



UNIVERSIDAD MICHOACANA  
DE SAN NICOLÁS DE HIDALGO

FACULTAD DE INGENIERÍA ELÉCTRICA

DISEÑO DE UNA INTERFAZ GRÁFICA USANDO LAS  
PLATAFORMAS LAB-VIEW Y ARDUINO APLICADO A LA  
DOMÓTICA.

TESIS  
QUE PRESENTA  
JUAN SOTO CARRILLO

Para obtener el título de

INGENIERO ELECTRÓNICO  
DIRECTOR DE TESIS  
DR. JUAN ANZUREZ MARÍN



Morelia Michoacán, México, abril 2018.



# **Universidad Michoacana de San Nicolás de Hidalgo.**

## **Facultad de Ingeniería Eléctrica.**

**Diseño de una interfaz gráfica usando las plataformas LabVIEW y arduino aplicado a la domótica.**

### **Tesis**

Que presenta

**Juan Soto Carrillo.**

---

Para obtener título de

**Ingeniero Electrónico**

Director de tesis.

**Dr. Juan Anzures Marín.**

---

Morelia, Michoacán, México, abril 2018.

## Dedicatoria

Este trabajo lo dedico antes que nada a Dios y la virgen de San Juan de los Lagos por haberme dado salud, sabiduría y por toda la fortaleza espiritual que me han brindado.

También se lo dedico a toda mi familia, principalmente a mis padres **Juan Soto Bello**, que a pesar de todo de tantos problemas y de tanto esfuerzo debe sentirte muy orgulloso de mi y de mi hermana que finalmente lo logramos, te amo papa y a mi madre **Ernestina Carrillo Alejo** que gracias a todos sus esfuerzos, consejos y dedicación, nos dio estudio a todos mis hermanos y hasta ahora lo ha seguido haciendo conmigo y mi hermana, gracias madre por todo lo que me has dado en pocas palabras nos diste tu vida entera, te amo mamá.

Además, a todos mis hermanos; A mi hermano **Ernesto** por su apoyo económico y quien por ser el mayor tomo una gran responsabilidad y nos dio siempre sus consejos él fue quien nos guiaba, a mi hermana **Lidia** quien más que una hermana fue como una segunda madre para mí, y gracias por tu apoyo hermana. Y especialmente lo dedico a mi hermano **Gustavo** con quien compartí gran parte de nuestra preparación básica escolar y gracias por el gran apoyo económico y por siempre estar pendiente de mí y de mi hermana ya que sin tu apoyo y enseñanzas nunca hubiéramos logrado concluir nuestra carrera. Igualmente lo dedico a mi hermana **María Elena** quien siempre me acompañó durante nuestra infancia y preparación escolar , a mi hermanita **Cecilia** que aunque ella no lo crea la quiero mucho y gracias hermana por trabajar tan duro siempre a lado de mi madre y sacrificar tus estudios para ayudarnos, a mi hermana pequeña **Ing. Laura** con quien estudiamos gran parte de la carrera juntos y tantas cosas que aprendimos juntos a pesar de todas las diferencias que hemos tenido, mi mayor logro es el haberte motivado a continuar para que llegaras a ser ingeniera felicidades hermana y no olvides que te quiero mucho hermanita.

Y finalmente a mi compañera de vida **Yolanda** que prácticamente desde el comienzo de esta larga carrera universitaria me acompañó incondicionalmente y me ha dado el mejor regalo del mundo a mi hijita Vanessa gracias mi amor te amo tanto, y a ti hijita **Vanessa** gracias por llegar a mi vida para darle un nuevo giro y ser el mejor motivo para seguir adelante cada día, llegar a casa y encontrarlas esperándome es lo mejor que puede existir en la vida.

Decir un número de gracias nunca será suficiente por eso les digo infinitamente gracias a todos por su paciencia y esfuerzos para que pudiera ser Ingeniero los amo mucho Familia.

# Agradecimientos

Agradezco a ti señor dios por la fortaleza sabiduría que me has dado y que gracias a la fe que te tengo me has ayudado paso a paso a lograr esta meta y me has dejado demostrarme a mí mismo de lo que puedo ser capaz de lograr, gracias a ti virgen de Guadalupe por interceder ante mi dios por mis plegarias.

Premio de todo corazón a mi madre **Ernestina Carrillo Alejo** por todo el esfuerzo y por haber dejado prácticamente su vida para que mis hermanos y yo tuviéramos la oportunidad de estudiar y mi hermana menor y yo lográramos concluir una carrera.

De igual manera le agradezco especialmente a mi hermano **Gustavo** que siempre estuvo al pendiente de nosotros a cada momento sin importar salud o situaciones adversas en su vida siempre estaba ahí para lo que yo y mi hermana Laura necesitáramos, gracias hermano nunca terminare de agradecerte.

Honro a mi asesor de tesis al **Dr. Juan Anzures Marín** por todo el conocimiento transmitido hacia mi persona para que este trabajo de investigación haya sido un éxito, por el tiempo que dedico a atender mis dudas a pesar de tantas responsabilidades que tiene en su profesión, gracias profesor por todo lo que me ha apoyado este logro sin su apoyo jamás se hubiese realizado.

Galardono a todos los **maestros** que me impartieron clases, por su dedicación y esfuerzo que hicieron, para lograr trasmitirme de alguna manera sus conocimientos y más que nada por lograr ser un guía en el desarrollo de mi carrera.

Y a todos mis **familiares, amigos y compañeros de escuela**, que estuvieron siempre ahí para lo que ocupara.

También reconozco a **toda la población en general**, ya que, con el pago de sus impuestos, logran mantener en pie esta universidad pública, mediante los diferentes entes de gobierno. Y finalmente agradezco a la **Universidad Michoacana de San Nicolás de Hidalgo** a todos sus empleados directivos, maestros trabajadores en general que, gracias a sus esfuerzos, es lo que es esta gran universidad.

# Índice

|  |           |
|--|-----------|
| <b>Dedicatoria.....</b>  | <b>0</b>  |
| <b>Agradecimientos .....</b>   | <b>1</b>  |
| <b>Índice.....</b>   | <b>2</b>  |
| <b>Resumen .....</b>   | <b>5</b>  |
| <b>Palabras clave .....</b>  | <b>5</b>  |
| <b>Abstract .....</b>  | <b>6</b>  |
| <b>Keyboard.....</b>   | <b>6</b>  |
| <b>Lista de figuras.....</b>   | <b>7</b>  |
| <b>Lista de Tablas .....</b>   | <b>9</b>  |
| <b>Glosario de términos (Lista de Símbolos y Abreviaciones).....</b> | <b>10</b> |
| <b>Capítulo 1 Introducción. ....</b>                                 | <b>11</b> |
| 1.1 Antecedentes, Descripción General del Problema .....             | 11        |
| 1.1.1 Antecedentes (Estado del arte).....                            | 11        |
| 1.1.2 Descripción general del problema.....                          | 11        |
| 1.2. Objetivos de la Tesis .....                                     | 12        |
| 1.2 1 Objetivo general.....  | 12        |
| 1.2 2 Objetivos particulares.....                                    | 12        |
| 1.3. Justificación .....   | 12        |
| 1.4 Metodología.....   | 12        |
| 1.5 Contenido de la tesis .....                                      | 13        |
| 1.6 Descripción de capítulos.....                                    | 13        |
| <b>Capítulo 2 Conceptos fundamentales .....</b>                      | <b>15</b> |
| 2.1 Introducción.....  | 15        |
| 2.1.1 Instrumentación electrónica y virtual .....                    | 15        |
| 2.1.2 ¿Qué es un edificio inteligente? .....                         | 17        |
| 2.2 Domótica e Inmótica.....   | 17        |
| 2.2.1 ¿Que es domótica?.....   | 17        |
| 2.2.2 ¿Que es Inmótica?.....   | 18        |
| 2.3 Lenguaje Gráfico de Programación.....                            | 18        |
| 2.3.1 Antecedentes .....   | 18        |
| 2.3.2 Introducción .....   | 19        |
| 2.3.3 ¿Qué es LabVIEW? .....   | 20        |

|  |           |
|--|-----------|
| 2.3.4 Entorno .....  | 21        |
| 2.3.5 Ejemplo básico de diseño de un Vi .....  | 24        |
| 2.3.6 Aplicaciones de LabVIEW .....  | 27        |
| 2.4 Plataforma Electrónica Arduino (microcontroladores) .....  | 27        |
| 2.4.1 Antecedentes .....   | 27        |
| 2.4.2 Introducción .....   | 28        |
| 2.4.3 ¿Qué es Arduino? .....   | 29        |
| 2.4.4 Software y Hardware del sistema Arduino .....  | 30        |
| 2.4.5 Sistemas embebidos.....  | 30        |
| 2.4.6. Arquitectura abierta del sistema Arduino .....  | 33        |
| 2.5 Interfaz gráfica de LabVIEW con microcontrolador arduino (LIFA).....   | 35        |
| 2.5.1. Código a cargar a Arduino .....   | 35        |
| <b>Capítulo 3. Caso de estudio .....</b>   | <b>37</b> |
| Introducción.....  | 37        |
| 3.1 Tarjeta para adquisición de datos.....   | 37        |
| 3.1.1 Esquema y disposición de pines.....  | 37        |
| 3.1.2 Especificaciones generales de la tarjeta.....  | 40        |
| 3.2. Alcance y delimitación de la tesis.....   | 40        |
| 3.3. Desarrollo de los planos en AutoCAD de la casa automatizada.....  | 40        |
| 3.3.1 Desarrollo de la maqueta basada en los planos en AutoCAD de la casa automatizada.....                                      | 42        |
| 3.4 Instalación de las Plataformas y paquetes.....   | 42        |
| 3.4.1. Instalación de LabVIEW.....   | 42        |
| 3.4.2 Instalación de paquete NI VISA.....  | 46        |
| 3.4.3 Instalación de paquete <i>VI Package Manager</i> .....   | 53        |
| 3.4.4 Instalacion de la interfaz gráfica de LabVIEW con microcontrolador arduino (LIFA).....                                     | 57        |
| 3.4.5 Instalacion del ambiente de programación IDE de Arduino.....   | 63        |
| 3.4.6 Instalación de paquete de módulos de Desarrollo Visión.....  | 66        |
| 3.4.7. Instalación del Software de adquisición (video) Visión.....   | 70        |
| 3.5 Desarrolló de interfaz gráfica en LabVIEW .....  | 72        |
| 3.5.1 Carga del LIFA-Base a la placa arduino Mega 2560 .....   | 73        |
| 3.5.2 Desarrollo y Programación de la Interfaz Gráfica en LabVIEW .....  | 75        |
| <b>Capítulo 4. Comunicación Inalámbrica y Resultados: Uso y funcionamiento de la interfaz gráfica aplicada a la maqueta.....</b> | <b>87</b> |
| 4.1 Introducción .....   | 87        |
| 4.2 Resultados de las interfaces graficas .....  | 87        |

|   |            |
|---|------------|
| 4.2.1 Los resultados de la automatización de apertura y cierre de <b>una puerta y una ventana</b> .....   | 87         |
| 4.2.2 Resultados y funcionamiento de la automatización de encendido y apagado de <b>una lámpara</b> .....   | 87         |
| 4.2.3 Resultados y funcionamiento de la automatización de encendido y apagado de <b>la alarma inteligente</b> .....   | 88         |
| 4.2.4 Resultados de la automatización de la <b>Interfaz Completa</b> para la casa domótica (1 Alarma inteligente, 5 Puertas, 4 Ventanas, 5 Luces internas y 1 externa)..... | 90         |
| 4.3 Comunicación inalámbrica mediante internet, en una computadora o cualquier dispositivo. ....  | 95         |
| 4.4 Propuestas de actuadores (motores) y sensores en aplicación real. ....  | 99         |
| <b>Capítulo 5. Conclusiones y Trabajos Futuros .....</b>  | <b>102</b> |
| 5.1 Conclusiones .....  | 102        |
| 5.1.1 Conclusiones Generales .....  | 102        |
| 5.1.2 Conclusiones Particulares .....   | 102        |
| 5.2. Trabajos Futuros .....   | 102        |
| 5.2.1 Control remoto. ....  | 102        |
| <b>Bibliografía .....</b>   | <b>104</b> |
| <b>Apéndices. Conexiones físicas en la maqueta desarrollada.....</b>  | <b>106</b> |
| Apéndice A. ....  | 106        |
| Apéndice B. ....  | 107        |

## Resumen

Mediante la herramienta y entorno de programación gráfica LabVIEW, se realizó una interfaz gráfica, para un control domótico de una casa habitación que consta de dos recamaras, un baño, una área para sala, una área para cocina y un garaje, en la cual se controlaron cinco puertas, cuatro ventanas, un sistema de alarma, luces internas y externas, todo lo anterior integrado en una interface gráfica de usuario, la cual se diseñó por medio de las dos plataformas de desarrollo una en LabVIEW y la otra en un microcontrolador arduino, mediante la interface que LabVIEW tiene para comunicarse con Arduino, la cual se llama **LabVIEW interface para arduino** con el lenguaje de programación gráfico de LabVIEW, posteriormente se llevó acabo la comunicación vía remota mediante otra computadora conectada a la red mediante la **Publicación del Instrumento virtual en la red**, finalmente se comprobó de manera física que el sistema completo funciona de una manera adecuada logrando los objetivos planteados.

## Palabras clave

Domótica; interface gráfica; LabVIEW; Arduino; automatización; casa inteligente; instrumento virtual.

## Abstract

*Using the tool and analysis environment, LabVIEW, a graphical interface was made for a home automation control of a house that consists of two bedrooms, a bathroom, a living room, an area for kitchen and a garage, in which five doors were controlled, four windows, an alarm system, internal and external lights, all the previous integrated in a graphical user interface, that was designed through the two development platforms, one in LabVIEW and the other in an arduino microcontroller, through the interface that LabVIEW has to communicate with Arduino, which is called LabVIEW interface for Arduino with the graphic programming language LabVIEW, later on, communication was carried out through remote observation through another computer connected to the network through the publication of the Virtual Instrument in the network, finally the physical form was verified that the complete system works in an adequate way achieving the proposed objectives.*

## Keyboard

*Home automation; graphic interface; LabVIEW; Arduino; automation; smart House; virtual instrument.*

## Lista de figuras

|  |    |
|--|----|
| Figura 1 Ejemplo de un dispositivo con instrumentación electrónica. ....   | 15 |
| Figura 2 Osciloscopio diseñado con Instrumentación virtual. ....   | 16 |
| Figura 3 Logo LabVIEW.....   | 21 |
| Figura 4 Panel Frontal (Izquierdo) Diagrama de bloques (derecho) .....   | 22 |
| Figura 5 Barra de herramientas del panel frontal.....  | 22 |
| Figura 6 Ventana de ayuda.....   | 23 |
| Figura 7 Paleta de controles (superior) y Paleta de funciones (inferior).....  | 24 |
| Figura 8 De izquierda a derecha: David Mellis, David Cuartielles, Gianluca Martino,<br>Massimo Banzi, Tom Igoe.....                | 28 |
| Figura 9 Logo Arduino.....   | 29 |
| Figura 10 Diagrama a bloques del sistema Arduino básico (AT mega 328) .....  | 32 |
| Figura 11 IDE ambiente de programación de arduino.....   | 32 |
| Figura 12 Tablero (Tarjeta) Electrónica Arduino Leonardo .....   | 33 |
| Figura 13 Pasos secuenciales para ejecutar un programa en arduino .....  | 34 |
| Figura 14 Juego de herramientas, interfaz de NI LabVIEW para arduino .....   | 35 |
| Figura 15 Esquema de la tarjeta Arduino MEGA .....   | 38 |
| Figura 16 Disposición de pines Arduino Mega .....  | 39 |
| Figura 17 Plano 2D de la casa automatizada. ....   | 41 |
| Figura 18 Plano 3D de la casa Automatizada .....   | 41 |
| Figura 19 Maqueta para implementar la automatización. ....   | 42 |
| Figura 20 Diagrama de bloques de proceso de control.....   | 72 |
| Figura 21 Diagrama de bloques de la interfaz gráfica.....  | 73 |
| Figura 22 Programa subido a la placa Arduino Mega 2560 .....   | 74 |
| Figura 23 Configuración de placa Arduino Mega 2560 en puerto Com 4.....  | 75 |
| Figura 24 Creación del proyecto .....  | 75 |
| Figura 25 Almacenado del proyecto y Creación del VI .....  | 76 |
| Figura 26 Almacenado del VI.....   | 76 |
| Figura 27 VI para programar la Interfaz Grafica.....   | 77 |
| Figura 28 Programación para control de 2 servos 1 Puerta y 1 Ventana (Panel frontal<br>y Diagrama de bloques respectivamente)..... | 78 |
| Figura 29 Programación para control de 1 lámpara con sensor, (Panel frontal y<br>Diagrama de bloques respectivamente).....         | 78 |
| Figura 30 Panel frontal de la alarma inteligente .....   | 79 |
| Figura 31 Diagrama de bloques de la alarma inteligente .....   | 79 |
| Figura 32 Interfaz gráfica de usuario desarrollada para el control domótico de las<br>puertas de la casa (Panel Frontal). ....     | 80 |

|  |     |
|--|-----|
| Figura 33 Programación para la Interfaz gráfica de usuario desarrollada para el control domótico de las puertas de la casa (Diagrama de Bloques) “Superior casos en verdadero, inferior casos en falso” .                            | 81  |
| Figura 34 Programación para la interfaz gráfica de usuario desarrollada para el control domótico de las ventanas de la casa (Diagrama de Bloques) “Superior casos en verdadero, inferior casos en falso” .                           | 82  |
| Figura 35 Interfaz gráfica de usuario desarrollada para el control domótico de las ventanas de la casa (Panel Frontal).  | 83  |
| Figura 36 Interfaz gráfica de usuario desarrollada para el control domótico de las luces internas de la casa (Panel Frontal).  | 83  |
| Figura 37 Programación para la interfaz gráfica de usuario desarrollada para el control domótico de las luces internas de la casa (Diagrama de Bloques) “Superior casos en verdadero, inferior casos en falso” .                     | 84  |
| Figura 38 Programación para la Interfaz gráfica de usuario desarrollada para el control domótico de la alarma inteligente y luces externas de la casa (Diagrama de Bloques) “Superior casos en verdadero, inferior casos en falso” . | 85  |
| Figura 39 Interfaz gráfica de usuario desarrollada para el control domótico de la alarma inteligente y luces externas la casa (Panel Frontal).   | 86  |
| Figura 40 Comprobación del funcionamiento de la Automatización de 1 puerta y 1 ventana .   | 88  |
| Figura 41 Comprobación del funcionamiento de la Automatización de 1 lámpara desde el pc y con el sensor de movimiento .  | 89  |
| Figura 42 Funcionamiento de la Alarma Inteligente (secuencia de Izq. Derecha).   | 89  |
| Figura 43 Imágenes del funcionamiento de las puertas.  | 91  |
| Figura 44 Imágenes del funcionamiento de las ventanas.   | 91  |
| Figura 45 Imágenes del funcionamiento de las luces interiores.   | 92  |
| Figura 46 Imágenes de la alarma en funcionamiento.   | 93  |
| Figura 47 Imágenes tomadas y almacenadas por la cámara cuando detecta movimiento.  | 94  |
| Figura 48 Herramienta Web Publishing Tool.   | 95  |
| Figura 49 Paso 1 selección del VI y Modo de Publicación.   | 96  |
| Figura 50 Personalización de la página web.  | 96  |
| Figura 51 Guardado del fichero.  | 97  |
| Figura 52 Dirección de la página web (URL).  | 97  |
| Figura 53. Comunicación vía remota por medio de la página web desde otro ordenador.  | 98  |
| Figura 54 Activación de descarga de controles Active X en el navegador Internet Explorer.  | 99  |
| Figura 55 Especificaciones de los servomotores SGMJV SGLGW.  | 100 |
| Figura 56 Sensor siemens PDM-IXx12/T y PDM-IXx18/T .   | 101 |
| Figura 57 Logo entorno de desarrollo de Android Studio.  | 103 |

## Lista de Tablas

|  |     |
|--|-----|
| Tabla 1 Comparación de instrumento virtual frente a instrumento electrónico..... | 17  |
| Tabla 2 Evolución de las versiones de LabVIEW .....                              | 19  |
| Tabla 3 Atajos mediante el teclado. ....   | 25  |
| Tabla 4 Especificaciones de la tarjeta Electrónica Arduino Mega .....            | 40  |
| Tabla 5 Conexiones físicas en el bus de datos del Baño.....                      | 106 |
| Tabla 6 Conexiones físicas en el bus de datos del garaje. ....                   | 107 |

## Glosario de términos (Lista de Símbolos y Abreviaciones).

|                  |   |
|------------------|---|
| Delinquir        | Cometer un delito, (se puede delinquir por ignorancia a la norma)   |
| Domótica         | Conjunto de técnicas orientadas a automatizar una vivienda, que integran la tecnología en los sistemas de seguridad, gestión energética, bienestar o comunicaciones.              |
| Edificaciones    | son obras que diseña, planifica y ejecuta el ser humano en diferentes espacios, tamaños y formas, en la mayoría de los casos para habitarlas o usarlas como espacios de resguardo |
| Empotrar         | Incrustar una cosa en otra, especialmente al hacerlas chocar con violencia entre sí   |
| GPIB             | Bus de Interfaz de Uso General  |
| Hito             | Acontecimiento puntual y significativo que marca un momento de desarrollo importante de un proceso.   |
| IDE              | Entorno de desarrollo interactivo por sus siglas en inglés <i>Integrated Development Environment</i>  |
| Interfaz         | Conexión, física o lógica, entre una computadora y el usuario, un dispositivo periférico o un enlace de comunicaciones.   |
| Interfaz gráfica | Medio con que el usuario puede comunicarse con una máquina, equipo, computadora o dispositivo, y comprende todos los puntos de contacto entre el usuario y el equipo.             |
| Intrusarse       | Apropiarse, sin razón ni derecho, de un cargo, una autoridad, una jurisdicción, cosas, etc.   |
| Intrusión        | Acción y efecto de intrusarse   |
| Plataforma       | En informática, una plataforma es un sistema que funciona como una base para hacer funcionar determinados módulos de hardware y/o de software con los que es compatible.          |
| Sensor PIR       | Sensores infrarrojos pasivos (PIR) son dispositivos para la detección de movimiento.  |
| Sketch           | Nombre técnico específico de Arduino para representar los programas desarrollados en el ambiente de programación.   |
| Sub VI           | Instrumento Virtual contenido dentro de otro Instrumento Virtual.   |
| VI               | Instrumento Virtual   |

# Capítulo 1 Introducción.

## 1.1 Antecedentes, Descripción General del Problema

### 1.1.1 Antecedentes (Estado del arte).

En [1] los autores utilizaron las plataformas LabVIEW y arduino para diseñar e implementar un sistema de transferencia de información a través de la Red eléctrica de Baja tensión con ayuda de la tecnología *Power Line Communications* (PLC). Este sistema fue diseñado e implementado en el laboratorio de Telecomunicaciones de la Universidad Politécnica Salesiana, sede Guayaquil, Ecuador.

En [2] se utilizan las plataformas LabVIEW y arduino para crear un generador virtual de funciones predefinidas y arbitrarias, (utilizan una técnica llamada Síntesis Digital Directa DDS), en el desarrollo se comparan las ondas generadas (Cuadradas sinusoidales señales pulsantes, etc.), con las ondas que genera un generador físico ya comercial de hardware.

En [3] utilizan las plataformas LabVIEW y arduino para un sistema que controla un brazo robótico a través de un dispositivo móvil con sistema operativo Android a distancia, un sistema de Joystick, Bluetooth, o con LabVIEW.

### 1.1.2 Descripción general del problema

Actualmente los inconvenientes que han surgido en torno a las casas habitación son problemas de gestión de energética, seguridad, confort y comunicación, a continuación, se da una pequeña descripción de cada uno de ellos:

Comunicación: Las personas requieren poder tener una comunicación remota con su casa, para obtener información mediante monitoreo y poder realizar alguna acción de control desde cualquier parte y en todo momento.

Seguridad: Las personas requieren saber que su casa y sus pertenencias dentro de esta, están seguras mientras se encuentra sola y poder prevenir posibles intrusiones, o actuar en caso de llevarse a cabo una intrusión. También requieren saber en todo momento del estado de aparatos electrónicos, puertas, ventanas, etc.

Confort o bien estar: Los usuarios requieren tener un mayor confort y bien estar dentro de su domicilio; por ejemplo, quien no desea tener el control de toda su casa desde la comodidad del sillón de su sala o desde su propia cama todo ello con su propio teléfono inteligente, o computador portátil.

Gestión energética: Los propietarios desean que los gastos de su vivienda se minimicen al máximo y en cuestión de energía no es la excepción.

La **domótica** como solución, cubre de manera muy adecuada los problemas mencionados anteriormente. En este trabajo, la mayor parte de él, tendrá un enfoque dirigido a dar comodidad y seguridad; aunque con el uso de los sistemas de hardware y software, también se cubre de buena manera la gestión energética y la comunicación usuario-sistema. Es por esto que se decidió realizar este trabajo aprovechando la capacidad en conjunto de las plataformas LabVIEW y Arduino.

## 1.2. Objetivos de la Tesis

### 1.2 1 Objetivo general.

Diseñar e implementar una interfaz gráfica, mediante el desarrollo del entorno de programación visual gráfica (**LabVIEW**), realizando una interface con la plataforma electrónica de programación en arquitectura abierta (**Arduino**), para la automatización de una vivienda; es decir proporcionar al usuario, confort (bienestar), seguridad, gestión energética, y comunicación.

### 1.2 2 Objetivos particulares

1. Diseñar un software en LabVIEW mediante programación gráfica.
2. Instaurar la comunicación del software con el hardware mediante la Interfaz LabVIEW para Arduino (LIFA) *Toolkit*.
3. Establecer una comunicación vía inalámbrica mediante la publicación web del VI

## 1.3. Justificación

Una situación actual que vive nuestro país es la inseguridad por tal motivo es importante tener acceso en todo momento y desde cualquier parte a nuestro patrimonio (vivienda), es decir poder tener comunicación vía remota con la vivienda en la cual se logre encendido y/o apagado de iluminación y alarmas, apertura y/o cierre de puertas y ventanas etc. Lo cual nos brinda certeza en cuanto a seguridad. Al mismo tiempo el contar con una casa automática nos permite tener un control de la energía, logrando así tener un mejor ahorro energético; por ejemplo, mediante sensores de movimiento y/o proximidad permite poder encender solo las luces que se requieran. Una vivienda automática nos da acceso a tener un mejor confort y bien estar dentro de la vivienda.

## 1.4 Metodología

1. Revisión bibliográfica relacionada con el tema de desarrollo.
2. Analizar caso de estudio y delimitar las acciones físicas a controlar.
3. Diseñó de la interfaz gráfica de usuario en LabVIEW para el control domótico.

4. Desarrollo de la comunicación de software (Interface Gráfica LabVIEW) con hardware (Tarjeta electrónica Arduino)
5. Desarrollo de la comunicación inalámbrica mediante la pc y telefonía Inteligente.
6. Revisión del funcionamiento de la interfaz de forma física para el cumplimiento de los objetivos planteados.

## 1.5 Contenido de la tesis

La tesis está compuesta por 5 capítulos los cuales describen la filosofía de la tesis, los ambientes de programación con las plataformas usadas, características de las mismas, desarrollo del instrumento virtual, los resultados de este, uso del instrumento virtual de manera remota, así mismo como conclusiones y trabajos futuros, estos capítulos se describen de manera individual en el punto número 1.6.

## 1.6 Descripción de capítulos

### **Capítulo 1:** Introducción

En este capítulo se describe el estado del arte, es decir que se ha desarrollado usando estas dos plataformas en conjunto, los objetivos de la tesis, las metodologías a usar, la justificación de la tesis y un resumen breve del contenido del cada capítulo de la tesis.

### **Capítulo 2:** Conceptos Fundamentales

En este capítulo se dan a conocer algunos conceptos fundamentales necesarios para el desarrollo del instrumento virtual, entre ellos, el concepto de la domótica, también se presentan algunos métodos y ejemplos de lo que es la programación virtual en LabVIEW, así mismo se muestra conceptos fundamentales de los micro-controladores arduino y finalmente los fundamentos de la unión de las dos plataformas LabVIEW con Arduino.

### **Capítulo 3:** Caso de estudio

En este capítulo se presenta el caso de estudio, primeramente, se muestra el plano de la casa que se automatizo, después se da a conocer el proceso de desarrollo del instrumento virtual mediante la plataforma LabVIEW-Arduino.

## **Capítulo 4:** Resultados y Comunicación inalámbrica

En este capítulo se muestran los resultados obtenidos durante la aplicación de la Interfaz Gráfica de manera física en tiempo real, también se dará a conocer la metodología a usar para realizar una comunicación inalámbrica de manera remota con el instrumento virtual, así mismo como los procedimientos realizados y los resultados obtenidos.

## **Capítulo 5:** Conclusiones y trabajos futuros

En este capítulo se aborda las conclusiones generales basadas en los resultados obtenidos, también se presentan conclusiones personales, así mismo en este capítulo se abordan los trabajos futuros propuestos que se pueden realizar en esta área de la ingeniería.

# Capítulo 2

## Conceptos fundamentales

### 2.1 Introducción.

En este capítulo se desarrolló una serie de **conceptos fundamentales** para tener una amplia **perspectiva de la elaboración de la interfaz de usuario** basada en las plataformas de arduino y LabVIEW, tales conceptos como conceptos de **domótica inmótica, instrumentos electrónicos y virtuales**, también lo que es la **programación virtual en LabVIEW**, así mismo se muestra un contenido de los **micro-controladores arduino** y finalmente los fundamentos para la unión de las dos plataformas **LabVIEW con Arduino**.

#### 2.1.1 Instrumentación electrónica y virtual

##### Instrumentación electrónica

Es un área de la electrónica que estudia y se encarga de realizar sistemas (equipos) electrónicos mediante circuitería, destinados a la adquisición, medición visualización, generación y conversión de información electrónica ya sea de manera digital o analógica y generalmente estos sistemas se encargan de la conversión de una a otra, es decir de digital a analógico o viceversa; un ejemplo de un sistema con instrumentación electrónica es el de un osciloscopio como el que se muestra en la Figura 1 Ejemplo de un dispositivo con instrumentación electrónica.

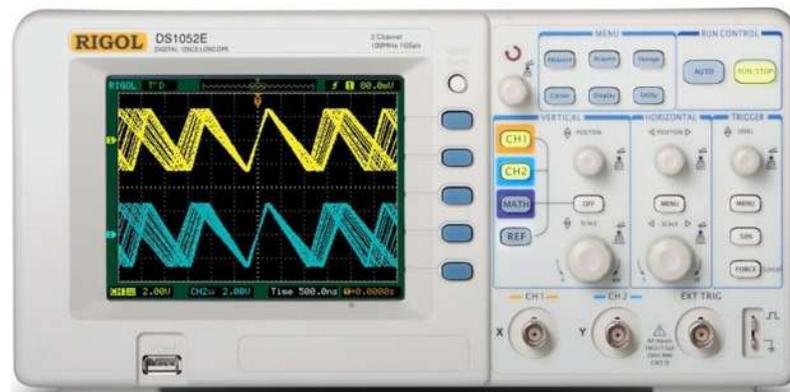


Figura 1 Ejemplo de un dispositivo con instrumentación electrónica.

Imagen recuperada de <https://www.excaliburengineering.com/condition/New/Rigol-DS1052E-50MHz-2Ch-1GSa-s-Digital-Oscilloscope-New.html>

##### Instrumentación virtual

Es un sistema de instrumentación con el cual el usuario además de medir puede procesar, analizar, almacenar, controlar, distribuir e incluso presentar la información; sobre procesos físicos que se llevan a cabo por medio de actuadores y sensores.

*“La instrumentación virtual se refiere a una poderosa aplicación de software, la cual, junto a una tarjeta electrónica, forman las funciones típicas de los instrumentos tradicionales: osciloscopios, generadores de funciones, voltímetros, etc.”*

*“Cada instrumento virtual consta de dos partes: software y hardware. Un instrumento virtual generalmente tiene un precio accesible y muchas veces mucho menor que los instrumentos tradicionales similares para una tarea de medición actual.” [2]*

El ejemplo de un instrumento diseñado mediante Instrumentación Virtual es el que se muestra en la Figura 2 Osciloscopio diseñado con Instrumentación virtual.

Sin duda la instrumentación virtual le lleva una gran ventaja frente a la instrumentación electrónica debido a su flexibilidad y menor coste y para dar una perspectiva de ello podemos ver la

Tabla 1 Comparación de instrumento virtual frente a instrumento electrónico

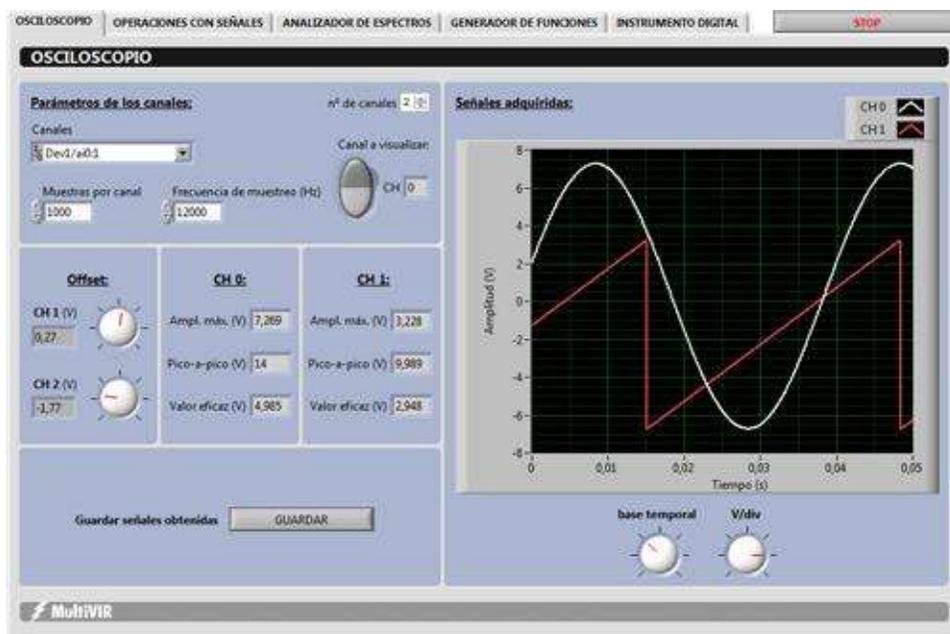


Figura 2 Osciloscopio diseñado con Instrumentación virtual.

