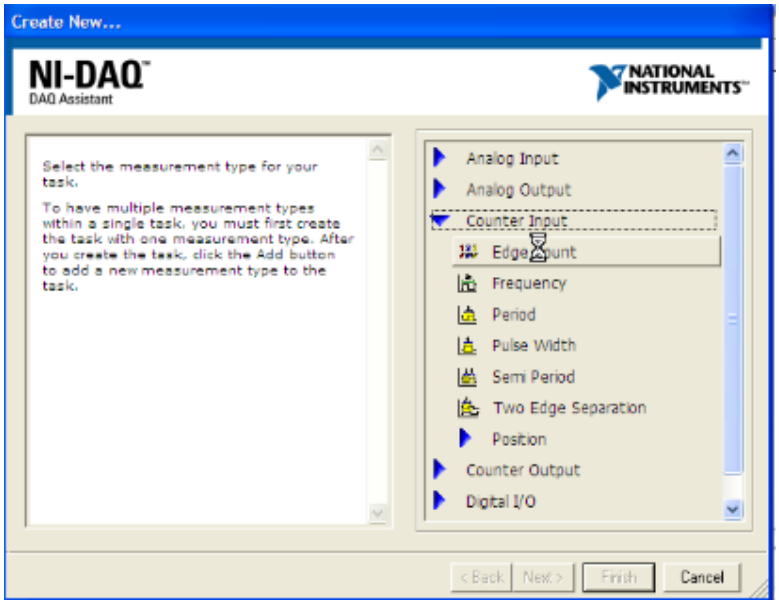


Figura T2 Configuración del canal de entrada contador de pulso

La señal que se va a recibir son los datos de entrada y vamos a crear un indicador numérico; que vienen hacer el número de botellas que pasan por nuestro sensor los Figura T3

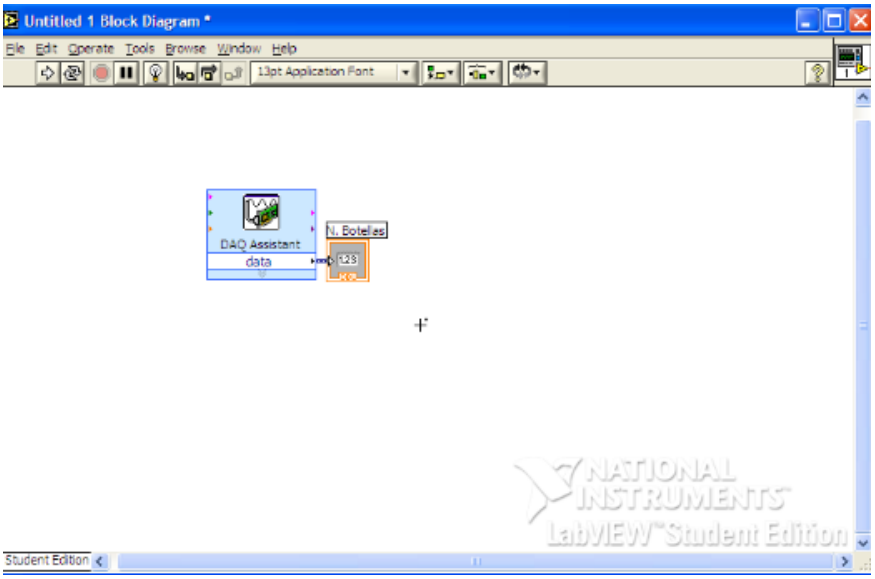


Figura C3 Programación para visualizar el número de botellas que pasan por el sensor

Del indicador numérico crear una variable local; así como insertar de la barra de funciones en las numéricas, la función de cuocientes y residuos la que permitirá saber cuantas jabas van pasando:

- En el diagrama de bloque en la barra de funciones.
- En las funciones numéricas
- SeleccionarQuotient &Remaninder Ver Figura C4

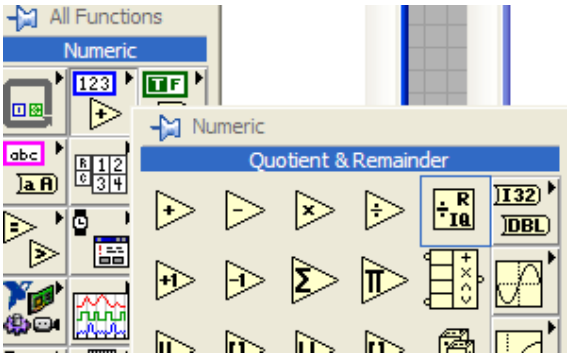


Figura C4 Función de residuos y cuocientes

Proceder a calcular el número de jabas, en la función de residuos y cuocientes en la primera Terminal unir con el número de botellas en la otra entrada la constante 12 ya que este número de botellas forman una jaba, se procede a comparar tanto el residuo como el cuociente si son igual a cero para así controlar la señal de salida del motor. Ver figura C5

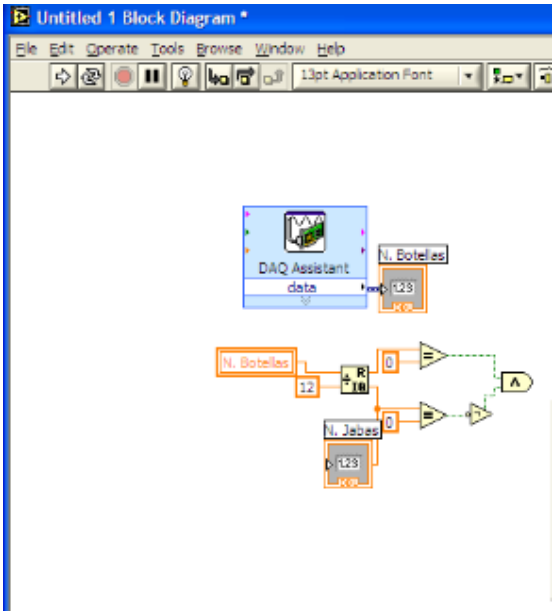


Figura C5 Programación de la cantidad de jabas que pasan por el contador

Una vez ya obtenido el número de jabas se procede a manipular el control del encendido del motor para lo cual insertamos un botón de control el que va he encender o apagar el motor:

- En el panel frontal en la barra de controles seleccionamos:
- Buttons&Switches
- Push Button
    - Crear una variable local:
- Inserta una estructura de comparación la que va a encender el botón de control
- En la barra de herramientas seleccionar Structure
- Elegir Case Strucrure

En la parte verdadera encender el botón Ver figura C6

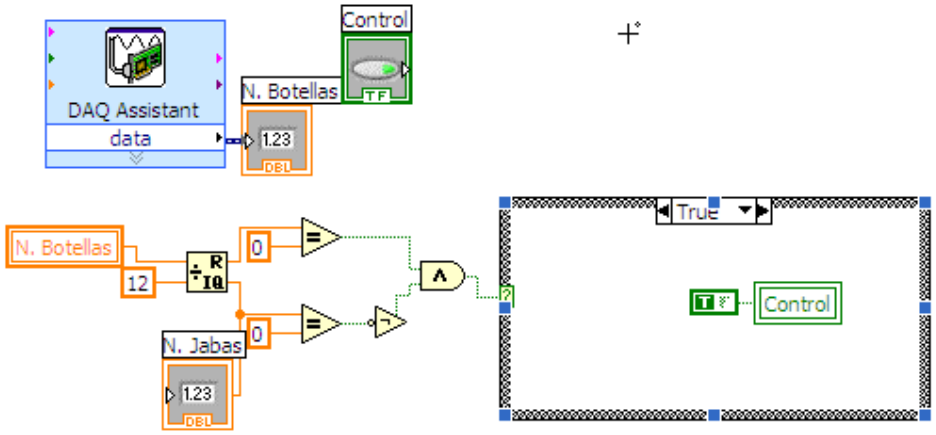


Figura C6 Activación del botón de encendido/apagado del motor

Ya que se ha programado el botón de activación se procede a :

- Insertar un DAQ Assistent que sera el que envié la señal de activa
- En el canal de la tarea de salida seleccionar:
    - Digital I/O
- En el canal a utilizar
    - Line Output
- En puerto a utilizar en la tarea
    - Port0/line1
- A al DAQ Assitent crear una constante la cual debe ser verdadera para encender la señal
- Para visualizar por unos minutos esta acción programar un tiempo de espera como se muestra en la figura con una variable retardo que se incrementara +1, Una vez completada la acción se procede a apagar DAQ Asisten y la variable de retardo tiene que volver se igual a cero

- Para visualizar en el programa que se encuentra encendido el motor en el panel frontal colocar un LED de control Ver figura C7.

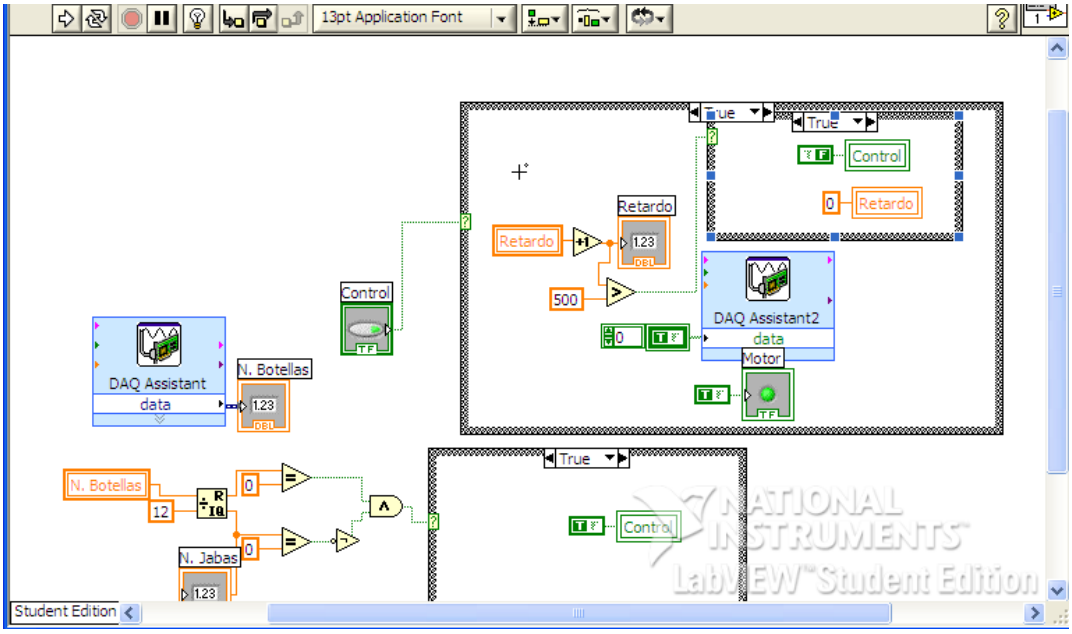


Figura C7 Programación del encendido del motor si se completo una jaba

En la parte Falsa del botón control se procede a inicializar las variables Ver Figura C8

