



**UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL**

CARRERA DE INGENIERÍA ELECTRÓNICA

Trabajo de titulación previo a la obtención del título de:

INGENIERO ELECTRÓNICO

Título:

**“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE
ADQUISICIÓN DE DATOS USANDO EL KIT DE SENSORES
SUNFOUNDER, MEDIANTE EL USO DEL HARDWARE
LATTEPANDA Y ARDUINO CON INTERFASE GRÁFICA
DESARROLLADA EN LABVIEW”.**

Autores:

JUAN IGNACIO PALOMEQUE GUERRA

DIEGO SEBASTIÁN LOZADA LÓPEZ

Tutor del proyecto Técnico:

ING. GABRIEL SANTIAGO GARCÍA VÁSQUEZ, Msc.

GUAYAQUIL – ECUADOR

2018

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN.

Nosotros, **JUAN IGNACIO PALOMEQUE GUERRA** y **DIEGO SEBASTIÁN LOZADA LÓPEZ**, autorizamos a la **UNIVERSIDAD POLITÉCNICA SALESIANA** la publicación total o parcial de este trabajo de titulación y su reproducción sin fines de lucro.

Además, declaramos que los conceptos, análisis desarrollados y las conclusiones del presente trabajo son de exclusiva responsabilidad de los autores.

Guayaquil, 2018

.....

Juan Ignacio Palomeque Guerra

Cédula: 0919660076

.....

Diego Sebastián Lozada López

Cédula: 0940780414

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO
DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, **JUAN IGNACIO PALOMEQUE GUERRA**, con documento de identificación N° **0919660076**, y **DIEGO SEBASTIÁN LOZADA LÓPEZ**, con documento de identificación N° **0940780414**, manifestamos nuestra voluntad y cedemos a la **UNIVERSIDAD POLITÉCNICA SALESIANA** la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de grado titulado **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ADQUISICIÓN DE DATOS USANDO EL KIT DE SENSORES SUNFOUNDER, MEDIANTE EL USO DEL HARDWARE LATTEPANDA Y ARDUINO CON INTERFASE GRÁFICA DESARROLLADA EN LABVIEW”** mismo que ha sido desarrollado para optar por el título de **INGENIERO ELECTRÓNICO**, en la Universidad Politécnica Salesiana, quedando la universidad facultada para ejercer plenamente los derechos antes cedidos.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en mi condición de autor me reservo los derechos morales de la obra antes citada. En concordancia, suscrito este documento en el momento que hago entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 2018

.....

Juan Ignacio Palomeque Guerra

Cédula: 0919660076

.....

Diego Sebastián Lozada López

Cédula: 0940780414

**CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN
SUSCRITO POR EL AUTOR**

Yo, **GABRIEL SANTIAGO GARCÍA VÁSQUEZ**, director del proyecto de Titulación denominado **“DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ADQUISICIÓN DE DATOS USANDO EL KIT DE SENSORES SUNFOUNDER, MEDIANTE EL USO DEL HARDWARE LATTEPANDA Y ARDUINO CON INTERFASE GRÁFICA DESARROLLADA EN LABVIEW”** realizado por los estudiantes, **JUAN IGNACIO PALOMEQUE GUERRA** y **DIEGO SEBASTIÁN LOZADA LÓPEZ**, certifico que ha sido orientado y revisado durante su desarrollo, por cuanto se aprueba la presentación del mismo ante las autoridades pertinentes.

Guayaquil, 2018

.....
Ing. Gabriel Santiago García Vásquez, Msc.

DEDICATORIA

A mi madre, Mariela Guerra, y a mi padre Juan Palomeque, quienes me han sido un pilar fundamental para cumplir mis objetivos.

A mi novia, Valeria Montoya, quien me ha brindado todo su apoyo durante mi etapa profesional.

Finalmente, a mi hermana, Macarena Palomeque, que a distancia estuvo día tras días brindándome su colaboración y consejos.

Juan Ignacio Palomeque Guerra

A mi madre, Paola López, y a mi padre Francisco Lozada, quienes son parte importante en mi vida y fuente de fuerzas para no desmayar en cada objetivo que me propongo brindando apoyo y confianza para superar cualquier adversidad.

A mis hermanos, Alexander Lozada y Axel Lozada por ser un ejemplo a seguir y brindarme apoyo a lo largo de mi carrera estudiantil.

A mi novia, Mishelle Alvear, quien me ha brindado su apoyo, consejos y comprensión durante todo el proceso que me he tomado en la elaboración del proyecto.

Diego Sebastián Lozada López

AGRADECIMIENTO

A Dios por dotarnos de sabiduría y conocimiento que día a día nos ayuda a cumplir nuestras metas. A mis padres, por el privilegio de otorgarme una educación de primera, buenos valores y brindarme todo su apoyo durante el transcurso de mi vida estudiantil. Agradecemos a la Universidad Politécnica Salesiana y a cada uno de los docentes de cada materia que hemos cursado, por haber formado parte fundamental en nuestro desarrollo académico y profesional.

A la empresa Artechnologies y al CEO Antonio Ricaurte, que nos brindó los recursos de su empresa, para el desarrollo de nuestro proyecto.

Agradecemos a su vez al Ingeniero Gabriel García como tutor de nuestro proyecto técnico, quien nos guió, orientó, apoyó y corrigió en el desarrollo del proyecto, a través de sus conocimientos y profesionalismo.

Juan Ignacio Palomeque Guerra

AGRADECIMIENTO

En primera instancia a mis padres por los valores inculcados, por brindarme la oportunidad de haber disfrutado de una educación de calidad gracias a su esfuerzo queriendo siempre lo mejor para mí y por la confianza depositada en cada uno de mis actos, a mis hermanos por el apoyo brindado todos los días.

A la Universidad Politécnica Salesiana ya que gracias a su oferta académica y laboratorios permitieron aplicar de manera práctica los conocimientos y cátedras impartidas por cada uno de los docentes logrando así un buen desempeño en el ámbito laboral.

Al Ingeniero Gabriel García quien como docente estuvo presto a aclarar cada una de las dudas, como tutor nos brindó ideas en el desarrollo del proyecto a través de sus conocimientos, profesionalismo, motivación y constancia.

A cada uno de los docentes quienes compartieron sus conocimientos y fueron los responsables de explotar nuestras capacidades logrando el crecimiento académico y profesional.

Diego Sebastián Lozada López

RESUMEN

| AÑO | ALUMNOS | DIRECTOR DE PROYECTO TÉCNICO | TEMA DE PROYECTO TÉCNICO |
|------|---|---|--|
| 2018 | JUAN IGNACIO PALOMEQUE GUERRA DIEGO SEBASTIÁN LOZADA LÓPEZ | ING. GABRIEL SANTIAGO GARCÍA VÁSQUEZ, Msc. | “DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE ADQUISICIÓN DE DATOS USANDO EL KIT DE SENSORES SUNFOUNDER, MEDIANTE EL USO DEL HARDWARE LATTEPANDA Y ARDUINO CON INTERFASE GRÁFICA DESARROLLADA EN LABVIEW”. |

El proyecto consta de un módulo principal y seis módulos didácticos intercambiables, en el módulo principal se encuentra: Pantalla táctil de 7 pulgadas, tarjeta Arduino MEGA 2560, tarjeta LattePanda, galleta de protoboard, batería de litio de 12V 8A, regulador de voltaje de 5V 1.5A y una bornera 2x6 de donde se pueden tomar los valores de voltaje indicados. En los seis módulos didácticos intercambiables se encuentran los elementos utilizados para desarrollar las prácticas definidas y disponibles para el desarrollo de nuevas prácticas.

La finalidad del presente proyecto es diseñar y construir un sistema SCADA de adquisición de datos, los datos son adquiridos a través de los sensores utilizados, los mismos son procesados y decodificados por el sistema embebido Arduino MEGA 2560.

El uso del Toolkit LINX de Makerhub facilita el desarrollo de prácticas de laboratorio, las mismas son implementadas con el fin de emular los diferentes procesos que se ejecutan en los diferentes campos de las industrias, con esto se busca que los estudiantes cuenten con una nueva herramienta que permita reforzar la utilidad del software LabVIEW 2016 de tal manera que tengan la oportunidad de cargar el VI en el sistema embebido LattePanda y a su vez comunicarse a través de conexión USB con el sistemas embebidos Arduino Mega 2560.

Los datos adquiridos por los sensores y actuadores son enviados al sistema embebido Arduino Mega que pueden ser visualizados en la pantalla táctil, la misma tiene la

función de interfaz HMI, esta característica permite que el usuario interactúe de manera didáctica con el módulo.

Dentro del desarrollo del proyecto se utiliza la herramienta virtual DataSocket que permite intercambiar datos a través de la red ya sea por Wi-Fi o Ethernet, esto en tiempo real, permitiendo así el monitoreo y control del comportamiento del sistema implementado desde otro terminal, al mismo tiempo se utiliza la herramienta Datalogging and Supervisory Control Module, que permite registrar base de datos, administrar alarma y eventos realizados por el módulo.

El proyecto que se desea implementar está dotado con una batería recargable Li-ion de cuyo voltaje es 12V y su corriente de 8A que permite utilizar elementos adicionales que demandan un consumo superior al proporcionado por el sistema embebido Arduino MEGA 2560, en la cual los estudiantes son capaces de ejecutar prácticas debido a la amplia variedad de aplicaciones que ofrecen los sistemas embebidos, esto contribuye a que los estudiantes apliquen los conocimientos adquiridos en las clases impartidas por los docentes.

Palabras Claves: LABVIEW, ARDUINO, LATTEPANDA, SENSORES, SCADA, DATASOCKET, LINX, HMI, DSC.

ABSTRACT

| YEAR | STUDENTS | DIRECTOR OF TECHNICAL PROJECT | TECHNICAL PROJECT THEME |
|------|--|---|---|
| 2018 | <p style="text-align: center;">JUAN IGNACIO PALOMEQUE GUERRA</p> <p style="text-align: center;">DIEGO SEBASTIÁN LOZADA LÓPEZ</p> | <p style="text-align: center;">ING. GABRIEL SANTIAGO GARCÍA VÁSQUEZ, Msc.</p> | <p style="text-align: center;">"DESIGN AND IMPLEMENTATION OF A DATA ACQUISITION SYSTEM USING THE SUNFOUNDER SENSORS KIT, THROUGH THE USE OF LATTEPANDA AND ARDUINO HARDWARE WITH GRAPHIC INTERFACE DEVELOPED IN LABVIEW".</p> |

The project consists of a main module and six interchangeable didactic modules, in the main module is: 7-inch touch screen, Arduino MEGA 2560 card, LattePanda card, breadboard biscuit, 12V 8A lithium battery, 5V voltage regulator 1.5A and a 2x6 terminal board from which the indicated voltage values can be taken. In the six interchangeable didactic modules are the elements used to develop the defined and available practices for the development of new practices.

The purpose of the present project is to design and build a SCADA system for data acquisition, the data will be acquired through the sensors used, they are processed and decoded by the Arduino MEGA 2560 embedded system.

The use of the LINK Toolkit Makerhub facilitated the development of laboratory practices, they were implemented in order to emulate the different processes that are executed in different fields of the industries, with this it is sought that students have a new tool that allows to reinforce the usefulness of LabVIEW 2016 software in such a way that they have the opportunity to load the VI in the embedded system LattePanda and in turn communicate through USB connection with the embedded Arduino Mega 2560 systems.

The data acquired by the sensors and actuators are sent to the Arduino Mega embedded system that can be visualized on the touch screen, it has the function of HMI interface, this feature allows the user to interact in a didactic way with the module.

Within the development of the project, the virtual tool DataSocket is used, which allows exchanging data through the network through Wi-Fi or Ethernet, this in real time, allowing the monitoring and control of the behavior of the system implemented from another terminal, At the same time, the Datalogging and Supervisory Control Module is used, which allows to register the database, manage the alarm and events carried out by the module.

The project to be implemented is equipped with a rechargeable Li-ion battery whose voltage is 12V and its current of 8A that allows the use of additional elements that demand a higher consumption than the one provided by the Arduino MEGA 2560 embedded system, in which the students are able to execute practices due to the wide variety of applications offered by embedded systems, this will help students apply the knowledge acquired in the classes taught by teachers.

Keywords: LABVIEW, ARDUINO, LATTEPANDA, SENSORES, SCADA, DATASOCKET, LINX, HMI, DSC.

ÍNDICE GENERAL

| | |
|---|------|
| CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN..... | I |
| CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA | II |
| CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN SUSCRITO POR EL AUTOR | III |
| DEDICATORIA | IV |
| AGRADECIMIENTO | V |
| RESUMEN..... | VII |
| ABSTRACT..... | IX |
| ÍNDICE GENERAL..... | XI |
| ÍNDICE DE FIGURAS..... | XIII |
| INTRODUCCIÓN | 1 |
| 1. EL PROBLEMA..... | 2 |
| 1.1. Antecedentes | 2 |
| 1.2. Importancia y Alcances | 2 |
| 1.3. Delimitación | 3 |
| 1.3.1. Temporal | 3 |
| 1.3.2. Espacial | 3 |
| 1.3.3. Académica..... | 3 |
| 1.4. Objetivos | 4 |
| 1.4.1. Objetivo general | 4 |
| 1.4.2. Objetivos específicos | 4 |
| 1.5. Marco Metodológico | 5 |
| 2. Fundamentos teóricos | 6 |
| 2.1. Hardware | 6 |
| 2.1.1. LattePanda..... | 6 |
| 2.1.2. Pantalla IPS de 7”..... | 9 |
| 2.1.3. Pantalla Capacitiva de 7” | 10 |
| 2.1.4. Arduino | 11 |
| 2.1.5. Arduino Mega 2560 | 12 |
| 2.1.6. Sunfounder | 13 |
| 2.2. Software..... | 14 |
| 2.2.1. LabVIEW | 14 |
| 2.2.2. Toolkit LINX de Digilent/LabVIEW MakerHub | 16 |
| 2.2.3. NI VISA | 16 |

| | | |
|--------|--|-----|
| 2.2.4. | DataSocket | 17 |
| 2.2.5. | Datalogging and Supervisory Control Module. | 17 |
| 2.2.6. | Arduino Software | 18 |
| 3. | MARCO METODOLÓGICO..... | 19 |
| 3.1. | Diseño de planos en impresora 3D | 19 |
| 3.2. | Construcción de la estructura | 23 |
| 3.3. | Conexiones eléctricas | 26 |
| 3.3.1. | Conexión de Pantallas IPS y Capacitiva | 27 |
| 3.3.2. | Conexión de antena WI-FI..... | 30 |
| 3.3.2. | Puesta en marcha de sistema embebido LattePanda..... | 31 |
| 3.4. | Configuración de Software y Drivers..... | 33 |
| 3.4.1. | Instalación del Software NI-VISA..... | 34 |
| 3.4.2. | Instalación del toolkit LINX de Digilent/LabVIEW MakerHub | 38 |
| 3.4.3. | Instalación del módulo Datalogging and Supervisory Control Module | 41 |
| 4. | PRÁCTICAS DE LABORATORIO | 46 |
| 4.1. | Práctica #1 | 47 |
| 4.2. | Práctica # 2 | 60 |
| 4.3. | Práctica # 3 | 74 |
| 4.4. | Práctica # 4 | 93 |
| 4.5. | Práctica # 5 | 106 |
| 4.6. | Práctica # 6 | 119 |
| 4.7. | Práctica # 7 | 131 |
| 4.8. | Práctica # 8 | 141 |
| 4.9. | Práctica # 9 | 153 |
| 4.10. | Práctica # 10 | 170 |
| | RESULTADOS..... | 188 |
| | IMPACTO E INNOVACIÓN..... | 191 |
| | FUNCIONALIDAD..... | 192 |
| | CONCLUSIONES | 193 |
| | RECOMENDACIONES | 195 |
| | REFERENCIAS BIBLIOGRÁFICAS..... | 197 |
| | ANEXOS | 200 |

ÍNDICE DE FIGURAS

| | |
|---|----|
| Figura 1: Sistema embebido LattePanda 2G/32G. (DFRobots, 2016)..... | 8 |
| Figura 2: Pantalla IPS de 7". (DFRobots, 2016)..... | 9 |
| Figura 3: Pantalla Capacitiva de 7". (DFRobots, 2016)..... | 10 |
| Figura 4: Logo oficial de Arduino. (Arduino, 2013). | 11 |
| Figura 5: Tarjeta Arduino MEGA 2560. (Arduino, 2013)..... | 13 |
| Figura 6: Logo oficial de Sunfounder. (Sunfounder, 2013)..... | 14 |
| Figura 7: Logo LabVIEW. (National Instruments, 2016)..... | 15 |
| Figura 8: Logo Linx. (National Instruments, 2016)..... | 16 |
| Figura 9: NI VISA SOFTWARE. (National Instruments, 2016)..... | 17 |
| Figura 10: Arduino Software. | 18 |
| Figura 11: Diseño para impresora 3D de la base del sistema embebido LattePanda. | 19 |
| Figura 12: Diseño para impresora 3D de la base del sistema embebido Arduino Mega 2560..... | 19 |
| Figura 13: Diseño para impresora 3D de la base de la pantalla táctil..... | 20 |
| Figura 14: Diseño para impresora 3D de la base de la batería de 12V. | 20 |
| Figura 15: Diseño para impresora 3D de la base del protoboard..... | 20 |
| Figura 16: Diseño para impresora 3D de la base de la antena Wifi del sistema embebido LattePanda..... | 20 |
| Figura 17: Diseño para impresora 3D de la base del motor DC de la práctica 7..... | 21 |
| Figura 18: Diseño para impresora 3D de la base del rotor con ranura para imán del motor DC de la práctica 7. | 21 |
| Figura 19: Diseño para impresora 3D de la base del Sensor Hall de la práctica 7. ... | 21 |
| Figura 20: Diseño para impresora 3D de la base del servo-motor de la práctica 8.... | 21 |
| Figura 21: Diseño para impresora 3D de la base del foco de 12V de la práctica 10. | 22 |
| Figura 22: Diseño para impresora 3D de la base de los rieles y motores de la práctica 9..... | 22 |
| Figura 23: Diseño para impresora 3D del sensor IR de obstáculos de la práctica 9.. | 22 |
| Figura 24: Diseño para impresora 3D T indicadora de nivel de la práctica 9..... | 22 |
| Figura 25: Recorte de plancha de acrílico negro para base de la estructura. | 23 |
| Figura 26: Bases de acrílico negro limpias y recortadas..... | 23 |
| Figura 27: Montaje de bases 3D del módulo principal. | 24 |
| Figura 28: Montaje de equipos en bases 3D del módulo principal..... | 24 |
| Figura 29: Montaje y etiquetado del módulo 1. | 24 |
| Figura 30: Montaje y etiquetado del módulo 2. | 25 |
| Figura 31: Montaje y etiquetado del módulo 3. | 25 |
| Figura 32: Montaje y etiquetado del módulo 4. | 25 |
| Figura 33: Montaje y etiquetado del módulo 5. | 26 |
| Figura 34: Montaje y etiquetado del módulo 6. | 26 |
| Figura 35: Diagrama de conexión batería de litio de 12V 8AH, regulador de voltaje de 5V 1.5A..... | 26 |
| Figura 36: Retiro de mica de pantalla IPS. | 27 |
| Figura 37: Retiro de protector de la cinta adhesiva de la pantalla capacitiva. | 27 |
| Figura 38: Montaje de pantalla Capacitiva sobre pantalla IPS. | 28 |
| Figura 39: Ubicación de las ranuras para Capacitiva e IPS. (LattePanda, 2016)..... | 28 |
| Figura 40: Ranura para pantalla IPS. (LattePanda, 2016)..... | 29 |
| Figura 41: Conexión de cable flexible plano dentro de la ranura para pantalla IPS.. | 29 |
| Figura 42: Ajuste del actuador de la ranura de la pantalla IPS. (LattePanda, 2016). | 29 |

| | |
|--|----|
| Figura 43: Conexión del cable plano flexible de la pantalla Capacitiva..... | 30 |
| Figura 44: Conexión de antena Wifi. (LattePanda, 2016). | 30 |
| Figura 45: Cable USB 2.0. (DFRobots, 2016). | 31 |
| Figura 46: Fuente de alimentación NILLKIN. (DFRobots, 2016)..... | 31 |
| Figura 47: Conexión de alimentación del sistema embebido LattePanda. (LattePanda, 2016). | 32 |
| Figura 48: LED rojo de inicialización del sistema embebido LattePanda. (LattePanda, 2016). | 32 |
| Figura 49: Botón de encendido y pantalla encendida del módulo principal..... | 33 |
| Figura 50: Ventana inicial del proceso de instalación del software NI VISA. | 34 |
| Figura 51: Ventana de selección de destino de instalación del software NI VISA.... | 35 |
| Figura 52: Ventana para recibir notificaciones del software NI VISA. | 35 |
| Figura 53: Ventana donde se muestran los todos los archivos a instalarse en el software NI VISA. | 36 |
| Figura 54: Ventana donde se muestra el contrato de licencia del software NI VISA. | 36 |
| Figura 55: Ventana de aceptación de términos de confianza con los softwares de National Instruments Corporation..... | 37 |
| Figura 56: Ventana de inicio de proceso de instalación del software NI VISA y sus controladores. | 37 |
| Figura 57: Ventana donde se confirma la instalación del software NI VISA. | 38 |
| Figura 58: Ventana de alerta de reinicio de equipo para completar el proceso de instalación. | 38 |
| Figura 59: Búsqueda de toolkit Digilent LINX en el software VI Package Manager. | 39 |
| Figura 60: Se inicia instalación del toolkit LINX de Digilent en el software LabVIEW. | 39 |
| Figura 61: Se aceptan los acuerdos de licencia del toolkit LINX de Digilent. | 40 |
| Figura 62: Instalación completada del toolkit LINX de Digilent. | 40 |
| Figura 63: Ventana inicial del proceso de instalación del módulo DSC..... | 41 |
| Figura 64: Ventana de información del usuario del proceso de instalación del módulo DSC. | 42 |
| Figura 65: Ventana de registro de número serial de activación del módulo DSC. | 42 |
| Figura 66: Ventana donde se muestran los todos los archivos a instalarse en el módulo DSC. | 43 |
| Figura 67: Ventana de aceptación de términos de confianza con los softwares de National Instruments Corporation..... | 43 |
| Figura 68: Ventana donde se muestra el contrato de licencia del módulo DSC..... | 44 |
| Figura 69: Ventana de inicio de proceso de instalación del módulo DSC y sus controladores. | 44 |
| Figura 70: Ventana donde se confirma la instalación del módulo DSC. | 45 |
| Figura 71: Sensor de Gas MQ-2 práctica 1. (Sunfounder, 2013)..... | 48 |
| Figura 72: Buzzer Activo práctica 1. (Sunfounder, 2013). | 48 |
| Figura 73: Sensor Táctil práctica 1. (Sunfounder, 2013). | 49 |
| Figura 74: Bloque de apertura de comunicación práctica 1..... | 49 |
| Figura 75: Bloque de Lectura Analógico sensor de gas práctica 1. | 50 |
| Figura 76: Bloque de Lectura Analógico SET POINT del sensor de gas práctica 1. | 50 |
| Figura 77: Bloque de Lectura Digital sensor de gas práctica 1..... | 50 |
| Figura 78: Cálculo de concentración de Gas en PPM..... | 51 |
| Figura 79: Bloque de Lectura Digital buzzer activo práctica 1. | 51 |

| | |
|--|----|
| Figura 80: Bloque de Lectura Digital sensor táctil práctica 1..... | 51 |
| Figura 81: Bloque de cierre de comunicación práctica 1..... | 51 |
| Figura 82: Bloque de apertura de conexión de Database práctica 1..... | 51 |
| Figura 83: Bloque para inserta datos en Database práctica 1..... | 52 |
| Figura 84: Bloque de cierre de conexión de Database práctica 1..... | 52 |
| Figura 85: Creación de base de datos en MICROSOFT ACCESS práctica 1..... | 52 |
| Figura 86: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 1..... | 52 |
| Figura 87: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 1..... | 53 |
| Figura 88: Ventana de propiedades de vínculo de datos práctica 1..... | 53 |
| Figura 89: Selección de ruta donde se ubica la base de datos práctica 1..... | 54 |
| Figura 90: Venta de prueba de conexión de la vinculación de datos práctica 1..... | 54 |
| Figura 91: Ventana de asignación de nombre para la tabla de registro de datos práctica 1..... | 54 |
| Figura 92: Control de ruta de base de datos práctica 1..... | 55 |
| Figura 93: Diagrama de conexiones de la Práctica 1..... | 56 |
| Figura 94: Panel Frontal de la Práctica 1..... | 57 |
| Figura 95: Diagrama de Bloques de la Práctica 1..... | 58 |
| Figura 96: Sensor de Flama práctica 2. (Sunfounder, 2013)..... | 61 |
| Figura 97: Buzzer Activo práctica 2. (Sunfounder, 2013)..... | 61 |
| Figura 98: RGB LED práctica 2. (Sunfounder, 2013)..... | 62 |
| Figura 99: Sensor Táctil práctica 2. (Sunfounder, 2013)..... | 62 |
| Figura 100: Bloque de apertura de comunicación práctica 2..... | 63 |
| Figura 101: Bloque de Lectura Analógica sensor de flama práctica 2..... | 63 |
| Figura 102: Bloque de Lectura Analógico SET POINT del sensor de flama práctica 2. | 63 |
| Figura 103: Bloque de Lectura Digital sensor de flama práctica 2..... | 64 |
| Figura 104: Bloque de Escritura Digital buzzer activo práctica 2..... | 64 |
| Figura 105: Bloque de Constante de color práctica 2..... | 64 |
| Figura 106: Bloque conversión Color a RGB práctica 2..... | 64 |
| Figura 107: Bloque de Escritura Digital LED RGB práctica 2..... | 65 |
| Figura 108: Bloque de Lectura Digital sensor táctil práctica 2..... | 65 |
| Figura 109: Bloque de cierre de comunicación práctica 2..... | 65 |
| Figura 110: Bloque de apertura de conexión de Database práctica 2..... | 65 |
| Figura 111: Bloque para inserta datos en Database práctica 2..... | 65 |
| Figura 112: Bloque de cierre de conexión de Database práctica 2..... | 66 |
| Figura 113: Creación de base de datos en MICROSOFT ACCESS práctica 2..... | 66 |
| Figura 114: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 2..... | 66 |
| Figura 115: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 2..... | 67 |
| Figura 116: Ventana de propiedades de vínculo de datos práctica 2..... | 67 |
| Figura 117: Selección de ruta donde se ubica la base de datos práctica 2..... | 68 |
| Figura 118: Venta de prueba de conexión de la vinculación de datos práctica 2..... | 68 |
| Figura 119: Ventana de asignación de nombre para la tabla de registro de datos práctica 2..... | 68 |
| Figura 120: Control de ruta de base de datos práctica 2..... | 69 |
| Figura 121: Diagrama de conexiones de la Práctica 2..... | 70 |
| Figura 122: Panel Frontal de la Práctica 2..... | 71 |

| | |
|--|----|
| Figura 123: Diagrama de Bloques de la Práctica 2..... | 72 |
| Figura 124: Sensor de Flama práctica 3. (Sunfounder, 2013)..... | 75 |
| Figura 125: Sensor de Gas práctica 3. (Sunfounder, 2013) | 75 |
| Figura 126: Buzzer Pasivo práctica 3. (Sunfounder, 2013). | 76 |
| Figura 127: Pulsador Manual práctica 3. (Sunfounder, 2013). | 76 |
| Figura 128: Sensor Táctil práctica 3. (Sunfounder, 2013). | 77 |
| Figura 129: Bloque de apertura de comunicación práctica 3..... | 78 |
| Figura 130: Bloque de Lectura Analógico sensor de gas práctica 3. | 78 |
| Figura 131: Bloque de Lectura Analógico SET POINT del sensor de gas práctica 3. | 78 |
| Figura 132: Bloque de Lectura Digital sensor de gas práctica 3..... | 78 |
| Figura 133: Cálculo de concentración de Gas en PPM..... | 79 |
| Figura 134: Bloque de Lectura Analógica sensor de flama práctica 3. | 79 |
| Figura 135: Bloque de Lectura Analógico SET POINT del sensor de flama práctica 3. | 79 |
| Figura 136: Bloque de Lectura Digital sensor de flama práctica 3..... | 79 |
| Figura 137: Bloque de Lectura Digital pulsador manual práctica 3. | 80 |
| Figura 138: Bloque de Escritura Digital buzzer pasivo práctica 3..... | 80 |
| Figura 139: Bloque de Lectura Digital sensor táctil práctica 3..... | 80 |
| Figura 140: Bloque de cierre de comunicación práctica 3. | 80 |
| Figura 141: Configuración de data socket en VI de Escritura práctica 3..... | 81 |
| Figura 142: Configuración de data socket en VI de Lectura práctica 3..... | 82 |
| Figura 143: Bloque de apertura de conexión de Database práctica 3. | 82 |
| Figura 144: Bloque para inserta datos en Database práctica 3..... | 82 |
| Figura 145: Bloque de cierre de conexión de Database práctica 3. | 83 |
| Figura 146: Creación de base de datos en MICROSOFT ACCESS práctica 3. | 83 |
| Figura 147: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 3. | 83 |
| Figura 148: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 3. | 84 |
| Figura 149: Ventana de propiedades de vínculo de datos práctica 3. | 84 |
| Figura 150: Selección de ruta donde se ubica la base de datos práctica 3. | 85 |
| Figura 151: Venta de prueba de conexión de la vinculación de datos práctica 3. | 85 |
| Figura 152: Ventana de asignación de nombre para la tabla de registro de datos práctica 3. | 85 |
| Figura 153: Control de ruta de base de datos práctica 3. | 86 |
| Figura 154: Diagrama de conexiones de la Practica 3. | 87 |
| Figura 155: Panel Frontal VI de Escritura de la Práctica 3..... | 89 |
| Figura 156: Diagrama de Bloques VI de Escritura de la Práctica 3. | 90 |
| Figura 157: Panel Frontal VI de Lectura de la Práctica 3. | 91 |
| Figura 158: Diagrama de Bloques VI de Lectura de la Práctica 3..... | 91 |
| Figura 159: Emisor Láser práctica 4. (Sunfounder, 2013)..... | 94 |
| Figura 160: Modulo Relé práctica 4. (Sunfounder, 2013). | 94 |
| Figura 161: Esquema de un relé práctica 4. (Sunfounder, 2013)..... | 95 |
| Figura 162: Foto-resistencia práctica 4. (Sunfounder, 2013)..... | 96 |
| Figura 163: Bloque de apertura de comunicación práctica 4. | 96 |
| Figura 164: SubVI Photocell de Linx práctica 4..... | 97 |
| Figura 165: Bloque de Escritura Digital activación del relé práctica 4. | 97 |
| Figura 166: Bloque de cierre de comunicación práctica 4. | 97 |
| Figura 167: Bloque de apertura de conexión de Database práctica 4. | 97 |

| | |
|---|-----|
| Figura 168: Bloque para inserta datos en Database práctica 4..... | 97 |
| Figura 169: Bloque de cierre de conexión de Database práctica 4. | 98 |
| Figura 170: Creación de base de datos en MICROSOFT ACCESS práctica 4. | 98 |
| Figura 171: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 4. | 98 |
| Figura 172: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 4. | 99 |
| Figura 173: Ventana de propiedades de vínculo de datos práctica 4. | 99 |
| Figura 174: Selección de ruta donde se ubica la base de datos práctica 4. | 100 |
| Figura 175: Venta de prueba de conexión de la vinculación de datos práctica 4. ... | 100 |
| Figura 176: Ventana de asignación de nombre para la tabla de registro de datos práctica 4. | 100 |
| Figura 177: Control de ruta de base de datos práctica 4. | 101 |
| Figura 178: Diagrama de conexiones de la Práctica 4. | 102 |
| Figura 179: Panel Frontal de la Práctica 4. | 103 |
| Figura 180: Diagrama de Bloques de la Práctica 4. | 104 |
| Figura 181: Dual Color LED práctica 5. (Sunfounder, 2013)..... | 107 |
| Figura 182: Joystick práctica 5. (Sunfounder, 2013). | 107 |
| Figura 183: LED de flash automático práctica 5. (Sunfounder, 2013). | 108 |
| Figura 184: Bloque de apertura de comunicación práctica 5. | 108 |
| Figura 185: Bloque de Lectura Analógica Eje X Joystick práctica 5. | 108 |
| Figura 186: Bloque de Lectura Analógica Eje Y Joystick práctica 5. | 109 |
| Figura 187: Bloque de Lectura Digital Botón Joystick práctica 5. | 109 |
| Figura 188: Bloque de Escritura Digital color rojo Dual Color LED práctica 5. | 109 |
| Figura 189: Bloque de Escritura Digital color verde Dual Color LED práctica 5. ... | 109 |
| Figura 190: Bloque de Escritura Digital. Auto-Flash LED práctica 5. | 109 |
| Figura 191: Bloque de cierre de comunicación práctica 5. | 109 |
| Figura 192: Bloque de apertura de conexión de Database práctica 5. | 110 |
| Figura 193: Bloque para inserta datos en Database práctica 5..... | 110 |
| Figura 194: Bloque de cierre de conexión de Database práctica 5. | 110 |
| Figura 195: Creación de base de datos en MICROSOFT ACCESS práctica 5. | 110 |
| Figura 196: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 5. | 111 |
| Figura 197: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 5. | 111 |
| Figura 198: Ventana de propiedades de vínculo de datos práctica 5. | 112 |
| Figura 199: Selección de ruta donde se ubica la base de datos práctica 5. | 112 |
| Figura 200: Venta de prueba de conexión de la vinculación de datos práctica 5. ... | 112 |
| Figura 201: Ventana de asignación de nombre para la tabla de registro de datos práctica 5. | 113 |
| Figura 202: Control de ruta de base de datos práctica 5. | 113 |
| Figura 203: Diagrama de conexiones de la Práctica 5. | 115 |
| Figura 204: Panel Frontal de la Práctica 5. | 116 |
| Figura 205: Diagrama de Bloques de la Práctica 5. | 117 |
| Figura 206: Barómetro bmp180 práctica 6. (Sunfounder, 2013). | 120 |
| Figura 207: Bloque Configuración de puerto serial práctica 6. | 120 |
| Figura 208: Bloque Nodo de propiedad VISA práctica 6. | 120 |
| Figura 209: Bloque de Lectura VISA práctica 6. | 120 |
| Figura 210: Bloque de Fract/Exp String To Number Function práctica 6. | 121 |
| Figura 211: Bloque Close de VISA práctica 6. | 121 |

| | |
|---|-----|
| Figura 212: Bloque de apertura de conexión de Database práctica 6. | 121 |
| Figura 213: Bloque para inserta datos en Database práctica 6..... | 121 |
| Figura 214: Bloque de cierre de conexión de Database práctica 6. | 121 |
| Figura 215: Creación de base de datos en MICROSOFT ACCESS práctica 6. | 121 |
| Figura 216: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 6. | 122 |
| Figura 217: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 6. | 122 |
| Figura 218: Ventana de propiedades de vínculo de datos práctica 6. | 123 |
| Figura 219: Selección de ruta donde se ubica la base de datos práctica 6. | 123 |
| Figura 220: Venta de prueba de conexión de la vinculación de datos práctica 6. ... | 123 |
| Figura 221: Ventana de asignación de nombre para la tabla de registro de datos práctica 6. | 124 |
| Figura 222: Control de ruta de base de datos práctica 6. | 124 |
| Figura 223: Diagrama de conexiones de la Práctica 6. | 125 |
| Figura 224: Panel Frontal de la Práctica 6. | 126 |
| Figura 225: Diagrama de Bloques de la Práctica 6. | 127 |
| Figura 226: Sensor Hall práctica 7. (Sunfounder, 2013). | 131 |
| Figura 227: Potenciómetro práctica 7. (Sunfounder, 2013). | 132 |
| Figura 228: Motor DC práctica 7. (García, 2013). | 132 |
| Figura 229: Bloque de apertura de comunicación práctica 7. | 133 |
| Figura 230: Bloque de Lectura Analógico Potenciómetro práctica 7. | 134 |
| Figura 231: Bloque de Escritura Digital PWM Motor DC práctica 7. | 134 |
| Figura 232: Bloque de Lectura Digital Sensor Hall práctica 7. | 134 |
| Figura 233: While Loop con Shift Register práctica 7. | 135 |
| Figura 234: Función Increment práctica 7. | 135 |
| Figura 235: Función Select práctica 7. | 135 |
| Figura 236: Contador de vueltas práctica 7. | 135 |
| Figura 237: Bloque de cierre de comunicación práctica 7. | 136 |
| Figura 238: Diagrama de conexiones de la Práctica 7. | 137 |
| Figura 239: Panel Frontal de la Práctica 7. | 138 |
| Figura 240: Diagrama de Bloques de la Práctica 7. | 139 |
| Figura 241: Servo-Motor práctica 8. (García, 2016). | 142 |
| Figura 242: Diagrama de bloques del servo-motor práctica 8. (García, 2016). | 142 |
| Figura 243: Bloque de apertura de comunicación práctica 8. | 143 |
| Figura 244: Bloque de apertura de comunicación del servo-motor práctica 8. | 144 |
| Figura 245. Bloque de Escritura Digital PWM del servo-motor práctica 8. | 144 |
| Figura 246: Bloque de cierre de comunicación del servo-motor práctica 8. | 144 |
| Figura 247: Bloque de cierre de comunicación práctica 8. | 144 |
| Figura 248: Bloque de apertura de conexión de Database práctica 8. | 145 |
| Figura 249: Bloque para inserta datos en Database práctica 8. | 145 |
| Figura 250: Bloque de cierre de conexión de Database práctica 8. | 145 |
| Figura 251: Creación de base de datos en MICROSOFT ACCESS práctica 8. | 145 |
| Figura 252: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 8. | 146 |
| Figura 253: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 8. | 146 |
| Figura 254: Ventana de propiedades de vínculo de datos práctica 8. | 147 |
| Figura 255: Selección de ruta donde se ubica la base de datos práctica 8. | 147 |
| Figura 256: Venta de prueba de conexión de la vinculación de datos práctica 8. ... | 147 |

| | |
|--|-----|
| Figura 257: Ventana de asignación de nombre para la tabla de registro de datos práctica 8..... | 148 |
| Figura 258: Control de ruta de base de datos práctica 8. | 148 |
| Figura 259: Diagrama de conexiones de la Práctica 8. | 149 |
| Figura 260: Panel Frontal de la Práctica 8. | 150 |
| Figura 261: Diagrama de Bloques de la Práctica 8. | 151 |
| Figura 262: Sensor de evasión de obstáculos práctica 9. (Sunfounder, 2013)..... | 154 |
| Figura 263: Micro-Switch de palanca larga práctica 9. | 154 |
| Figura 264: Micro-Switch roller práctica 9..... | 154 |
| Figura 265: Motor DC práctica 9. (García, 2013)..... | 155 |
| Figura 266: Bloque de apertura de comunicación práctica 9..... | 156 |
| Figura 267: Bloque de Lectura Digital Micro-Switch Roller práctica 9..... | 156 |
| Figura 268: Bloque de Lectura Analógica Sensor de obstáculos práctica 9. | 157 |
| Figura 269: Bloque de Lectura Digital Sensor de Obstáculos práctica 9. | 157 |
| Figura 270: Bloque de escritura PWM motor de nivel práctica 9..... | 157 |
| Figura 271: Bloque de Escritura Digital motor de nivel práctica 9. | 158 |
| Figura 272: Animación de la variable Botella práctica 9..... | 158 |
| Figura 273: Animación de la variable Llena práctica 9. | 158 |
| Figura 274: Bloque de Escritura Digital motor de nivel práctica 9. | 158 |
| Figura 275: Bloque de Escritura Digital motor de nivel práctica 9. | 158 |
| Figura 276: Bloque de escritura digital activación motor inyector práctica 9. | 158 |
| Figura 277: Movimiento ascendente del inyector práctica 9. | 159 |
| Figura 278: Bloque de escritura digital desactivación motor inyector práctica 9... | 159 |
| Figura 279: Animación de la variable Tapas práctica 9..... | 159 |
| Figura 280: Movimiento ascendente del colocador de tapas práctica 9..... | 159 |
| Figura 281: Bloque de Lectura Digital Sensor de Obstáculos práctica 9. | 159 |
| Figura 282: Animación de la variable Botellas2 práctica 9. | 160 |
| Figura 283: Bloque de escritura PWM motor de nivel práctica 9..... | 160 |
| Figura 284: Bloque de Escritura Digital motor de nivel práctica 9. | 160 |
| Figura 285: Contador de botellas práctica 9. | 160 |
| Figura 286: Bloque de Lectura Digital Micro-Switch Roller práctica 9..... | 160 |
| Figura 287: Bloque de escritura PWM motor de nivel práctica 9..... | 161 |
| Figura 288: Bloque de Escritura Digital motor de nivel práctica 9. | 161 |
| Figura 289: Bloque de cierre de comunicación práctica 9..... | 161 |
| Figura 290: Bloque de apertura de conexión de Database práctica 9. | 161 |
| Figura 291: Bloque para inserta datos en Database práctica 9..... | 162 |
| Figura 292. Bloque de cierre de conexión de Database práctica 9. | 162 |
| Figura 293: Creación de base de datos en MICROSOFT ACCESS práctica 9. | 162 |
| Figura 294: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 9. | 162 |
| Figura 295: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 9. | 163 |
| Figura 296: Ventana de propiedades de vínculo de datos práctica 9. | 163 |
| Figura 297: Selección de ruta donde se ubica la base de datos práctica 9. | 164 |
| Figura 298: Venta de prueba de conexión de la vinculación de datos práctica 9. ... | 164 |
| Figura 299: Ventana de asignación de nombre para la tabla de registro de datos práctica 9..... | 164 |
| Figura 300: Control de ruta de base de datos práctica 9. | 165 |
| Figura 301: Diagrama de conexiones de la Práctica 9. | 166 |
| Figura 302: Panel Frontal de la Práctica 9. | 167 |

| | |
|---|-----|
| Figura 303: Diagrama de Bloques de la Práctica 9. | 168 |
| Figura 304: Módulo de sensor de temperatura práctica 10. (Sunfounder, 2013)..... | 171 |
| Figura 305: Modulo Relé doble práctica 10. (Sunfounder, 2013). | 171 |
| Figura 306: Esquema de un relé práctica 10. (Sunfounder, 2013)..... | 172 |
| Figura 307: Creación de variables en el Matlab..... | 173 |
| Figura 308: Tabla de datos de variable de entrada denominada in. | 173 |
| Figura 309: Tabla de datos de variable de salida denominada out. | 174 |
| Figura 310: Ventana de herramienta Ident. | 174 |
| Figura 311: Introducción de datos..... | 175 |
| Figura 312: Selección de polos y ceros para la función de transferencia. | 175 |
| Figura 313: Obtención de valores de la función de transferencia. | 176 |
| Figura 314: Modelo resultante. | 176 |
| Figura 315: Parámetros para la función de transferencia. | 177 |
| Figura 316: Parámetros de PID. | 177 |
| Figura 317: Bloque de apertura de comunicación práctica 10. | 178 |
| Figura 318: Bloque de Lectura Analógica sensor LM35 práctica 10. | 178 |
| Figura 319: Bloque de Escritura Digital PWM para giro de ventilador práctica 10. | 178 |
| Figura 320: SubVI PID. | 179 |
| Figura 321: Bloque de Escritura Digital PWM intensidad lumínica del foco halógeno práctica 10. | 179 |
| Figura 322: Bloque de cierre de comunicación práctica 10. | 179 |
| Figura 323: Bloque de apertura de conexión de Database práctica 10. | 179 |
| Figura 324: Bloque para inserta datos en Database práctica 10. | 180 |
| Figura 325: Bloque de cierre de conexión de Database práctica 10. | 180 |
| Figura 326: Creación de base de datos en MICROSOFT ACCESS práctica 10. | 180 |
| Figura 327: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 10. | 180 |
| Figura 328: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 10. | 181 |
| Figura 329: Ventana de propiedades de vínculo de datos práctica 10. | 181 |
| Figura 330: Selección de ruta donde se ubica la base de datos práctica 10. | 182 |
| Figura 331: Venta de prueba de conexión de la vinculación de datos práctica 10. . | 182 |
| Figura 332: Ventana de asignación de nombre para la tabla de registro de datos práctica 10. | 182 |
| Figura 333: Control de ruta de base de datos práctica 10. | 183 |
| Figura 334: Diagrama de conexiones de la Práctica 10. | 184 |
| Figura 335: Panel Frontal de la Práctica 10. | 185 |
| Figura 336: Diagrama de Bloques de la Práctica 10. | 186 |

ÍNDICE DE TABLAS

| | |
|--|-----|
| Tabla 1: Características técnicas de tarjeta Arduino MEGA 2560. | 12 |
| Tabla 2: Conexiones de la Práctica 1. | 57 |
| Tabla 3: Conexiones de la Práctica 2. | 71 |
| Tabla 4: Conexiones de la Práctica 3. | 88 |
| Tabla 5: Conexiones de la Práctica 4. | 103 |
| Tabla 6: Conexiones de la Práctica 5. | 116 |
| Tabla 7: Conexiones de la Práctica 6. | 125 |
| Tabla 8: Conexiones de la Práctica 7. | 138 |
| Tabla 9: Conexiones de la Práctica 8. | 149 |
| Tabla 10: Conexiones de la Práctica 9. | 167 |
| Tabla 11: Conexiones de la Práctica 10. | 185 |

ÍNDICE DE ECUACIONES

| | |
|---|-----|
| Ecuación 1: Concentración de Gas en PPM práctica 1..... | 50 |
| Ecuación 2: Ecuación de la recta práctica 1..... | 50 |
| Ecuación 3: Conversión de color Rojo práctica 2..... | 64 |
| Ecuación 4: Conversión de color Verde práctica 2..... | 64 |
| Ecuación 5: Conversión de color Azul práctica 2..... | 65 |
| Ecuación 6: Concentración de Gas en PPM práctica 3..... | 78 |
| Ecuación 7: Ecuación de la recta práctica 3..... | 78 |
| Ecuación 8: Fuerza de Lorents práctica 7..... | 133 |
| Ecuación 9: Ecuación conversión Voltaje a PWM práctica 7..... | 134 |
| Ecuación 10: Ecuación de la recta conversión PWM a RPM Aprox. práctica 7..... | 134 |
| Ecuación 11: Grados de posición del servo-motor práctica 8..... | 144 |
| Ecuación 12: Fuerza de Lorents práctica 9..... | 155 |
| Ecuación 13: Ecuación de la recta práctica 9..... | 157 |
| Ecuación 14: Voltaje de salida del sensor LM35 práctica 10..... | 171 |

INTRODUCCIÓN

El tema del presente Proyecto de Titulación es “Diseño e implementación de un sistema de adquisición de datos usando el kit de sensores Sunfounder, mediante el uso del hardware LattePanda y Arduino con interfase gráfica desarrollada en LabVIEW”, el cual consiste en la implementación a escala de un módulo de entrenamiento que permite realizar la adquisición de datos de varios sensores utilizados en el medio Ingeniería Electrónica, utilizando en este caso en el kit de sensores de marca Sunfounder; garantizando la confiabilidad del sistema, comunicación, lectura y adquisición de información de los sensores. El módulo permite el análisis de conceptos, fundamentos técnicos y elementos principales que intervienen en la materia de Sensores y Transductores.

Para el desarrollo del proyecto fue necesaria la segmentación de cuatro apartados fundamentales.

- Se inicia con la definición del problema principal, dando lugar al propósito del proyecto, donde se descubre el alcance y sus beneficios hacia la sociedad, plasmando así los objetivos principales y específicos.
- Se define el marco teórico recopilando la información más relevante sobre la investigación, citando textos de autores de los últimos años.
- Se establece el proceso de diseño, construcción y montaje del módulo, instalación de los softwares controladores; junto con sus soportes fotográficos.
- Finalmente, se elabora un manual con las prácticas más importantes de este módulo.

La importancia del proyecto se enfoca en suministrar a la carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana sede Guayaquil, un módulo de laboratorio para el estudio de la materia de Sensores y Transductores, dando como resultado que los estudiantes puedan desarrollar un criterio práctico del manejo de los sensores y que sepan cómo realizar las configuraciones básicas en la programación, aportando de esta manera una visión amplia dentro de este campo de la Ingeniería Electrónica.

1. EL PROBLEMA

1.1. Antecedentes

La Universidad Politécnica Salesiana, Sede Guayaquil, es una entidad con especializaciones en las ramas científico-técnicas (Martillo & Zambrano, 2015).

Las materias que conforman la malla de la carrera de Ingeniería Electrónica se encuentran en constante actualización debido a las nuevas tecnologías desarrolladas, las mismas son aplicables a sus funciones en los diferentes campos externos.

Es importante que, a los estudiantes de la carrera de Ingeniería Electrónica, se les facilite el manejo de equipos para la elaboración de prácticas que afiancen los conocimientos teóricos impartidos, además facilitar el desarrollo de proyectos utilizando sistemas que empleen tecnologías actuales vigentes y de libre desarrollo (Martillo & Zambrano, 2015).

En vista de las tendencias de desarrollo se propone este proyecto, que lo componen sistemas embebidos que al tener la capacidad de adquisición de datos y ser didácticos, permite que los estudiantes tengan más opciones al querer implementar sus prácticas de laboratorio, el mismo es un sistema que cuenta con mayor simplicidad, desempeño, flexibilidad y menor costo (Orrellana, 2015).

1.2. Importancia y Alcances

El desarrollo del proyecto, significa un aporte a la Carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana sede Guayaquil, dotándola de un módulo de laboratorio para el estudio de sensores y transductores, dando como resultado que los estudiantes puedan desarrollar un criterio práctico del manejo de los sensores y que sepan cómo realizar las configuraciones básicas en la programación, aportando de esta manera una visión amplia dentro de este campo de la Ingeniería Electrónica.

El alcance del proyecto se limita al estudio de los elementos que conforman el kit de sensores Sunfounder, se busca ampliar las opciones de equipos disponibles para el aprendizaje de la carrera, de manera que los estudiantes pongan en práctica los conocimientos facilitados por parte del docente.

Dentro de los beneficios que se obtienen al resolver esta problemática, están la experimentación de los problemas y planteamiento de soluciones propias para eventos que se suscitan con frecuencia en la cotidianidad de los profesionales que trabajan en el área de Ingeniería Electrónica en Sistemas Industriales, complementando así los conocimientos recibidos de manera teórica en las asignaturas de la carrera, con el desarrollo de prácticas orientadas a un modelo real.

1.3. Delimitación

1.3.1. Temporal

La implementación de este proyecto se realizó entre Marzo del 2017 y Septiembre del 2018.

1.3.2. Espacial

El presente proyecto técnico, está dirigido a la Universidad Politécnica Salesiana sede Guayaquil en el año 2018, para la carrera de Ingeniería Electrónica y sus estudiantes, dotando de una herramienta para la materia de Sensores y Transductores.

1.3.3. Académica

El proyecto está diseñado para trabajar con el kit de sensores Sunfounder, especializados en la medición de diferentes parámetros, a su vez, esta no es una limitante pues la tarjeta Arduino MEGA 2560 posee una gran compatibilidad con diferentes tipos de sensores lo que permite una potenciación del proyecto. Por esta

razón se elaboraron 10 prácticas de laboratorio utilizando los sensores considerados de mayor utilidad para el aprendizaje de la materia dando como resultado la elaboración de 4 módulos didácticos.

La programación que se realiza en el proyecto técnico está basada en la programación por bloques del software de desarrollo LabVIEW 2016 y su herramienta LINX de Digilent/LabVIEW MakerHub, el cual, a través de la interfaz VISA 2016, permite la comunicación entre el sistema embebido LattePanda y la tarjeta Arduino MEGA 2560.

El proyecto técnico se encuentra dotado de una pantalla táctil de 7" que cumple la función de interfaz HMI, la misma que permite a los usuarios interactuar con el programa desarrollado en el software LabVIEW 2016.

1.4. Objetivos

1.4.1. Objetivo general

Diseñar e implementar un sistema de adquisición de datos usando el kit de sensores de temperatura, infrarrojo y Hall marca Sunfounder, mediante el uso de los sistemas embebidos LattePanda y Arduino Mega 2560, a esto sumando una interfaz gráfica desarrollada en el software LabVIEW 2016.

1.4.2. Objetivos específicos

- Diseñar un sistema SCADA con interfaz gráfica desarrollada en LabVIEW 2016 para la visualización del comportamiento de los sensores.
- Adquirir datos a través de las entradas/salidas, analógicas y digitales disponibles de los sistemas embebidos LattePanda y Arduino Mega 2560, para el desarrollo prácticas de laboratorio.
- Elaborar una guía completa de 10 prácticas didácticas en las cuales se desarrolle de manera gradual el manejo y aplicación de los sensores.

- Emplear los sistemas embebidos LattePanda y Arduino Mega 2560 con el software LabVIEW 2016, como un conjunto de trabajo teniendo así una herramienta de desarrollo para futuras aplicaciones prácticas en la materia de sensores y transductores de la carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana Sede Guayaquil.
- Elaborar un módulo utilizando los sensores Sunfounder de tal manera que los elementos estén en sitios estratégicos para dar facilidad al momento de implementar las prácticas.
- Proveer de nuevas herramientas que permitan desarrollar e implementar los proyectos, poniendo en práctica los conocimientos adquiridos en las clases impartidas por los docentes.

1.5. Marco Metodológico

En el proyecto se ponen en práctica varios métodos de la investigación. Se utiliza el método analítico para fragmentar el tema de estudio en varias partes y poder observar los principios de funcionamiento, comportamiento y los efectos de cada sensor; gracias a este método se pudo comprender mejor el comportamiento del sistema y establecer ciertas teorías.

La experimentación que se realiza durante el desarrollo del proyecto hace posible modificar las condiciones de operación de los sensores para estudiarlos como fenómenos que no se presentan dentro de parámetros normales; cabe recalcar y establecer que el objetivo de este método permite establecer una relación que incluye a las variables analizadas, en la misma, la causas deben ser asignadas a las variables independientes y los efectos se hallarán en las variables dependientes.

El método sintético se empleó con la finalidad de poder reconstruir el todo y volver a integrar las partes fragmentadas por el análisis en una perspectiva de totalidad; con este método se pudo lograr la comprensión de la esencia del proyecto y sus particularidades. El método inductivo permitió pasar de los resultados obtenidos en

experimentos al planteamiento de teorías que generalizan aquellos resultados; las conclusiones obtenidas a partir de la inducción tienen un carácter probable.

2. Fundamentos teóricos

2.1. Hardware

2.1.1. LattePanda

LattePanda es un mini-ordenador que ofrece un hardware potente y una serie de características adicionales respecto a las demás tarjetas de su clase cumpliendo de manera perfecta las necesidades de los usuarios (Velasco, 2016). Este sistema embebido es capaz de ejecutar programas de ofimática tales como Microsoft Office, Open Office, reproducir vídeos en calidad HD y desarrollar aplicaciones convencionales de Windows igual que en cualquier otro ordenador, esto se debe a que LattePanda no está usando un procesador ARM o Edison: es un procesador Cherry Trail Atom de cuatro núcleos, con 2GB o 4GB de memoria, y un co-procesador ATmega32u4 que maneja las características compatibles con Arduino, junto con 20 GPIO (conexiones para sus sensores y actuadores) (Velasco, 2016).

Si bien se puede usar un monitor con entrada HDMI para poder visualizar el entorno de la LattePanda, también hay la opción de adaptar una pantalla de 7 pulgadas de bajo costo con una superposición táctil que se conecta directamente a la placa. Eso hace que LattePanda sea una opción más interesante, ya que permite construir en un interfaz táctil de bajo costo (Bisson, 2016).

La clave de LattePanda es el coprocesador incorporado compatible con Arduino. Así es como se maneja el trabajo con hardware GPIO. Costa de un bloque conector, que maneja la mayoría de las conexiones IP, a su vez consta de 6 bloques de 3 pines, para utilizar los kits de sensores Gravity de DFRobot. La capacidad de comunicarse con un kit de sensores permite el desarrollo de aplicaciones, ya que otorga una ruta rápida para trabajar con sensores comunes (Bisson, 2016).