Imagen recuperada de <https://www.ehu.es/es/web/multivir/osciloscopio>

Tabla 1 Comparación de instrumento virtual frente a instrumento electrónico

| <b>Instrumento virtual</b>  | <b>Instrumento electrónico</b>  |
|---|---|
| <i>Lo diseña el usuario o a petición de este.</i>                   | <i>Lo diseña un fabricante</i>  |
| <i>Las funciones pueden ser mejoradas</i>                           | <i>Funciones fijas</i>  |
| <i>Su control es de un ordenador o un dispositivo móvil</i>         | <i>Su control es directamente desde la interfaz que el fabricante creo.</i> |
| <i>Es de menor costo (mismas funciones)</i>                         | <i>Es mayor costo (mismas funciones)</i>                                    |
| Requiere microprocesador  | Requiere microprocesador  |
| Requiere Módulos de adquisición de datos, y puertos de comunicación | Requiere Módulos de adquisición de datos, y puertos de comunicación         |
| Requiere de interfaz de usuario generalmente gráfica                | Requiere de interfaz de usuario no necesariamente gráfica                   |
| Maneja más Software   | Maneja más Hardware   |
| Arquitectura abierta  | Arquitectura cerrada  |
| Menor tamaño en cuanto a hardware                                   | Mayor tamaño en cuanto a hardware   |

### 2.1.2 ¿Qué es un edificio inteligente?

Un edificio inteligente es aquel que incorpora un sistemas de información que soporta el flujo de está a lo largo de todo el edificio, ofreciendo servicios avanzados de automatización de la actividad y telecomunicaciones; permitiendo además un control automatizado, monitorización, gestión y mantenimiento de los diferentes subsistemas o servicios del edificio, de manera óptima e integrada, local o remota y, finalmente, diseñados con la suficiente flexibilidad como para posibilitar de manera sencilla y económica la implantación de sistemas futuros.

Antes que nada, debemos aclarar que una casa domótica no es una casa inteligente, es simplemente una casa automatizada. [4]

## 2.2 Domótica e Inmótica

### 2.2.1 ¿Que es domótica?

Domótica: Según el diccionario de la Real Academia Española es el conjunto de sistemas que automatizan las diferentes instalaciones de una vivienda. [5]

Domotica:Es un sistema el cual permite a los usuarios de una casa habitación,realizar acciones de manera automática, de tal forma que no se requiera que el usuario realice la acción de manera física, sino mas bien, esta acción se realice por un actuador, mediante un sistema de control el cual sea función de una interface que el usuario manipule, de manera muy simple [6].

Instalación domotica consta de sensores y actuadores como motores los cuales se encargan de recabar la informacion de diferentes magnitudes físicas y de realizar la acción física respectivamente, almenos una unidad de control para procesar la informacion y tambien cuenta con una interfaz de usuario para visualizar y controlar el sistema domotico.

*“La Domótica nace a escala pública en Estados Unidos a en la década del 70, tras la crisis petrolera con el objetivo principal de generar un ahorro en los consumos. Domótica es el término científico que se utiliza para denominar la parte de la tecnología (electrónica e informática), que integra el control y supervisión de los elementos existentes en un hogar” [7]*

*“Etimológicamente la palabra domótica fue fijada en Francia, país pionero en el tema en el continente europeo. El termino procede de la unión de Domus (Casa en latín) y robótica. La domótica es en esencia un conjunto de servicios realizados por automatismo dentro del hogar y dirigidos a la gestión de cuatro funciones: Control energético, Seguridad, confort y telecomunicaciones.” [4]*

## 2.2.2 ¿Que es Inmótica?

Inmótica: es cuando la domótica se integra a edificios como oficinas edificios instituciones públicas, escuelas..., con el mismo objeto que busca la domótica la única diferencia es el entorno donde esta se aplica, aunque sacrificando un poco el confort, por mayor seguridad y comunicación [6].

*“La Inmótica puede ser definida como la incorporación al equipamiento de edificios de uso terciario o industrial (oficinas, edificios corporativos, hoteles, aeropuertos, y similares), de sistema de gestión técnica automatizada de las instalaciones, que presenta como objetivo principal el ayudar a la gestión del edificio, ofreciendo así, el reducir el consumo de energía, el aumentar el confort, bienestar y la seguridad”[7]*

## 2.3 Lenguaje Gráfico de Programación.

### 2.3.1 Antecedentes

*National Instruments* fue la empresa que creo esta herramienta en el año 1976 en Austin, Texas, sus primeros productos eran dispositivos para el bus de instrumentación GPIB. En el mes de abril de 1983 comienza el desarrollo de LabVIEW, pero fue hasta el año de 1986, cuando salió la primera versión de esta

herramienta llamada LabVIEW 1.0, la cual funcionaba inicialmente sobre máquinas MAC para ser más específico sobre Macintosh, que era las máquinas más populares del momento por su capacidad de manejo en interfaz gráfica. La primera versión para Windows salió en septiembre de 1992. Las principales versiones de desarrollo de LabVIEW se pueden ver en la Tabla 2 Evolución de las versiones de LabVIEW. Actualmente está disponible para las plataformas Windows, UNIX, MAC y GNU/Linux. Las versiones más actuales se pueden usar simultáneamente para el diseño del firmware de un instrumento RF de última generación, a la programación de alto nivel del mismo instrumento, todo esto con código abierto. Y en algunas versiones están disponibles versiones demo para estudiantes y profesionales, estas versiones se pueden descargar directamente de la página *National Instruments* en el sitio web <http://www.ni.com/es-mx.html>. [8]

### 2.3.2 Introducción

Las interfaces desarrolladas con LabVIEW se llaman Instrumentos Virtuales, o VI's, ya que, aunque hoy en día se ha expandido ampliamente no sólo al control de todo tipo de electrónica (Instrumentación electrónica) sino también a su programación embebida, comunicaciones, matemáticas, etc.

*Tabla 2 Evolución de las versiones de LabVIEW*

| <b>Fecha</b>       | <b>Versión</b>   |
|--------------------|--|
| Octubre de 1986    | LabVIEW 1.0 para Macintosh   |
| Enero de 1990      | LabVIEW 2.0  |
| Septiembre de 1992 | LabVIEW para Windows   |
| Octubre de 1992    | LabVIEW para Sun   |
| Octubre de 1993    | LabVIEW 3.0 multiplataforma  |
| Abril de 1994      | LabVIEW para Windows NT  |
| Octubre de 1994    | LabVIEW para Power Macintosh   |
| Octubre de 1995    | LabVIEW para Windows 95  |
| Mayo de 1997       | LabVIEW 4.0  |
| Marzo de 1998      | LabVIEW 5.0  |
| Febrero 1999       | LabVIEW 5.1, LV para Linux y LV Real-Time  |
| Agosto de 2000     | LabVIEW 6i   |
| Enero de 2002      | LabVIEW 6.1  |
| Mayo de 2003       | LabVIEW 7 Express, LabVIEW PDA y FPGA  |
| Mayo de 2004       | LabVIEW 7.1  |
| Mayo de 2005       | LabVIEW DSP  |
| Junio de 2005      | LabVIEW Embedded   |
| Octubre de 2005    | LabVIEW 8  |
| Agosto de 2006     | LabVIEW 8.20(Edición especial por el 20 aniversario)                             |
| Abril de 2007      | LabVIEW 8.5, 1 <sup>era</sup> version del toolkit FPGA y del toolkit State chart |

|                 |   |
|-----------------|---|
| Octubre de 2008 | LabVIEW 8.6, limpieza automática de los diagramas |
| 2009            | LabVIEW 2009                                      |
| 2010            | LabVIEW 10.0 (2010)                               |
| 2010            | LabVIEW 2010 SP1                                  |
| 2011            | LabVIEW 2011                                      |
| 2011            | LabVIEW 2011 SP1                                  |
| 2012            | LabVIEW 2012                                      |
| 2012            | LabVIEW 2012 SP1                                  |
| 2013            | LabVIEW 2013                                      |
| 2013            | LabVIEW 2013 SP1                                  |
| 2014            | LabVIEW 2014                                      |
| 2014            | LabVIEW 2014 SP1                                  |
| 2015            | LabVIEW 2015                                      |
| 2015            | LabVIEW 2015 SP1                                  |
| 2016            | LabVIEW 2016                                      |
| 2017            | LabVIEW 2017                                      |

Un lema tradicional de LabVIEW es: "**La potencia está en el Software**", que con la aparición de los sistemas multi núcleo se ha hecho aún más potente. Entre sus objetivos están el reducir el tiempo de desarrollo de aplicaciones de todo tipo (no sólo en ámbitos de Pruebas, Control y Diseño) y el permitir la entrada a la informática a profesionales de cualquier otro campo. LabVIEW consigue combinarse con todo tipo de software y hardware, tanto del propio fabricante - tarjetas de adquisición de datos, PAC, Visión, instrumentos y otro Hardware- como de otros fabricantes. [7]

### 2.3.3 ¿Qué es LabVIEW?

*“LabVIEW es un software de ingeniería de sistemas que requiere pruebas, medidas y control con acceso rápido a hardware e información de datos”.* [8]

Es un entorno o plataforma de desarrollo basado en un lenguaje de programación gráfico (lenguaje G), para creación de aplicaciones de manera rápida y sencilla y sus siglas vienen del acrónimo en inglés de *Laboratory Virtual Instrument Engineering Workbench* (Laboratorio de instrumentos virtuales en una plataforma de ingeniería) el logo oficial de LabVIEW se puede ver en la **Figura 3 Logo LabVIEW** [7]

*El entorno de programación de LabVIEW simplifica la integración de hardware para aplicaciones de ingeniería, así usted tiene una manera consistente de adquirir datos desde hardware de NI y de terceros. LabVIEW reduce la complejidad de la programación, así usted puede enfocarse en su problema de ingeniería. LabVIEW le permite visualizar resultados inmediatamente con la creación integrada de interfaces de usuario de clic-y-arrastre y visualizadores de datos integrados. Para convertir sus datos adquiridos en resultados del negocio reales, usted puede*

*desarrollar algoritmos para análisis de datos y control avanzado con IP de matemáticas y procesamiento de señales o reutilizar sus propias bibliotecas desde una variedad de herramientas. Para garantizar la compatibilidad con otras herramientas de ingeniería, LabVIEW puede interactuar o reutilizar bibliotecas de otros software y lenguajes de fuente abierta. [9]*

Desde lograr desarrollar máquinas inteligentes hasta poder garantizar la calidad de los dispositivos conectados, LabVIEW ha sido la solución predilecta para crear e implementar y probar el internet de las cosas. [1]



Figura 3 Logo LabVIEW

Recuperado de <https://www.diagnosys.com/es/products/test-programming-tools/ni-labview>

#### 2.3.4 Entorno

Los programas creados en LabVIEW se llaman VI y se guardan con la misma extensión (.vi) y el nombre de VI significa Instrumento Virtual (*Virtual Instrument*).

Estos programas tienen dos vistas principales: **Panel frontal**, donde están los botones, pantallas, indicadores y controles ya que esta es la parte que ve el usuario, suele tener fondo gris. Y el **Diagrama de Bloques**, el cual tiene una circuitería interna, este es donde se realiza la programación y suele tener fondo blanco véase **Figura 4 Panel Frontal (Izquierdo) Diagrama de bloques (derecho)**.

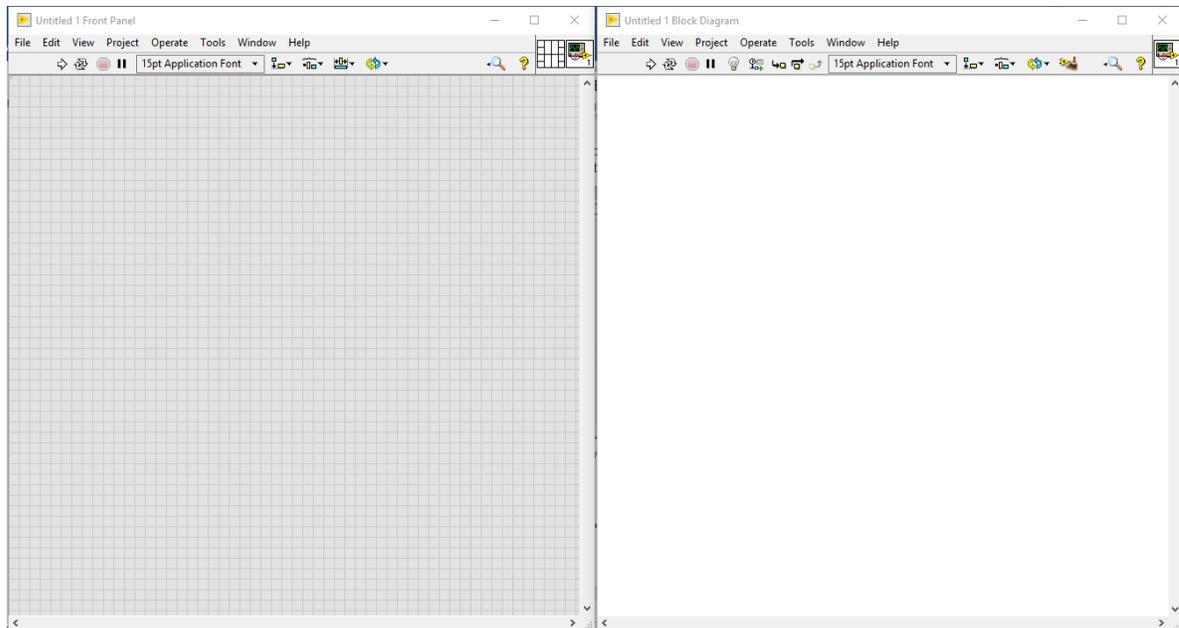


Figura 4 Panel Frontal (Izquierdo) Diagrama de bloques (derecho)

En la parte superior de las dos ventanas tenemos una barra de herramientas que nos sirve para, controlar la ejecución y la depuración de un programa y así mismo para dar formato a textos, para alinear, distribuir, agrupar y ordenar objetos y obtener la ventana de ayuda, etc. La cual la podemos ver en la **Figura 5 Barra de herramientas del panel frontal**. Cabe destacar que varía un poco la barra de herramientas en cada una de las dos ventas principales debido a sus requerimientos de funcionalidad.

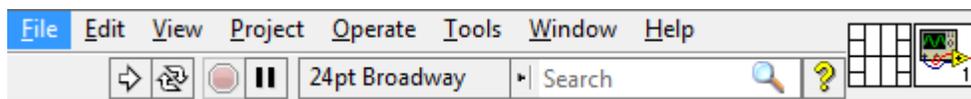


Figura 5 Barra de herramientas del panel frontal

La ventana de ayuda es de gran utilidad ya que en ella te indica la información del objeto situado bajo el cursor, esta ventana de ayuda muestra generalmente las maneras de conexión de dicho objeto y un enlace a una página de ayuda completa sobre el objeto esta ventana la podemos ver en la **Figura 6 Ventana de ayuda** se puede abrir desde la barra de herramientas en el botón de  o desde *Help> Show Context Help* o con *Ctrl+H*.

Para insertar algunas funciones al diagrama de bloques y terminales al el panel frontal se cuenta con las paletas o menús flotantes: llamado paleta de funciones (diagrama de bloques) y paleta de controles (panel frontal) véase **Figura 7 Paleta de controles (superior) y Paleta de funciones** [8]

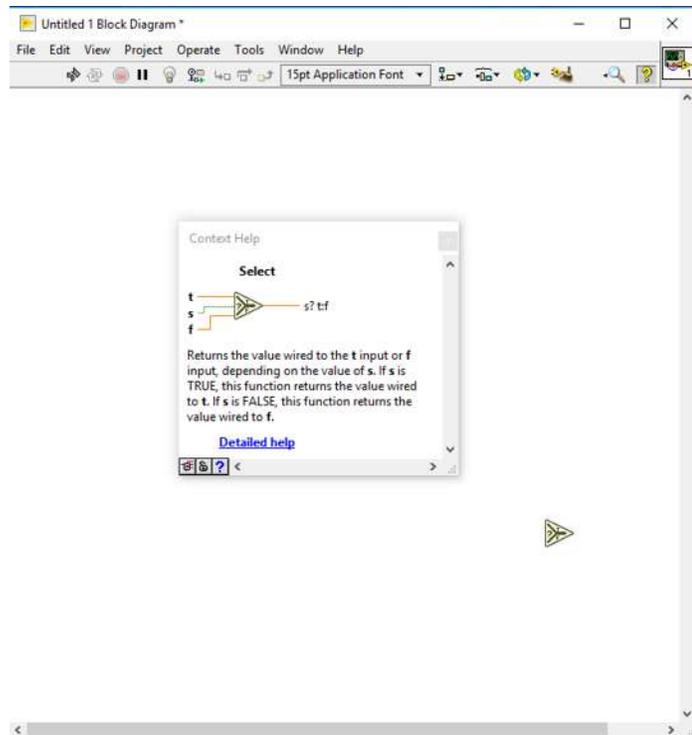


Figura 6 Ventana de ayuda

También es de suma importancia tener en cuenta los atajos mediante el teclado que esta plataforma de LabVIEW nos brinda los cuales se muestran en Tabla 3 Atajos mediante el teclado.

Y se pueden personalizar desde Tool>Options>Menu Shortcuts.

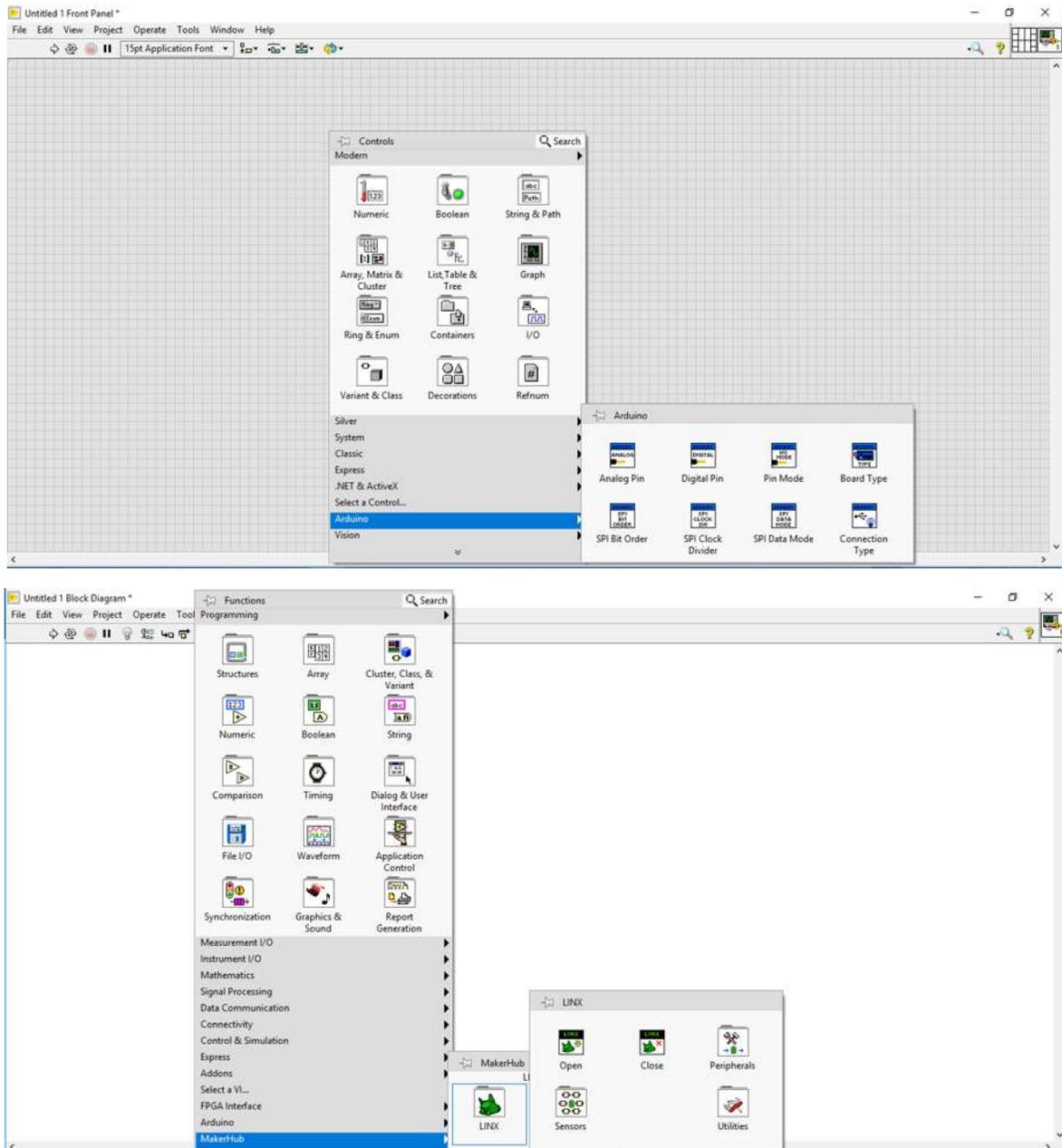


Figura 7 Paleta de controles (superior) y Paleta de funciones (inferior).

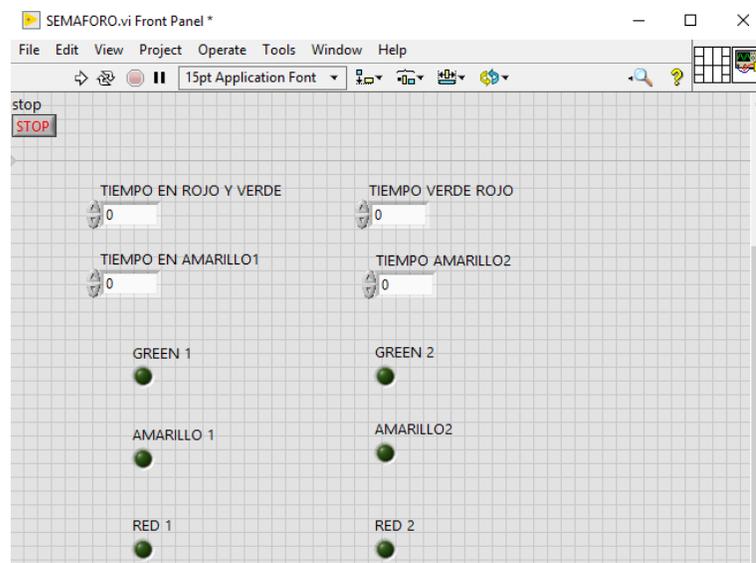
### 2.3.5 Ejemplo básico de diseño de un Vi

Como ejemplo básico de un diseño de un instrumento virtual VI mostramos el funcionamiento de un semáforo en un cruce de dos calles de manera virtual donde podemos ver los dos semáforos y cómo van cambiando sus tiempos de encendido y apagado de las 3 luces (rojo amarillo y verde) de forma sincronizada y en el cual también tenemos el control de los tiempos de encendido y apagado de las luces.

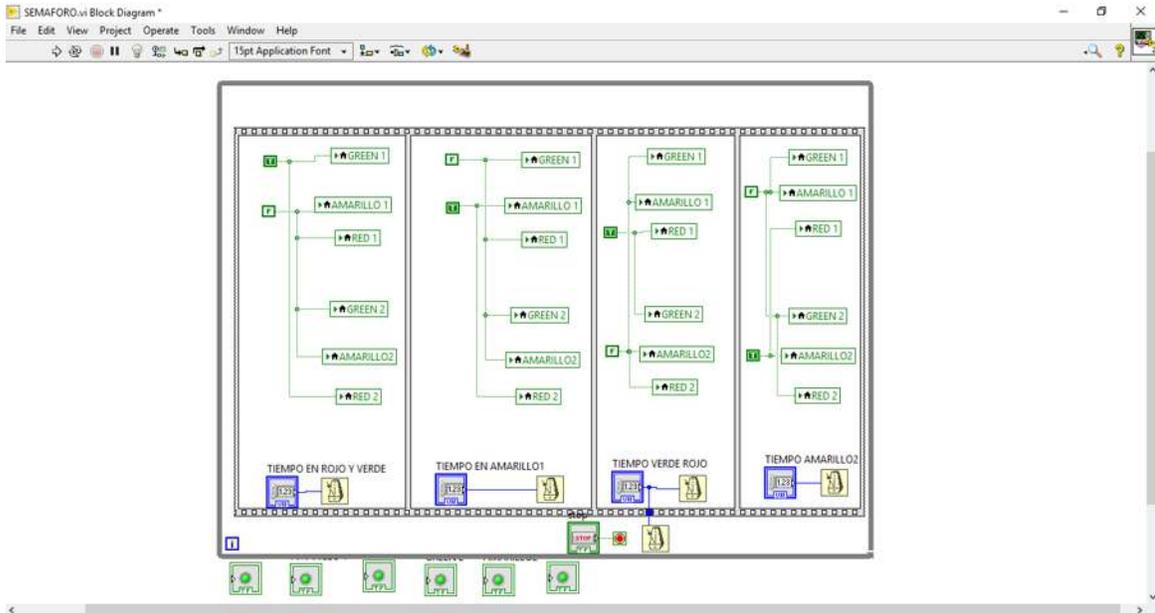
Tabla 3 Atajos mediante el teclado.

| <b>Combinaciones de teclas</b> | <b>Función</b>   |
|--------------------------------|--|
| Ctrl + .                       | Aborta ejecución                                       |
| Ctrl + ?                       | Muestra la ayuda                                       |
| Ctrl + arrastrar con el ratón  | Crea copias de los objetos seleccionados               |
| Ctrl + B                       | Elimina hilos rotos                                    |
| Ctrl + C                       | Copia objetos seleccionados al portapapeles            |
| Ctrl + E                       | Conmuta entre dos ventanas principales                 |
| Ctrl + H                       | Muestra y oculta ventana de Ayuda contextual           |
| Ctrl + R                       | Ejecuta el programa                                    |
| Ctrl + S                       | Guarda el VI   |
| Ctrl + Shift + Z               | Rehacer  |
| Ctrl + V                       | Pega los objetos desde el porta papeles                |
| Ctrl + X                       | Corta objetos seleccionados al portapapeles            |
| Ctrl + Z                       | Deshace la última acción                               |
| Ctrl + T                       | Muestra las dos ventanas principales                   |
| Shift + arrastrar con el ratón | Mueve objetos en una sola dirección                    |
| Tab                            | Cambia entre herramientas de la paleta de herramientas |

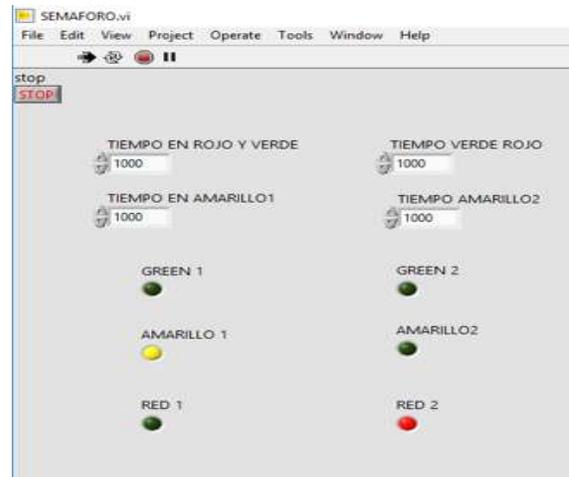
Panel frontal de semáforo

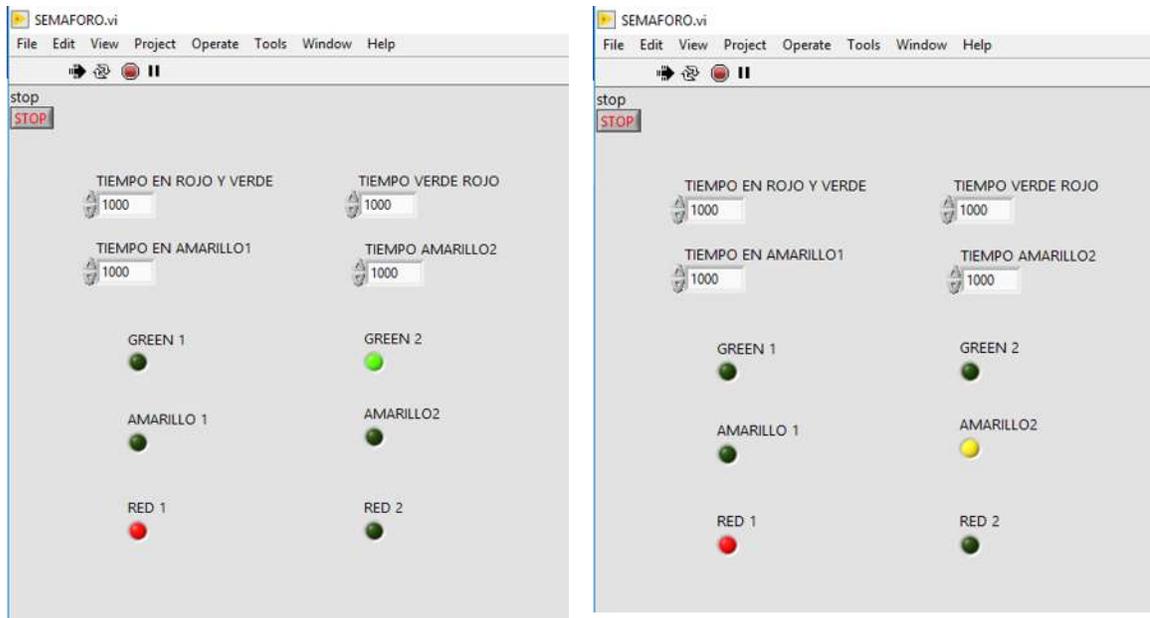


## Diagrama de bloques de semáforo



Para ello se describe su funcionamiento por medio de imágenes secuenciales que se muestran a continuación:





### 2.3.6 Aplicaciones de LabVIEW

Educativa o académica: Se utiliza para realizar de manera práctica ejemplos de temas de contenido en los planes de estudio en el área de ingeniería en nivel universitario.

Adquisición de datos y procesamiento de señales: se usa principalmente para estas cuestiones ya que para poder tener sistemas de control se requiere de la adquisición y procesamiento de la información.

Control: Para control de sistemas físico mediante el uso de VI's.

Automatización de sistemas de pruebas: se usa para automatizar cualquier sistema de pruebas para realizar un producto nuevo, o ya existente

Sistemas embebidos de monitoreo y control: puedes hacer uso de LabVIEW para este fin ya que se tiene acceso para el uso de herramientas personalizadas para las áreas de instrumentación, robótica, comunicaciones, control etc. [1, p. 15]

## 2.4 Plataforma Electrónica Arduino (microcontroladores).

### 2.4.1 Antecedentes

Arduino surge en el año 2005 como una propuesta de solución a la necesidad de un sistema electrónico para la instrumentación y automatización con una arquitectura abierta y de bajo coste; los cofundadores fueron David Cuartiles, Gialunca Martino, Massimo Banzi, David Mellis y tom Igoe.

## 2.4.2 Introducción

El nombre de arduino se toma de un bar que frecuentaban los fundadores, el cual se encuentra en un pueblo Ivream ubicado al norte de Italia dicho lugar se llama Bar di Re Arduino, llamado de esta manera en honor a el Rey Arduino. [10]

El rey Arduino primero fue marqués de Ivrea y después rey de Italia desde el año 1002 (tras el emperador Otón III) hasta el año 1014, cuando fue destronado por Enrique II. Las placas arduino tiene color azul en honor a las paredes una antigua fábrica de Olivetti, también la procedencia italiana quedó grabada en todas las placas con **un pequeño mapa de este país**, en la parte posterior de la placa, y el deseo de un diseño distinto y reconocible se plasmó en la propia fisonomía de la placa, que tiene un tamaño aproximado al de una tarjeta de crédito o de identificación. El equipo compuesto por dos italianos, dos americanos y un español: fueron los cinco cofundadores para revolucionar el 'hardware' libre de arduino, estos cinco fundadores los podemos ver en la foto de la **Figura 8 De izquierda a derecha: David Mellis, David Cuartielles, Gianluca Martino, Massimo Banzi, Tom Igoe.** [11]

La idea original fue del italiano Massimo Banzi y del español David Cuartielles como un sistema académico de arquitectura abierta, para que no existan problemas legales de uso y que este sistema sea de bajo precio y del tipo Plug and Play (enchufar y usar). También en el proyecto participaron varios tesis de ingeniería, los cuales realizaron comandos, programación de los bloques y funciones del sistema, entre ellos estaba el colombiano Hernando Barragán, el realizo la plataforma *Wiring para programar el microcontrolador*. La primera producción oficial de estas tarjetas fue alrededor de 300 ejemplares para los alumnos del instituto IVRAE para proyectos académicos, hoy en día estas tarjetas se usan para proyectos de investigación y desarrollo de tecnología y en la docencia de las principales universidades de todo el mundo. [10]



Figura 8 De izquierda a derecha: David Mellis, David Cuartielles, Gianluca Martino, Massimo Banzi, Tom Igoe.

Recuperada de [http://www.eldiario.es/hojaderouter/tecnologia/hardware/arduino-hardware-libre-revolucion-origenes-historia\\_0\\_289771082.html](http://www.eldiario.es/hojaderouter/tecnologia/hardware/arduino-hardware-libre-revolucion-origenes-historia_0_289771082.html)

### 2.4.3 ¿Qué es Arduino?

Es una plataforma electrónica con un sistema de programación en arquitectura abierta (software- hardware libre, gratuito) basado en lenguaje (C/C++), y esta utiliza micro-controladores de la familia ATMEL para así concebir un sistema embebido (*embedded system*) diseñado especialmente para un amplio espectro de automatización de procesos enfocados al área de la ingeniería. El símbolo que identifica a esta plataforma es el símbolo del infinito y dentro de cada ovoide que esta forma, se encuentra los signos de sustracción (-) y adicción (+) el cual lo podemos observar la **Figura 9 Logo Arduino..**



Figura 9 Logo Arduino.

Recuperado de <http://www.arduino.org/>

Existen definiciones más detalladas las cuales presento a continuación:

*“Arduino es una plataforma electrónica de código abierto basada en hardware y software fáciles de usar. Las tarjetas Arduino son capaces de leer entradas - luz en un sensor, un dedo en un botón, o un mensaje de Twitter - y convertirlo en una salida - activar un motor, encender un LED, publicar algo en línea. Usted puede decirle a su junta lo que debe hacer enviando un conjunto de instrucciones al microcontrolador en el tablero. Para ello se utiliza el lenguaje de programación*

*Arduino (basado en el cableado) y el software Arduino (IDE), basado en el procesamiento “. [8]*

*“Arduino es una plataforma electrónica y de programación en arquitectura abierta con amplia gama de aplicaciones en ciencias exactas e ingeniería. El icono que identifica al Arduino es el símbolo infinito, que se emplea en el área de matemáticas, insertando en su interior los signos – y +. La plataforma de este sistema Arduino gira alrededor de la familia de microcontroladores ATMEL y por medio de un entorno de programación, con las herramientas necesarias integradas para diseñar y desarrollar aplicaciones de manera sencilla en los lenguajes c y c++, representa un sistema embebido (embedded system) de propósito específico para automatizar procesos físicos.” [9, p. 3]*

## 2.4.4 Software y Hardware del sistema Arduino

Software del sistema Arduino: El software utilizado por este sistema es libre y se puede obtener de la página oficial <https://www.arduino.cc/> el cual tiene un ambiente de programación IDE, quien cuenta con herramientas integradas para edición, compilación y descargas de sketches a los tableros electrónicos. Este software también tiene ejemplos y por sus características se puede expandir o hacer compatible con otras plataformas, como LabVIEW que es el caso que daremos especial interés en esta tesis.

Hardware del sistema arduino: Los tableros (tarjetas) oficiales pueden ser ensamblados a mano (siempre y cuando se cumpla con algunos requerimientos, <https://www.arduino.cc/en/Main/FAQ>) o adquirirlos a un bajo coste en la misma página oficial los cuales forman parte de una amplia gama de productos oficiales que actualmente se ofertan, estos productos oficiales los podemos encontrar en esta liga <https://www.arduino.cc/en/Main/Products>

## 2.4.5 Sistemas embebidos

Ya he hablado de los sistemas embebidos, pero ¿Qué es un sistema embebido?