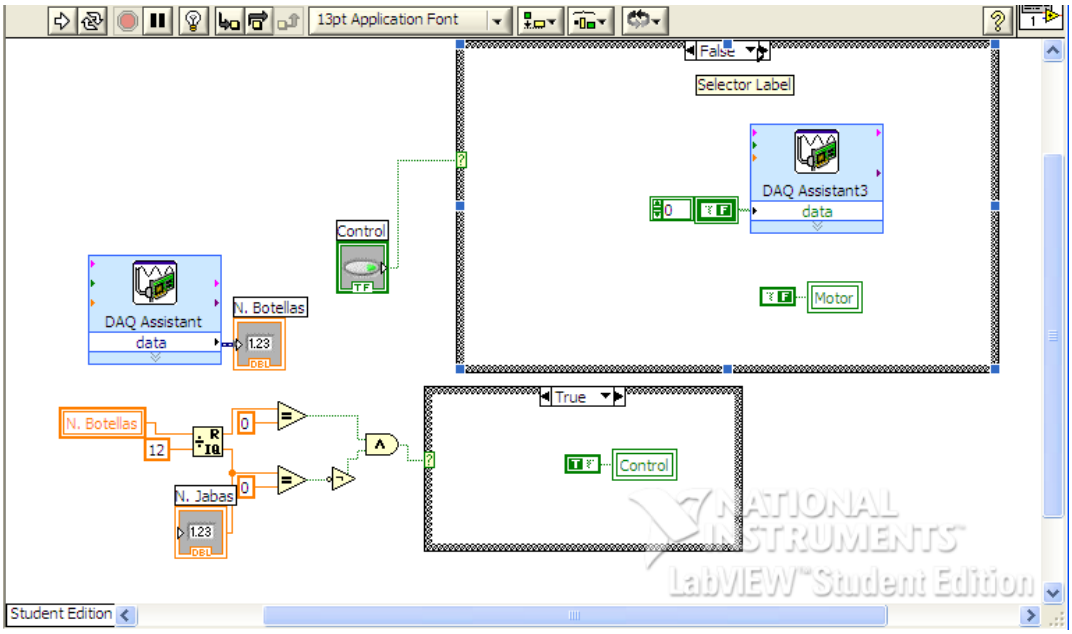


Figura C8 Inicialización de las variables en la parte falsa

Para que el programa se repita en un lazo cíclico hasta que el usuario así lo determine proceder a toda la programación anteriormente realizada encerrar en un While Loop

Por ultimo al ejecutar nuestro programa tendremos las siguientes pantallas Ver figuras C9-C10

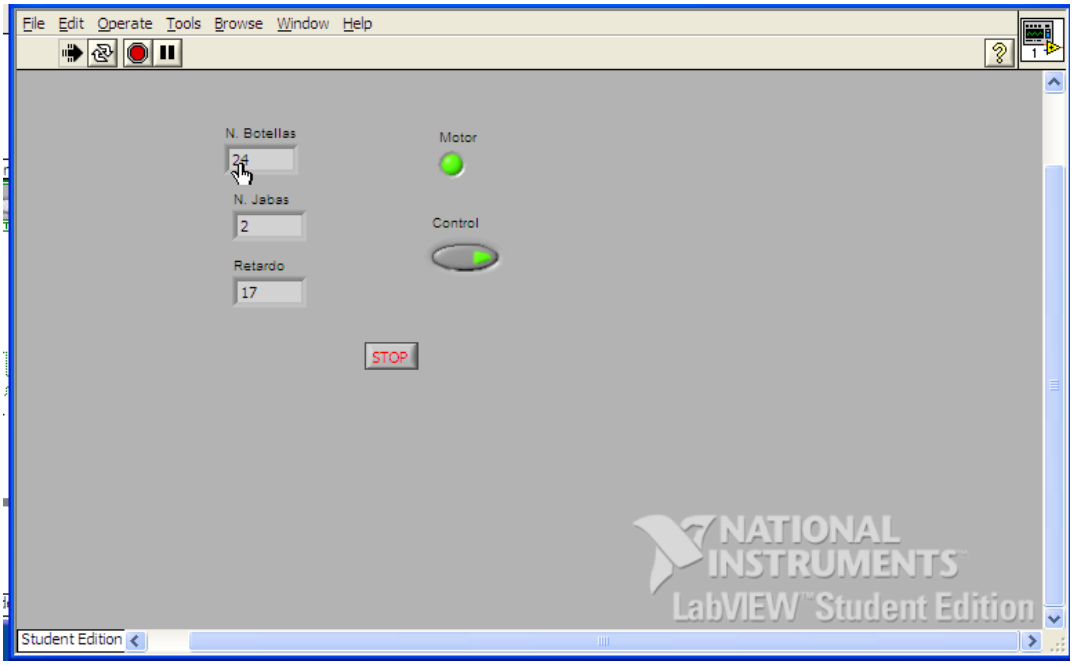


Figura C9 Panel de control ejecutándose programa contador

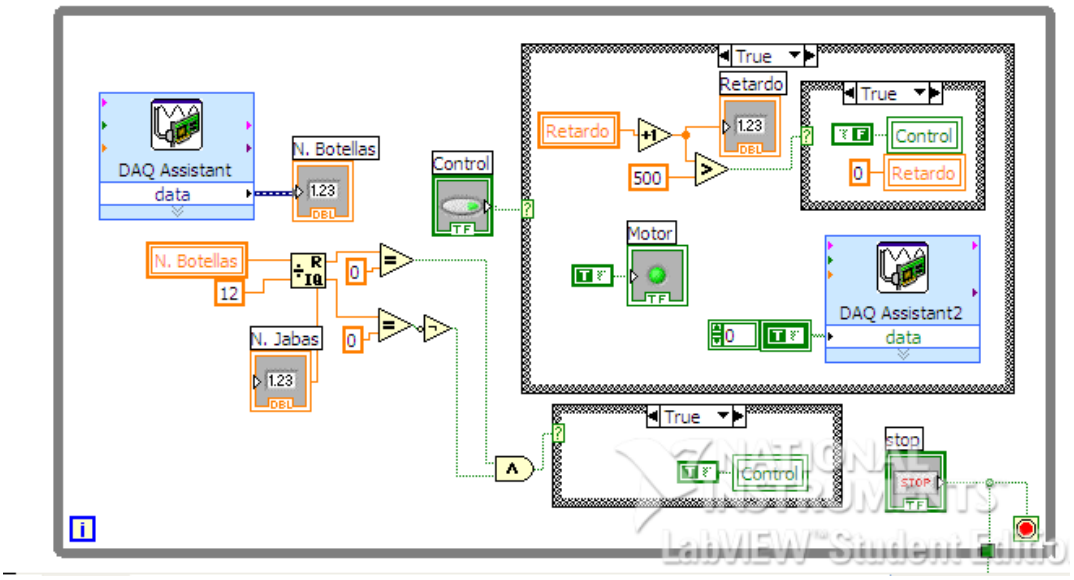


Figura C10 diagrama de bloque programa terminado Contador

Para Mayor información ver:

Contador. AVI; Temperatura y Contador.Avi

Anexo No 4

Practica No4

Tema: Controlar la frecuencia de un motor AC.

Objetivo:

Realizar un proyecto en LabView que interactué con el modulo USB 6009 y un circuito electrónico que sirva de salida analógica para poder así variar la aceleración de un motor AC.

Descripción:

En esta práctica no se va a tener datos de entrada por medio del modulo USB, la aceleración se la suministrara por medio de una perilla donde se puede aumentar o disminuir la velocidad de rotación del motor.

Salida analógica puerto a0

Se necesitaran los siguientes materiales para realizar la placa electrónica:

Materiales:

1	Regulador 7805
1	Regulador 7812
3	Diodos 1N4007
	Condensadores
4	0,1uF
2	Borneras 1x2
2	Espadines 1x3
1	Resistencias 4,7K
2	Resistencias 390
1	Transistor Tip110
2	Led de 5mm
1	Baquelita
1	Motor 5V

Paso 1

Realizar la placa electrónico ver diagrama de conexión figura M1. Se debe colocar los instrumentos electrónicos según el diagrama. Para mayor información de las hojas técnicas de los instrumentos electrónicos ver anexo7

Conectar al Modulo Adquisidor de Datos la placa electrónica esta tiene que estar conectada a la fuente así como el motor y los pines de los puertos respectivos, para guía ver en la placa los números que corresponden:

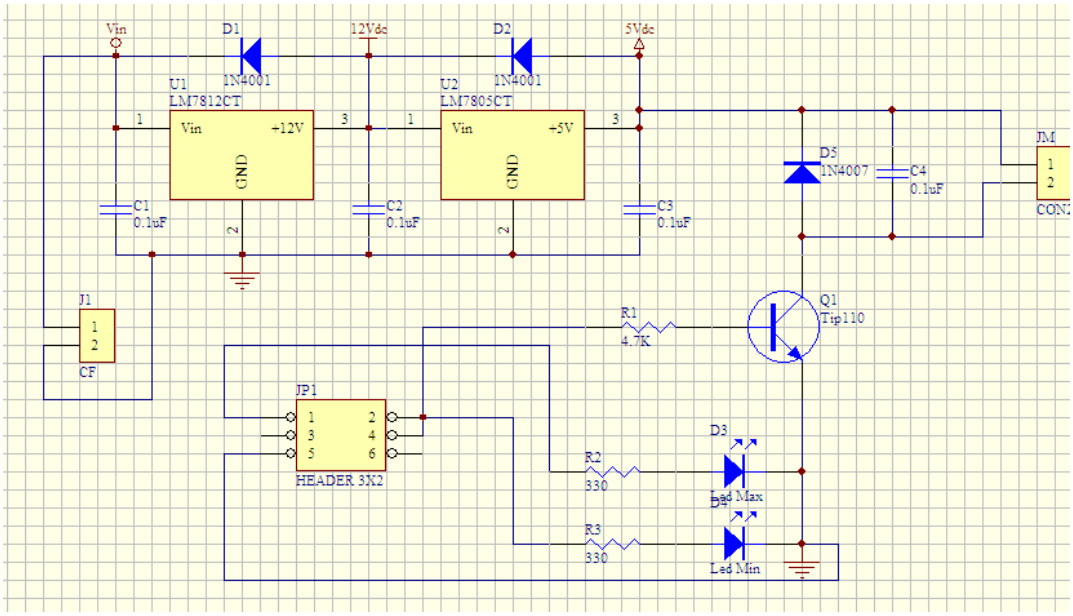


Figura M1 diagrama de conexión de la placa electrónica

Paso 2

Programación en LABVIEW

Crear la señal de entrada; en esta práctica se la va a dar por medio del panel frontal para lo cual se debe insertar un control de perilla

En la barra de controles

Seleccionar Num Inds.