Se encuentran las herramientas de desarrollo de Arduino incluidas en LattePanda, incluyendo, códigos de ejemplo que puede compilar y ejecutar rápidamente. También existe la opción de usar el soporte Arduino de Microsoft (incluyendo sus herramientas de depuración remota) para trabajar con LattePanda en Visual Studio. El código debe ser cargado en el software Arduino de la tarjeta usando una conexión en serie. Ese es un enfoque familiar para cualquier persona que utilice Arduino a través de USB, y al menos aquí no hay necesidad de configurar los puertos serie - puede cargar el código tan rápido como desee (Bisson, 2016).

El desarrollo de Visual Studio puede aprovechar el soporte de cableado remoto, por lo que puede incrustar código Arduino en sus proyectos y generar un código, capaz de aprovechar ambos lados de LattePanda: una interfaz de usuario de Windows y un sensor Arduino y un código de actuador. No es una solución perfecta, ya que está limitado a una conexión en serie entre las dos mitades de la placa. Hay dos puertos Atom GPIO, pero la mayor parte de la codificación tiene que ser manejada mediante código, en el lado de Windows de LattePanda y luego usando las herramientas de despliegue en Visual Studio o en el IDE de Arduino para cargar código a El Arduino y luego ejecutarlo por separado de la plataforma Windows. Mantener Windows y Arduino separados tiene sus ventajas.

Si bien no se obtiene acceso directo a GPIO con Windows 10 IoT Core y Raspberry Pi, se obtiene una plataforma de desarrollo combinada y un dispositivo IoT. Eso es un gran triunfo, ya que se puede escribir código en el LattePanda, cargarlo al Arduino, depurar usando las herramientas de depuración en serie de Windows, y luego hacer y cargar los cambios sobre la marcha (Bisson, 2016).

Usar LattePanda como una plataforma de desarrollo IoT tiene mucho sentido (Bisson, 2016). Aunque Atom no tenga el poder de manejar una instalación completa de Visual Studio, puede elegir los elementos que desea aprovechar de sus características de desarrollo y depuración de Arduino. Como alternativa, ya que está ejecutando una instalación completa de Windows, podría usar el propio IDE de Arduino (que está incluido como parte de la imagen de instalación de LattePanda) y su elección del editor de los programadores (Bisson, 2016).

LattePanda permite comunicarse prácticamente cualquier dispositivo, gracias a su flexibilidad se puede usar desde mandos de control hasta sensores, webcams, pantallas táctiles, etc, gracias a sus periféricos (Velasco, 2016).

Algunas características que ofrece el sistema embebido LattePanda son:

- La capacidad de operar como un ordenador convencional.
- Su facilidad en el uso de la robótica.
- Capaz de controlar sistemas de seguridad, con opción a reconocimiento facial.
- Posee una memoria gráfica que permite utilizarlo para diseño gráfico (incluyendo diseños en 3D).
- Ideal para el análisis y recolección de datos.

Especificaciones técnicas:

- Procesador integrado Quad Core 1.8Ghz
- Memoria RAM DDR3 de 2GB de capacidad
- Memoria interna de 32 GB (expandible mediante tarjeta Micro-SD)
- Memoria gráfica Intel Gen8
- Co-procesador Arduino integrado
- Una entrada USB 3.0 y 2 entradas USB 2.0
- Conectividad mediante tarjeta de red Fast Ethernet
- Antena Wi-Fi externa y Bluetooth 4.0 integrado



Figura 1: Sistema embebido LattePanda 2G/32G. (DFRobots, 2016).

Cabe mencionar que este sistema embebido viene pre-instalado por defecto una versión de Windows 10, lo que simplifica la puesta en marcha para los usuarios.

2.1.2. Pantalla IPS de 7"

La pantalla IPS (In-Plane Switching) es un panel LCD clásico que contiene una capa interna de cristales líquidos que evita las pérdidas de luz, ya que estas pérdidas producen degradación de la calidad de imagen y la intensidad de colores, en especial la definición de los oscuros (Oyanedel, 2014).

LattePanda agrega nuevas dimensiones a sus proyectos con esta gran pantalla a color.

Solo requiere una única conexión al sistema embebido LattePanda y tiene una interfaz de usuario visual para su proyecto.



Figura 2: Pantalla IPS de 7". (DFRobot, 2016).

Especificaciones técnicas:

- Pantalla IPS de 7 "
- 1024 x 600 de alta resolución
- Dimensiones: 164.0 mm x 96.0 mm x 2.6 mm
- Compatible con LattePanda
- Conector de 10 cm FPC

2.1.3. Pantalla Capacitiva de 7"

Por su tecnología, el cristal de la pantalla está compuesto por óxido de indio y estaño, por lo que al tocarla se genera una corriente eléctrica que pasa por esta capa hasta un sensor de tal manera que, como cualquier capacitor, se carga con una determinada cantidad de electrones (Baccaglioni, 2009). La conductividad eléctrica de la piel humana permite variar la capacitancia y generará una distorsión, de tal manera que al medir esta distorsión se puede ubicar exactamente la ubicación del dedo humano en la pantalla (Baccaglioni, 2009). A su vez, es capaz de detectar varias pulsaciones, lo que permite diferentes formas de actuar con ellas, aumentando su capacidad de control.

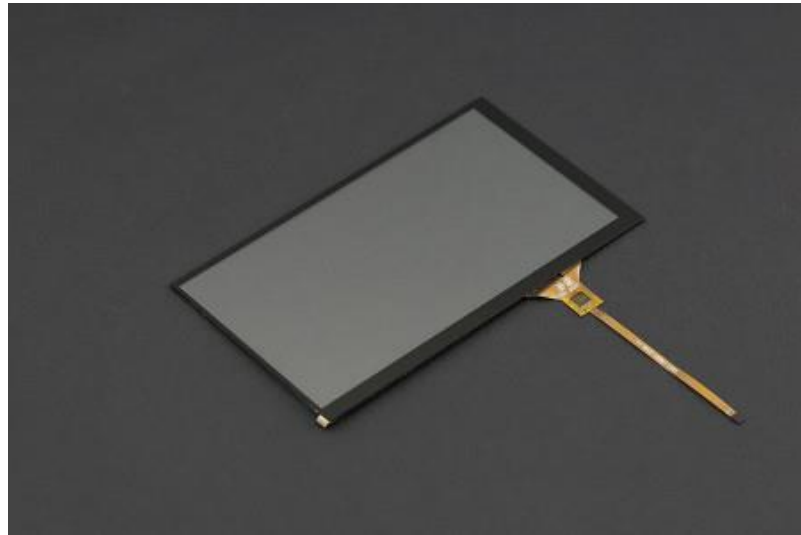


Figura 3: Pantalla Capacitiva de 7". (DFRobots, 2016).

Especificaciones técnicas:

- Pantalla táctil capacitiva
- Transmitancia: 85%
- Dureza superficial: 6H
- Tiempo de respuesta: ≤ 16 ms
- Aprovechar la durabilidad: 200,000,000 veces
- Compatible con LattePanda
- Cable de extensión de 10cm FPC
- Dimensiones: 164.0 mm x 96.0 mm

2.1.4. Arduino

Arduino es una compañía italiana que diseña dispositivos electrónicos desarrollados en código abierto (open-source), de tal manera que sus equipos y software de programación son flexibles y fáciles de combinar con su ambiente de desarrollo, facilitando así el uso de los dispositivos en proyectos electrónicos (Valenzuela, 2012; Herrador, 2009).

El hardware está formado por una placa controlada por un microcontrolador programable Atmel AVR y puertos de entradas/salidas digitales y analógicas (Crespo, 2015). Los microcontroladores más comunes son el Atmega168, Atmega328, Atmega1280, ATmega8 debido a su bajo costo de producción y su sencillez que facilitan el desarrollo de varias versiones de placas Arduino (Crespo, 2015). Por otro lado, el microcontrolador integrado en la placa es programado mediante los softwares libres Arduino Programming Language y el Arduino Development Environment basados en Wiring y Processing respectivamente (Crespo, 2015).

Los proyectos desarrollados con las placas Arduino pueden funcionar de manera autónoma o se pueden combinar con otros softwares en ejecución en un computador (Herrador, 2009).



Figura 4: Logo oficial de Arduino. (Arduino, 2013).

Arduino es capaz de desarrollar proyectos interactivos, autónomos o puede comunicarse a través de un software del computador, por ejemplo: LabVIEW, Macromedia Flash, MATLAB, entre otros (Crespo, 2015). Al ser un hardware libre, es capaz de desarrollar cualquier tipo de proyecto sin la necesidad de adquirir una licencia pagada (Crespo, 2015).

Arduino puede obtener información de su entorno a través de sus entradas, digitales y analógicas, de toda una gran variedad de sensores y a su vez es capaz de controlar

aquello que le rodea por medio de sus salidas, digitales y analógicas, controlando luces, motores y otros actuadores (Herrador, 2009).

2.1.5. Arduino Mega 2560

El Arduino Mega 2560 es una tarjeta electrónica controlada por el microcontrolador más potente de 8 bits ATmega2560 y el que más pines de entrada y salidas tiene (Crespo, 2015).

Formada por 54 Entradas /Salidas Digitales, de las cuales 14, tienen la capacidad de ser utilizadas como salidas PWM, además de 16 entradas analógicas, 4 puertos seriales UARTs, un cristal de 16MHz, una conexión USB Jack tipo B, un conector de alimentación 7-12V, una entrada ICSP de programación serial y un botón de Reset (Arduino, 2018).

Arduino Mega 2560 tiene todo lo necesario para soporte del microcontrolador ATmega2560, solo hace falta conectarlo a un computador mediante un cable USB, o alimentarlo a través de un adaptador de corriente AC a DC 7-12 V para empezar a utilizarlo (Arduino, 2018).

Tabla 1: Características técnicas de tarjeta Arduino MEGA 2560.

| Características | Descripción |
|------------------------------------|--------------------------|
| Micro Controlador | ATmega2560 |
| Voltaje de Operación | 5V |
| Voltaje (In) | 7-12V |
| Recomendado | |
| Límite de Voltaje (In) | 6-20V |
| Entradas/ Salidas Digitales | 54 (14 como salidas PWM) |
| Entradas Análogas | 16 |
| Corriente por Pin In/Out | 40 Ma |

Corriente Para el Pin

50 Ma

3.3V

Memoria Flash

256 KB (Solo 8 KB son usados por la secuencia de arranque “bootloader”)

SRAM

8 KB (Dos bancos: 64 KB y 32KB)

EEPROM

4 KB

Velocidad de Reloj

16 MHz

Nota: Características técnicas de tarjeta Arduino MEGA 2560. (Arduino, 2013), recuperado de: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>



Figura 5: Tarjeta Arduino MEGA 2560. (Arduino, 2013).

2.1.6. Sunfounder

Fundada en 2012, SunFounder es una empresa de tecnología comprometida con el desarrollo de productos de educación en código abierto. Entre la variedad de equipos desarrollados se incluyen hardware de código abierto, robótica y el desarrollo de drones. Se ha invertido mucha energía en el diseño de productos en etapas iniciales tales como sensores y sistemas embebidos para que los estudiantes puedan aprender más sobre el código abierto (Sunfounder, 2012).

El Kit de sensores V2.0 para Arduino es una revisión basada en la versión anterior. SunFounder ha rediseñado los sensores de tal manera que los componentes de este kit se proporcionan en forma de módulos que integran algunos componentes necesarios,

como comparadores, resistencias y condensadores, entre otros. Por lo tanto, es conveniente para la conexión de circuito. Ellos emiten señales directamente a la placa Arduino (Sunfounder, 2013).



Figura 6: Logo oficial de Sunfounder. (Sunfounder, 2013).

2.2. Software

2.2.1. LabVIEW

LabVIEW es un entorno de programación diseñado para mejorar la productividad y alcance de ingenieros y científicos (National Instruments, 2016). Con un tipo de programación gráfica que facilita al usuario la visualización, creación y codificación de sistemas de ingeniería, LabVIEW es una fuente de ayuda para que los ingenieros puedan convertir sus ideas en realidad, reduciendo tiempos de pruebas y ofreciendo análisis de información basado en datos recolectados (National Instruments, 2016).

Compuesto por una recopilación sin precedentes de hardware de medidas, software legado de sus versiones anteriores e IP para obtener una tecnología de punta (National Instruments, 2016).

LabVIEW ha sido por décadas la solución eficaz para crear, implementar y probar el Internet de las cosas, se lo ha utilizado desde el desarrollo de máquinas inteligentes hasta garantizar el correcto funcionamiento de los dispositivos conectados (National Instruments, 2016).



Figura 7: Logo LabVIEW. (National Instruments, 2016).

Aplicaciones:

- **Adquisición de Datos y Procesar Señales:** Es capaz de adquirir información de cualquier sensor en cualquier bus, realizar el análisis y procesamiento de señales avanzadas, la visualización de datos en interfaces personalizadas por los usuarios y el registro de datos y generación de reportes (National Instruments, 2016).
- **Control de Instrumentos:** Automatiza la colección de datos, controla múltiples instrumentos y el análisis y visualización de señales (National Instruments, 2016).
- **Automatizar Sistemas de Pruebas y Validación:** Permite automatizar las pruebas de validación y producción de productos, control de múltiples instrumentos de manera simultánea y el análisis y visualización de resultados de pruebas con interfaces personalizadas por los usuarios (National Instruments, 2016).
- **Sistemas Embebidos de Monitoreo y Control:** Es capaz de reutilizar código ANSI C y HDL, integrar hardware comercial, generar prototipos con tecnología FPGA y permite acceso a herramientas personalizadas para medicina, robótica y más (National Instruments, 2016).
- **Enseñanza Académica:** Utiliza un enfoque práctico e interactivo de fácil aprendizaje combinando el diseño de algoritmos con información recopilada de datos reales, aumenta el rendimiento de aplicaciones con procesamiento multinúcleo (National Instruments, 2016).

2.2.2. Toolkit LINX de Digilent/LabVIEW MakerHub

El toolkit LINX es una herramienta de fuente abierta de Digilent y diseñada para facilitar el uso y desarrollo de aplicaciones embebidas usando LabVIEW (National Instruments, 2016).

LINX incluye VI's ejemplares sobre los sensores embebidos más comunes, así como aplicaciones de programación de hardware agnóstico para tener acceso a periféricos como Entradas/Salidas digitales, Entradas/Salidas analógicas, PWM, I2C, SPI y UART (National Instruments, 2016).

Ya sea que esté controlando por un hardware remoto como un chipKIT o Arduino por USB/Serial, Ethernet o Wi-Fi o implementando VIs para ejecutar en plataformas como Raspberry Pi 2/3, LINX y LabVIEW hacen más sencilla la visualización de los datos con los que está trabajando, depurando su código y creando aplicaciones embebidas avanzadas de manera rápida (Stanton, 2017).



Figura 8: Logo Linx. (National Instruments, 2016).

2.2.3. NI VISA

NI-VISA Virtual Instrument Software Architecture (VISA), es un software de Interfaz de Entradas/Salidas Universal, este software forma un modelo para configuración, programación y depuración de sistemas de instrumentación compuestos de interfaces de comunicación GPIB, VXI, PXI, serial (RS232/RS485), Ethernet/LXI y/o interfaces USB (National Instruments, 2016).

NI-VISA es la implementación de National Instruments del estándar de Entradas/Salidas VISA. Proporciona la interfaz que hace posible comunicación entre el hardware y los softwares de diseño de sistemas como NI LabVIEW, LabWindows™/CVI y Measurement Studio para Microsoft Visual Studio (National Instruments, 2016).



Figura 9: NI VISA SOFTWARE. (National Instruments, 2016).

2.2.4. DataSocket

DataSocket es una herramienta de programación desarrollada por LabVIEW, fácil de usar y de alto rendimiento, diseñada específicamente para transferencia de datos entre diferentes aplicaciones y dispositivos a través de la red (National Instruments, 2016).

DataSocket permite simplificar el intercambio de datos entre diferentes aplicaciones en un mismo terminal o entre varios ordenadores conectados a través de una red (National Instruments, 2016).

2.2.5. Datalogging and Supervisory Control Module.

El Módulo de registro de datos y control de supervisión (DSC) es un complemento del software LabVIEW, que permite desarrollar sistemas HMI/SCADA o aplicaciones de registro de datos recopilados de varios canales (National Instruments, 2016).

Gracias a las funciones de este módulo, se puede desarrollar un sistema de control y monitoreo distribuido, de manera interactiva (National Instruments, 2016).

Las herramientas incluidas hacen posible la creación de un registro de información en una base de datos histórica en red, que permite monitorear tendencias en los datos en

tiempo real, administrar alarmas y eventos, a su vez permite crear redes de objetivos LabVIEW Real-Time y dispositivos OPC en un sistema completo y añadir seguridad a interfaces de usuarios (National Instruments, 2016).

2.2.6. Arduino Software

Es un entorno de programación fácil de usar y sumamente flexible permitiendo a los usuarios aprovechar todas sus funciones, el software Arduino es proporcionado como un software libre, disponible para cualquier usuario (Herrador, 2009).

El microcontrolador integrado en la placa es programado a través del lenguaje de programación basado en Wiring y el entorno de desarrollo basado en Processing (Doutel, 2017). El software Arduino se programa mediante el uso de un lenguaje propio, el cual puede ser expandido mediante librerías C++ (Herrador, 2009).

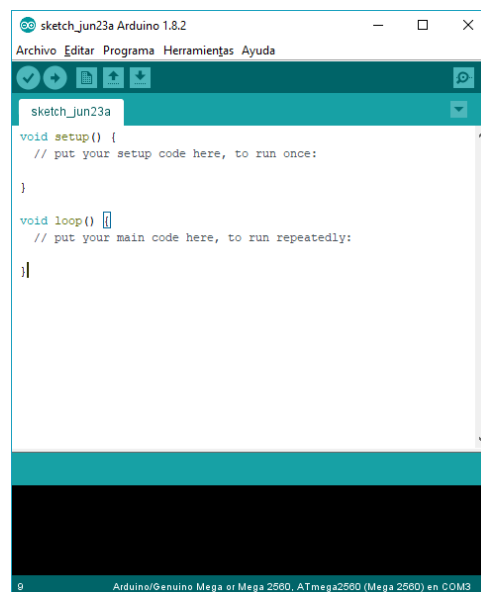


Figura 10: Arduino Software.

3. MARCO METODOLÓGICO

3.1. Diseño de planos en impresora 3D

Para montar los elementos de control, sean estos los sistemas embebidos, pantalla táctil y sensores que necesiten una estructura para facilitar su uso, fue necesario la elaboración de bases para cada elemento, las cuales fueron hechas a través de una impresora 3D.

A continuación, se muestran los diseños de cada base, que posteriormente fueron ingresados a la impresora 3D, para su fabricación:

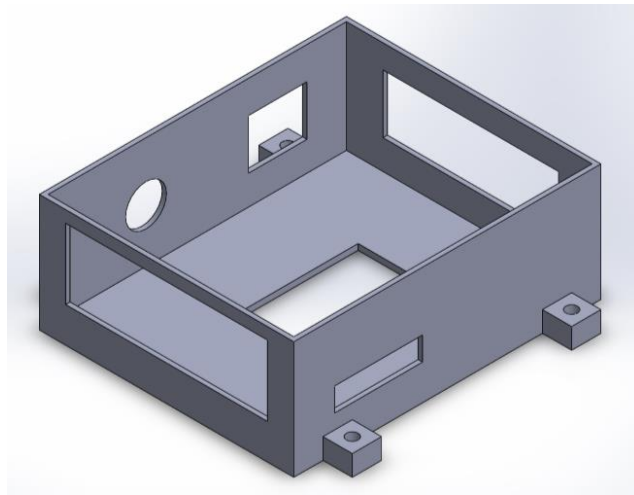


Figura 11: Diseño para impresora 3D de la base del sistema embebido LattePanda.

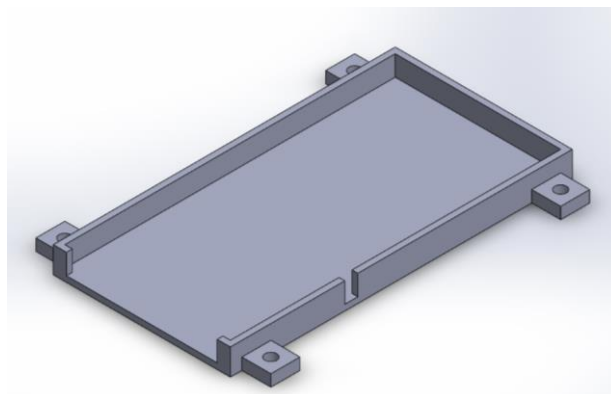


Figura 12: Diseño para impresora 3D de la base del sistema embebido Arduino Mega 2560.

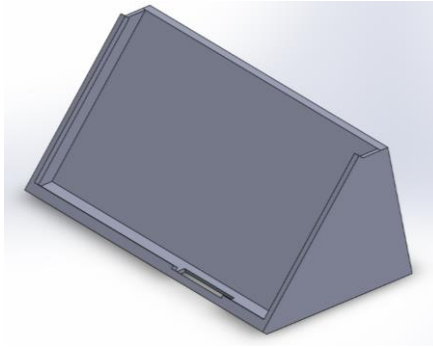


Figura 13: Diseño para impresora 3D de la base de la pantalla táctil.

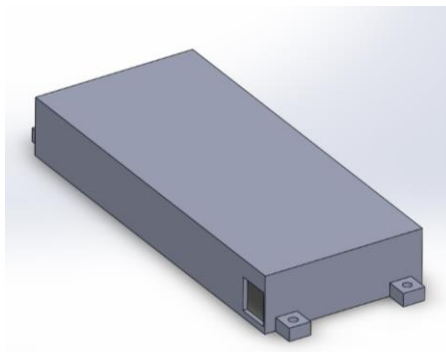


Figura 14: Diseño para impresora 3D de la base de la batería de 12V.

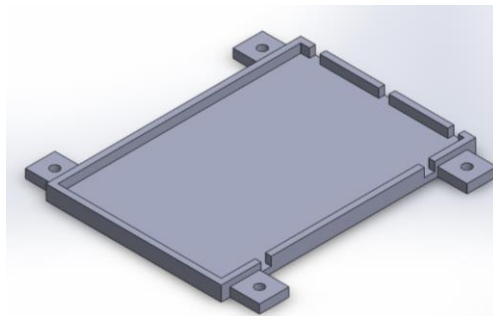


Figura 15: Diseño para impresora 3D de la base del protoboard.

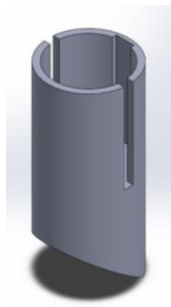


Figura 16: Diseño para impresora 3D de la base de la antena Wifi del sistema embebido LattePanda.

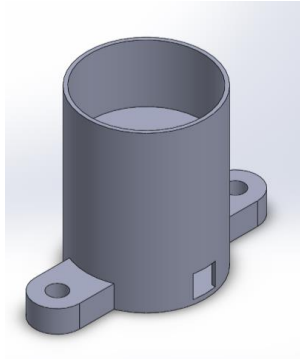


Figura 17: Diseño para impresora 3D de la base del motor DC de la práctica 7.

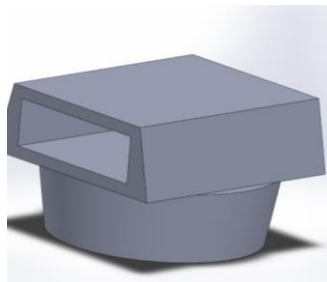


Figura 18: Diseño para impresora 3D de la base del rotor con ranura para imán del motor DC de la práctica 7.

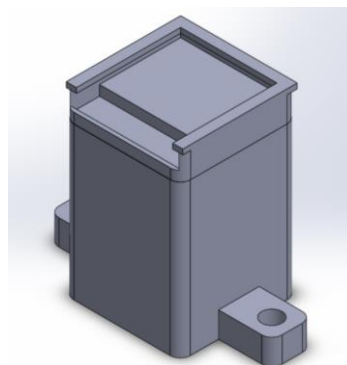


Figura 19: Diseño para impresora 3D de la base del Sensor Hall de la práctica 7.

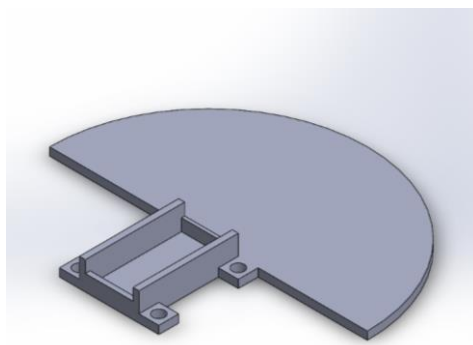


Figura 20: Diseño para impresora 3D de la base del servo-motor de la práctica 8.

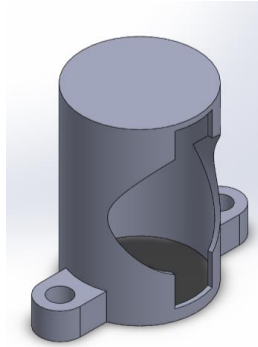


Figura 21: Diseño para impresora 3D de la base del foco de 12V de la práctica 10.

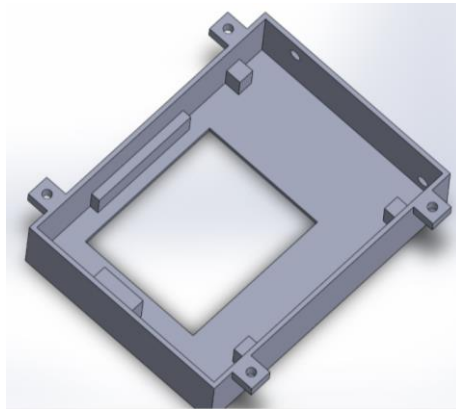


Figura 22: Diseño para impresora 3D de la base de los rieles y motores de la práctica 9.

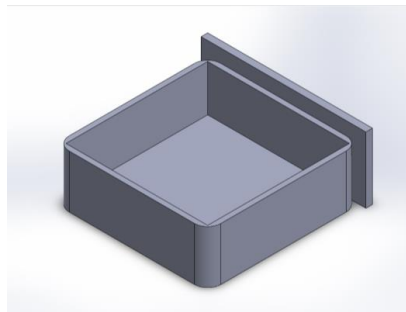


Figura 23: Diseño para impresora 3D del sensor IR de obstáculos de la práctica 9.

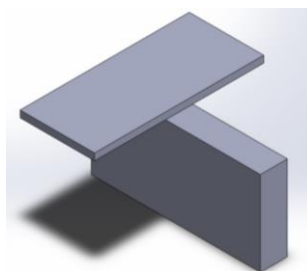


Figura 24: Diseño para impresora 3D T indicadora de nivel de la práctica 9.

Una vez terminadas las impresiones 3D se procede a la construcción de la estructura.

3.2. Construcción de la estructura

Como base para la estructura fue necesaria una plancha de acrílico negro de 3mm, la cual fue recortada en 7 bases, una para el módulo principal donde se encuentran los equipos de control y 6 bases secundarias donde se encuentran distribuidos los sensores de tal manera que combinen para diferentes prácticas de laboratorio. Las bases fueron recortadas de tal manera que puedan encajar con el módulo principal, lo que facilita el intercambio de los módulos didácticos, para el uso de diversas prácticas de laboratorio.



Figura 25: Recorte de plancha de acrílico negro para base de la estructura.

Una vez recortada la plancha se procede a limpiarla para luego atornillar los elementos de control y periféricos en cada base, estos fueron distribuidos de tal manera que tenga fácil acceso de manipulación y cableado. Para una mejor identificación de los sensores se procedió a etiquetar cada uno de ellos.



Figura 26: Bases de acrílico negro limpias y recortadas.

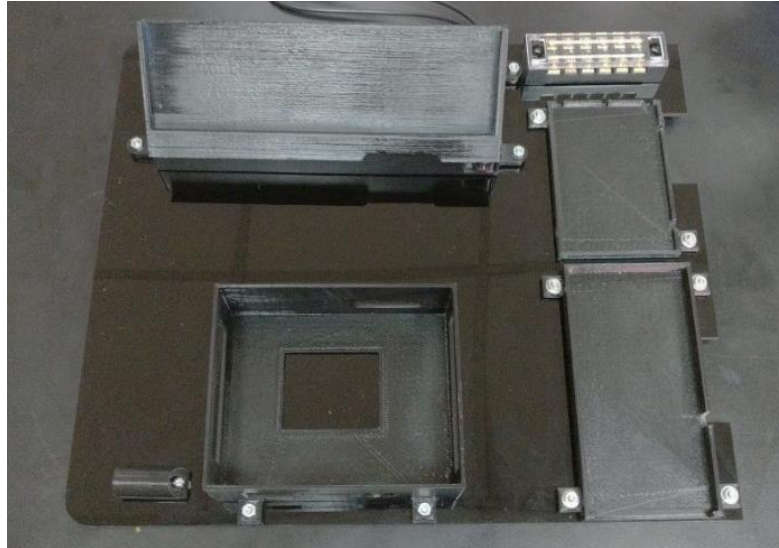


Figura 27: Montaje de bases 3D del módulo principal.

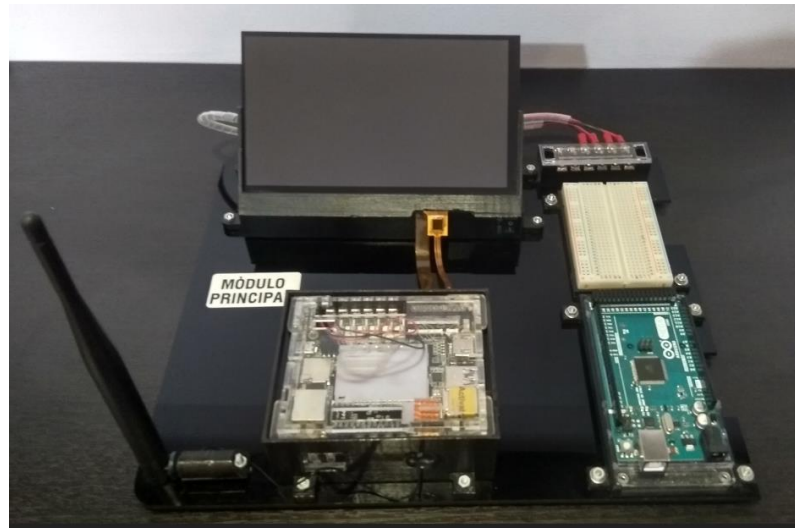


Figura 28: Montaje de equipos en bases 3D del módulo principal.



Figura 29: Montaje y etiquetado del módulo 1.



Figura 30: Montaje y etiquetado del módulo 2.

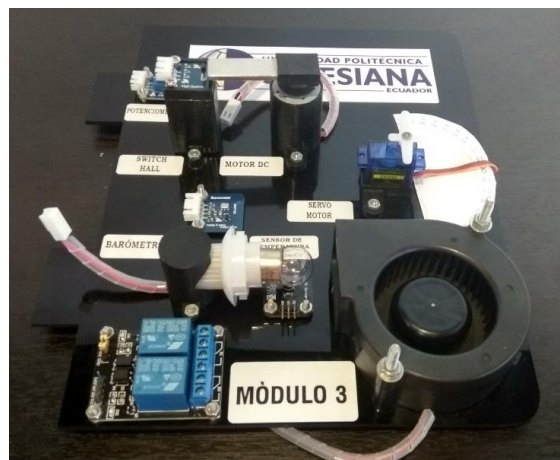


Figura 31: Montaje y etiquetado del módulo 3.



Figura 32: Montaje y etiquetado del módulo 4.



Figura 33: Montaje y etiquetado del módulo 5.



Figura 34: Montaje y etiquetado del módulo 6.

3.3. Conexiones eléctricas

Una vez montados los equipos, se procede a realizar las conexiones eléctricas entre los elementos que conforman el módulo principal.

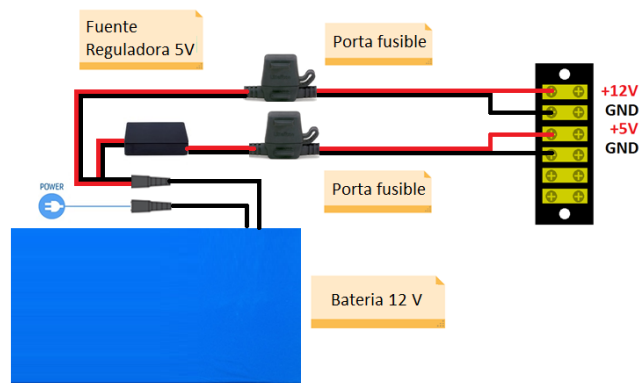


Figura 35: Diagrama de conexión batería de litio de 12V 8AH, regulador de voltaje de 5V 1.5A.

3.3.1. Conexión de Pantallas IPS y Capacitiva

Antes de realizar las conexiones de las pantallas IPS y capacitiva con el sistema embebido LattePanda, se debe montar la pantalla Capacitiva sobre la pantalla IPS siguiendo los siguientes pasos:

1. Retirar la mica protectora de la pantalla IPS.

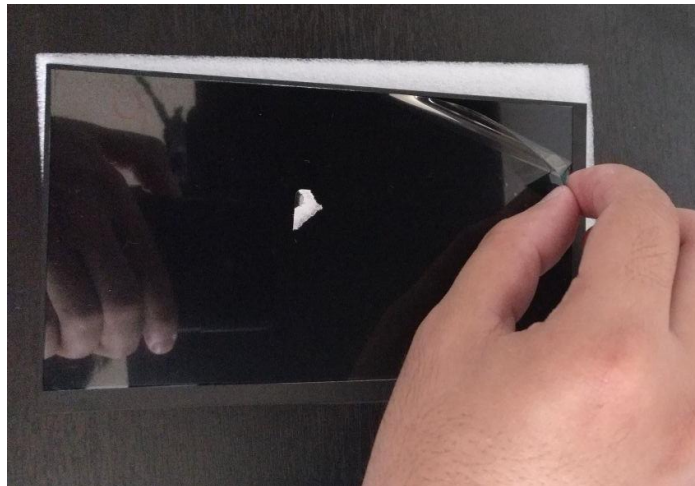


Figura 36: Retiro de mica de pantalla IPS.

2. Retirar el protector de la cinta adhesiva de la pantalla capacitiva.

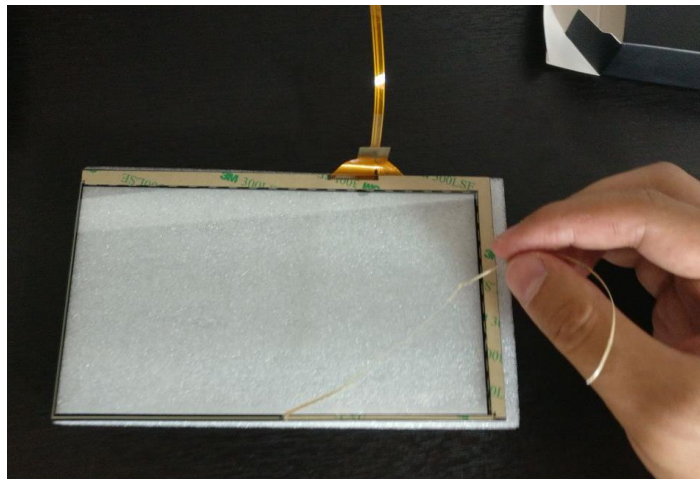


Figura 37: Retiro de protector de la cinta adhesiva de la pantalla capacitiva.

3. Montar la pantalla Capacitiva sobre la pantalla IPS.



Figura 38: Montaje de pantalla Capacitiva sobre pantalla IPS.

Una vez montada la pantalla Capacitiva sobre la pantalla IPS, se procede a colocar la pantalla en su base impresa en 3D, para poder realizar las conexiones de los cables planos flexibles (FPC) en las ranuras respectivas del sistema embebido LattePanda como se muestra a continuación.

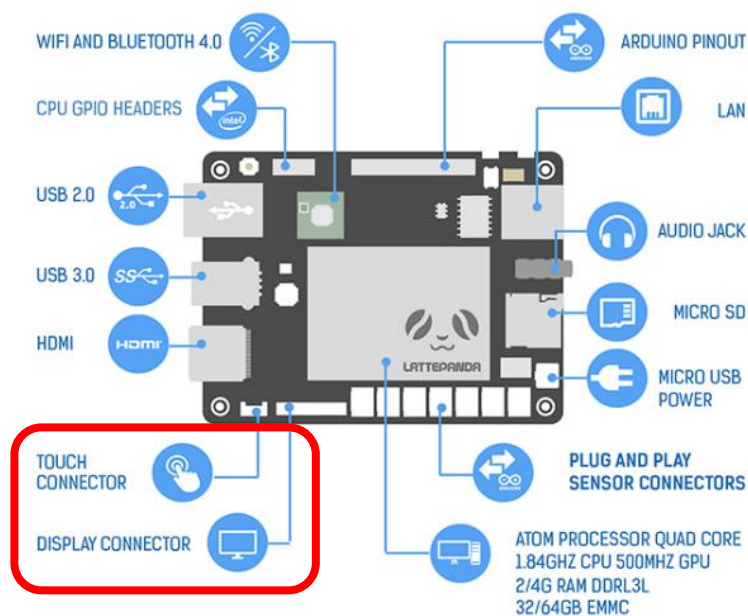


Figura 39: Ubicación de las ranuras para Capacitiva e IPS. (LattePanda, 2016).

1. Levantar la ranura de la pantalla IPS.



Figura 40: Ranura para pantalla IPS. (LattePanda, 2016).

2. Insertar el cable plano flexible dentro de la ranura para pantalla IPS.



Figura 41: Conexión de cable flexible plano dentro de la ranura para pantalla IPS.

3. Girar hacia abajo el actuador de la ranura de la pantalla IPS hasta que esté firmemente cerrado.



Figura 42: Ajuste del actuador de la ranura de la pantalla IPS. (LattePanda, 2016).

4. Insertar el cable plano flexible de la pantalla Capacitiva en igual manera que la pantalla IPS repitiendo los pasos del 1 al 3 sobre la ranura de la pantalla Capacitiva.

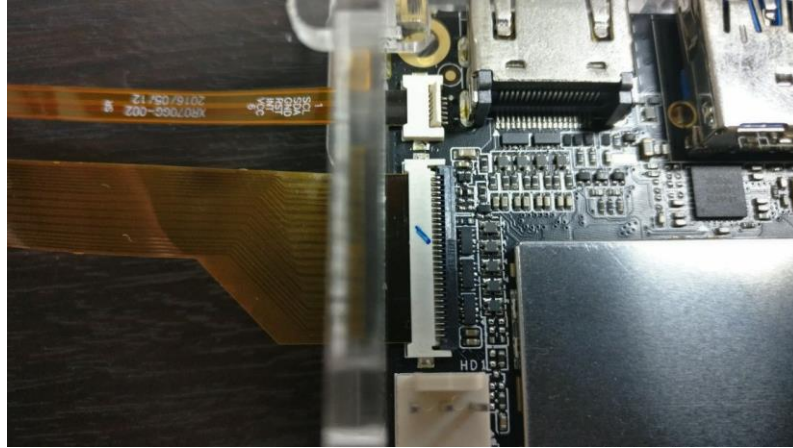


Figura 43: Conexión del cable plano flexible de la pantalla Capacitiva.

3.3.2. Conexión de antena WI-FI

Instalar la antena Wi-Fi conectando el extremo redondo en el zócalo con la etiqueta "ANT" ubicada junto a los pines GPIO en la placa como se muestra en la figura 44.

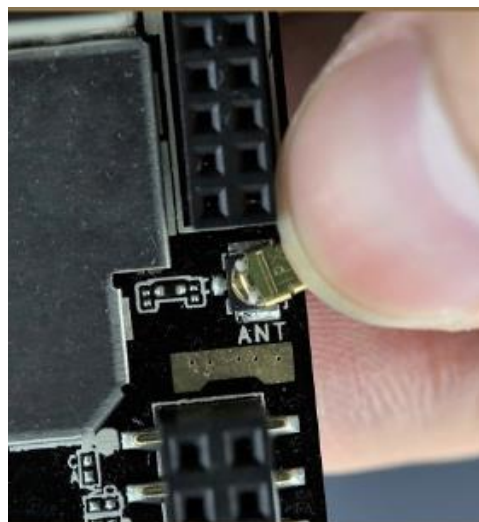


Figura 44: Conexión de antena Wifi. (LattePanda, 2016).

3.3.2. Puesta en marcha de sistema embebido LattePanda

Una vez conectadas las pantallas capacitivas e IPS y la antena Wifi, puede alimentar el sistema embebido LattePanda a través del puerto micro USB o a través de los pines de cabecera CN2, para el proyecto se utiliza la alimentación a través del puerto micro USB.

Para esto se necesita un cable USB 2.0 (conexión de USB a microUSB) y un adaptador USB estándar (como un cargador de pared para teléfono celular) con al menos 2 A de corriente, para usarse como fuente de alimentación del sistema embebido LattePanda.

Para el proyecto se utiliza esta fuente de alimentación NILLKIN de alta calidad que es la solución perfecta como fuente de alimentación del sistema embebido LattePanda.



Figura 45: Cable USB 2.0. (DFRobot, 2016).



Figura 46: Fuente de alimentación NILLKIN. (DFRobot, 2016).

Una vez identificados el adaptador y el cable micro USB, siga las instrucciones a continuación para comenzar.

1. Conecte el dispositivo USB al adaptador de alimentación USB NILLKIN y el micro USB al puerto micro USB del sistema embebido LattePanda (el puerto micro USB se encuentra junto al conector de la tarjeta SD).



Figura 47: Conexión de alimentación del sistema embebido LattePanda. (LattePanda, 2016).

2. Cuando está enchufado, el indicador LED rojo se enciende en la parte inferior de la placa. Esto significa que LattePanda se está inicializando. Espere pacientemente durante unos segundos hasta que el LED se apague.

Nota: No desenchufe el cable de alimentación mientras el sistema está funcionando. (Si el LED rojo está encendido, esto indica que el sistema se está ejecutando)



Figura 48: LED rojo de inicialización del sistema embebido LattePanda. (LattePanda, 2016).

3. Cuando el LED se apaga, mantenga presionado el botón de encendido por un segundo para encender el sistema embebido LattePanda. Se debe observar que el LED se ilumina nuevamente.



Figura 49: Botón de encendido y pantalla encendida del módulo principal.

3.4. Configuración de Software y Drivers

National Instruments ofrece toolkits para comunicar LabVIEW con el sistema embebido Arduino Mega 2560 y poder configurarlo sin necesidad de escribir en lenguaje C.

Para el proyecto se utiliza comunicación serial, por lo cual fue necesario instalar el software controlador NI-VISA que proporciona la interfaz de programación entre el hardware (Arduino Mega 2560) y el entorno de desarrollo de aplicaciones LabVIEW, y el toolkit LINX de Digilent/LabVIEW MakerHub que está diseñado para facilitar el desarrollo de aplicaciones embebidas usando LabVIEW (National Instruments, 2016).

3.4.1. Instalación del Software NI-VISA

Una vez instalado el software LabVIEW en el sistema embebido LattePanda, se debe instalar el software controlador NI-VISA para permitir la comunicación con el sistema embebido Arduino Mega 2560, para esto se descarga de forma gratuita el software directamente de la página oficial de National Instruments a través de una cuenta personal creada por el usuario. La versión del software controlador depende de la versión del software LabVIEW que se esté utilizando, en este caso la versión 16.0 que puede ser encontrada en el siguiente enlace: <http://www.ni.com/download/ni-visa-16.0/6184/en/>.

Descargue y ejecute el archivo autoextraíble. Para instalar NI-VISA 16.0 y todos los archivos necesarios del controlador debe seguir los siguientes pasos:

1. Ejecutar el archivo autoextraíble.

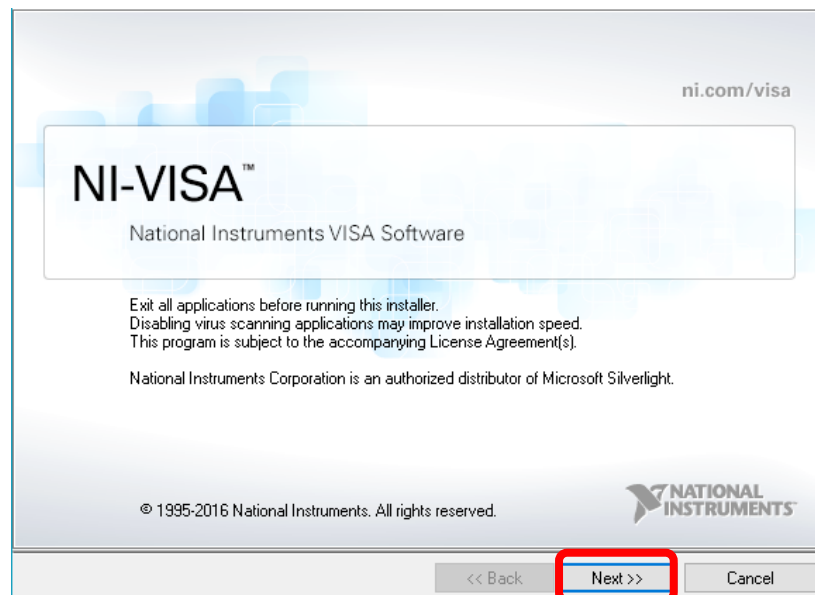


Figura 50: Ventana inicial del proceso de instalación del software NI VISA.

2. Seleccionar la carpeta de destino de instalación en el disco local C.

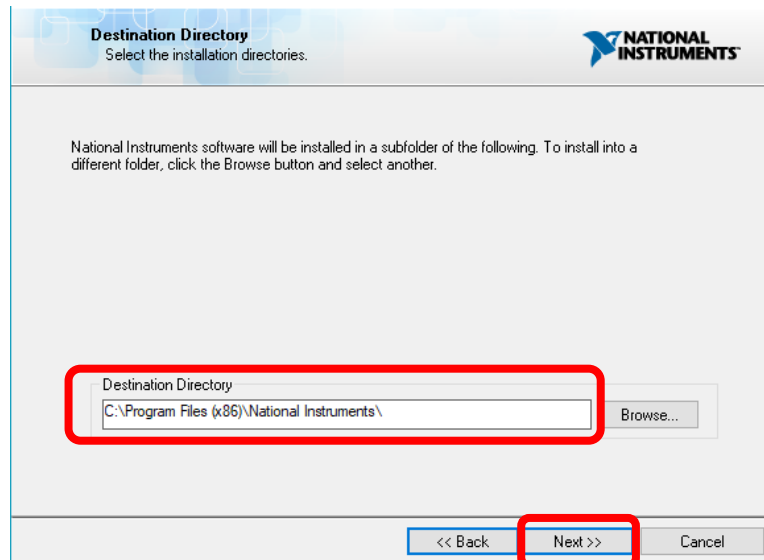


Figura 51: Ventana de selección de destino de instalación del software NI VISA.

3. Seleccionar la opción para recibir notificaciones software NI VISA para facilitar actualizaciones.

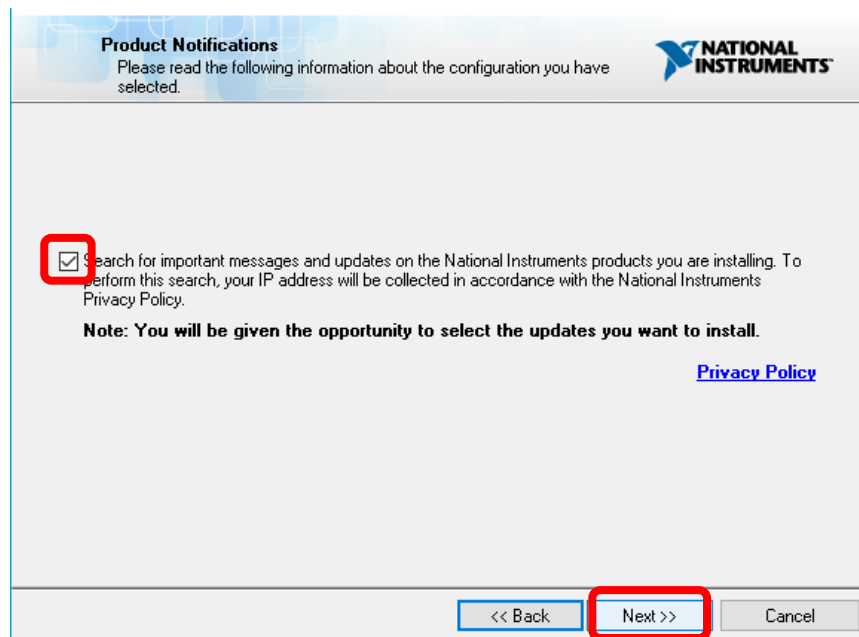


Figura 52: Ventana para recibir notificaciones del software NI VISA.

4. A continuación, se visualizan todos los archivos que se instalaran en el controlador.

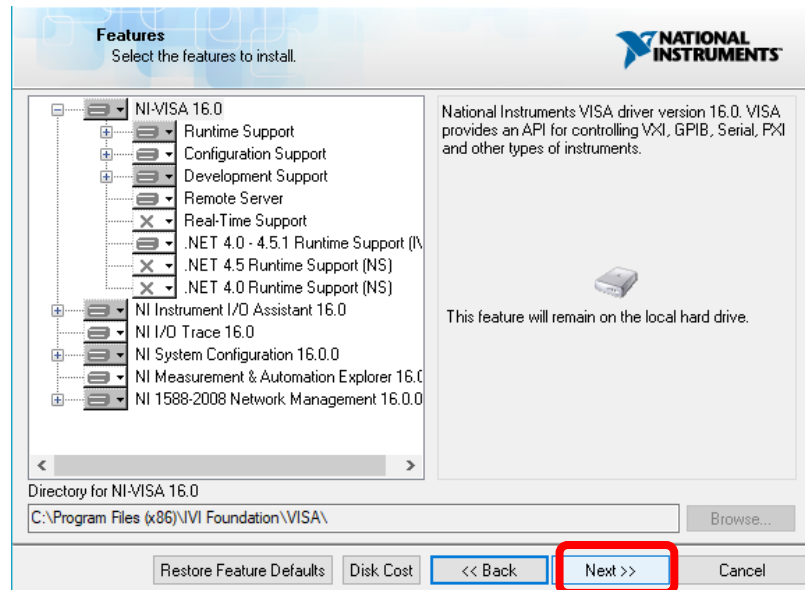


Figura 53: Ventana donde se muestran todos los archivos a instalarse en el software NI VISA.

5. Se procede a aceptar el contrato de licencia.

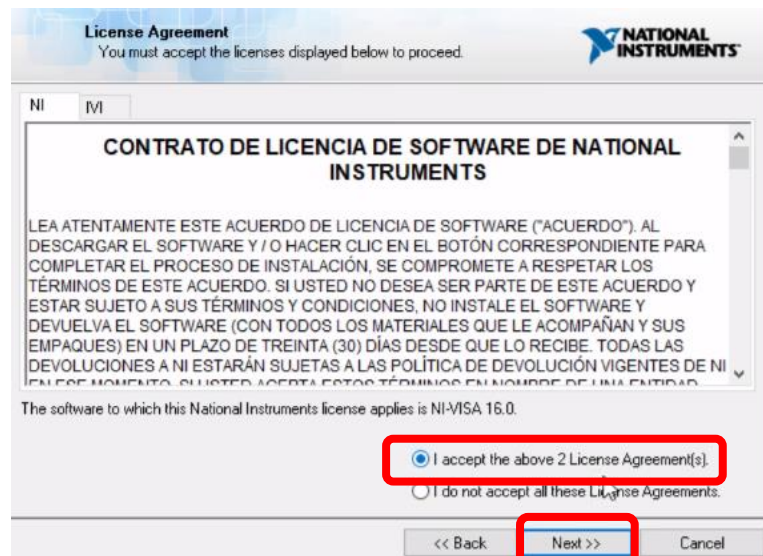


Figura 54: Ventana donde se muestra el contrato de licencia del software NI VISA.

6. Para que Windows no interrumpa el proceso de instalación, seleccionar el check-box correspondiente al acuerdo de confianza que hace referencia a la instalación de softwares provenientes de National Instruments Corporation.

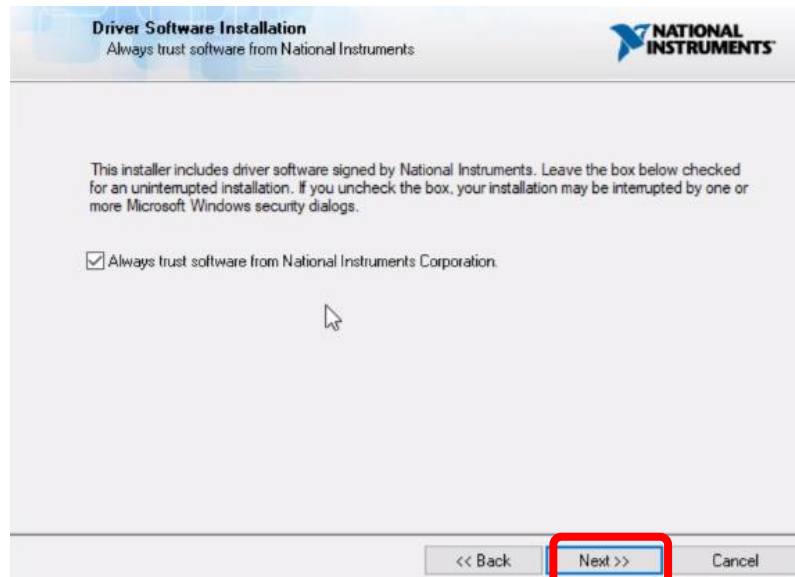


Figura 55: Ventana de aceptación de términos de confianza con los softwares de National Instruments Corporation.

7. Se procede a aceptar el proceso de instalación del software NI VISA y sus controladores.

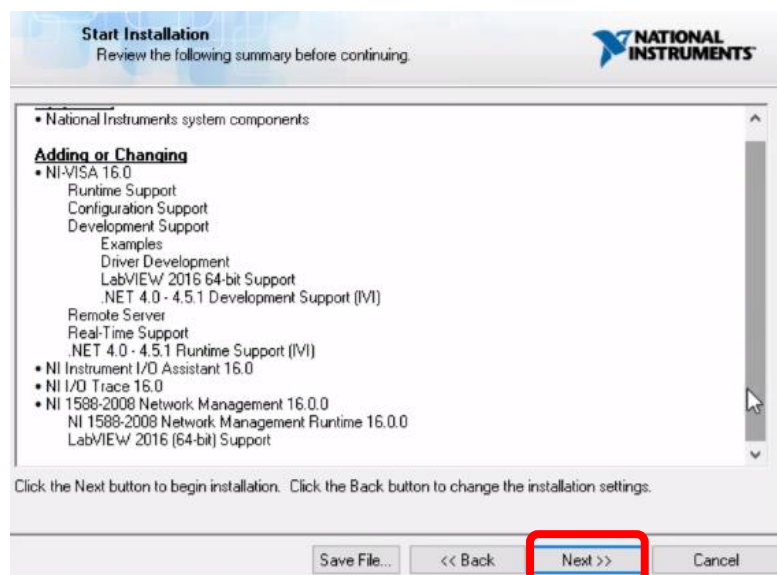


Figura 56: Ventana de inicio de proceso de instalación del software NI VISA y sus controladores.