Los sistemas embebidos son una plataforma digital que consta de circuitos electrónicos integrados, los cuales cuentan al menos con un microprocesador con lenguaje de programación adecuado, también cuentan con una serie de circuitos electrónicos de instrumentación para realizar acoplamientos (digital/analógico y viceversa) mediante una serie de periféricos de entrada y salida todo ello diseñado para tener una interacción con el medio físico mediante el uso de actuadores de tal manera que se pueda dar un control de procesos en tiempo real.

*Un sistema embebido está definido como un sistema mínimo digital de propósito específico, cuya plataforma electrónica depende de un microcontrolador con el lenguaje de programación y periféricos adecuados para la realización de una tarea exclusiva de automatización en tiempo real. [9, p. 5]*

Ejemplos de sistemas embebidos: Consolas de manipulación de equipos industriales (Brazos robóticos), Reproductores de Mp3, Controles de sistemas, como en el sistema de control de velocidad en automóviles etc.

¿Qué es un sistema de control de procesos en tiempo real?

Es un sistema capaz de llevar a cabo el control de un proceso físico en tiempo real, es decir que el tiempo que le tome a sistema embebido en la adquisición, procesamiento y envío de comandos de acción sea el menor tiempo posible, de tal manera que no se puedan percibir retrasos de tiempo, entre la acción de control con la reacción física de los actuadores como lo son motores, altavoces, interruptores, relevadores, etc.

También podemos encontrar definiciones más elaboradas como la que se describe a continuación:

*“Se refiere a el tiempo de máquina que emplea el microcontrolador para realizar todas las operaciones aritméticas, desde la adquisición de datos, el procesamiento de la información y el envío de comandos, de tal manera que este tiempo de computo debe ser menor al periodo de muestreo.*

*La respuesta de un sistema en tiempo real debe de realizarse en el menor tiempo posible, en el caso de que existan retardos de tiempo, estos deben ser aceptables en relación al tipo de proceso a controlar y al valor del periodo de muestreo.” [9, p. 5]*

Sistema embebido arduino

Un sistema básico embebido de arduino contiene una unidad central de procesamiento (CPU) la cual es un microcontrolador de la familia Atmel (*ATmega 48PA/88PA/168PA/328PA*; ARM CortexM3 de 32bits a 84Mhz; Intel Quarrk Soc X1000 32bits Pentium 400MHz), una memoria flash que es utilizada para almacenar código de máquina y la ejecución eficiente del sketch, este sistema también tiene una memoria EEPROM (*Electrically Erasable Programmable Read-Only Memory*) que almacena un sistema operativo encargado de iniciarla tarjeta electrónica, configurar puertos I/O y demás funciones, también este sistema interacciona con el exterior físico mediante circuitos de instrumentación como convertidores(A/D y D/A), puertos de entrada y salida I/O y señales analógicas, on/off, pmw. También este sistema embebido puede manejar interrupciones útiles en la adquisición y manejo de datos, procesamiento y control de procesos. Todo ello con la finalidad de tener un control de procesos en tiempo real. Un diagrama a bloque del sistema embebido arduino se puede observar en la **Figura 10 Diagrama a bloques del sistema Arduino básico (AT mega 328)**.

**Los componentes principales que integran un sistema arduino son 2:**

El **primero** es un ambiente de programación IDE para el desarrollo de programas denominados *sketchs* con la finalidad de llevar acabo aplicaciones de automatización en tiempo real. Este debe ser instalado en una computadora personal, en la **Figura 11 IDE ambiente de programación de arduino** podemos visualizar la ventana principal del programa.

El ambiente de programación IDE, viene con un paquete de cómputo en el cual podemos encontrar un editor de texto para realizar los *sketchs*, compiladores para los lenguajes C/C++, los drivers necesarios para ser instalado en cualquiera de los más comunes sistemas operativos como lo son Linux, MaC OS, y Windows y una correspondiente fase de generación de código máquina para ser transmitido por medio de USB a la tarjeta electrónica. El paquete arduino se puede obtener de forma gratuita desde la página oficial en el siguiente link <https://www.arduino.cc/en/Main/Software>

Block Diagram

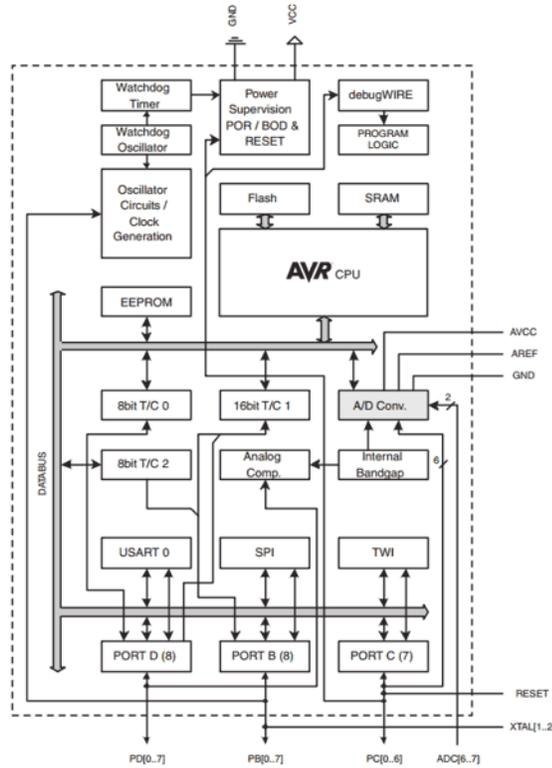


Figura 10 Diagrama a bloques del sistema Arduino básico (AT mega 328)

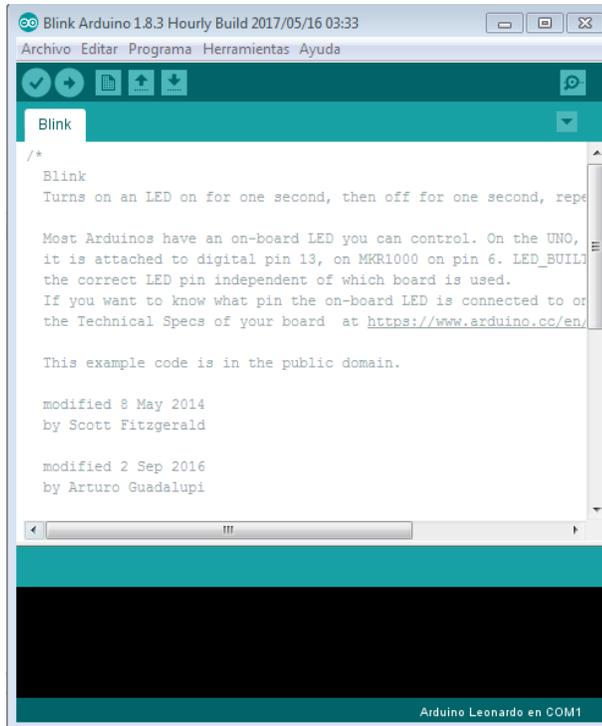


Figura 11 IDE ambiente de programación de arduino

Y el **segundo** una tarjeta electrónica la cual tiene la capacidad de recibir la información del sketch en código máquina a través de una comunicación USB sin necesidad de un programador externo como los que usan algunos tipos de microcontroladores como los pic's. Dicha tarjeta o tablero oficial se puede observar en la **Figura 12 Tablero (Tarjeta) Electrónica Arduino Leonardo**.



Figura 12 Tablero (Tarjeta) Electrónica Arduino Leonardo

Recuperado de <http://arduino.cl/arduino-leonardo/>

Mientras se encuentra conectada la tarjeta electrónica al computador y el sketch o programa está en ejecución se puede tener una comunicación en todo momento entre tarjeta electrónica y el ambiente de programación IDE para llevar a cabo cualquier tipo de intercambio bidireccional de información entre los dos componentes.

Además, una vez se ha descargado el programa (Sketch) en la tarjeta electrónica el programa queda residente en la memoria flash no volátil independientemente si se desenergiza la tarjeta electrónica o se desconecta del computador.

#### 2.4.6. Arquitectura abierta del sistema Arduino

Esta consiste en hacer una programación e implementación personalizada en función de las características específicas del proceso físico a automatizar, la arquitectura abierta toma importancia ya que hace uso de un ambiente de programación, con un conjunto de comandos y funciones, que proporcionan al diseñador, las herramientas necesarias para el diseño, desarrollo y aplicación de proyectos con la menor pérdida de tiempo a diferencia de cómo se haría con otro tipo de microcontroladores.

*“En el ambiente de programación se realiza la edición del sketch, compilación y depuración de errores de sintaxis de acuerdo con las reglas gramaticales de los lenguajes C y C++; así como la generación automática del código de máquina del correspondiente tipo de microcontrolador de la familia ATMEL y la ejecución del*

sketch es realizada por el microcontrolador. Información de interés al usuario como variables de estado y errores de posición son tramitadas a la computadora para presentarla en formato numérico o gráfico esto último en Matlab o LabVIEW admiten modificación en línea de parámetros, sin suspender la ejecución del sketch.” [9, p. 9]

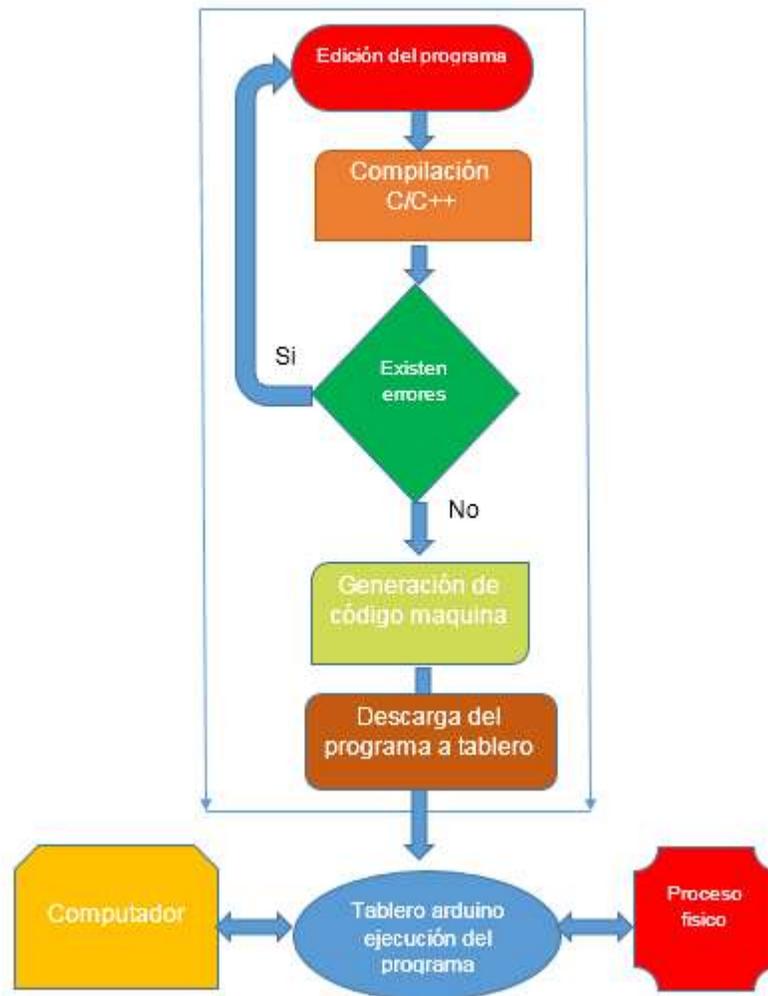


Figura 13 Pasos secuenciales para ejecutar un programa en arduino

Los pasos secuenciales para llevar a cabo la ejecución de un programa (Sketch) en el tablero de arduino y desplegar información que el usuario requiera en la computadora se ilustran en el diagrama de flujo de la **Figura 13 Pasos secuenciales para ejecutar un programa en arduino**

## 2.5 Interfaz gráfica de LabVIEW con microcontrolador arduino (LIFA).

La Interfaz LabVIEW para Arduino (LIFA) *Toolkit* es una descarga gratuita que permite a los desarrolladores adquirir datos del microcontrolador Arduino y procesarlos en el entorno de programación gráfica de LabVIEW, de la cual podemos ver su logo en la **Figura 14 Juego de herramientas, interfaz de NI LabVIEW para arduino. [13]**



Figura 14 Juego de herramientas, interfaz de NI LabVIEW para arduino

Recuperada de <https://neutrongeek.wordpress.com/tag/interface-para-arduino/>

### 2.5.1. Código a cargar a Arduino

El código que hay que cargar en la tarjeta electrónica de arduino (para este caso el arduino Mega) es el código **LIFA\_Base Arduino 1.8.2**, este código viene en el paquete de herramientas en la interfaz gráfica de LabVIEW con microcontrolador arduino (LIFA) o también se puede obtener desde la página oficial de arduino en el siguiente link [www.ni.com/arduino](http://www.ni.com/arduino), este código se carga con la finalidad de que la tarjeta electrónica funcione como una tarjeta de adquisición de datos muy similar a las DAQ de NI, el cual se muestra a continuación:

```
/*  
*****  
** LVFA_Firmware - Provides Basic Arduino Sketch For Interfacing With LabVIEW.  
** Written By: Sam Kristoff - National Instruments  
** Written On: November 2010  
** Last Updated: Dec 2011 - Kevin Fort - National Instruments  
** This File May Be Modified And Re-Distributed Freely. Original File Content  
** Written By Sam Kristoff And Available At www.ni.com/arduino.  
*****  
/*  
*****  
** Includes.
```

```
*****/  
  
// Standard includes. These should always be included.  
  
#include <Wire.h>  
#include <SPI.h>  
#include <Servo.h>  
#include "LabVIEWInterface.h"  
  
/*****  
  
** setup()  
  
** Initialize the Arduino and setup serial communication.  
  
** Input: None  
  
** Output: None  
  
*****/  
  
void setup()  
{  
  // Initialize Serial Port With The Default Baud Rate  
  syncLV();  
  // Place your custom setup code here  
}  
  
/*****  
  
** loop()  
  
** The main loop. This loop runs continuously on the Arduino. It  
** receives and processes serial commands from LabVIEW.  
  
** Input: None  
  
** Output: None  
  
*****/  
  
void loop()  
{  
  // Check for commands from LabVIEW and process them.  
  checkForCommand();  
  // Place your custom loop code here (this may slow down communication with LabVIEW)  
  if(acqMode==1)  
  {  
    sampleContinuously();  
  }  
}
```

## Capítulo 3. Caso de estudio

### Introducción.

En este capítulo se dará a conocer la tarjeta electrónica a usarse, así como sus especificaciones tanto generales como de pines, también se da a conocer el alcance y delimitación del proyecto de tesis, los planos desarrollados en AutoCAD en 2D y 3D de casa a automatizar, de igual manera se dará una explicación detallada de los programas que se requiere ser instalados en el ordenador que ejecutará la interfaz gráfica.

Y finalmente se desarrolla una **interface gráfica** usando las plataformas LabVIEW y Arduino mediante la herramienta que NI proporciona, con lo cual esta interfaz gráfica permite que el usuario tenga una comunicación y control del sistema domótico de su casa habitación, mediante una computadora personal de manera local.

### 3.1 Tarjeta para adquisición de datos

La tarjeta electrónica para la adquisición de datos que se uso es el **Arduino Mega**

Arduino Mega es una placa que tiene un microcontrolador ATmega1280, esta placa tiene 54 pines digitales de entrada / salida (de los cuales 14 se pueden usar como salidas PWM), 16 entradas analógicas, 4 UART (puertos serie de hardware), un oscilador de cristal de 16 MHz, una conexión USB, un conector de alimentación, un encabezado ICSP, y un botón de reinicio. [12]

#### 3.1.1 Esquema y disposición de pines

*El esquema de esta placa se puede observar en la **Figura 15 Esquema de la tarjeta Arduino MEGA** y la disposición de pines se puede ver de manera más sintetizada en la **Figura 16 Disposición de pines Arduino Mega***

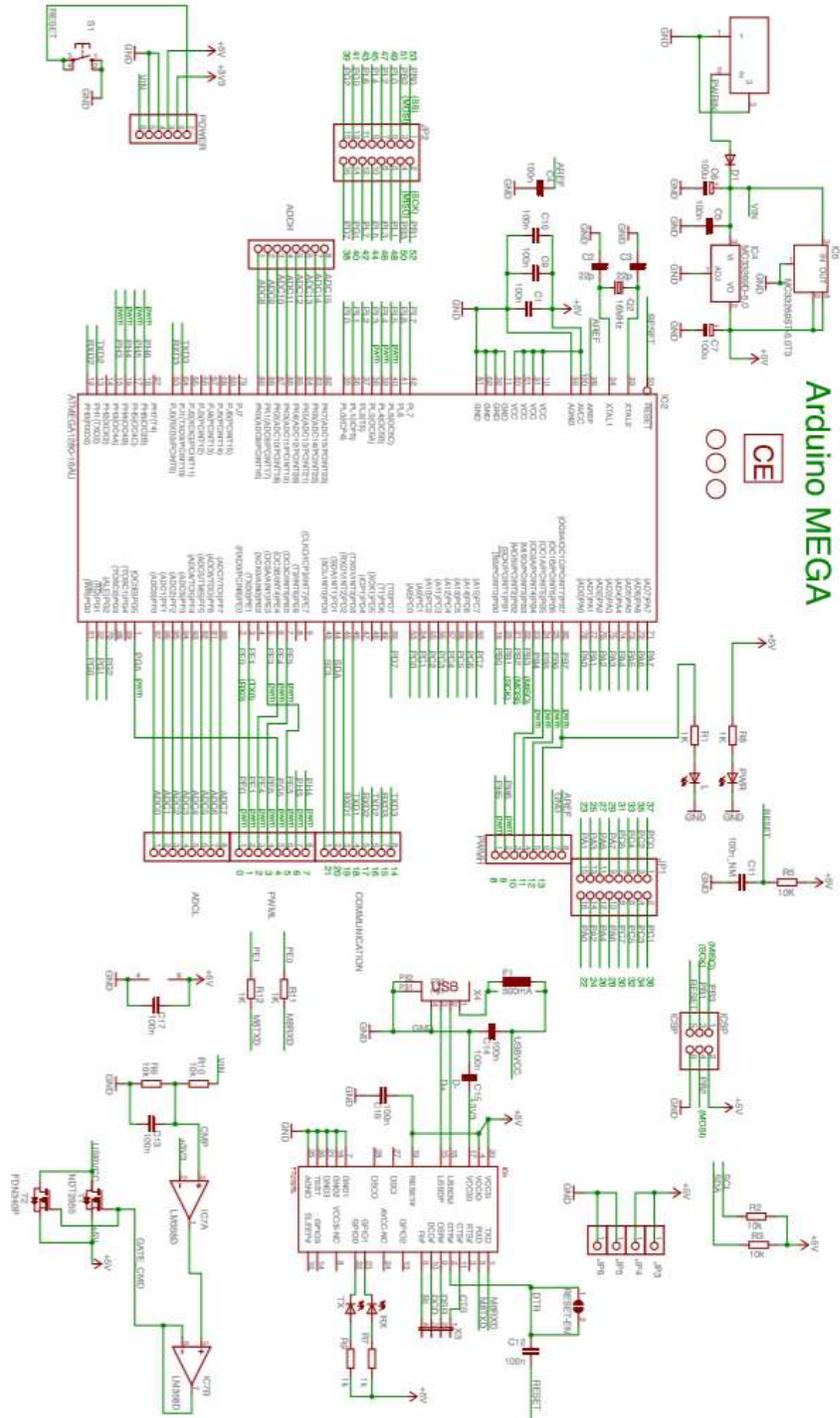
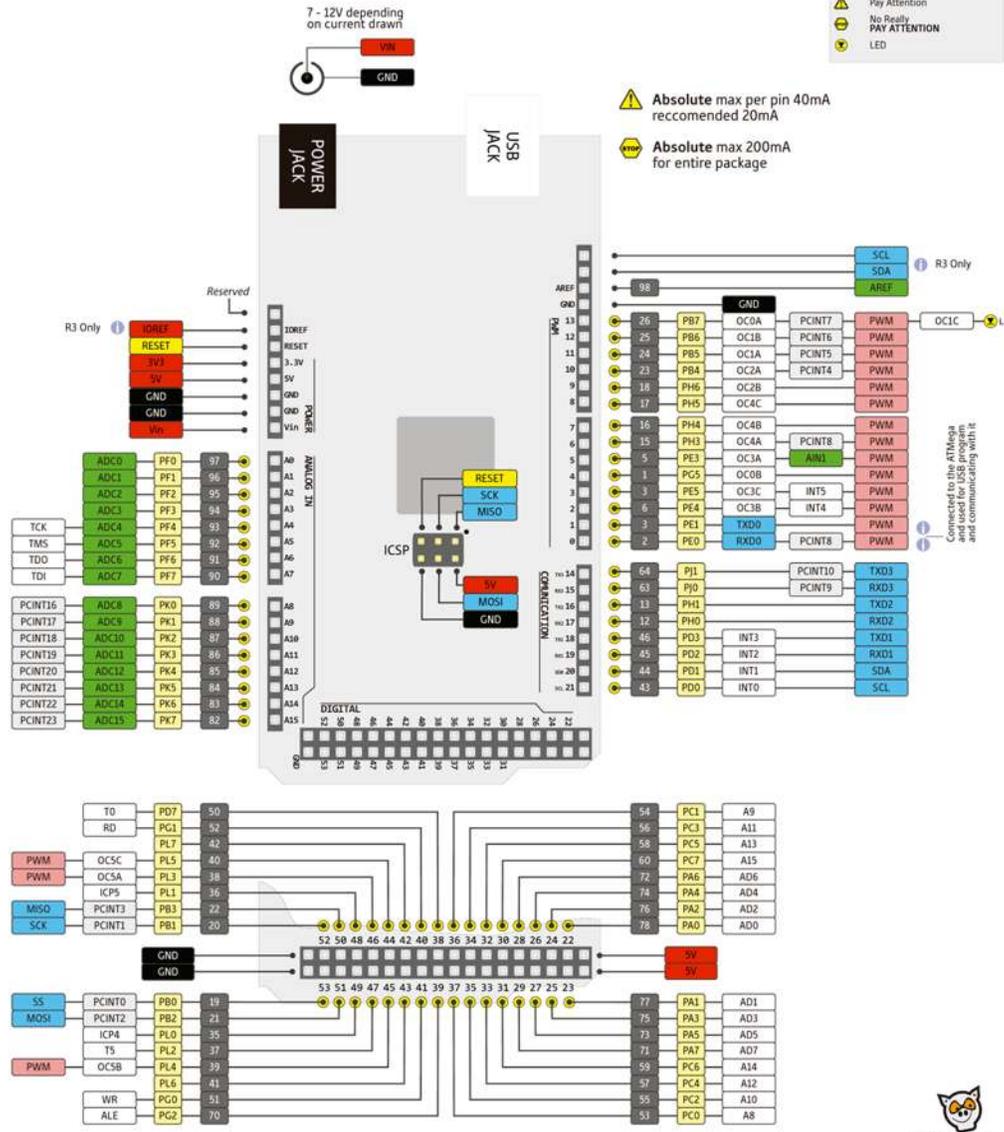


Figura 15 Esquema de la tarjeta Arduino MEGA

Recuperada de <https://www.arduino.cc/en/uploads/Main/arduino-mega-schematic.pdf>

THE UNOFFICIAL  
**ARDUINO MEGA**  
 PINOUT DIAGRAM



version 1.0 05/02/2012

Figura 16 Disposición de pines Arduino Mega

Recuperada de <https://articulo.mercadolibre.com.mx/MLM-551039155-arduino-mega-2560-usb-programador-oferta-JM?source=gps>

### 3.1.2 Especificaciones generales de la tarjeta.

Las especificaciones de generales de esta tarjeta electrónica se muestran en la **Tabla 4 Especificaciones de la tarjeta Electrónica Arduino Mega**

Tabla 4 Especificaciones de la tarjeta Electrónica Arduino Mega

| <b>Descripción</b>               | <b>Características</b>   |
|----------------------------------|--|
| Microcontrolador                 | ATmega1280   |
| Tensión de funcionamiento        | 5V   |
| Voltaje de entrada (recomendado) | 7-12V  |
| Voltaje de entrada (límites)     | 6-20V  |
| Pines de E / S digitales         | 54 (de los cuales 15 proporcionan salida de PWM)               |
| Clavijas de entrada analógica    | dieciséis  |
| Corriente DC por Pin E / S       | 40 mA  |
| Corriente DC para 3.3V Pin       | 50 mA  |
| Memoria flash                    | 128 KB de los cuales 4 KB utilizados por el gestor de arranque |
| SRAM                             | 8 KB   |
| EEPROM                           | 4 KB   |
| Velocidad de reloj               | 16 MHz   |

### 3.2. Alcance y delimitación de la tesis.

La interfaz gráfica se encarga de controlar y mantener una comunicación del usuario con la casa domótica, esta interfaz gráfica de usuario tiene la capacidad de controlar **cinco puertas, cuatro ventanas, un sistema de alarma (cámara y bocina), las luces de dos recamaras, un baño, un garaje espacio para sala y cocina y luces exteriores** en la **Figura 18 Plano 3D de la casa Automatizada** podemos ver los planos que tiene dicha casa domótica.

### 3.3. Desarrollo de los planos en AutoCAD de la casa automatizada.

Se desarrolló el plano en 2d mediante la herramienta de AutoCAD, el cual se puede observar en la **Figura 17 Plano 2D de la casa automatizada**.

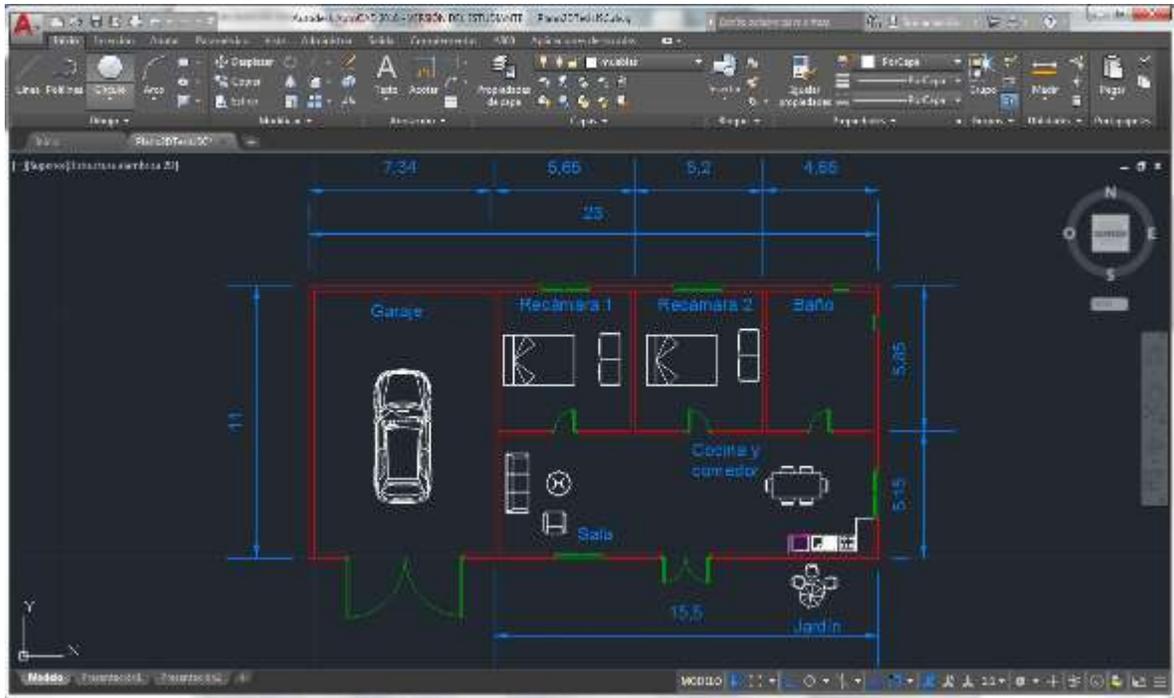


Figura 17 Plano 2D de la casa automatizada.

De igual manera mediante esta herramienta se realizó el plano en 3D este se puede apreciar en la **Figura 18 Plano 3D de la casa Automatizada**

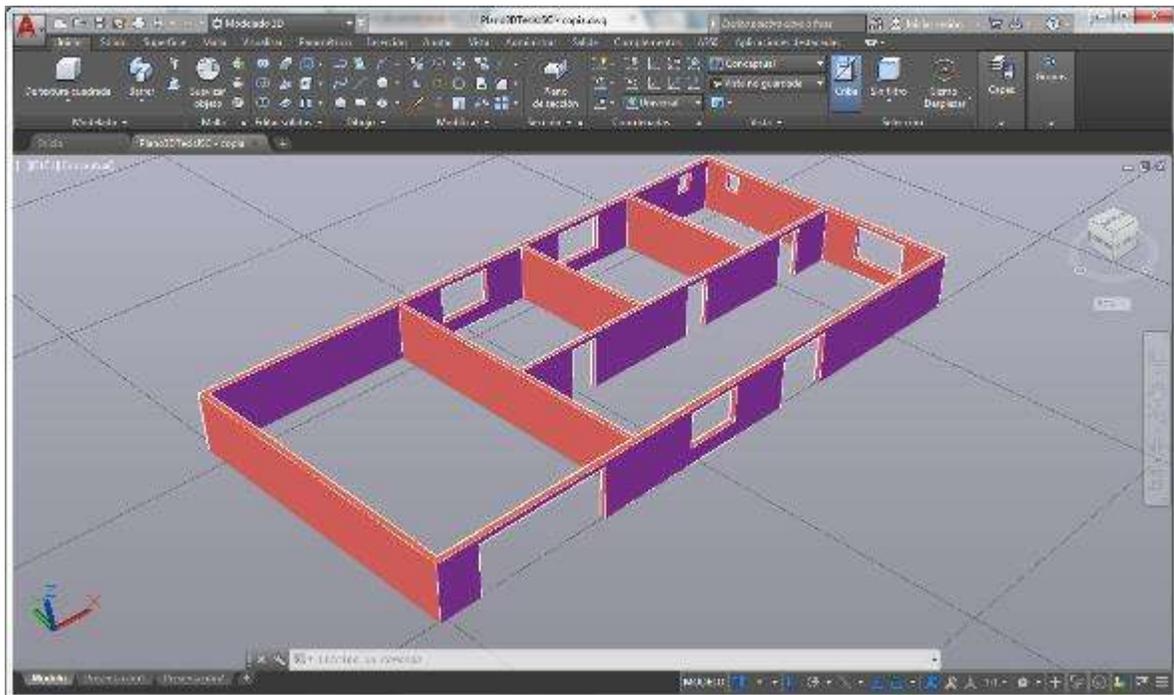


Figura 18 Plano 3D de la casa Automatizada

### 3.3.1 Desarrollo de la maqueta basada en los planos en AutoCAD de la casa automatizada.

Se creó una maqueta a escala con la finalidad de verificar el funcionamiento de la interface gráfica a realizar, esta maqueta se realizó mediante los planos presentados anteriormente y la cual muestro en la **Figura 19 Maqueta para implementar la automatización.**



*Figura 19 Maqueta para implementar la automatización.*

## 3.4 Instalación de las Plataformas y paquetes

Inicialmente para desarrollar la interfaz gráfica, se requiere tener una serie de programas instalados en la computadora, como lo es el programa de LabVIEW y varios programas de NI que son necesarios para el funcionamiento de nuestra interfaz, es por ello que a continuación dedico una parte para explicar estos procesos de instalación.

### 3.4.1. Instalación de LabVIEW

Para la instalación de los productos LabVIEW se sigue una serie de pasos que muestro a continuación:

1. Descargar el paquete para 32 o 64 bits (según corresponda) que contiene los productos de LabVIEW desde la página oficial de NI en el siguiente link <http://www.ni.com/downloads/products/esa/>
2. Una vez descargados procedemos a descomprimir el archivo.