Dial Ver figura N2

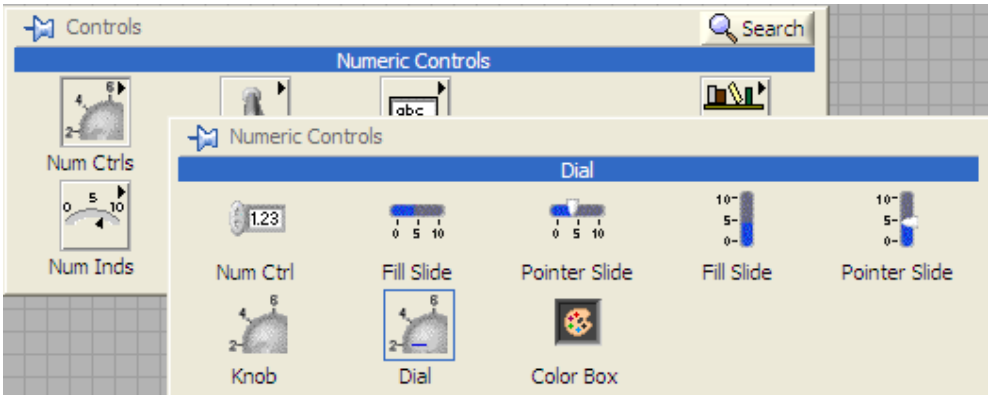


Figura M2 Ingreso control perilla

Este indicador será el encargado de manipular el porcentaje de ciclo con que el motor gira. Para calcular el tiempo en alto se va a proceder a dividir este ingreso para cien y el resultado se multiplica por diez ; a este resultado restar 10. El resultado total será el tiempo en bajo lo que se precede a guardarlo en un indicador numérico Ver figura M3

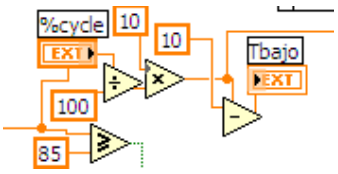


Figura M3 Programación en bajo del motor

Programar en un lazo repetitivo en donde el número de veces a repetir será el porcentaje ingresado por la perilla: en este lazo ingresar el DAQ assistent que controlara la salida del motor; también ingresar un constructor de Array que ayudara que DAQ reciba la señal en alto, incorporar un indicador visual LED para visualizar en el panel frontal

En la barra de funciones en las funciones Structure

Seleccionar For Loop

Insertar un DAQ Assistent que sera el que envié la señal de activa

En el canal de la tarea de salida seleccionar:

Digital I/O

En el canal a utilizar

Line Output

En puerto a utilizar en la tarea

Port0/line1

Crear una constante para conectar con el DAQ Assistent

Clic derecho sobre la línea de conexión elegir constructor de Array Ver Figura M4

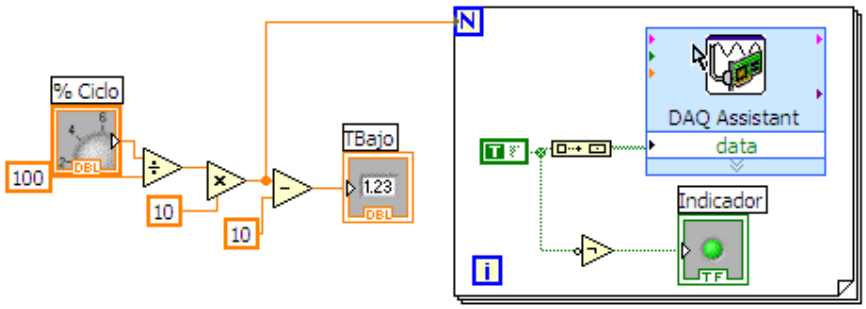


Figura M4 Programación de la salida en alto del control del motor

Insertar un indicador grafico para ver el comportamiento de la frecuencia en tiempo alto para lo cual dentro del lazo repetitivo crear una constante con el valor de 5 y conectar al indicador grafico Ver Figura M5

Clic en la barra de controles en el panel frontal

Graph Inds

Seleccionar Waveform Chart

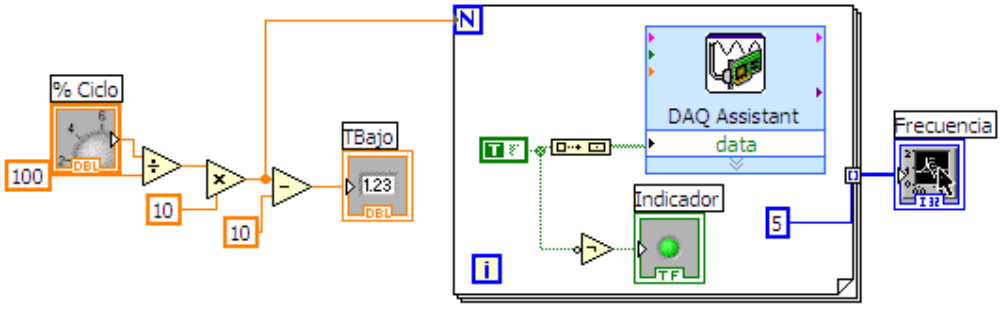


Figura M5 Programación en alto con un indicador gráfico

En el diagrama que se muestra en la figura M6 se procede a programar limites superior cuando sea mayor o igual a 85 se encuentra en el máximo de velocidad y al asistente de tarjeta de procede a en enviar una salida verdadera; en el caso falso se procede a apagar el indicador cambiando la constante de salida del asistente, recuerde que para ambos casos trabajan con el mismo canal.

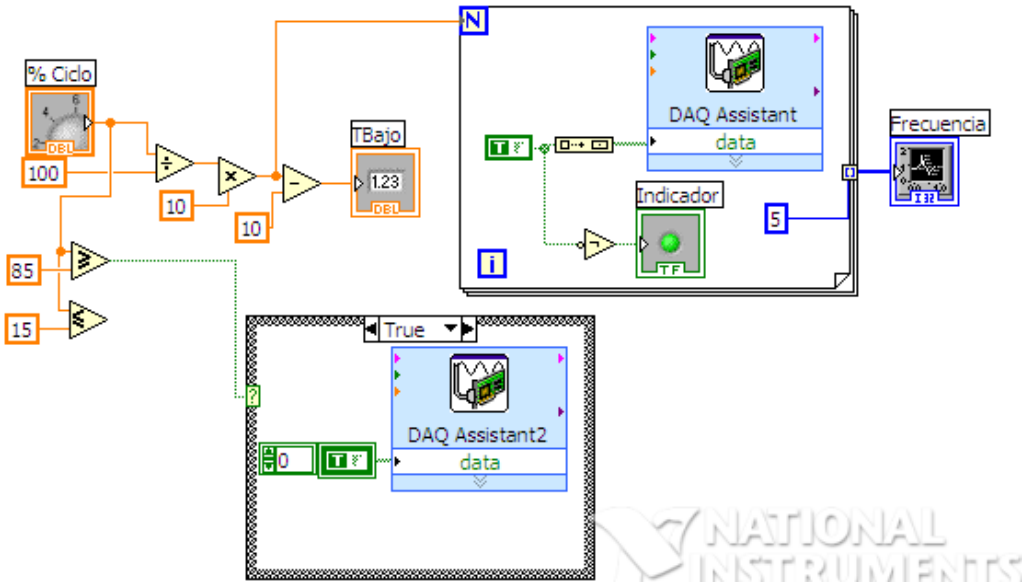


Figura M6 Programación del límite superior

Encendido o Apagado de los LED de la tarjeta electrónica:

En el diagrama que se muestra en la figura M7 se procede a programar limites inferior cuando sea mayor o igual a 15 se encuentra en el mínimo de velocidad y al asistente de tarjeta de procede a en enviar una salida verdadera además este asistente tiene que ser configurado con el canal siguiente de salida ; en el caso falso se procede a apagar el indicador cambiando la constante de salida del asistente, recuerde que para ambos casos trabajan con el mismo canal.