8. Una vez terminado el proceso de instalación se procede a reiniciar el equipo.

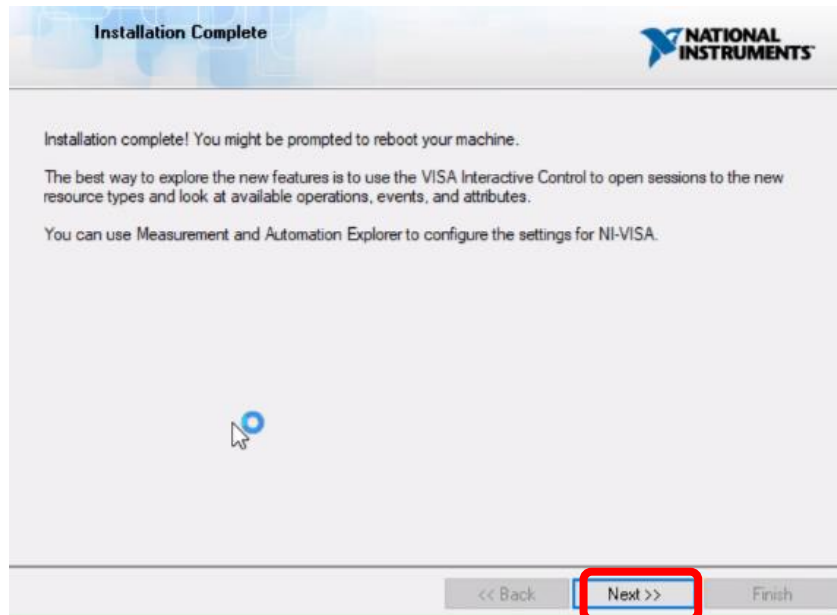


Figura 57: Ventana donde se confirma la instalación del software NI VISA.



Figura 58: Ventana de alerta de reinicio de equipo para completar el proceso de instalación.

3.4.2. Instalación del toolkit LINX de Digilent/LabVIEW MakerHub

El toolkit LINX es una herramienta de fuente abierta de Digilent, diseñada para facilitar el desarrollo de aplicaciones embebidas usando el software LabVIEW por el cual, es necesario para el desarrollo de las prácticas (National Instruments, 2016).

Para obtener este toolkit se debe utilizar el software VI Package Manager incluido en el programa LabVIEW 2016 como se muestra a continuación:

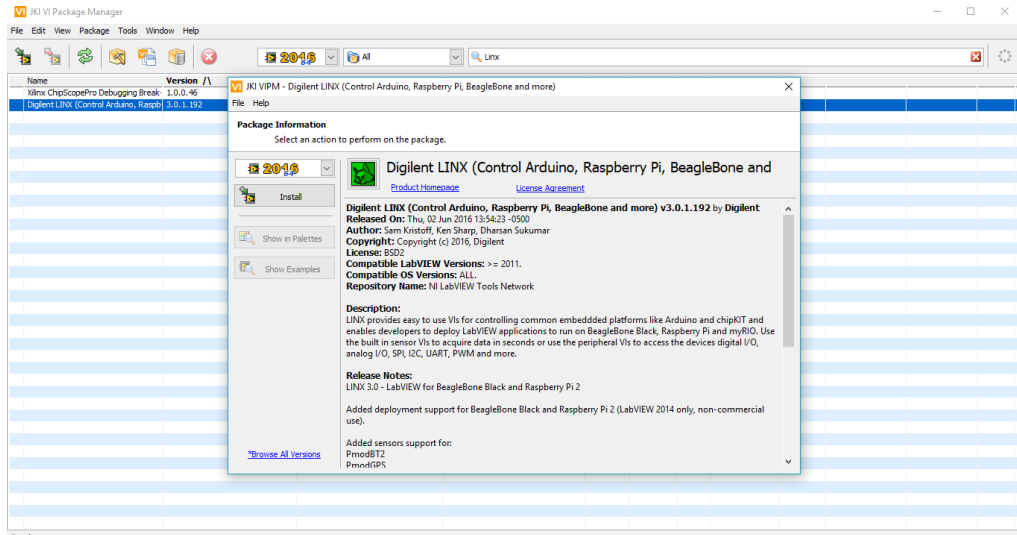


Figura 59: Búsqueda de toolkit Digilent LINX en el software VI Package Manager.

A continuación, se procede a realizar la instalación del toolkit LINX, haciendo clic en el botón Install, como se muestra en la figura 60.

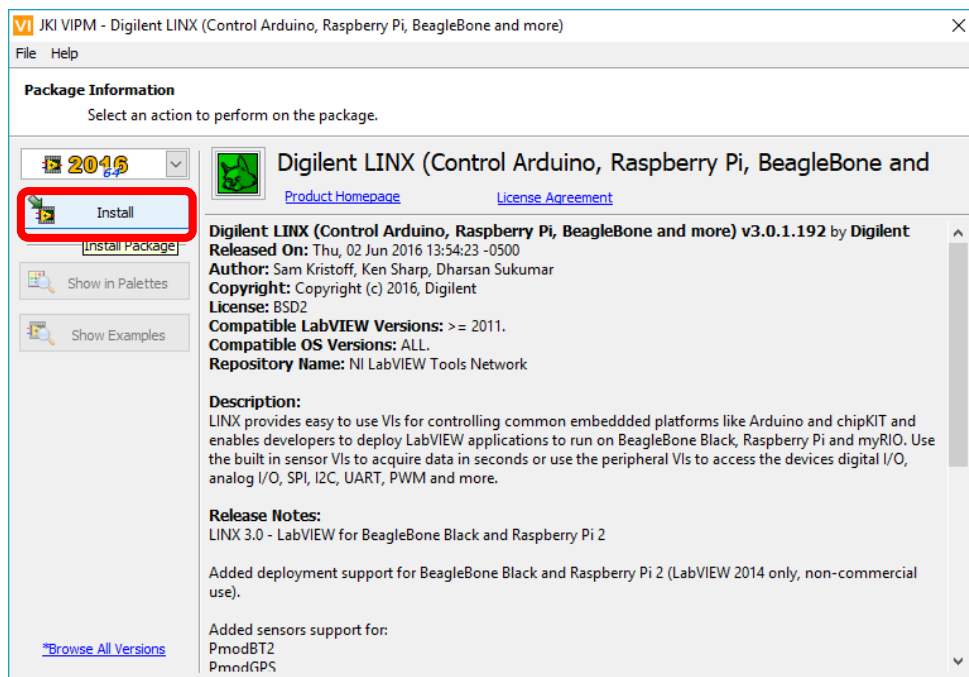


Figura 60: Se inicia instalación del toolkit LINX de Digilent en el software LabVIEW.

Como siguiente paso, se procede a aceptar los acuerdos de licencia del toolkit LINX de Digilent.

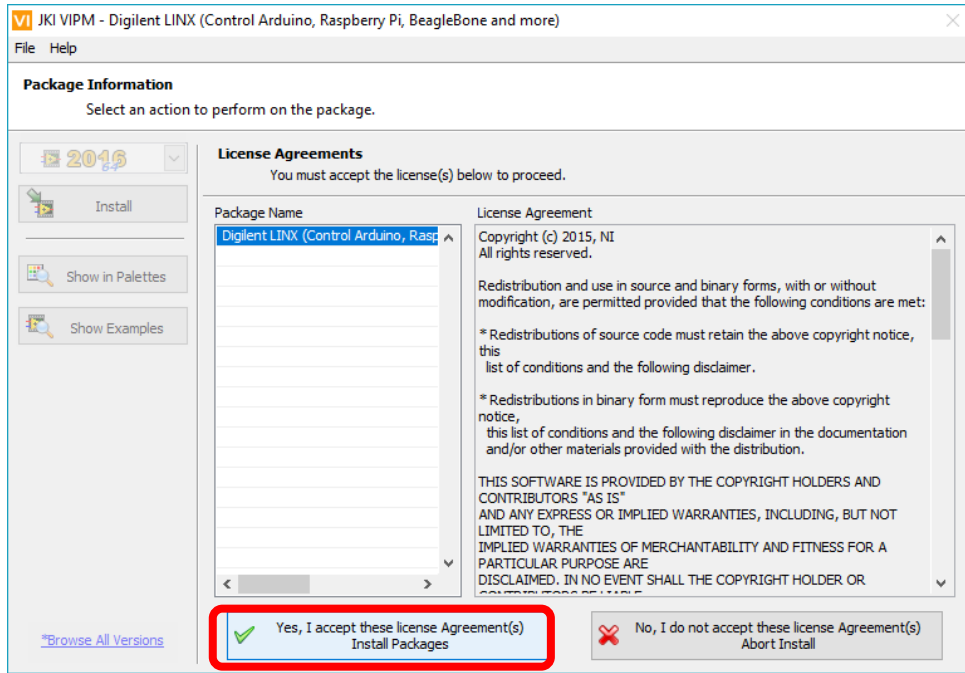


Figura 61: Se aceptan los acuerdos de licencia del toolkit LINX de Digilent.

Una vez aceptados los acuerdos de licencia, automáticamente se inicia la instalación del toolkit LINX de Digilent.

Para finalizar la instalación se procede a dar clic en el botón Finish.

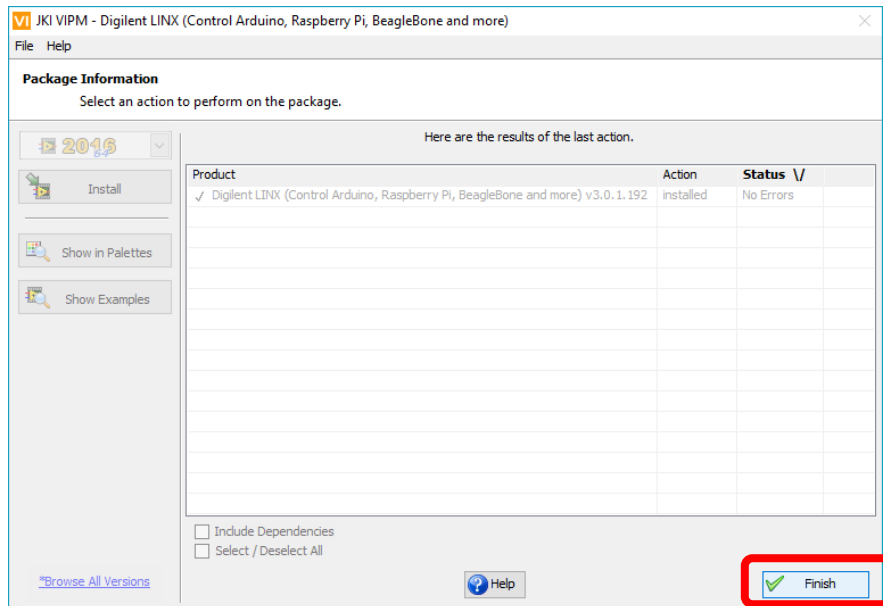


Figura 62: Instalación completada del toolkit LINX de Digilent.

3.4.3. Instalación del módulo Datalogging and Supervisory Control Module

Una vez instalado el software LabVIEW en el sistema embebido LattePanda, se debe instalar el módulo de control DSC que permite registrar base de datos, administrar alarma y eventos realizados por el módulo. Para esto se puede descargar de forma gratuita el software directamente de la página oficial de National Instruments a través de una cuenta personal creada por los estudiantes. La versión del software controlador dependerá de la versión del software LabVIEW que se esté utilizando, en este caso la versión 16.0 que puede ser encontrada en el siguiente enlace:

<http://www.ni.com/download/labview-datalogging-and-supervisory-control-module-2016/6156/en/>

Descargue y ejecute el archivo autoextraíble. Para instalar DSC 16.0 y todos los archivos necesarios del controlador debe seguir los siguientes pasos:

1. Descargar el archivo autoextraíble.

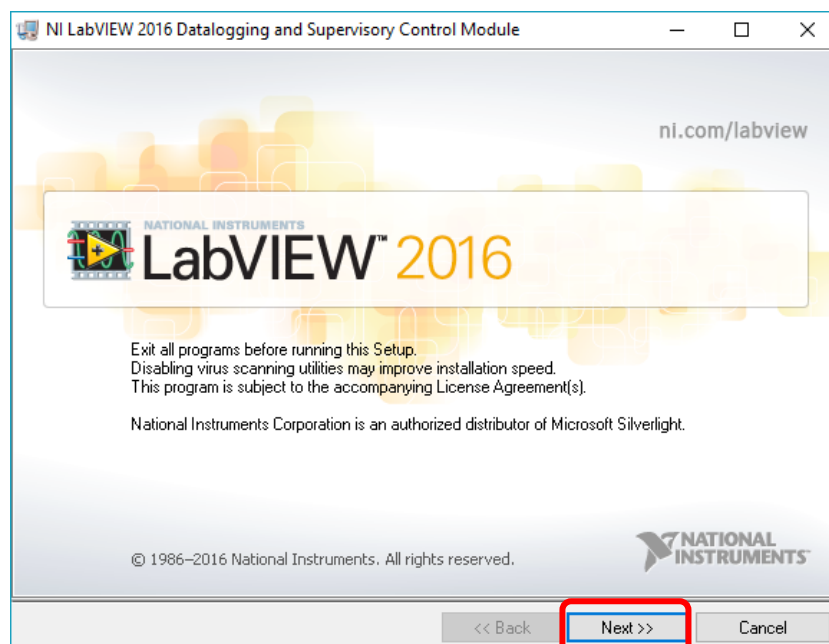


Figura 63: Ventana inicial del proceso de instalación del módulo DSC.

2. Colocar un nombre a la información del usuario.

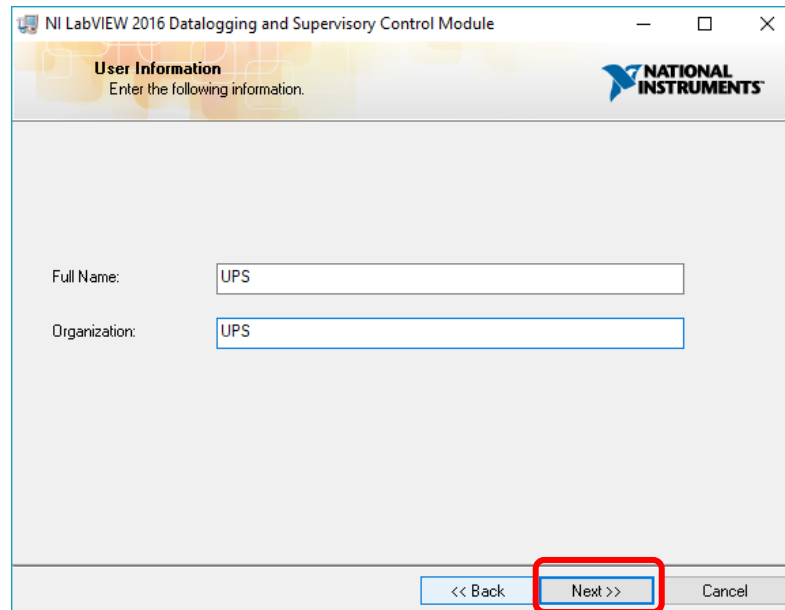


Figura 64: Ventana de información del usuario del proceso de instalación del módulo DSC.

3. Colocar el número de serie del producto.

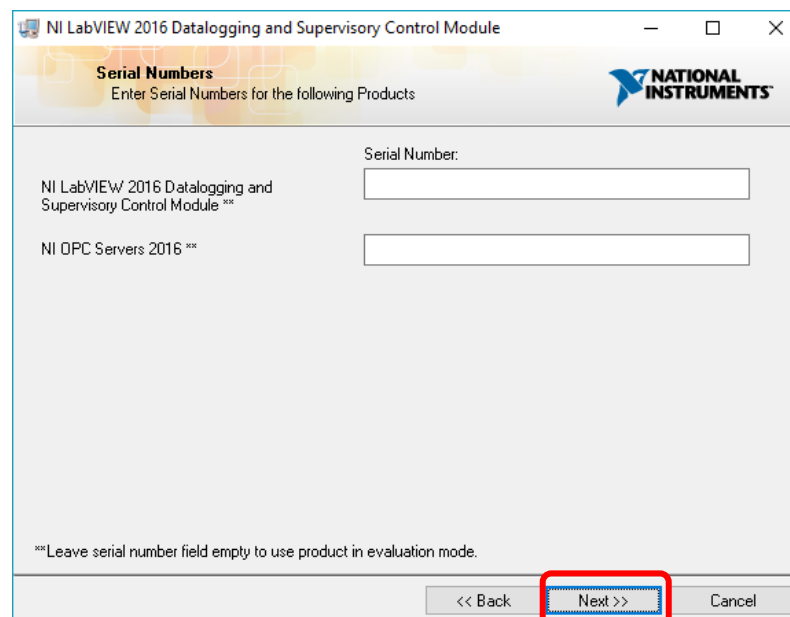


Figura 65: Ventana de registro de número serial de activación del módulo DSC.

4. A continuación, se visualiza todos los archivos que se instalados en el controlador.

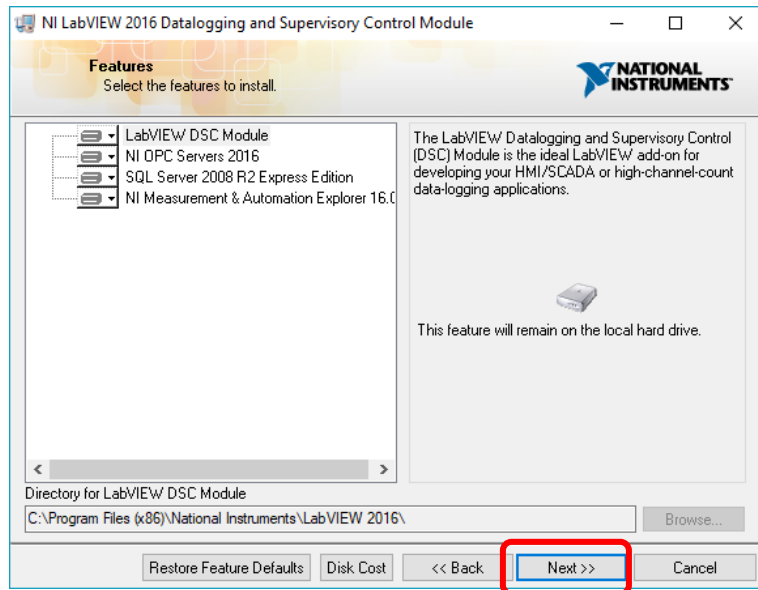


Figura 66: Ventana donde se muestran los todos los archivos a instalarse en el módulo DSC.

5. Seleccionar la opción para recibir notificaciones del módulo DSC para facilitar actualizaciones.

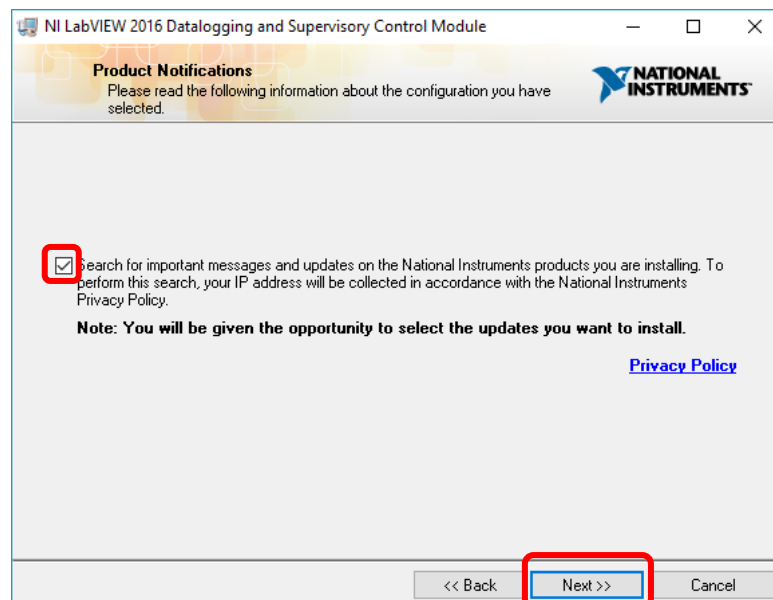


Figura 67: Ventana de aceptación de términos de confianza con los softwares de National Instruments Corporation.

6. Se procede a aceptar el contrato de licencia.

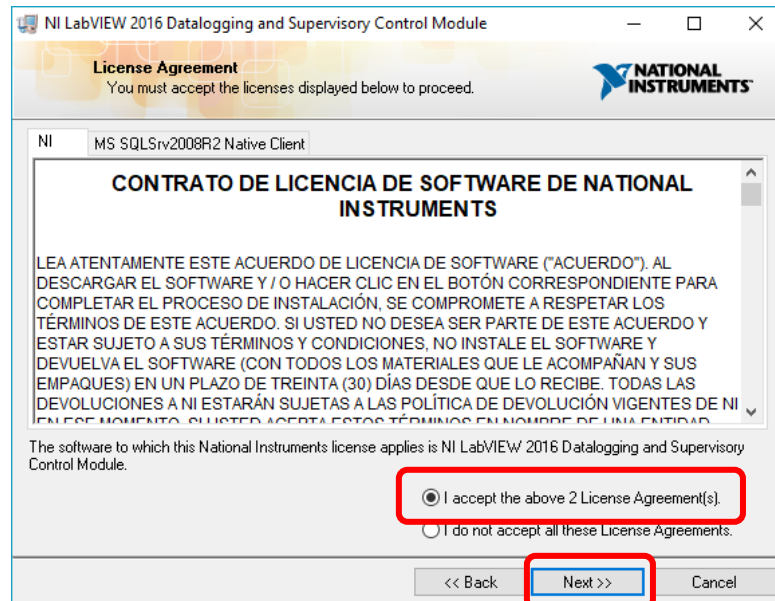


Figura 68: Ventana donde se muestra el contrato de licencia del módulo DSC.

7. Se procede a aceptar el proceso de instalación del módulo DSC y sus controladores.

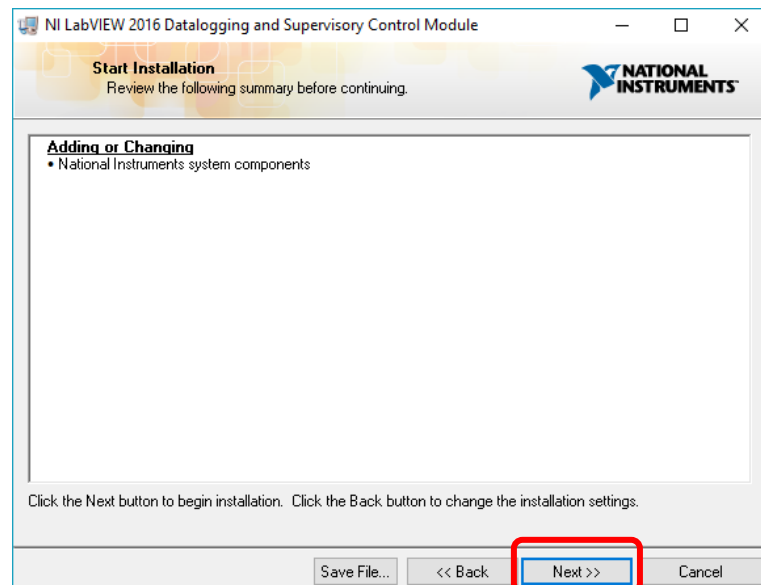


Figura 69: Ventana de inicio de proceso de instalación del módulo DSC y sus controladores.

8. Una vez terminado el proceso de instalación se procede a reiniciar el equipo.

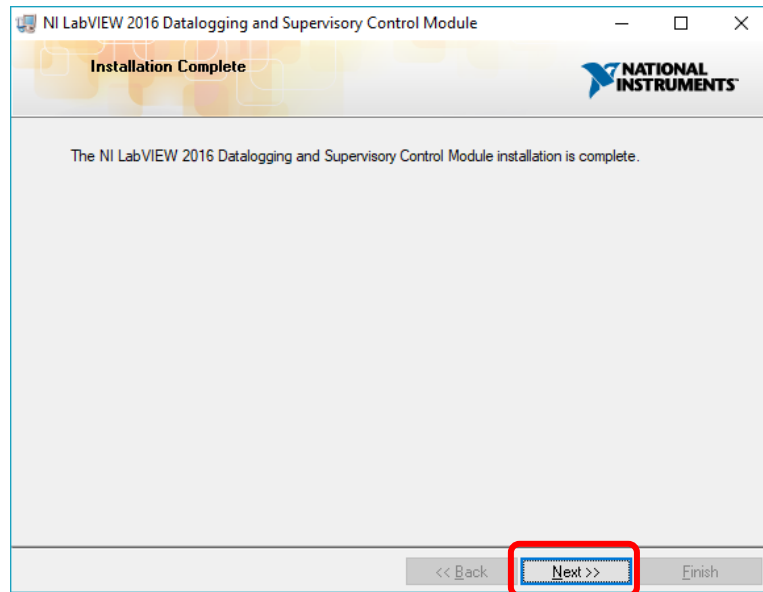


Figura 70: Ventana donde se confirma la instalación del módulo DSC.

4. PRÁCTICAS DE LABORATORIO

En el siguiente capítulo se desarrollan 10 prácticas de laboratorio, las cuales fueron establecidas dentro de los objetivos específicos. Las prácticas fueron elaboradas para analizar el comportamiento de los sensores del kit de Sunfounder exclusivamente, sin embargo, se pueden realizar experimentos más complejos gracias a las tarjetas open source que se utilizan.

Se recomienda analizar cada punto de la práctica previo a la experimentación de esta. El procedimiento debe cumplirse de manera secuencial y sin saltarse ningún paso.

El listado de prácticas a realizarse es el siguiente:

1. Monitor de Gas con alarma sonora.
2. Monitor de Flama con alarma sonora e indicador RGB.
3. Control y monitoreo de un sistema contra incendio.
4. Encendido de un Emisor Láser.
5. Encendido del Auto-Flash LED y Dual Color LED.
6. Barómetro BMP180.
7. Contador de pulsos utilizando Sensor Hall.
8. Control de posición de un Servo-Motor.
9. Simulación de proceso de embotellado.
10. Control de temperatura.

4.1. Práctica #1


| | | |
|--|---|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 1 | TÍTULO PRÁCTICA: Monitor de Gas con alarma sonora. |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual permita detectar la presencia de Gas en el ambiente. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar el sensor de Gas MQ-2 en el software LabVIEW 2016. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Elaborar una base de datos donde registren los eventos y datos obtenidos por el sensor. | | |
| INSTRUCCIONES: | | 1. Alimentar el módulo principal con 110v. |
| | | 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. |
| | | 3. Seguir la guía paso a paso para realizar la conexión del módulo. |
| | | 4. Realizar una base de datos en ACCESS para registrar los eventos y datos obtenidos por el sensor. |
| MARCO TEORICO | | |
| Sensor de Gas MQ-2 Introducción El sensor de gas MQ-2 es un sensor de gases inflamables y humo que permite la detección de presencia o concentración de gas combustible en el aire dentro de un área determinada. En multitud de ocasiones este sensor es implementado en equipos de detección de gases para el humo y gases inflamables como GLP, i-butano, propano, metano y alcohol en el hogar, la industria o el automóvil (Sunfounder, 2013). | | |



Figura 71: Sensor de Gas MQ-2 práctica 1. (Sunfounder, 2013).

Principio experimental

El sensor de gas, serie MQ-2 aplica SnO₂ (un semiconductor de tipo n deficiente en oxígeno) que tiene una conductividad más baja en el aire claro como un material sensor de gas. En una atmósfera donde puede haber gases inflamables, la conductividad del sensor de gas aumenta junto con la concentración de gas inflamable. MQ-2 realiza una buena detección de diferentes gases inflamables, como el gas natural, especialmente sensible al gas licuado, al propano y al hidrógeno (Sunfounder, 2013).

En este experimento, suelte un poco de gas combustible alrededor del sensor y el valor en A0 aumenta. Una vez que la concentración del gas excede un límite, D0 produce un bajo nivel y se enciende el LED correspondiente. Tomar en cuenta que el sensor necesita tiempo de grabación para que las lecturas puedan ser más consistentes y confiables (Sunfounder, 2013).

Buzzer Activo

Introducción

Un Buzzer o zumbador es un dispositivo de señalización/aviso de funcionamiento auditivo. Los buzzers o zumbadores pueden clasificarse en activos y pasivos (Sunfounder, 2013).

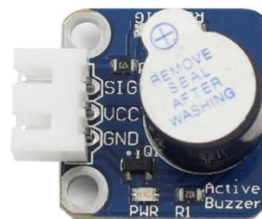


Figura 72: Buzzer Activo práctica 1. (Sunfounder, 2013).

Principio experimental

Un zumbador o buzzer activo tiene una fuente oscilante incorporada, por lo que emite sonidos cuando el mismo sea electrificado. El zumbador activo tiende a ser más costoso que el pasivo debido a múltiples circuitos oscilantes que se encuentran incorporados en su interior (Sunfounder, 2013).

Sensor Táctil

Introducción

Un sensor táctil de metal es un tipo de interruptor que solo funciona cuando es tocado por un cuerpo cargado. Tiene una alta frecuencia transistor que se activa cuando recibe una señal electromagnética (Sunfounder, 2013).

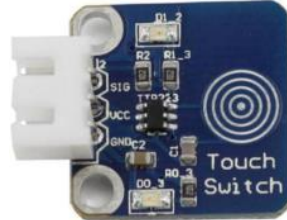


Figura 73: Sensor Táctil práctica 1. (Sunfounder, 2013).

Principio experimental

Cuando el electrodo de base del transistor es tocado se produce la conducción de electricidad a través del mismo, esto debido a que el cuerpo humano es una especie de conductor y una antena que puede captar las ondas electromagnéticas que circulan en el aire (Sunfounder, 2013).

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Sensor de Gas MQ-2** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Conectar el **Buzzer Activo** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
4. Conectar el **Sensor Táctil** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
5. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
6. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.
7. Para utilizar los bloques de las funciones de **LINX**, seguir los siguientes pasos:
 - Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
 - Seleccionar el submenú **MakerHub**.
 - Seleccionar el submenú **LINX**.
8. Para establecer la comunicación, colocar el bloque **OPEN** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 74: Bloque de apertura de comunicación práctica 1.

9. Para visualizar los datos obtenidos por el sensor de gas, utilizar los siguientes bloques:

DATO ANALOGICO: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 0.

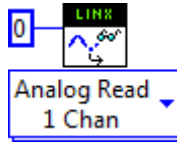


Figura 75: Bloque de Lectura Analógico sensor de gas práctica 1.

SET POINT: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 1.

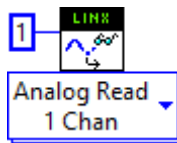


Figura 76: Bloque de Lectura Analógico SET POINT del sensor de gas práctica 1.

DATO DIGITAL: Utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 2. Se recomienda colocar un inversor a la salida del bloque debido a que el mismo funciona con lógica negativa.



Figura 77: Bloque de Lectura Digital sensor de gas práctica 1.

Nota: Para determinar la concentración de Gas en partes por millón se debe realizar el siguiente cálculo referido al Gas LP.

Ecuación 1: Concentración de Gas en PPM práctica 1.

$$\text{Concentración} = 10^{(A \times \log(\frac{RS}{R})) + B}$$

Para determinar la concentración se necesita la recta que la aproxima, para lo cual se debe seleccionar dos puntos cuales quiera de la gráfica P0=(X0,Y0) y P1=(X1,Y1), resultando la ecuación de la recta:

Ecuación 2: Ecuación de la recta práctica 1.

$$Y = Ax + B$$

Siendo $A = \frac{Y1-Y0}{X1-X0}$ & $B = Y0 - AX0$

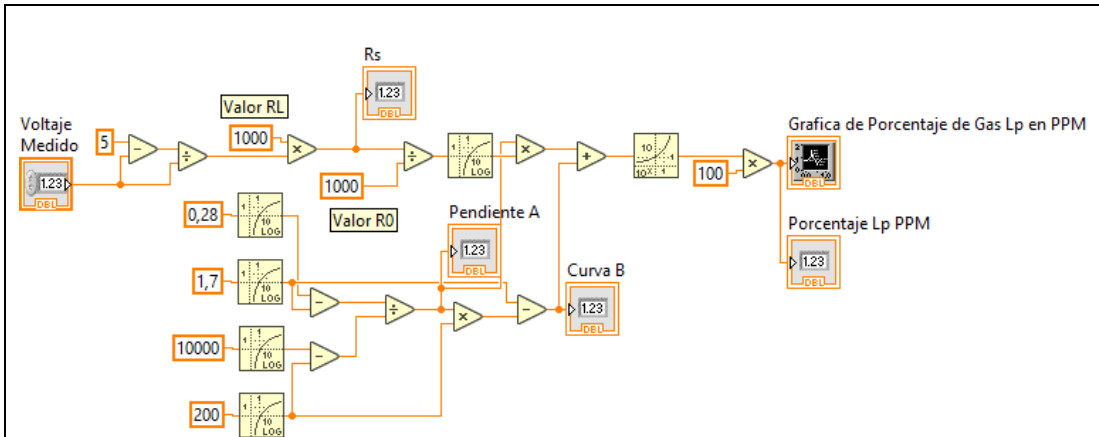


Figura 78: Cálculo de concentración de Gas en PPM.

Para realizar el cálculo del Set-Point utilizar la fórmula de la ecuación 1.

10. Para activar el Buzzer Activo, utilizar un bloque de escritura digital (**Digital Write**) y seleccione el puerto digital número 4.

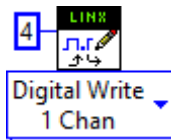


Figura 79: Bloque de Lectura Digital buzzer activo práctica 1.

11. Para recibir la señal del pulsador "**Paro físico**", utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 45.

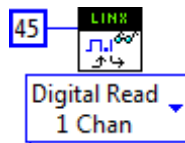


Figura 80: Bloque de Lectura Digital sensor táctil práctica 1.

12. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 81: Bloque de cierre de comunicación práctica 1.

13. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

14. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 82: Bloque de apertura de conexión de Database práctica 1.

15. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 83: Bloque para inserta datos en Database práctica 1.

16. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 84: Bloque de cierre de conexión de Database práctica 1.

CONFIGURACIÓN DE DATABASE

17. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

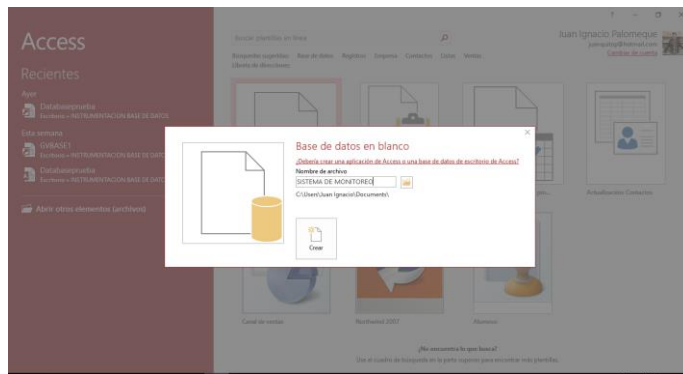


Figura 85: Creación de base de datos en MICROSOFT ACCESS práctica 1.

18. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESS 2002-2003**.

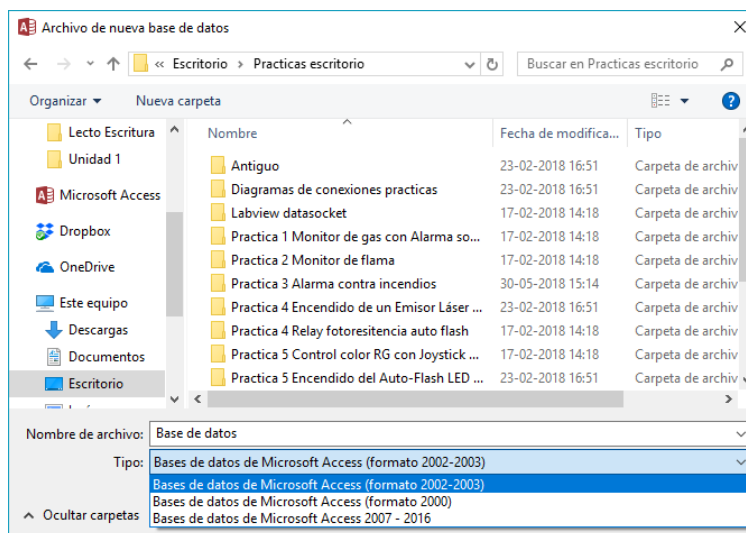


Figura 86: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 1.

19. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

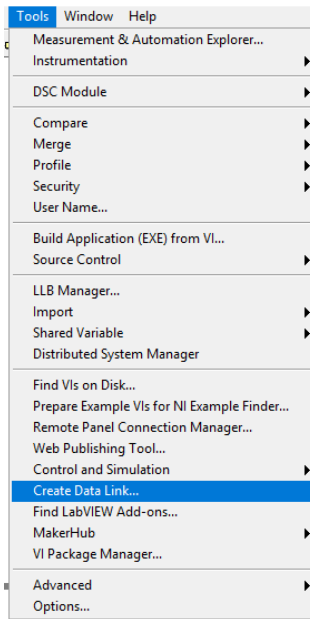


Figura 87: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 1.

20. Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

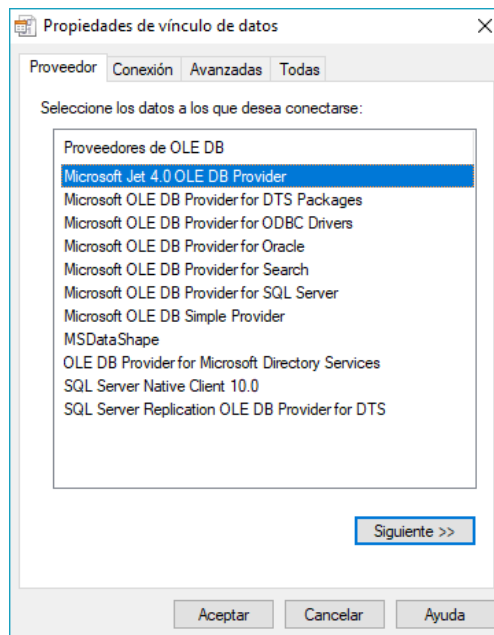


Figura 88: Ventana de propiedades de vínculo de datos práctica 1.

21. Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

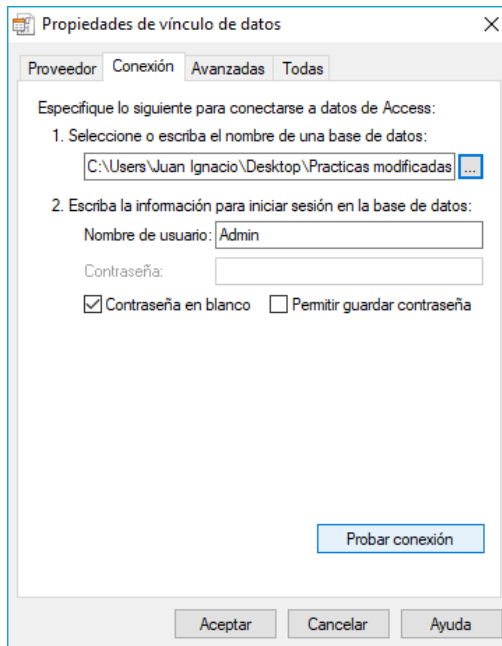


Figura 89: Selección de ruta donde se ubica la base de datos práctica 1.

22. Una vez enlazada la ruta se procede a probar conexión.

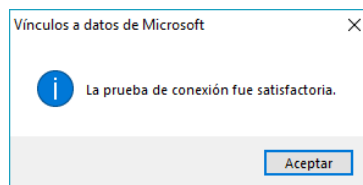


Figura 90: Vista de prueba de conexión de la vinculación de datos práctica 1.

23. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

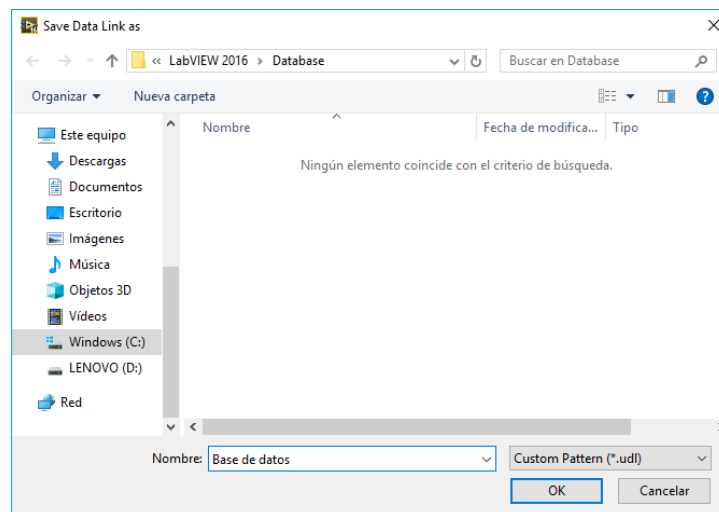


Figura 91: Ventana de asignación de nombre para la tabla de registro de datos práctica 1.

24. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

connection information



Figura 92: Control de ruta de base de datos práctica 1.

25. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:

- Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard**.
- En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next**.
- Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next**.
- Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next**.
- Verificar que los LEDs **TX** y **RX** deben permanezcan encendidos durante unos segundos mientras se implementa el firmware.
- Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.

26. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

27. Dar clic en el botón **Run Continuously**.

28. Liberar manualmente gas propano frente Sensor de Gas MQ-2.

29. Observar el comportamiento a través de la pantalla del módulo de las salidas analógica y digital del Sensor de Gas MQ-2.

30. Comprobar la activación de la alarma sonora mediante el funcionamiento del Buzzer Activo.

31. Utilizar el Sensor Táctil en caso de querer interrumpir el programa de manera definitiva.

DIAGRAMA DE CONEXIONES

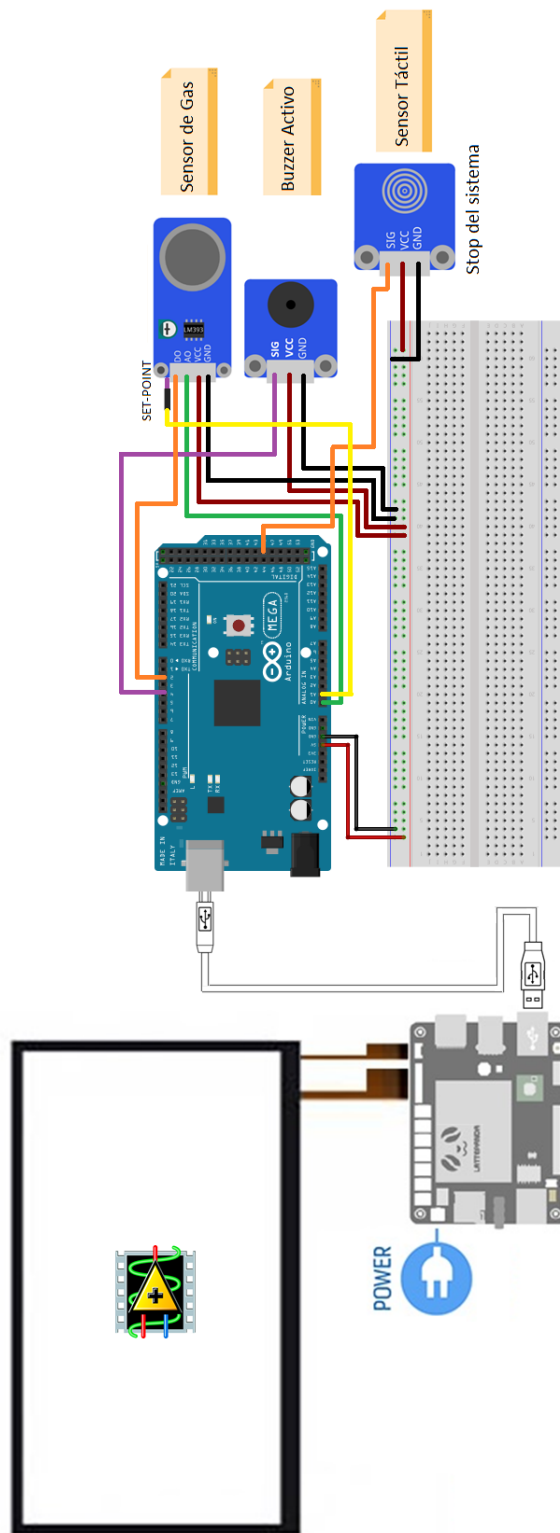


Figura 93: Diagrama de conexiones de la Práctica 1.

Tabla 2: Conexiones de la Práctica 1.

| SENSOR DE GAS | ARDUINO MEGA 2560 |
|---------------|-------------------|
| D0 | 2 |
| A0 | A0 |
| A1(set-point) | A1 |
| VCC | 5V |
| GND | GND |
| BUZZER ACTIVO | ARDUINO MEGA 2560 |
| SIG | 4 |
| VCC | 5V |
| GND | GND |
| SENSOR TÁCTIL | ARDUINO MEGA 2560 |
| SIG | 45 |
| VCC | 5V |
| GND | GND |

DISEÑOS EN LABVIEW

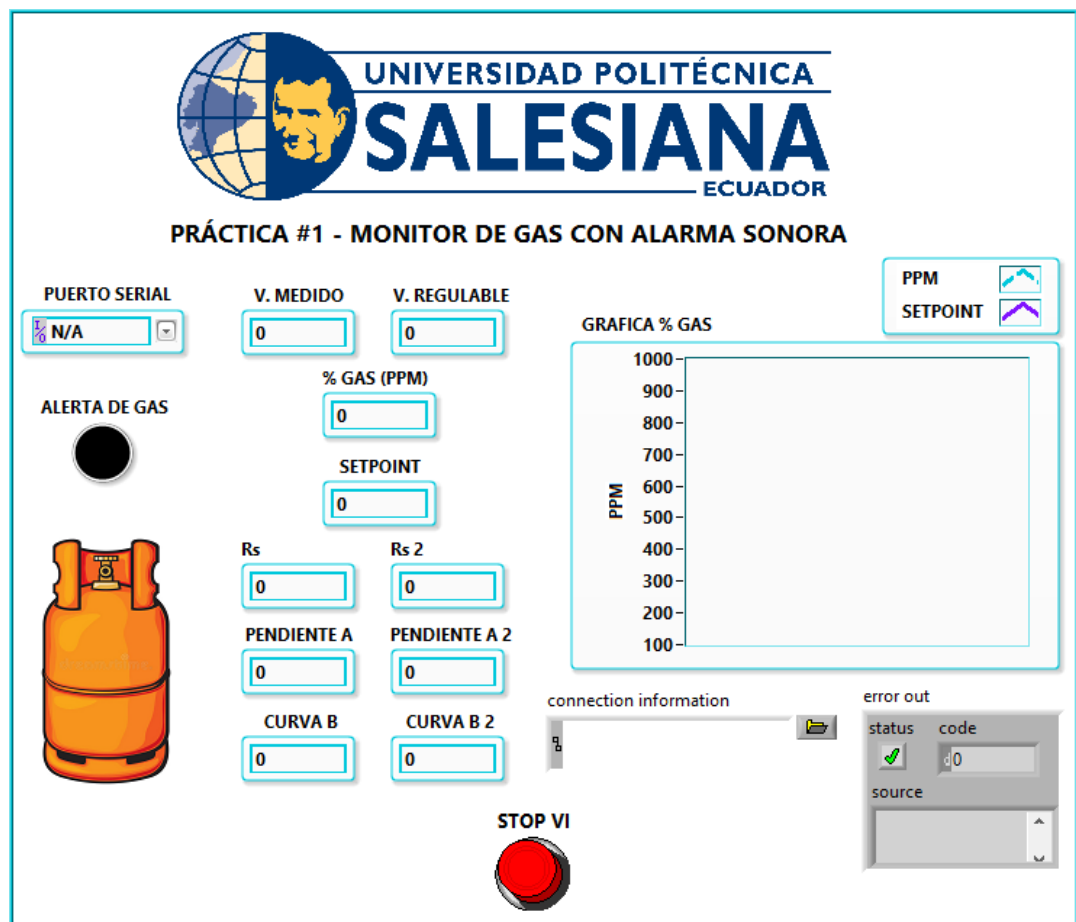


Figura 94: Panel Frontal de la Práctica 1.

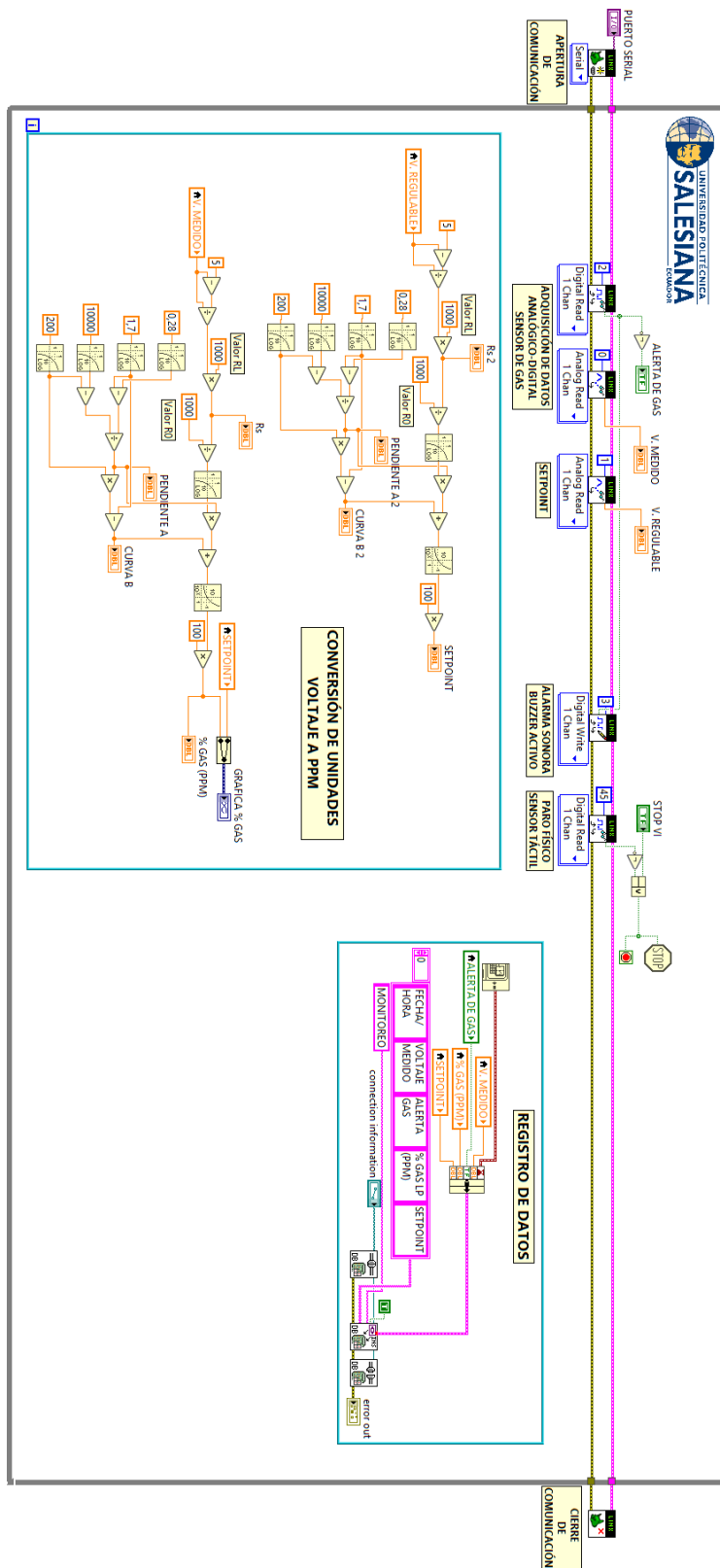


Figura 95: Diagrama de Bloques de la Práctica 1.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.
3. Gracias a las salidas que posee el Sensor de Gas MQ-02 se pueden mostrar y almacenar valores analógicos y digitales generados por el usuario.
4. Utilizando el potenciómetro que posee el Sensor de Gas MQ-02 se logró establecer un Set-Point que condiciona la activación del Buzzer Activo.
5. Se comprobó el funcionamiento del Sensor de Gas MQ-02 mediante la exposición del mismo frente a cierta cantidad de gas LP.

ANALISIS DE RESULTADO:

Cuando el sensor de gas MQ-02 detecta la presencia de gas propano en el ambiente se produce un cambio en el valor de voltaje analógico entregado mediante el pin A0, dicho valor convertido a PPM puede ser observado en la gráfica ubicada en el panel frontal.

Si la cantidad de gas que se encuentra en el ambiente excede la cantidad definida mediante el potenciómetro o Set-Point, se emite un pulso mediante el pin D0 del módulo.

El Buzzer Activo se acciona en el momento en el que el valor medido excede el Set-Point.

Cuando el Sensor Táctil es activado el VI deja de ejecutarse.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

Todos los datos adquiridos tales como Set-Point, presencia de gas y activación del Buzzer Activo se registran en una base de datos en tiempo real.

4.2. Práctica # 2


| | | |
|--|---|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 2 | TÍTULO PRÁCTICA: Monitor de flama con alarma sonora e indicador RGB. |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual permita detectar la presencia de flama en el área determinada. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar el sensor de Flama en el software LabVIEW 2016. Configurar una alarma sonora con el Buzzer activo y una alarma lumínica con el LED RGB. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Elaborar una base de datos donde registren los eventos y datos obtenidos por el sensor. | | |
| INSTRUCCIONES: | | 1. Alimentar el módulo principal con 110v. |
| | | 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. |
| | | 3. Seguir la guía paso a paso para realizar la conexión del módulo. |
| | | 4. Realizar una base de datos en ACCESS para registrar los eventos y datos obtenidos por el sensor. |
| MARCO TEORICO | | |
| Sensor de Flama Introducción Un módulo de sensor de flama realiza la detección capturando las longitudes de onda infrarrojas de la llama. Se puede usar para detectar y advertir llamas. En este experimento, se utiliza el sensor para hacer una alarma de llama (Sunfounder, 2013). | | |



Figura 96: Sensor de Flama práctica 2. (Sunfounder, 2013).

Principio experimental

Hay varios tipos de sensores de flama. En este experimento, se utiliza un sensor de flama de infrarrojo lejano. Puede detectar la luz infrarroja con una longitud de onda de 700 nm a 1000 nm. Una sonda de llama de infrarrojo lejano convierte los cambios de fuerza de la luz infrarroja externa en cambios actuales. Y luego convierte las cantidades analógicas en digitales (Sunfounder, 2013).

Buzzer Activo

Introducción

Un buzzer o zumbador es un dispositivo de señalización de audio. Los zumbadores pueden clasificarse en activos y pasivos (Sunfounder, 2013).



Figura 97: Buzzer Activo práctica 2. (Sunfounder, 2013).

Principio experimental

Un zumbador activo tiene una fuente oscilante incorporada, por lo que emite sonidos cuando se electrifique. El zumbador activo a menudo es más caro que el pasivo debido a múltiples circuitos oscilantes incorporados (Sunfounder, 2013).

RGB LED

Introducción

Los módulos RGB LED pueden emitir varios colores de luz. Tres LED de rojo, verde y azul se empaquetan en una carcasa de plástico transparente o semitransparente con cuatro pasadores. Los tres colores primarios rojo, verde y azul se pueden mezclar y componer todo tipo de colores por brillo, por lo que puede hacer que un LED RGB emita luz de colores controlando el circuito (Sunfounder, 2013).

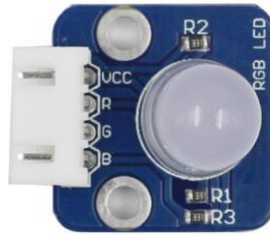


Figura 98: RGB LED práctica 2. (Sunfounder, 2013).

Principio experimental

En este experimento, se utiliza la tecnología PWM para controlar el brillo de RGB. Cada uno de los tres canales de color: rojo, verde y azul, tiene 255 niveles de brillo. Cuando los tres colores primarios son todos 0, la luz LED se apaga. Cuando los colores son todos 255, el LED se vuelve más brillante. Aquí se ingresa cualquier valor entre 0 y 255 a los tres pines del LED RGB para que muestre diferentes colores.

Los LED RGB pueden clasificarse en LED de ánodo común y LED de cátodo común. En este experimento, se utiliza un LED RGB de ánodo común (Sunfounder, 2013).

Sensor Táctil

Introducción

Un sensor táctil de metal es un tipo de interruptor que solo funciona cuando es tocado por un cuerpo cargado. Tiene una alta frecuencia transistor que se activa cuando recibe una señal electromagnética (Sunfounder, 2013).

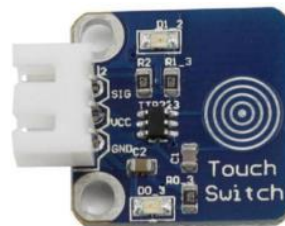


Figura 99: Sensor Táctil práctica 2. (Sunfounder, 2013).