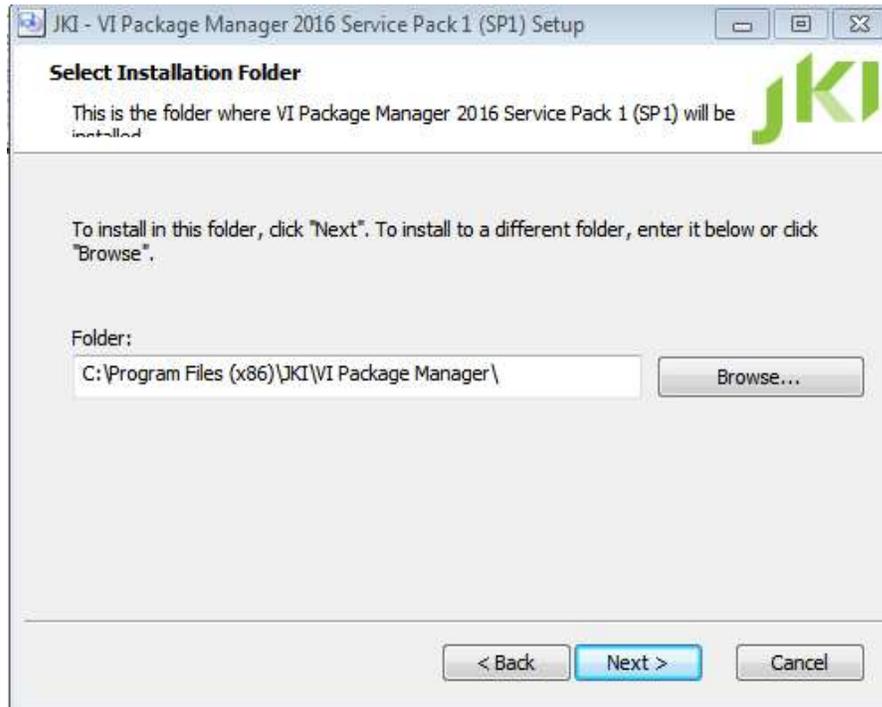
3. Queda una carpeta que contiene el ejecutable (.exe) dar clic derecho sobre este y en ejecutar como administrador y otorgar permisos.
4. En la ventana que aparece dar clic en *Next >*



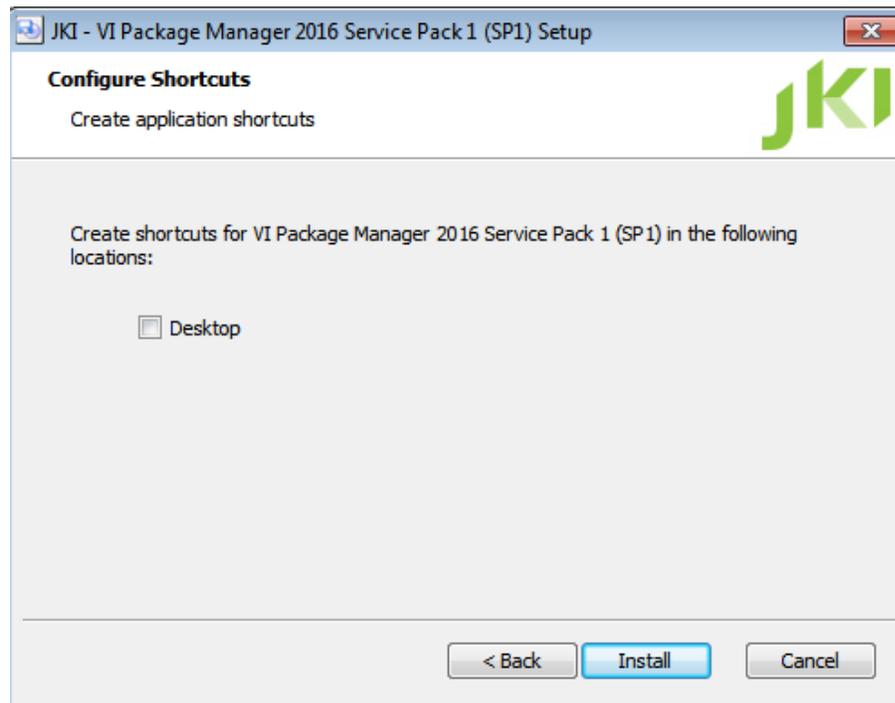
5. Aceptar términos y condiciones y clic en *Next >*



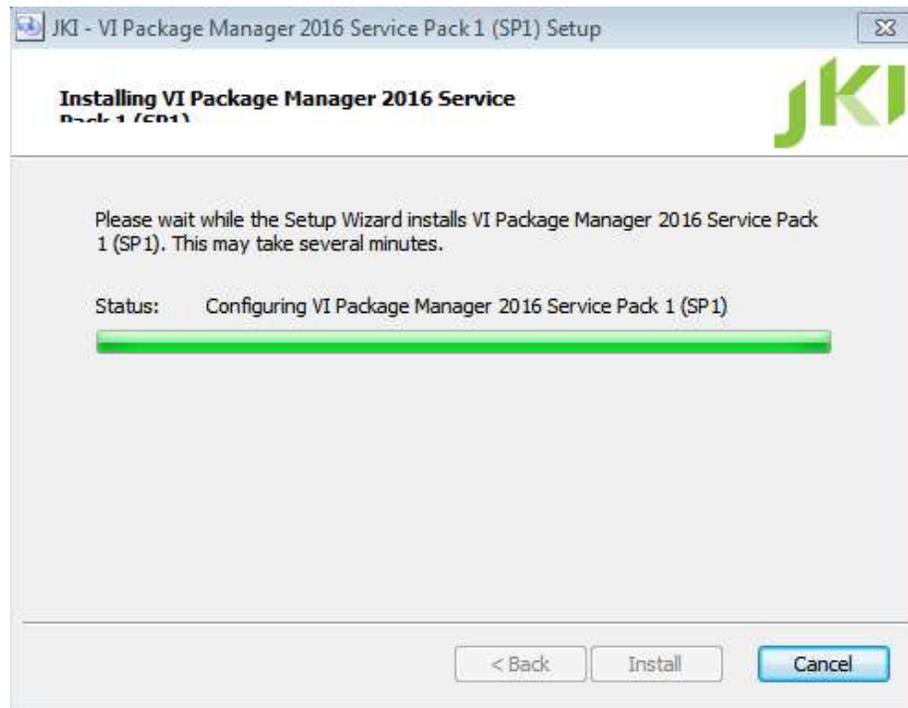
6. Seleccionar la carpeta donde se desee instalar o dejarla predeterminada(recomendado) y clic en *Next >*



7. Después seleccionar o no para crear un icono de acceso directo en escritorio según se desee y clic en *Install*.



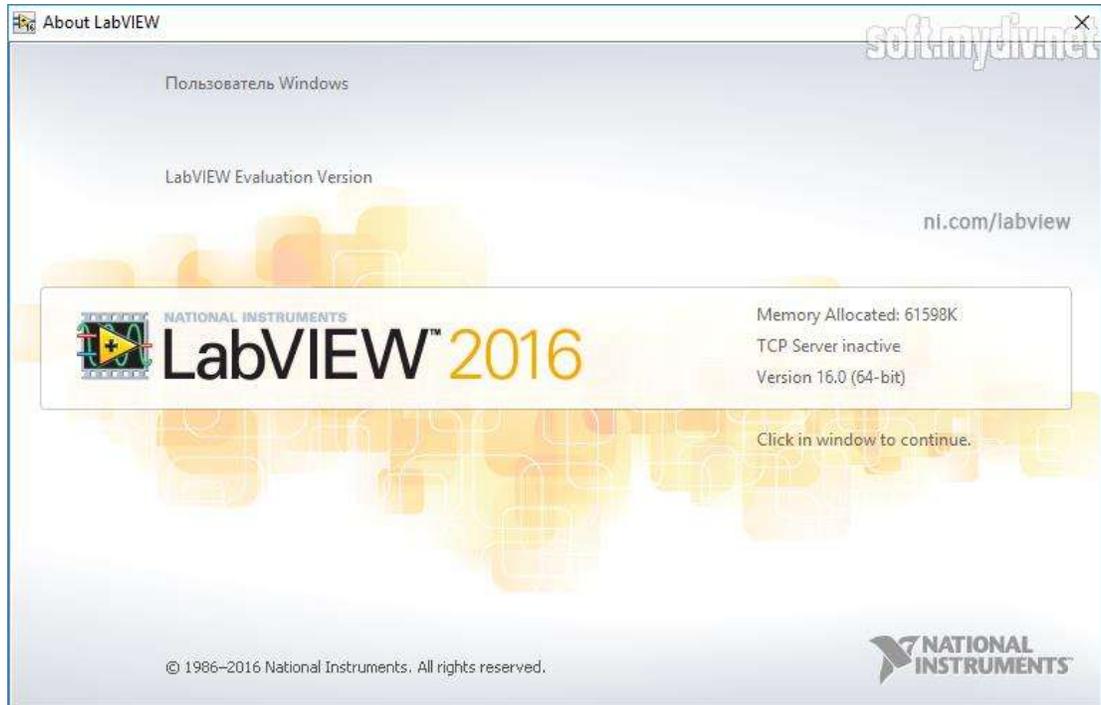
8. Esperar que se instale esto puede tardar varios minutos



9. Listo si queremos abrir el programa de LabVIEW seleccionamos la casilla si no simplemente clic en *Finish*



10. Al abrir el LabVIEW revisamos en *about* LabVIEW y podemos ver que nuestro producto está listo para su uso.



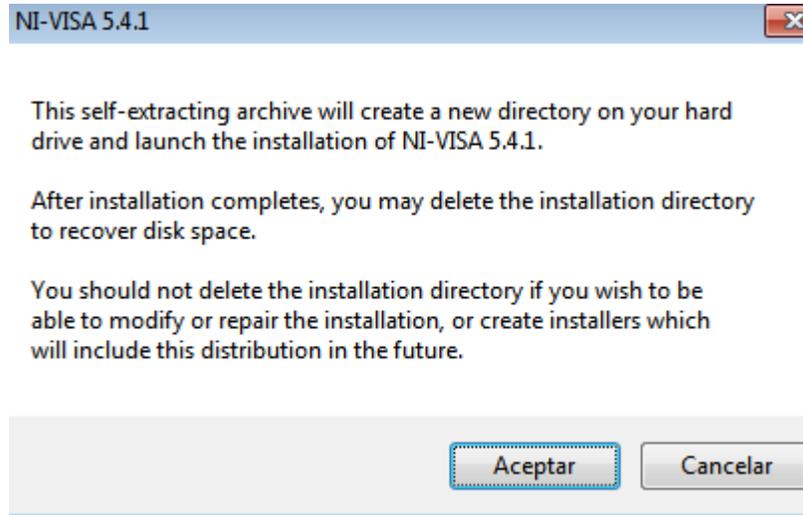
### 3.4.2 Instalación de paquete NI VISA.

También requerimos el paquete VISA de NI

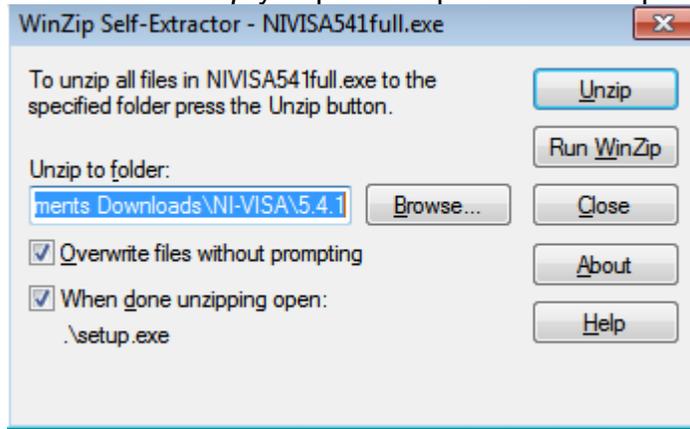
¿Qué es VISA?

*La arquitectura de software de instrumento virtual (VISA) es un estándar para configurar, programar y solucionar problemas de sistemas de instrumentación que comprenden interfaces GPIB, VXI, PXI, serie, Ethernet y / o USB. VISA proporciona la interfaz de programación entre el hardware y los entornos de desarrollo como LabVIEW, LabWindows / CVI y Measurement Studio para Microsoft Visual Studio. NI-VISA es la implementación de National Instruments del estándar VISA I / O. NI-VISA incluye bibliotecas de software, utilidades interactivas como NI I / O Trace y VISA Interactive Control, y programas de configuración a través de Measurement & Automation Explorer para todas sus necesidades de desarrollo. NI-VISA es estándar en toda la línea de productos de National Instruments. [16] Así que muestro la serie de pasos para su instalación:*

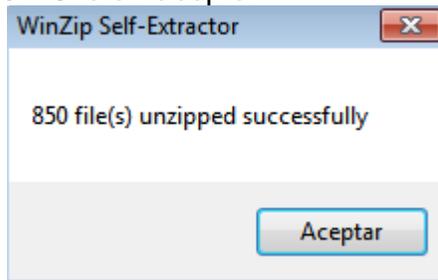
1. Descargar el paquete desde la página oficial de NI en el siguiente link <http://www.ni.com/download/ni-visa-17.0/6646/en/>
2. Una vez descargado en el ejecutable.exe dar clic derecho sobre este y en ejecutar como administrador y otorgar permisos.
3. Clic en aceptar,



4. Clic en *Un Zip* y esperar a que se descomprima



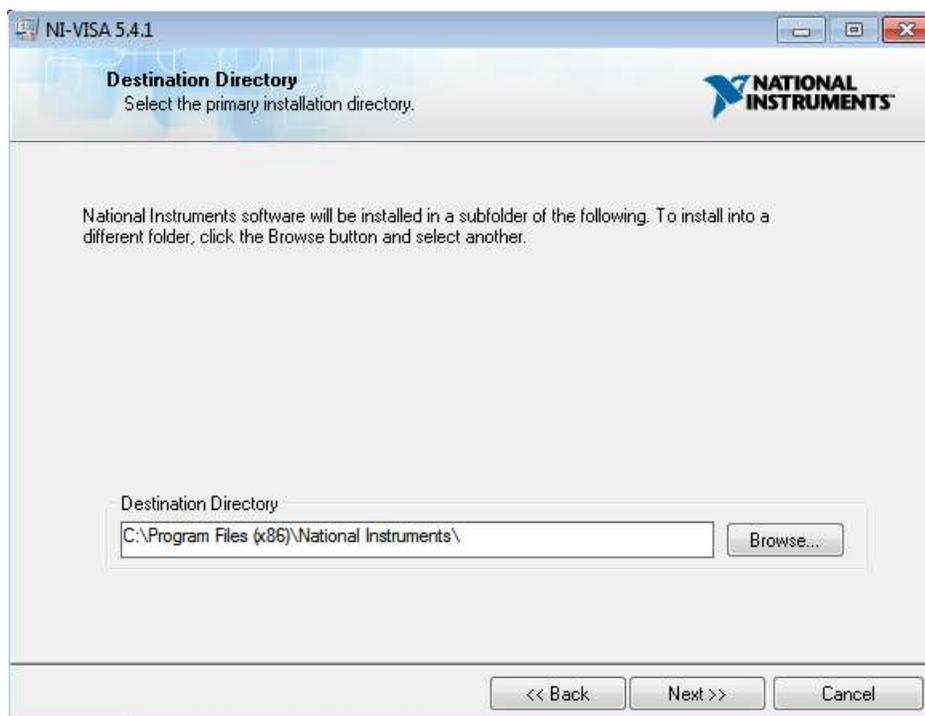
5. Clic en aceptar



6. En esta ventana dar clic en *Next >>*



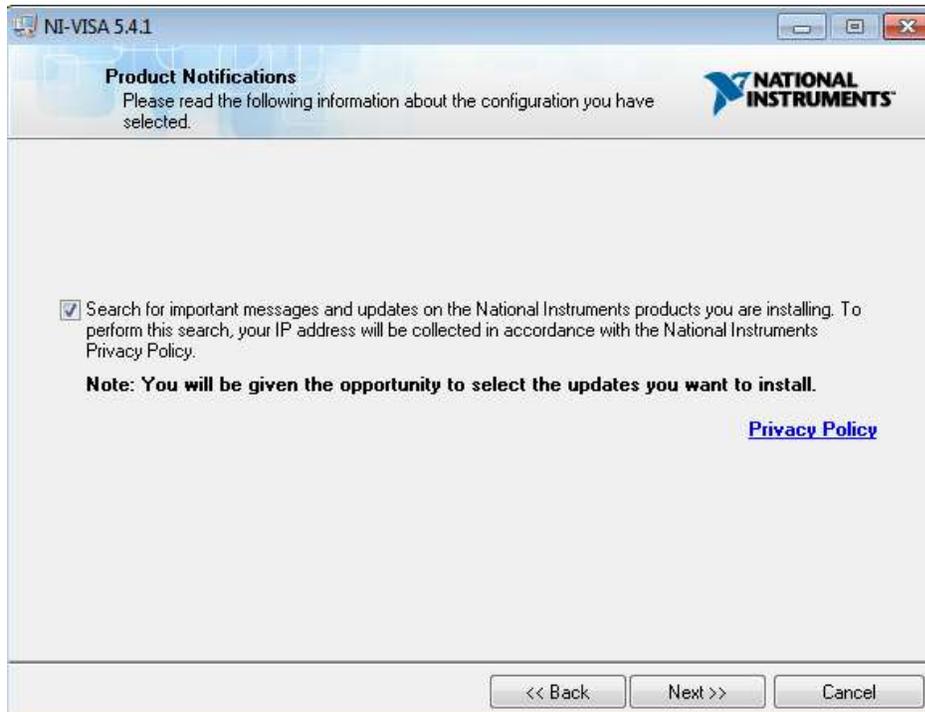
7. Seleccionar la carpeta donde instalar o dejarla preestablecida (recomendado) y clic en *Next>>*



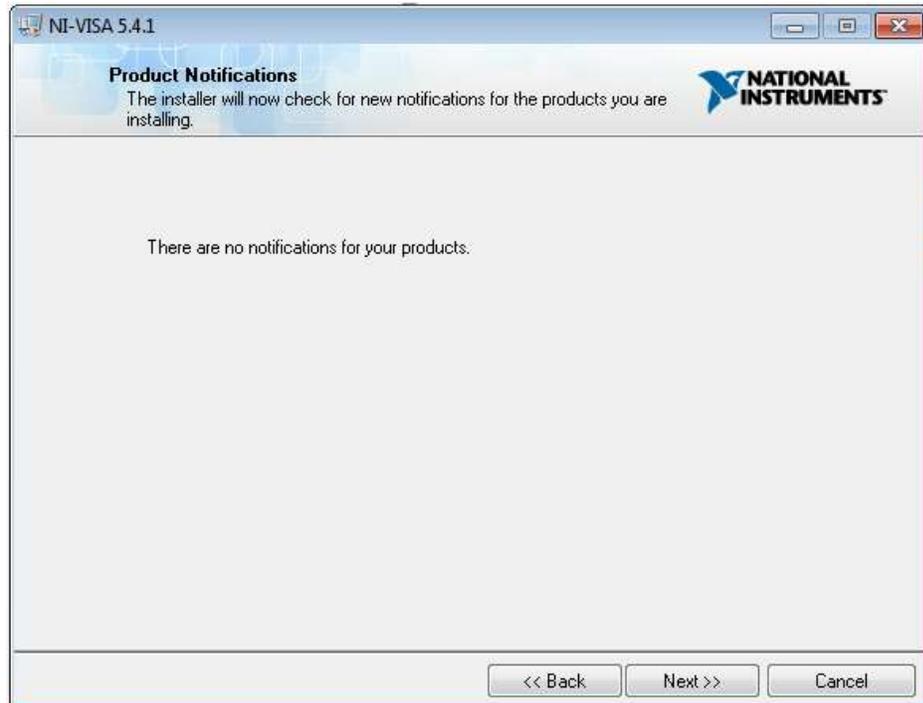
### 8. Después clic en *Next>>*



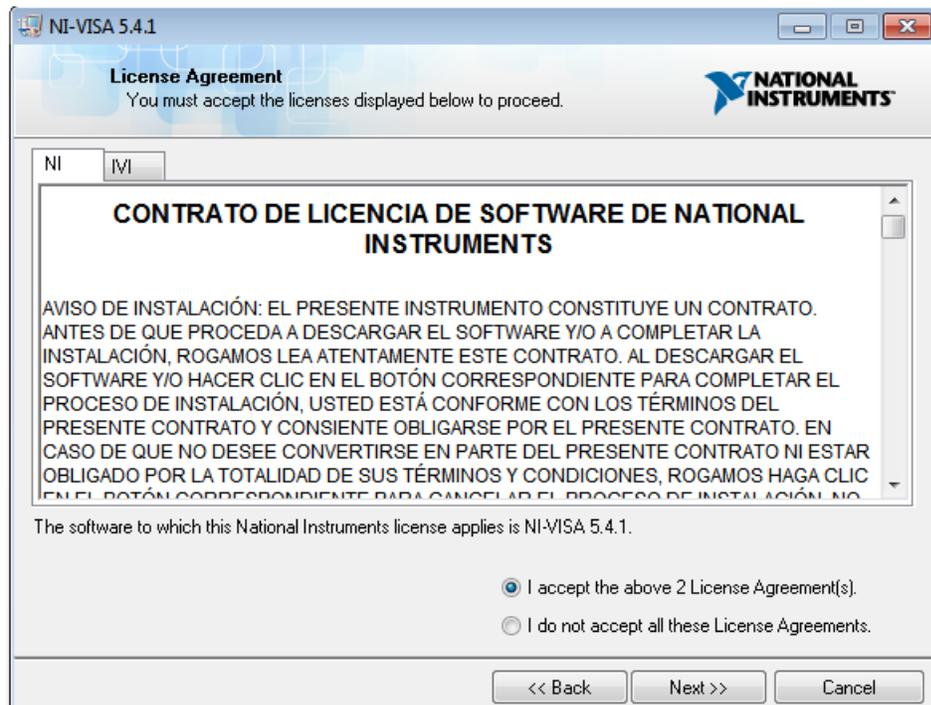
### 9. Nuevamente *Next>>*



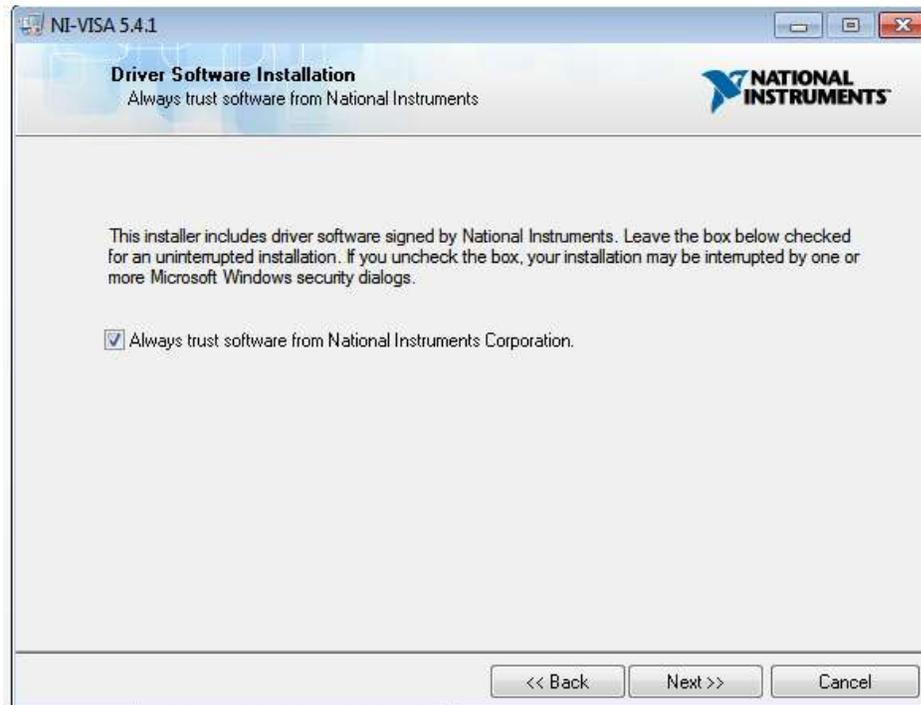
## 10. De nuevo *Next>>*



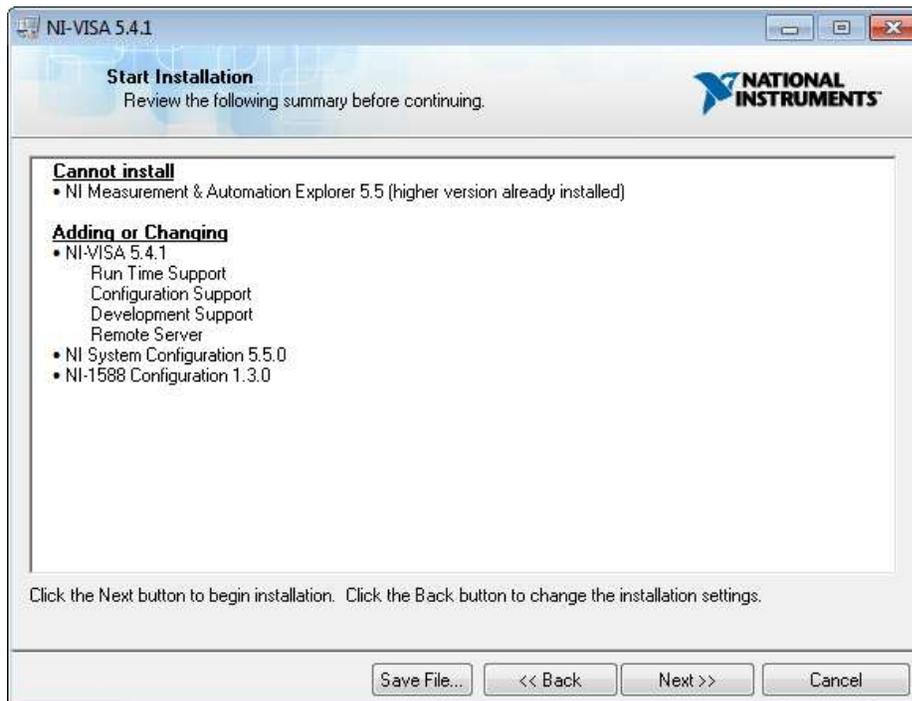
## 11. Aceptar términos y condiciones y clic en *Next>>*



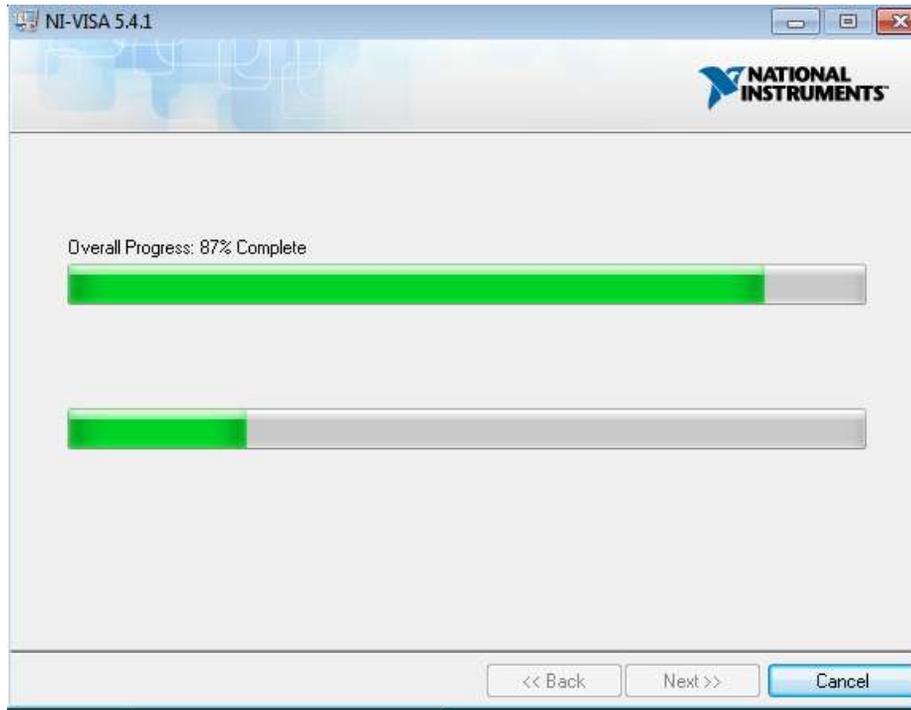
## 12. Clic en *Next>>*



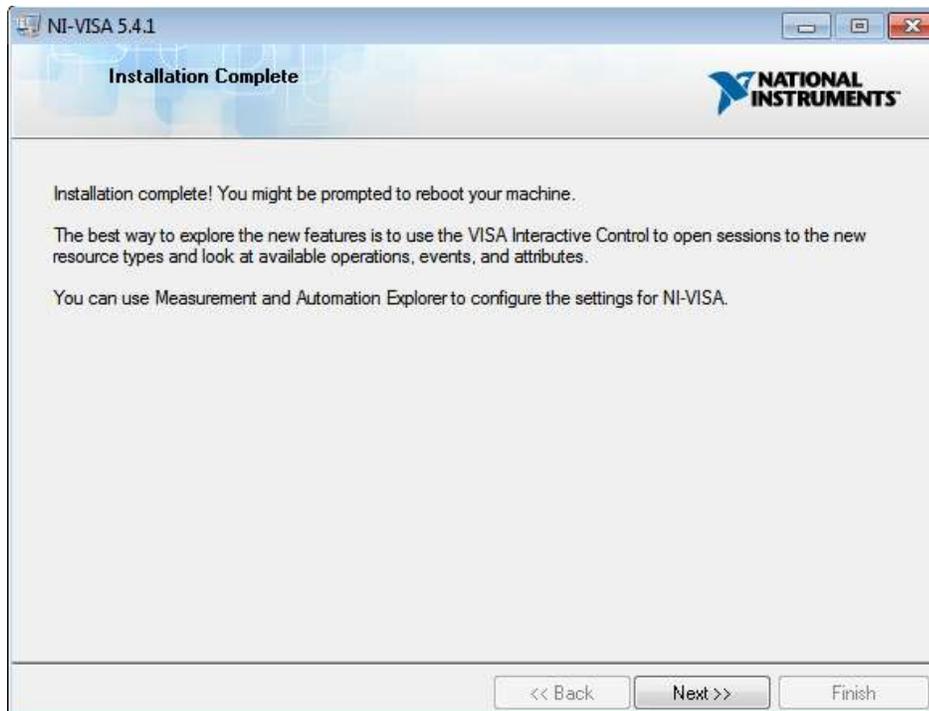
## 13. Clic en *Next>>*



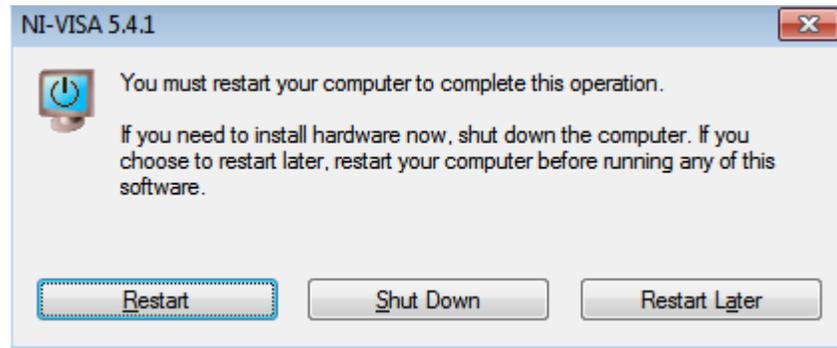
## 14. Inicia el proceso de instalación



## 15. Al terminar la instalación clic en *Next>>*



16. Y finalmente clic en *Restart* quedando completa la instalación.



### 3.4.3 Instalación de paquete *VI Package Manager*

También es necesario contar con el paquete *VI Package Manager*, para la instalación de este seguimos los siguientes pasos:

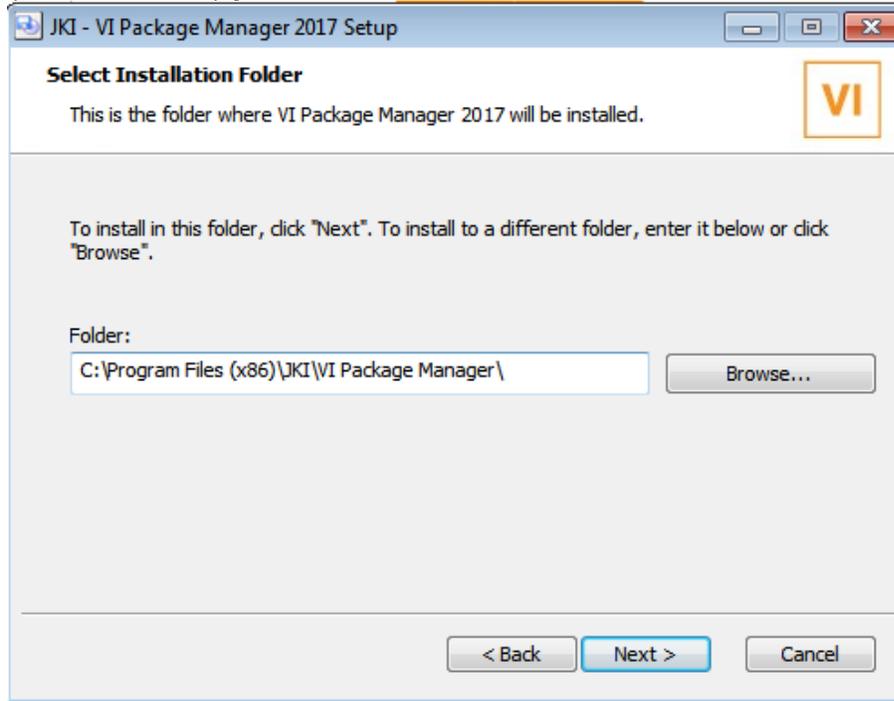
1. Descarga el paquete desde el link <https://vipm.jki.net/get> se selecciona el sistema operativo, se coloca un correo y se dan clic en descargar.
2. En el archivo ejecutable que se descarga se da clic con botón derecho y en ejecutar como administrador.
3. Se abre el asistente de configuración en el cual damos en el botón *Next>*



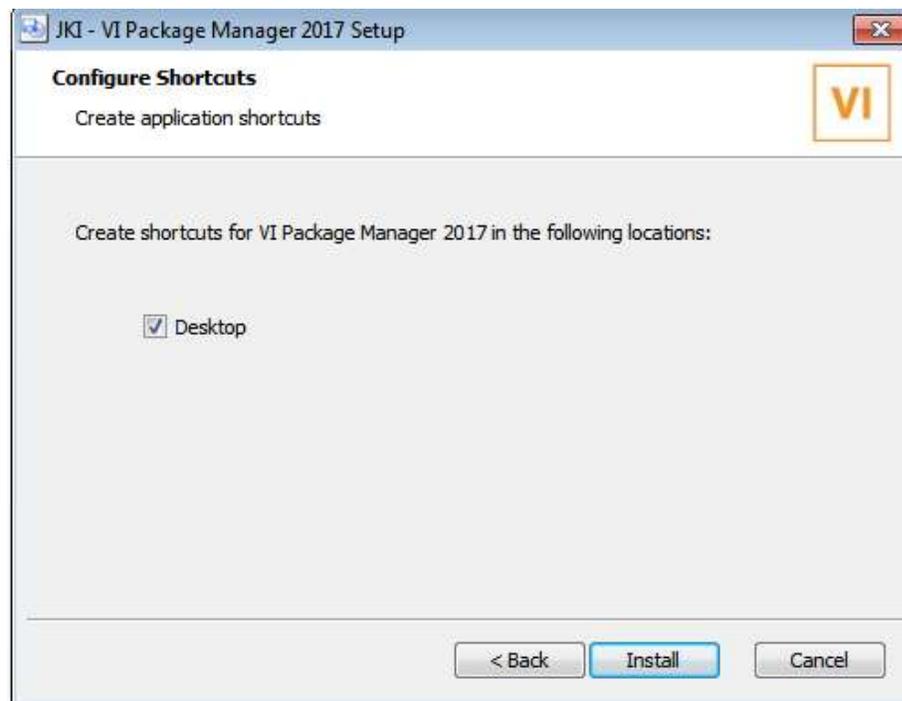
4. Aceptar términos y condiciones y dar clic en Next>



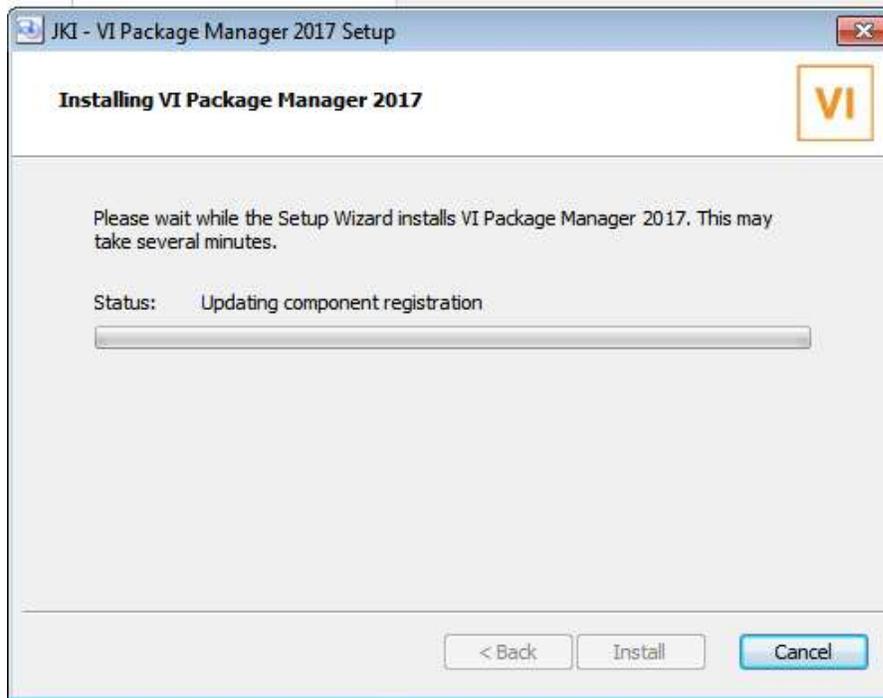
5. Seleccionar carpeta para la instalación o dejar la predeterminada (recomendado) y clic en Next>