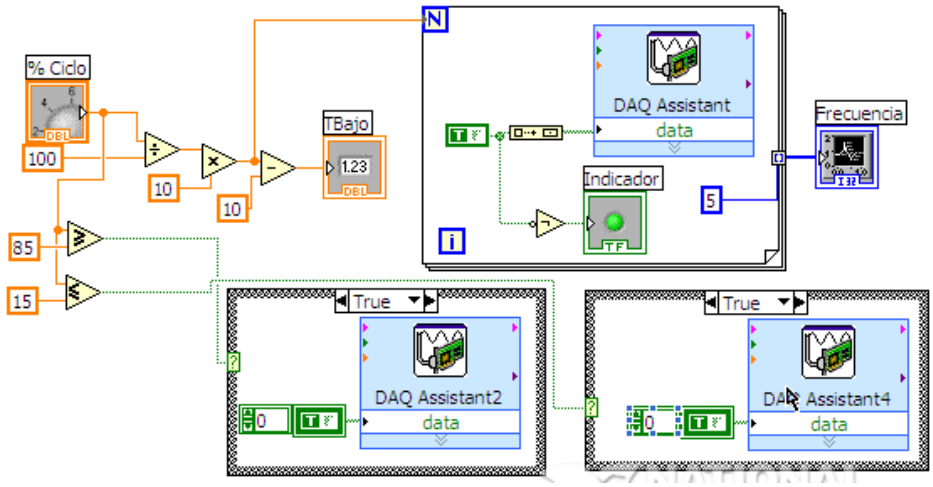


Figura M7 Limite Mínimo de velocidad

Se ha completado la programación en alto, en este paso se procede a programar el tiempo en bajo para lo cual se procede:

Crear una variable local del indicador ya que indica si se encuentra prendido o apagado, que será la variable de activación del lazo de comparación

Seleccionar un lazo de comparación Case en la parte verdadera queda lo anteriormente realizado. En la parte Falsa realizar:

Crear una variable local de Tbajo ya que este dará el número de iteraciones para el lazo repetitivo

Realizar los mismos pasos para controlar la salida del motor que se realizo anteriormente Ver Figura M8

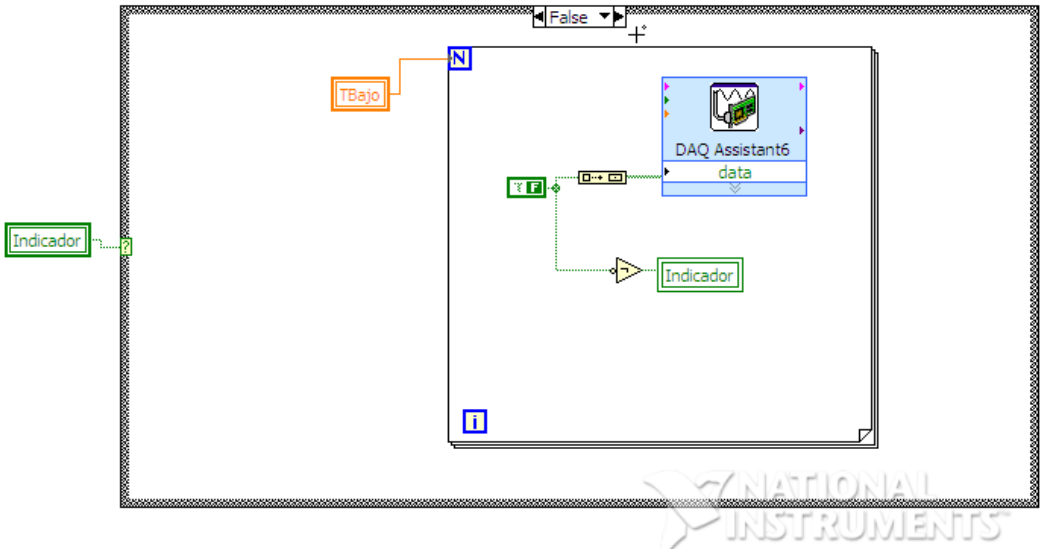


Figura M8 salida en bajo del control del motor

Para que la practica funcione en un lazo cíclico de iteraciones hasta que el usuario así lo desee se procede a toda la programación realizada a incluirla en un lazo While Loop como la creación de un botón de Stop que será el que termine la ejecución cuando así lo desee el usuario Ver figura M9

En la barra de funciones seleccionar Structure

Elegir While Loop

En el panel frontal en la barra de controles seleccionar Buttons

Stop Button

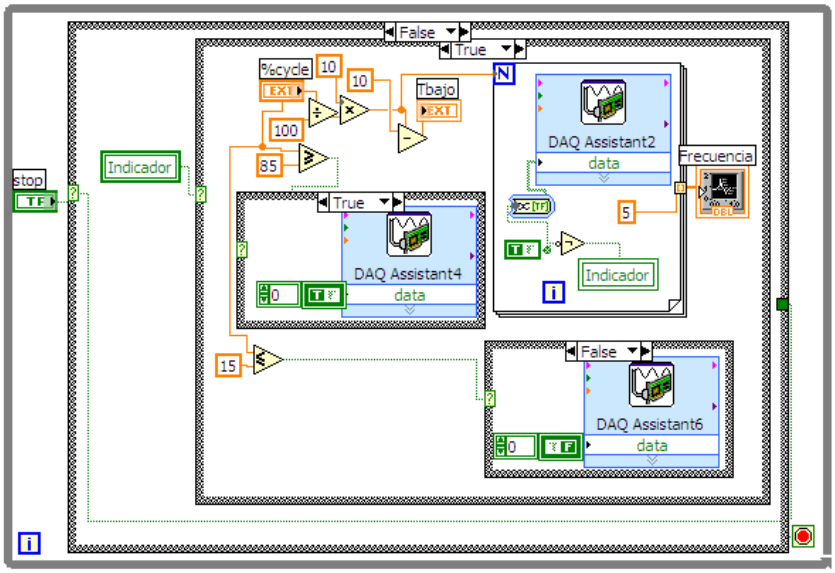


Figura M9 Diagrama de Bloque programa terminado

Una vez terminado el programa la pantalla del panel frontal nos quedara como muestra la figura M10

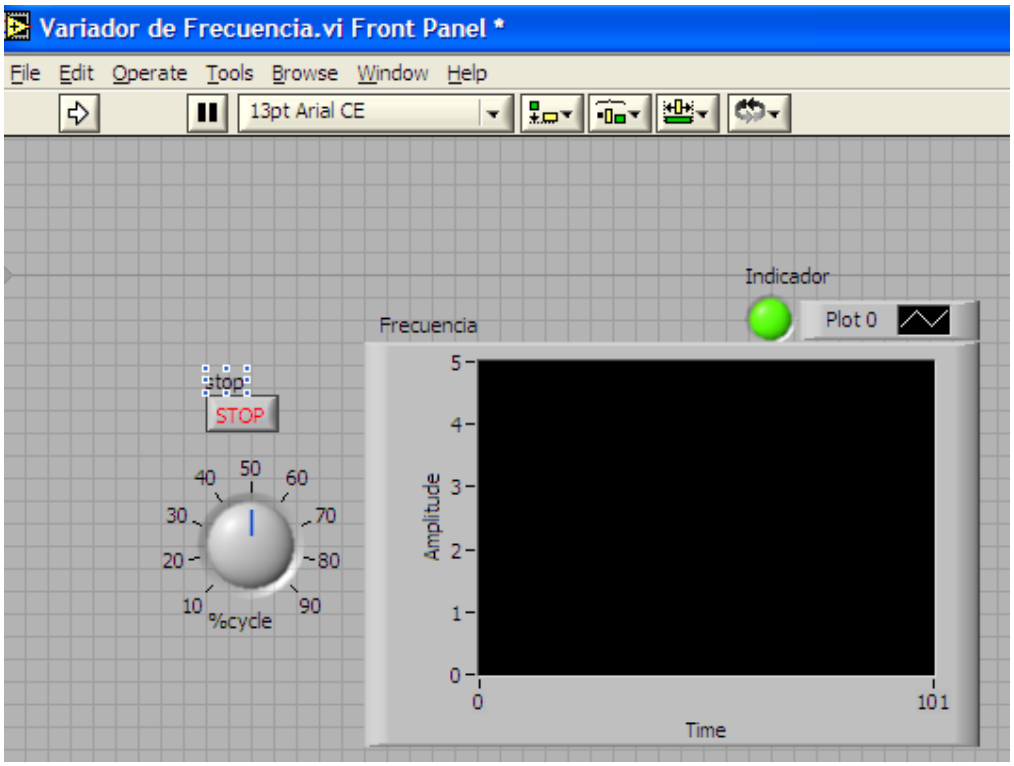


Figura M10 Panel Frontal Programa terminado

Anexo No 5

Practica No5

Tema: Detector de Incendio

Objetivo:

Realizar un proyecto en LabView que interactué con el modulo USB 6009 y un circuito electrónico y el sensor de incendio ECO 1000; de entrada digital y salida digital para poder así encender una alarma sonora.

Descripción:

Se manipulara el control de encendido y apagado de un parlante; por el puerto de salida digital del modulo USB6009, una vez que el sensor de incendio se active sea por emanación de humo así como por el aumento brusco de temperatura

Entrada digital puerto Port0/line0

Salida digital puerto port0/line1

Se necesitaran los siguientes materiales para realizar la placa electrónica:

Materiales:

1	Regulador 7805
1	Regulador 7812
2	Diodos 1N4007
	Condensadores
4	0,1uF
2	Borneras 1x3
2	Borneras 1x2
1	Espadines 1x3
2	Resistencias 4,7K
1	Resistencias 150K
1	Resistencias 100K
1	Resistencias 470K
1	Transistor Tip110
1	Condensadores 10uF
	Condensadores
1	100pF
1	Baquelita
1	Base DIP8
1	Parlante 0,5W
1	LM555

Paso 1

Realizar la placa electrónico ver diagrama de conexión figura H1. Para mayor información ver hojas técnicas de los instrumentos electrónicos ver anexo7

- Primero referencia
- Segundo señal
- Tercero alimentación.

Conectar al Modulo Adquisidor de Datos la placa electrónica esta tiene que estar conectada a la fuente así como el parlante y los pines de los puertos respectivos, para guía ver en la placa los números que corresponden:

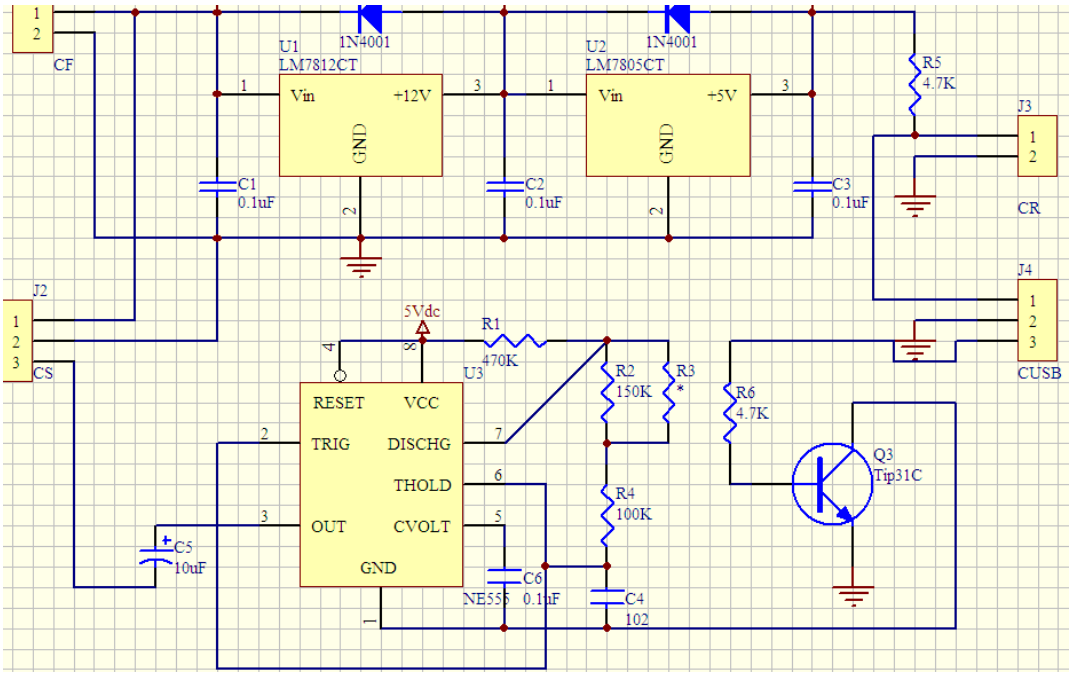


Figura H1 diagrama de conexión de la placa electrónica

Paso 2

Programación en LABVIEW

Crear la señal de entrada; para lo cual elegir un DAQ Assisten en el cual debe especificarse que es: de entrada, señal digital, de voltaje, en el puerto a0. en la configuración cambiar la entrada diferencial, el rango máximos 5V y mínimo 0V de voltaje Figura T2

- En LabView crear un VI en Blanco
- En el diagrama de bloque
- Barra de Funciones seleccionar Input

- Elegir DAQ Assistent
- Seleccionar el tipo de canal
    - Digital I/O
- Seleccionar el tipo de medida que se va a utilizar en nuestra práctica
    - Line Input
- Añadir el canal físico para nuestra tarea
    - Port0/line0

Una vez obtenida la señal por medio del asistente, este trabaja con 8bit por lo tanto es un array, el dato con que se va a trabajar es un escalar por lo que se necesita ingresar un index array para trabajar con el dato necesario:

- En la ventana de funciones
- All Funnctions
- Seleccionar Indeax Array Ver figura H2

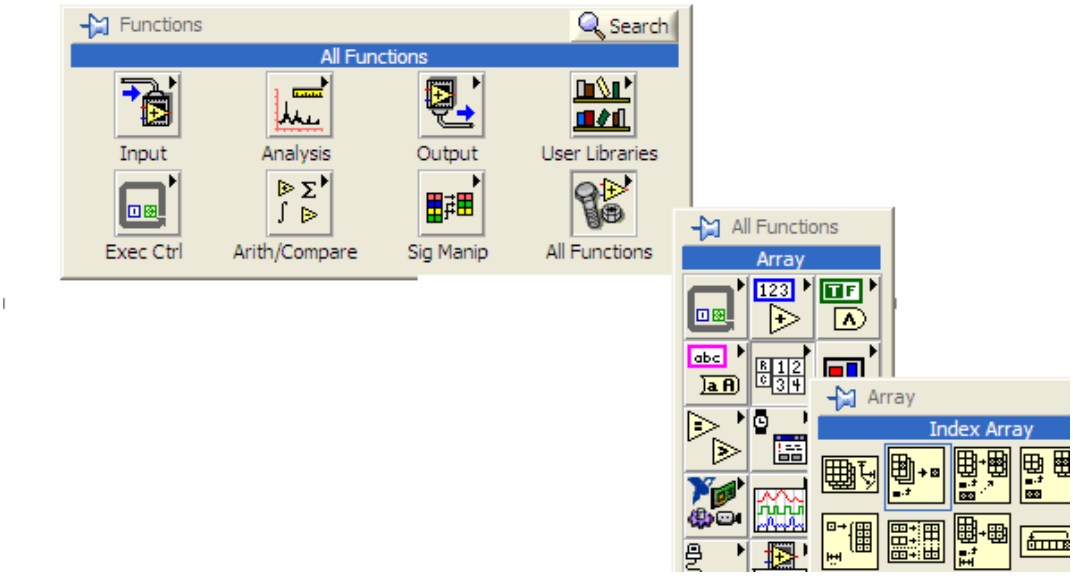


Figura H2 seleccionar un Index Array

La primera Terminal del Index array conectar con el dato de ingreso del DAQ, a la otra entrada crear un constante con el valor 0 ya que estamos trabajando con el puerto cero; la salida de este vamos a conectar con un indicador que recibira la señal de encendido y apagado Ver figura H3

- Clic en la salida del array
- Seleccionar Create indicador

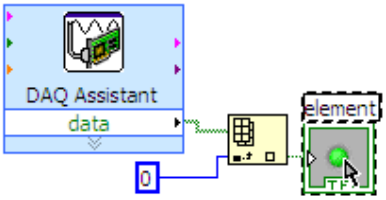


Figura H3 Censar la detección de un incendio por la activación del sensor

Desplegar imágenes para demostrar que el área de una casa va incendiándose para lo cual:

- En la barra de funciones Graphics&Sound
- Seleccionar Graphics Formats
- Read JPEG file.vi Ver figura H3

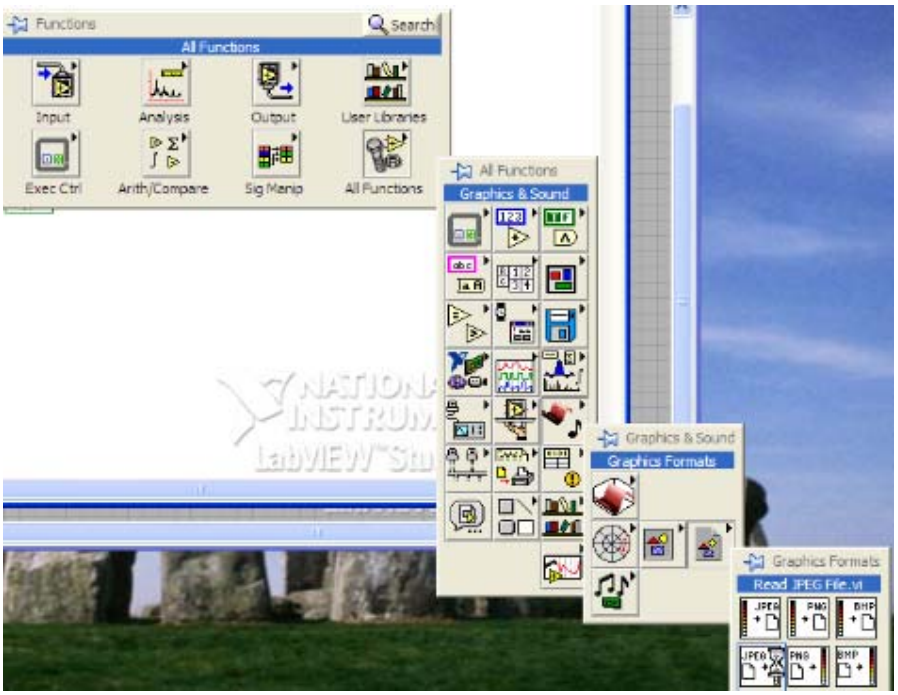


Figura H3 Función de escritura y lectura de un formato JPEG

A esta función tenemos que darle la dirección de la imagen a leer para esto se tiene que crear una constante con la dirección, esta función nos va a devolver un dato de imagen la cual se tiene que llamar a Ver Figuras H4 y H5:

- Picture función de la barra de funciones
- Draw Flattened Pixmap.vi
- Crear una constante y escribir la dirección de la imagen a leer
- A la salida de esta función crear un indicador

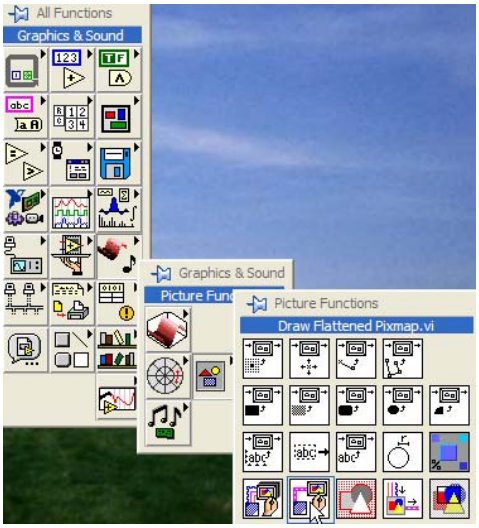


Figura H4 Barra de funciones de gráficos y sonidos

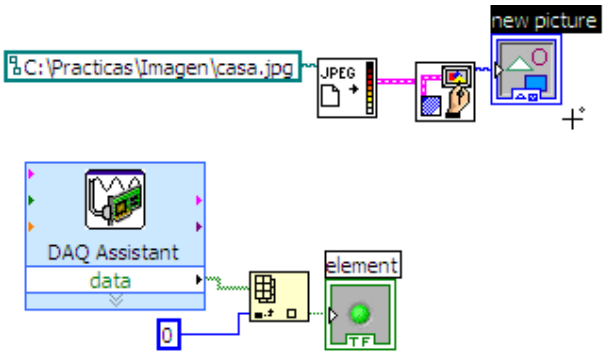


Figura H5 Programación llamada de una figura

Crear una secuencia de imágenes una vez que se ha activado el sensor:

- En la barra de funciones seleccionar structure
- Seleccionar flat secuence

En cada parte de la escena llamar a una imagen diferente y darle un tiempo especifico para visualizar la secuencia de la simulación de un incendio Insertar un DAQ Assistent en la ultima escena llamar a un DAQ ya que por ultimo hay que activar la señal sonora

Ver figura H6

- En el canal de la tarea de salida seleccionar:
  - Digital I/O
- En el canal a utilizar
  - Line Output
- En puerto a utilizar en la tarea
  - Port0/line0