Principio experimental

Cuando el electrodo de base del transistor es tocado se produce la conducción de electricidad a través del mismo, esto debido a que el cuerpo humano es una especie de conductor y una antena que puede captar las ondas electromagnéticas que circulan en el aire (Sunfounder, 2013).

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Sensor de Flama** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.

3. Conectar el **LED RGB** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
4. Conectar el **Buzzer Activo** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
5. Conectar el **Sensor Táctil** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
6. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
7. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.
8. Para utilizar los bloques de las funciones de **LINX**, seguir los siguientes pasos:
 - Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
 - Seleccionar el submenú **MakerHub**.
 - Seleccionar el submenú **LINX**.
9. Para establecer la comunicación, colocar el bloque **OPEN** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 100: Bloque de apertura de comunicación práctica 2.

10. Para visualizar los datos obtenidos por el sensor de flama, utilizar los siguientes bloques:

DATO ANALOGICO: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 0.

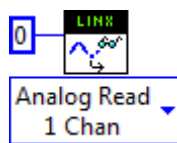


Figura 101: Bloque de Lectura Analógica sensor de flama práctica 2.

SET POINT: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 1.

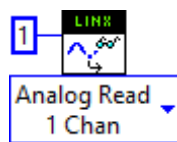


Figura 102: Bloque de Lectura Analógico SET POINT del sensor de flama práctica 2.

DATO DIGITAL: Utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 2. Se recomienda colocar un inversor a la salida del bloque debido a que el mismo funciona con lógica negativa.



Figura 103: Bloque de Lectura Digital sensor de flama práctica 2.

11. Para activar el Buzzer Activo, utilizar un bloque de escritura digital (**Digital Write**) y seleccione el puerto digital número 7.

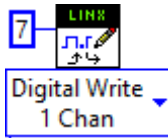


Figura 104: Bloque de Escritura Digital buzzer activo práctica 2.

12. Utilizar la herramienta **Color Box Constant** (*Functions > Programming > Graphics & Sound > Picture Functions*) en el diagrama de bloques para definir el color deseado en el **LED RGB**.

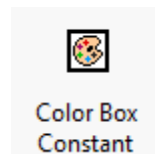


Figura 105: Bloque de Constante de color práctica 2.

13. Utilizar el subVI **Color to RGB** (*Functions > Programming > Graphics & Sound > Picture Functions*) en el diagrama de bloques para realizar la conversión del color deseado a sus componentes rojo, verde y azul (R, G y B) para luego ser convertidos y utilizados en el **LED RGB**.

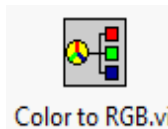


Figura 106: Bloque conversión Color a RGB práctica 2.

14. Debido a que el Duty Cycle de este módulo trabaja en el rango de **0–1**, no **0–255**, debido a que el módulo es ánodo común se debe elaborar una fórmula en donde obtengamos los valores invertidos, es decir 0 equivalga a 255 y 1 equivalga a 0, se procede a establecer la relación mediante la ecuación mostrada a continuación:

Ecuación 3: Conversión de color Rojo práctica 2.

$$\text{Duty Cycle R} = \left| -\left(\frac{R}{255} - 1\right) \right|$$

Ecuación 4: Conversión de color Verde práctica 2.

$$\text{Duty Cycle G} = \left| -\left(\frac{G}{255} - 1\right) \right|$$

Ecuación 5: Conversión de color Azul práctica 2.

$$\text{Duty Cycle B} = \left| - \left(\frac{B}{255} - 1 \right) \right|$$

Donde R, G y B son los valores (0-255) resultantes de la conversión realizada por el subVI **Color to RGB**.

15. Para activar el LED RGB, utilizar un bloque de escritura digital PWM (**Set Duty Cycle**) de N canales y seleccione los puertos digitales 4,5 y 6, ya que este módulo utiliza 3 parámetros para mostrar el color deseado.

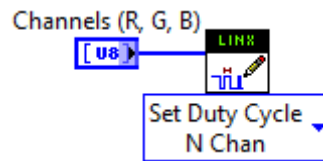


Figura 107: Bloque de Escritura Digital LED RGB práctica 2.

16. Para recibir la señal del pulsador "**Paro físico**", utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 45.

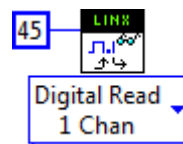


Figura 108: Bloque de Lectura Digital sensor táctil práctica 2.

17. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 109: Bloque de cierre de comunicación práctica 2.

18. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

19. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 110: Bloque de apertura de conexión de Database práctica 2.

20. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 111: Bloque para inserta datos en Database práctica 2.

21. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 112: Bloque de cierre de conexión de Database práctica 2.

CONFIGURACIÓN DE DATABASE

22. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

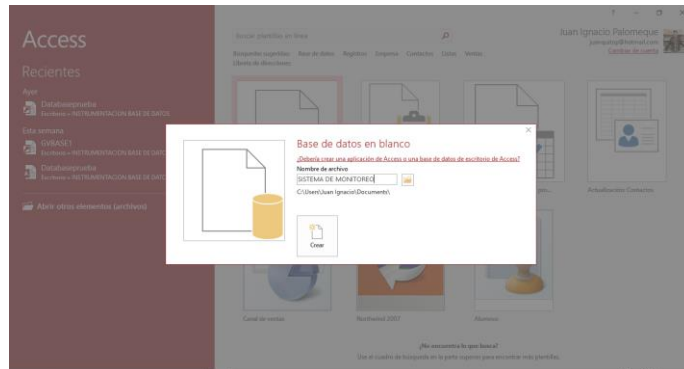


Figura 113: Creación de base de datos en MICROSOFT ACCESS práctica 2.

23. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESSS 2002-2003**.

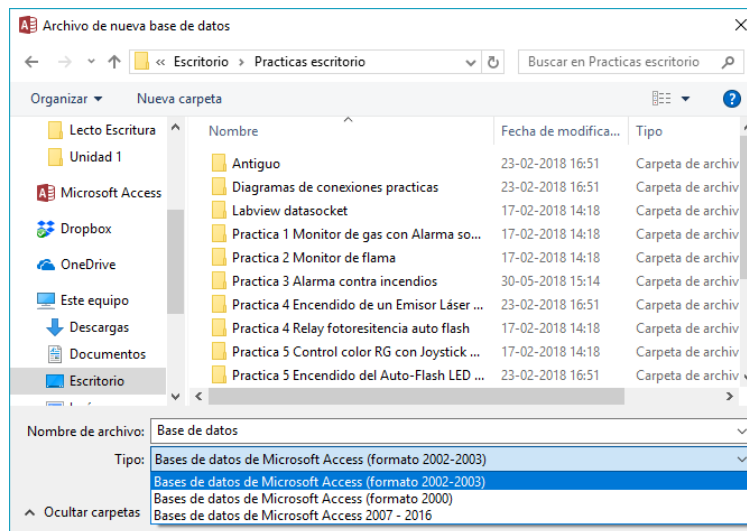


Figura 114: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 2.

24. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

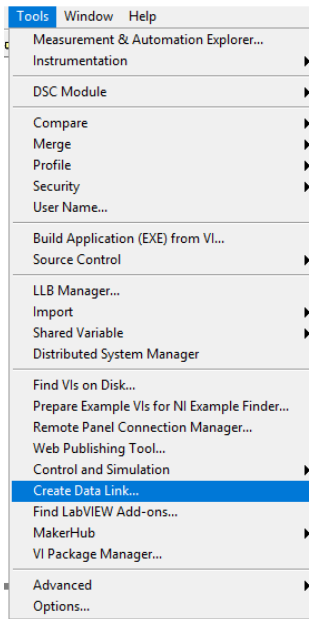


Figura 115: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 2.

25. Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

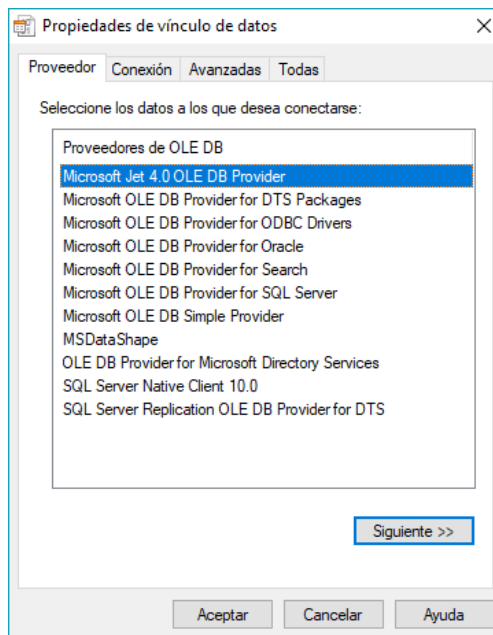


Figura 116: Ventana de propiedades de vínculo de datos práctica 2.

26. Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

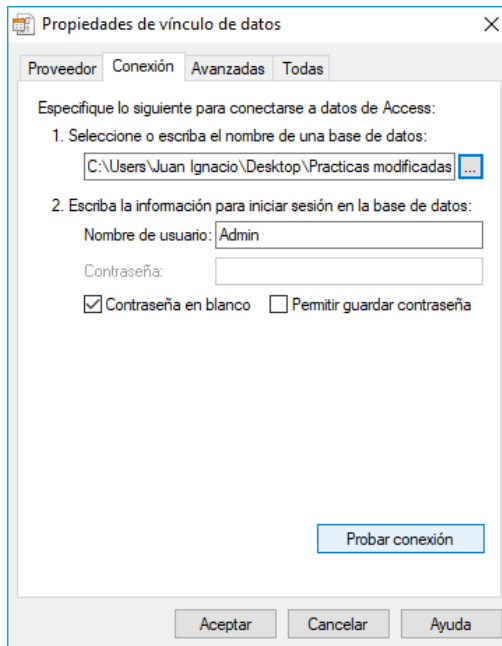


Figura 117: Selección de ruta donde se ubica la base de datos práctica 2.

27. Una vez enlazada la ruta se procede a probar conexión.

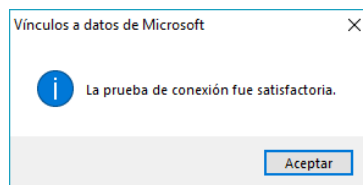


Figura 118: Venta de prueba de conexión de la vinculación de datos práctica 2.

28. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

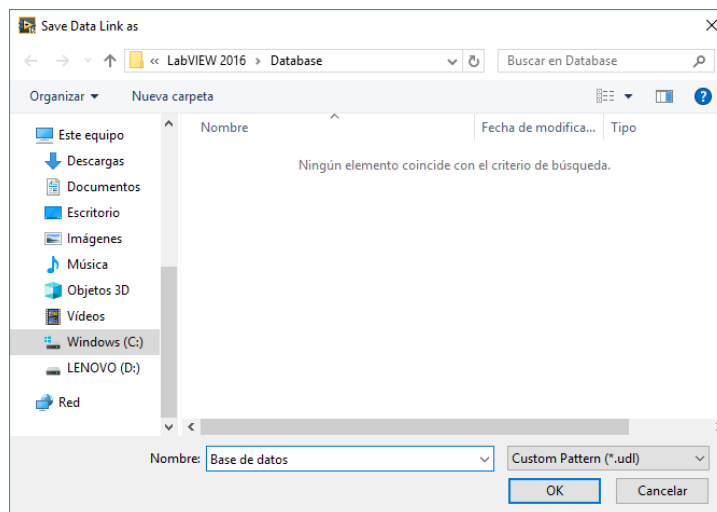


Figura 119: Ventana de asignación de nombre para la tabla de registro de datos práctica 2.

29. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

connection information



Figura 120: Control de ruta de base de datos práctica 2.

30. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:

- Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard**
- En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next**.
- Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next**.
- Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next**.
- Verificar que los LEDs **TX** y **RX** deben permanecer encendidos durante unos segundos mientras se implementa el firmware.
- Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.

31. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

32. Dar clic en el botón **Run Continuously**.

33. Acercar de manera manual una llama frente al Sensor de Flama.

34. Observar el comportamiento a través de la pantalla del módulo de las salidas analógica y digital del Sensor de Flama.

35. Comprobar la activación de la alarma sonora mediante el funcionamiento del Buzzer Activo.

36. Comprobar la activación del indicador lumínico a través del cambio de color del LED RGB.

37. Utilizar el Sensor Táctil en caso de querer interrumpir el programa de manera definitiva.

DIAGRAMA DE CONEXIONES

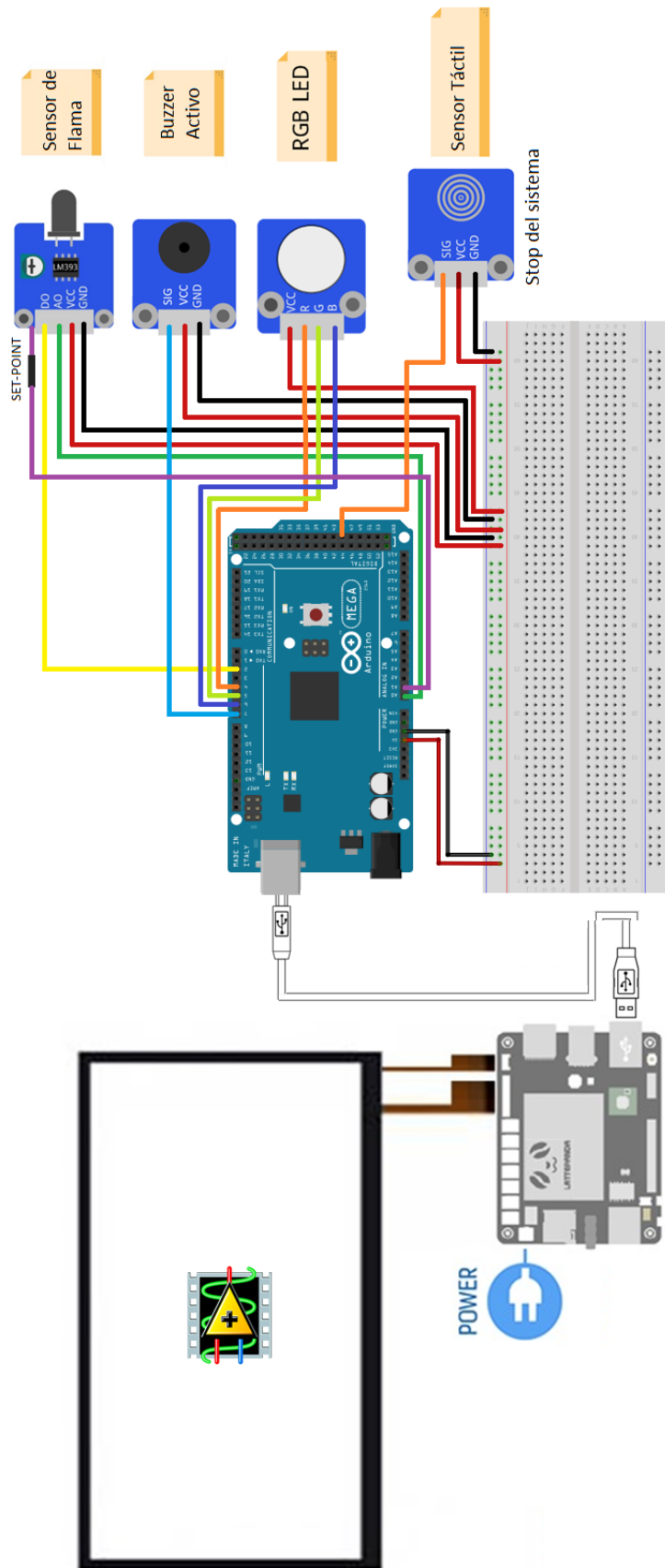


Figura 121: Diagrama de conexiones de la Práctica 2.

Tabla 3: Conexiones de la Práctica 2.

| SENSOR DE FLAMA | ARDUINO MEGA 2560 |
|-----------------|-------------------|
| D0 | 2 |
| A0 | 0 |
| A1(set-point) | 1 |
| VCC | 5V |
| GND | GND |
| BUZZER ACTIVO | ARDUINO MEGA 2560 |
| SIG | 7 |
| VCC | 5V |
| GND | GND |
| RGB LED | ARDUINO MEGA 2560 |
| VCC | 5V |
| R | 4 |
| G | 5 |
| B | 6 |
| SENSOR TÁCTIL | ARDUINO MEGA 2560 |
| SIG | 45 |
| VCC | 5V |
| GND | GND |

DISEÑOS EN LABVIEW

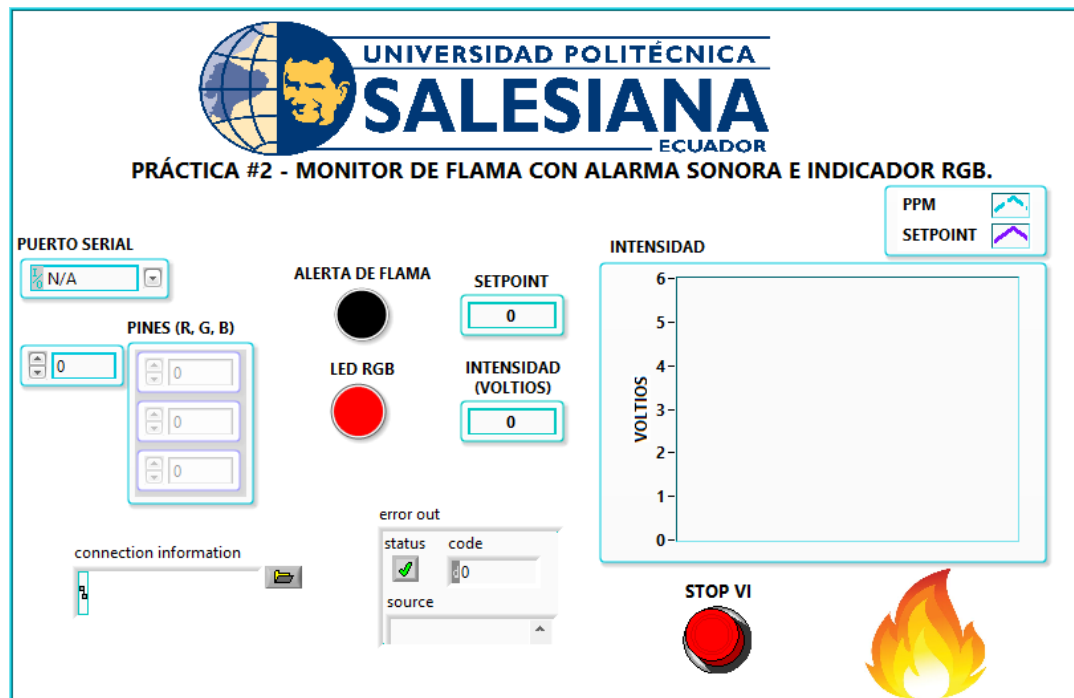


Figura 122: Panel Frontal de la Práctica 2.

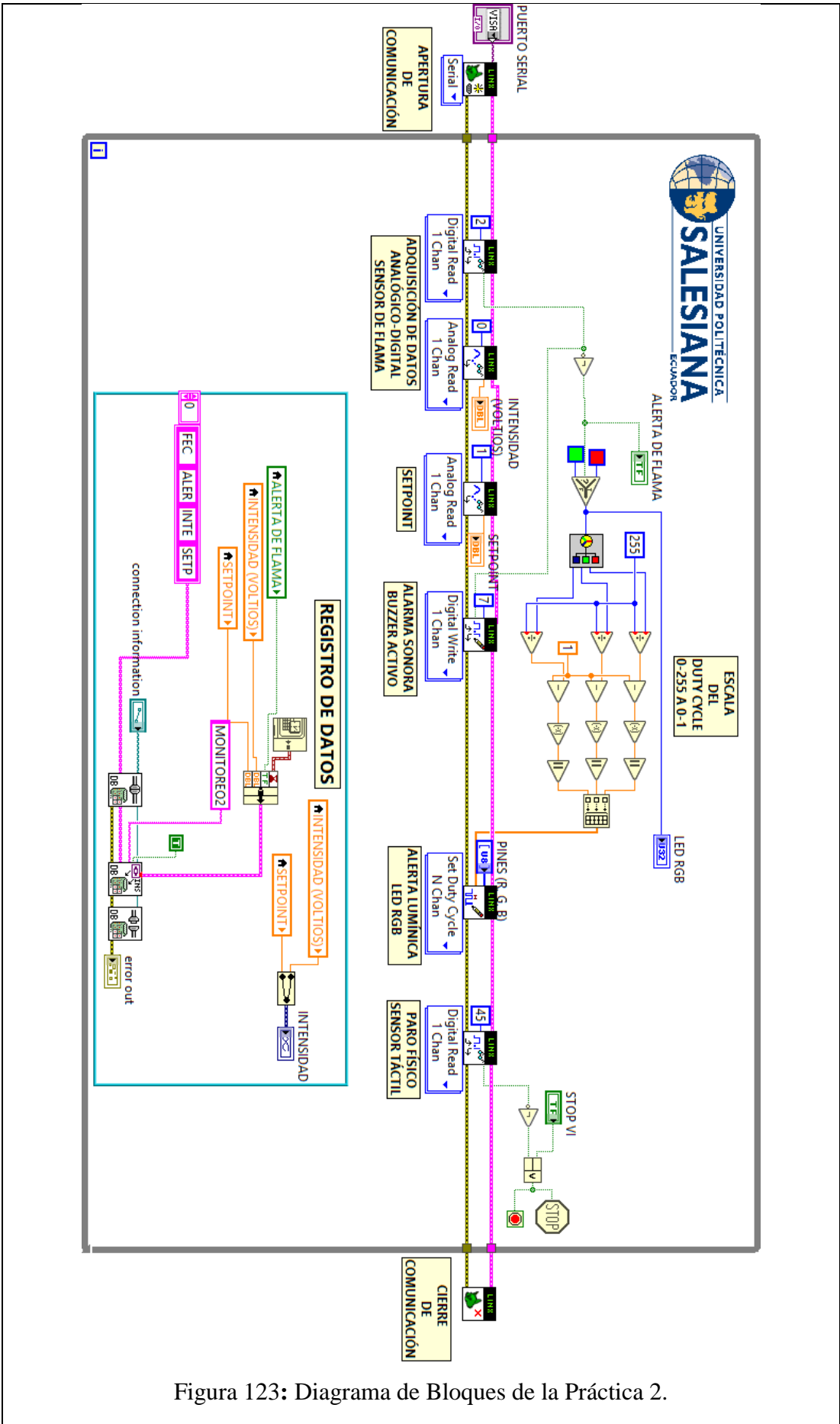


Figura 123: Diagrama de Bloques de la Práctica 2.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.
3. Se comprobó el funcionamiento del Led RGB mediante su activación en distintos eventos.
4. Gracias a las salidas que posee el Sensor de Flama se pueden mostrar y almacenar valores analógicos y digitales generados por el usuario.
5. Utilizando el potenciómetro que posee el Sensor de Flama se logró establecer un Set-Point que condiciona la activación del Buzzer Activo.
6. Se comprobó el funcionamiento del Sensor de Flama mediante la exposición del mismo frente a una llama.

ANALISIS DE RESULTADO:

Cuando el Sensor de Flama detecta la presencia de flama, se produce un cambio en el valor de voltaje analógico entregado mediante el pin A0, dicho valor es entregado en voltio y puede ser observado en la gráfica ubicada en el panel frontal.

Si el valor leído por el Sensor de Flama excede el fijado mediante el potenciómetro o Set-Point, se emite un pulso mediante el pin D0 del módulo.

El Buzzer Activo se acciona en el momento en el que el valor medido excede el Set-Point.

El color del Led RGB se mantiene en verde siempre y cuando no se detecte la presencia de flama.

Cuando el Sensor Táctil es activado el VI deja de ejecutarse.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

Todos los datos adquiridos tales como lectura de presencia de flama, Set-Point, presencia de flama se registran en una base de datos en tiempo real.

4.3. Práctica # 3


| | | |
|---|---|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 3 | TÍTULO PRÁCTICA: Control y monitoreo de un sistema contra incendio. |
| Objetivo: Diseñar un programa que permita monitorear u observar desde otro ordenador el comportamiento de un sistema contra incendio. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar el Sensor de Flama y Sensor de Gas MQ-2 en el software LabVIEW 2016. Configurar una alarma sonora con el Buzzer Pasivo. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Mediante el uso de la herramienta DataSocket, crear una red donde se pueda monitorear una base de datos donde registren los eventos y datos obtenidos por el sensor. | | |
| INSTRUCCIONES: | | <ol style="list-style-type: none"> 1. Alimentar el módulo principal con 110v. 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. 3. Seguir la guía paso a paso para realizar la conexión del módulo. 4. Realizar una base de datos en ACCESS para registrar los eventos y datos obtenidos por el sensor. |
| MARCO TEORICO | | |
| Sensor de Flama Introducción Un módulo de sensor de flama realiza la detección capturando las longitudes de onda infrarrojas de la llama. Se puede usar para detectar y advertir llamas. En este experimento, se utiliza el sensor para hacer una alarma de llama (Sunfounder, 2013). | | |



Figura 124: Sensor de Flama práctica 3. (Sunfounder, 2013).

Principio experimental

Hay varios tipos de sensores de flama. En este experimento, se utiliza un sensor de flama de infrarrojo lejano. Puede detectar la luz infrarroja con una longitud de onda de 700 nm a 1000 nm. Una sonda de llama de infrarrojo lejano convierte los cambios de fuerza de la luz infrarroja externa en cambios actuales. Y luego convierte las cantidades analógicas en digitales (Sunfounder, 2013).

Sensor de Gas MQ-2

Introducción

El sensor de gas MQ-2 es un sensor para gases inflamables y humo al detectar la concentración de gas combustible en el aire. A menudo se utilizan en equipos de detección de gases para el humo y gases inflamables como el GLP, el i-butano, el propano, el metano y el alcohol en el hogar, la industria o el automóvil (Sunfounder, 2013).



Figura 125: Sensor de Gas práctica 3. (Sunfounder, 2013)

Principio experimental

El sensor de gas MQ-2 aplica SnO₂ (un semiconductor de tipo n deficiente en oxígeno) que tiene una conductividad más baja en el aire claro como un material sensor de gas. En una atmósfera donde puede haber gases inflamables, aumenta la conductividad del sensor de gas junto con la concentración de gas inflamable. MQ-2 realiza una buena detección de diferentes gases inflamables, como el gas natural, especialmente sensible al gas licuado, al propano y al hidrógeno (Sunfounder, 2013).

En este experimento, suelte un poco de gas combustible alrededor del sensor y el valor en A0 aumenta. Una vez que la concentración del gas excede un límite, D0 produce un bajo nivel y se enciende el LED correspondiente. Tenga en cuenta que el sensor necesita tiempo de grabación para que las lecturas puedan ser más consistentes y confiables (Sunfounder, 2013).

Buzzer Pasivo

Introducción

Un buzzer o zumbador es un dispositivo de señalización de audio. Los zumbadores pueden clasificarse en activos y pasivos (Sunfounder, 2013).

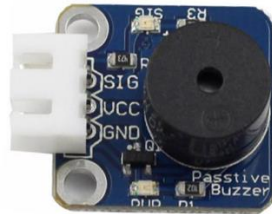


Figura 126: Buzzer Pasivo práctica 3. (Sunfounder, 2013).

Principio experimental

Un zumbador activo tiene una fuente oscilante incorporada, por lo que emite sonidos cuando se electrifique. El zumbador activo a menudo es más caro que el pasivo debido a múltiples circuitos oscilantes incorporados (Sunfounder, 2013).

Pulsador Manual (Botón)

Introducción

Un botón es un mecanismo simple de cambio de estado, para controlar algún aspecto de un equipo o un proceso. Los botones suelen estar fabricados de materiales duros, generalmente de plástico o metal, de superficie plana o con forma para acomodar el dedo o la mano humana, para que pueda presionarse o empujarse con facilidad.



Figura 127: Pulsador Manual práctica 3. (Sunfounder, 2013).

Principio experimental

Cuando el módulo de botón identifica las señales inductivas de presionar botones, el LED estará encendido. De lo contrario, se apagará.

Sensor Táctil

Introducción

Un sensor táctil de metal es un tipo de interruptor que solo funciona cuando es tocado por un cuerpo cargado. Tiene una alta frecuencia transistor que se activa cuando recibe una señal electromagnética (Sunfounder, 2013).

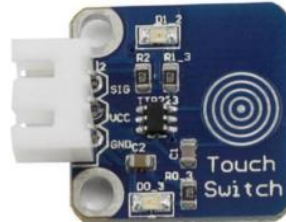


Figura 128: Sensor Táctil práctica 3. (Sunfounder, 2013).

Principio experimental

Cuando el electrodo de base del transistor es tocado se produce la conducción de electricidad a través del mismo, esto debido a que el cuerpo humano es una especie de conductor y una antena que puede captar las ondas electromagnéticas que circulan en el aire (Sunfounder, 2013).

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Sensor de MQ-2** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Conectar el **Sensor de Flama** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
4. Conectar el **Buzzer Pasivo** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
5. Conectar el **Pulsador Manual (Botón)** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
6. Conectar el **Sensor Táctil** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
7. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
8. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.
9. Para utilizar los bloques de las funciones de **LINX**, seguir los siguientes pasos:
 - Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
 - Seleccionar el submenú **MakerHub**.
 - Seleccionar el submenú **LINX**.
10. Para establecer la comunicación, colocar el bloque **OPEN** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 129: Bloque de apertura de comunicación práctica 3.

11. Para visualizar los datos obtenidos por el sensor de gas, utilizar los siguientes bloques:

DATO ANALOGICO: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 1.

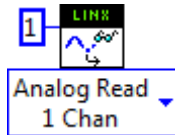


Figura 130: Bloque de Lectura Analógico sensor de gas práctica 3.

SET POINT: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 3.

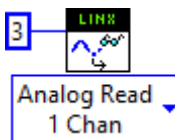


Figura 131: Bloque de Lectura Analógico SET POINT del sensor de gas práctica 3.

DATO DIGITAL: Utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 6. Se recomienda colocar un inversor a la salida del bloque debido a que el mismo funciona con lógica negativa.

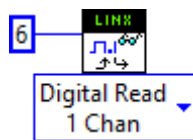


Figura 132: Bloque de Lectura Digital sensor de gas práctica 3.

Nota: Para determinar la concentración de Gas en partes por millón se debe realizar el siguiente cálculo referido al Gas LP.

Ecuación 6: Concentración de Gas en PPM práctica 3.

$$\text{Concentración} = 10^{(A \times \log(\frac{RS}{R}) + B)}$$

Para determinar la concentración se necesita la recta que la aproxima, para lo cual se debe seleccionar dos puntos cuales quiera de la gráfica $P_0=(X_0,Y_0)$ y $P_1=(X_1,Y_1)$, resultando la ecuación de la recta:

Ecuación 7: Ecuación de la recta práctica 3.

$$Y = Ax + B$$

Siendo $A = \frac{Y1-Y0}{X1-X0}$ & $B = Y0 - AX0$

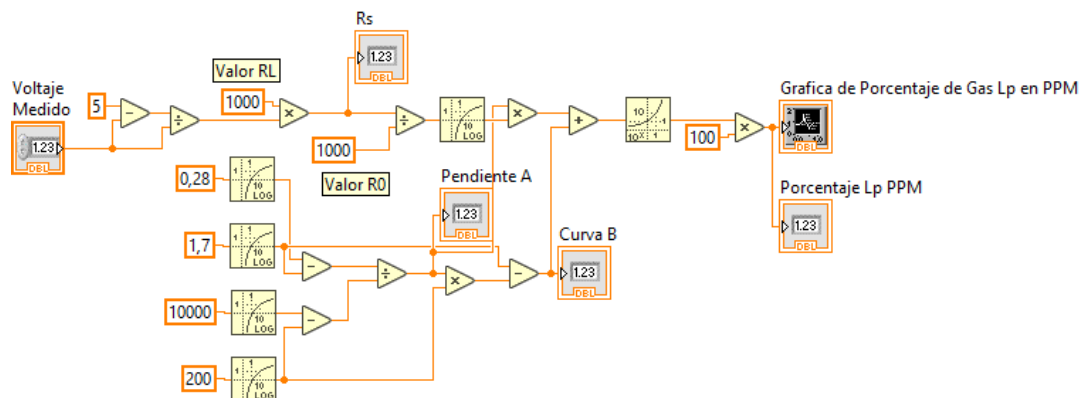


Figura 133: Cálculo de concentración de Gas en PPM.

Para realizar el cálculo del Set-point utilizar la fórmula de la ecuación 6.

12. Para visualizar los datos obtenidos por el sensor de flama, utilizar los siguientes bloques:

DATO ANALOGICO: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 0.

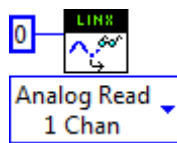


Figura 134: Bloque de Lectura Analógica sensor de flama práctica 3.

SET POINT: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 2.

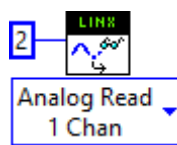


Figura 135: Bloque de Lectura Analógico SET POINT del sensor de flama práctica 3.

DATO DIGITAL: Utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 2. Se recomienda colocar un inversor a la salida del bloque debido a que el mismo funciona con lógica negativa.

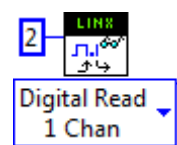


Figura 136: Bloque de Lectura Digital sensor de flama práctica 3.

13. Para recibir la señal del pulsador **Alarma manual**, utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 4.

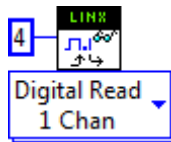


Figura 137: Bloque de Lectura Digital pulsador manual práctica 3.

14. Para activar el Buzzer Pasivo, utilizar un bloque PWM (**Set Duty Cycle**) y seleccione el puerto digital número 5. Se debe utilizar este bloque debido a las propiedades que posee el buzzer pasivo, en el pin Duty Cycle se conectará la salida del selector antes colocado.



Figura 138: Bloque de Escritura Digital buzzer pasivo práctica 3.

15. Para recibir la señal del pulsador "**Paro físico**", utilizar un bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 45.

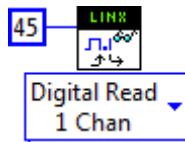


Figura 139: Bloque de Lectura Digital sensor táctil práctica 3.

16. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 140: Bloque de cierre de comunicación práctica 3.

CONFIGURACIÓN DATASOCKET

VI ESCRITURA

17. Revisar la dirección IP asignado al terminal en el cual se encuentran conectados los elementos.

18. Para que el VI de lectura pueda monitorear los datos obtenidos por el VI Escritura se utiliza herramienta DataSocket.

19. Configurar cada indicador ubicado en el VI de escritura de la siguiente manera:

- Clic derecho sobre el indicador
- Seleccionar la opción **Propiedades**.
- Seleccionar la pestaña **Data Binding**.
- En la casilla **Data Binding Selection** seleccionar la opción **DataSocket**.
- En la casilla **Access Type** se define la función del indicador que se está configurando, para esta práctica seleccionar la opción **Read/Write**.
- En el text box **Path** se debe escribir la dirección URL utilizando la IP consultada antes y el nombre del indicador configurado:

"dstp://IP del terminal de lectura/Nombre del indicador"

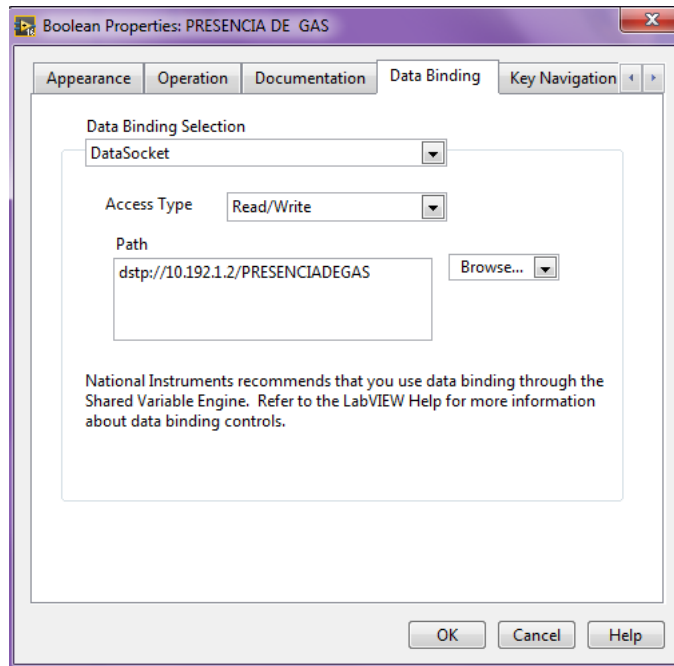


Figura 141: Configuración de data socket en VI de Escritura práctica 3.

VI DE LECTURA

20. Copiar los elementos indicadores del VI de lectura al VI en el panel frontal.

21. Configurar cada indicador ubicado en el VI de lectura de la siguiente manera:

- Clic derecho sobre el indicador
- Seleccionar la opción **Propiedades**.
- Seleccionar la pestaña **Data Binding**.
- En la casilla **Data Binding Selection** seleccionar la opción **DataSocket**.
- En la casilla **Access Type** se define la función del indicador que se está configurando, para los indicadores de este VI se debe elegir la opción **Read Only**, esto debido a que solo se va a monitorear los datos obtenidos.
- En el text box **Path** se debe escribir la dirección URL utilizando la IP del terminal de lectura y el nombre del indicador del que se obtendrán los datos:

“dstp://IP del terminal de lectura/Nombre del indicador”

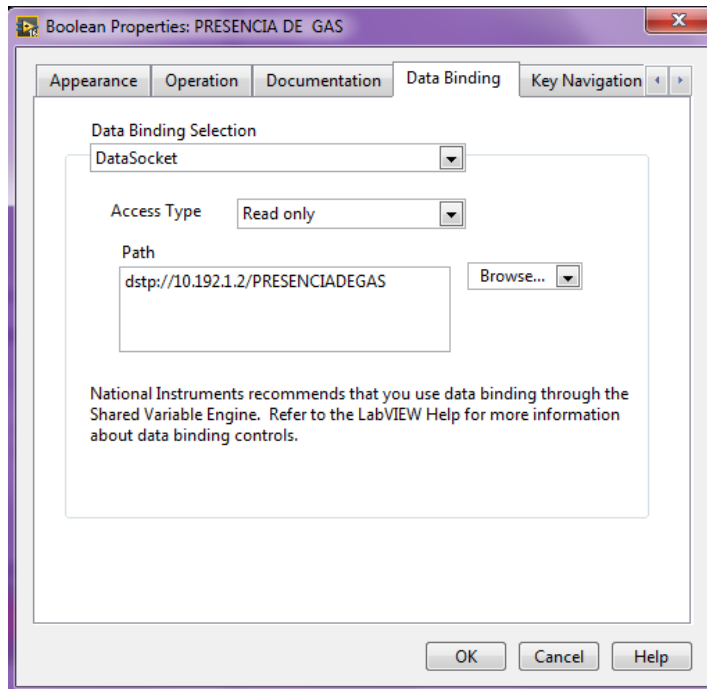


Figura 142: Configuración de data socket en VI de Lectura práctica 3.

- Para que un indicador realice la tarea de lectura y escritura en ambos VI's se debe crear el mismo con función Write en el vi de lectura, y ocultarlo en el vi de escritura.
- Configurar el data socket manager para permitir el control de cualquier host.

22. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

23. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 143: Bloque de apertura de conexión de Database práctica 3.

24. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 144: Bloque para inserta datos en Database práctica 3.

25. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 145: Bloque de cierre de conexión de Database práctica 3.

CONFIGURACIÓN DE DATABASE

26. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

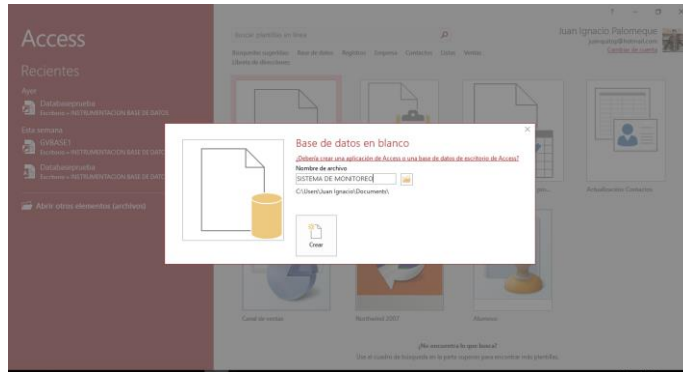


Figura 146: Creación de base de datos en **MICROSOFT ACCESS** práctica 3.

27. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESSS 2002-2003**.

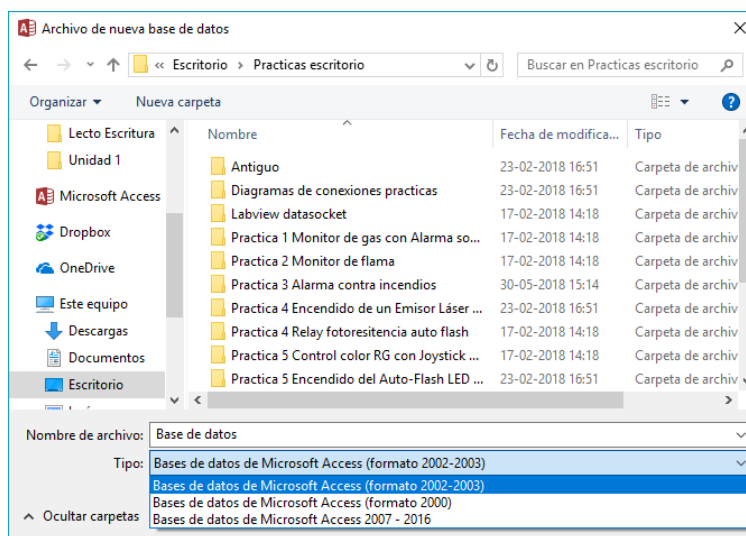


Figura 147: Creación de base de datos formato **MICROSOFT ACCESS 2002-2003** práctica 3.

28. Una vez creada la Data base, establecer un data link entre **LabVIEW 2016** y la base de datos creada en **MICROSOFT ACCESS 2016**.

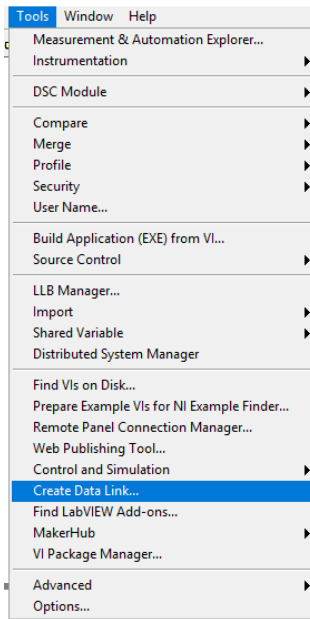


Figura 148: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 3.

29. Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

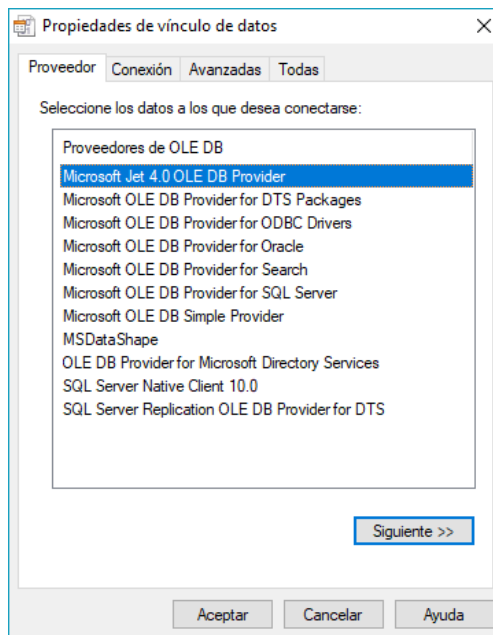


Figura 149: Ventana de propiedades de vínculo de datos práctica 3.

30. Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

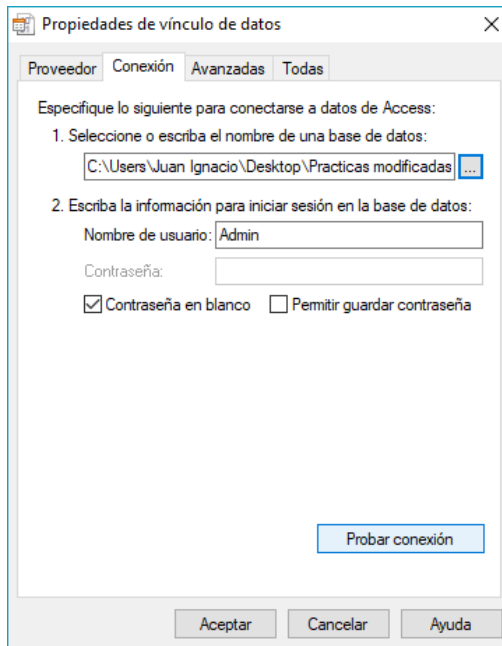


Figura 150: Selección de ruta donde se ubica la base de datos práctica 3.

31. Una vez enlazada la ruta se procede a probar conexión.

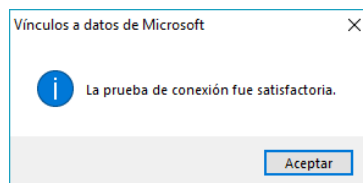


Figura 151: Venta de prueba de conexión de la vinculación de datos práctica 3.

32. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

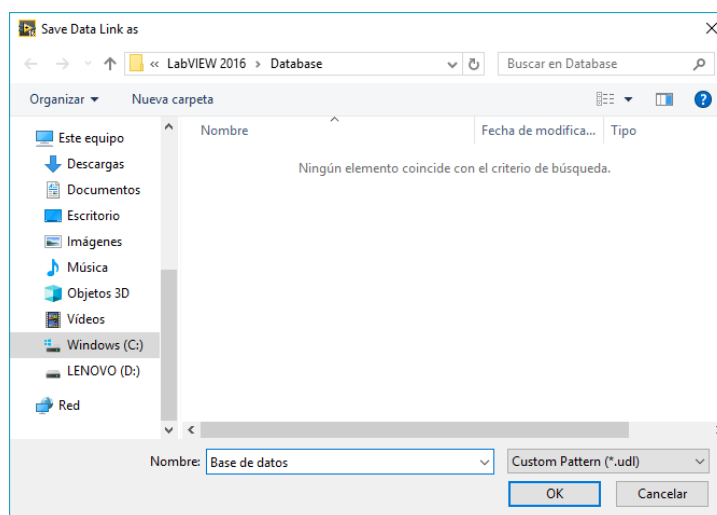


Figura 152: Ventana de asignación de nombre para la tabla de registro de datos práctica 3.

33. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

connection information



Figura 153: Control de ruta de base de datos práctica 3.

CARGA DEL VI DE ESCRITURA

34. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:

- Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard**
- En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next**.
- Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next**.
- Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next**.
- Verificar que los LEDs **TX y RX** deben permanezcan encendidos durante unos segundos mientras se implementa el firmware.
- Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.

35. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

36. Dar clic en el botón **Run Continuously**.

37. Liberar manualmente gas propano frente sensor de gas, para procesar los datos obtenidos por el mismo.

38. Observar la activación de los indicadores ubicados en el panel frontal y de la alarma sonora mediante el Buzzer.

39. Acercar lentamente una llama frente sensor de flama, para procesar los datos obtenidos por el mismo.

40. Observar la activación de los indicadores ubicados en el panel frontal y de la alarma sonora mediante el Buzzer.

41. Pulsar el Botón para activar el indicador ubicado en el panel frontal y la alarma sonora mediante el Buzzer.

42. La alarma sonora funciona bajo las tres acciones expuestas anteriormente.

43. La activación del Buzzer es controlada por el estado del indicador “Alarma Sonora”.

44. La modulación del sonido del Buzzer se realiza de manera manual mediante el scroll colocado en el panel frontal.

45. Utilizar el botón virtual Reinicio para desactivar la alarma.

46. Utilizar el sensor táctil en caso de querer interrumpir el programa de manera definitiva.

CARGA DEL VI DE LECTURA

47. Dar clic en el botón **Run Continuously**.

48. Confirme que el VI de lectura pueda monitorear los datos obtenidos por el VI Escritura.

49. Confirme el almacenamiento de eventos realizados por los sensores en la base de datos, creada en MICROSOFT ACCESS 2016.

DIAGRAMA DE CONEXIONES

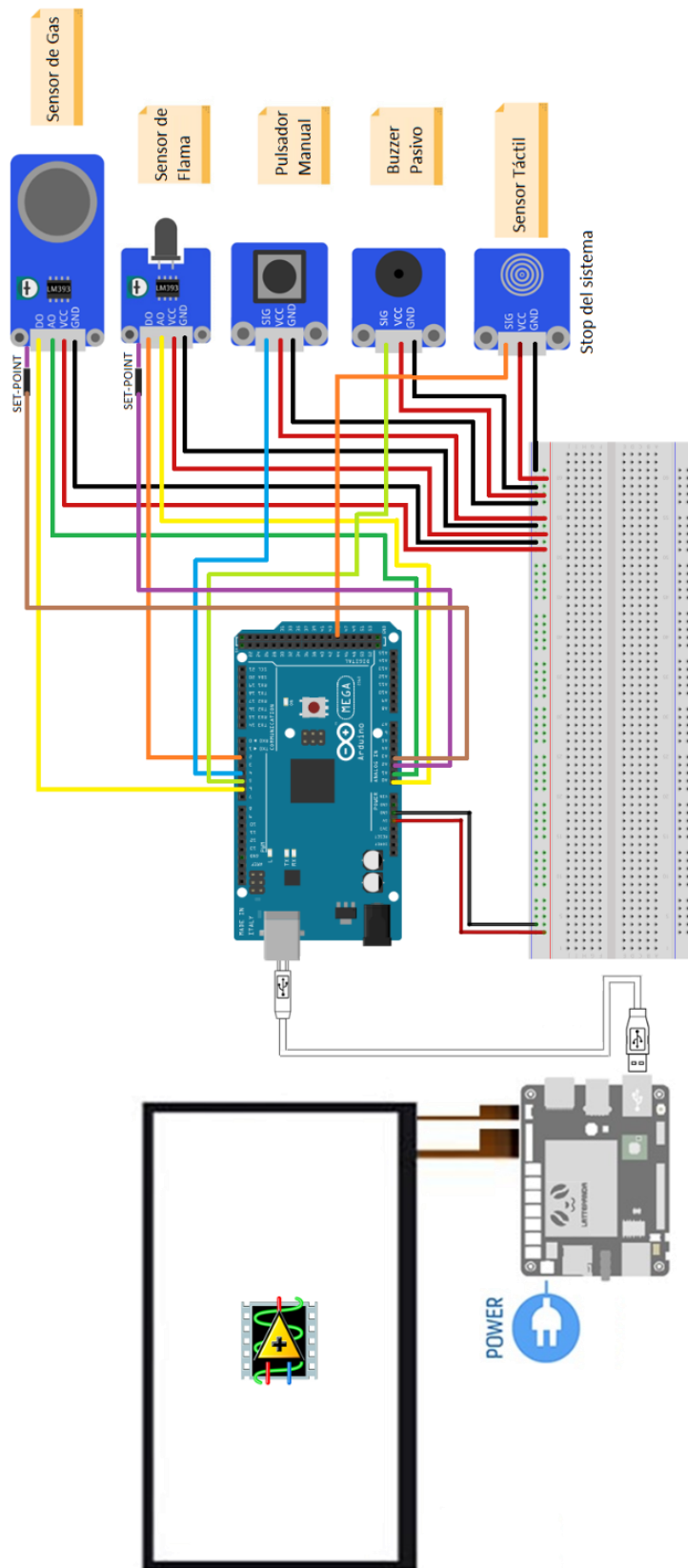
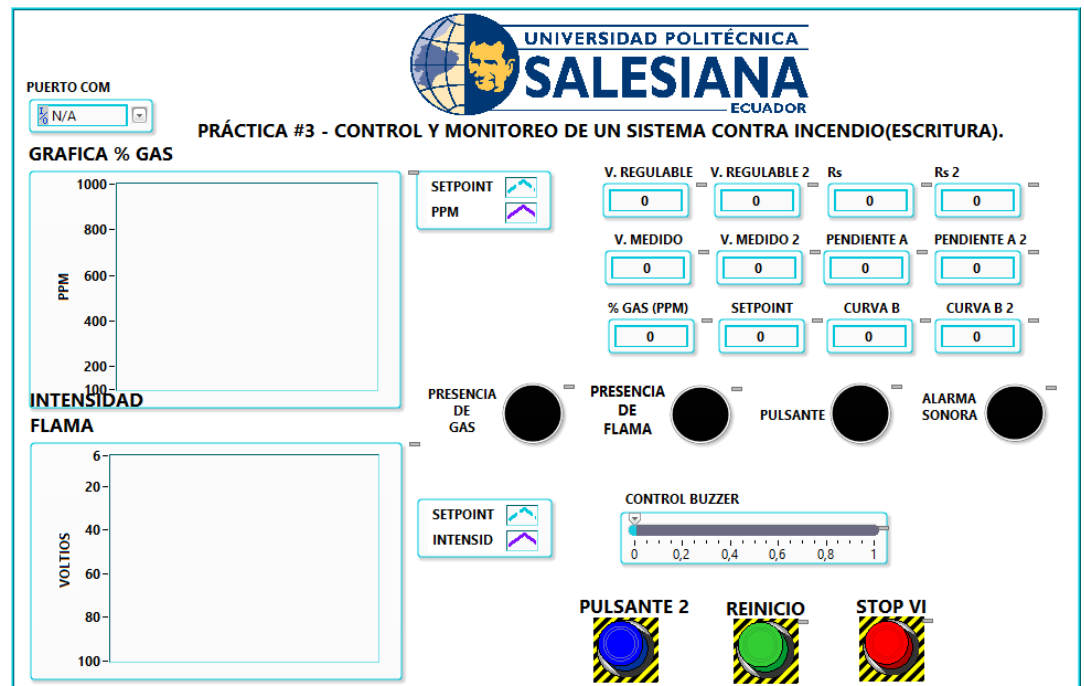


Figura 154: Diagrama de conexiones de la Practica 3.

Tabla 4: Conexiones de la Practica 3.

| | |
|------------------------|--------------------------|
| SENSOR DE FLAMA | ARDUINO MEGA 2560 |
| D0 | 2 |
| A0 | A0 |
| A1 (set point) | A2 |
| VCC | 5V |
| GND | GND |
| SENSOR DE GAS | ARDUINO MEGA 2560 |
| D0 | 6 |
| A0 | A1 |
| A1 (set point) | A3 |
| VCC | 5V |
| GND | GND |
| BUZZER PASIVO | ARDUINO MEGA 2560 |
| SIG | 5 |
| VCC | 5V |
| GND | GND |
| PULSADOR MANUAL | ARDUINO MEGA 2560 |
| SIG | 4 |
| VCC | 5V |
| GND | GND |
| SENSOR TÁCTIL | ARDUINO MEGA 2560 |
| SIG | 45 |
| VCC | 5V |
| GND | GND |

DISEÑOS EN LABVIEW



1. Revisar la dirección IP.
2. La dirección IP a utilizar debe ser la del equipo que lee la información
3. Cada indicador debe tener su propio nombre
4. Cada indicador se configura en propiedades-data binding opción datasocket
5. ejemplo de nombre [dstp://192.168.0.105/nombredensensor](http://192.168.0.105/nombredensensor)
6. en el vi de escritura los indicadores pueden ser read/write
7. en el vi de lectura los indicadores deben ser solo read
8. para que un indicador controle ambos vi se debe crear el mismo con función write en el vi de lectura, y ocultarlo en el vi de escritura.
9. se debe configurar data socket manager para permitir el control de cualquier host.

Figura 155: Panel Frontal VI de Escritura de la Práctica 3.