6. Si se desea crear acceso directo en el escritorio seleccionar y dar clic en *Instaló*.



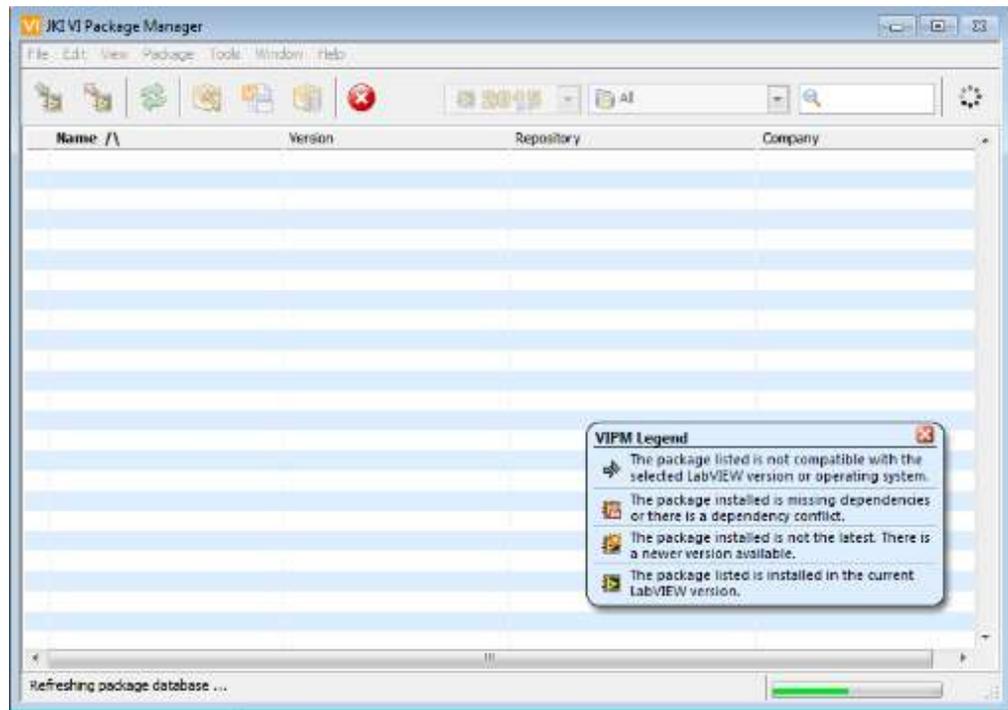
## 7. Esperar a que se instale



## 8. Al terminar de instalar sólo dar clic en *Finish* y se abrirá el programa.



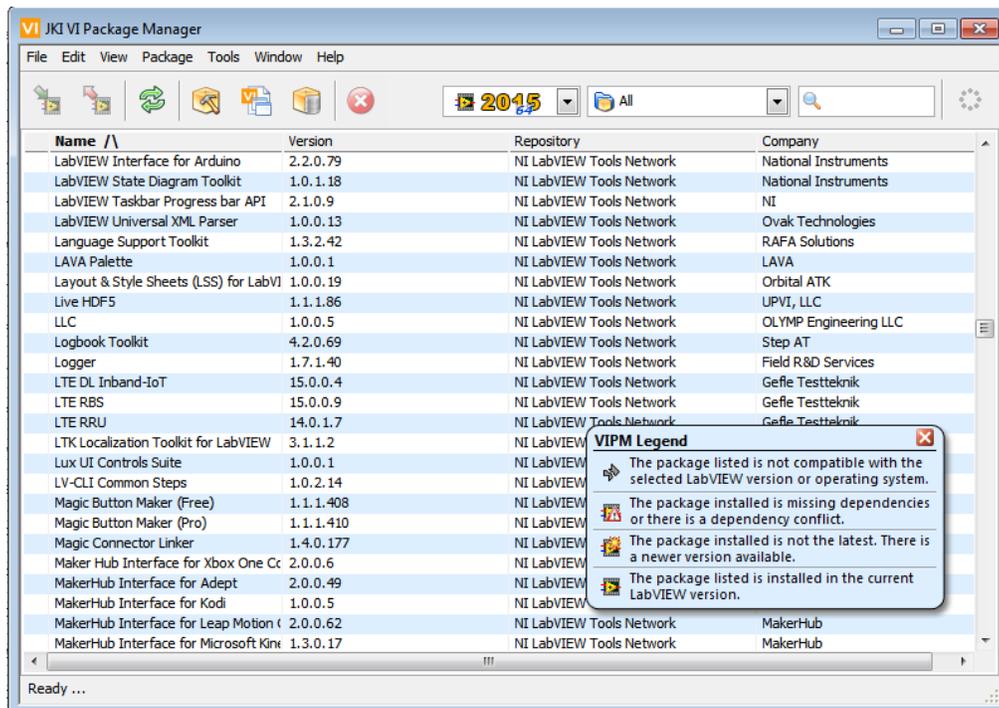
## 9. El programa cargará todos los datos



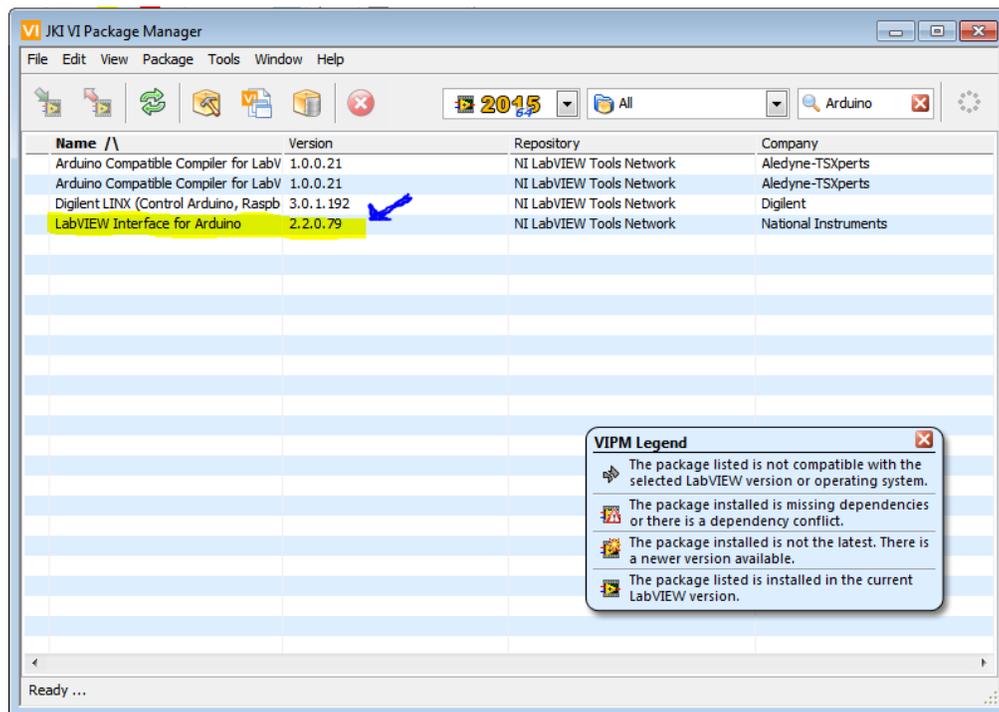
### 3.4.4 Instalacion de la interfaz gráfica de LabVIEW con microcontrolador arduino (LIFA).

A continuación, expongo los pasos para instalar este paquete en el LabVIEW ya que este contiene los bloques para el desarrollo de la interfaz gráfica a realizar, cabe hacer mención que para instalar la interface de LabVIEW para arduino se hace uso del programa anterior VI *Package Manager*.

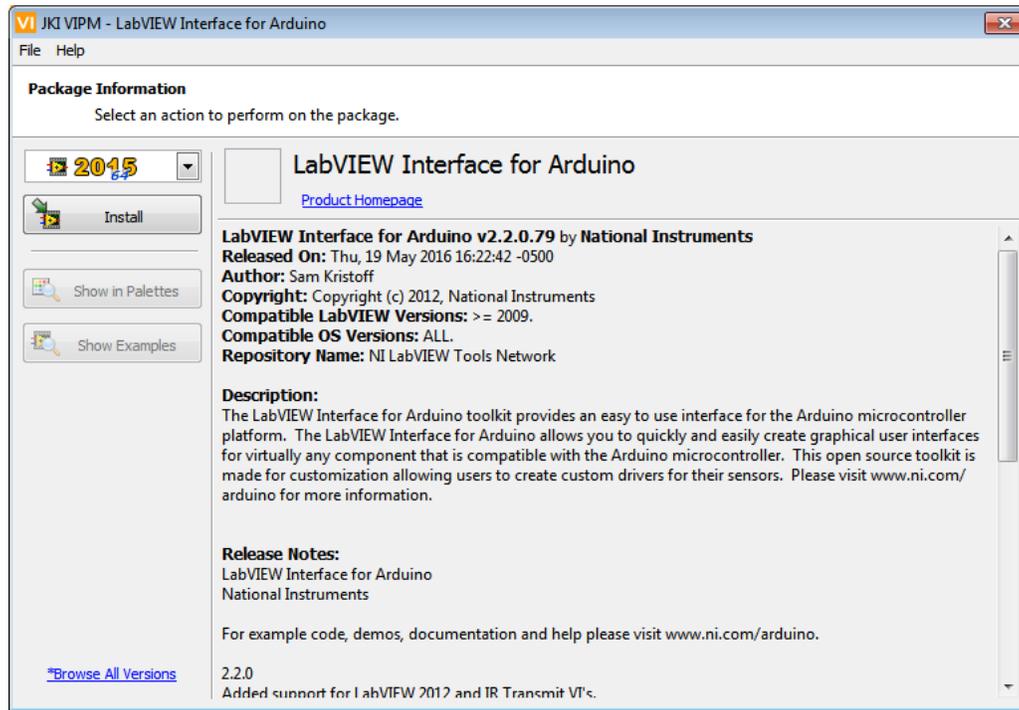
## 1. Primero se abre el Programa VI Package Manager



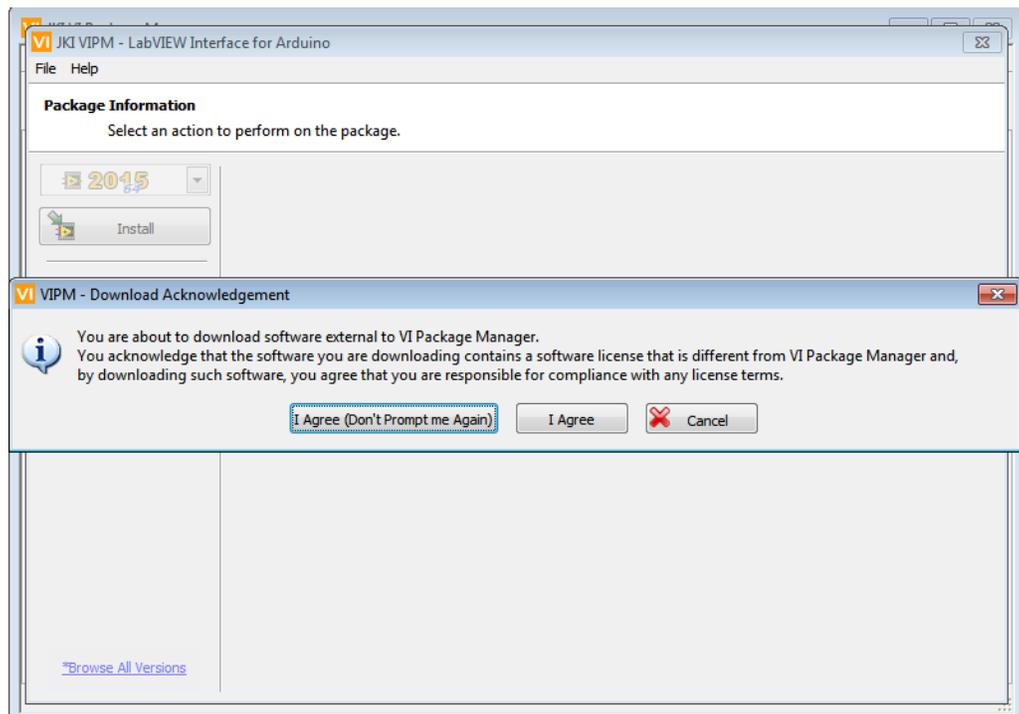
## 2. Después en el botón de buscar se busca la palabra Arduino y se da clic en donde aparece LabVIEW interface para arduino.



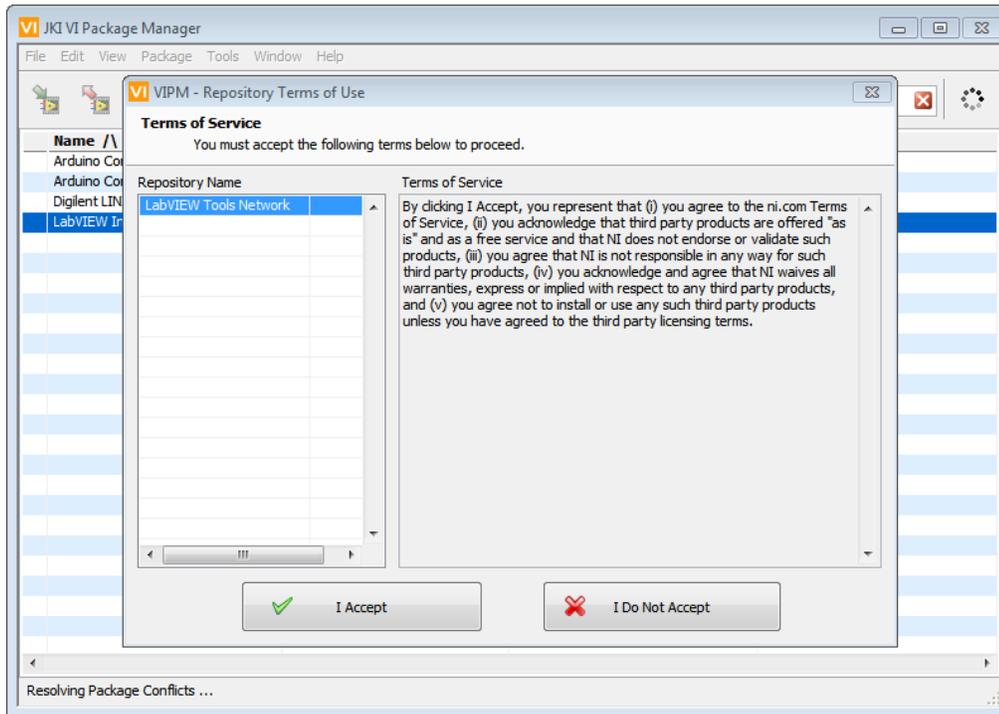
### 3. Dar clic en *install*



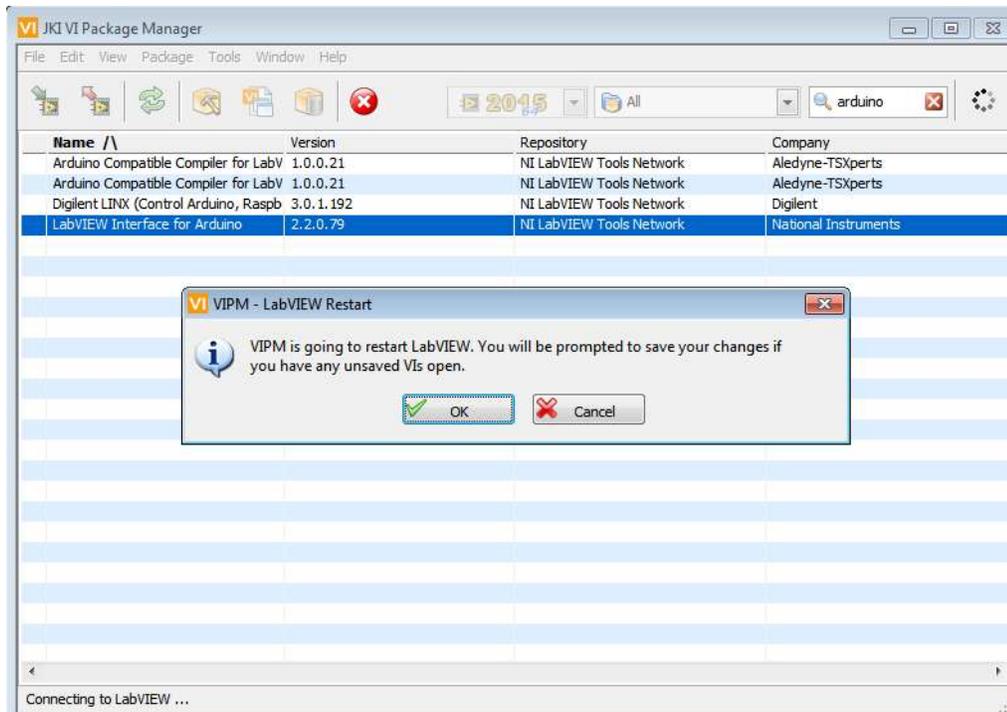
### 4. En la siguiente ventana dar en *I Agree*



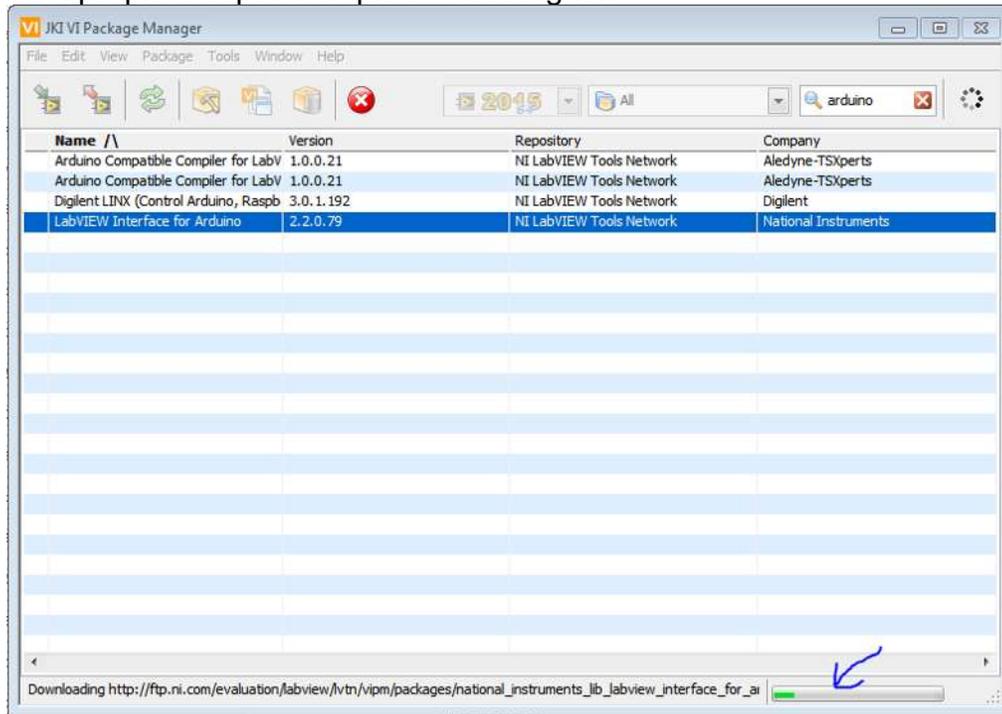
5. Después dar clic en *I Accept*.



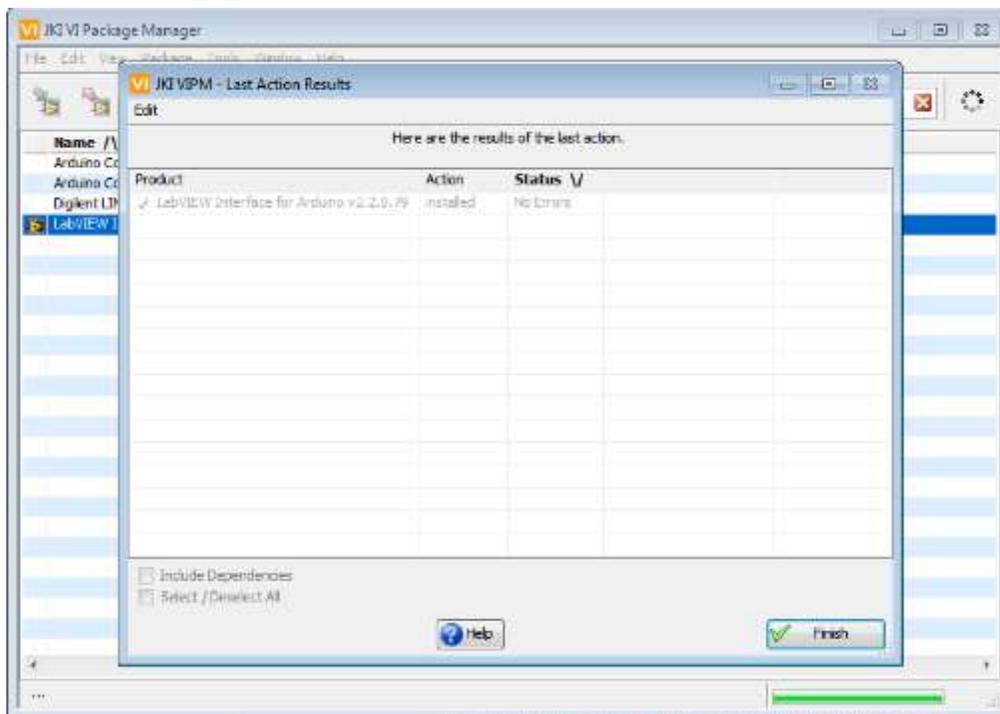
6. Clic en ok



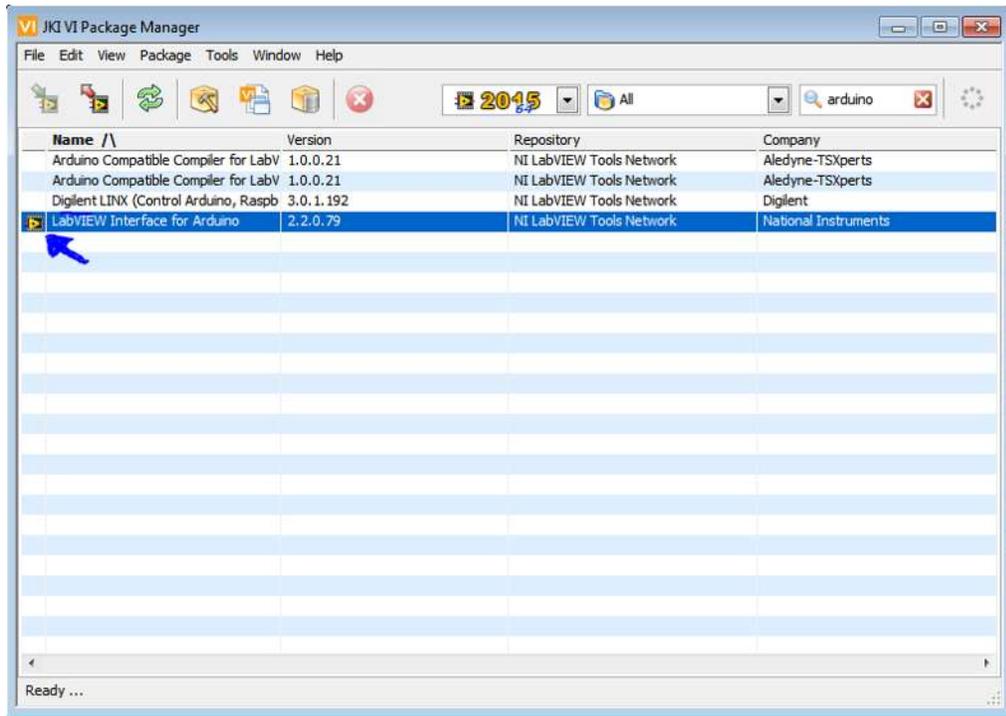
7. En la parte de abajo se puede ver como se están descargando los archivos del paquete esperar a que se descarguen



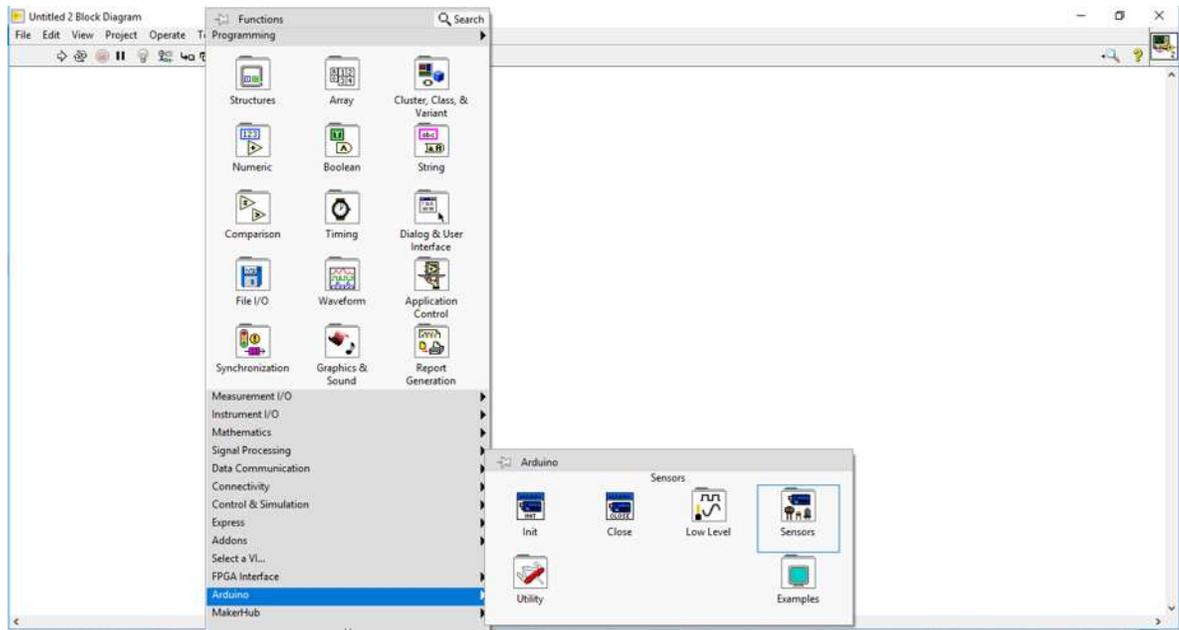
8. Una vez que termina solo dar clic en *Finish*



### 9. Como se puede ver ya aparece instalado el paquete en LabVIEW



### 10. Finalmente podemos ir a LabVIEW y comprobamos que ya contamos con las herramientas de arduino

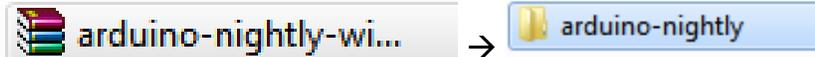


### 3.4.5 Instalacion del ambiente de programación IDE de Arduino.

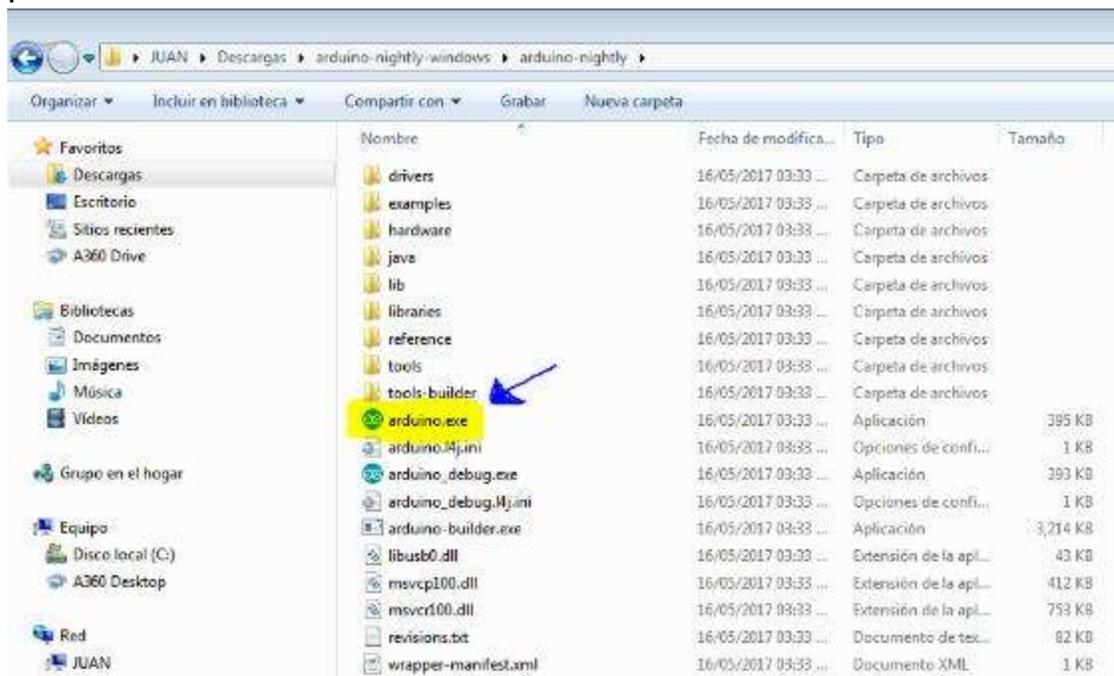
Para instalar el ambiente de programación IDE de Arduino se requiere una sencilla serie de pasos que muestro a continuación:

Descargar desde la página oficial de Arduino en este link <https://www.arduino.cc/en/Main/Software>

1. Después hay que descomprimir el archivo.



2. Al abrir la carpeta de archivos, podemos observar que este archivo ya viene portable y, por lo tanto, ya no tenemos que realizar la instalación en el pc, simplemente cada vez que se dese usar el software de arduino solo hay que ejecutar como administrador el archivo .exe

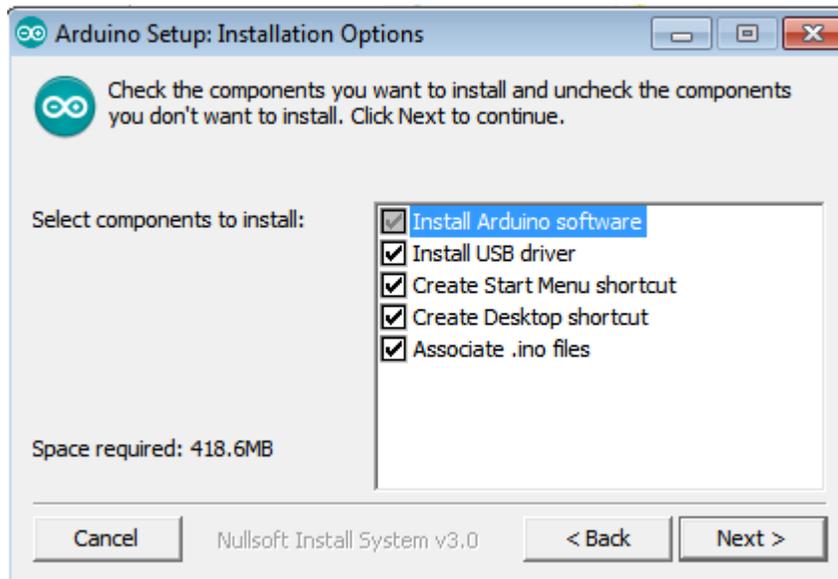


3. En el caso de que el archivo no sea portable se debe dar clic derecho en el archivo.exe y ejecutar como administrador dar permisos y se abrirá el asistente de instalación.