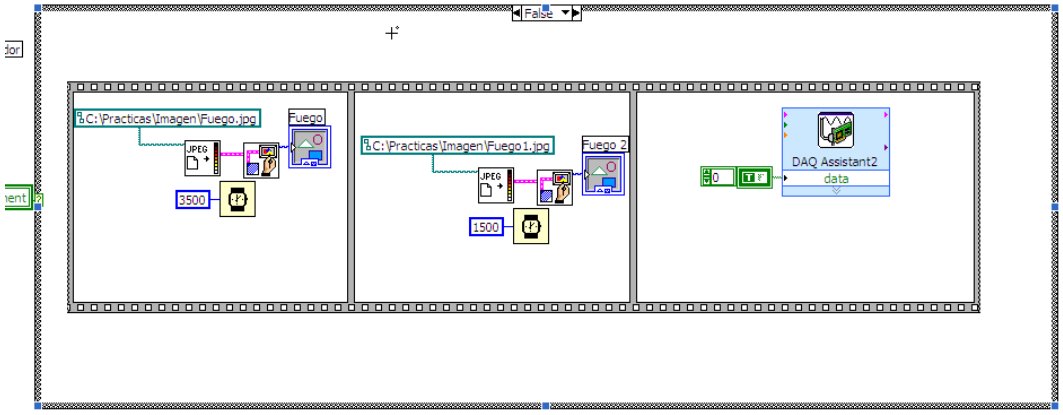


Figura H6 Programación la simulación de un incendio

Por último crear un lazo cíclico que ejecutara el programa mientras el usuario lo desee, edemas incorporar un botón de Stop Ver Figura H7

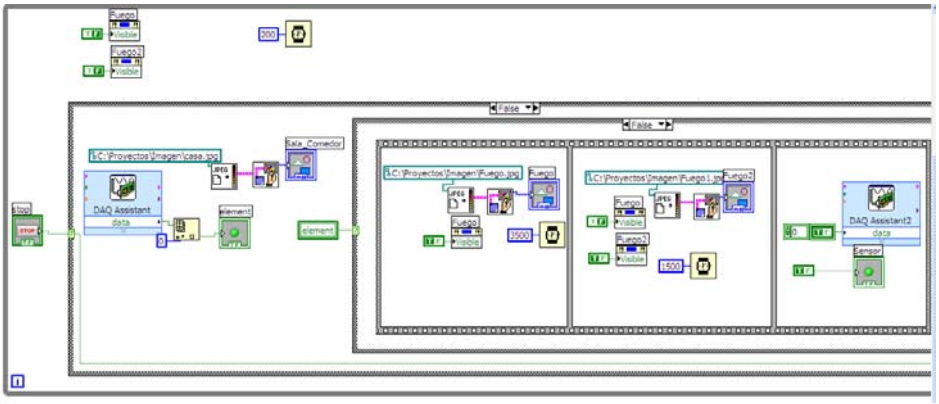


Figura H7 diagrama de bloque del Programa terminado

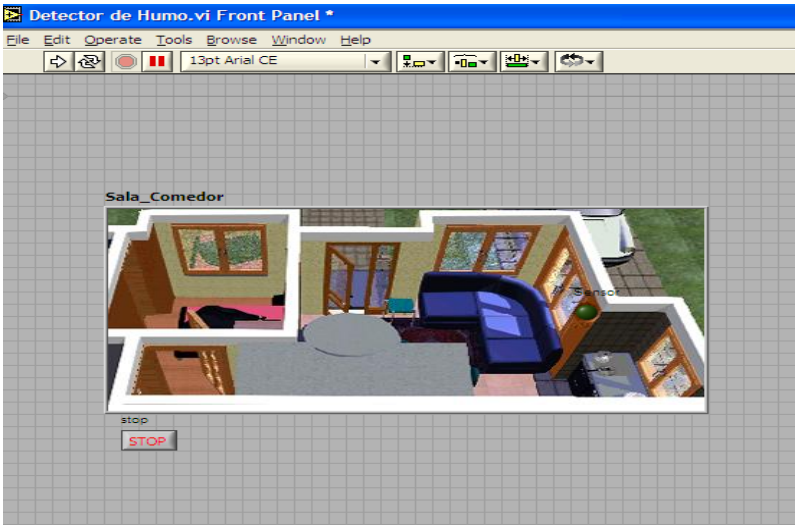


Figura H8 Panel frontal programa Humo

Mayor información ver archivo multimedia Programación el LabVIEW

Humo.AVI

Anexo No 6

Practica N6

Tema: Control de clave con el uso de un display por medio de pulsadores .

Objetivo:

Realizar un proyecto en LabView que interactué con el modulo USB 6009 y un circuito electrónico que sirva de entrada digital; y salida digital para poder activar un dispositivo electrónico con alimentación externa

Descripción:

En esta práctica se controlara el ingreso de un clave de tres dígitos la cual será ingresada por medio de tres pulsadores(Up, Down, Enter); cada número seleccionado se visualizara en un display como en el panel frontal del programa LabVIEW.

Se necesitaran los siguientes materiales para realizar la placa electrónica:

Materiales:

1	Regulador 7805
1	Regulador 7812
2	Diodos 1N4007
	Condensadores
3	0,1uF
1	CI 74LS48
3	Pulsadores NA
2	Borneras 1x3
1	Borneras 1x2
3	Espadines 1x3
4	Resistencias 4,7K
7	Resistencias 330
1	Display
1	Transistor 2N3904
1	Rele 12 V
1	Baquelita

Paso 1

Realizar la placa electrónico ver diagrama de conexión figura T1. Se debe colocar los instrumentos electrónicos según el diagrama. Para mayor información de las hojas técnicas de los instrumentos electrónicos ver anexo7

Conectar al Modulo Adquisidor de Datos la placa electrónica esta tiene que estar conectada a la fuente así como el motor y los pines de los puertos respectivos, para guía ver en la placa los números que corresponden:

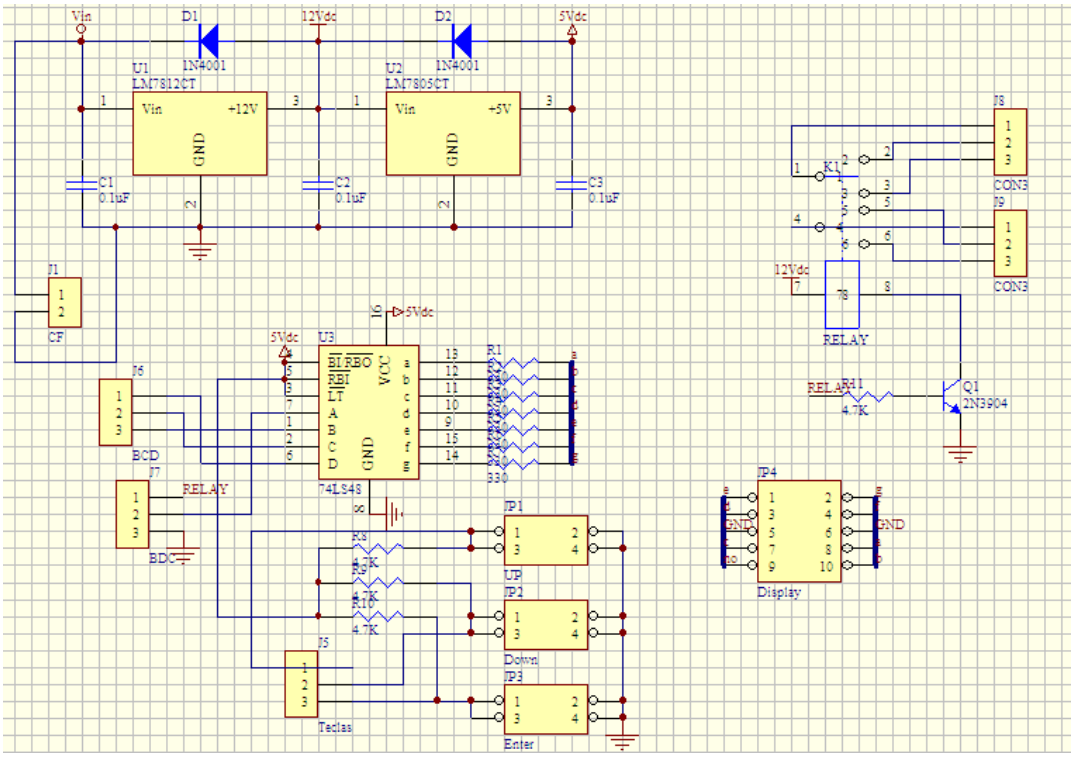


Figura T1 diagrama de conexión de la placa electrónica

Paso 2

Programación en LABVIEW

En el Programa LabVIEW crear el display para lo cual se procede:

En la barra de funciones del panel frontal

Array&Cluster

Seleccionar Cluster Ver Figura T2

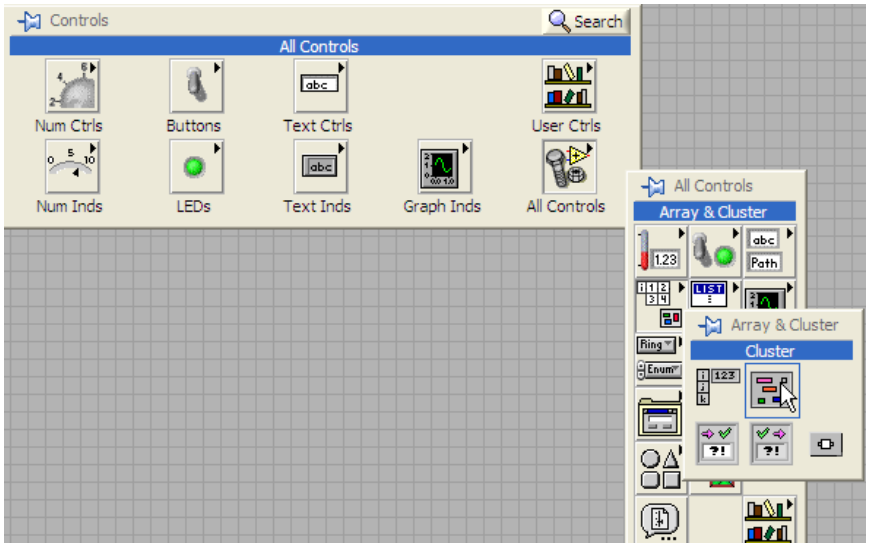


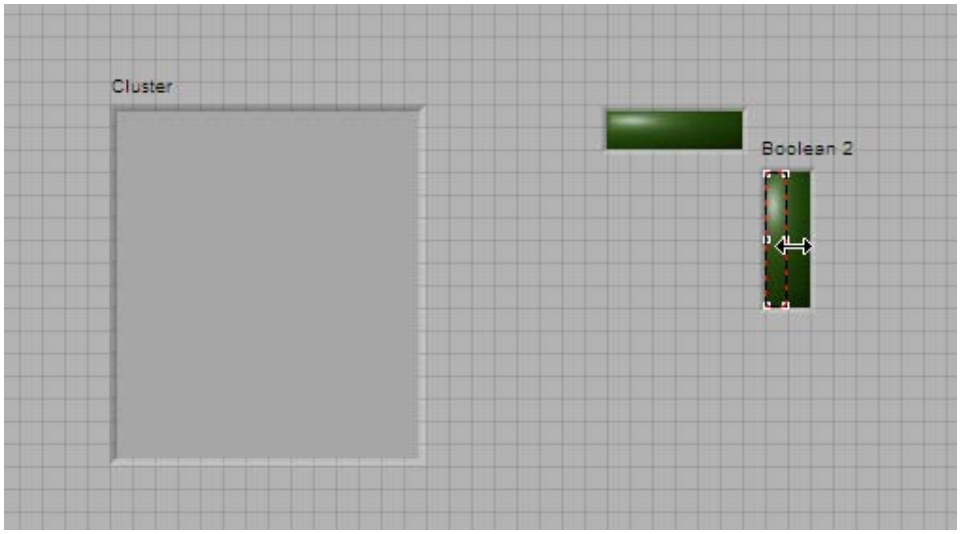
Figura T2 Control Cluster

Se procede a crear el display de siete segmentos, con Led´s; ocultar los label y además darles la forma del display Ver Figura T3