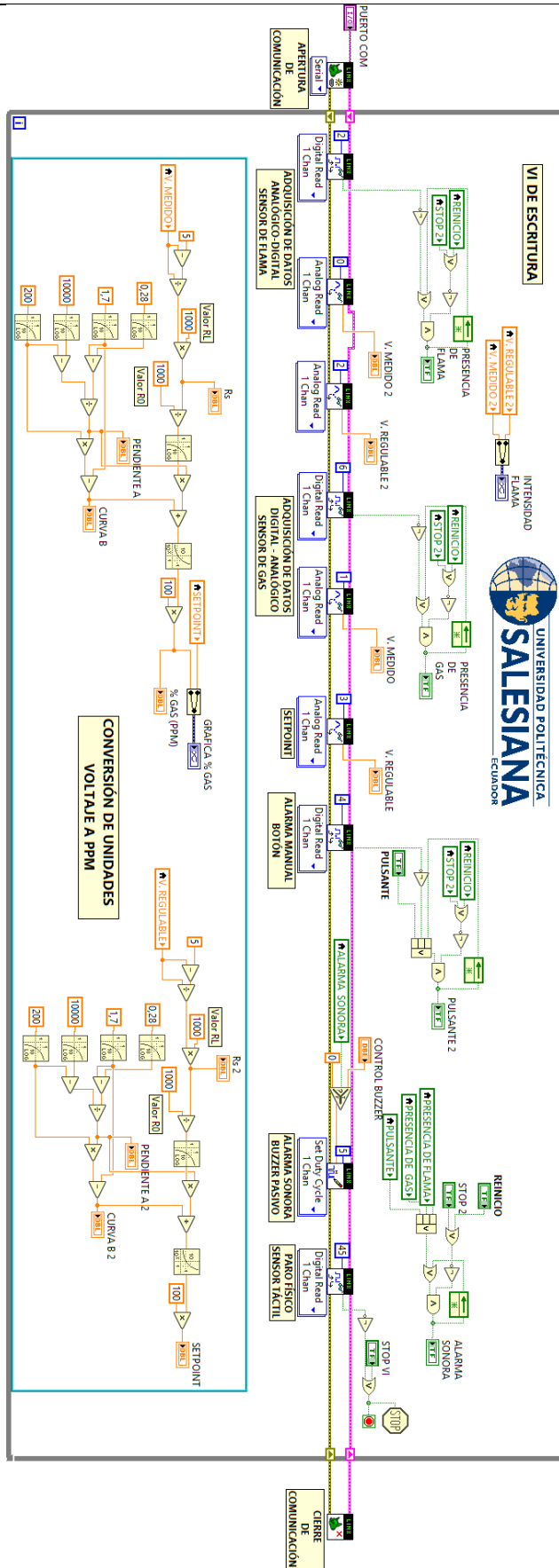


Figura 156: Diagrama de Bloques VI de Escritura de la Práctica 3.

PRÁCTICA #3 - SISTEMA CONTRA INCENDIO (LECTURA).

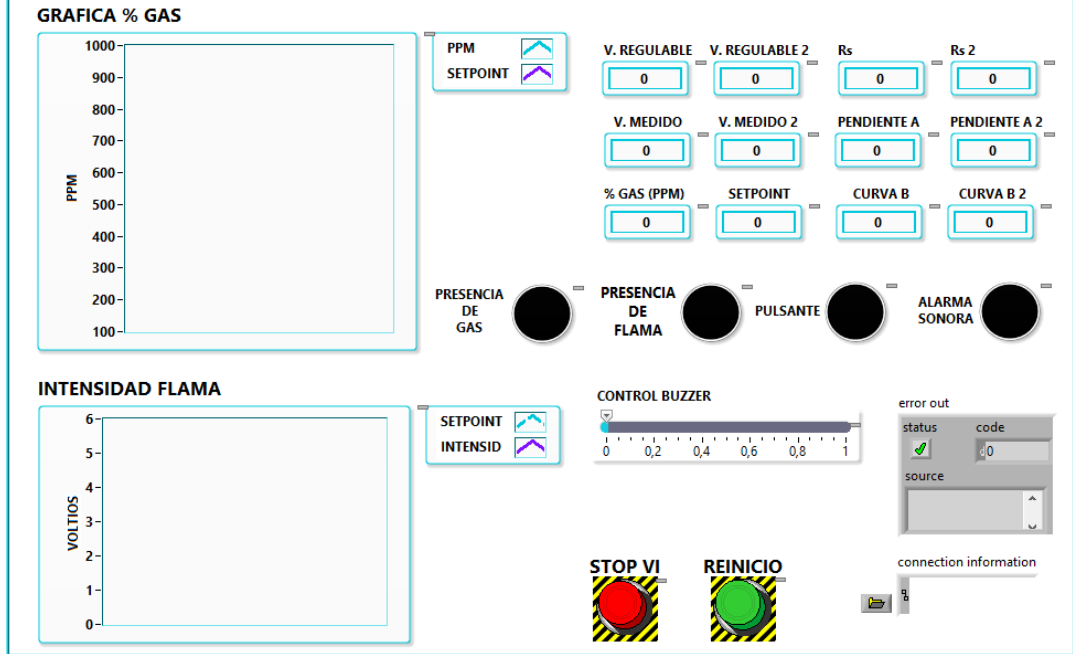


Figura 157: Panel Frontal VI de Lectura de la Práctica 3.

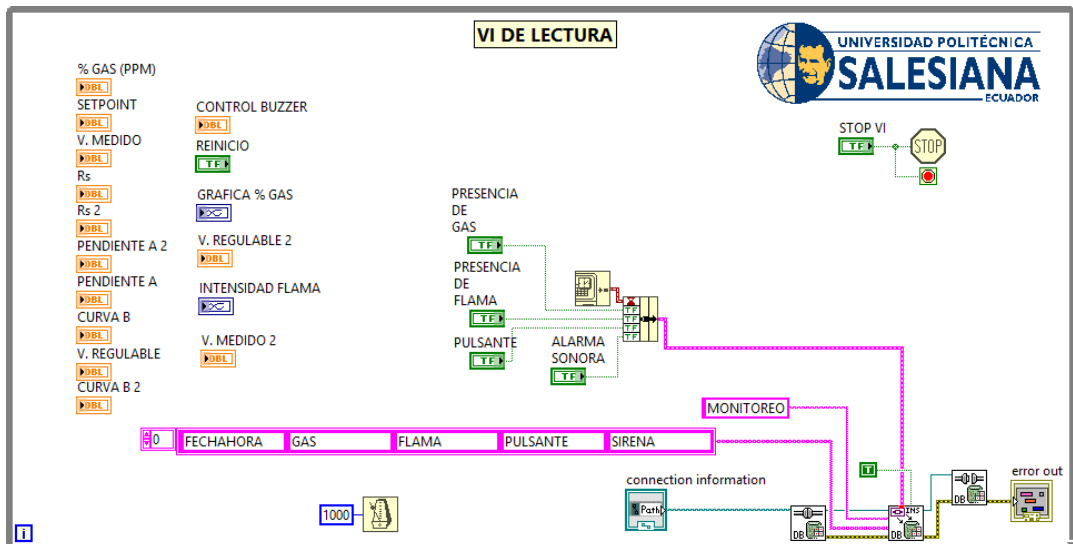


Figura 158: Diagrama de Bloques VI de Lectura de la Práctica 3.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.
3. Mediante el uso de la herramienta DataSocket se logró implementar un sistema de comunicación versátil capaz de monitorear de manera remota desde otro terminal ya sea a través de las redes Ethernet o WiFi emulando los sistemas de monitoreo implementados en algunas industrias.
4. Gracias al uso de los sensores se logró emular en funcionamiento de un sistema contra-incendios.
5. Se logró diseñar un sistema de monitoreo remoto, el cual registra y permite visualizar al usuario los eventos en tiempo real.

ANALISIS DE RESULTADO:

Cuando el Sensor de Gas MQ-02 detecta la presencia de gas propano en el ambiente se produce un cambio en el valor de voltaje analógico entregado mediante el pin A0, dicho valor convertido a PPM puede ser observado en la gráfica ubicada en el panel frontal.

Si la cantidad de gas que se encuentra en el ambiente excede la cantidad definida mediante el potenciómetro o Set-Point, se activa el Buzzer Pasivo.

Cuando el Sensor de Flama detecta la presencia de flama, se produce un cambio en el valor de voltaje analógico entregado mediante el pin A0, dicho valor es entregado en voltio y puede ser observado en la gráfica ubicada en el panel frontal.

Si el valor leído por el Sensor de Flama excede el fijado mediante el potenciómetro o Set-Point, se emite un pulso mediante el pin D0 del módulo, activando el Buzzer Pasivo.

Cuando se presiona el pulsador, este emite un pulso que permite la activación del Buzzer Pasivo.

Cuando se presiona el botón denominado "Reinicio" haciendo uso de la pantalla táctil, la alarma sonora se apaga.

4.4. Práctica # 4


| | | |
|--|---|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 4 | TÍTULO PRÁCTICA: Encendido de un Emisor Láser. |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual accione un Emisor Láser cuando la luz no esté presente en el ambiente. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar la Foto-resistencia software LabVIEW 2016 para que active el funcionamiento de un módulo Relé que será el encargado activar el Emisor Láser. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Elaborar una base de datos donde registren los eventos y datos obtenidos por el sensor. | | |
| INSTRUCCIONES: | | 1. Alimentar el módulo principal con 110v. |
| | | 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. |
| | | 3. Seguir la guía paso a paso para realizar la conexión del módulo. |
| | | 4. Realizar una base de datos en ACCESS para registrar los eventos y datos obtenidos por el sensor. |
| MARCO TEORICO | | |
| Emisor láser Introducción El láser es ampliamente utilizado en tratamientos médicos, militares y otros campos debido a su buena directividad y concentración de energía. El módulo del transmisor láser, como su nombre indica, tiene la capacidad de emitir luz láser (Sunfounder, 2013). | | |



Figura 159: Emisor Láser práctica 4. (Sunfounder, 2013).

Principio experimental

Un emisor láser es un dispositivo que irradia luz a través de un proceso de amplificación óptica basado en la emisión de radiación electromagnética (Sunfounder, 2013). Los láseres difieren de otras fuentes de luz porque emiten luz de forma de fase constante (Sunfounder, 2013).

La coherencia espacial permite enfocar un láser en un lugar apretado, lo que permite que aplicaciones como el corte por láser y la litografía, y un rayo láser se mantengan estrechas en largas distancias (colimación), lo que permite aplicaciones como el puntero láser (Sunfounder, 2013). Los láseres también pueden tener una alta relación temporal que les permite tener un espectro muy estrecho, es decir, solo emiten luz de un solo color (Sunfounder, 2013). Y su coherencia temporal puede usarse para producir pulsos de luz, tan cortos como un femtosegundo (Sunfounder, 2013).

Relé

Introducción

El relé es un dispositivo que se utiliza para proporcionar la conexión entre dos o más puntos o dispositivo en respuesta a la señal de entrada aplicada. En otras palabras, el relé proporciona aislamiento entre el controlador y el dispositivo, ya que los dispositivos pueden funcionar tanto en CA como en DC. Sin embargo, reciben señales del microcontrolador que funciona en CC, por lo que se requiere un relé para cerrar la brecha. El relé es extremadamente útil cuando necesita controlar una gran cantidad de corriente o voltaje con poca señal eléctrica (Sunfounder, 2013).



Figura 160: Modulo Relé práctica 4. (Sunfounder, 2013).

Principio experimental

Relé: hay 5 partes en cada relé:

1. Electroimán: Consiste en un núcleo de hierro herido por una bobina de cables. Cuando pasa la electricidad, se vuelve magnética. Por lo tanto, se llama electroimán.

2. Armadura: la banda magnética móvil se conoce como armadura. Cuando la corriente fluye a través de ellos, la bobina está energizada, produciendo así un campo magnético que se utiliza para crear o romper los puntos normalmente abiertos (N / O) o normalmente cercanos (N / C). Y la armadura se puede mover con corriente continua (CC) así como con corriente alterna (AC).
3. Resorte: cuando no circulan corrientes a través de la bobina en el electroimán, el resorte aleja la armadura para que el circuito no pueda completarse.
4. Conjunto de contactos eléctricos: hay dos puntos de contacto:
 - Normalmente abierto: conectado cuando el relé está activado y desconectado cuando está inactivo.
 - Normalmente cerca: no está conectado cuando el relé está activado y se conecta cuando está inactivo.
5. Marco moldeado - Los relés están cubiertos con plástico para protección.

Trabajo de retransmisión

El principio de funcionamiento del relé es simple. Cuando se suministra energía al relé, las corrientes comienzan a fluir a través de la bobina de control; Como resultado, el electroimán comienza a energizarse. Luego, la armadura se atrae hacia la bobina, bajando el contacto móvil y conectándose así con los contactos normalmente abiertos. Entonces, el circuito con la carga está energizado. Luego, romper el circuito sería un caso similar, ya que el contacto móvil se elevará a los contactos normalmente cerrados bajo la fuerza del resorte. De esta manera, la conexión y desconexión del relé puede controlar el estado de un circuito de carga (Sunfounder, 2013).

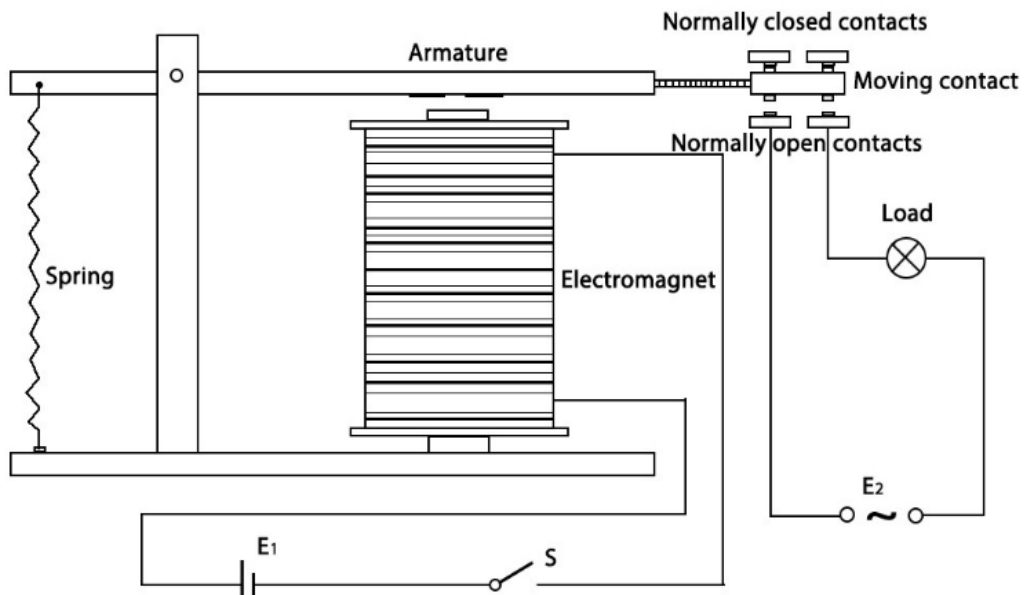


Figura 161: Esquema de un relé práctica 4. (Sunfounder, 2013).

Foto-resistencia

Introducción

El sensor es de hecho un foto-resistor que cambia su resistencia con intensidad de luz variable. Se puede usar para hacer un foto-interruptor (Sunfounder, 2013).



Figura 162: Foto-resistencia práctica 4. (Sunfounder, 2013).

Principio experimental

Una resistencia foto-resistente o dependiente de la luz (LDR) o fotocélula es una resistencia variable controlada por luz. La resistencia de un foto-resistor disminuye al aumentar la intensidad de la luz incidente; En otras palabras, exhibe fotoconductividad. Se puede utilizar la foto-resistencia en los circuitos detectores sensibles a la luz y en los circuitos de conmutación activados por luz y oscuridad (Sunfounder, 2013).

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar la **Foto-Resistencia** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Conectar el **Emisor láser** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
4. Conectar el **Relé** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
5. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
6. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.
7. Para utilizar los bloques de las funciones de **LINX**, seguir los siguientes pasos:
 - Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
 - Seleccionar el submenú **MakerHub**.
 - Seleccionar el submenú **LINX**.
8. Para establecer la comunicación, colocar el bloque **OPEN** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 163: Bloque de apertura de comunicación práctica 4.

9. Para visualizar los datos obtenidos por la Foto-Resistencia, utilizar el subVI llamado **Photocell** de LINX para este elemento (*Functions > Makerhub >*

Linx > *Sensors* > *Generic*) y asigne el puerto analógico 0 de la tarjeta Arduino.



Figura 164: SubVI Photocell de Linx práctica 4.

10. Establecer las condiciones para la activación del Relé de acuerdo a lo indicado por el docente.

11. Para activar el **Relé**, utilizar un bloque de escritura digital (**Digital Write**) y seleccione el puerto digital número 2, recordar que el Relé funciona con lógica negativa por lo tanto para activarlo debe enviar una señal baja (**0**) a SIG la bobina del **Relé** estará electrificada.

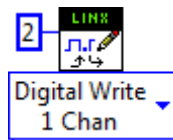


Figura 165: Bloque de Escritura Digital activación del relé práctica 4.

12. Conectar el puerto COM de los contactos del **Relé** a **VCC**, el puerto **NO (Normalmente Abierto)** al pin digital 5 de la tarjeta **Arduino MEGA 2560**, en el diagrama de bloques utilice el bloque de lectura digital (**Digital Read**) y seleccione el puerto digital número 5.

13. Para activar el **Emisor láser**, conectar el pin **VCC** del mismo al puerto **NO (Normalmente Abierto)** al pin **SIG a GND**.

14. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 166: Bloque de cierre de comunicación práctica 4.

15. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

16. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 167: Bloque de apertura de conexión de Database práctica 4.

17. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 168: Bloque para inserta datos en Database práctica 4.

18. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 169: Bloque de cierre de conexión de Database práctica 4.

CONFIGURACIÓN DE DATABASE

19. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

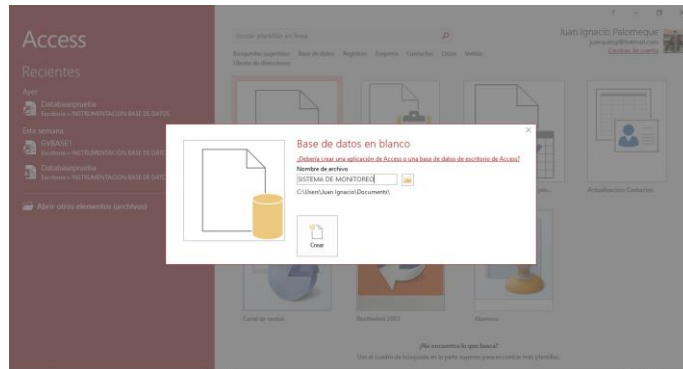


Figura 170: Creación de base de datos en MICROSOFT ACCESS práctica 4.

20. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESSS 2002-2003**.

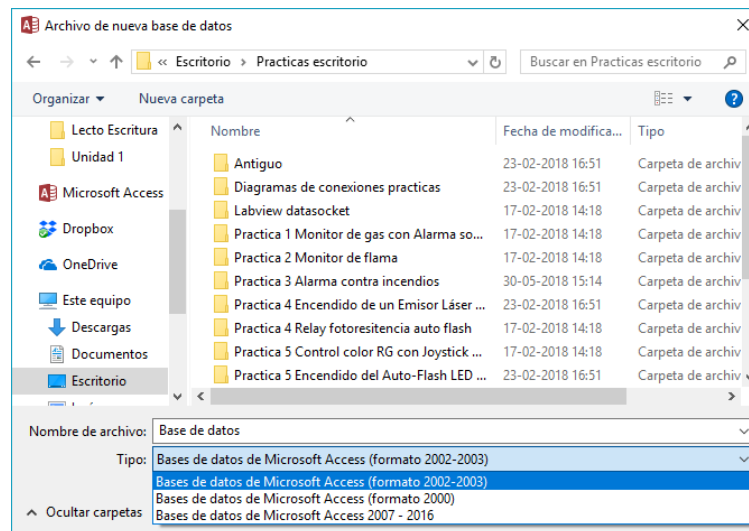


Figura 171: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 4.

21. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

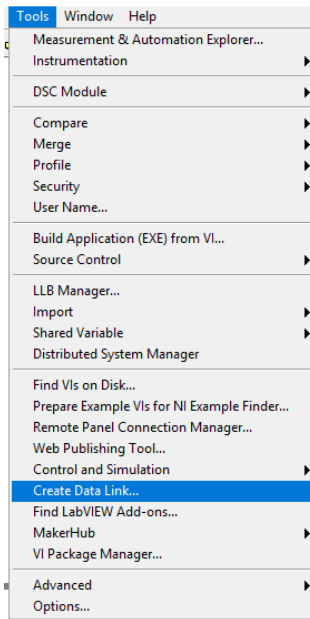


Figura 172: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 4.

- 22.** Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

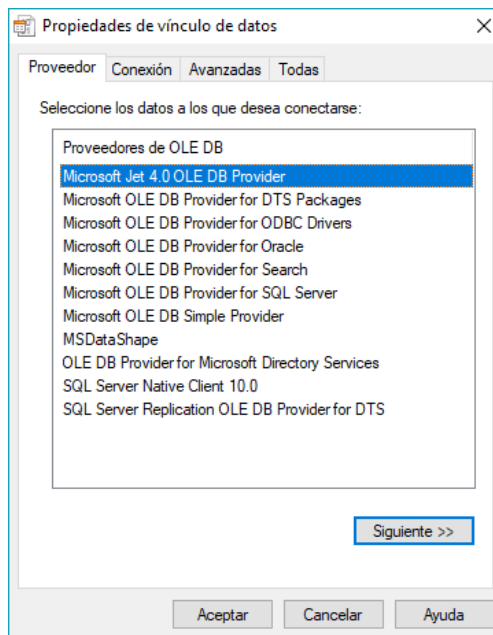


Figura 173: Ventana de propiedades de vínculo de datos práctica 4.

- 23.** Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

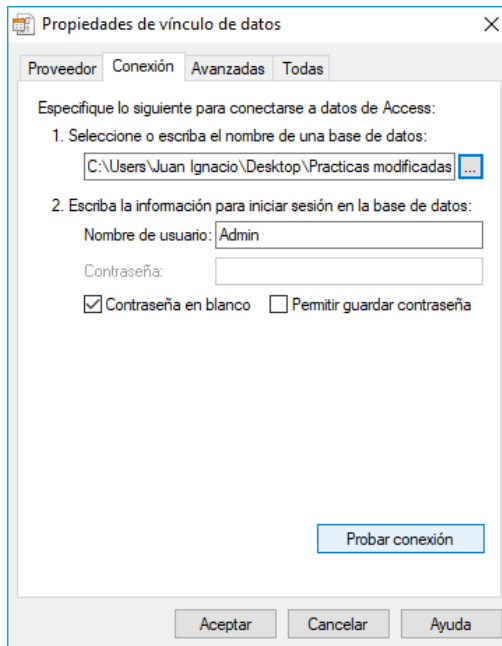


Figura 174: Selección de ruta donde se ubica la base de datos práctica 4.

24. Una vez enlazada la ruta se procede a probar conexión.

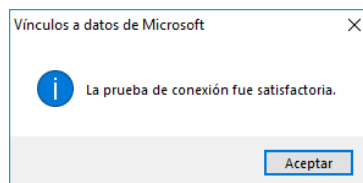


Figura 175: Venta de prueba de conexión de la vinculación de datos práctica 4.

25. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

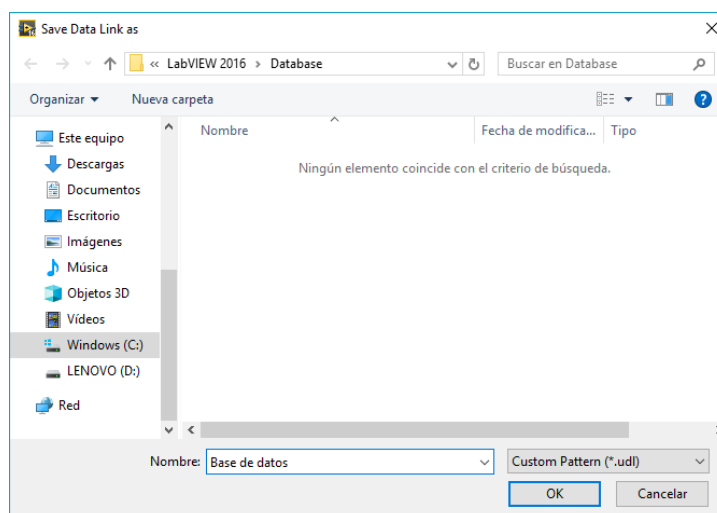


Figura 176: Ventana de asignación de nombre para la tabla de registro de datos práctica 4.

26. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

connection information



Figura 177: Control de ruta de base de datos práctica 4.

27. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:

- Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard**.
- En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next**.
- Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next**.
- Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next**.
- Verificar que los LEDs **TX** y **RX** deben permanezcan encendidos durante unos segundos mientras se implementa el firmware.
- Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.

28. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

29. Dar clic en el botón **Run Continuously**.

30. Obstruir el paso de luz hacia la foto-resistencia para variar el valor obtenido.

31. Observar el comportamiento a través de la pantalla del módulo analógica de la **Foto-Resistencia**.

32. Comprobar la activación del **Relé** ante la ausencia de luz sobre la foto-resistencia.

33. Comprobar la activación del indicador lumínico a través de la activación del **Emisor láser**.

DIAGRAMA DE CONEXIONES

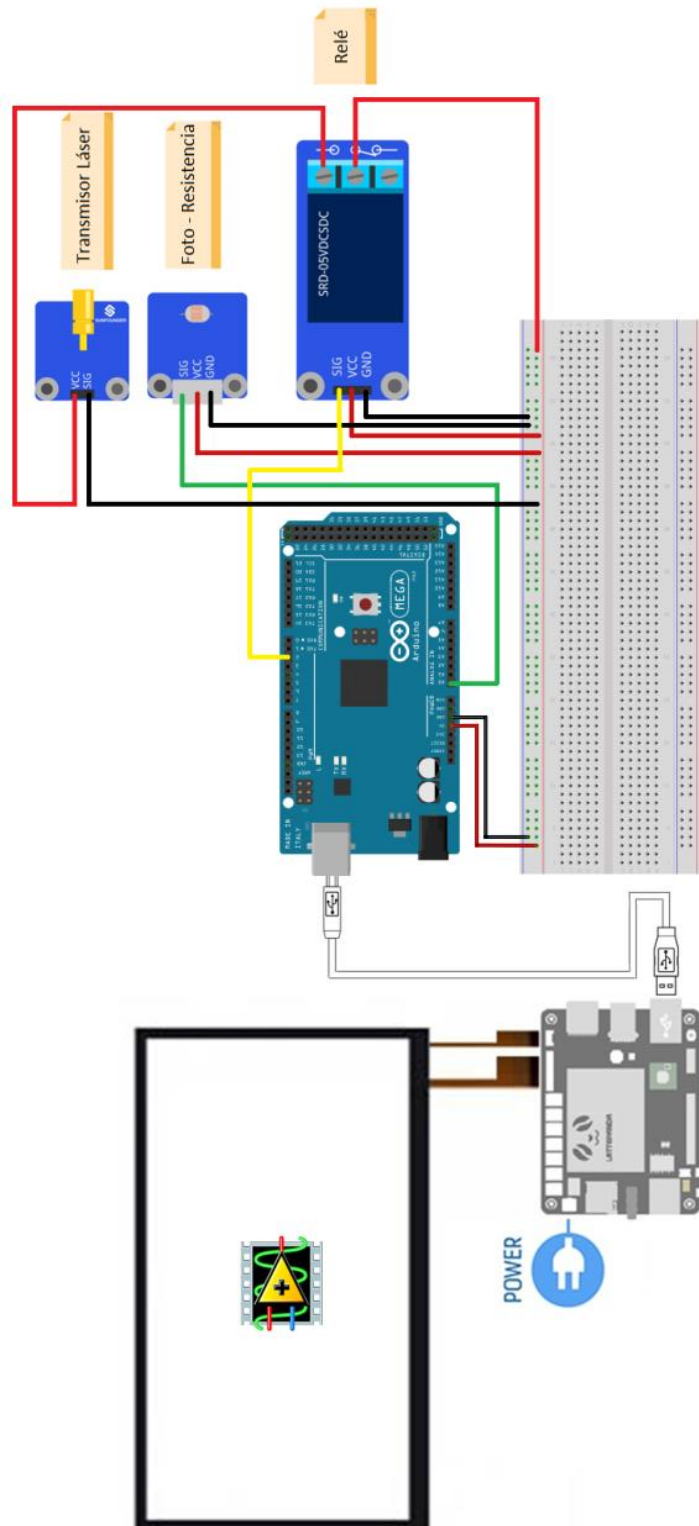


Figura 178: Diagrama de conexiones de la Práctica 4.

Tabla 5: Conexiones de la Práctica 4

| | |
|-------------------------|--------------------------|
| FOTO-RESISTENCIA | ARDUINO MEGA 2560 |
| SIG | A0 |
| VCC | 5V |
| GND | GND |
| RELÉ | ARDUINO MEGA 2560 |
| SIG | 2 |
| VCC | 5V |
| GND | GND |
| COM | 5V |
| NO | VCC EMISOR LÁSER |
| EMISOR LÁSER | ARDUINO MEGA 2560 |
| VCC | NO RELE |
| GND | GND |

DISEÑOS EN LABVIEW



Figura 179: Panel Frontal de la Práctica 4.

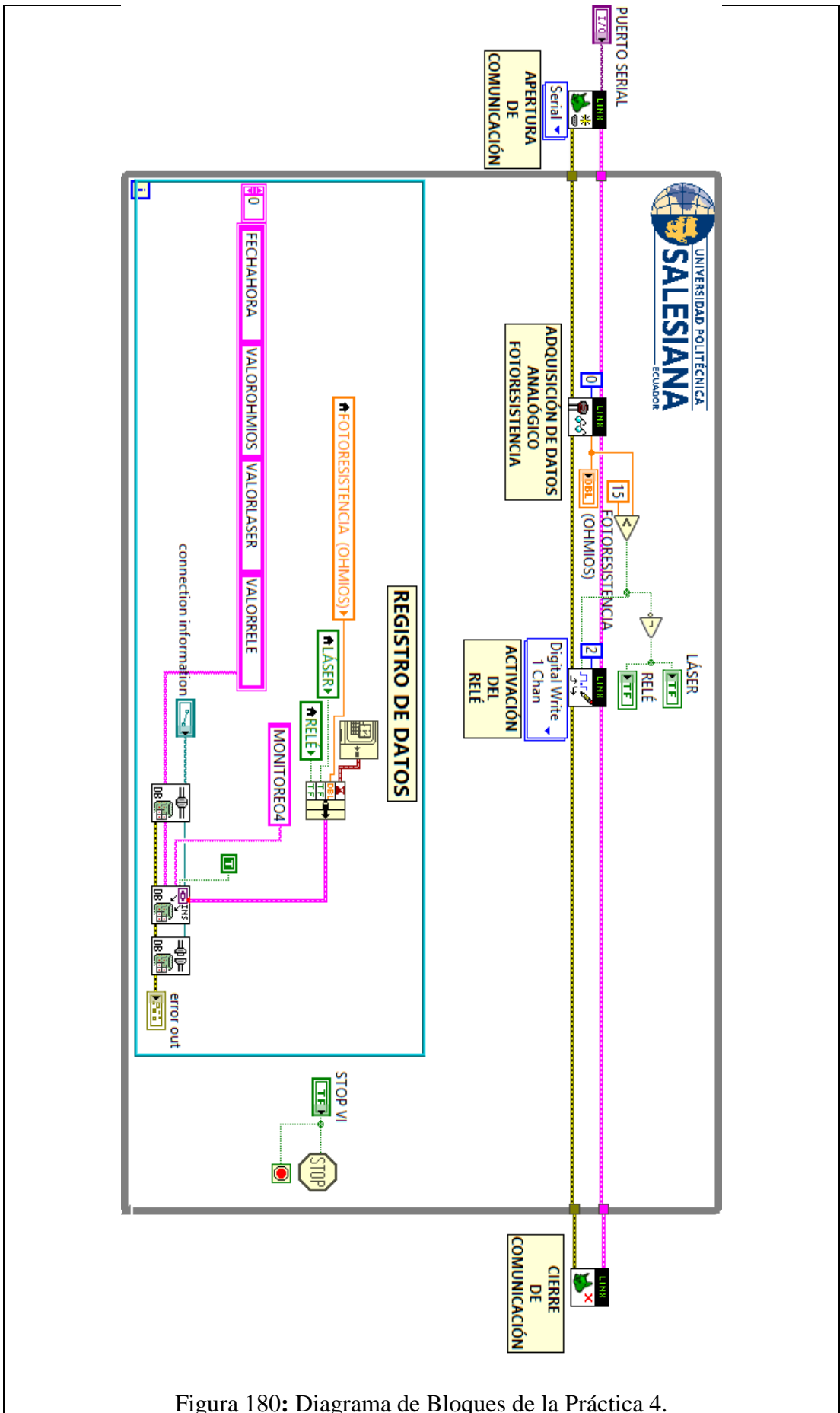


Figura 180: Diagrama de Bloques de la Práctica 4.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.
3. Mediante la variación de luz en el ambiente se logró la activación del módulo relé, esta acción está condicionada en función del valor entregado por la foto-resistencia.
4. Se comprobó el correcto funcionamiento del emisor láser, activando el mismo mediante la conmutación del relé.


ANALISIS DE RESULTADO:

Al acercar una linterna a la Foto-resistencia el valor en ohmios de la misma decrece, cuando dicho valor es menor a 15, se produce la activación del module relé.

Cuando la bobina del Relé se energiza cierra su contacto NO, permitiendo el paso de corriente hacia el emisor laser, logrando así la activación del mismo.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

4.5. Práctica # 5

| | | | |
|--|---|--|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO | |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores | |
| NRO. PRÁCTICA: | 5 | TÍTULO PRÁCTICA: Encendido del Auto-Flash LED y Dual Color LED. | |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual accione un Auto-Flash LED y un Dual Color LED por medio del control de un Joystick. | | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar el Joystick en el software LabVIEW 2016 para que active el funcionamiento del Dual Color LED y del Auto-Flash LED según las condiciones establecidas. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Elaborar una base de datos donde se registren los eventos y datos obtenidos por el sensor. | | | |
| INSTRUCCIONES: | | 1. Alimentar el módulo principal con 110v. | |
| | | 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. | |
| | | 3. Seguir la guía paso a paso para realizar la conexión del módulo. | |
| | | 4. Realizar una base de datos en ACCESS para registrar los eventos y datos obtenidos por el sensor. | |
| MARCO TEORICO | | | |
| DUAL COLOR LED | | | |
| Introducción Un diodo emisor de luz (LED) de dos colores es capaz de emitir dos colores diferentes de luz, típicamente rojo y verde. Un Dual Color LED presenta dos pines, dispuestos en el circuito en antiparalelo y conectado por un cátodo / ánodo. Voltaje positivo puede ser dirigido hacia uno de los terminales LED, haciendo que ese terminal emita la luz del color correspondiente; cuando la dirección de la tensión se invierte, la luz de la otra el color es emitido. En un LED de dos colores, solo uno de los pines puede recibir voltaje a la vez. Como resultado, este tipo de LED funciona | | | |

con frecuencia como luces indicadoras para una variedad de dispositivos, incluyendo televisores, cámaras digitales y controles remotos (Sunfounder, 2013).



Figura 181: Dual Color LED práctica 5. (Sunfounder, 2013).

Principio experimental

Controle el brillo del LED por el puerto digital. El color del LED cambia de rojo a verde, así como destellos de un color mezclado (Sunfounder, 2013).

Joystick

Introducción

Un joystick es un dispositivo de entrada que consiste en una palanca que gira sobre una base e informa su ángulo o dirección al dispositivo que está controlando. Joysticks se utilizan a menudo para controlar videojuegos y robots (Sunfounder, 2013).

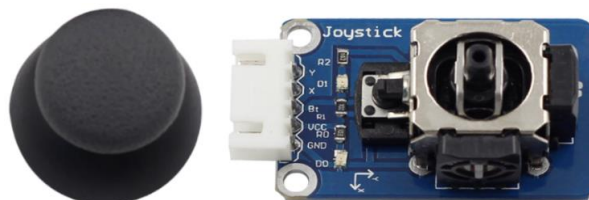


Figura 182: Joystick práctica 5. (Sunfounder, 2013).

Principio experimental

Este módulo tiene dos salidas analógicas (correspondientes a coordenadas X e Y) y una salida digital que representa si se presiona en el eje Z (Sunfounder, 2013).

LED de flash automático

Introducción

El módulo LED de 7 colores con flash automático puede flashear automáticamente los colores incorporados después de encenderlo. Se puede usar para hacer efectos de luces bastante fascinantes (Sunfounder, 2013).



Figura 183: LED de flash automático práctica 5. (Sunfounder, 2013).

Principio experimental

Cuando se enciende, el LED de flash automático de 7 colores destella los colores incorporados (Sunfounder, 2013).

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Joystick** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Conectar el **Dual Color LED** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
4. Conectar el **Auto-Flash LED** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
5. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
6. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.
7. Para utilizar los bloques de las funciones de **LINX**, seguir los siguientes pasos:
 - Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
 - Seleccionar el submenú **MakerHub**.
 - Seleccionar el submenú **LINX**.
8. Para establecer la comunicación, colocar el bloque **OPEN** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 184: Bloque de apertura de comunicación práctica 5.

9. Para visualizar los datos analógicos generados por el Joystick, utilizar los siguientes bloques:

EJE X: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 0.

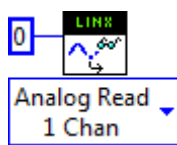


Figura 185: Bloque de Lectura Analógica Eje X Joystick práctica 5.

EJE Y: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 1.

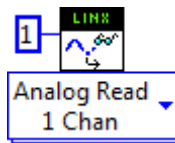


Figura 186: Bloque de Lectura Analógica Eje Y Joystick práctica 5.

10. Para visualizar los estados digitales generados por el Joystick al presionar el centro del mismo, utilizar el bloque de lectura digital (**Digital Read**):

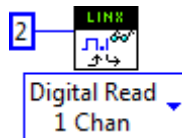


Figura 187: Bloque de Lectura Digital Botón Joystick práctica 5.

11. Para activar el **Dual Color LED** en color Rojo, utilizar un bloque de escritura digital (**Digital Write**) seleccione el puerto digital 4.

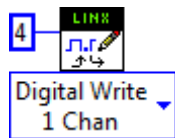


Figura 188: Bloque de Escritura Digital color rojo Dual Color LED práctica 5.

12. Para activar el **Dual Color LED** en color Verde, utilizar un bloque de escritura digital (**Digital Read**) seleccione el puerto digital 6.



Figura 189: Bloque de Escritura Digital color verde Dual Color LED práctica 5.

13. Para activar el **Auto-Flash LED**, conectar el pin **SIG** del mismo al puerto digital 5 utilizando un bloque de escritura digital (**Digital Write**). Recordar que este funciona con lógica negativa por lo tanto para activarlo debe enviar una señal baja (**0**) a SIG para activarlo.

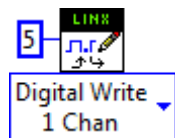


Figura 190: Bloque de Escritura Digital. Auto-Flash LED práctica 5.

14. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 191: Bloque de cierre de comunicación práctica 5.

15. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

16. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 192: Bloque de apertura de conexión de Database práctica 5.

17. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 193: Bloque para inserta datos en Database práctica 5.

18. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 194: Bloque de cierre de conexión de Database práctica 5.

CONFIGURACIÓN DE DATABASE

19. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

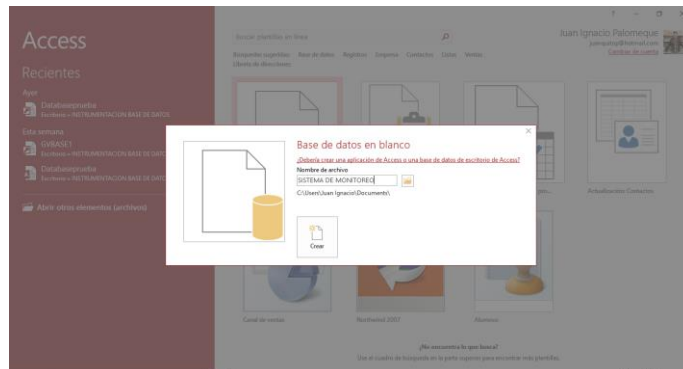


Figura 195: Creación de base de datos en MICROSOFT ACCESS práctica 5.

20. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESSS 2002-2003**.

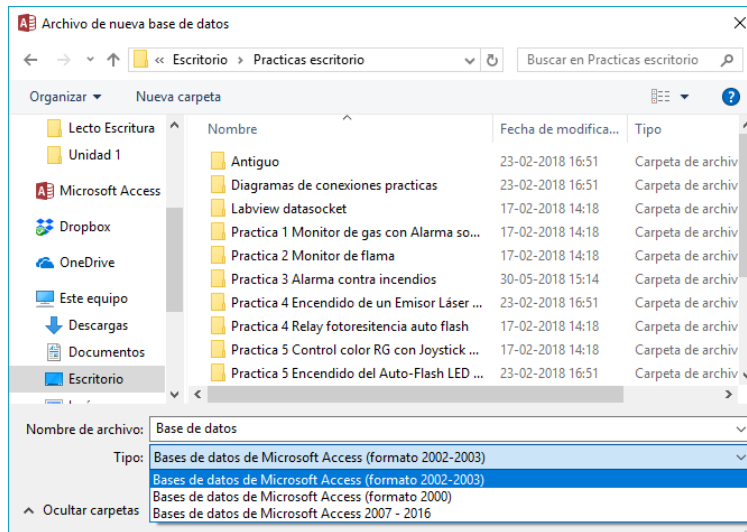


Figura 196: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 5.

21. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

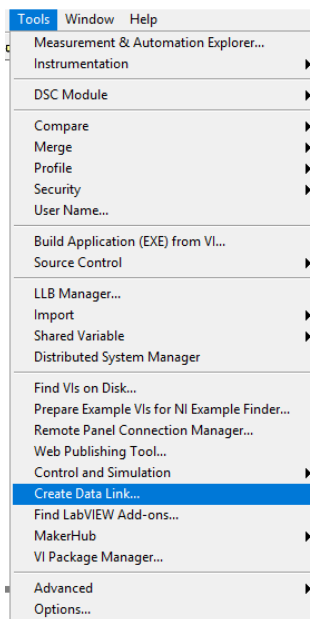


Figura 197: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 5.

22. Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

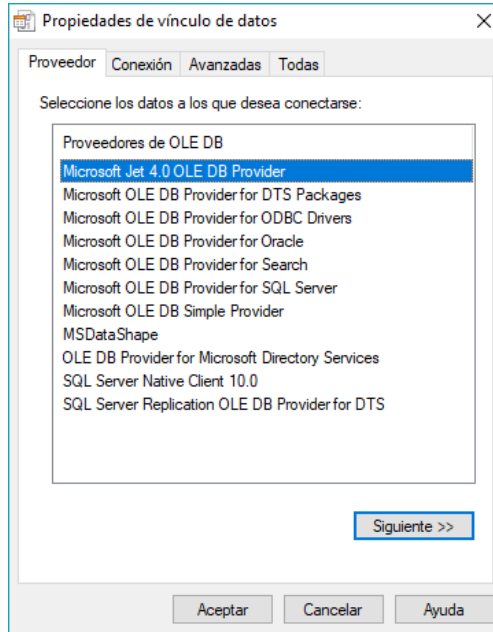


Figura 198: Ventana de propiedades de vínculo de datos práctica 5.

23. Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

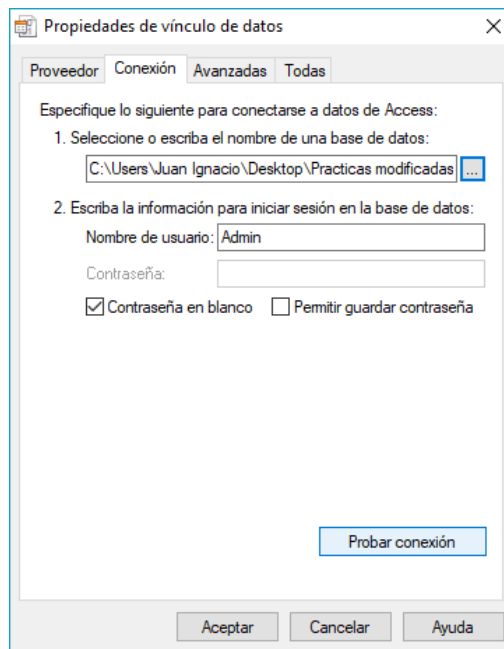


Figura 199: Selección de ruta donde se ubica la base de datos práctica 5.

24. Una vez enlazada la ruta se procede a probar conexión.

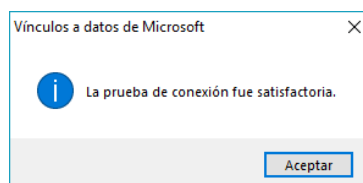


Figura 200: Venta de prueba de conexión de la vinculación de datos práctica 5.

25. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

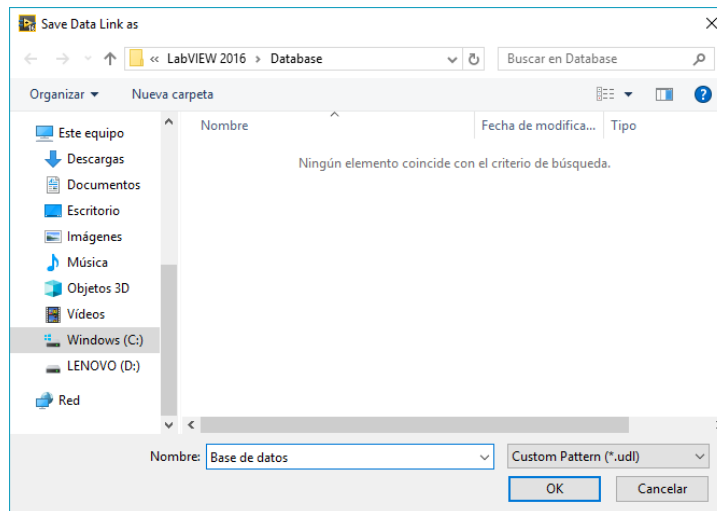


Figura 201: Ventana de asignación de nombre para la tabla de registro de datos práctica 5.

26. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

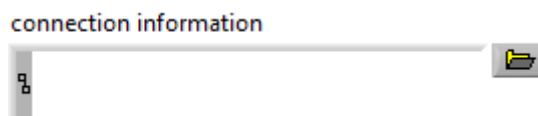


Figura 202: Control de ruta de base de datos práctica 5.

27. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:

- Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard.**
- En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next.**
- Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next.**
- Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next.**
- Verificar que los LEDs **TX** y **RX** deben permanezcan encendidos durante unos segundos mientras se implementa el firmware.
- Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.

28. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

29. Dar clic en el botón **Run Continuously.**

30. Realizar suavemente el movimiento desde el centro hacia la derecha con el Joystick hasta lograr que el Dual Color LED muestre un color verde intenso, esto debido a que se colocó como condición, cuando el valor analógico del eje X sea mayor a 3 se envíe una señal al pin G del módulo.

31. Realizar suavemente el movimiento desde el centro hacia arriba con el Joystick hasta lograr que el **Dual Color LED** muestre un color rojo intenso, esto debido a que se colocó como condición, cuando el valor analógico del eje Y sea mayor a 3 se envíe una señal al pin R del módulo.

32. Presionar en el centro del joystick y observe como se enciende el **Auto-Flash LED**.

33. Observar el comportamiento a través de la pantalla del módulo analógico del **Joystick**.

34. Comprobar la activación del **Dual Color LED** mediante el cumplimiento de la condición establecida en relación al botón del **Joystick**.

DIAGRAMA DE CONEXIONES

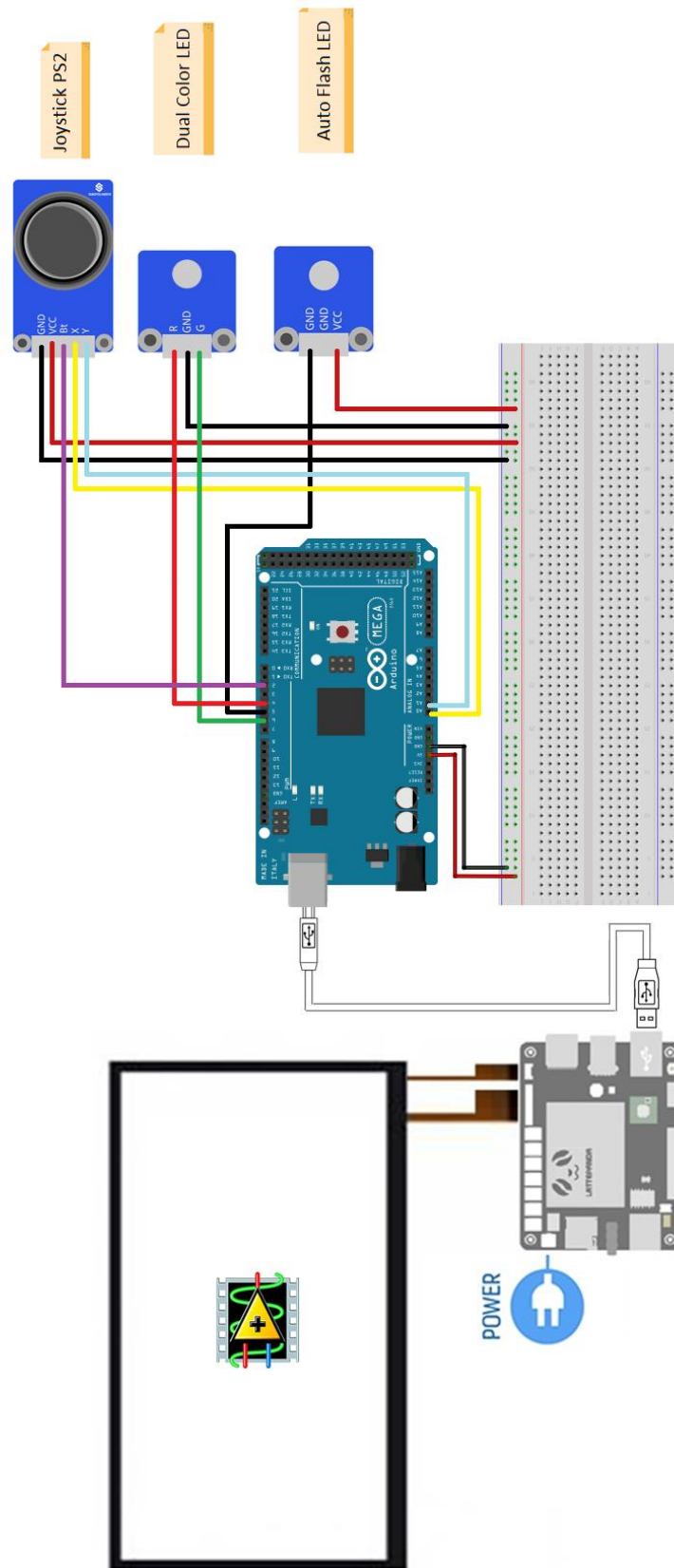


Figura 203: Diagrama de conexiones de la Práctica 5.

Tabla 6: Conexiones de la Práctica 5

| AUTO-FLASH LED | ARDUINO MEGA 2560 |
|----------------|-------------------|
| SIG | 5 |
| VCC | 5V |
| JOYSTICK | ARDUINO MEGA 2560 |
| VCC | 5V |
| GND | GND |
| BT | 2 |
| X | A0 |
| Y | A1 |
| DUAL COLOR LED | ARDUINO MEGA 2560 |
| VCC | 5V |
| R | 4 |
| G | 6 |

DISEÑOS EN LABVIEW



Figura 204: Panel Frontal de la Práctica 5.

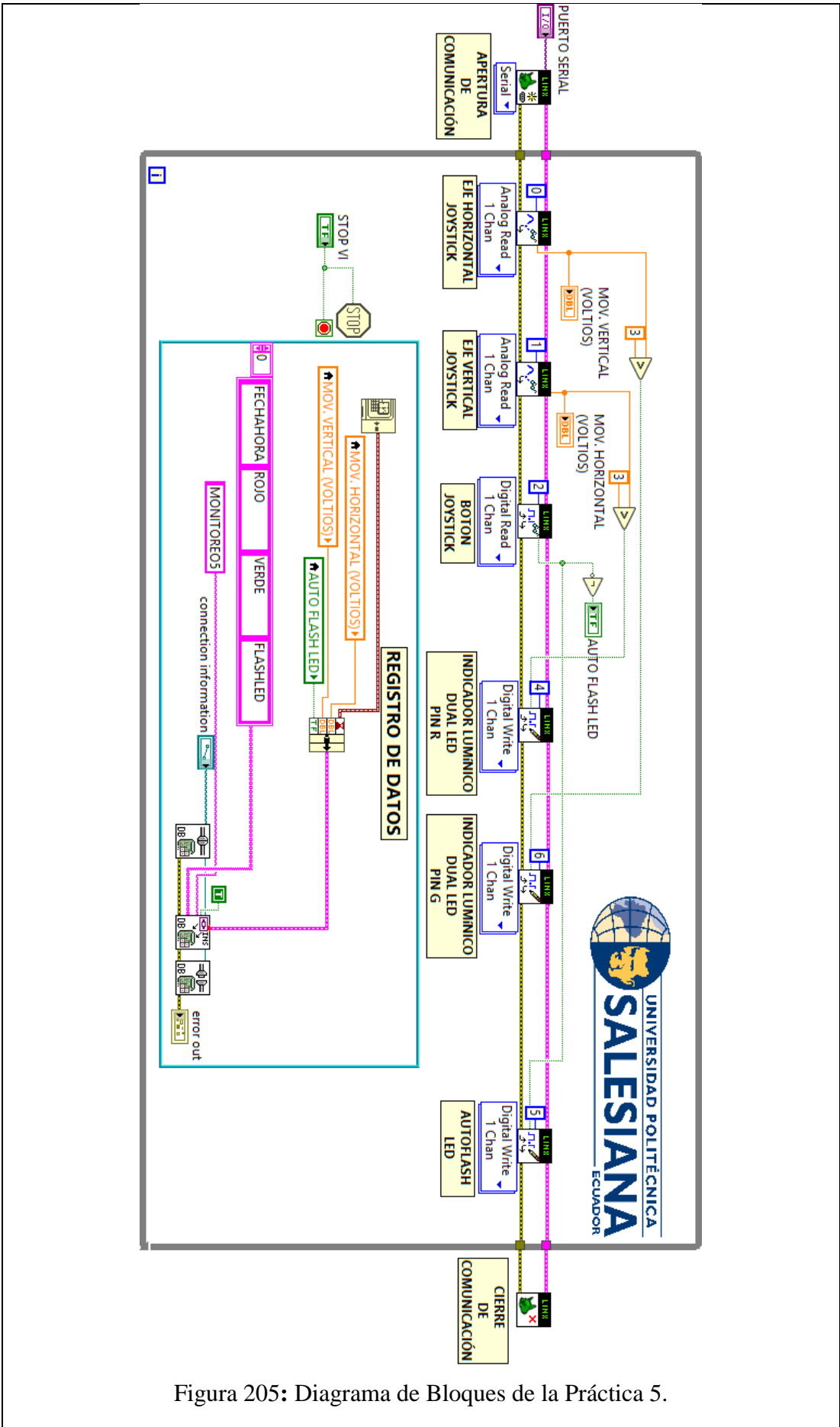


Figura 205: Diagrama de Bloques de la Práctica 5.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.
3. Mediante la variación de voltaje producida por el movimiento del Joystick se establecieron condiciones para la activación del Dual Color LED, el mismo que permite mostrar los colores ROJO y VERDE.
4. Mediante la variación de estado de tipo booleano generado al presionar el Joystick se logró la activación del Auto Flash LED.
5. Los eventos generados por los sensores fueron registrados en un base de datos.

ANALISIS DE RESULTADO:

El movimiento desde el centro hacia la derecha del Joystick genera una variación de voltaje, cuando el valor analógico es mayor a 3 se envía una señal al pin G del módulo, logrando de esta manera el encendido en color verde del Dual Color LED.


El movimiento desde el centro hacia arriba del Joystick genera una variación de voltaje, cuando el valor analógico es mayor a 3 se envía una señal al pin R del módulo, logrando de esta manera el encendido en color rojo del Dual Color LED.