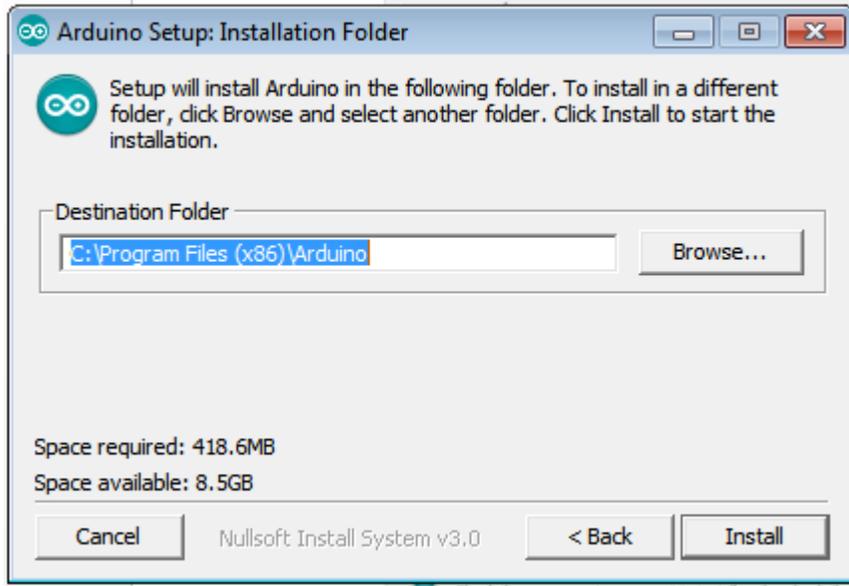
#### 4. Dar clic en *I Agree*



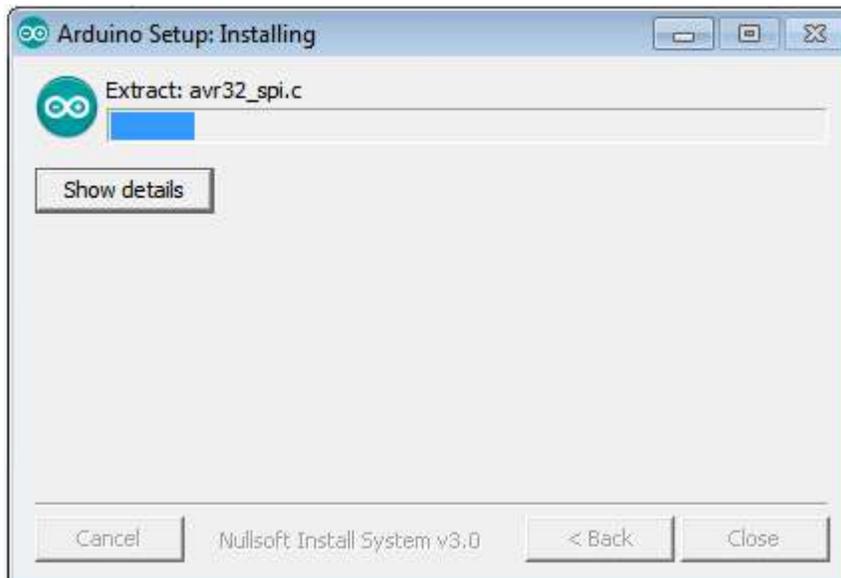
#### 5. Dar clic en Next >



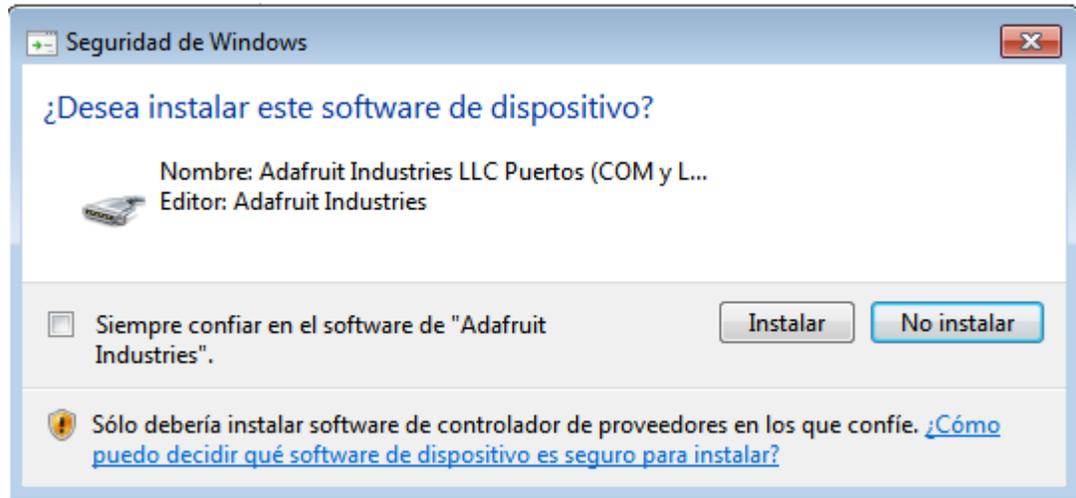
6. Dar clic en *Install*



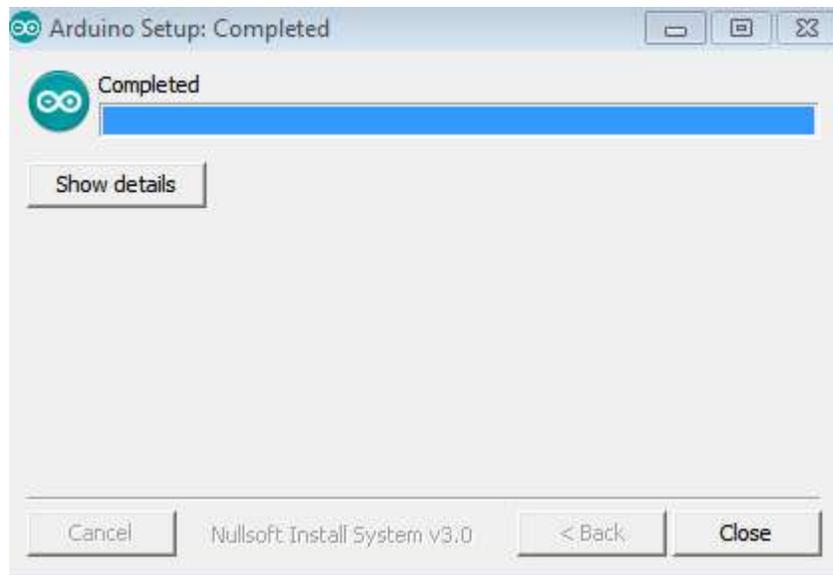
7. Esperar a que se instale



8. Dar clic en instalar



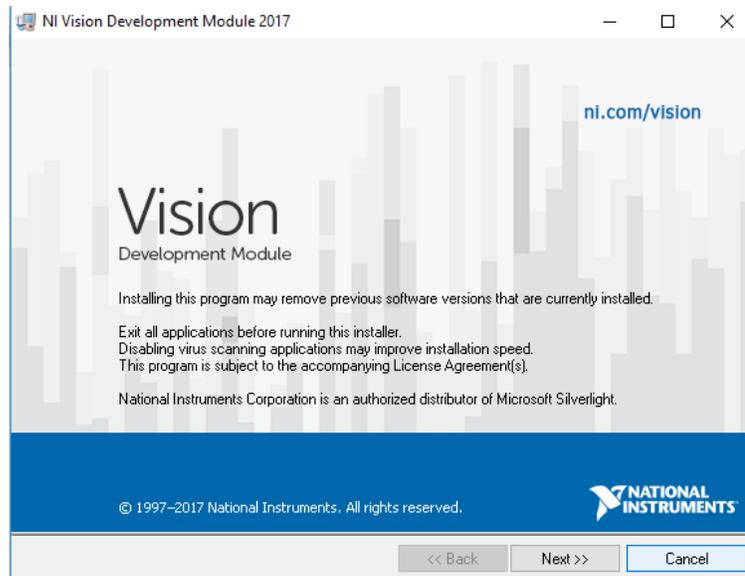
9. Al completarse la instalación simplemente dar clic en *close*



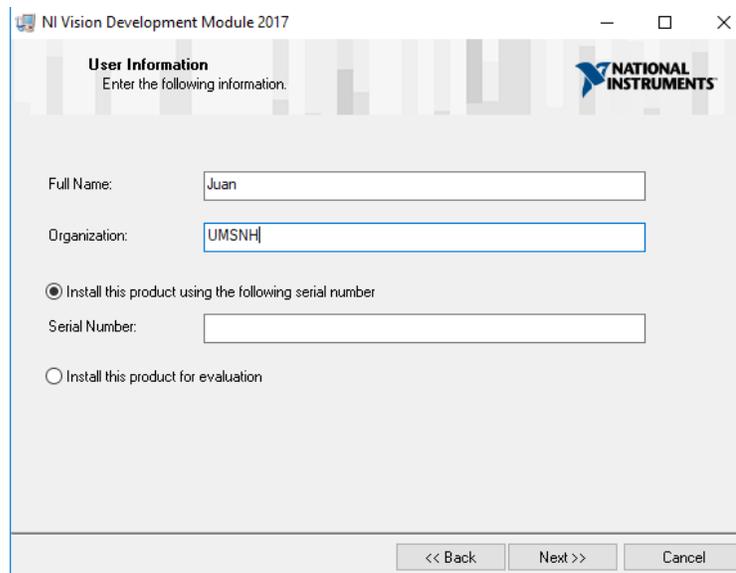
También se requiere instalar otros 2 paquetes para la adquisición de imágenes, por lo cual muestro como hacer la instalación a continuación:

### 3.4.6 Instalación de paquete de módulos de Desarrollo Visión.

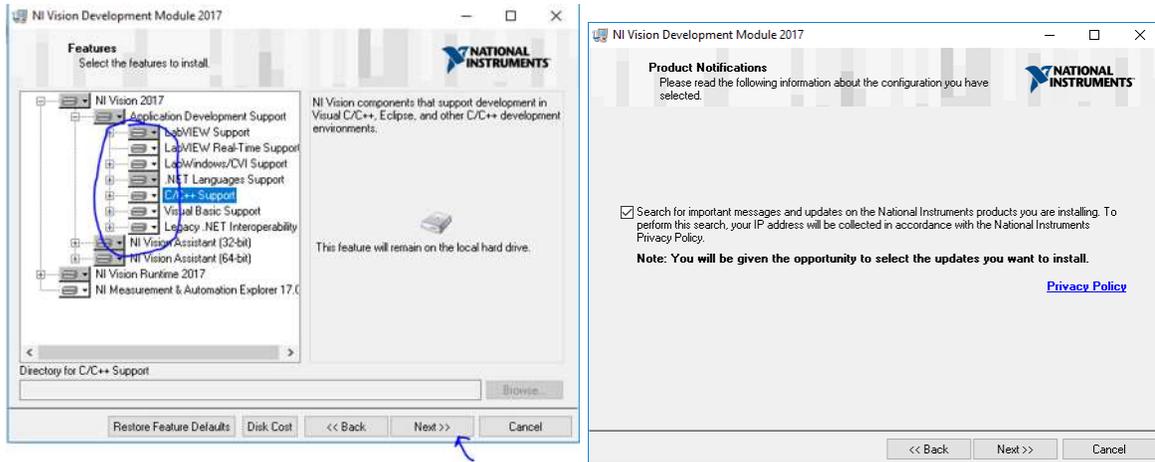
1. se descargan los paquetes una vez descargados se procede a extraer del archivo comprimido se ejecuta como administrador el archivo.exe y se abre el gestor de instalación que a continuación se muestra:



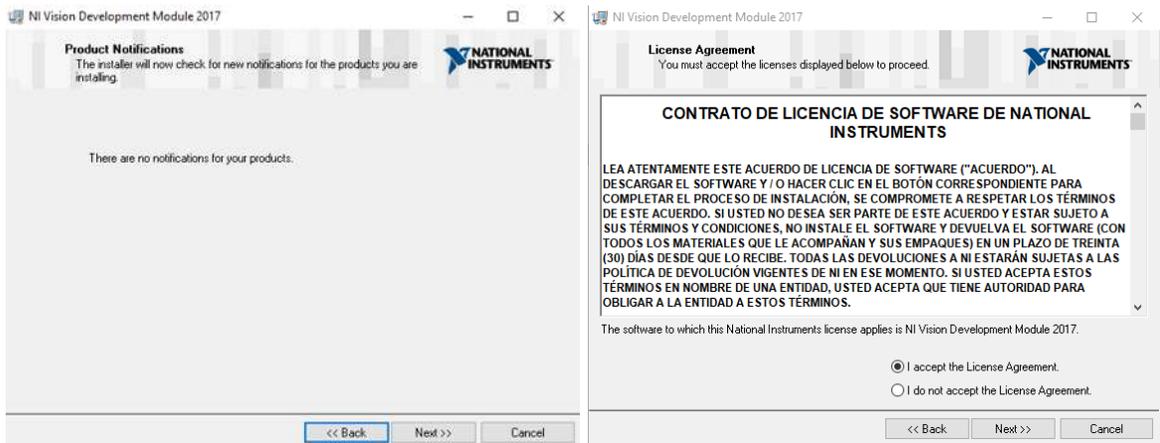
2. se da clic en *Next>>*, se colocan los datos de nombre de usuario y organización y el serial o si se desea solo instalar la versión de evaluación



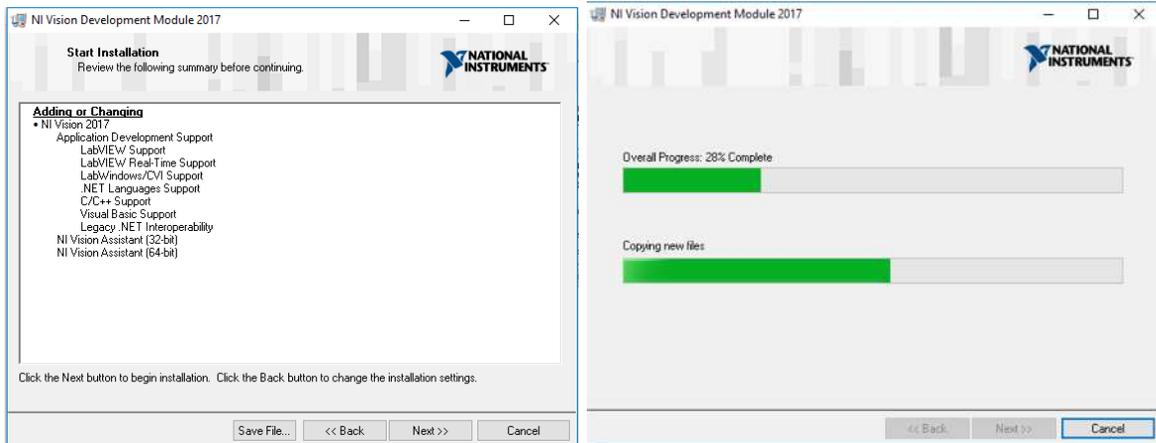
3 clic en *Next>>*, en la siguiente ventana verificar que todos los paquetes estén activados, nuevamente clic en *Next>>*.



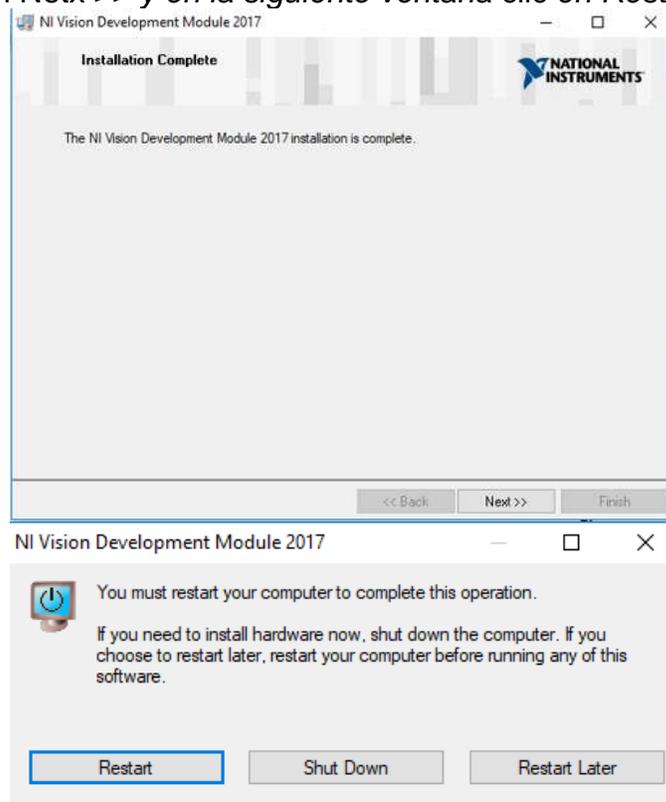
4. Clic en *Next>>*, y en la siguiente ventana Aceptar términos y condiciones y clic en *Next>>*



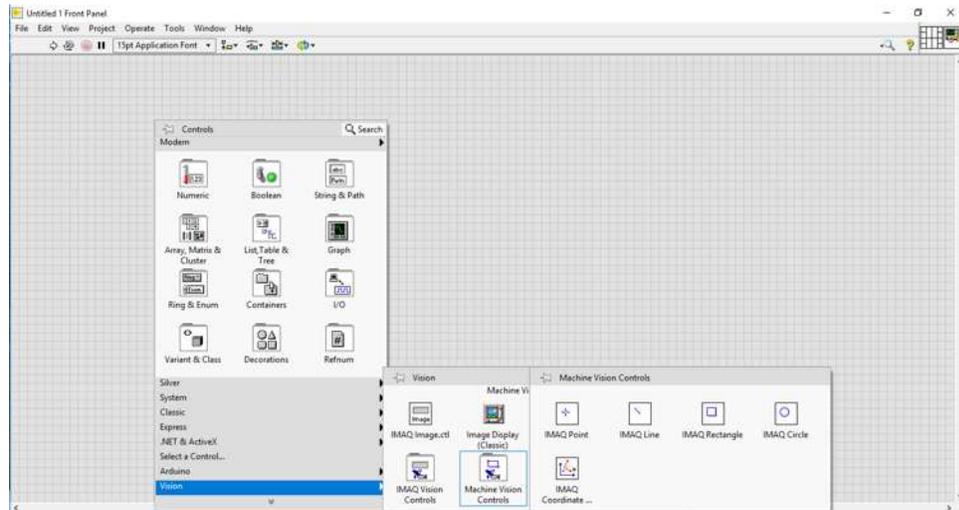
5. Clic en *Next>>*, y Inicia el proceso de instalación



6. Clic en *Next >>* y en la siguiente ventana clic en *Restart*

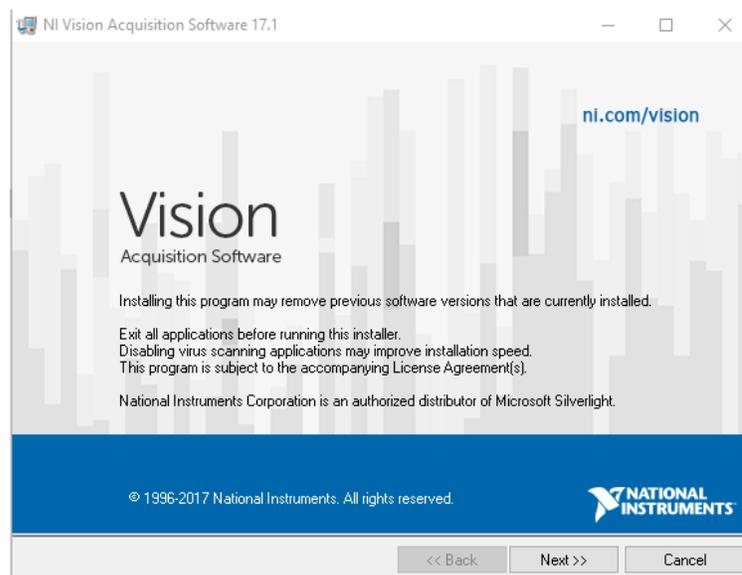


## 7. Finalmente, solo verificar que ya se cuenta con los módulos en LabVIEW

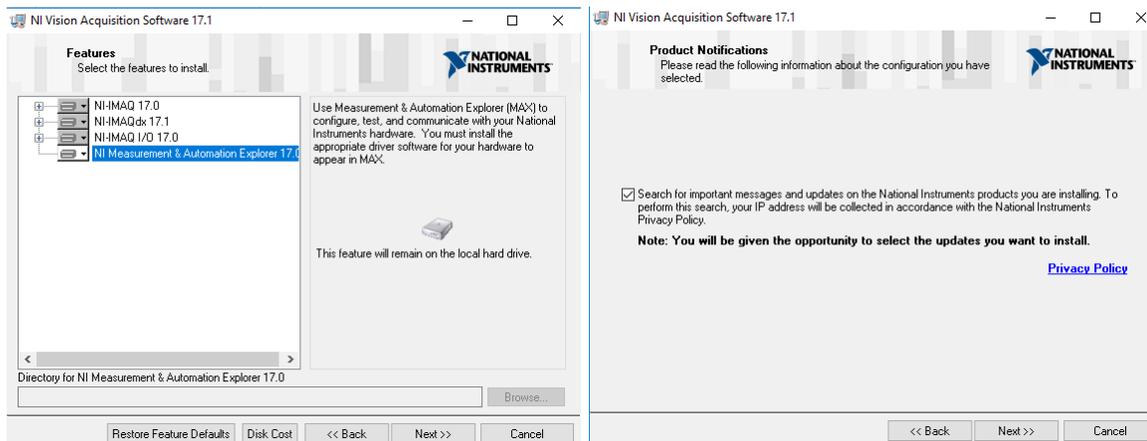


### 3.4.7. Instalación del Software de adquisición (video) Visión.

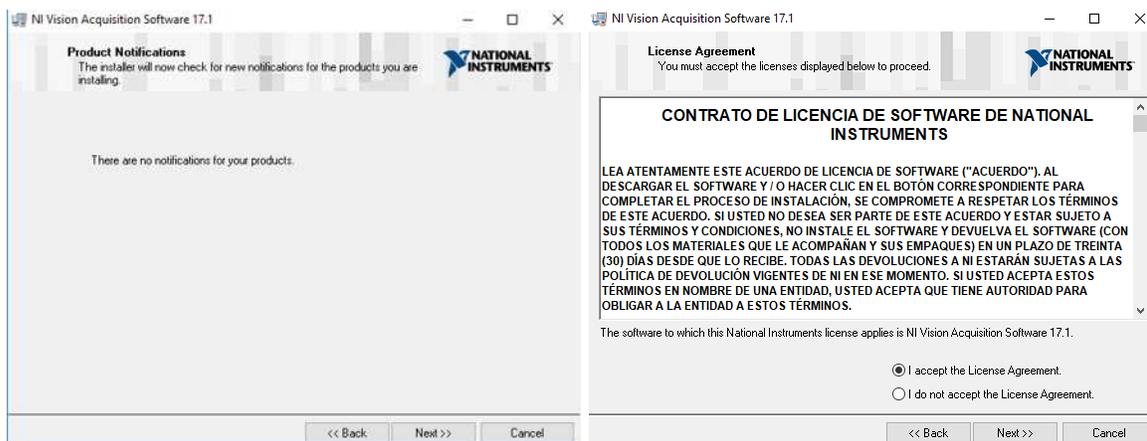
1. Se descargan los paquetes, una vez descargados se procede a extraer del archivo comprimido, se ejecuta como administrador el archivo.exe y se abre el gestor de instalación que se muestra en el paso 2.
2. Se da clic en *Next*.



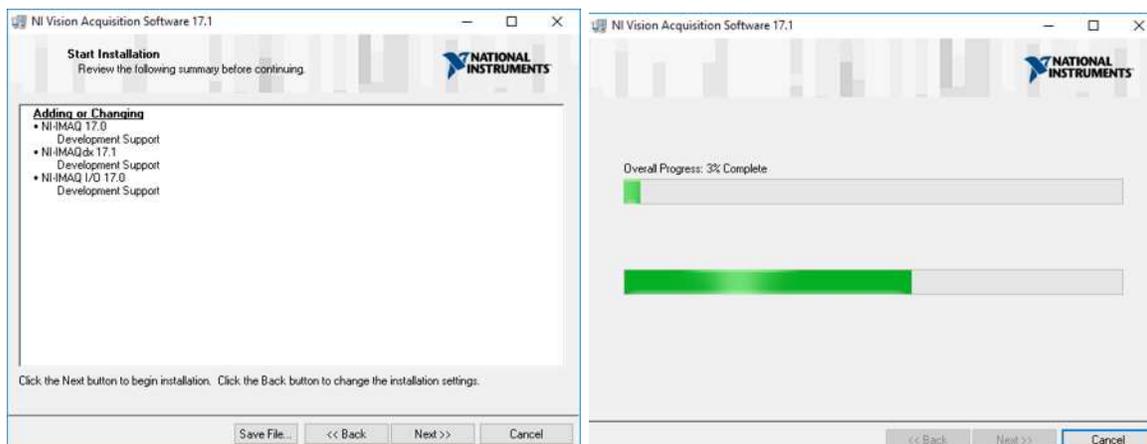
### 3. se activan los paquetes deseados(recomendado activar todos) clic en Next



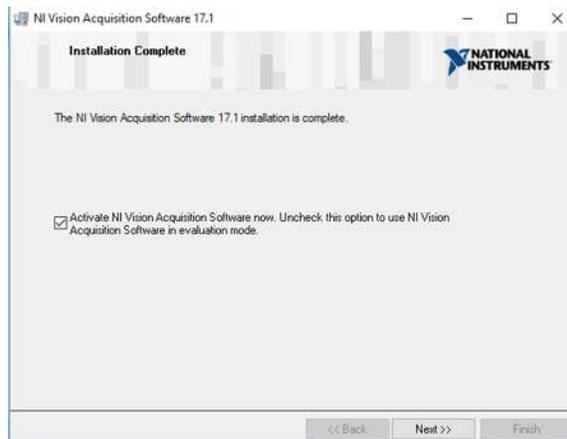
### 4. Clic en Next en la siguiente ventana aceptar terminos y condiciones y clic en Next.



### 5. Clic en Next>> , esperar a que se instale esto tardara un poco



6. Y finalmente clic en Next y listo quedo instalado.



### 3.5 Desarrollo de interfaz gráfica en LabVIEW

Para el desarrollo de esta interfaz se creó un diagrama de bloques del proceso completo de control el cual se lo podemos observar en la Figura 20 Diagrama de bloques de proceso de control, en este diagrama se puede observar a grande escala el proceso de control completo; el cual incluye el bloque de la interfaz gráfica a desarrollarse, un bloque para adquisición de datos, uno de actuadores, el de la planta y el bloque de sensores de retroalimentación.

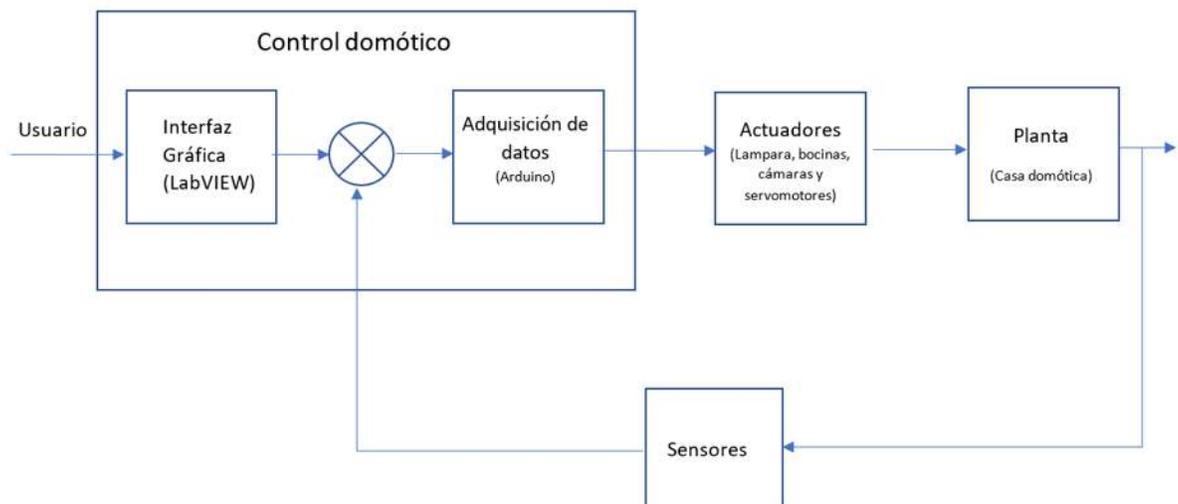


Figura 20 Diagrama de bloques de proceso de control

De este sistema desarrollaremos de manera específica el bloque de la interfaz gráfica tal como se muestra en la Figura 21 Diagrama de bloques de la interfaz gráfica, ahora bien, para el bloque de adquisición de datos se utilizará la tarjeta electrónica Arduino Mega y para los otros dos bloques solo se harán propuestas de

los actuadores y sensores que se pueden utilizar en la aplicación real, estas propuestas se darán al final del capítulo 4 en el punto 4.4.

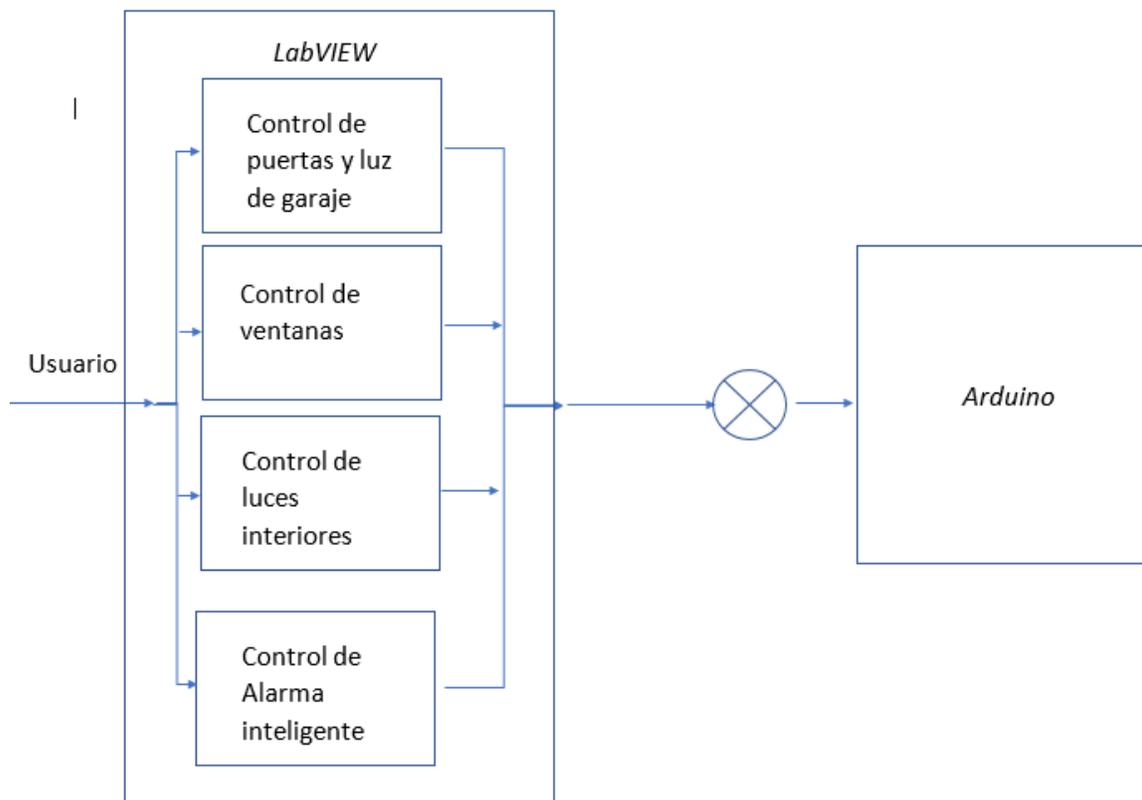


Figura 21 Diagrama de bloques de la interfaz gráfica

El bloque de la interfaz grafica se desarrolla en las cuatro secciones que muestra el diagrama de bloques de la Figura 21 Diagrama de bloques de la interfaz gráfica; y los cuales se explicara su diseño, uso y desarrollo de manera mas especifica en la sección 3.5.2.

### 3.5.1 Carga del LIFA-Base a la placa arduino Mega 2560

Una vez ya teniendo todos los programas anteriores instalados vamos a la liga

C:\Program Files\National Instruments\LabVIEW 2016\vi.lib\LabVIEW Interface for Arduino\Firmware\LIFA\_Base y se busca el archivo



LIFA\_Base.ino

Se abre y ya tenemos el programa este se carga a la tarjeta electrónica Arduino MEGA (también se puede descargar desde la página oficial de Arduino).

En la **Figura 22 Programa subido a la placa Arduino Mega 2560** tenemos 3 aspectos importantes que resaltar el primero en la parte superior derecha, tenemos que dar clic en la **flecha** para que el programa se compile y se cargue en la tarjeta electrónica, y para ello el IDE debe reconocer la tarjeta electrónica (parte inferior derecha), es decir el tipo de tarjeta, en este caso **arduino mega 2560** y en el puerto al que está conectada, en este caso el **puerto com4**, si esto no aparece hay que configurarlo desde el menú herramientas como se muestra en la **Figura 23 Configuración de placa Arduino Mega 2560 en puerto Com 4** y el tercer aspecto es que hay que verificar que se haya subido correctamente, es decir debe aparecer la palabra **subido** en la parte inferior izquierda.



Figura 22 Programa subido a la placa Arduino Mega 2560

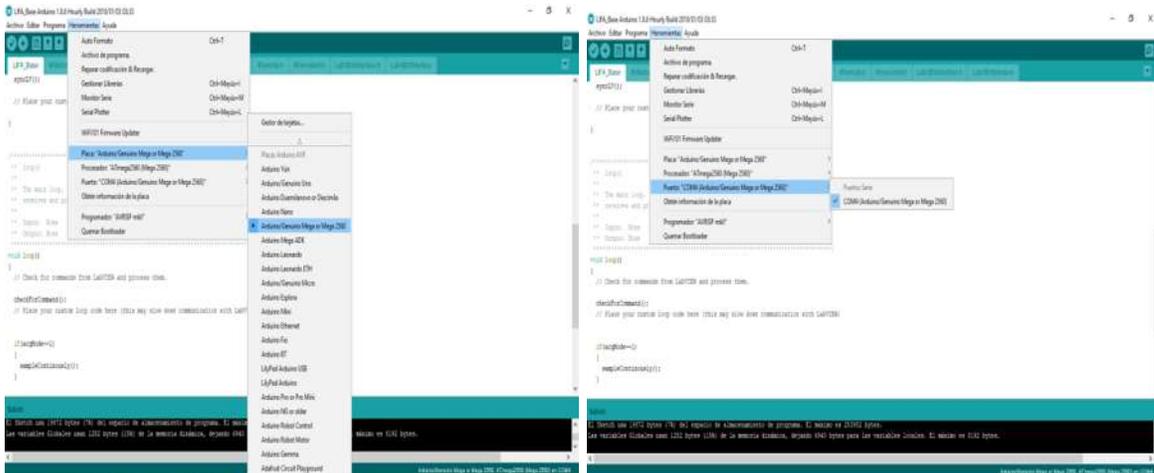


Figura 23 Configuración de placa Arduino Mega 2560 en puerto Com 4

### 3.5.2 Desarrollo y Programación de la Interfaz Gráfica en LabVIEW

En los **apéndices A y B** se muestra las conexiones físicas que se llevaron a cabo en la maqueta, en la **Tabla 5 Conexiones físicas en el bus de datos del Baño** y **Tabla 6 Conexiones físicas en el bus de datos del garaje**.

A continuación, muestro la serie de pasos que se siguió para el diseño de la interfaz,

1. Se creó un nuevo proyecto

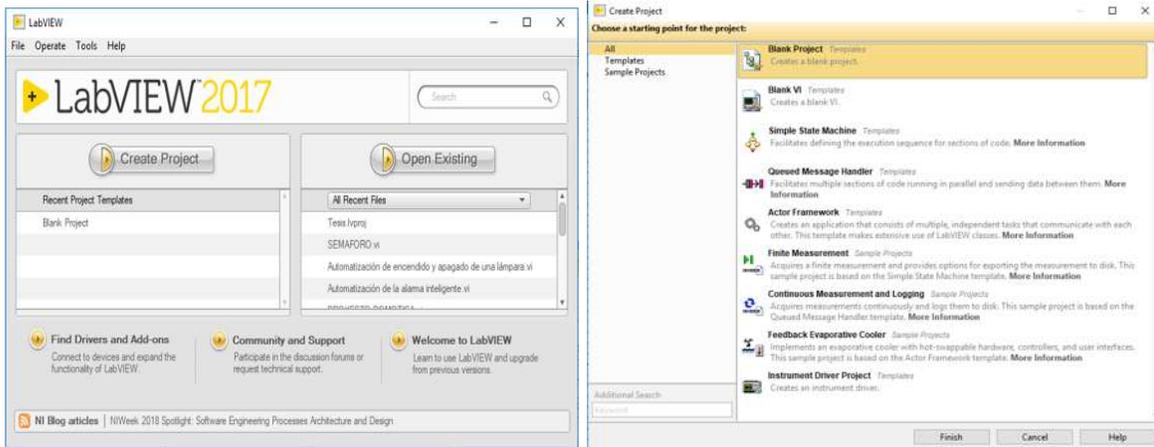


Figura 24 Creación del proyecto

2. Una vez creado el proyecto se guardó con el nombre deseado y se creó un nuevo VI (instrumento Virtual).

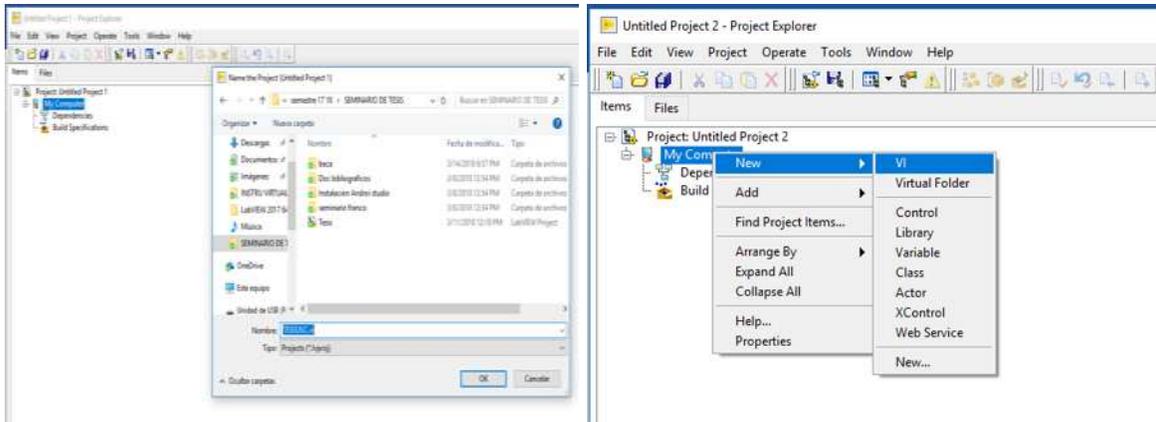


Figura 25 Almacenado del proyecto y Creación del VI

### 3. De igual manera se guardó el VI con el nombre deseado

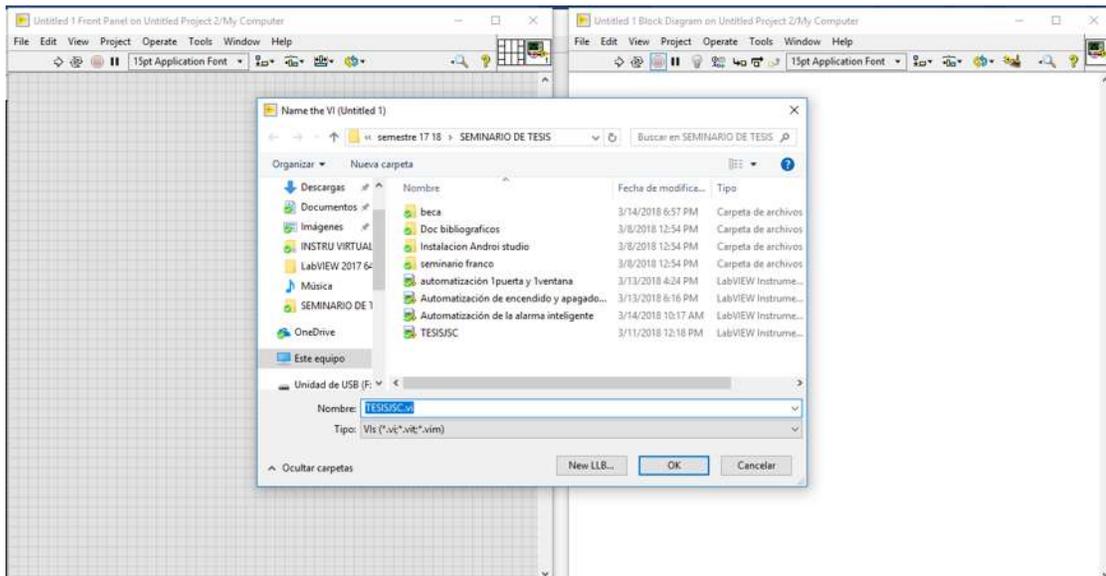


Figura 26 Almacenado del VI

#### 4. Y así se tiene listo el VI para programar la interfaz gráfica.

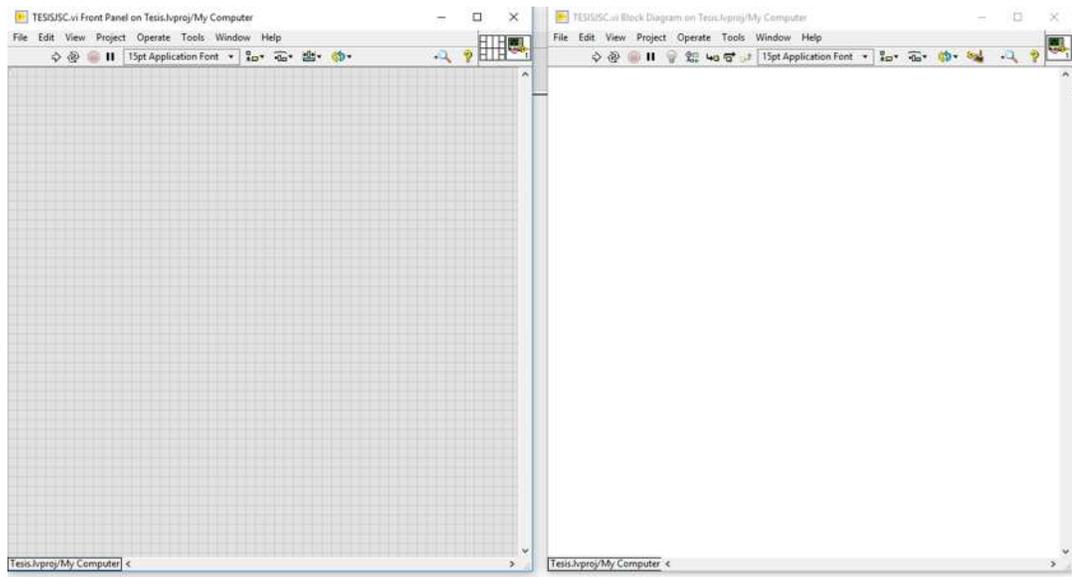


Figura 27 VI para programar la Interfaz Gráfica

La programación no se detalló paso a paso ya que hacerlo de esta manera se ocuparía mucho espacio; Por lo tanto, solo presento los avances de la automatización de un elemento de cada tipo, es decir de 1 puerta y 1 ventana (dos servos), 1 luz (lámpara) con sensor de movimiento (sensor tipo PIR) y la alarma de seguridad (cámara y sensor y luces) y finalmente presento la programación final donde se hace la unión de todas las programaciones anteriores para cubrir el alcancé del proyecto de tesis.