En la barra de controles

Buttons

Seleccionar ocho Square LED



T3 Creación del un display En el panel frontal

Una vez creado el display se procede a manipular el cluster ya que en el se coloca el display para lo cual se realiza:

En el diagrama de bloque en la barra de funciones

- Seleccionar Cluster
- Seleccionar Cluster Constan Ver Figura T4

Proceder a colocar en la Constante de cluster, ocho constantes Boleanas ya que el display contiene siete segmentos:

- En la barra de funciones Bolean
- Seleccionar False Constant Ver Figura T5

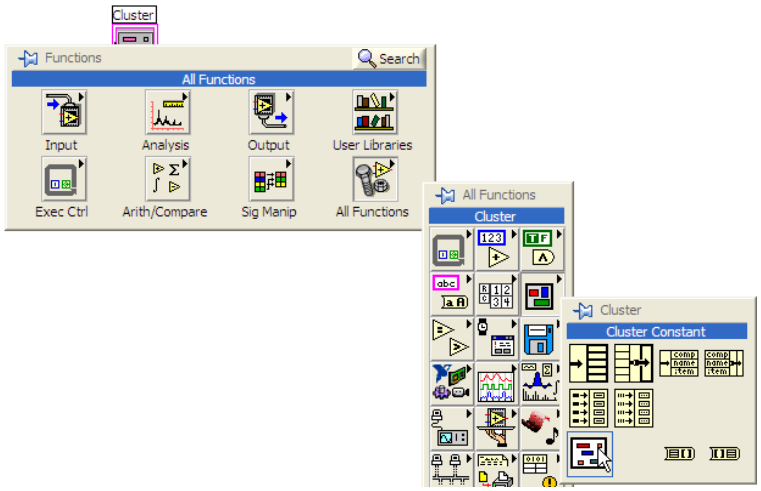


Figura T4 Función Cluster Constant

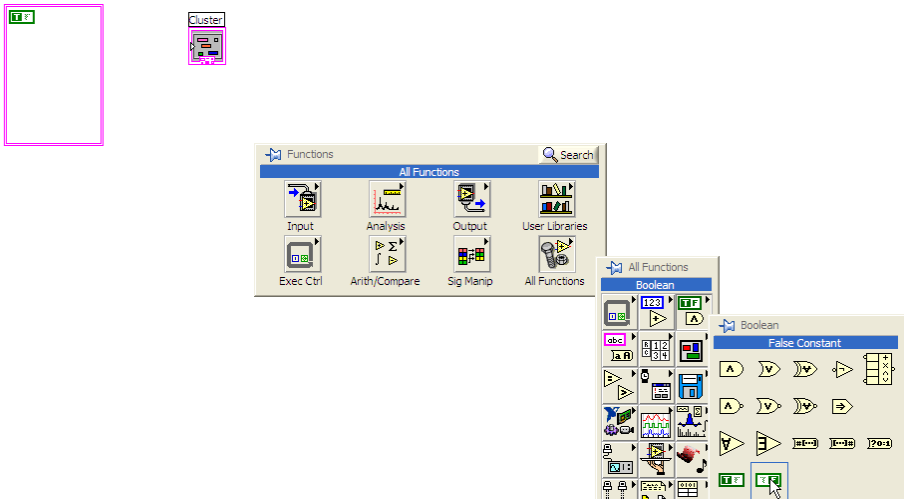


Figura T5 Programación de las variables de cada segmento

Proceder a crear cada uno de los números que aparecerán en el display para lo cual se tiene que dar valores de verdadero a las constantes de falso de acuerdo a cada segmento que se desea encender para crear el número así como encerrar a la constante

de cluster en un lazo Case Structure que compare del 0-9, también crear un indicador numérico que servirá de comparación. Para encender el Dispay que se encuentra en la tarjeta electrónica se necesita de colocar cuatro DAQ assistent ya que en la tarjeta contiene un decodificador de siete. Los DAQ tiene que ser de salida digital, se necesita crear de cada uno de ellos variables locales a las cuales se debe crear constantes para enviar al decodificador la señal del numero Ejemplo

En el caso cero la señal a enviar será:

Data Falso

Data2 Falso

Data3 Faso

Data4 Falso

Se tiene que modificar dependiendo del número que sea Ver Figuras T6 Y T7

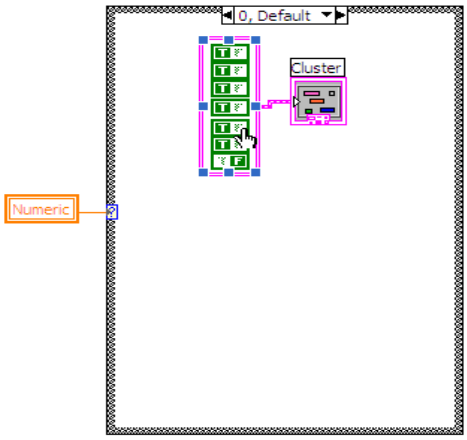


Figura T6 Creación de cada uno de los números en el Panel Frontal

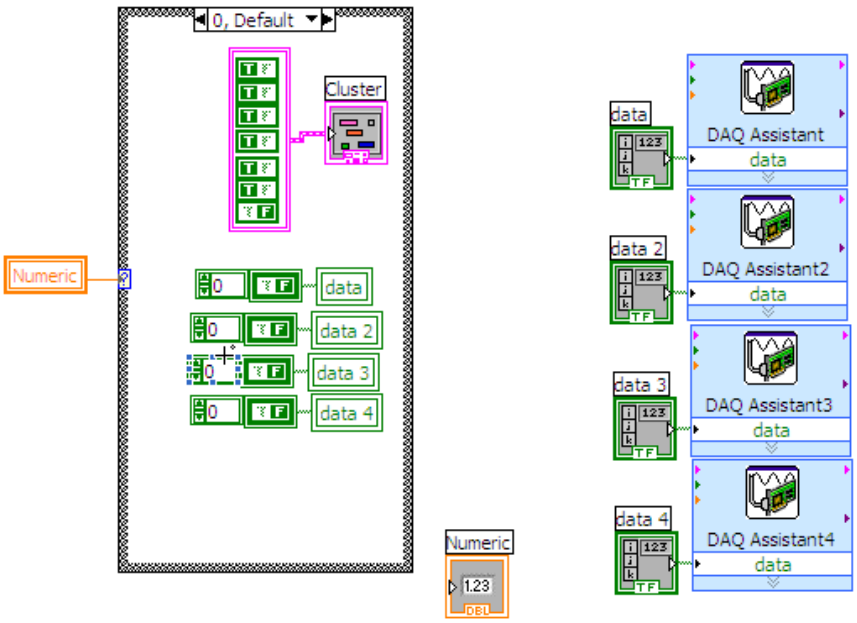


Figura T7 Creación del número cero en el display de la tarjeta

La programación anterior se tiene que realizar en los nueve casos pero se tiene que ir cambiando las posiciones de los Leds así como el dato binario a enviar al decodificador por medio de las variables de los DAQ Ver Figura T8

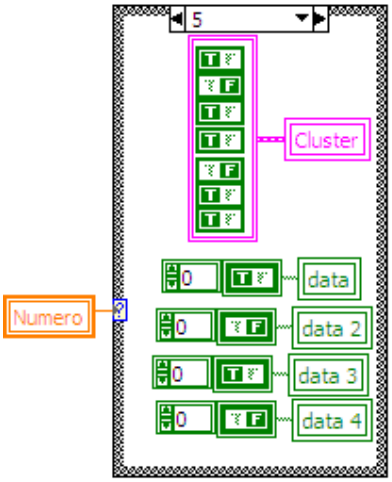


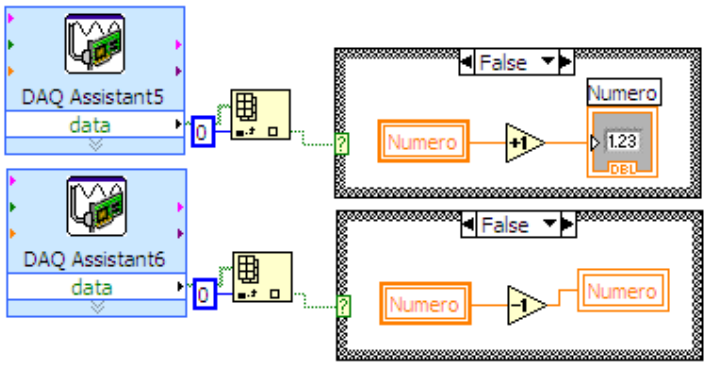
Figura T8 Programación del número Cinco

Se ha programado el control del Display; se procede a programar el control de los pulsadores de entrada tanto el ascendente como el descendente para lo cual: insertar un asistente de datos para cada uno, de entrada digital como de puertos diferentes, además de incrementar o disminuir la variable número Ver Figura T9