Cuando se presiona en el centro del joystick se envía un pulso, el mismo que enciende el Auto-Flash LED.

Todos los eventos generados por cada acción realizada se graban en una base de datos.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

4.6. Práctica # 6

| | | |
|--|----------|---|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 6 | TÍTULO PRÁCTICA: Barómetro BMP180. |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual permita leer los datos de presión, temperatura del ambiente y altura a nivel del mar entregado por el Barómetro BMP180. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar el Barómetro BMP180 a través del software ARDUINO y por medio del monitor serial proyectar en LabVIEW 2016 los datos de presión, temperatura del ambiente y altura a nivel del mar. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. | | |
| INSTRUCCIONES: | | <ol style="list-style-type: none"> 1. Alimentar el módulo principal con 110v. 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. 3. Seguir la guía paso a paso para realizar la conexión del módulo. 4. Realizar una base de datos a través del monitor serial de Arduino para registrar los eventos y datos obtenidos por el sensor. |
| MARCO TEORICO | | |
| Barómetro BMP180 Introducción Un barómetro es un instrumento científico utilizado para medir la presión del aire. En este experimento, la placa Arduino Mega 2560 se utiliza como la placa maestra para leer el valor de BMP180 y mostrarlo en el Monitor en serie (Sunfounder, 2013). | | |

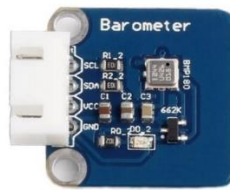


Figura 206: Barómetro bmp180 práctica 6. (Sunfounder, 2013).

Principio experimental

Use un barómetro para medir la presión y temperatura del aire.

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Barómetro BMP180** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
4. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.
5. Utilizar comunicación I2C para realizar la lectura del barómetro.
6. Para utilizar los bloques de las funciones de **VISA**, seguir los siguientes pasos:
 - Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
 - Seleccionar el submenú **Instrument I/O**.
 - Seleccionar el submenú **SERIAL**.
7. Para establecer la comunicación, colocar el bloque **Configure Port** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 207: Bloque Configuración de puerto serial práctica 6.

8. Para acondicionar los datos obtenidos por el Barómetro, utilizar la herramienta **VISA Node Property (Functions > Instrument I/O > VISA > VISA Advanced)**:

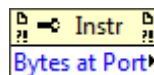


Figura 208: Bloque Nodo de propiedad VISA práctica 6.

9. Para leer los datos obtenidos por el Barómetro, utilizar la herramienta **Read (Functions > Instrument I/O > VISA)**:



Figura 209: Bloque de Lectura VISA práctica 6.

10. El bloque de lectura entrega un dato tipo String, para filtrar este dato, utilizar la herramienta **Fract/Exp String To Number Function (Functions > Addons > Arduino Compatible Compiler for LabVIEW Standard > String > Number String Conversion)**:



Figura 210: Bloque de Fract/Exp String To Number Function práctica 6.

11. Para cerrar la comunicación, colocar el bloque CLOSE al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 211: Bloque Close de VISA práctica 6.

12. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

13. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 212: Bloque de apertura de conexión de Database práctica 6.

14. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 213: Bloque para inserta datos en Database práctica 6.

15. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 214: Bloque de cierre de conexión de Database práctica 6.

CONFIGURACIÓN DE DATABASE

16. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

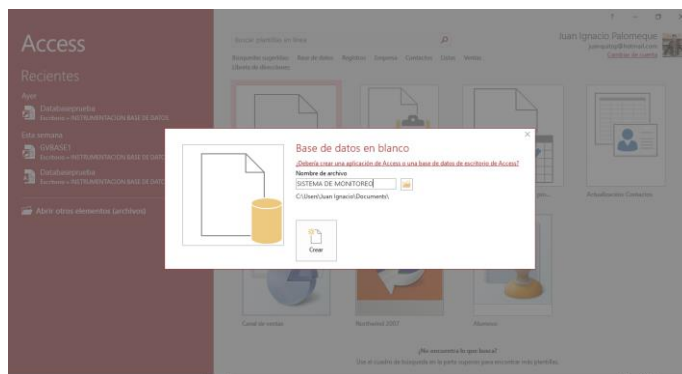


Figura 215: Creación de base de datos en MICROSOFT ACCESS práctica 6.

17. Crear un nombre para la base de datos; guardar el tipo de archivo como MICROSOFT ACCESS 2002-2003.

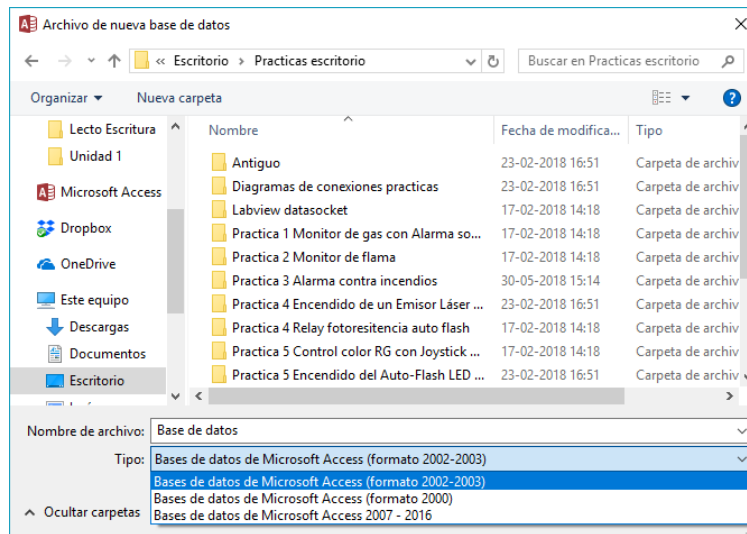


Figura 216: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 6.

18. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

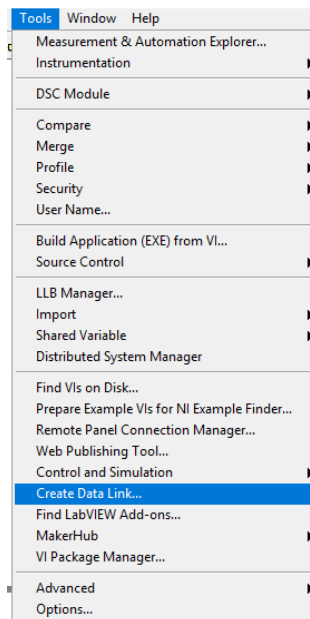


Figura 217: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 6.

19. Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

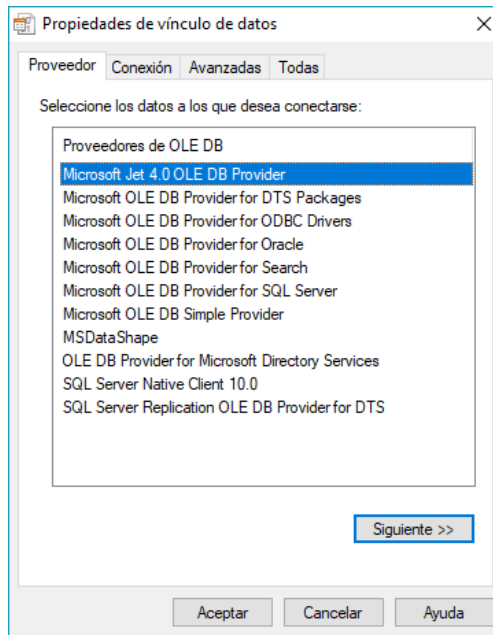


Figura 218: Ventana de propiedades de vínculo de datos práctica 6.

20. Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

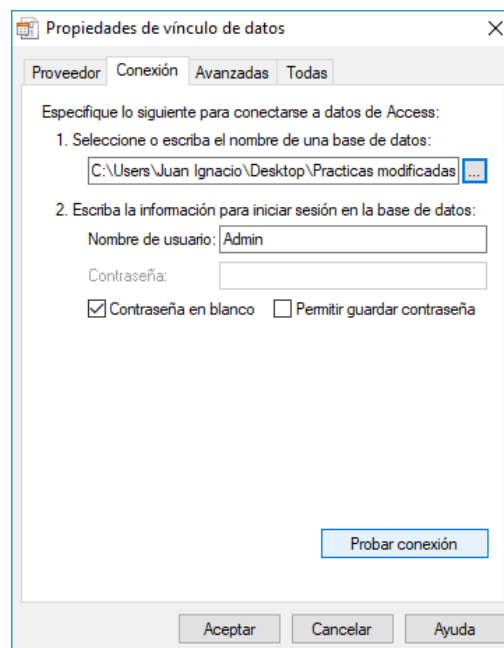


Figura 219: Selección de ruta donde se ubica la base de datos práctica 6.

21. Una vez enlazada la ruta se procede a probar conexión.

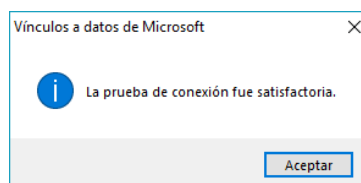


Figura 220: Venta de prueba de conexión de la vinculación de datos práctica 6.

22. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

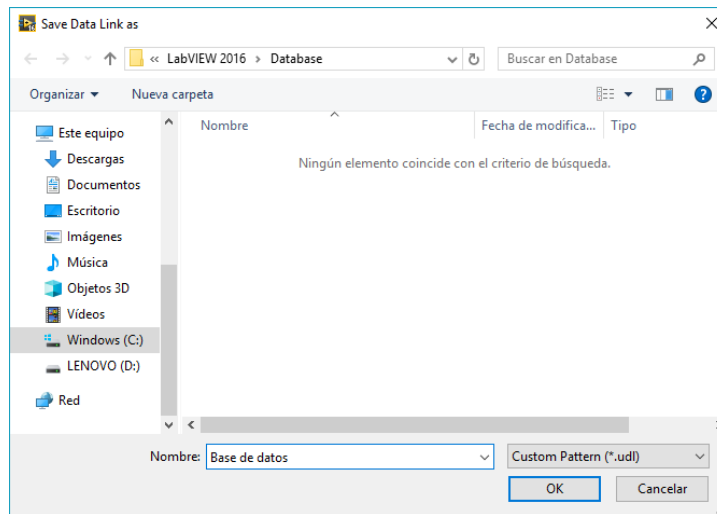


Figura 221: Ventana de asignación de nombre para la tabla de registro de datos práctica 6.

23. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

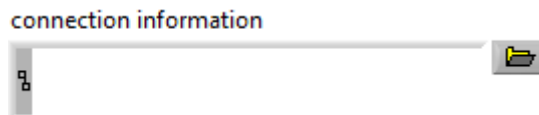


Figura 222: Control de ruta de base de datos práctica 6.

24. Transcribir el código adjunto en el software de Arduino, ejecutar y compilar el archivo.

25. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

26. Dar clic en el botón **Run Continuously**.

27. Observar las variables de temperatura, presión y altura a nivel del mar obtenidas por el Barómetro.

DIAGRAMA DE CONEXIONES

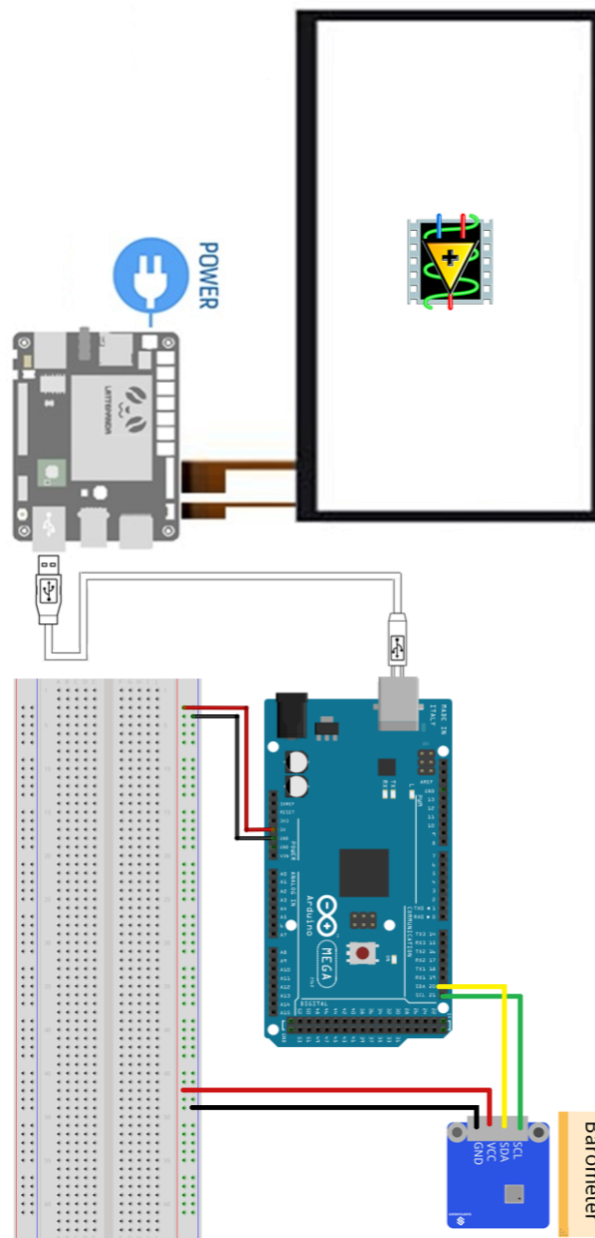


Figura 223: Diagrama de conexiones de la Práctica 6.

Tabla 7: Conexiones de la Práctica 6

| BARÓMETRO BMP280 | ARDUINO MEGA 2560 |
|------------------|-------------------|
| VCC | 5V |
| GND | GND |
| SCL | 21 SCL |
| SDA | 20 SDA |

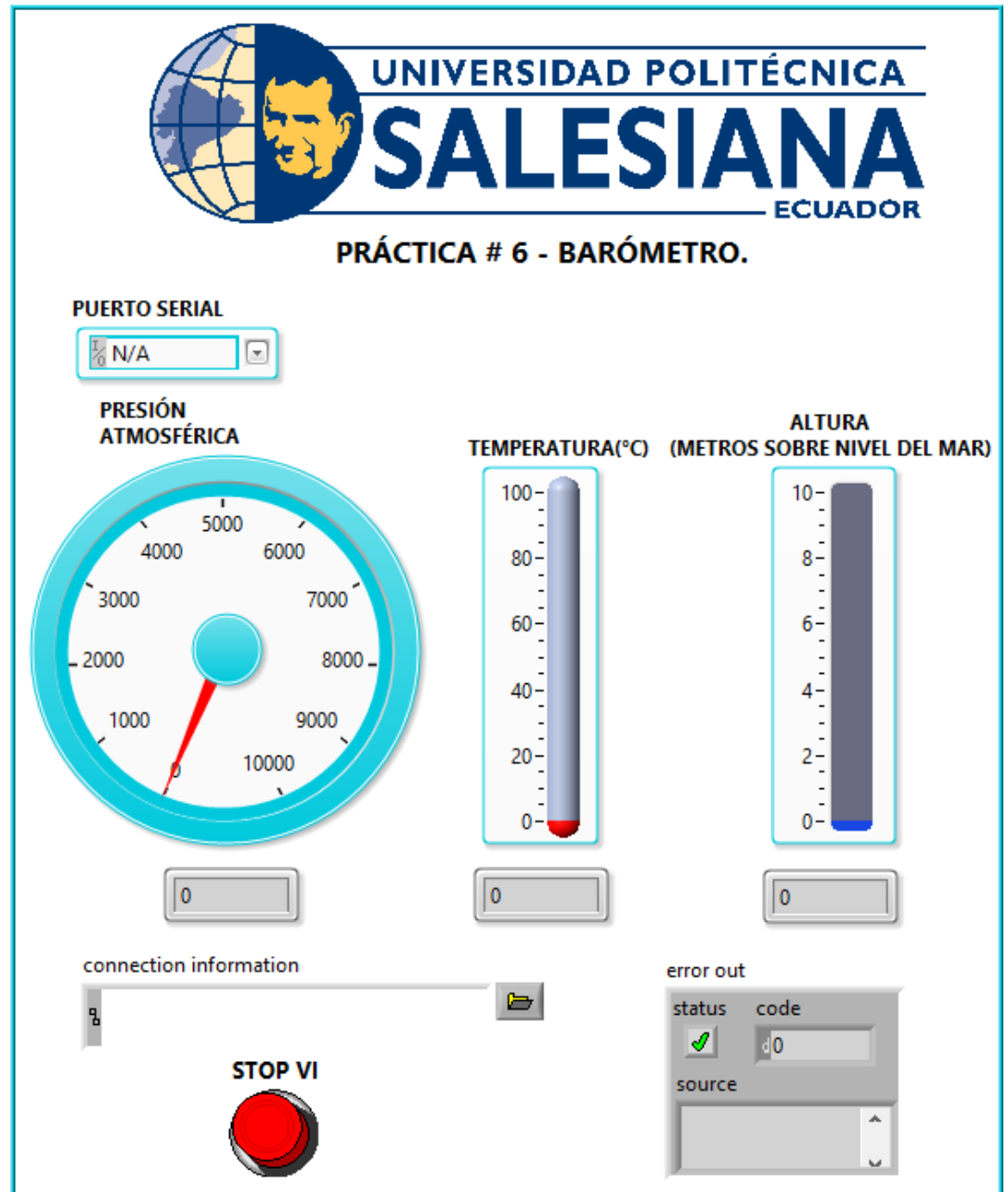


Figura 224: Panel Frontal de la Práctica 6.

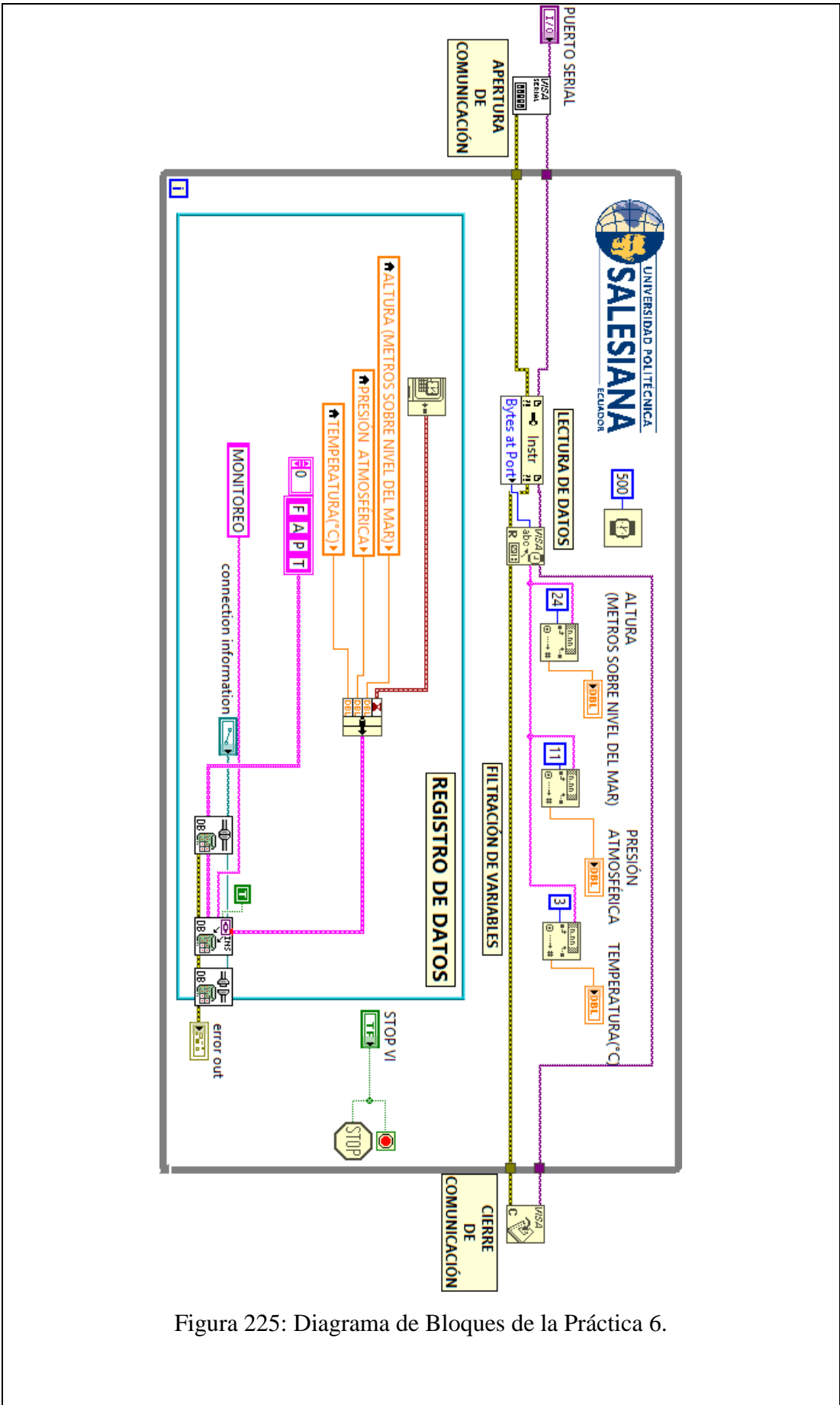


Figura 225: Diagrama de Bloques de la Práctica 6.

PROGRAMACIÓN EN ARDUINO

```
#include <SFE_BMP180.h>
#include <Wire.h>

SFE_BMP180 pressure; //Declaramos la variable para la presion

#define ALTITUDE 4.02 // Definimos nuestra Altitude local

void setup()
{
  Serial.begin(9600);

  // Inicializa el sensor (se inicializa con la configuracion de altura previa).

  if (pressure.begin())
    Serial.println("BMP180 init success");
  else
  {
    Serial.println("BMP180 init fail\n\n");
    while(1);
  }
}

void loop()
{
  char status;
  double T,P,p0,a;

status = pressure.startTemperature();
if (status != 0)
{
  // espera a que se complete la medicion de Temp.
  delay(status);

  // Retrieve the completed temperature measurement:
  // Note la medicion de temperatura se almacena en la variable T.
  // Function regresa: 1 si la lectura es satisfactoria, 0 si no es posible obtener la lectura de Temp.

  status = pressure.getTemperature(T);
  Serial.print("T: ");
  Serial.print(T);
  Serial.print("\t");
  if (status != 0)
  {
    // Inicia la medicion de presion:
    // El parametro esta estableciendo un sobremuestreado, de 0 to 3 (residui mas alto, mayor tiempo de es
    // Si la respuesta es satisfactoria, Regresa el numero en ms para espera.
    // Si la respuesta no es satisfactoria, Regresa un 0.

    status = pressure.startPressure(3);
    if (status != 0)
    {
      // Espera a que la medicion sea completada:
      delay(status);
```

```

// Retoma la medicion de presion completada:
// Note que la medicion es almacenada en la variable P.
// Note tambien que la funcion requiere la medicion de temperatura previa(T).

status = pressure.getPressure(P,T);
if (status != 0)
{
  // Print out the measurement:
  Serial.print("P: ");
  Serial.print(P,2);
  Serial.print("mb");
  Serial.print("\t");

  p0 = pressure.sealevel(P,ALTITUDE); // presion relativa a nivel del mar

  // Para obtener la altitud dependiendo de la presion obtenida.
  // usar la funcion de altitud con la presion de referencia(Presion a nivel de mar)
  // Parameters: P = presion absoluta en mb, p0 = presion de referencia en mb.
  // Result: a = altitud en m.

  a = pressure.altitude(P,p0);
  Serial.print("A ");
  Serial.println(a,0);

  }
  else Serial.println("error retrieving pressure measurement\n");
}
else Serial.println("error starting pressure measurement\n");
}
else Serial.println("error retrieving temperature measurement\n");
}
else Serial.println("error starting temperature measurement\n");

delay(2000); // Pause for 2 seconds.
}
|

```

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. La comunicación entre los sensores y el sistema embebido Arduino MEGA 2560 fue posible gracias a la herramienta VISA desarrollada por LabVIEW.
3. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.
4. El Barómetro BMP180, consta de una librería propia por lo cual se debe realizar la programación en el software Arduino.
5. Mediante el uso del bus I2C por medio de los puertos SCL y SDA del sistema embebido Arduino MEGA 2560 puede proyectar los datos adquiridos al programa desarrollado en LabVIEW 2016.

ANALISIS DE RESULTADO:

La comunicación en esta práctica se realiza mediante las herramientas de VISA, software LabVIEW realiza una lectura de datos adquiridos en el programa de Arduino.

La programación del barómetro BMP 180 se realiza mediante su librería en el software de Arduino.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

4.7. Práctica # 7

| | | |
|--|---|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 7 | TÍTULO PRÁCTICA: Contador de pulsos utilizando Sensor Hall. |
| Objetivo: <p>Diseñar en el software LabVIEW 2016 un programa, el cual permita regular la velocidad de un motor DC y a su vez se capaz de detectar el número de pulsos a través de un Sensor Hall.</p> | | |
| Objetivos específicos: <p>Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar la velocidad de giro de un motor DC a través de los bloques de PWM en el software LabVIEW 2016, a su vez configurar el Sensor Hall para que sea capaz de detectar el número de pulsos. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560.</p> | | |
| INSTRUCCIONES: | | <ol style="list-style-type: none"> 1. Alimentar el módulo principal con 110v. 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. 3. Seguir la guía paso a paso para realizar la conexión del módulo. |
| MARCO TEORICO | | |
| Sensor Hall Introducción <p>El interruptor Hall consta de regulador de voltaje, elemento Hall, amplificador diferencial, disparador Schmitt y terminal de salida. Emite valores digitales (Sunfounder, 2013).</p> | | |
|  | | |
| <p>Figura 226: Sensor Hall práctica 7. (Sunfounder, 2013).</p> | | |

Principios experimentales

Conecte el interruptor del módulo Hall con a la placa Arduino Mega 2560. Cuando un controlador energizado se acerca al módulo (Pieza imantada), el terminal de salida SIG genera un nivel bajo.

Potenciómetro

Introducción

Un potenciómetro es una resistencia variable conformado por 2 resistencias en serie, las cuales permiten modificar el valor de la resistencia por el usuario (García, 2016).

Existen varios tipos de potenciómetros, alterando su forma y el método cómo modifican los valores de las resistencias (García, 2016).



Figura 227: Potenciómetro práctica 7. (Sunfounder, 2013).

Principio experimental

Un potenciómetro analógico es un componente electrónico analógico. ¿Cuál es la diferencia entre uno analógico y uno digital? En pocas palabras, un potenciómetro digital se refiere a solo dos estados como encendido / apagado, niveles alto / bajo, es decir, 0 o 1, mientras que uno digital admite señales analógicas como un número del 1 al 1000. El valor de la señal cambia con el tiempo en lugar de manteniendo un número exacto (Sunfounder, 2013). Las señales analógicas incluyen la intensidad de la luz, la humedad, la temperatura, etc (Sunfounder, 2013).

Motor DC

Introducción

El motor de corriente continua DC es una máquina que transforma la energía eléctrica en energía mecánica, generando un movimiento de rotación, gracias a la acción de un campo magnético (Gottlieb, 1994).



Figura 228: Motor DC práctica 7. (García, 2013).

Principio Experimental

Cuando la corriente eléctrica transita a través del bobinado del electroimán giratorio, se genera un campo electromagnético que interactúa con el campo magnético del imán permanente (Álvarez, 2015). El conductor genera una fuerza perpendicular al plano formado por el campo magnético y la corriente, de acuerdo con la Fuerza de Lorentz (Álvarez, 2015):

Ecuación 8: Fuerza de Lorentz práctica 7.

$$F = B \cdot L \cdot I \cdot \text{sen}(\phi)$$

- F: Fuerza en newtons
- I: Intensidad que recorre el conductor en amperios
- L: Longitud del conductor en metros
- B: Densidad de campo magnético o densidad de flujo teslas
- Φ : Ángulo que forma I con B

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Potenciómetro** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Conectar el **Sensor Hall** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
4. Conectar el **Motor DC** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
5. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
6. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.
7. Para utilizar los bloques de las funciones de **LINX**, seguir los siguientes pasos:
 - Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
 - Seleccionar el submenú **MakerHub**.
 - Seleccionar el submenú **LINX**.
8. Para establecer la comunicación, colocar el bloque **OPEN** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 229: Bloque de apertura de comunicación práctica 7.

9. Para visualizar los datos obtenidos por el Potenciómetro, utilizar los siguientes bloques:

DATO ANALOGICO: Utilizar un bloque de lectura analógica (**Analog Read**) y seleccione el puerto analógico número 0.

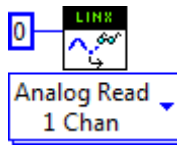


Figura 230: Bloque de Lectura Analógico Potenciómetro práctica 7.

10. El valor adquirido por el potenciómetro varía entre 0V y 5V, el mismo es escalado al intervalo 0.5 PWM y 0.99 PWM, que sirve para variar la velocidad del motor.

Se necesita la ecuación de la recta que la aproxima, para lo cual se debe seleccionar dos puntos cuales quiera de la gráfica P0=(X0,Y0) y P1=(X1,Y1), resultando las siguientes ecuaciones:

Ecuación 9: Ecuación conversión Voltaje a PWM práctica 7.

$$Y - Y_0 = mx + (X - X_0)$$

$$\text{Siendo } m = \frac{Y_1 - Y_0}{X_1 - X_0} = \frac{0.99 - 0.5}{4.99 - 0} = 0.1$$

Donde:

$$\begin{aligned} 0 \text{ Voltios} &\rightarrow 0.5 \text{ PWM} \\ 4.99 \text{ Voltios} &\rightarrow 0.99 \text{ PWM} \\ Y &= 0.5 + 0.1X \end{aligned}$$

Ecuación 10: Ecuación de la recta conversión PWM a RPM Aprox. práctica 7.

$$Y - Y_0 = mx + (X - X_0)$$

$$\text{Siendo } m = \frac{Y_1 - Y_0}{X_1 - X_0} = \frac{262 - 240}{0.99 - 0.5} = 44.89$$

Donde:

$$\begin{aligned} 0.5 \text{ PWM} &\rightarrow 240 \text{ Vueltas} \\ 0.99 \text{ PWM} &\rightarrow 262 \text{ Vueltas} \end{aligned}$$

$$Y = 217.55 + 44.89X$$

11. Para activar el Motor DC, utilizar un bloque de escritura digital PWM (Set Duty Cycle) y seleccione el puerto digital 2.



Figura 231: Bloque de Escritura Digital PWM Motor DC práctica 7.

12. Para recibir los pulsos emitidos por el Sensor Hall cada vez que el imán pasa frente a el mismo, utilizar el bloque de Lectura Digital (**Digital Read**) y seleccione el puerto digital número 4.



Figura 232: Bloque de Lectura Digital Sensor Hall práctica 7.

13. Para realizar el conteo de pulsos del motor DC se debe usar las siguientes herramientas:

Shift Register (Clic derecho sobre el While Loop > Add Shift Register).

Esta herramienta permitirá que el conteo no se reinicie y además que se actualice correctamente.

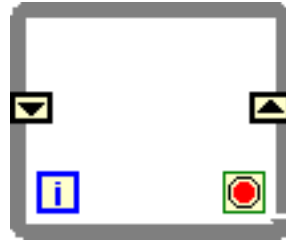


Figura 233: While Loop con Shift Register práctica 7.

Increment (Functions > Numeric)

Esta función suma una unidad al valor de entrada y muestra el resultado a la salida.



Figura 234: Función Increment práctica 7.

Select (Functions > Comparison)

Esta función devuelve el valor conectado a la entrada t o entrada f, dependiendo del valor de s. Si s es VERDADERO, esta función devuelve el valor conectado a t. Si s es FALSO, esta función devuelve el valor conectado a f.



Figura 235: Función Select práctica 7.

14. Cuando el Sensor Hall emita un pulso negativo la función **Select** muestra el valor alojado en la entrada t que es el incremento de unidad al valor alojado en el **Shift Register** del lado izquierdo y será actualizado de acuerdo a los giros que de rotor del motor.

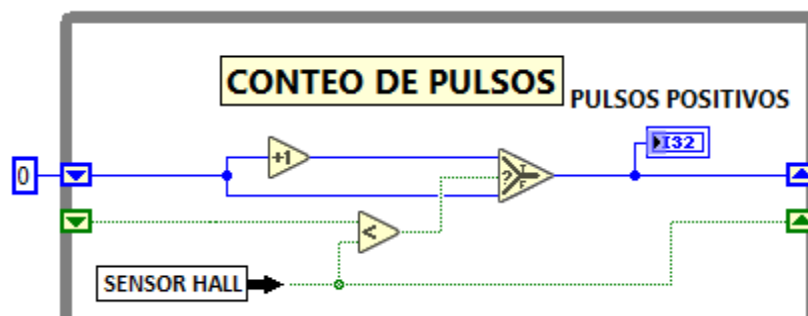


Figura 236: Contador de vueltas práctica 7.

15. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 237: Bloque de cierre de comunicación práctica 7.

16. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:
 - Haga clic en **Tools > MakerHub > LINX >LINX Firmware Wizard.**
 - En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next**.
 - Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next**.
 - Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next**.
 - Verificar que los LEDs **TX** y **RX** deben permanecer encendidos durante unos segundos mientras se implementa el firmware.
 - Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.
17. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.
18. Dar clic en el botón **Run Continuously**.
19. Regular la velocidad de giro del motor utilizando el Potenciómetro
20. Observar la variación del indicador numérico a través de la pantalla del módulo.

DIAGRAMA DE CONEXIONES

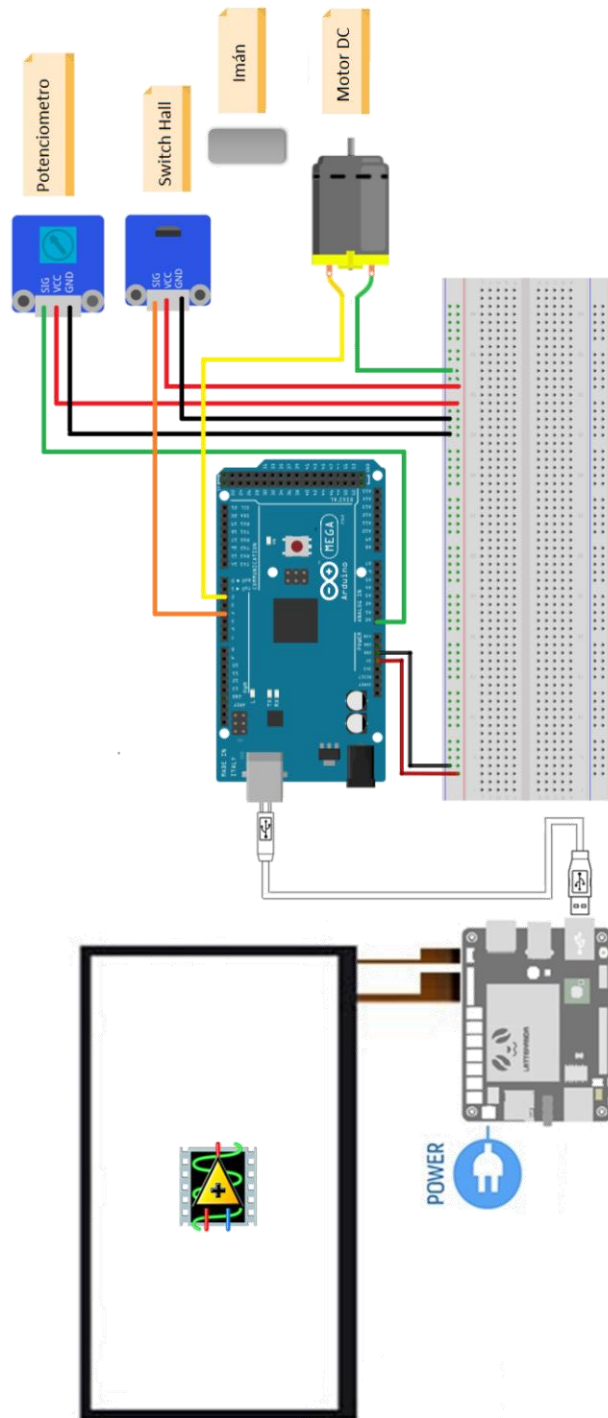


Figura 238: Diagrama de conexiones de la Práctica 7.

Tabla 8: Conexiones de la Práctica 7.

| POTENCIÓMETRO | ARDUINO MEGA 2560 |
|---------------|-------------------|
| SIG | A0 |
| VCC | 5V |
| GND | GND |
| SENSOR HALL | ARDUINO MEGA 2560 |
| SIG | 2 |
| VCC | 5V |
| GND | GND |
| MOTOR DC | ARDUINO MEGA 2560 |
| VCC | 4 |
| GND | GND |

DISEÑOS EN LABVIEW



Figura 239: Panel Frontal de la Práctica 7.

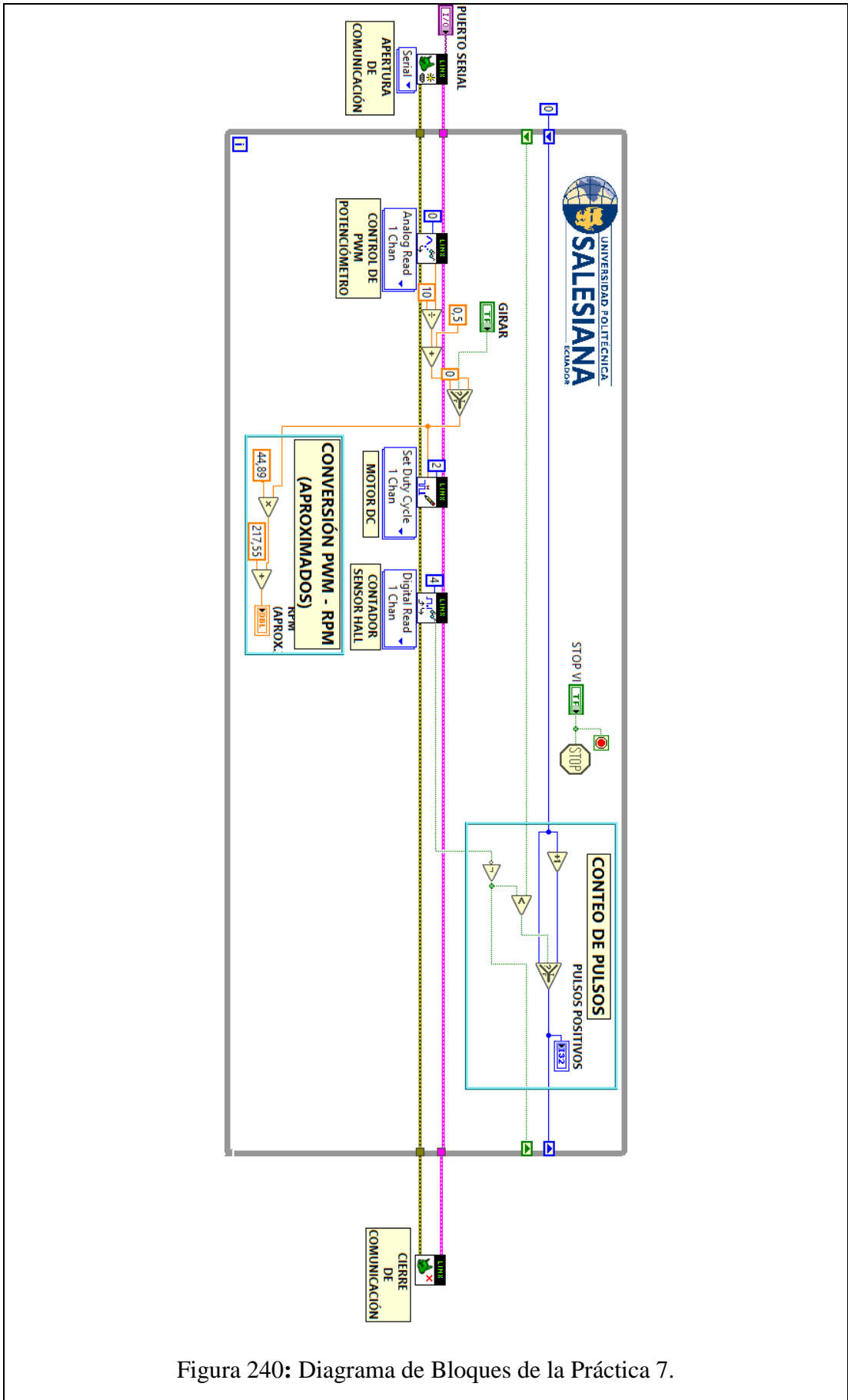


Figura 240: Diagrama de Bloques de la Práctica 7.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. Mediante el uso del Sensor Hall se capturó los pulsos, los mismos son generados cuando al imán colocado en el rotor del motor pasa delante del sensor.
3. Para la regulación del PWM fue necesario realizar la medición con carga del motor DC, de tal manera que se obtuvieron los datos que fueron colocados en la ecuación de la recta donde se obtienen los valores de regulación que serán controlados por el potenciómetro.

ANALISIS DE RESULTADO:


El control de velocidad del motor se realiza mediante PWM en un rango de 0.5 a 0.9, dicho PWM es controlado mediante el potenciómetro.

El sensor hall emite un pulso cada vez que el imán pasa delante del mismo, con esta acción se produce un incremento en el indicador numérico ubicado en el panel frontal.

Mediante un Switch virtual ubicado en el panel frontal y haciendo uso de la pantalla táctil se puede detener el giro del motor.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

4.8. Práctica # 8

| | | |
|--|---|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 8 | TÍTULO PRÁCTICA: Control de posición de un Servo-Motor. |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual permita controlar la posición entre 0 a 180° de un servo-motor a través de un teclado virtual. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar el Servo-motor en el software LabVIEW 2016 para que a través del uso de un teclado virtual controlar su posición en un rango entre 0 a 180°. | | |
| Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Elaborar una base de datos donde registren los eventos y datos obtenidos por el sensor. | | |
| INSTRUCCIONES: | | 1. Alimentar el módulo principal con 110v. |
| | | 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. |
| | | 3. Seguir la guía paso a paso para realizar la conexión del módulo. |
| | | 4. Realizar una base de datos en ACCESS para registrar los eventos y datos obtenidos por el sensor. |
| MARCO TEORICO | | |
| Servo-Motor Introducción Un servo-motor, es un tipo de motor con la capacidad especial de controlar su posición. Al hablar de un servo-motor se hace referencia a un sistema compuesto por componentes mecánicos como la caja de engranaje y electrónicos como el controlador (García, 2016). | | |

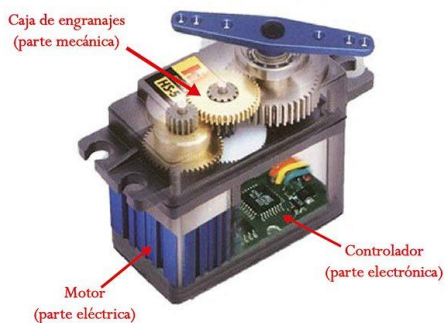


Figura 241: Servo-Motor práctica 8. (García, 2016).

Principios experimentales

El diagrama de bloque del servo-motor representa de forma visual el funcionamiento del servo-motor como un sistema (García, 2016). El bloque del circuito electrónico es el encargado de recibir la señal de entrada PWM y traducirla en movimiento del Motor DC (García, 2016). El eje del motor DC está conectado a un potenciómetro, el cual permite formar un divisor de voltaje (García, 2016). El voltaje en la salida del divisor varía en función de la posición del eje del motor DC (García, 2016).

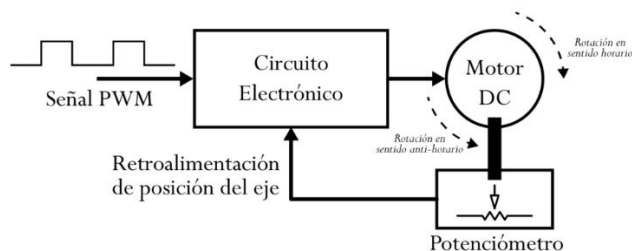


Figura 242: Diagrama de bloques del servo-motor práctica 8. (García, 2016).

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Servo-motor** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Diseñar el esquema de la práctica planteada en el panel frontal del software **LabVIEW 2016**.
4. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.

ELABORACIÓN DEL TECLADO

5. Colocar en el panel frontal los botones de del 1 al 9 y uno extra que contenga la leyenda limpiar, cambiar el nombre de todos los botones por el número y leyenda correspondiente.
6. Definir el valor inicial colocando fuera del While Loop usando una constante String, utilizaremos la constante **Empty String Constant (Functions > Programming > String)**.
7. Colocar un **Shift Register** en el **While Loop** y conectar a la constante ingresada.


| |
|---|
| 8. Dentro del diagrama de bloques colocar la función Estructura de evento (<i>Functions > Programming > Structures > Event Structure</i>). |
| 9. El evento 0 “Time out” debe quedar vacío. |
| 10. El pin derecho conectarlo al pin izquierdo del Shift Register derecho. |
| 11. Agregar 1 evento a la estructura (<i>Clic derecho en el selector de eventos > Add event case</i>). |
| 12. Seleccionar uno de los botones que se encuentran identificados en la ventana Event Sources > Controls y automáticamente se añadirán en la ventana Event Specifiers . |
| 13. En la ventana Events dar clic sobre el menú Mouse y seleccionar la opción Mouse Down . |
| 14. Para configurar los siguientes botones dar clic en el botón Add Event . |
| 15. Repetir el mismo pasó para cada uno de los dígitos del 2 al 0. |
| 16. Desplegar el Event Data Node hasta que aparezca la opción CtlRef y agregar un Boolean Strict (<i>Clic derecho el pin > Create > Property for Boolean (strict) Class > Boolean Text > Text</i>) que permitirá convertir el carácter del botón pulsado en una variable String. |
| 17. Para lograr concatenar los dígitos que presionaremos se debe utilizar la función Concatenate Strings (<i>Functions > Programming > String</i>). |
| 18. Conectar el pin derecho del shift register izquierdo al primer pin del bloque Concatenate String , el pin del Boolean Strict debe conectarse en el segundo pin, la salida debe conectarse al túnel creado anteriormente. |
| 19. Agregar 1 evento a la estructura para configurar el botón Limpiar . |
| 20. Repetir los pasos realizados con los dígitos para el botón Limpiar . |
| 21. Dentro de este evento colocar una Empty String Constant conectarla al túnel derecho, esto permitirá que al momento de presionar el botón Limpiar se borre todo lo escrito. |
| 22. Para convertir el dato String en un número se debe usar la función Decimal String to Number (<i>Functions > Programming > String > Number/String Conversion</i>) y conectar en el pin izquierdo el extremo del túnel. |
| 23. Para utilizar los bloques de las funciones de LINX , seguir los siguientes pasos: <ul style="list-style-type: none"> • Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques. • Seleccionar el submenú MakerHub. • Seleccionar el submenú LINX. |
| 24. Para establecer la comunicación, colocar el bloque OPEN al principio y conectarlo en serie con los demás bloques a utilizarse. |
|  |
| <p>Figura 243: Bloque de apertura de comunicación práctica 8.</p> |
| 25. Para establecer la comunicación con el servo-motor, colocar el subVI de LINX llamado Servo OPEN (<i>Functions > Makerhub > Linx > Sensors > Motion > Servo</i>) a continuación del bloque de apertura de comunicación, asignar el puerto digital PWM número 2 y conectarlo en serie con los demás bloques a utilizarse. |



Figura 244: Bloque de apertura de comunicación del servo-motor práctica 8.

26. Para escribir la posición del Servo-motor, utilizar el subVI llamado **Servo Set Pulse Width** (*Functions > Makerhub > Linx > Sensors > Motion > Servo*) de LINX y asigne el puerto digital PWM número 2 de la tarjeta Arduino.



Figura 245. Bloque de Escritura Digital PWM del servo-motor práctica 8.

27. El servo-motor utilizado para esta práctica trabaja en el rango de 600-2300us.

Se utilizará la siguiente fórmula para realizar la conversión de grados a ancho de pulso de acuerdo al valor ingresado:

Ecuación 11: Grados de posición del servo-motor práctica 8.

$$\mu S(\text{Ancho de pulso}) = 9.44 * \text{GRADOS} + 600$$

Dónde **GRADOS** es el valor ingresado por el usuario mediante el teclado virtual.

28. Se colocaron condiciones del tamaño de caracteres o cantidad de dígitos posibles de ingresar, utilizando la función **String Length** (*Functions > Programming > String*) se obtiene el tamaño de la variable string conectada en su izquierdo.

29. Para cerrar la comunicación con el servo-motor, colocar el subVI de LINX llamado **Servo Close** (*Functions > Makerhub > Linx > Sensors > Motion > Servo*) detrás del bloque de cierre comunicación, asignar el puerto digital PWM número 2.



Figura 246: Bloque de cierre de comunicación del servo-motor práctica 8.

30. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 247: Bloque de cierre de comunicación práctica 8.

31. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

32. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 248: Bloque de apertura de conexión de Database práctica 8.

33. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 249: Bloque para inserta datos en Database práctica 8.

34. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 250: Bloque de cierre de conexión de Database práctica 8.

CONFIGURACIÓN DE DATABASE

35. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

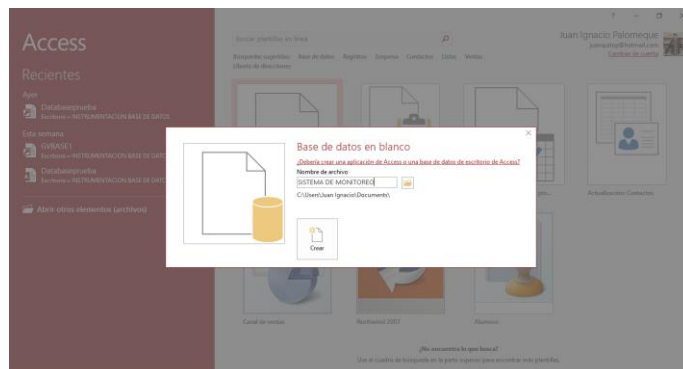


Figura 251: Creación de base de datos en MICROSOFT ACCESS práctica 8.

36. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESSS 2002-2003**.

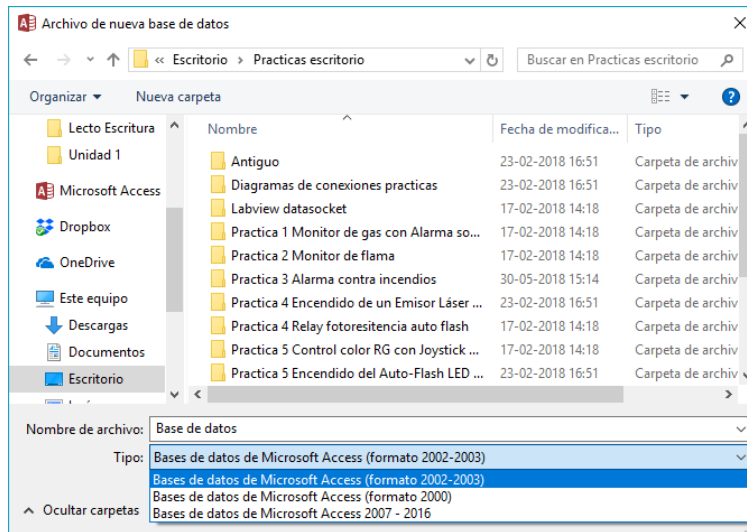


Figura 252: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 8.

37. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

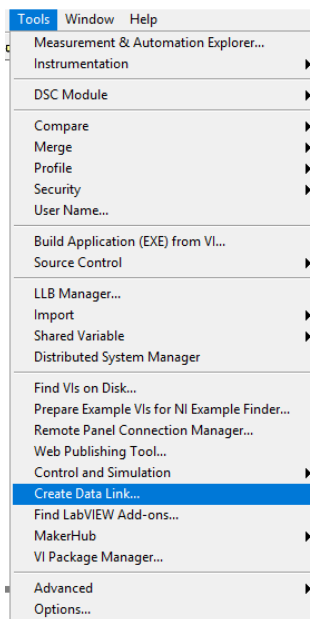


Figura 253: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 8.

38. Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

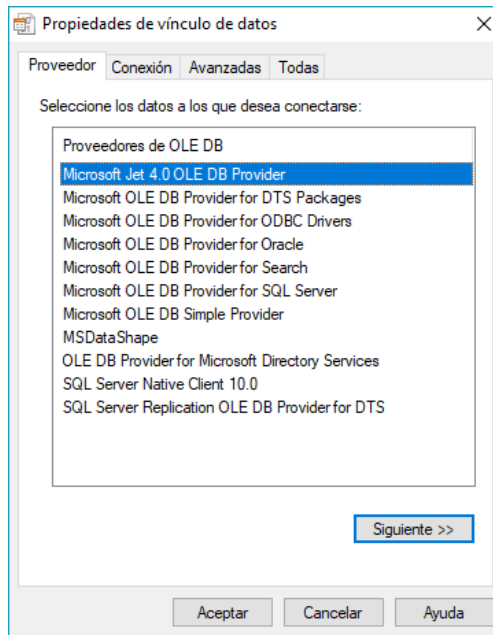


Figura 254: Ventana de propiedades de vínculo de datos práctica 8.

39. Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

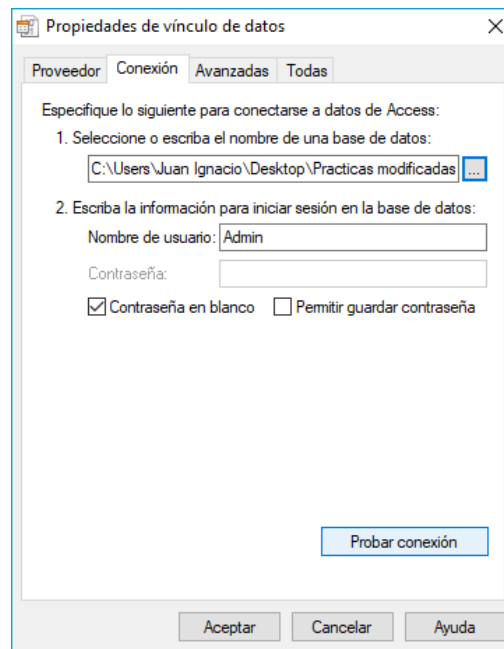


Figura 255: Selección de ruta donde se ubica la base de datos práctica 8.

40. Una vez enlazada la ruta se procede a probar conexión.

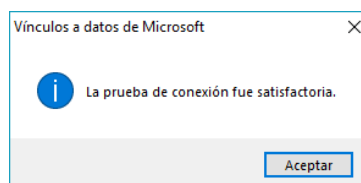


Figura 256: Venta de prueba de conexión de la vinculación de datos práctica 8.

41. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

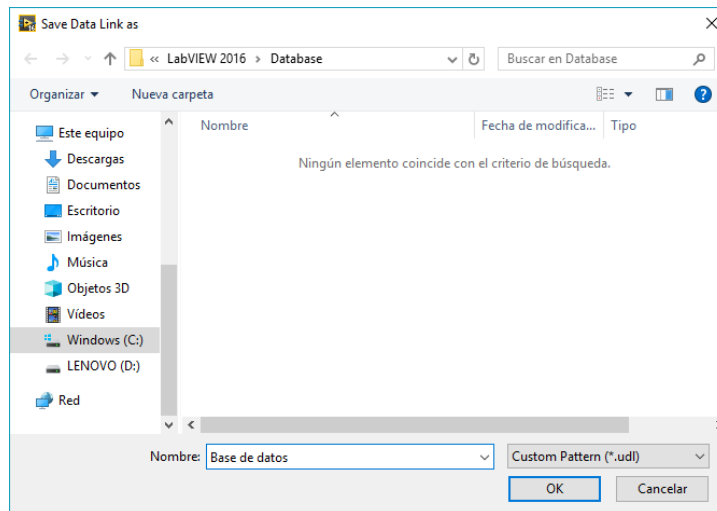


Figura 257: Ventana de asignación de nombre para la tabla de registro de datos práctica 8.

42. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

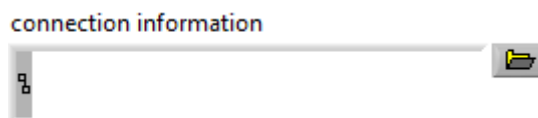


Figura 258: Control de ruta de base de datos práctica 8.

43. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:

- Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard.**
- En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next.**
- Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next.**
- Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next.**
- Verificar que los LEDs **TX** y **RX** deben permanezcan encendidos durante unos segundos mientras se implementa el firmware.
- Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.

44. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

45. Dar clic en el botón **Run Continuously.**

46. Regular la posición del servo-motor utilizando el potenciómetro del panel frontal.

47. Observar el movimiento del servo-motor ubicándose en la posición correcta de acuerdo a los grados colocados en el teclado virtual entre 0-180°.

48. Verificar que la conversión se esté realizando de manera correcta.

DIAGRAMA DE CONEXIONES

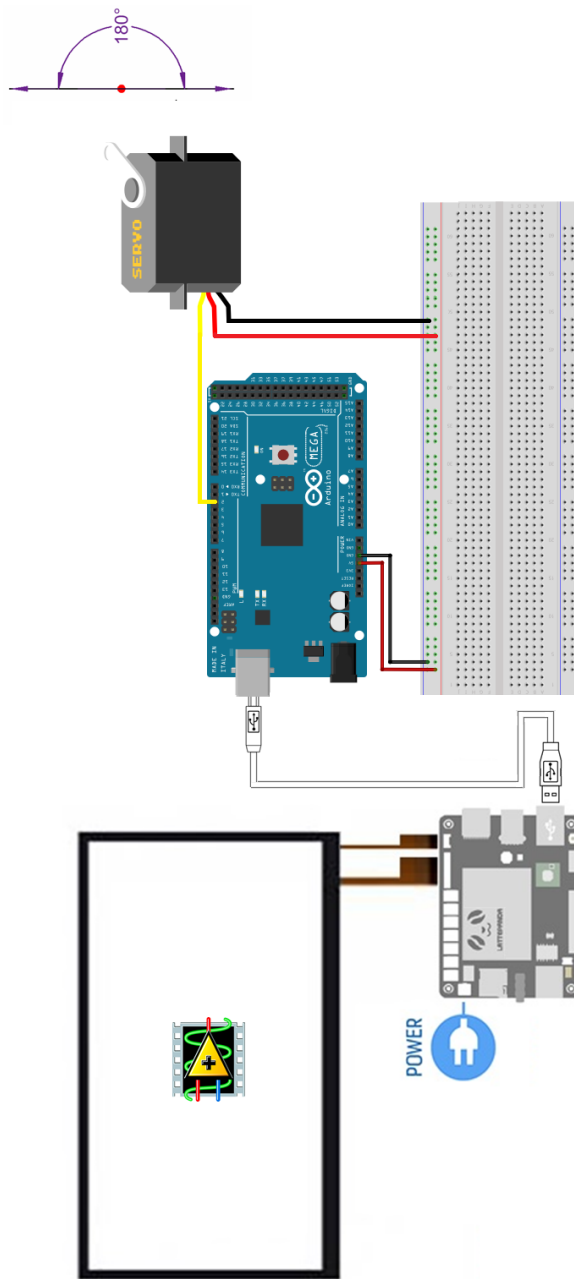


Figura 259: Diagrama de conexiones de la Práctica 8.

Tabla 9: Conexiones de la Práctica 8.

| SERVO-MOTOR | ARDUINO MEGA 2560 |
|-------------|-------------------|
| SIG | 2 |
| VCC | 5V |
| GND | GND |



Figura 260: Panel Frontal de la Práctica 8.

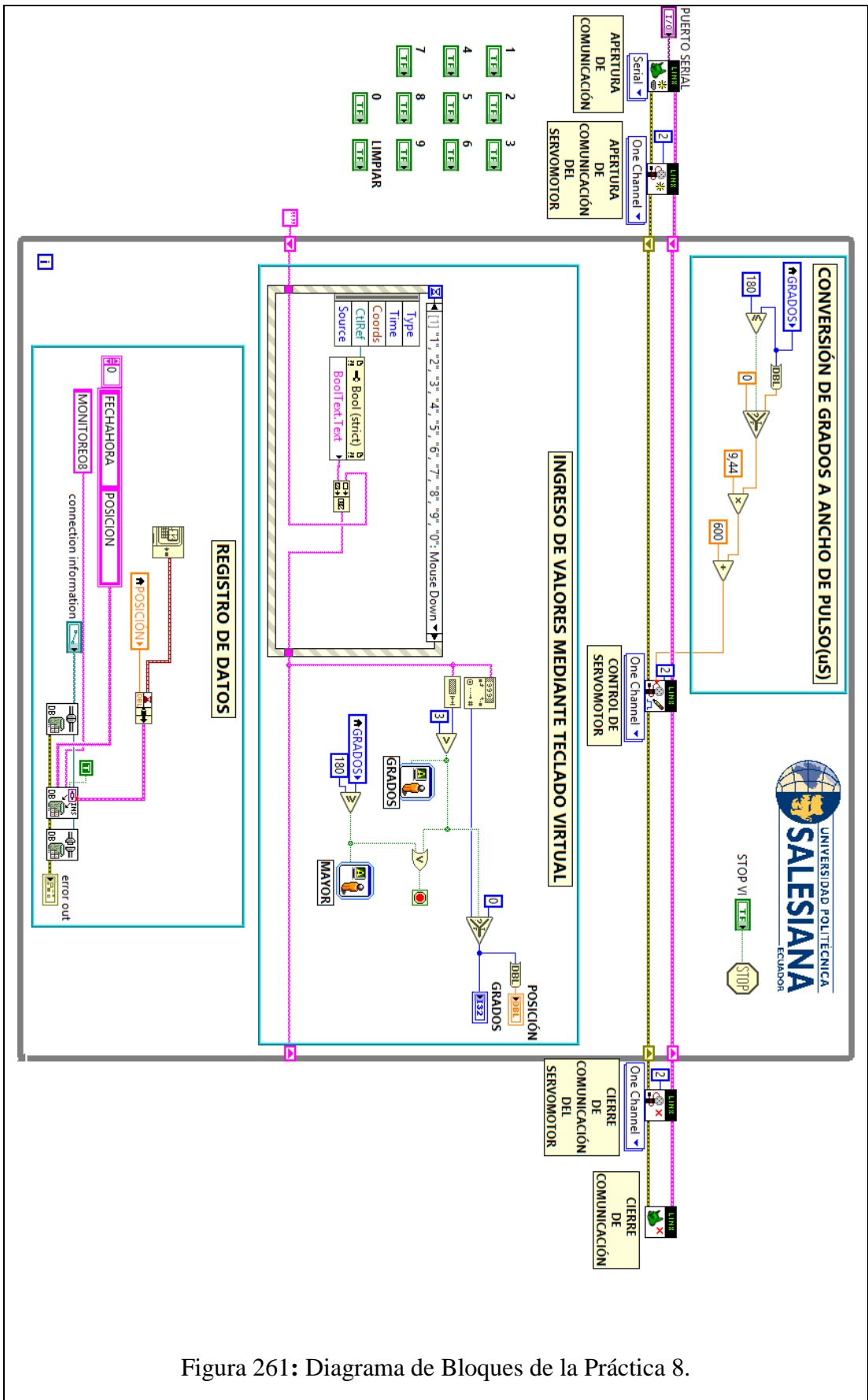


Figura 261: Diagrama de Bloques de la Práctica 8.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. El servo-motor utilizado para esta práctica trabaja en el rango de 600-2300us. Por lo cual fue necesaria la conversión de grados a ancho de pulso.
3. Por medio del uso de la pantalla táctil, se logró implementar un teclado virtual, el cual permite introducir la posición deseada para el servo-motor en un rango de 0 a 180 grados.
4. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.

ANALISIS DE RESULTADO:

El control del servo-motor se realiza mediante un teclado virtual, con el mismo se selecciona la posición en un rango de 0 grados hasta 180 grados.

Si el valor ingresado excede el rango se muestra un mensaje que indica al usuario esta novedad regresando a su posición inicial.

Si el valor ingresado excede el tamaño de 3 dígitos se muestra un mensaje que indica al usuario esta novedad regresando a su posición inicial.

Los datos ingresados mediante el teclado son grabados en una base de datos.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

4.9. Práctica # 9

| | | |
|---|---|---|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 9 | TÍTULO PRÁCTICA: Simulación de proceso de embotellado. |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual simule el proceso de una fábrica embotelladora. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Utilizar los elementos del módulo tales como Micro-Switch Roller y Micro-Switch Palanca, Sensor de obstáculos infrarrojo en el software LabVIEW 2016, para que controlen el funcionamiento del proceso, el cual simule el proceso de una fábrica embotelladora. El proceso debe incluir simulación de elaboración del envase, llenado del envase, tapado del envase. El proceso debe ser repetitivo. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Elaborar una base de datos donde registre la cantidad de botellas elaboradas. | | |
| INSTRUCCIONES: | | <ol style="list-style-type: none"> 1. Alimentar el módulo principal con 110v. 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. 3. Seguir la guía paso a paso para realizar la conexión del módulo. 4. Realizar una base de datos en ACCESS para registrar la cantidad de botellas elaboradas. |
| MARCO TEORICO | | |
| Sensor de evasión de obstáculos Introducción Un sensor de obstáculos IR funciona de acuerdo con el principio de reflexión infrarroja para detectar obstáculos. Cuando delante del mismo no se encuentra algún objeto, el receptor infrarrojo no recibe señales; cuando hay un objeto por delante que bloquea y refleja la luz infrarroja, el receptor de infrarrojos recibirá señales (Sunfounder, 2013). | | |



Figura 262: Sensor de evasión de obstáculos práctica 9. (Sunfounder, 2013).

Principios experimentales

Un sensor de evitación de obstáculos consiste principalmente en un transmisor de infrarrojos, un infrarrojo receptor y un potenciómetro el cual permite el ajuste de la distancia de detección del sensor infrarrojos. De acuerdo con el carácter reflector de un objeto, si no hay obstáculo, el rayo infrarrojo emitido se debilitará con la distancia que se propaga y finalmente desaparecer. Si hay un obstáculo, cuando el rayo infrarrojo lo encuentra, el rayo se reflejará de nuevo en el receptor de infrarrojos. Entonces el receptor de infrarrojos detecta esta señal y confirma un obstáculo en el frente (Sunfounder, 2013).

Micro-Switch

Introducción

Un Micro-switch o Micro-interruptor es un elemento eléctrico/electrónico que se activa con la aplicación de una fuerza en su terminal, la conmutación se produce en posiciones específicas y repetibles (Perez, 2013). Son muy utilizados debido a su bajo costo y durabilidad (Perez, 2013).



Figura 263: Micro-Switch de palanca larga práctica 9.



Figura 264: Micro-Switch roller práctica 9.

Principio Experimental

Existen varios tipos de micro-switch, para esta práctica se utilizaron 2 de ellos, el micro-switch de palanca larga AM1701 y el micro-switch roller AM1703.

Motor DC

Introducción

El motor de corriente continua DC es una máquina que transforma la energía eléctrica en energía mecánica, generando un movimiento de rotación, gracias a la acción de un campo magnético (Gottlieb, 1994).



Figura 265: Motor DC práctica 9. (García, 2013).

Principio Experimental

Cuando la corriente eléctrica transita a través del bobinado del electroimán giratorio, se genera un campo electromagnético que interactúa con el campo magnético del imán permanente (Álvarez, 2015). El conductor genera una fuerza perpendicular al plano formado por el campo magnético y la corriente, de acuerdo con la Fuerza de Lorentz (Álvarez, 2015):



Ecuación 12: Fuerza de Lorents práctica 9.

$$F = B \cdot L \cdot I \cdot \text{sen}(\phi)$$

- F: Fuerza en newtons
- I: Intensidad que recorre el conductor en amperios
- L: Longitud del conductor en metros
- B: Densidad de campo magnético o densidad de flujo teslas
- Φ : Ángulo que forma I con B

ACTIVIDADES POR DESARROLLAR

1. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.
2. Conectar el **Micro-Switch Roller** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
3. Conectar el **Micro-Switch Palanca** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.
4. Conectar el **Sensor de evasión de obstáculos** a la placa **Arduino MEGA 2560** de acuerdo al diagrama de conexiones y la tabla de conexiones.

| |
|---|
| 5. Conectar el Motor DC simulador de llenado a la placa Arduino MEGA 2560 de acuerdo al diagrama de conexiones y la tabla de conexiones. |
| 6. Conectar el Motor DC encargado del movimiento del nivel a la placa Arduino MEGA 2560 de acuerdo al diagrama de conexiones y la tabla de conexiones. |
| 7. Diseñar el esquema de la práctica planteada en el panel frontal del software LabVIEW 2016 . |
| 8. Debido a que el proceso debe ser repetitivo se recomienda usar un While Loop y dentro del mismo colocar la secuencia. |
| 9. Para realizar la secuencia de manera ordenada utilizar la herramienta Flat Sequence . |
| 10. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI. |
| 11. Para utilizar los bloques de las funciones de LINX , seguir los siguientes pasos: <ul style="list-style-type: none"> • Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques. • Seleccionar el submenú MakerHub. • Seleccionar el submenú LINX. |
| 12. Para establecer la comunicación, colocar el bloque OPEN al principio y conectarlo en serie con los demás bloques a utilizarse. |
|  |
| Figura 266: Bloque de apertura de comunicación práctica 9. |
| 13. El proceso empieza una vez que se presione el botón “Inicio”. |
| 14. Utilizar la estructura Flat Sequence (Functions > Structures > Flat Sequence) para realizar cada paso de acuerdo al proceso establecido. |
| 15. Dentro de cada secuencia o frame debe colocar un While Loop (Functions > Structures) , dentro del mismo la acción correspondiente al proceso, deben ser montados tal como se muestra en el diagrama de bloques de la práctica. |
| 16. Para leer el estado del Micro-Switch Roller utilizado en esta práctica utilizar el subVI Read (Functions > Maker Hub > LINX > Peripherals > Digital) desarrollado por LINX , colocar el número 2 en el pin DI Channel. |
|  |
| Figura 267: Bloque de Lectura Digital Micro-Switch Roller práctica 9. |
| 17. Para leer el valor analógico dado por el potenciómetro del sensor de obstáculos utilizar el subVI Read (Functions > Maker Hub > LINX > Peripherals > Analog) desarrollado por LINX , colocar el número 0 en el pin A0 Channel. |

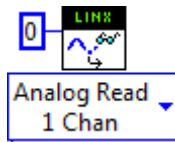


Figura 268: Bloque de Lectura Analógica Sensor de obstáculos práctica 9.

18. Se realizó la conversión de voltaje entregado por el potenciómetro a centímetros, se tomó como punto final o máximo el nivel se realizó la siguiente ecuación:

Para determinar el valor final en centímetros se necesita la recta que la aproxima, para lo cual se debe seleccionar dos puntos cuales quiera de la gráfica P0=(X0,Y0) y P1=(X1,Y1), resultando la ecuación de la recta:

Ecuación 13: Ecuación de la recta práctica 9.

$$Y - Y_0 = mx + (X - X_0)$$

$$\text{Siendo } m = \frac{Y_1 - Y_0}{X_1 - X_0} = \frac{3.9 - 2.7}{2 - 3.3} = -0.92$$

Donde:

$$\begin{aligned} 2 \text{ Centímetros} &\rightarrow 3.9 \text{ Voltios} \\ 3.3 \text{ Centímetros} &\rightarrow 2.7 \text{ Voltios} \end{aligned}$$

$$Y = 5.74 - 0.92X$$

19. Para leer el estado del **Sensor de obstáculos infrarrojo** utilizado en esta práctica utilizar el subVI **Digital Read (Functions > Maker Hub > LINX > Peripherals)** desarrollado por **LINX**, colocar el número 3 en el pin DI Channel.



Figura 269: Bloque de Lectura Digital Sensor de Obstáculos práctica 9.

20. Para controlar el movimiento se utiliza el motor de la estructura encargada de mover el nivel, mediante el uso de los siguientes bloques:

Set Duty Cycle: Definir el pin digital 4 y conectar al pin GND del motor al puerto digital 4, definir el valor 0.2 para el pin PWM, esto permite un movimiento suave del nivel.

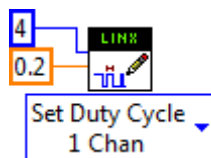


Figura 270: Bloque de escritura PWM motor de nivel práctica 9.

Digital Write: Definir el pin digital 8 y conectar al pin VCC del motor, definir el estado False para el valor de salida, de esta manera se entrega la tierra necesaria para energizar la bobina del motor.



Figura 271: Bloque de Escritura Digital motor de nivel práctica 9.

21. Dentro de este proceso se coloca la animación de la variable “Botellas” para que dicha animación vaya acompañada con el movimiento del nivel, utilizar el Loop iteration.

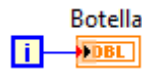


Figura 272: Animación de la variable Botella práctica 9.

22. Agregar un frame y codificar el movimiento del inyector quien realiza un movimiento descendente en línea recta.

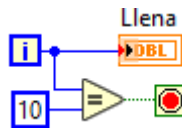


Figura 273: Animación de la variable Llena práctica 9.

Aquí también establecer las condiciones de detección del motor correspondiente al riel de la siguiente manera:

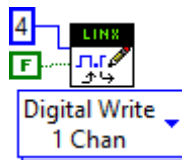


Figura 274: Bloque de Escritura Digital motor de nivel práctica 9.



Figura 275: Bloque de Escritura Digital motor de nivel práctica 9.

23. Utilizar un bloque de Escritura Digital y asignar el pin 5 para activar el motor que se encuentra debajo del nivel, el mismo estará activado durante 5 segundos simulando el llenado de la botella.

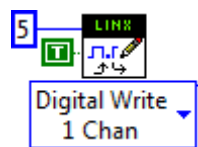


Figura 276: Bloque de escritura digital activación motor inyector práctica 9.

24. En el siguiente frame realizar el movimiento ascendente del inyector y condicionar para que una vez en su posición de inicio pase al siguiente frame, utilizar el siguiente diseño:

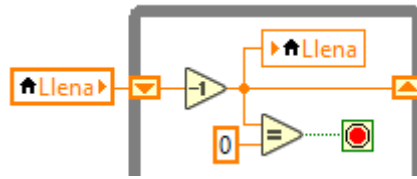


Figura 277: Movimiento ascendente del inyector práctica 9.

Además, configurar la detección del motor de llenado:



Figura 278: Bloque de escritura digital desactivación motor inyector práctica 9.

25. Agregar un frame y codificar el movimiento del colocador de tapas quien realiza un movimiento descendente en línea recta.

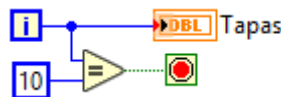


Figura 279: Animación de la variable Tapas práctica 9.

26. En el siguiente frame realizar el movimiento ascendente del colocador de tapas y condicionar para que una vez en su posición de inicio pase al siguiente frame, utilizar el siguiente diseño:

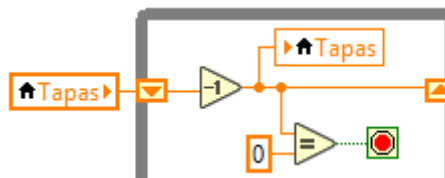


Figura 280: Movimiento ascendente del colocador de tapas práctica 9.

27. Añadir un nuevo frame, aquí se produce el avance hacia el proceso final de la botella, este movimiento físico irá acompañado de la animación respectiva:

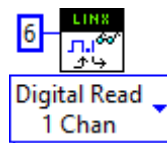


Figura 281: Bloque de Lectura Digital Sensor de Obstáculos práctica 9.

28. Establecer como condición que cuando se active el micro-switch palanca y la animación haya llegado a su fin, es decir la variable botellas 2 sea igual a 5 se detenga el VI pase al siguiente frame:

Dentro de este proceso se coloca la animación de la variable botellas 2 para que dicha animación vaya acompañada con el movimiento del nivel, utilizar el Loop iteration.

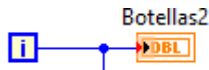


Figura 282: Animación de la variable Botellas2 práctica 9.

29. Para controlar el movimiento se utiliza el motor de la estructura encargada de mover el nivel, mediante el uso de los siguientes bloques:

Set Duty Cycle: Definir el pin digital 4 y conectar al pin GND del motor al puerto digital 4, definir el valor 0.5 para el pin PWM, esto permite un movimiento suave del nivel.

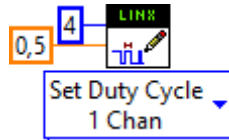


Figura 283: Bloque de escritura PWM motor de nivel práctica 9.

Digital Write: Definir el pin digital 8 y conectar al pin VCC del motor, definir el estado False para el valor de salida, de esta manera se entrega la tierra necesaria para energizar la bobina del motor.



Figura 284: Bloque de Escritura Digital motor de nivel práctica 9.

Realizar un contador que sume cada vez que se cumpla la condición planteada anteriormente, seguir la siguiente estructura:

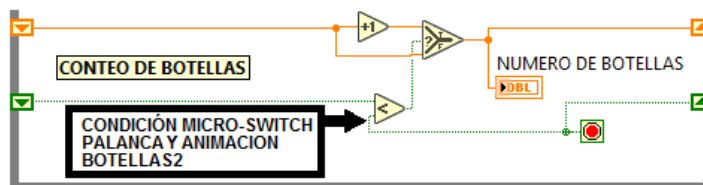


Figura 285: Contador de botellas práctica 9.

30. En el siguiente frame se programa el retroceso del nivel a su estado inicial, para ello seguir los siguientes pasos:

Utilizar el bloque de lectura digital y definir el pin de lectura digital 2, conectar el pin de lectura del bloque al Loop Condition, esto permite que el proceso se reinicie debido a que la activación del Micro-Switch Roller detiene el While Loop y la marcha del motor.



Figura 286: Bloque de Lectura Digital Micro-Switch Roller práctica 9.

Para controlar el movimiento se utiliza el motor de la estructura encargada de mover el nivel, mediante el uso de los siguientes bloques:

Set Duty Cycle: Definir el pin digital 4 y conectar al pin GND del motor al puerto digital 4, el valor PWM para este pin está condicionado por el estado del Micro-Switch Roller, por ello en el pin PWM se coloca un selector de la siguiente manera:

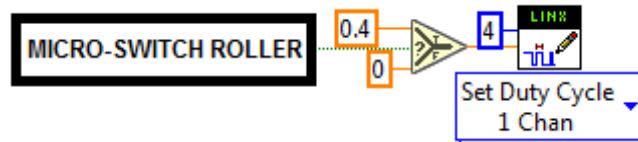


Figura 287: Bloque de escritura PWM motor de nivel práctica 9.

Digital Write: Definir el pin digital 8 y conectar al pin VCC del motor, definir el estado True para el valor de salida, de esta manera se entrega la tierra necesaria para energizar la bobina del motor.



Figura 288: Bloque de Escritura Digital motor de nivel práctica 9.

31. Para que el proceso se repita de manera cíclica todos los pasos deben ir colocados dentro de un While Loop y colocar una constante booleana false conectado al Loop Condition.
32. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 289: Bloque de cierre de comunicación práctica 9.

33. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

34. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 290: Bloque de apertura de conexión de Database práctica 9.

35. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 291: Bloque para inserta datos en Database práctica 9.

36. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 292. Bloque de cierre de conexión de Database práctica 9.

CONFIGURACIÓN DE DATABASE

37. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

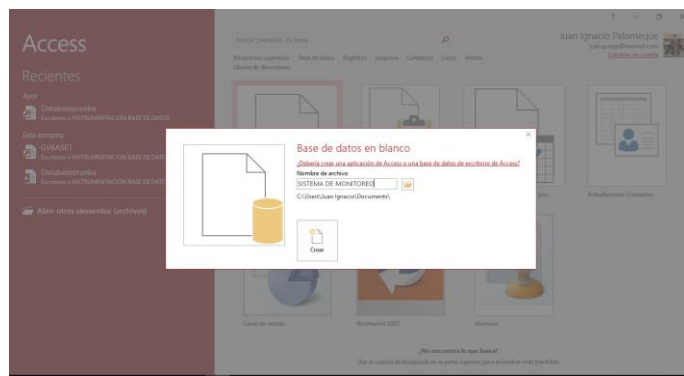


Figura 293: Creación de base de datos en MICROSOFT ACCESS práctica 9.

38. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESSS 2002-2003**.

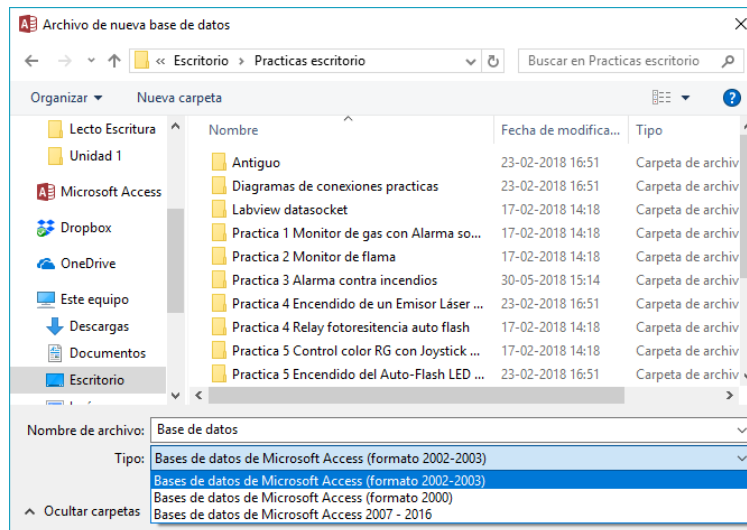


Figura 294: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 9.

39. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

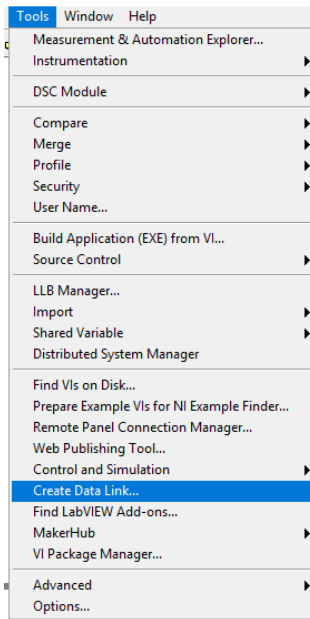


Figura 295: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 9.

40. Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

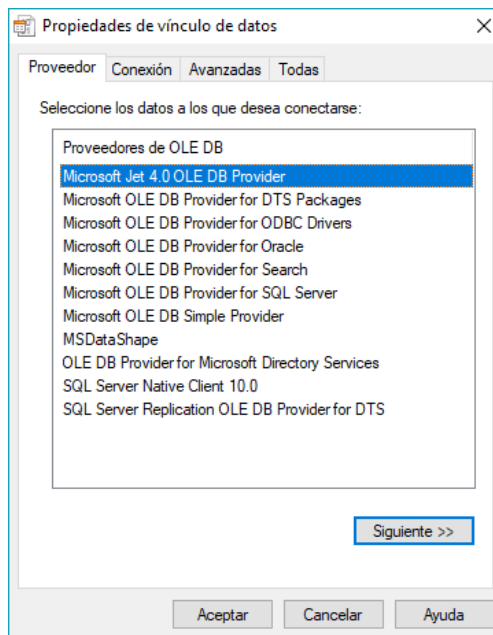


Figura 296: Ventana de propiedades de vínculo de datos práctica 9.

41. Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

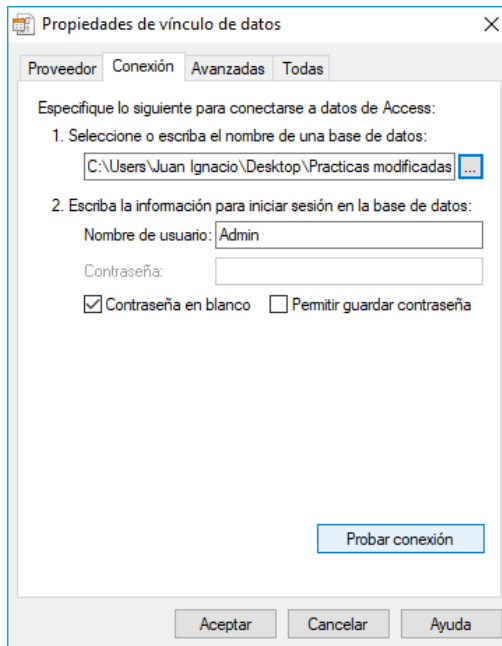


Figura 297: Selección de ruta donde se ubica la base de datos práctica 9.

42. Una vez enlazada la ruta se procede a probar conexión.

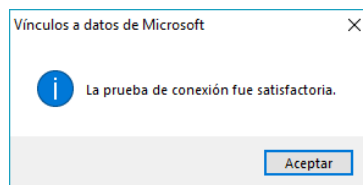


Figura 298: Venta de prueba de conexión de la vinculación de datos práctica 9.

43. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

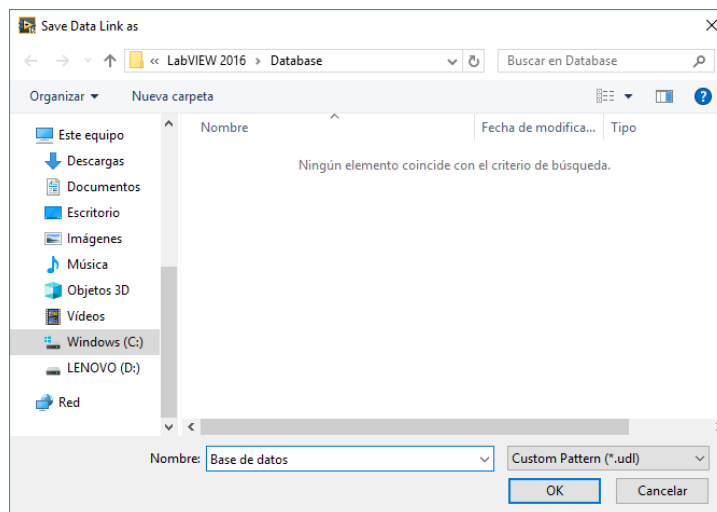


Figura 299: Ventana de asignación de nombre para la tabla de registro de datos práctica 9.

44. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

connection information



Figura 300: Control de ruta de base de datos práctica 9.

45. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:

- Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard.**
- En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next.**
- Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next.**
- Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next.**
- Verificar que los LEDs **TX** y **RX** deben permanezcan encendidos durante unos segundos mientras se implementa el firmware.
- Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.

46. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.

47. Dar clic en el botón **Run Continuously.**

48. Presionar el botón verde denominado **Inicio.**

49. Observar el funcionamiento del proceso en cada una de sus etapas.

50. Verificar el funcionamiento del contador de botellas.

51. Comprobar que el proceso sea cíclico de manera que no se interrumpa en ninguna de sus etapas.

52. En caso de querer detener la simulación pulsar el botón rojo denominado **Paro VI.**

DIAGRAMA DE CONEXIONES

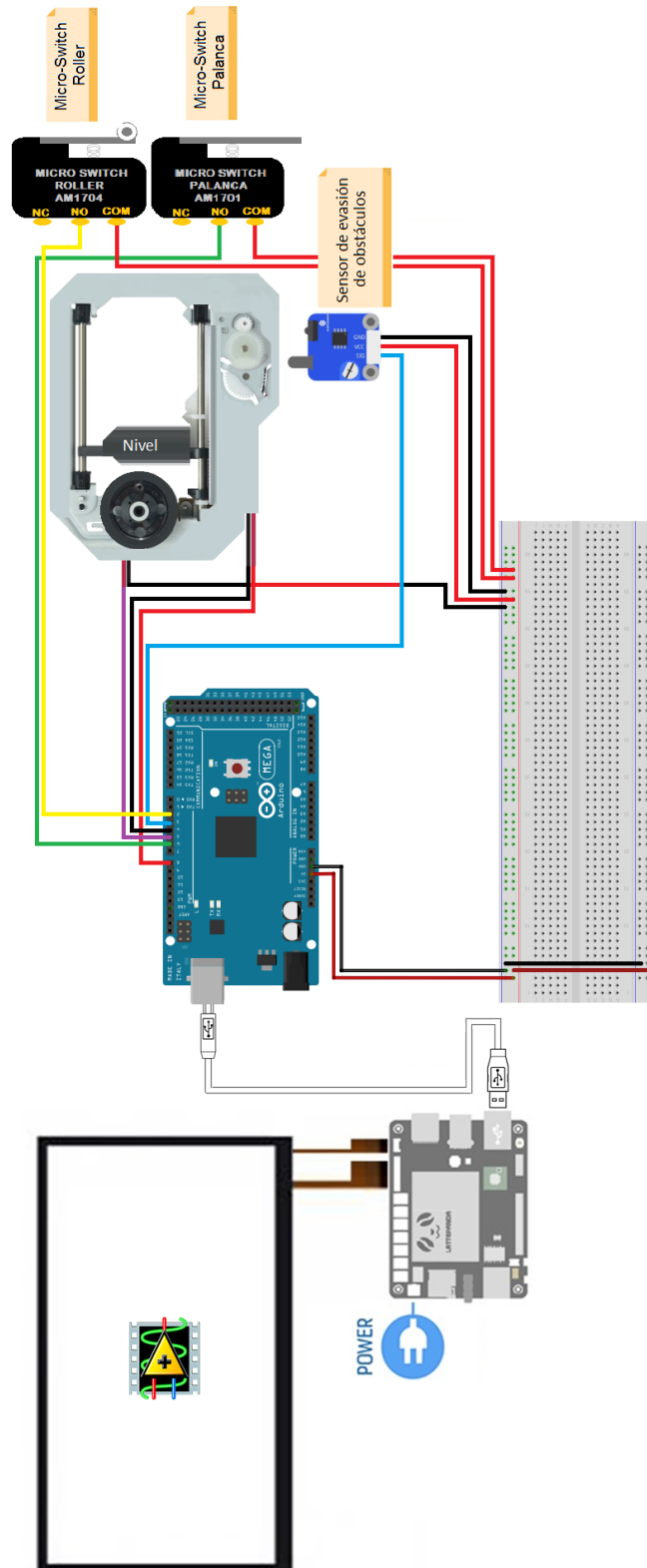


Figura 301: Diagrama de conexiones de la Práctica 9.

Tabla 10: Conexiones de la Práctica 9.

| | |
|--|--------------------------|
| MOTOR DE NIVEL | ARDUINO MEGA 2560 |
| NEGATIVO | 4 |
| POSITIVO | 8 |
| MOTOR DE LLENADO | ARDUINO MEGA 2560 |
| VCC | 5 |
| GND | GND |
| MICRO-SWITCH ROLLER | ARDUINO MEGA 2560 |
| COM | 5V |
| NO | 2 |
| MICRO-SWITCH PALANCA | ARDUINO MEGA 2560 |
| COM | 5V |
| NO | 6 |
| SENSOR DE EVASIÓN DE OBSTÁCULOS | ARDUINO MEGA 2560 |
| VCC | VCC |
| GND | GND |
| SIG | 3 |

DISEÑOS EN LABVIEW



Figura 302: Panel Frontal de la Práctica 9.

CONCLUSIONES:

1. Se confirmó el funcionamiento de la práctica a través de la intervención de todos los elementos planteados para el desarrollo de la misma.
2. Los Micro-Switch utilizados en esta práctica son de gran utilidad ya que permiten emular elementos utilizados en la industria tales como fines de carrera.
3. El uso del Sensor IR de Obstáculos es de gran utilidad para proyectos de escala reducida como este caso ya que puede comportarse como un interruptor no mecánico y fácil de usar.

ANALISIS DE RESULTADO:

El Micro-Switch tipo Roller actúa como condición de inicio del proceso, es decir, si no se encuentra activado, el proceso no inicia.

El motor encargado de mover la estructura es regulado mediante PWM, esto permite realizar movimientos suaves y así evitar choques bruscos.

Cuando el nivel se encuentra frente al sensor de obstáculos, el mismo se activa y envía la señal y así detener el movimiento del nivel


Cuando el nivel se encuentra frente al sensor de obstáculos, el mismo se activa y envía la señal y así detener el movimiento del nivel.

El Micro-Switch tipo Palanca se activa cuando el nivel ha recorrido todo el camino disponible, en este momento, el nivel retrocede hasta llegar al principio del camino, activando el Micro-Switch Roller, reiniciando así el proceso.

El conteo de las botellas elaboradas se registra en una base de datos al momento de finalizar cada ciclo de elaboración.

Cuando se presiona el botón denominado “Paro VI” haciendo uso de la pantalla táctil, el VI se detiene.

4.10. Práctica # 10

| | | |
|--|----|--|
|  | | GUÍA DE PRÁCTICA DE LABORATORIO |
| CARRERA: Ingeniería Electrónica | | ASIGNATURA: Sensores y Transductores |
| NRO. PRÁCTICA: | 10 | TÍTULO PRÁCTICA: Control de temperatura. |
| Objetivo: Diseñar en el software LabVIEW 2016 un programa, el cual se capaz de mantener una temperatura deseada. | | |
| Objetivos específicos: Reconocer las conexiones principales del módulo. Establecer comunicación entre el software de desarrollo gráfico LabVIEW 2016 y la placa Arduino Mega 2560. Configurar un PID mediante los datos obtenidos por el sensor de temperatura LM35 en el software LabVIEW 2016 que sea capaz de mantener una temperatura constante por medio de un foco de 12V y un ventilador. Elaborar un programa en el software LabVIEW 2016, que sea capaz de comunicar el sistema embebido LattePanda con la placa Arduino Mega 2560. Elaborar una base de datos donde registren los eventos y datos obtenidos por el sensor. | | |
| INSTRUCCIONES: | | 1. Alimentar el módulo principal con 110v. |
| | | 2. Verificar que se tienen todos los elementos necesarios para realizar la práctica. |
| | | 3. Seguir la guía paso a paso para realizar la conexión del módulo. |
| | | 4. Realizar una base de datos en ACCESS para registrar los eventos y datos obtenidos por el sensor. |
| MARCO TEORICO | | |
| Sensor de temperatura LM35 | | |
| Introducción LM35 es un sensor de temperatura producido por National Semiconductor. Tiene una precisión operativa muy alta y un amplio rango de operación. Con un tamaño pequeño, bajo costo y confiabilidad, LM35 se aplica ampliamente en ingeniería. Como usa una compensación interna, la salida puede comenzar con 0 °C. LM35 tiene muchos paquetes diferentes. Por debajo de la temperatura normal, el LM35 no requiere calibración adicional para alcanzar la precisión de más o menos 1/4 °C. El modo de fuente de alimentación se puede clasificar como fuente de alimentación | | |

única y fuente de alimentación doble positiva y negativa. Sus pines son como se muestra a continuación. En el modo de fuente de alimentación dual positiva y negativa, puede medir una temperatura negativa. En el modo de fuente de alimentación única y 25 °C, la corriente de reposo es de aproximadamente 50µA y tiene un amplio rango de tensión de funcionamiento, entre 4 y 20V, ahorrando electricidad (Sunfounder, 2013).

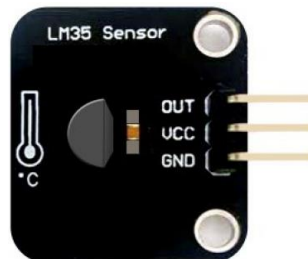


Figura 304: Módulo de sensor de temperatura práctica 10. (Sunfounder, 2013).

Principios experimentales

El voltaje de salida del LM35 es proporcional a la temperatura Celsius. Cuando se coloca en 0 °C temperatura ambiente, dará salida a 0V. La tensión de salida aumentará 10 mV cada vez que la temperatura aumente en 1 °C. La fórmula de cálculo es la siguiente:

Ecuación 14: Voltaje de salida del sensor LM35 práctica 10.

$$V_{out_LM35}(T) = 10 \text{ mV}/^{\circ}\text{C} \times T^{\circ}\text{C}$$

Modulo Relé doble

Introducción

El relé es un dispositivo que se utiliza para proporcionar la conexión entre dos o más puntos o dispositivo en respuesta a la señal de entrada aplicada. En otras palabras, el relé proporciona aislamiento entre el controlador y el dispositivo, ya que los dispositivos pueden funcionar tanto en CA como en DC. Sin embargo, reciben señales del micro-controlador que funciona en CC, por lo que se requiere un relé para cerrar la brecha. El relé es extremadamente útil cuando necesita controlar una gran cantidad de corriente o voltaje con poca señal eléctrica (Sunfounder, 2013).



Figura 305: Modulo Relé doble práctica 10. (Sunfounder, 2013).

Principio experimental

Relé: hay 5 partes en cada relé:

1. Electroimán: Consiste en un núcleo de hierro herido por una bobina de cables. Cuando pasa la electricidad, se vuelve magnética. Por lo tanto, se llama electroimán.
2. Armadura: la banda magnética móvil se conoce como armadura. Cuando la corriente fluye a través de ellos, la bobina está energizada, produciendo así un campo magnético que se utiliza para crear o romper los puntos normalmente abiertos (N / O) o normalmente cercanos (N / C). Y la armadura se puede mover con corriente continua (CC) así como con corriente alterna (AC).
3. Resorte: cuando no circulan corrientes a través de la bobina en el electroimán, el resorte aleja la armadura para que el circuito no pueda completarse.
4. Conjunto de contactos eléctricos: hay dos puntos de contacto:
 - Normalmente abierto: conectado cuando el relé está activado y desconectado cuando está inactivo.
 - Normalmente cerca: no está conectado cuando el relé está activado y se conecta cuando está inactivo.
5. Marco moldeado - Los relés están cubiertos con plástico para protección.

Trabajo de retransmisión

El principio de funcionamiento del relé es simple. Cuando se suministra energía al relé, las corrientes comienzan a fluir a través de la bobina de control; Como resultado, el electroimán comienza a energizarse. Luego, la armadura se atrae hacia la bobina, bajando el contacto móvil y conectándose así con los contactos normalmente abiertos. Entonces, el circuito con la carga está energizado. Luego, romper el circuito sería un caso similar, ya que el contacto móvil se elevará a los contactos normalmente cerrados bajo la fuerza del resorte. De esta manera, la conexión y desconexión del relé puede controlar el estado de un circuito de carga (Sunfounder, 2013).

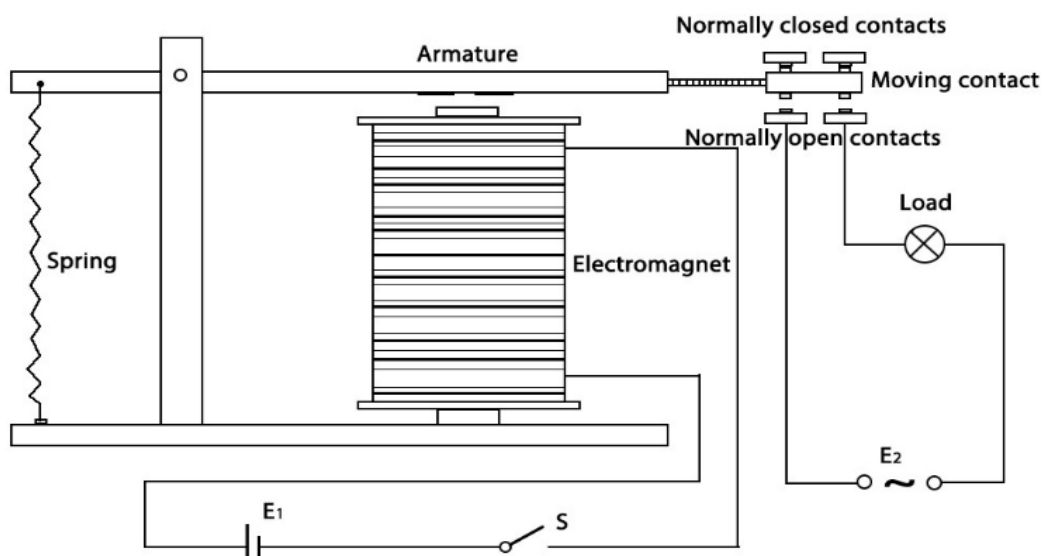


Figura 306: Esquema de un relé práctica 10. (Sunfounder, 2013).

ACTIVIDADES POR DESARROLLAR

MATLAB

1. Tomar datos de muestra para establecer un rango de temperatura entregado por el elemento electrónico LM35, y se arma una tabla que será usada para el control PID.
2. Abrir el software Matlab 2013 y se procede a crear variables en el Workspace, ambas variables contienen los datos obtenidos anteriormente.

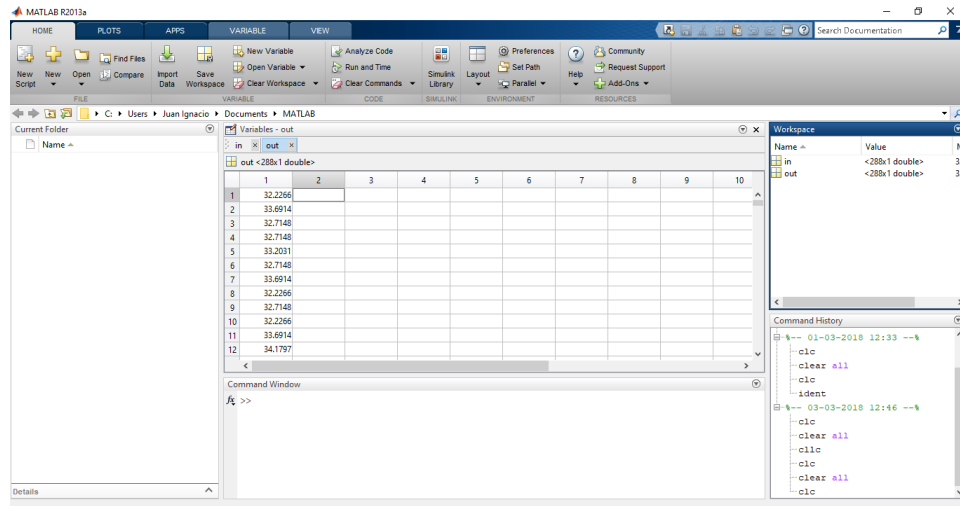


Figura 307: Creación de variables en el Matlab.

The screenshot shows the 'Variables - in' window in MATLAB. The window displays a table with 21 rows and 9 columns. The data in the 'in' window is as follows:

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|----|---------|---|---|---|---|---|---|---|---|
| 1 | 27.1000 | | | | | | | | |
| 2 | 27.2000 | | | | | | | | |
| 3 | 27.4000 | | | | | | | | |
| 4 | 27.3000 | | | | | | | | |
| 5 | 27.3000 | | | | | | | | |
| 6 | 28.2000 | | | | | | | | |
| 7 | 28.2000 | | | | | | | | |
| 8 | 28.2000 | | | | | | | | |
| 9 | 28.2000 | | | | | | | | |
| 10 | 28.2000 | | | | | | | | |
| 11 | 28.4000 | | | | | | | | |
| 12 | 28.9000 | | | | | | | | |
| 13 | 29.5000 | | | | | | | | |
| 14 | 30.1000 | | | | | | | | |
| 15 | 30.4000 | | | | | | | | |
| 16 | 30.8000 | | | | | | | | |
| 17 | 32.8000 | | | | | | | | |
| 18 | 32.9000 | | | | | | | | |
| 19 | 36.1000 | | | | | | | | |
| 20 | 40.2000 | | | | | | | | |
| 21 | | | | | | | | | |

Figura 308: Tabla de datos de variable de entrada denominada in.

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|----|---------|---|---|---|---|---|---|---|---|
| 1 | 27.1000 | | | | | | | | |
| 2 | 27.2000 | | | | | | | | |
| 3 | 27.4000 | | | | | | | | |
| 4 | 27.3000 | | | | | | | | |
| 5 | 27.3000 | | | | | | | | |
| 6 | 28.2000 | | | | | | | | |
| 7 | 28.2000 | | | | | | | | |
| 8 | 28.2000 | | | | | | | | |
| 9 | 28.2000 | | | | | | | | |
| 10 | 28.2000 | | | | | | | | |
| 11 | 28.4000 | | | | | | | | |
| 12 | 28.9000 | | | | | | | | |
| 13 | 29.5000 | | | | | | | | |
| 14 | 30.1000 | | | | | | | | |
| 15 | 30.4000 | | | | | | | | |
| 16 | 30.8000 | | | | | | | | |
| 17 | 32.8000 | | | | | | | | |
| 18 | 32.9000 | | | | | | | | |
| 19 | 36.1000 | | | | | | | | |
| 20 | 40.2000 | | | | | | | | |
| 21 | | | | | | | | | |

Figura 309: Tabla de datos de variable de salida denominada out.

- Se procede a ingresar el comando “**ident**” en el Command Window, para de esta manera empezar a diseñar el control PID.
- Se abre la siguiente pantalla en la cual se elige la opción “**Time domain data**” (Dato en dominio del tiempo).

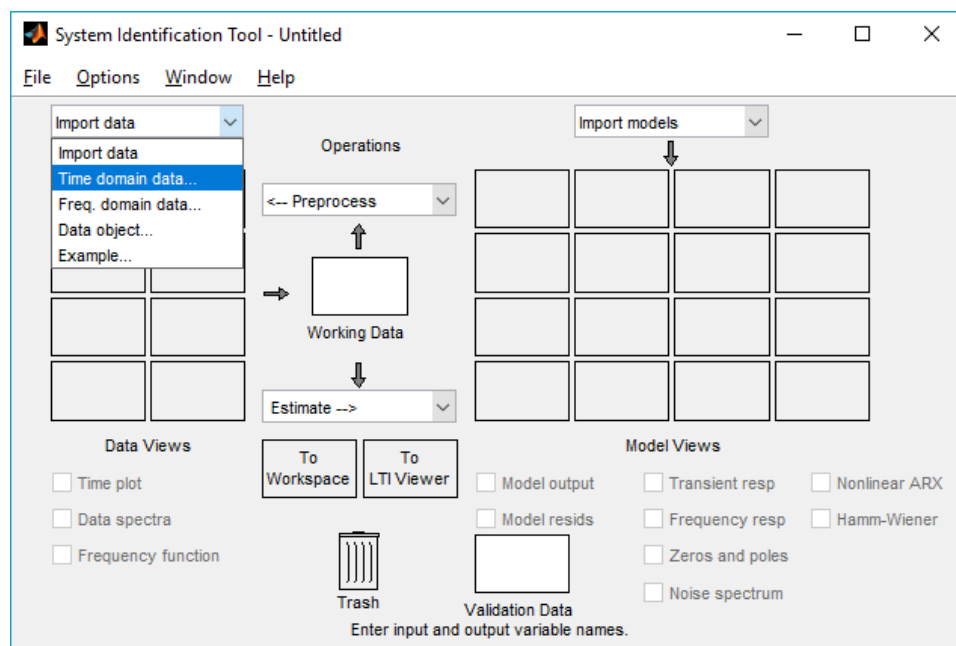


Figura 310: Ventana de herramienta Ident.

- En pantalla que aparece, introducir el nombre de las variables de entrada y de salida creadas, también puede modificar el nombre si lo desea y el tiempo de inicio, el tiempo de inicio será la contante 0 y en el campo denominado “**Sampling interval**” se debe introducir el tiempo que se tarde el sensor en obtener los datos, en este caso, 0.01s.

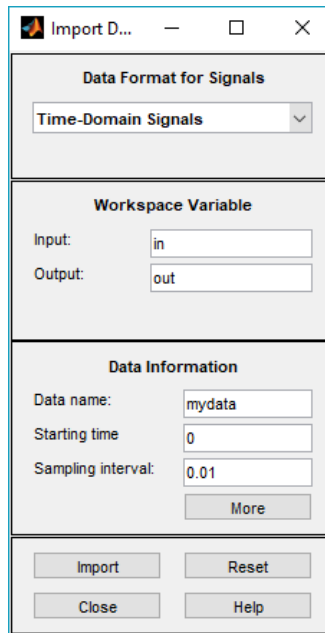


Figura 311: Introducción de datos.

6. Dar clic en el botón **“Import”** para visualizar la gráfica.
7. Dar clic en el casillero denominado **“Estimate”**, se despliega un menú en el cual se escoge la opción **“Transfer Function Models”** y se da clic.
8. A continuación, se abre una ventana en la cual se puede elegir la cantidad de polos y ceros necesarios para la función de transferencia. Una vez definido lo que se desea incluir con respecto a lo antes mencionado se procede a dar clic en el botón **“Estimate”**.

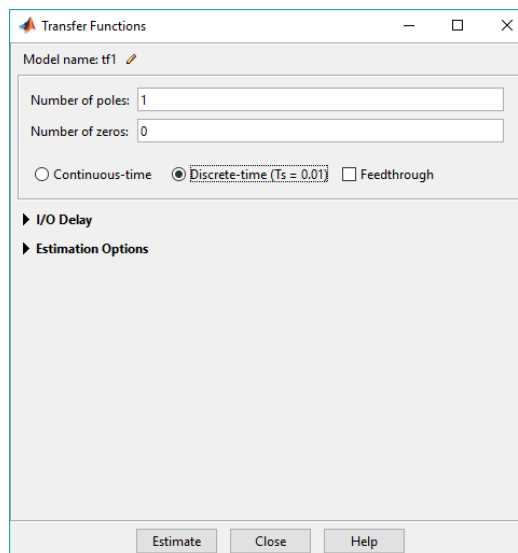


Figura 312: Selección de polos y ceros para la función de transferencia.

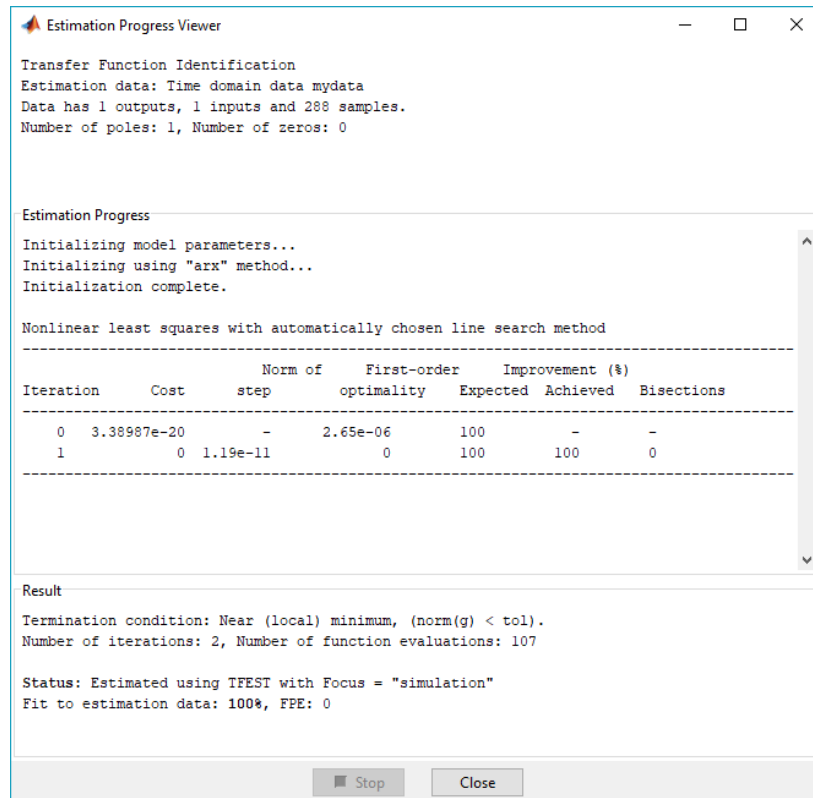


Figura 313: Obtención de valores de la función de transferencia.

- Posterior a esta acción se genera la gráfica del modelo y los valores que el programa definió para la misma grafica resultante.

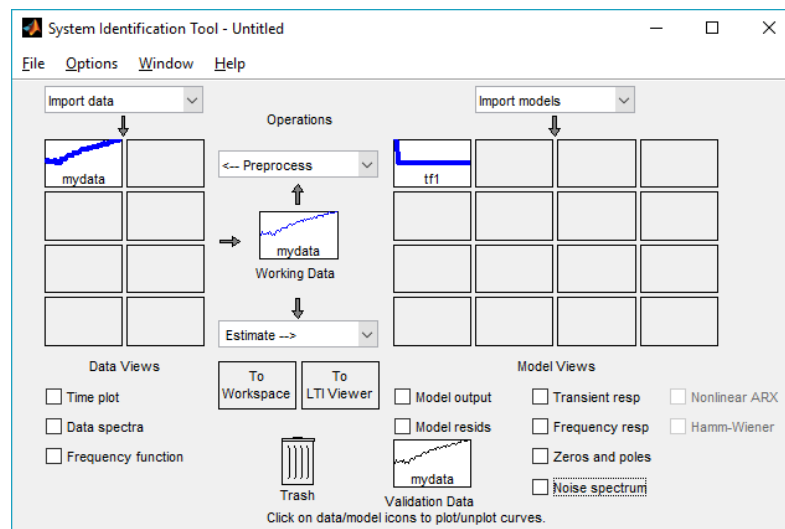


Figura 314: Modelo resultante.