##### 3.5.2.1 Automatización de una puerta y una ventana.

Para la automatización de apertura y cierre de **1 puerta y 1 ventana** se realizó la programación mostrada en la **Figura 28 Programación para control de 2 servos 1 Puerta y 1 Ventana (Panel frontal y Diagrama de bloques respectivamente)**.

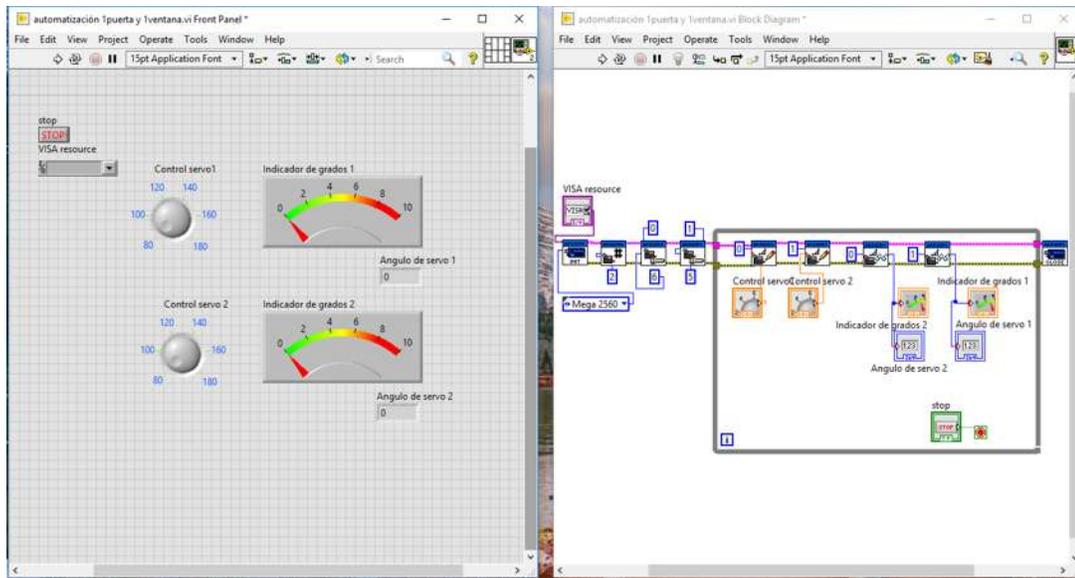


Figura 28 Programación para control de 2 servos 1 Puerta y 1 Ventana (Panel frontal y Diagrama de bloques respectivamente).

### 3.5.2.2 Automatización de encendido y apagado de una lámpara.

Para la automatización de encendido y apagado de una **lámpara** se realizó la programación mostrada en la **Figura 29 Programación para control de 1 lámpara con sensor**, (Panel frontal y Diagrama de bloques respectivamente):

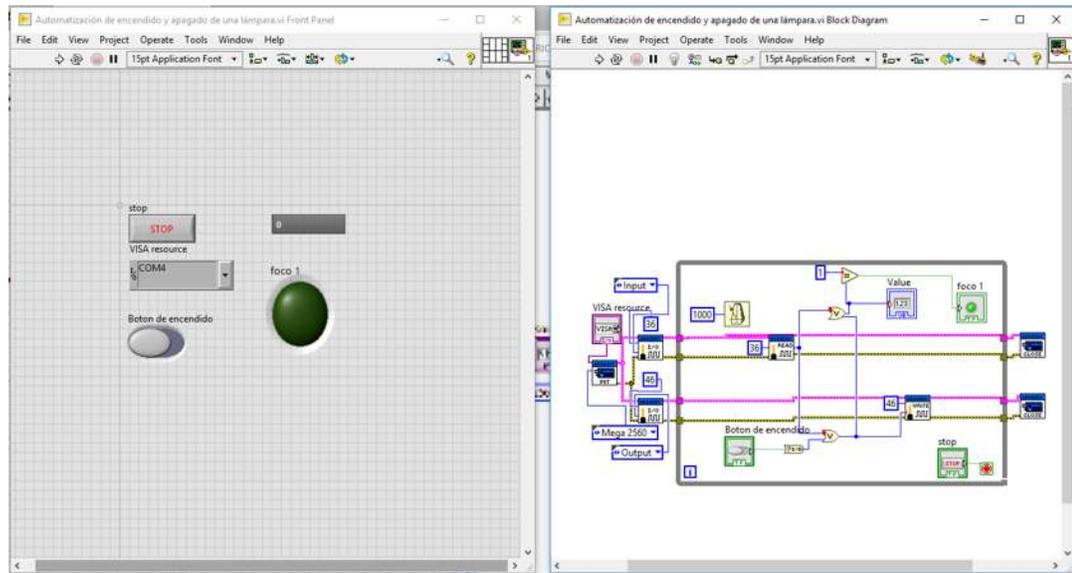


Figura 29 Programación para control de 1 lámpara con sensor, (Panel frontal y Diagrama de bloques respectivamente)

### 3.5.2.3 Automatización de la alarma inteligente.

Para la automatización de encendido y apagado de la **alarma inteligente** controlada por un sensor PIR, se ejecutó la programación que muestra la **Figura 30 Panel frontal de la alarma inteligente** y **Figura 31 Diagrama de bloques de la alarma inteligente**.

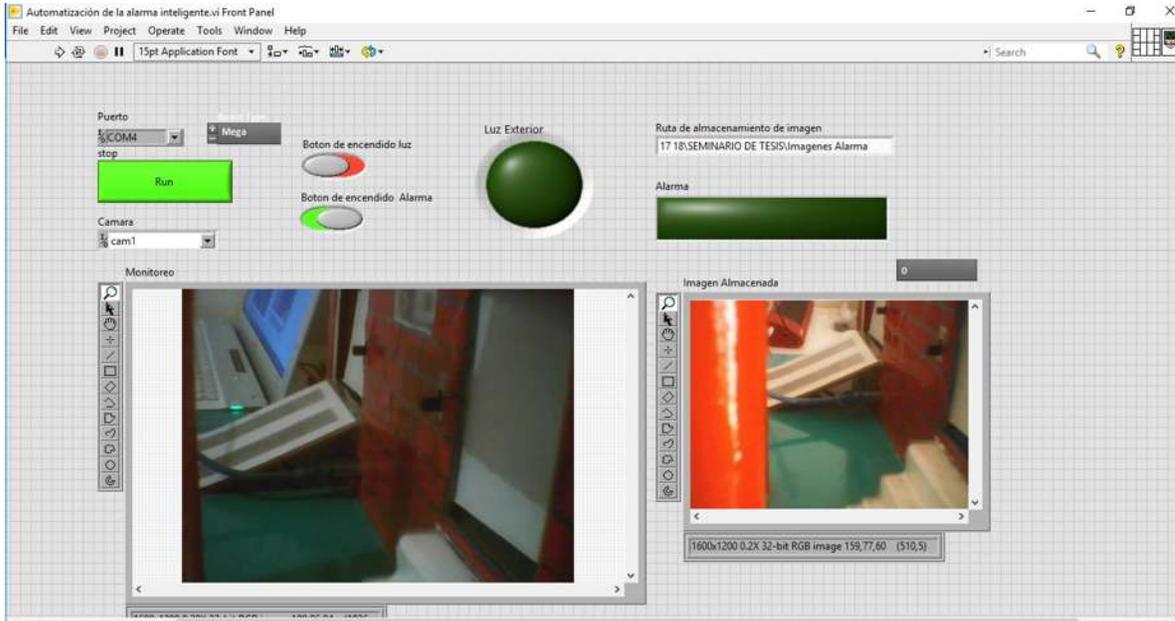


Figura 30 Panel frontal de la alarma inteligente

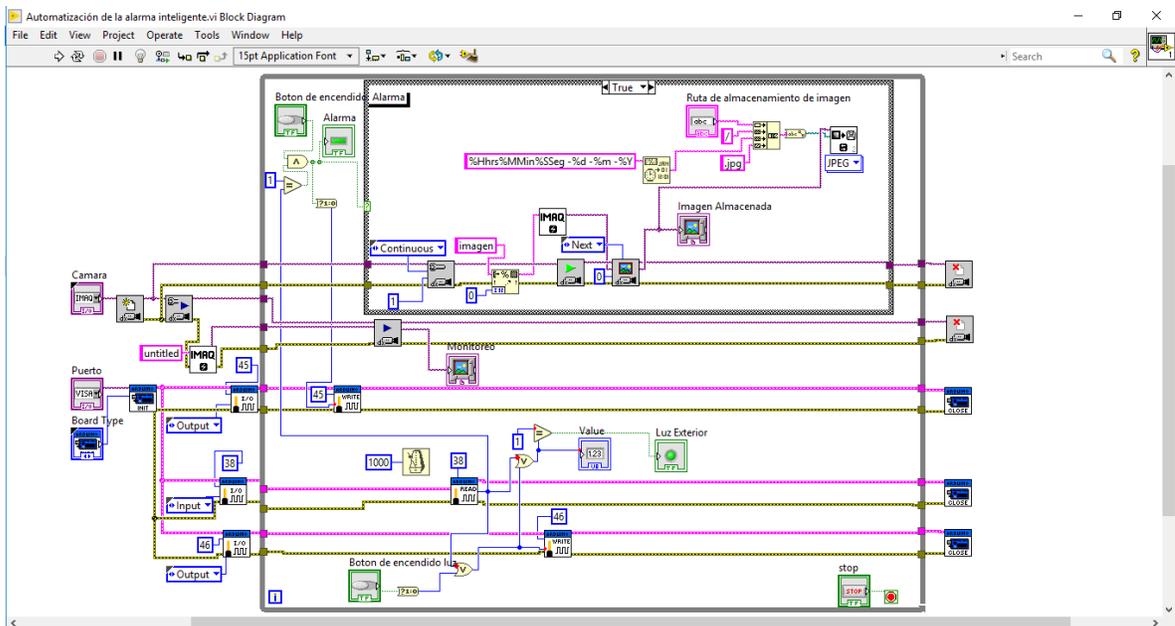


Figura 31 Diagrama de bloques de la alarma inteligente

### 3.5.2.4 Interfaz gráfica

Y finalmente se realizó la **interfaz gráfica** mediante una conjunción de las interfaces graficas anteriores y se multiplico por n veces según lo establecido en la delimitación de la tesis, para así tener la automatización de apertura y cierre de **cinco puertas y cuatro ventanas** el encendido y apagado automático de las **5 lámparas interiores y una exterior** y también para la automatización de encendido y apagado de la **1 sistema de alarma inteligente** controlada por un sensor PIR, para ello se realizó la programación mostrada de la **figura 28** a la **figura 35**. En la cual se desarrolló dividiéndola en 4 secciones, una para control de las puertas y la luz del garaje, otra para el control de las ventanas y la siguiente para el control de las luces interiores y la última para el control de la alarma inteligente y las luces exteriores.

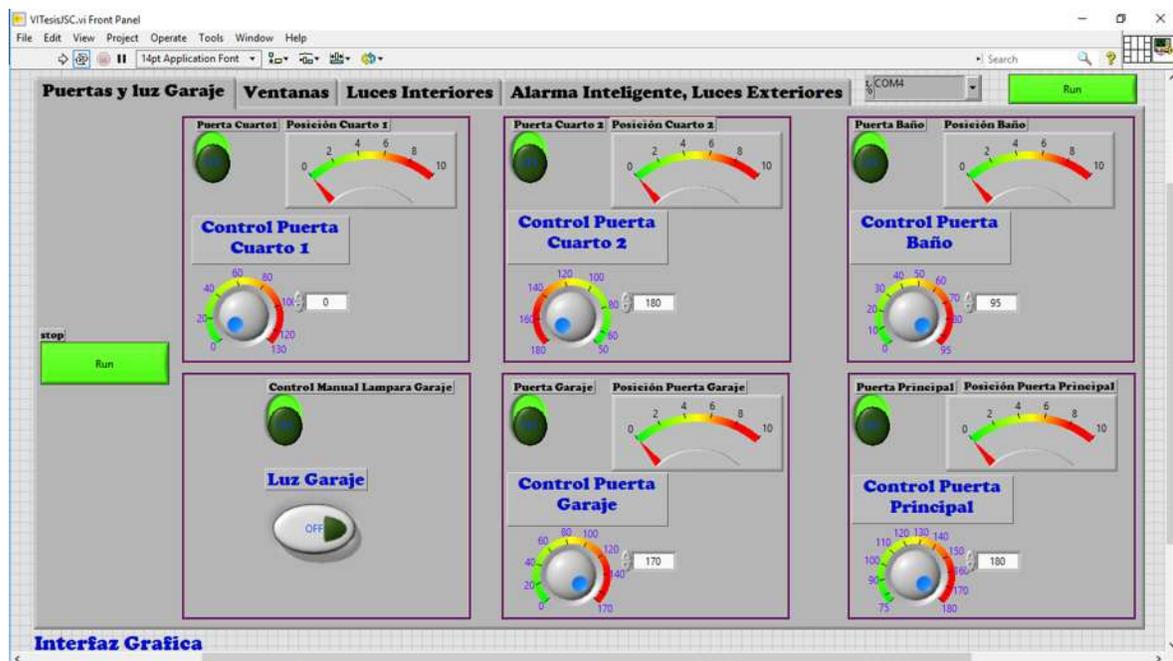


Figura 32 Interfaz gráfica de usuario desarrollada para el control domótico de las puertas de la casa (Panel Frontal).

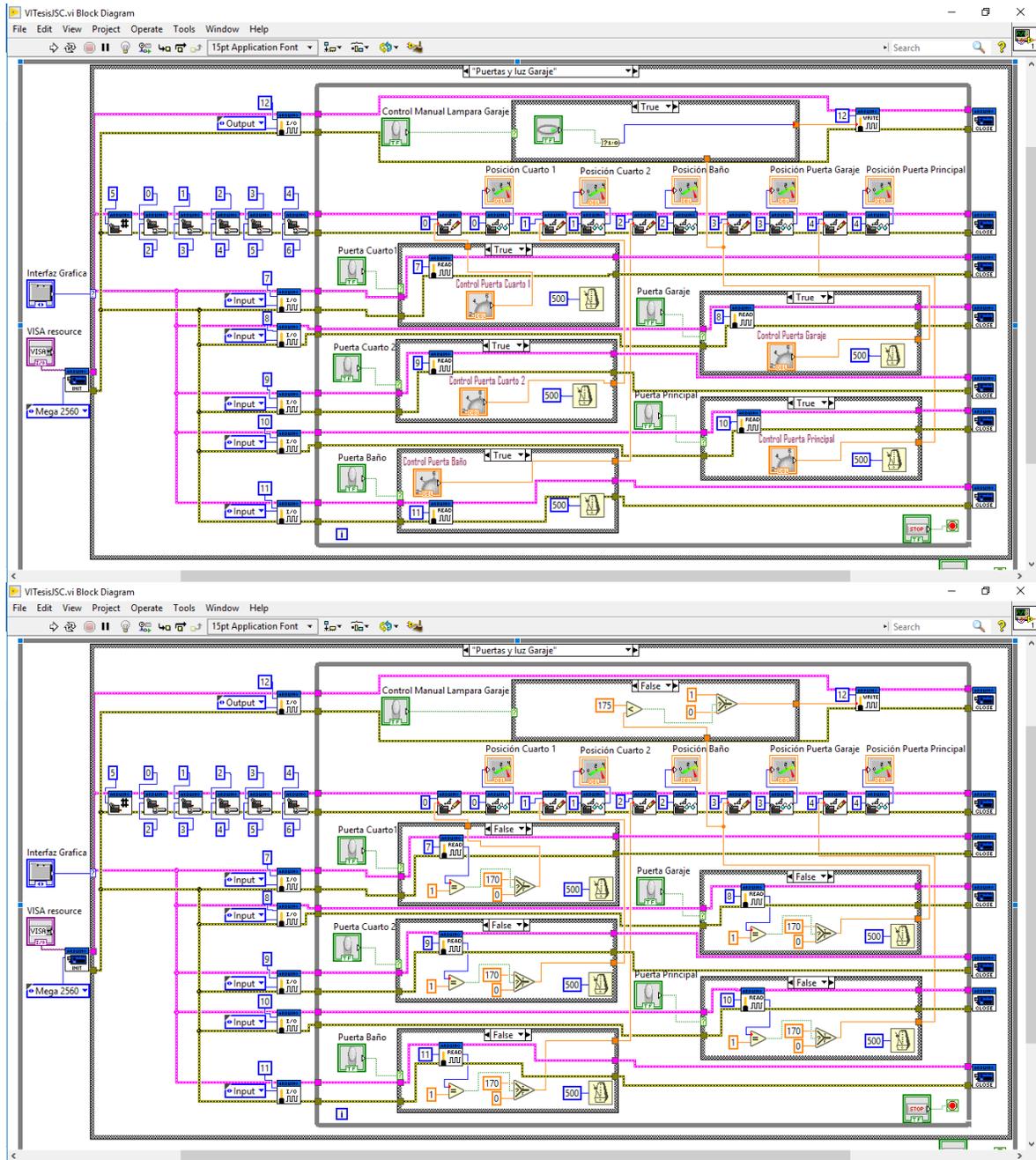


Figura 33 Programación para la Interfaz gráfica de usuario desarrollada para el control domótico de las puertas de la casa (Diagrama de Bloques) "Superior casos en verdadero, inferior casos en falso".

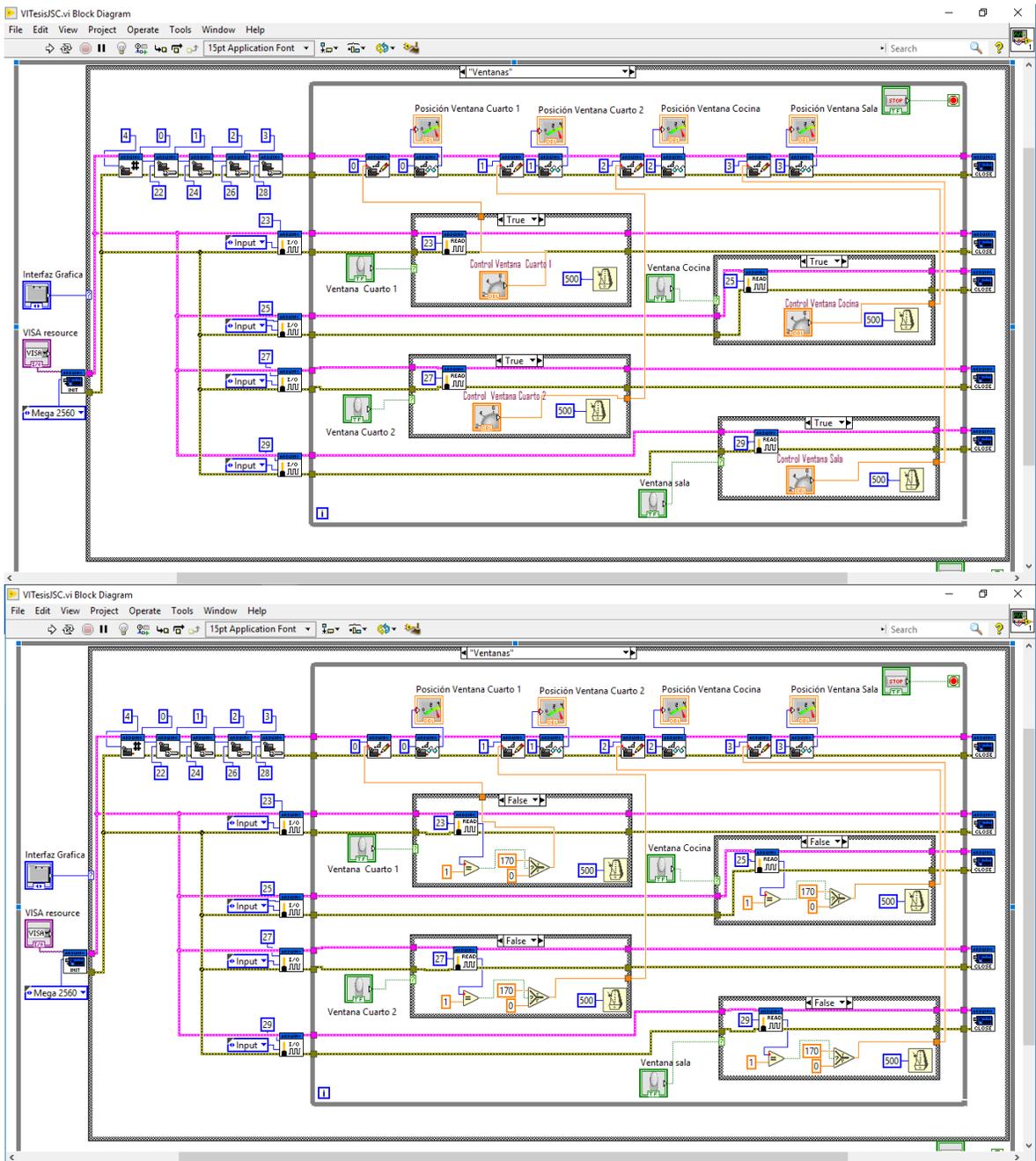


Figura 34 Programación para la interfaz gráfica de usuario desarrollada para el control domótico de las ventanas de la casa (Diagrama de Bloques) “Superior casos en verdadero, inferior casos en falso”.

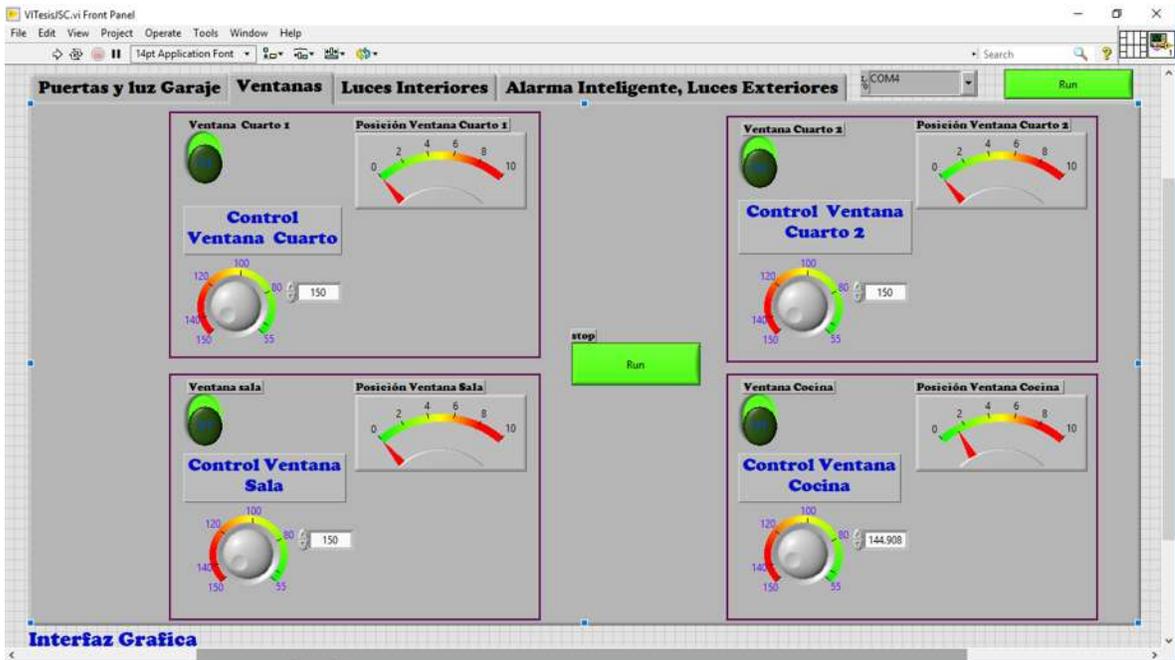


Figura 35 Interfaz gráfica de usuario desarrollada para el control domótico de las ventanas de la casa (Panel Frontal).

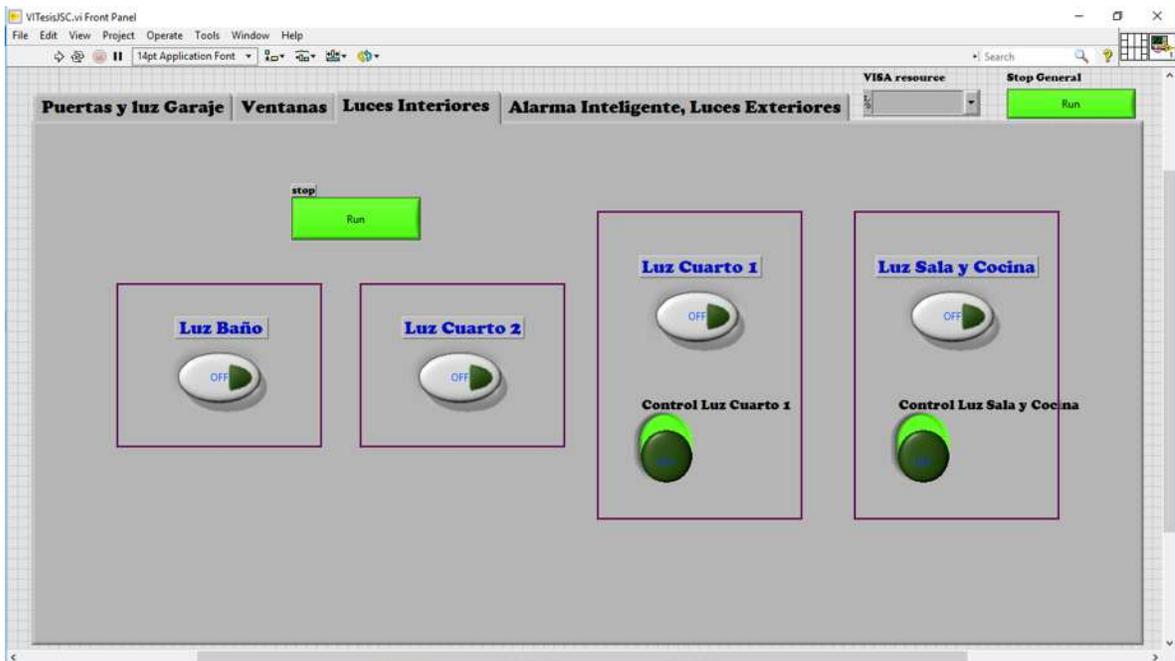


Figura 36 Interfaz gráfica de usuario desarrollada para el control domótico de las luces internas de la casa (Panel Frontal).

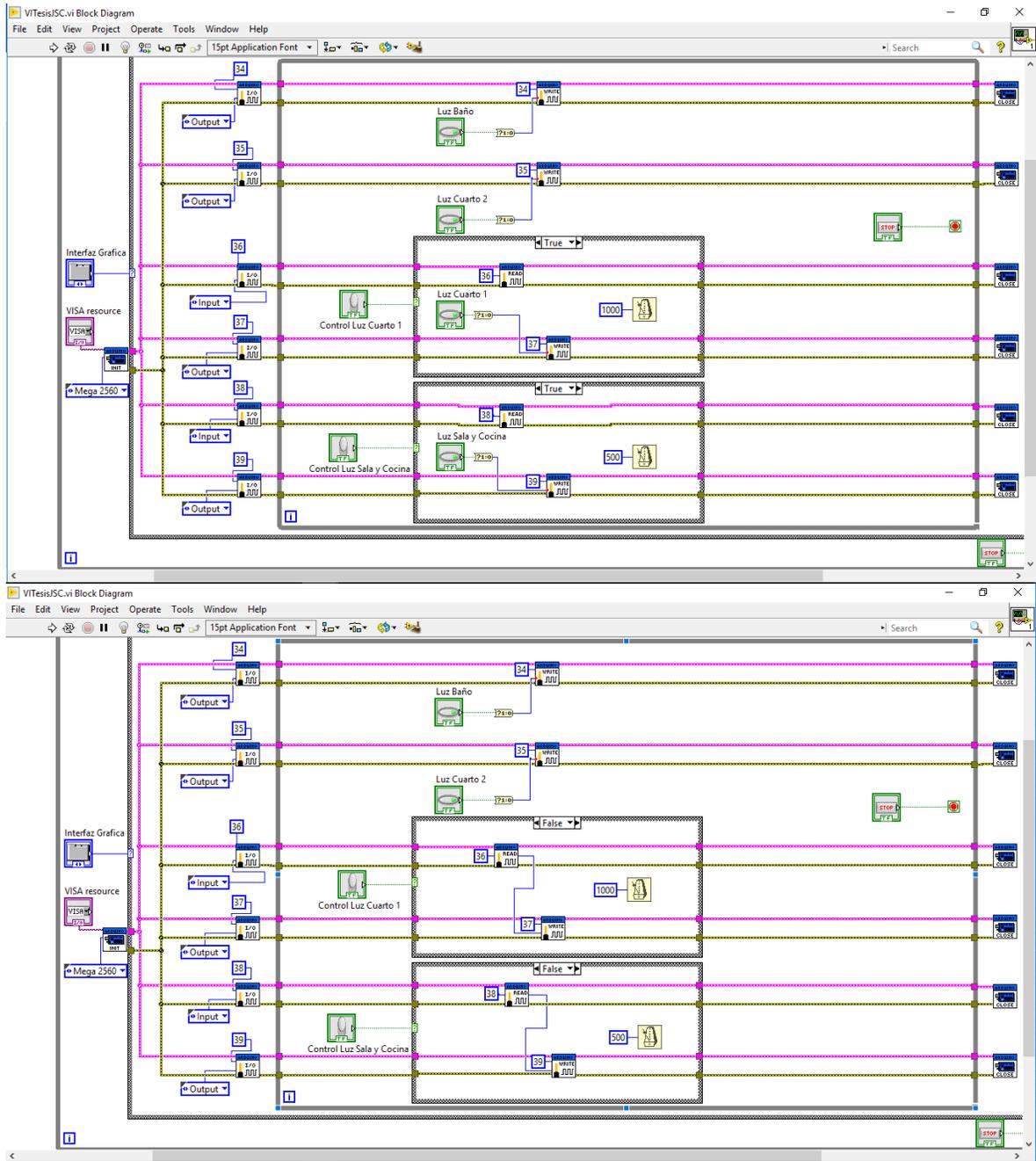


Figura 37 Programación para la interfaz gráfica de usuario desarrollada para el control domótico de las luces internas de la casa (Diagrama de Bloques) “Superior casos en verdadero, inferior casos en falso”.