En la barra de funciones Input

DAQ assistent

D/O digital escoger un canal libre para cada uno



T9 Programación de los pulsadores de aumento o disminución, de ingreso de datos

Una vez programado los pulsadores de aumento o disminución, queda programar el tercer pulsador que es el de aceptar para lo cual a los números ingresados se procede a almacenar en un indicador numérico, al que hay que proceder a transformar en unidades decenas y centenas ya que la clave se encuentra dada en tres dígitos Ver Figura T10

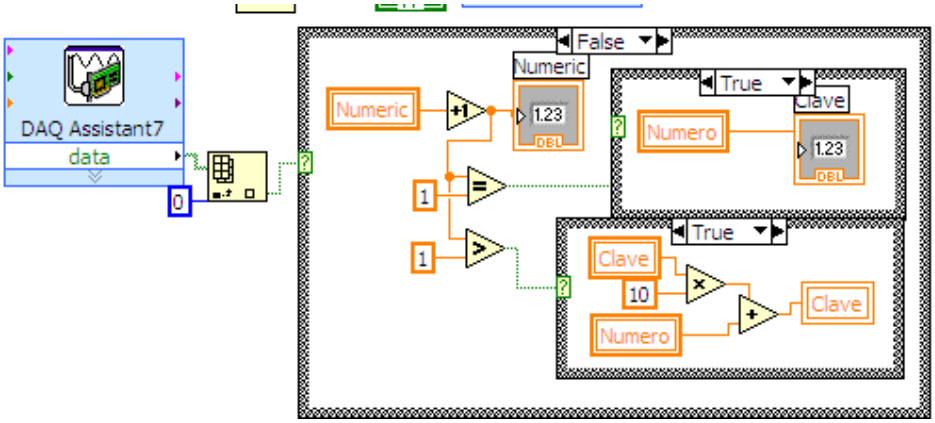


Figura T10 Programación del pulsador enter

Una vez ingresado los tres números se procede a realizar las comparaciones para saber si la clave fue correcta o incorrecta Ver Figura T11

- Si se ingresaron tres dígitos
- Los números ingresados con comparar con la clave almacenada

- Ingresar a un lazo For Loop para dar un tiempo de ejecución y poder simular la abertura de una puerta
- Insertar un DAQ asisistent para enviar la señal de activación al relay y así poder abrir el portero eléctrico
- Insertar un mensaje visual de que la puerta fue abierta
- En la parte falsa del caso case inicializar las variables.
- En el caso falso del case principal emitir un mensaje de “Clave Incorrecta”
- Para elegir la función para desplegar un mensaje realizar:
- En la ventana de Funciones Time&Dialog
- Seleccionar One Button Dialog
    - Esta función contiene la primera Terminal despliega el mensaje
    - La segunda y tercera se crean botones

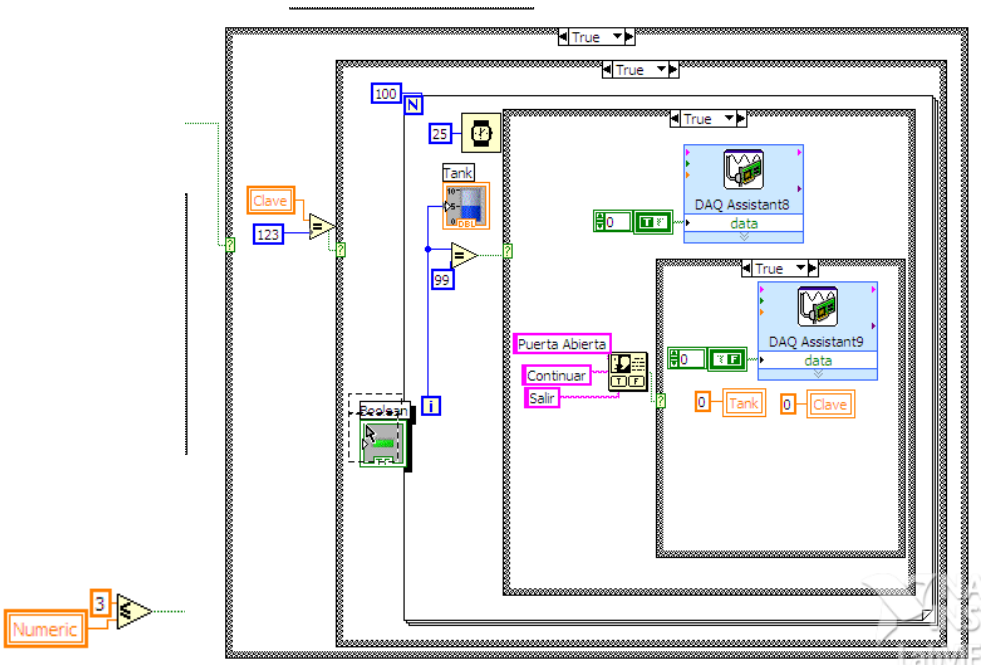


Figura T 11 Programación de la comparación de la clave ingresada con la clave almacenada

Una vez terminado el proyecto quedara de la siguiente forma tanto el diagrama de bloque como el Panel frontal Ver Figura T12 y T13

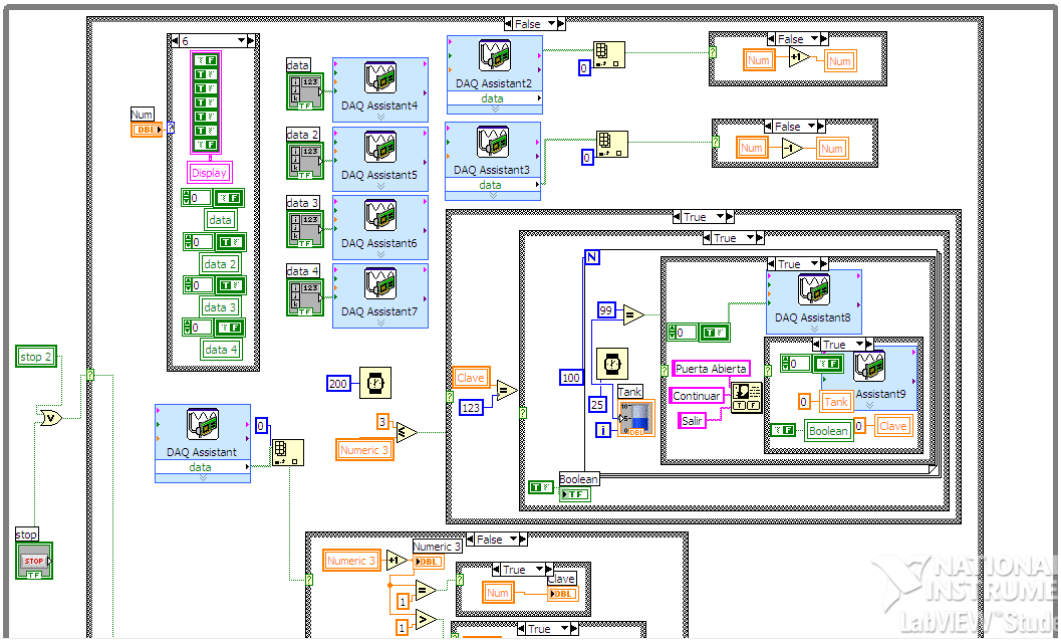


Figura T12 Diagrama de bloque programa Display& teclado

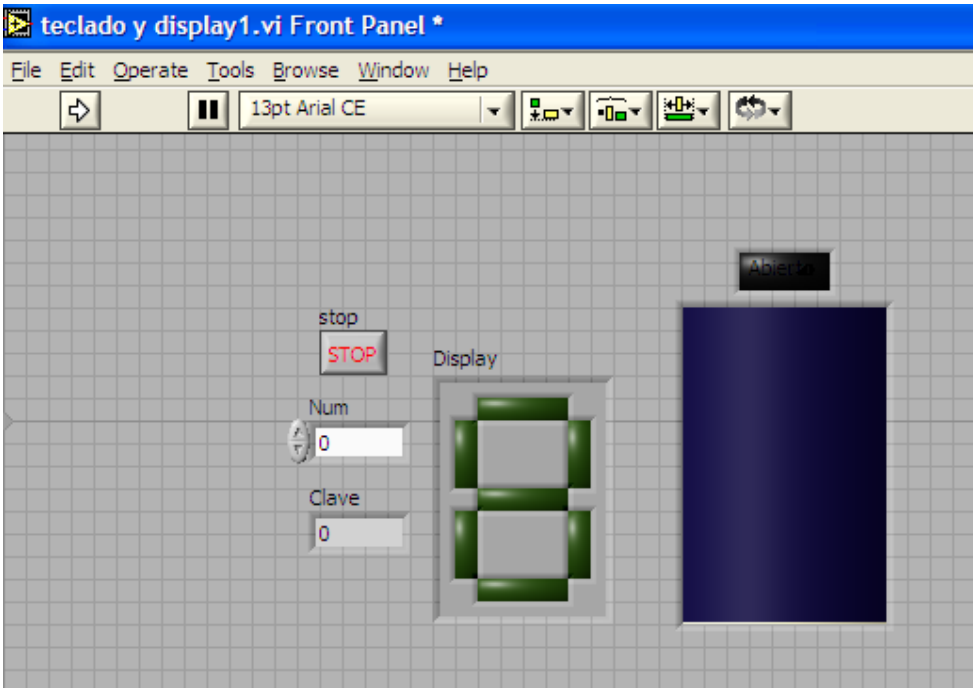


Figura T13 Panel Frontal programa Display& teclado

Para mayor información Ver archivos Multimedia

Display&Teclado.AVI

Bibliografía

- H.H. Skilling, "Circuitos en Ingeniería Eléctrica", Ed. CECSA, 1973
- T.L.Floyd, "Fundamentos de Sistemas Digitales", Prentice-Hall, 2000
- J.M. Angulo y J. García , "Sistemas Digitales y Tecnológicos de Computadores", Paraninfo, 1995
- H. Tabú, "Circuitos Digitales y Microprocesadores", McGraw-Hill,1996
- C. Roth, "Fundamental of Logic Design", West Publishing Co,1975
- J. Brophy, "Electrónica Fundamental para Científicos", Ed. Reverté,1974
- Manuales de National Instruments
- Tutoriales de Labview

INTERNET

[1] www.ni.com	National Instruments
[2] www.national.com	National Semiconductor
[3] www.electronicsworkbench.com/	Electronic Wokbench
[4]www.vernier.com/labview/nielvis.html	Educational Laboratories