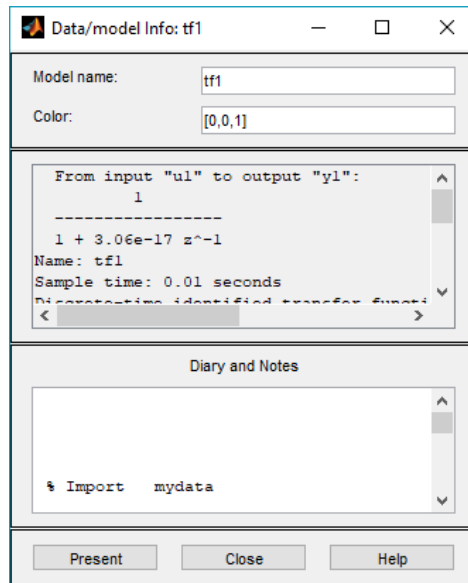


Figura 315: Parámetros para la función de transferencia.

10. Dando doble clic al resultado de la gráfica se visualizan los datos estimados por el programa, se debe tratar de que el parámetro “fit to estimation data” sea un valor muy cercano al 100%.
11. Se procede a arrastrar la gráfica resultante al Workspace.
12. Una vez realizado este paso ubicarse en la pestaña **Apps** y elegir la opción **PID Tuning** y cargar la función de transferencia.
13. En el menú de selección **Type** seleccionar los parámetros que se desean (P, PI o PID) necesarios para introducir en el software LabVIEW. En este caso escogió solo el parámetro P ya que su tiempo de estabilización es de 0 segundos, ideal para el proceso empleado.

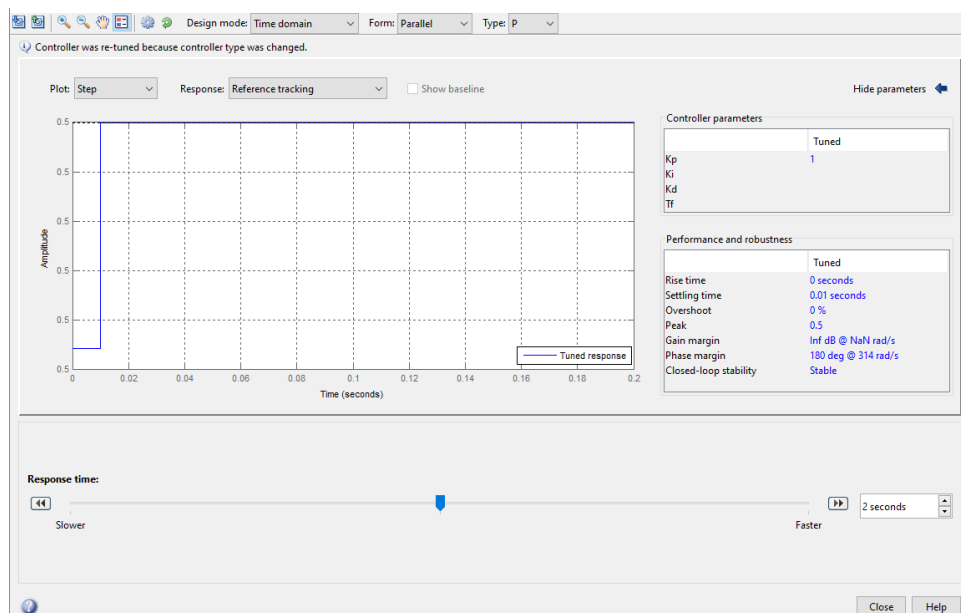

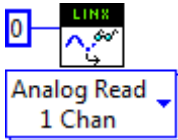



Figura 316: Parámetros de PID.

LABVIEW

14. Energizar el módulo principal para brindar la alimentación necesaria a la placa LattePanda.

| |
|---|
| <p>15. Conectar el Sensor de temperatura lm35 a la placa Arduino MEGA 2560 de acuerdo al diagrama de conexiones y la tabla de conexiones.</p> |
| <p>16. Conectar el Ventilador de acuerdo al diagrama y la tabla de conexiones.</p> |
| <p>17. Conectar el Halógeno de acuerdo al diagrama y la tabla de conexiones.</p> |
| <p>18. Diseñar el esquema de la práctica planteada en el panel frontal del software LabVIEW 2016.</p> |
| <p>19. Conectar en el diagrama de bloques del VI las funciones utilizadas en el panel frontal del VI.</p> |
| <p>20. Para utilizar los bloques de las funciones de LINX, seguir los siguientes pasos:</p> <ul style="list-style-type: none"> • Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques. • Seleccionar el submenú MakerHub. • Seleccionar el submenú LINX. |
| <p>21. Para establecer la comunicación, colocar el bloque OPEN al principio y conectarlo en serie con los demás bloques a utilizarse.</p> <div style="text-align: center;">  </div> <p style="text-align: center;">Figura 317: Bloque de apertura de comunicación práctica 10.</p> |
| <p>22. Para visualizar el dato entregado por el sensor de temperatura LM35, utilizar el bloque de lectura analógica (Analog Read) y seleccione el puerto analógico número 0. Los datos obtenidos deben ser multiplicados por 100 para obtener el valor real de temperatura.</p> <div style="text-align: center;">  </div> <p style="text-align: center;">Figura 318: Bloque de Lectura Analógica sensor LM35 práctica 10.</p> |
| <p>23. Cuando la temperatura exceda el valor del Set-Point empezará a funcionar el ventilador para controlar la temperatura, la activación del ventilador está definida por un comparador que toma los valores obtenidos por el LM35 y el Set-Point establecido.</p> |
| <p>24. Para controlar el encendido del ventilador, utilizar un bloque de escritura digital PWM (Set Duty Cycle) y seleccione el puerto digital 2, el valor ingresado en el pin de duty cycle será de 1 o 0, el mismo que es resultado de la comparación anterior.</p> <div style="text-align: center;">  </div> <p style="text-align: center;">Figura 319: Bloque de Escritura Digital PWM para giro de ventilador práctica 10.</p> |
| <p>25. Utilizar el subVI PID VI (Functions > Control & Simulation > PID) que implementa un controlador PID usando un algoritmo PID para aplicaciones PID simples o aplicaciones de control de alta velocidad que requieren un</p> |

algoritmo eficiente, cree un control en el pin **PID gains** y coloque los parámetros de PID obtenidos anteriormente.

En el pin **Output range** coloque como valor mínimo 0 y como valor máximo 1 ya que ese es el rango del duty cycle del PWM a usarse.

En el pin **Process variable**, conectar la salida del multiplicador usado para obtener los valores de temperatura.

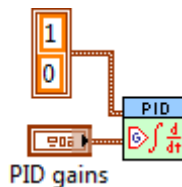


Figura 320: SubVI PID.

26. Para controlar el encendido del halógeno, utilizar un bloque de escritura digital PWM (Set Duty Cycle) y seleccione el puerto digital 4, el valor ingresado en el pin de duty cycle es de 1 o 0, el mismo que es resultado de la comparación entre el pin de salida Output del subVI PID y un valor constante de 0.5, cuando el output exceda dicho valor, se activa el relé y por ende el halógeno.



Figura 321: Bloque de Escritura Digital PWM intensidad lumínica del foco halógeno práctica 10.

27. Para cerrar la comunicación, colocar el bloque **CLOSE** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 322: Bloque de cierre de comunicación práctica 10.

28. Para utilizar los bloques de las funciones de Datalogging and Supervisory Control, seguir los siguientes pasos:

- Clic derecho en cualquier parte del espacio en blanco del diagrama de bloques.
- Seleccionar el submenú **CONNECTIVITY**.
- Seleccionar el submenú **DATABASE**.

29. Para establecer la base de datos, colocar el bloque **DB TOOLS OPEN CONNECTION** al principio y conectarlo en serie con los demás bloques a utilizarse.



Figura 323: Bloque de apertura de conexión de Database práctica 10.

30. Para insertar los datos obtenidos por los sensores en la base de datos, utilizar el bloque **DB TOOLS INSERT DATA**:



Figura 324: Bloque para inserta datos en Database práctica 10.

31. Para cerrar la conexión de la base de datos, colocar el bloque **DB TOOLS CLOSE CONNECTION** al final y conectarlo en serie con los demás bloques a utilizarse.



Figura 325: Bloque de cierre de conexión de Database práctica 10.

CONFIGURACIÓN DE DATABASE

32. Abrir el programa **MICROSOFT ACCESS**, para crear una nueva base de datos.

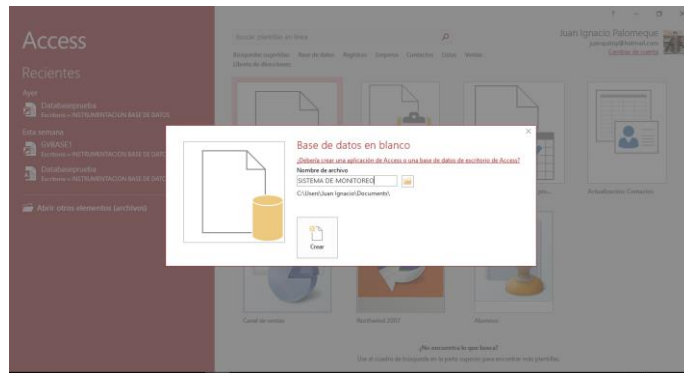


Figura 326: Creación de base de datos en MICROSOFT ACCESS práctica 10.

33. Crear un nombre para la base de datos; guardar el tipo de archivo como **MICROSOFT ACCESSS 2002-2003**.

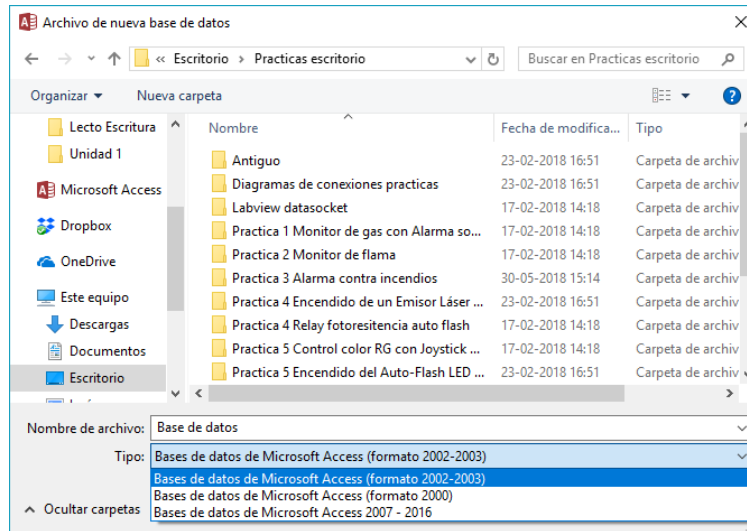


Figura 327: Creación de base de datos formato MICROSOFT ACCESS 2002-2003 práctica 10.

34. Una vez creada la Data base, establecer un data link entre LabVIEW 2016 y la base de datos creada en MICROSOFT ACCESS 2016.

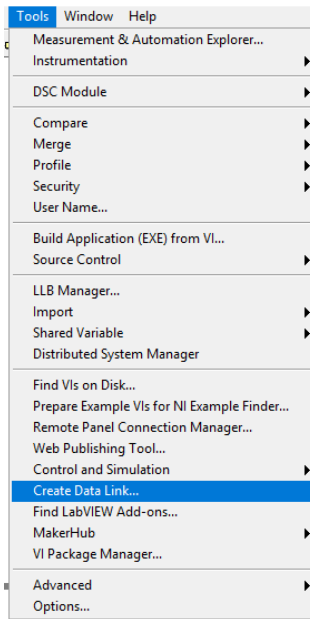


Figura 328: Inicio de creación de data link entre LabVIEW 2016 y la base de datos práctica 10.

- 35.** Configurar la vinculación de los datos. Para poder leer y escribir los datos con MICROSOFT ACCESS 2016 seleccionar la opción MICROSOFT JET 4.0 OLE DB PROVIDER.

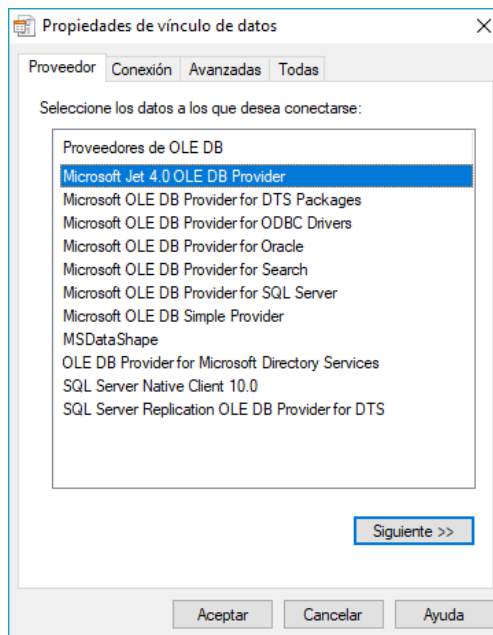


Figura 329: Ventana de propiedades de vínculo de datos práctica 10.

- 36.** Para realizar el vínculo de datos con la base de datos creada en MICROSOFT ACCESS 2016, seleccionar la ruta donde se encuentra la base de datos.

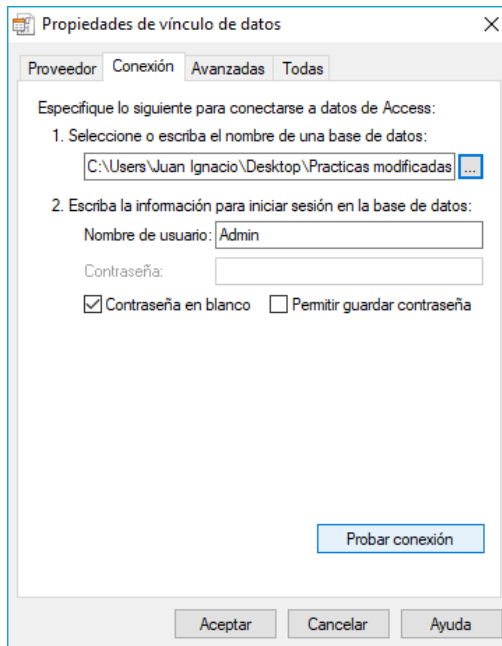


Figura 330: Selección de ruta donde se ubica la base de datos práctica 10.

37. Una vez enlazada la ruta se procede a probar conexión.

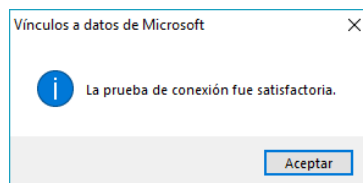


Figura 331: Vista de prueba de conexión de la vinculación de datos práctica 10.

38. Al aceptar la configuración, se visualiza la siguiente ventana, donde se debe asignar un nombre a la tabla donde se registran los datos.

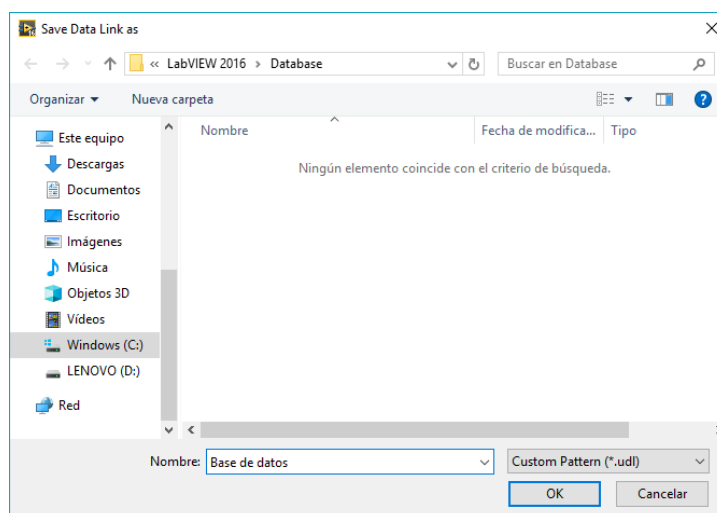


Figura 332: Ventana de asignación de nombre para la tabla de registro de datos práctica 10.

39. Para finalizar la configuración de la Base de datos, asignar la ruta de la carpeta donde se encuentra creada la Base de datos.

connection information



Figura 333: Control de ruta de base de datos práctica 10.

40. Cargar el archivo desarrollado en la placa Arduino mediante la herramienta LINX de MakerHub:
 - Haga clic en **Tools > MakerHub > LINX > LINX Firmware Wizard**.
 - En **Device Family** seleccione **Arduino**, **Device Type** seleccione **Arduino MEGA 2560** y haga clic en **Next**.
 - Seleccione el puerto **COM** en el que se encuentra conectado el dispositivo y haga clic en **Next**.
 - Seleccione la versión de firmware **LINX-Serial/USB** y haga clic en **Next**.
 - Verificar que los LEDs **TX** y **RX** deben permanezcan encendidos durante unos segundos mientras se implementa el firmware.
 - Para finalizar haga clic en **Launch Example** y siga las instrucciones en el panel frontal.
41. Una vez que se haya cargado de manera exitosa el VI en la placa Arduino MEGA 2560 se procede a ejecutar el programa.
42. Dar clic en el botón **Run Continuously**.
43. Observar la variación de la temperatura a través de la gráfica y del indicador ubicado en el panel frontal.
44. Variar el valor del Set-Point de acuerdo a las indicaciones del docente para comprobar el funcionamiento del VI.
45. Comprobar el perfecto funcionamiento del PID implementado mediante la estabilización de la temperatura del sensor LM35.

DIAGRAMA DE CONEXIONES

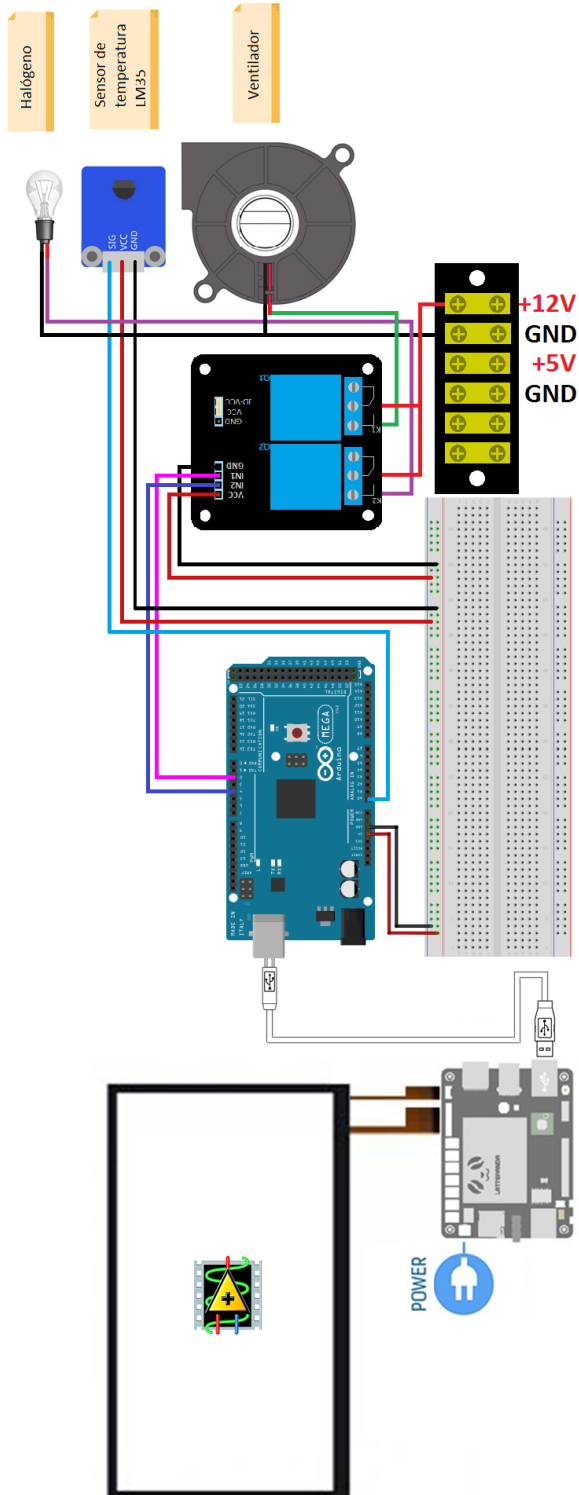


Figura 334: Diagrama de conexiones de la Práctica 10.

Tabla 11: Conexiones de la Práctica 10.

| | |
|------------------------------|--------------------------|
| VENTILADOR | RELÉ 1 |
| VCC | CONTACTO ABIERTO |
| GND | GND (BORNERA) |
| HALÓGENO 12V | RELÉ 2 |
| VCC | CONTACTO ABIERTO |
| GND | GND (BORNERA) |
| MÓDULO RELÉ | ARDUINO MEGA 2560 |
| VCC | VCC |
| IN1 | 2 |
| IN2 | 4 |
| GND | GND |
| MÓDULO RELÉ | BORNERA |
| COMUN 1 | 12V |
| COMUN 2 | 12V |
| SENSOR DE TEMPERATURA | ARDUINO MEGA 2560 |
| VCC | VCC |
| GND | GND |
| SIG | A0 |

DISEÑOS EN LABVIEW

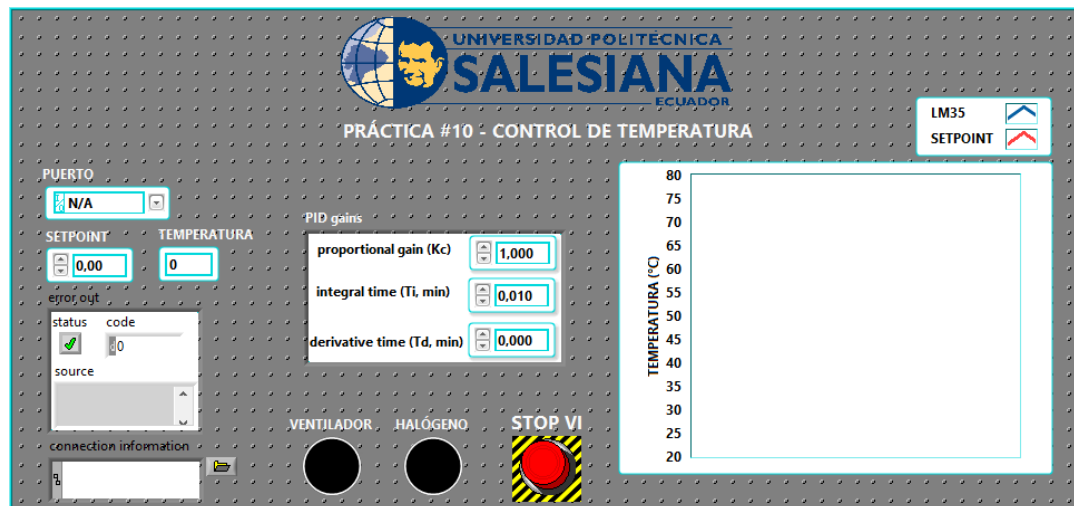


Figura 335: Panel Frontal de la Práctica 10.

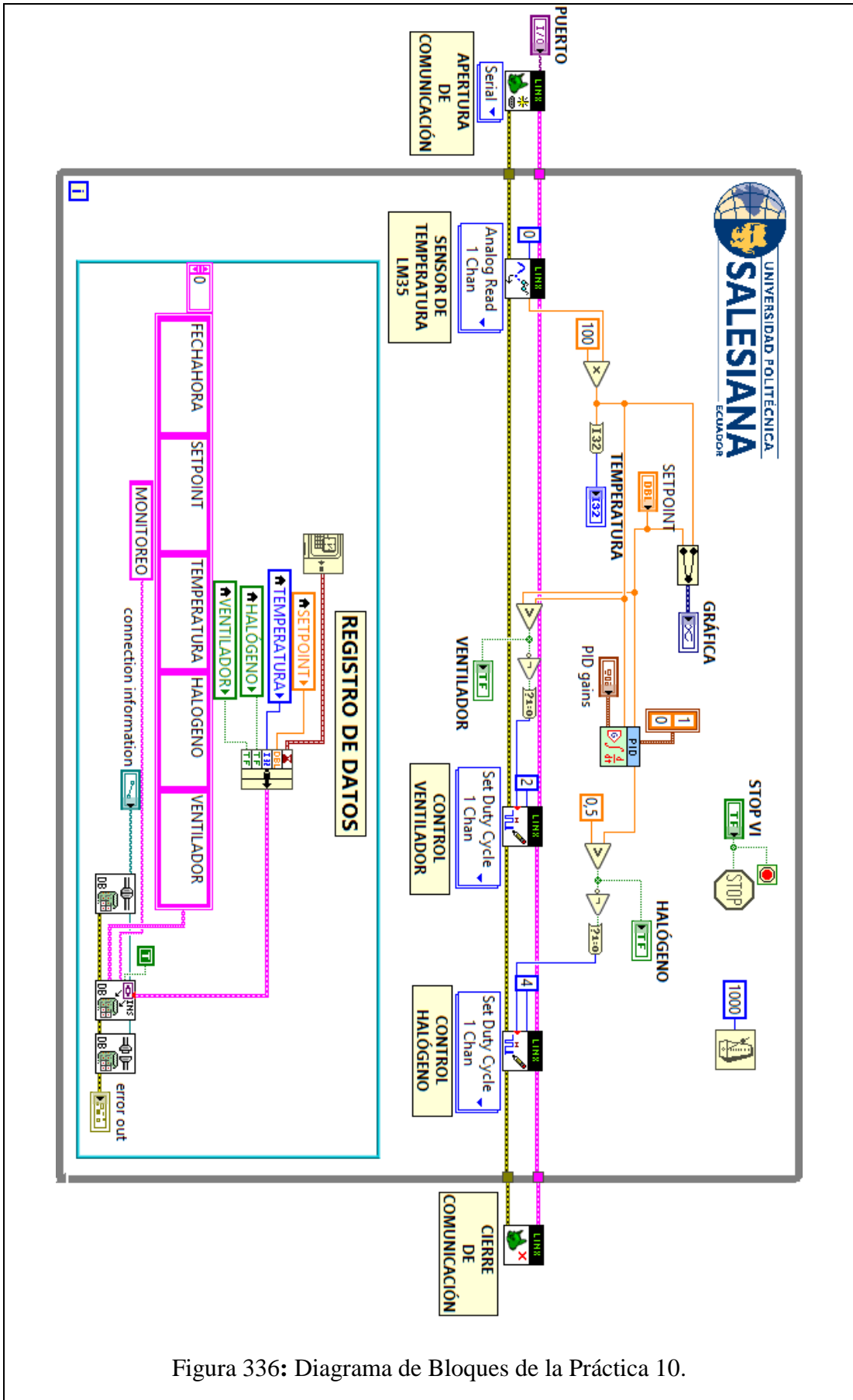


Figura 336: Diagrama de Bloques de la Práctica 10.

CONCLUSIONES:

1. Se logró establecer comunicación entre el sistema embebido Arduino MEGA 2560 y el software de desarrollo LabVIEW 2016.
2. Utilizando el módulo DSC 2016 se creó una base de datos en la cual se almacenaron los datos obtenidos en esta práctica, mostrando así la hora y fecha de los eventos suscitados.
3. Para obtener los valores utilizados en el bloque PID de LabVIEW 2016, fue necesario realizar una medición previa de los datos obtenidos por el sensor LM35.
4. Los datos obtenidos en la medición fueron colocados en el software MatLAB para obtener la función de transferencia y los parámetros PID.

ANÁLISIS DE RESULTADOS:

Cuando la temperatura medida por el Sensor LM35 es menor al Set-Point se produce la activación del halógeno, de esta manera la temperatura se eleva de manera gradual hasta alcanzar el Set-Point.

Cuando la temperatura medida por el Sensor LM35 es mayor al Set-Point se produce la activación del ventilador, de esta manera la temperatura se reduce de manera gradual hasta igualarse con el Set-Point.

Debido al gran consumo de corriente por parte de los elementos como el halógeno y ventilador, fue necesaria la implementación de relés. Estos permiten energizar elementos de gran consumo de corriente utilizando una corriente mínima como la del Arduino MEGA 2560.

Todos los datos de temperatura y eventos de encendido/apagado del ventilador y el halógeno se registran en una base de datos en tiempo real.

RESULTADOS

Para medir el resultado del proyecto técnico se realizó una clase explicativa a los estudiantes de la clase de Sensores y Transductores del periodo 52, a los cuales se explicó el funcionamiento, ventajas y desventajas, se explicó algunas de las funciones integradas en el sistema de adquisición de datos y al final se realizó la siguiente encuesta:

| |
|--|
| 1. ¿Le parece acertado el diseño del módulo de adquisición de datos? |
| Totalmente de acuerdo De acuerdo Poco de acuerdo En desacuerdo |
| 2. ¿Considera muy útil el uso de tarjetas electrónicas para la elaboración de módulos de bajo costo como las que posee el módulo de adquisición de datos? |
| Muy útil Útil Poco útil |
| 3. ¿Es sencillo trabajar una práctica en módulo de adquisición de datos? |
| Muy difícil Difícil Sencillo Muy sencillo |
| 4. ¿Estima necesario aprender mucho más de lo explicado en la exposición acerca de los sensores del módulo de adquisición de datos? |
| Si No |
| 5. ¿Considera satisfactoria la exposición dada? |
| Totalmente de acuerdo De acuerdo Poco de acuerdo En desacuerdo |
| 6. Sugiera parámetros de mejora en el proyecto. ¿Cree necesario realizar modificaciones en el proyecto? |
| |

Luego de realizar las encuestas a los 15 estudiantes asistentes al curso de Sensores y Transductores se obtuvo los siguientes resultados:

➤ ¿Le parece acertado el diseño del módulo de adquisición de datos?

3 pensaron opinaron que estaban totalmente de acuerdo.
12 personas opinaron que estaban de acuerdo.
0 personas opinaron que estaban poco de acuerdo.
0 personas opinaron que estaban en desacuerdo.

➤ ¿Considera muy útil el uso de tarjetas electrónicas para la elaboración de módulos de bajo costo como las que posee el módulo de adquisición de datos?

5 personas opinaron que es Muy útil.
10 personas opinaron que era útil.
0 personas opinaron que era poco útil.

➤ ¿Considera sencillo trabajar una práctica en este módulo de adquisición de datos?

0 pensaron opinaron que era muy difícil.
1 persona opino que era difícil.
11 personas opinaron que era Sencillo.
3 personas opinaron que era Muy sencillo.

➤ ¿Estima necesario aprender mucho más de lo explicado en la exposición acerca de los sensores del módulo de adquisición de datos?

15 pensaron opinaron que SI.
0 personas opinaron que NO.

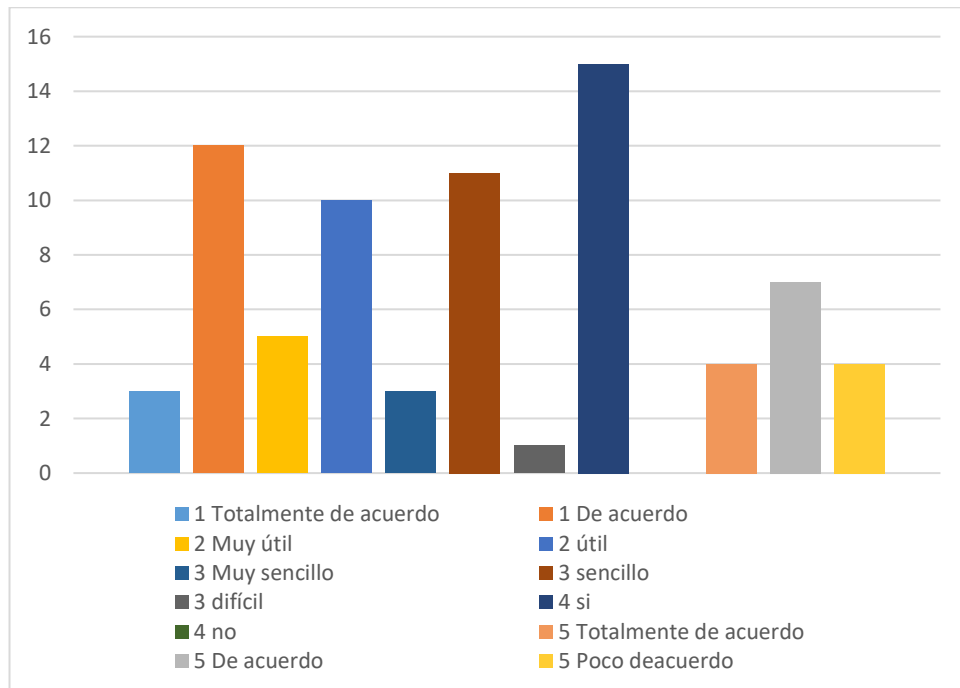
➤ ¿Considera satisfactoria la exposición dada?

4 pensaron opinaron que estaban totalmente de acuerdo.
7 personas opinaron que estaban de acuerdo.
4 persona opino que estaban poco de acuerdo.
0 personas opinaron que estaban en desacuerdo.



Estudiantes del curso de Sensores y Transductores asistentes a la exposición

Gráfico estadístico de la encuesta realizada a los estudiantes del curso de Sensores y Transductores de la Universidad Politécnica Salesiana Sede Guayaquil.



Como resultado adicional se realizó un análisis de costos en el cual se compara dos cotizaciones sobre un sistema SCADA utilizando PLC convencional marca SIEMENS y el valor real del proyecto técnico, donde se determinó que el proyecto técnico es una solución a la implementación de un sistema SCADA de bajo costo.

IMPACTO E INNOVACIÓN

Mediante el diseño e implementación de este módulo didáctico, se busca generar una herramienta de aprendizaje, de manera que los estudiantes de la carrera de Ingeniería Electrónica, pueda desarrollar prácticas de laboratorio, reforzando los conocimientos adquiridos con respecto al ámbito profesional de la carrera.

El hardware Arduino ofrece diferentes ventajas, las más destacables son: bajo costo, open source, infinidad de aplicaciones y la facilidad en cuanto a su programación se refiere. Su compatibilidad con los diferentes sistemas operativos lo convierten en una opción notable al momento de elegir un sistema embebido para desarrollar grandes proyectos.

Dentro del entorno de softwares utilizados para el desarrollo e interacción de los diferentes módulos que componen el proyecto, la herramienta LINX de fuente abierta, desarrollada por Digilent/LabVIEW MakerHub fue de gran ayuda, al ser didáctico, facilita el uso y comprensión de los SubVI's desarrollados para diferentes elementos electrónicos como sensores, módulos de comunicación Bluetooth, etc.

Se busca que la carrera de Ingeniería Electrónica, este a la vanguardia en la tecnología, optimizando costos, al igual que espacio de trabajo, ya que las tarjetas utilizadas en el proyecto implementado, emulan las funciones de un sistema complejo, con un hardware de tamaño reducido que actúa como terminal (computador) abaratando costos y generando espacio dentro del área de trabajo.

Además, los módulos cuentan con entradas/salidas binarias y entradas analógicas que dotan funciones adicionales que pueden ser utilizados en procesamiento de señales y en aplicaciones de diagnóstico avanzadas. Los diferentes diseños y técnicas de conexión permiten soluciones para una gran variedad de aplicaciones, incluso en condiciones ambientales externas.

FUNCIONALIDAD

Este proyecto se diseñó con la finalidad de ser usado por los estudiantes de la carrera de Ingeniería Electrónica como instrumento de aprendizaje, de manera específica en las materias de Sensores y transductores, Instrumentación y Comunicaciones, siendo así, una herramienta de trabajo para que los estudiantes puedan emplear sus conocimientos adquiridos a lo largo de su vida estudiantil.

La variedad de elementos que componen los diferentes módulos y la funcionalidad de los mismos, permiten optimizar tiempo y recursos económicos, ya que emulan el funcionamiento de algunos sistemas utilizados en la industria.

Una de las ventajas de trabajar con el entorno de programación LabVIEW es la configuración eficiente que se obtiene, permitiendo así desarrollar soluciones individuales, valiosas para el aprendizaje y ejecución de procesos complejos.

El poner en práctica la comunicación entre dos terminales a través de la red, ejecutando un patrón de diseño maestro-esclavo, le permite al usuario obtener un diagnóstico continuo del comportamiento que se está ejecutando, ya sea mediante conexión Ethernet o conexión WiFi desde puntos distantes.

CONCLUSIONES

1. La pantalla táctil ubicada en el módulo principal actúa como interfaz HMI ofreciendo al usuario controlar y visualizar los procesos de una manera óptima y en tiempo real.
2. La batería incorporada al módulo principal posee una gran autonomía y permite la ejecución de prácticas que demanden el uso de elementos externos que necesiten una fuente de gran potencia.
3. Gracias a las características y especificaciones de la tarjeta LattePanda, se pueden ejecutar diferentes prácticas de laboratorio de una manera fluida, emulando las funciones de un sistema complejo ya que se trata de un hardware de tamaño reducido que actúa como terminal (computador) abaratando costos y generando espacio dentro del área de trabajo.
4. Se aprovechó al máximo los recursos ofrecidos por el toolkit LINX desarrollado por Digilent/LabVIEW MakerHub, el mismo cuenta con una variedad de subVI's que permiten interactuar con los sensores.
5. NI-VISA de National Instruments proporciona la interfaz de programación entre el hardware y los entornos de desarrollo de aplicaciones como LabVIEW, esta herramienta permite establecer comunicación entre la tarjeta LattePanda y la tarjeta Arduino MEGA 2560.
6. Como hardware de procesamiento de datos adquiridos y entregados a los sensores, se utilizó la tarjeta Arduino MEGA 2560 ya que posee una gran cantidad de entradas/salidas digitales y entradas analógicas.
7. Los sensores utilizados, gracias a su flexibilidad de programación abierta y a las especificaciones que ofrecen, contribuyeron de manera favorable en el diseño e implementación de prácticas didácticas.

8. La programación se realizó en el entorno de desarrollo de aplicaciones LabVIEW 2016, que utiliza un lenguaje de programación flexible y útil al momento de desarrollar aplicaciones de pruebas, control y medidas.
9. Mediante el uso de la herramienta DataSocket, se logró implementar un sistema de comunicación versátil capaz de monitorear de manera remota desde otro terminal ya sea a través de las redes Ethernet o WiFi emulando los sistemas de monitoreo implementados en algunas industrias.
10. Mediante el uso del módulo DSC, se logró generar una base de datos que permite visualizar tendencias de datos históricos del módulo, en tiempo real.
11. La tarjeta LattePanda debido a sus características y especificaciones es capaz de ejecutar softwares y aplicaciones que demanden un procesamiento de información elevado permitiendo realizar el monitoreo y control de los procesos en tiempo real.
12. La guía de prácticas elaborada demuestra de forma detallada y puntual el funcionamiento de cada una de las plantas definidas.

RECOMENDACIONES

1. Debido a la flexibilidad del sistema, se recomienda realizar más prácticas de laboratorio con otra variedad de sensores, permitiendo ampliar la diversidad de equipos con los que puede trabajar el sistema.
2. Gracias a las características y especificaciones de la tarjeta LattePanda, es capaz de ejecutar softwares y aplicaciones que demanden un procesamiento de información elevado, se recomienda investigar el uso de software de programación libre con la tarjeta LattePanda.
3. Gracias a la batería incorporada al módulo principal, es posible la ejecución de prácticas que demanden el uso de elementos externos que necesiten una fuente de gran potencia, ampliando de diversidad de equipos con los que puede trabajar el sistema.
4. Mediante el uso de la herramienta DataSocket, se logró implementar un sistema de comunicación versátil capaz de monitorear de manera remota desde otro terminal, se recomienda realizar prácticas de laboratorio utilizando la red Ethernet de la universidad, para emular los sistemas de monitoreo implementados en algunas industrias.
5. La herramienta NI-VISA de National Instruments proporciona la interfaz de programación entre el hardware y los entornos de desarrollo de aplicaciones como LabVIEW esta herramienta permite establecer comunicación entre la tarjeta LattePanda y la tarjeta Arduino MEGA 2560, se recomienda investigar la capacidad de comunicación de la tarjeta LattePanda con otras tarjetas de adquisición de datos compatibles con el software LabVIEW.
6. La programación se realizó en el entorno de desarrollo de aplicaciones LabVIEW 2016, que utiliza un lenguaje de programación flexible y útil al momento de desarrollar aplicaciones de pruebas, control y medidas. Se recomienda investigar el uso de otros softwares de programación libre que pueda ser utilizado en la tarjeta LattePanda.

7. Se aprovechó al máximo los recursos ofrecidos por el toolkit LINX desarrollado por Digilent/LabVIEW MakerHub, el mismo cuenta con una variedad de subVI's que permiten interactuar con los sensores. Existe gran variedad de toolkits en LabVIEW que permiten comunicarnos con la tarjeta Arduino MEGA 2560, se recomienda la investigación de los toolkits, para variar el lenguaje de comunicación.
8. No conectar a las salidas digitales de la tarjeta Arduino MEGA 2560 elementos cuyo consumo sea mayor a 40 mA ya que este es el valor máximo de corriente que entrega la tarjeta.
9. Utilizar la galleta que se encuentra en el módulo principal en caso de que sea necesario el uso de elementos extras para el desarrollo de una práctica nueva.

REFERENCIAS BIBLIOGRÁFICAS

- Álvarez, J. A. (Septiembre de 2015). *www.asifunciona.com*. Obtenido de http://www.asifunciona.com/electrotecnia/af_motor_cd/af_motor_cd_6.htm
- Arduino. (11 de febrero de 2013). *Arduino*. Obtenido de Arduino: <https://www.arduino.cc/>
- Arduino. (2018). <https://store.arduino.cc/usa/>. Obtenido de <https://store.arduino.cc/usa/arduino-mega-2560-rev3>
- Baccaglioni, F. (28 de mayo de 2009). *tecnogeek*. Obtenido de https://www.tecnogeek.com/verpost.php?id_noticia=817
- Bisson, S. (4 de Julio de 2016). *ZNET*. Obtenido de <http://www.zdnet.com/article/meet-the-lattepanda-a-tiny-windows-10-pc-for-the-internet-of-things/>
- Crespo, J. (22 de marzo de 2015). *www.aprendiendoarduino.com* . Obtenido de <https://aprendiendoarduino.wordpress.com/2015/03/22/que-es-el-hardware-libre/>
- DFRobots. (2016). *DFRobots*. Obtenido de DFRobots: <https://www.dfrobot.com/product-1405.html>
- DFRobots. (2016). *DFRobots*. Obtenido de DFRobots: <https://www.dfrobot.com/product-1413.html>
- DFRobots. (2016). *DFRobots*. Obtenido de DFRobots: <https://www.dfrobot.com/product-1414.html>
- DFRobots. (2016). *DFRobots*. Obtenido de DFRobots: <https://www.dfrobot.com/product-1430.html>
- DFRobots. (2016). *DFRobots*. Obtenido de DFRobots: <https://www.dfrobot.com/product-933.html>
- Doutel, F. (25 de Enero de 2017). *www.xataka.com*. Obtenido de <https://www.xataka.com/especiales/guia-del-arduinomaniaco-todo-lo-que-necesitas-saber-sobre-arduino>
- García, A. (2 de Diciembre de 2016). *www.panamahitek.com*. Obtenido de [www.panamahitek.com: http://panamahitek.com/que-es-y-como-funciona-un-servomotor/](http://panamahitek.com/que-es-y-como-funciona-un-servomotor/)
- García, A. (16 de Enero de 2016). *www.panamahitek.com*. Obtenido de [www.panamahitek.com: http://panamahitek.com/que-es-y-como-funciona-un-potenciometro/](http://panamahitek.com/que-es-y-como-funciona-un-potenciometro/)
- García, Adria. (26 de noviembre de 2013). *diymakers*. Obtenido de <http://diymakers.es/control-velocidad-y-sentido-de-motor-dc/>

- Gottlieb, I. (1994). *Electric Motors & Control Techniques*. McGraw-Hill Education.
- Herrador, R. E. (13 de noviembre de 2009). *Guía de usuario de arduino*. Obtenido de <http://creativecommons.org/licenses/by-nc-sa/3.0/>:
http://electroship.com/documentos/Arduino_user_manual_es.pdf
- LattePanda. (2016). *LattePanda doc*. (DFRobots) Obtenido de LattePanda doc:
<http://docs.lattepanda.com/>
- Martillo, D., & Zambrano, E. (febrero de 2015). *Diseño de aplicaciones de sistemas embebidos basados en tecnología Raspberry y Odroid-U3*. Guayaquil. Obtenido de <http://dspace.ups.edu.ec/handle/123456789/11409>
- National Instruments. (2016). *National Instruments Corporation*. Obtenido de National Instruments Corporation: <http://www.ni.com/labview/applications/esa/>
- National Instruments. (2016). *National Instruments Corporation*. Obtenido de National Instruments Corporation:
<http://sine.ni.com/nips/cds/view/p/lang/es/nid/212478>
- National Instruments. (2016). *National Instruments Corporation*. (N. VISA, Productor) Obtenido de National Instruments Corporation:
<http://sine.ni.com/nips/cds/view/p/lang/es/nid/12145>
- National Instruments. (14 de Enero de 2016). *National Instruments Corporation*. Obtenido de National Instruments Corporation: <http://www.ni.com/white-paper/3223/en/>
- National Instruments. (2016). *National Instruments Corporation*. Obtenido de National Instruments Corporation: <http://www.ni.com/pdf/manuals/375310b.html>
- Orrellana, B. (2015). *Diseño e implementación de un sistema de control de dispositivos Maestro-Esclavo basados en la red industrial AS-I para el laboratorio de automatización industrial*. Guayaquil.
- Oyanedel, J. P. (17 de febrero de 2014). *www.lgblog.cl*. Obtenido de <http://www.lgblog.cl/reviews/que-significa-que-una-pantalla-sea-tipo-ips-por-que-es-mejor/>
- Perez, F. (6 de marzo de 2013). *nunoarcade.blogspot.com*. Obtenido de <https://nunoarcade.blogspot.com/2013/03/conexion-de-un-microswtich-arcade-como.html>
- Stanton, A. (12 de abril de 2017). *digilentinc*. Obtenido de <https://blog.digilentinc.com/how-linux-works/>
- Sunfounder. (2012). *Sunfounder*. Obtenido de <https://www.sunfounder.com/about-magento-demo-store/>

Sunfounder. (2013). *Sunfounder*. Obtenido de <https://www.sunfounder.com/starterkit/arduino/sensor-kit-v2-0/37-modules-sensor-kit-v2-0-for-arduino.html>




Velasco, R. (1 de Enero de 2016). *www.redeszone.net*. Obtenido de <https://www.redeszone.net/2016/01/01/lattepanda-un-mini-ordenador-con-windows-10-por-60-euros/>

ANEXOS

PRESUPUESTO DEL PROYECTO TÉCNICO.

| ITEM | DESCRIPCIÓN | PRECIO |
|--------------|--|-------------------|
| 1 | LattePanda (2G/32GB) | \$119,00 |
| 2 | LattePanda Starter Kit (American Power Adapter) | \$29,90 |
| 3 | 7-inch 1024x600 IPS Display for LattePanda | \$34,00 |
| 4 | 7-inch Capacitive Touch Panel Overlay for LattePanda Display | \$19,00 |
| 5 | Arduino Mega2560 Rev3 | \$56,00 |
| 6 | Costos de importación | \$150,00 |
| 7 | Kit de sensores 37 en 1 | \$50,00 |
| 8 | Cables | \$10,00 |
| 9 | Impresiones de bases 3D | \$100,00 |
| 10 | Servo-motor | \$12,00 |
| 11 | Motores DC | \$15,00 |
| 12 | Foco halógeno | \$3,00 |
| 13 | Batería de 12V | \$30,00 |
| 14 | Estructura de acrílico | \$100,00 |
| 15 | Mano de obra | \$600,00 |
| TOTAL | | \$1.327,90 |

COTIZACIÓN DE SISTEMA SCADA UTILIZANDO PLC CONVENCIONAL MARCA SIEMENS.

|  <small>Automatizando el país del futuro</small> | <small>Departamento de Proyectos e Innovaciones</small> |  Distribuidor Autorizado | | | | |
|---|---|---|--|-------|----------------------------|--|
| <small>Solution Partner</small>  <small>Autorizado</small> | CONTRIBUYENTES ESPECIALES RESOLUCIÓN No. www.inasecuador.com 826 22/12/2009 | <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 50%;"><small>FECHA:</small></td> <td style="width: 50%;"></td> </tr> <tr> <td><small>OFERTA No.:</small></td> <td></td> </tr> </table> | <small>FECHA:</small> | | <small>OFERTA No.:</small> | |
| <small>FECHA:</small> | | | | | | |
| <small>OFERTA No.:</small> | | | | | | |
| Jorge Juan N32-24 y Av. Meriona de Jesús Teléfonos: 2504423 / 2905464 / 2585487 Fax: 2565488 Quito -Ecuador | | | | | | |
| COTIZACIÓN | | | | | | |
| <small>Cliente:</small> Juan Palomeque | | | | | | |
| <small>Atención:</small> | <small>RUC:</small> | | | | | |
| <small>Dirección:</small> | <small>e-mail:</small> | | | | | |
| A continuación nos es grato presentarle nuestra mejor propuesta para el suministro de los siguientes materiales y servicios | | | | | | |
| ITEM | CANT | DESCRIPCIÓN | PRECIO | DESC. | TOTAL | |
| 1 | 1 | Starter Kit Simatic S7-1200 + Simatic HMI KTP400. Incluye una CPU1214C AC/DC/Relé, Panel Simatic KTP400 monocromático, Software TIA PORTAL Basic, simulador de entradas digitales, cable de comunicación Industrial Ethernet y CD con documentación. (15 a 20 días) | 1475,00 | 20% | 1180,00 | |
| 2 | 1 | Starter Kit Simatic S7-1200 + Simatic HMI KTP600 Color. Incluye una CPU1214C AC/DC/Relé, Panel Simatic KTP600 color, software TIA PORTAL Basic, simulador de entradas digitales, cable de comunicación Industrial Ethernet y CD con documentación. (15 a 20 días) | 2215,00 | 20% | 1772,00 | |
| 3 | 1 | FUENTE PARA LOGO POWER 24VDC /2.5A 6EP1332-1SH43 | 107,00 | 20% | 85,80 | |
| 5 | 1 | STARTER KIT S7-1200 CPU 1214 + HMI 4" KTP BASIC + SOFTWARE INMEDIATO | 1677,00 | 20% | 1341,60 | |
| 6 | 6 | RELE DE EST. S. 3-32 VDC INPUT 40-440 VAC OUTPUT 60A | 62,40 | 20% | 395,52 | |
| 7 | 3 | RELE DE ESTADO SOLIDO 60A 3-32VDC INPUT 40-440VAC OUTPUT | 36,00 | 20% | 86,40 | |
| 8 | 6 | RIEL DIN (1MTR) | 2,50 | 0% | 15,00 | |
| TIEMPO DE ENTREGA: | | | Item 1 y 2, segunda semana mayo. Resto inmediato | | SUBTOTAL 4876,12 | |
| | | | | | IVA 12% 585,13 | |
| FORMA DE PAGO: | | | CONTADO | | TOTAL 5461,25 | |
| VIGENCIA DE LA OFERTA: | | | 8 DIAS | | | |
| NOTAS | | | Realizado por: | | | |
| 1 El tiempo de entrega ofertado se hace efectivo a partir de la fecha de recibida la orden de compra. 2 Descuento para Tesi aplica presentando documento que valide uso de los items en la misma | | | Ing. Lenin Falconi Estrada Automatización de Procesos proyectos@inasecuador.com Cel.: 0999752149 | | | |

TABLEROS ELÉCTRICOS
Distribución y Control - Bancos de Condensadores
Transferencia Automática - Transferencia Manual



DIRECCIÓN: Urb. Ciudad de Quito Lote#41 (Conocoto)
Telefax: +593 (02)2145 420 / +593 (02)2075 195
E-mail: electrotecnica.comercial@gmail.com



PROFORMA
N° 000215



| | | | |
|-----------------|--------------------------------------|----------------|---------|
| CLIENTE: | Juan Palomeque | FECHA: | |
| DIRECCIÓN: | | TELÉFONO: | |
| SOLICITADO POR: | | CEL: | |
| ELABORADO POR: | Ing. Rubén Guayasamín (09 9867 3380) | FORMA DE PAGO: | CONTADO |



| ITEM | CANT. | DESCRIPCIÓN | P. UNITARIO | DESC. % | V. TOTAL |
|------|-------|---|-------------|---------|----------|
| 1 | 1 | CPU 1214C ACDC/RELE 57-1200 | 660,00 | 5% | 627,00 |
| | 1 | DIMATIC Basic Panel KTP400 PN | 547,00 | 5% | 519,65 |
| | 1 | FUENTE DE PODER LOGO 2.5A 120-230V- 24VDC | 107,00 | 5% | 101,65 |
| | 6 | CONTACTORES 3HP SIRIUS 3RT2023-1AN20 | 26,42 | 5% | 150,59 |
| | 12 | BREAKER 1P 6AMP. | 10,00 | 5% | 114,00 |
| | 1 | PULSANTE TIPO HONGO CON RETENCION SIEMENS | 15,10 | 5% | 14,35 |
| | 1 | SELECTOR DOS POSICIONES SIEMENS | 11,64 | 5% | 11,06 |
| | 20 | BORNERAS PARA CABLE 6AWG SIEMENS | 2,65 | 5% | 50,35 |
| | 4 | RIEL DIN | 5,00 | 5% | 19,00 |
| | 1 | Switch Industrial Ethernet PARA PLC 57-1200 | 216,00 | 5% | 205,20 |

GARANTIA:

La garantía es de un año calendario por fallas de fabricación, el cual empieza a partir de la fecha de entrega de los equipos y no incluye fallas por mal manejo, sobretensiones continuas y partes y piezas de desgaste normal. Quedamos a sus órdenes para cualquier información que soliciten en relación a la presente oferta.

GUAYASAMIN
FIRMA ELECTROTECNIA

| | |
|-----------|---------|
| SUBTOTAL | 1812,85 |
| I.V.A 12% | 217,54 |
| TOTAL \$ | 2030,39 |

CRONOGRAMA DE ACTIVIDADES

| ACTIVIDAD | mar-17 | abr-17 | may-17 | jun-17 | jul-17 | ago-17 | sept-17 | oct-17 | nov-17 | dic-17 | ene-18 | feb-18 | mar-18 | abr-18 | may-18 | jun-18 | jul-18 | ago-18 | |
|---|--------|--------|--------|--------|--------|--------|---------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--|
| 1. Investigación del tema | | | | | | | | | | | | | | | | | | | |
| 2. Diseño de la planta | | | | | | | | | | | | | | | | | | | |
| 3. Cotización de equipos | | | | | | | | | | | | | | | | | | | |
| 4. Compra de equipos | | | | | | | | | | | | | | | | | | | |
| 5. Prueba de comunicación | | | | | | | | | | | | | | | | | | | |
| 6. Diseño del sistema SCADA en LabVIEW | | | | | | | | | | | | | | | | | | | |
| 7. Comprobar la adquisición de datos | | | | | | | | | | | | | | | | | | | |
| 8. Elaboración de practicas | | | | | | | | | | | | | | | | | | | |
| 9. Etapa de pruebas | | | | | | | | | | | | | | | | | | | |
| 10. Prueba de control y visualización de las prácticas en la interfaz HMI | | | | | | | | | | | | | | | | | | | |
| 11. Escritura de la tesis | | | | | | | | | | | | | | | | | | | |
| 12. Revisión técnica | | | | | | | | | | | | | | | | | | | |
| 13. Revisión del libro | | | | | | | | | | | | | | | | | | | |