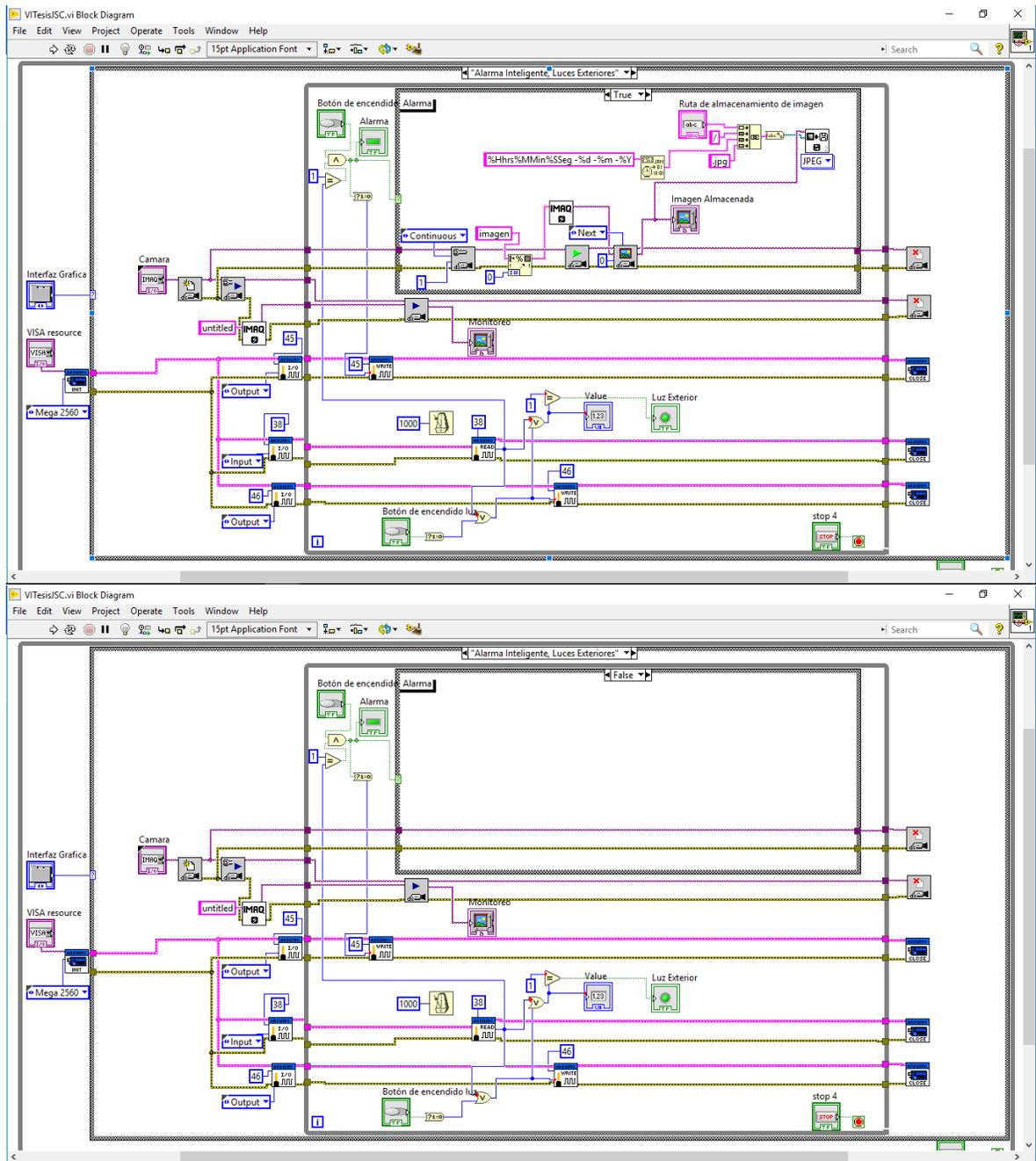


Figura 38 Programación para la Interfaz gráfica de usuario desarrollada para el control domótico de la alarma inteligente y luces externas de la casa (Diagrama de Bloques) "Superior casos en verdadero, inferior casos en falso"

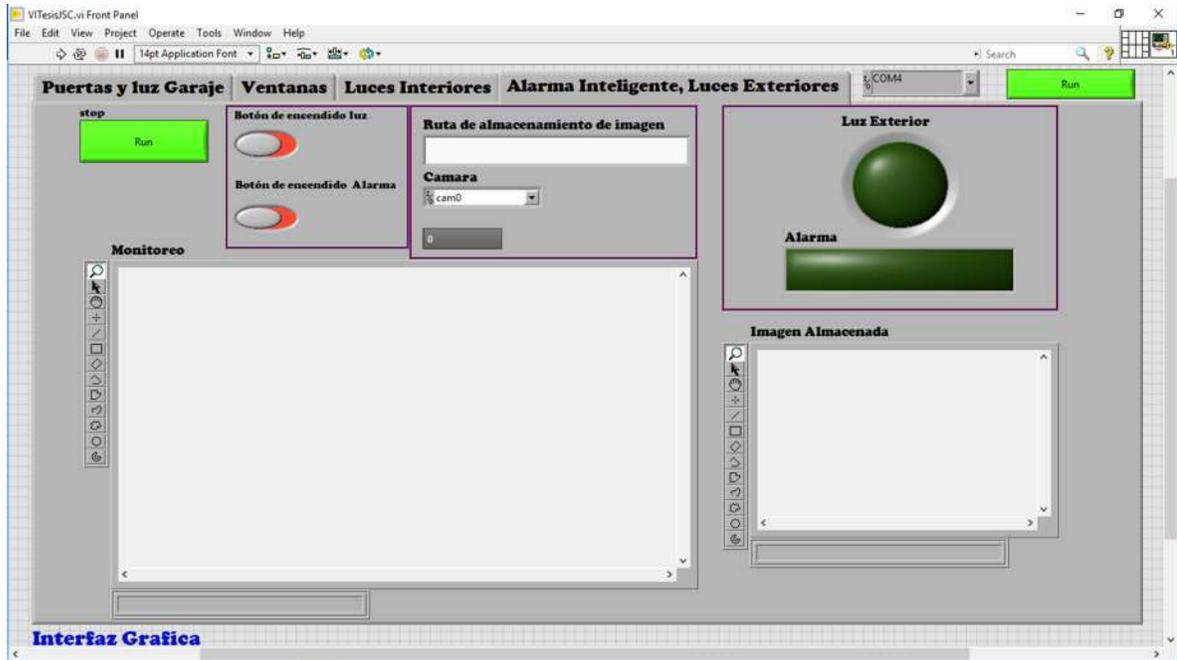


Figura 39 Interfaz gráfica de usuario desarrollada para el control domótico de la alarma inteligente y luces externas la casa (Panel Frontal).

## Capítulo 4. Comunicación Inalámbrica y Resultados: Uso y funcionamiento de la interfaz gráfica aplicada a la maqueta.

### 4.1 Introducción

Se muestran los resultados obtenidos al aplicar la **interfaz gráfica** desarrollada en la **maqueta**, permitiendo de esta manera que el usuario tenga una comunicación y control del sistema domótico de su casa habitación, todo esto desde una computadora personal de forma local, estos resultados se dan a conocer de manera secuencial de acuerdo a como se realizaron los avances en la interfaz tal como se mostró en el capítulo 3.

Primero se muestra los resultados de la automatización de apertura y cierre de **una puerta y una ventana** después los del encendido y apagado de una **lámpara**, posteriormente los del funcionamiento de la **alarma inteligente** y finalmente se muestran los resultados de la **Interfaz Gráfica completa**.

También se presenta una comunicación de manera remota, mediante la publicación de la **Interfaz Gráfica** en la red de internet en código HTML.

### 4.2 Resultados de las interfaces graficas

#### 4.2.1 Los resultados de la automatización de apertura y cierre de **una puerta y una ventana**.

Los resultados de la automatización de apertura y cierre de **una puerta y una ventana** se pueden observar en la **Figura 40 Comprobación del funcionamiento de la Automatización de 1 puerta y 1 ventana**, obtenido un adecuado funcionamiento:

#### 4.2.2 Resultados y funcionamiento de la automatización de encendido y apagado de **una lámpara**

Los resultados de la automatización de encendido y apagado de **una lámpara**, donde el control, funciona desde la pc mediante un botón virtual o con el sensor de movimiento de manera física, el funcionamiento se pueden observar en la **Figura 41 Comprobación del funcionamiento de la Automatización de 1 lámpara desde el pc y con el sensor de movimiento** , logrado un apropiado funcionamiento.

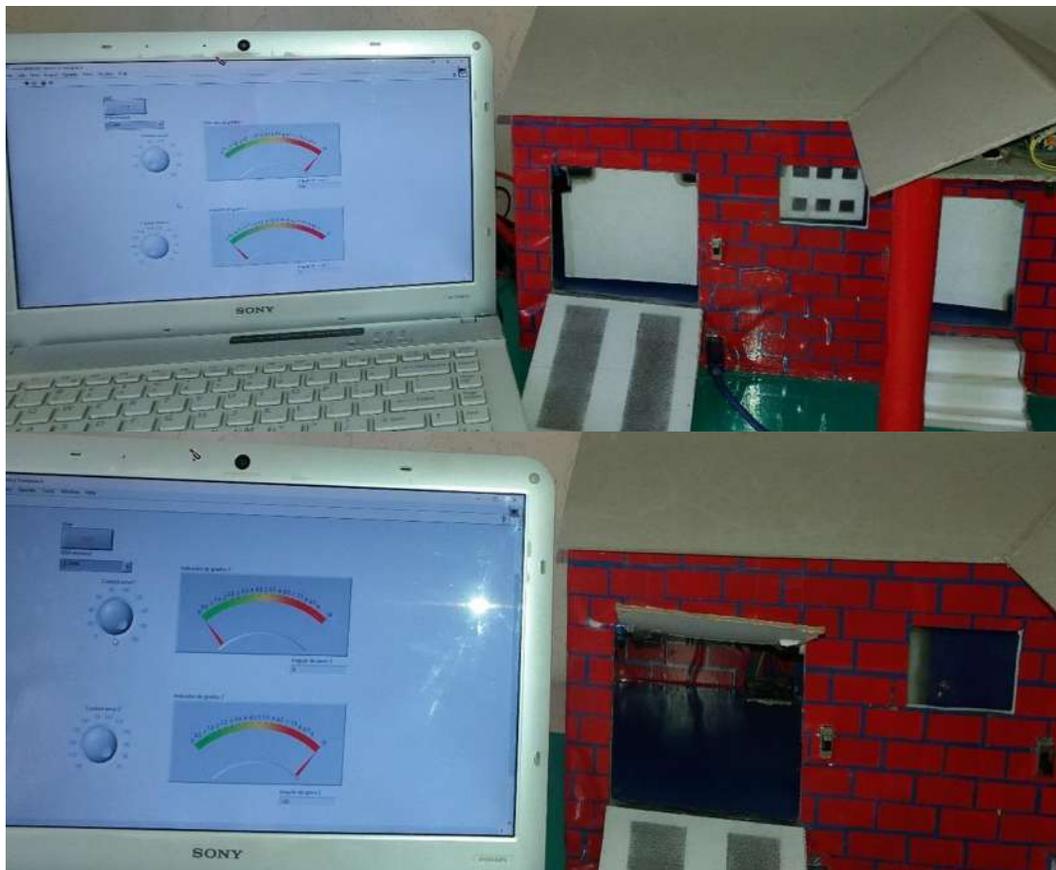


Figura 40 Comprobación del funcionamiento de la Automatización de 1 puerta y 1 ventana

#### 4.2.3 Resultados y funcionamiento de la automatización de encendido y apagado de la alarma inteligente.

A hora se muestran los resultados de la automatización de encendido y apagado de **la alarma inteligente**, la cual funciona, controlando el encendido y apagado de la luz exterior, desde el pc, así mismo como por medio del sensor de movimiento (PIR), mientras este desactivada la alarma; y cuando la alarma se activa el encendido del foco y del altavoz se hacen por medio del sensor PIR; de igual manera mediante la acción del sensor también se toman capturas de imagen y las guarda con fecha y hora, en la ruta que se le asigne (ver **Figura 47 Imágenes tomadas y almacenadas por la cámara cuando detecta movimiento.**), estas capturas las toma en el momento que el sensor detecta un movimiento; también mediante la cámara se está monitoreado todo el tiempo la entrada principal de la casa domótica, todo ello se puede observar en la **Figura 42 Funcionamiento de la Alarma**, en la cual se muestra un apropiado funcionamiento del sistema de alarma inteligente

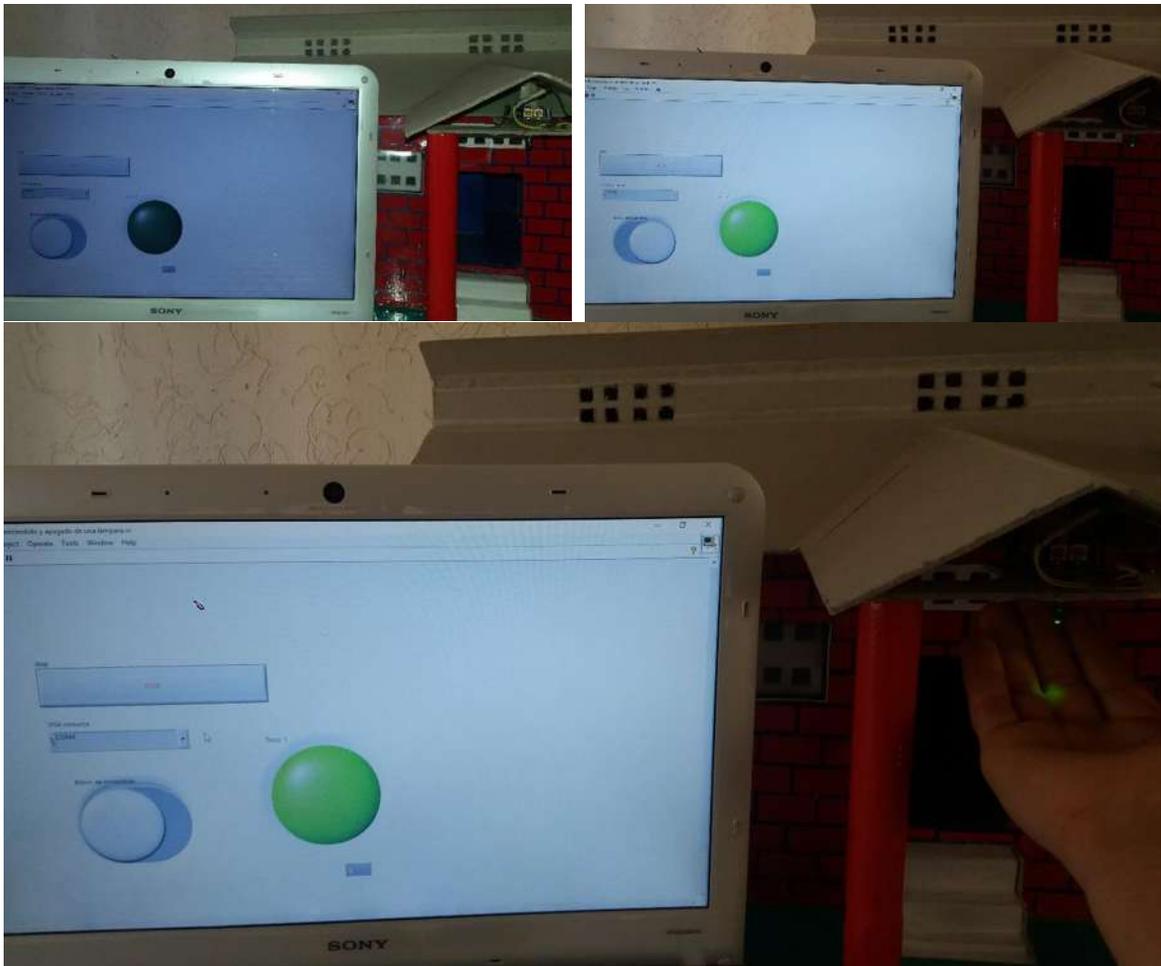


Figura 41 Comprobación del funcionamiento de la Automatización de 1 lámpara desde el pc y con el sensor de movimiento



Figura 42 Funcionamiento de la Alarma Inteligente (secuencia de Izq. Derecha).

#### 4.2.4 Resultados de la automatización de la **Interfaz Completa** para la casa domótica (1 Alarma inteligente, 5 Puertas, 4 Ventanas, 5 Luces internas y 1 externa)

Los resultados de la automatización de la interfaz completan para la automatización de la casa domótica se pueden observar desde la figura 40 hasta la figura 44, lo cual se explica a continuación.

##### 4.2.4.1 *Funcionamiento de la interfaz*

Esta interfaz tiene 4 conjuntos de controles; el primero es para el control de las puerta y la luz del garage; el segundo es para el control de las ventanas; el tercero es para el control de las luces internas; y el cuarto es para el control de la alarma inteligente y la luz externa.

**El primer bloque o conjunto de controles** se encarga de controlar cinco puertas (La puerta del cuarto principal, la del segundo cuarto, la del baño, la del acceso principal y la de acceso del garage), todas ellas se controlan de tal manera, que mediante dos controles en la interfaz, se pueden abrir y cerrar la puertas desde la misma interfaz y desde un interruptor físico; En el caso de la puerta del garage se diseñó del tal manera que la luz dependiera del estado de la puerta, es por ello que este conjunto también incluye el control de la luz del garage; El funcionamiento de este bloque se puede ver en la **Figura 43 Imágenes del funcionamiento de las puertas.**

**El segundo bloque de controles** contiene los controles necesarios para tener el control de igual manera que el control que se tiene en las puertas, pero ahora sobre las 4 ventanas (Ventana del baño, de la sala, la cocina y los 2 cuartos); El funcionamiento de este bloque se puede ver en la **Figura 44 Imágenes del funcionamiento de las ventanas.**

**El tercer bloque** contiene los controles para las 4 luces internas (lámpara de cuarto principal, lámpara del segundo cuarto, lámpara del baño y lámpara de la sala-cocina), en este caso la lámpara de sala y cocina y la lámpara del cuarto principal tiene una programación para que se pueda tener control desde la interfaz gráfica y dependan también del sensor de movimiento PIR, y las dos lámpara restantes del segundo cuarto y del baño solo se automatizaron de manera que solo se tiene control desde la propia interfaz gráfica: El funcionamiento de este bloque se puede ver en la **Figura 45 Imágenes del funcionamiento de las luces interiores.**

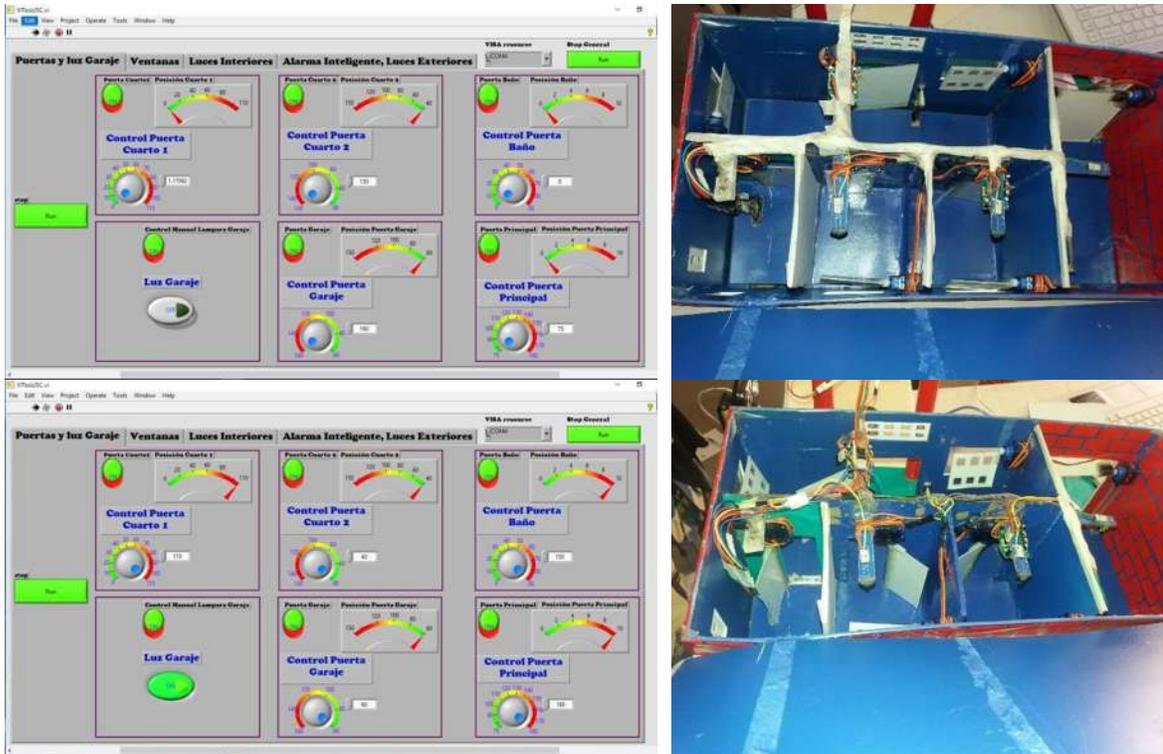


Figura 43 Imágenes del funcionamiento de las puertas.

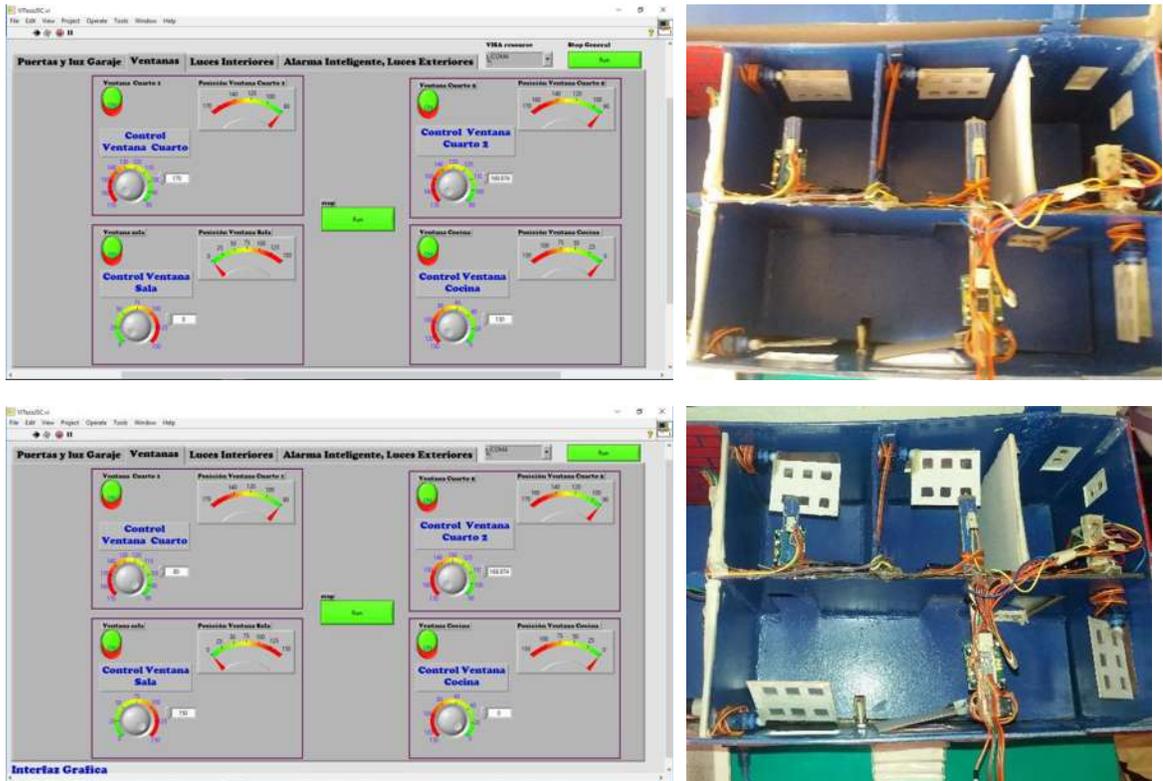


Figura 44 Imágenes del funcionamiento de las ventanas.

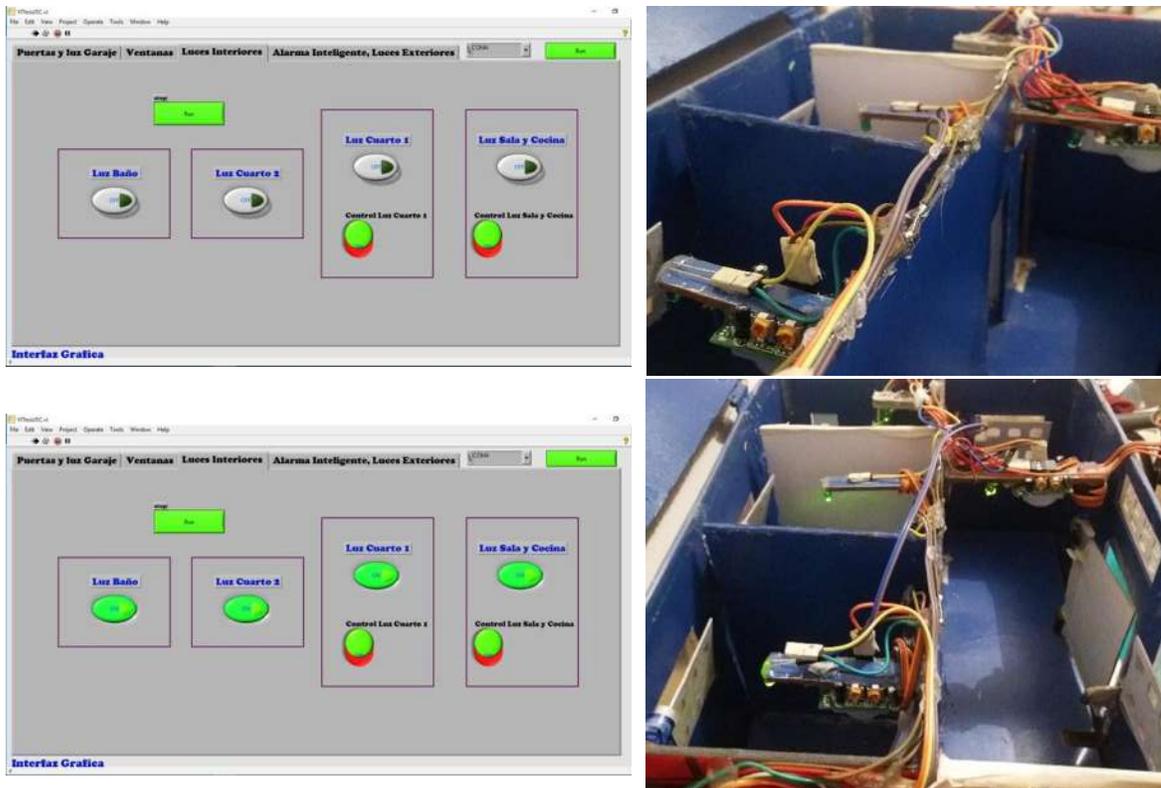


Figura 45 Imágenes del funcionamiento de las luces interiores.

**El cuarto bloque** contiene una programación en la cual mediante el monitoreo constante de una cámara se puede obtener información precisa de un posible robo, ya que la cámara está programada mediante la detección de movimiento del sensor PIR esta se encarga de tomar fotografías y almacenarlas en una carpeta dentro del computador para poder ser revisadas por el usuario y también mediante el sensor de movimiento se activa la luz y un altavoz para alertar al posible ladrón; Y debido a la dependencia de la luz exterior con la cámara este bloque también contiene los controles de la luz externa. Cabe destacar que la alarma se puede activar o desactivar mediante un control desde la misma interfaz gráfica: El funcionamiento de este bloque se puede ver en la **Figura 46 Imágenes de la alarma en funcionamiento**. Mientras que las imágenes almacenadas por la alarma inteligente se pueden ver en la **Figura 47 Imágenes tomadas y almacenadas por la cámara cuando detecta movimiento**.



Figura 46 Imágenes de la alarma en funcionamiento.

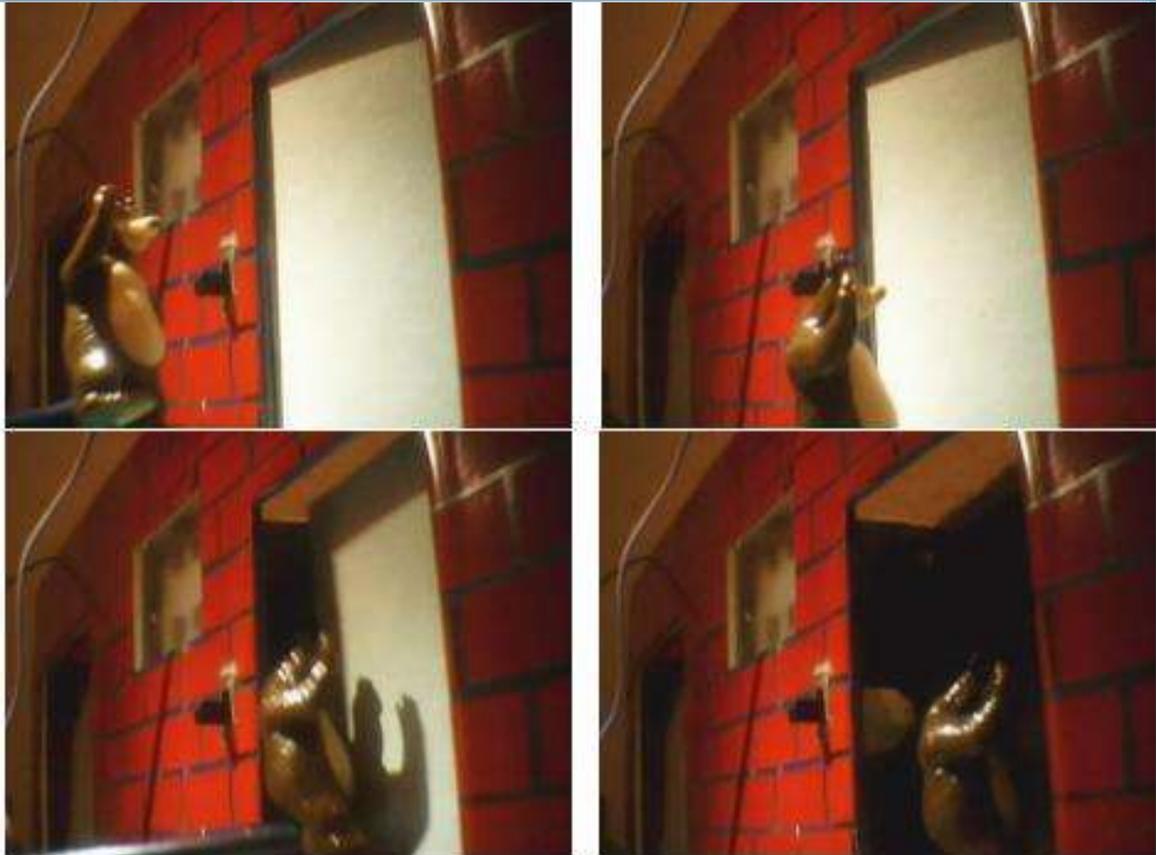
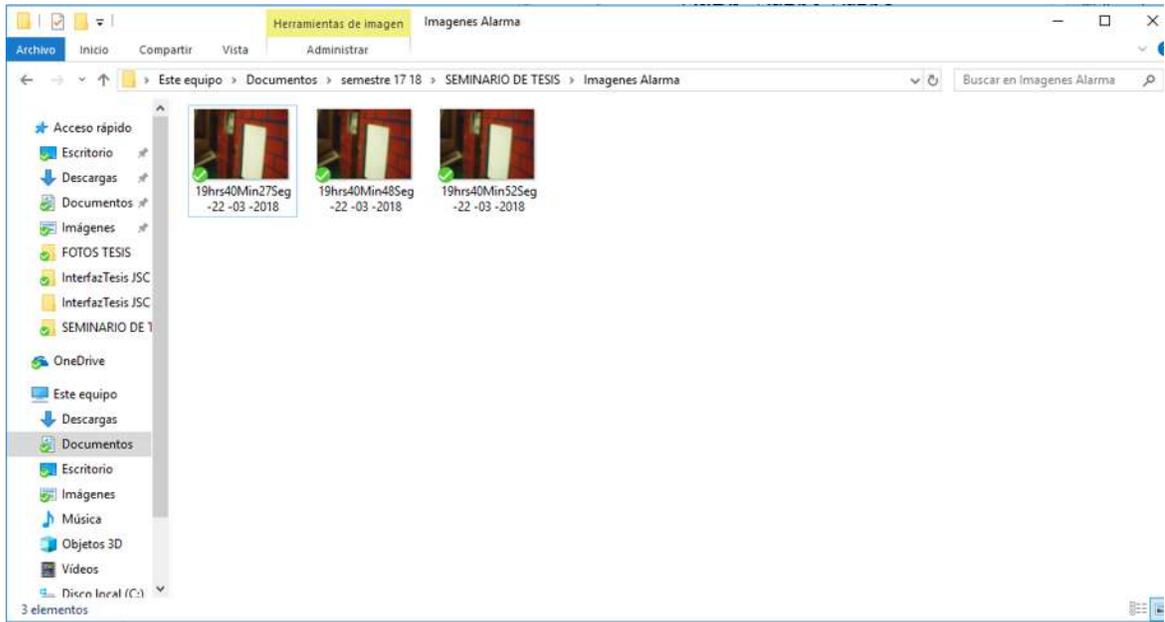


Figura 47 Imágenes tomadas y almacenadas por la cámara cuando detecta movimiento.

### 4.3 Comunicación inalámbrica mediante internet, en una computadora o cualquier dispositivo.

Para realizar la comunicación vida remota desde el internet, se utilizará la **Publicación Web**:

Para ello ya teniendo la interfaz gráfica utilizaremos la herramienta **Web Publishing Tool**, esta herramienta es un asistente que para crear una página web a través de tres simples pasos; el **primer paso** consiste en **elegir el VI** y el **modo de publicación**; el **segundo paso** es para personalizar la página colocándole un Nombre o título y texto que se mostraran en el panel frontal; y el **tercer paso** es para guardar el fichero.

En la versión profesional que es la que se está usando hay tres modos de publicación, los cuales describo a continuación:

*Embedded*: Este permite tener control y monitorización de manera remota del VI si este está cargado es decir está abierto en el pc local.

*Snapshot*: Solo muestra una captura de pantalla, es decir una imagen PNG del panel frontal del VI.

*Monitor*: A diferencia del el *Snapshot* este muestra la misma captura de pantalla, pero esta se actualiza cada cierto intervalo de tiempo. [8]

A continuación, se muestra el procedimiento anteriormente expuesto aplicado a la **Interfaz Gráfica** creada mediante los tres pasos ya mencionados, para ello se va al menú tools y dar clic en Web Publishing Tool... como se muestra en la **Figura 48 Herramienta Web Publishing Tool**.

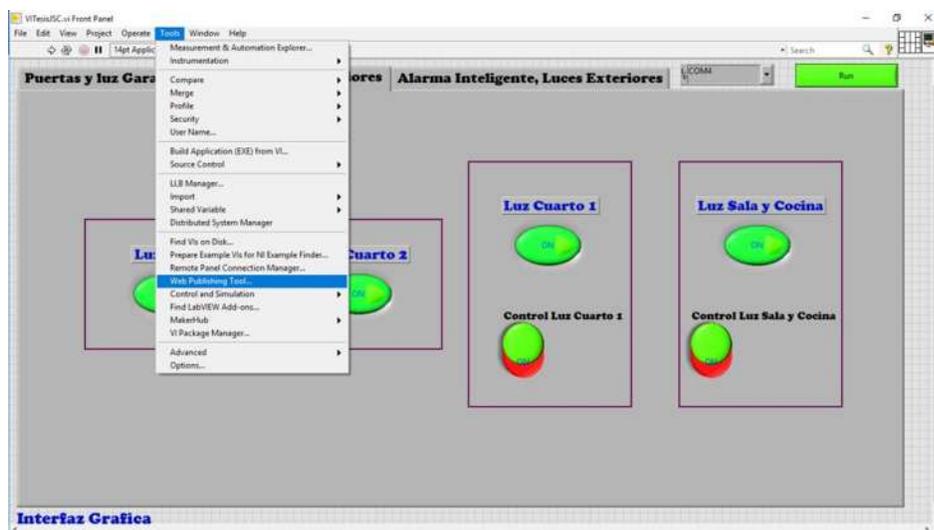


Figura 48 Herramienta Web Publishing Tool.

**Primer paso:** Elegir el VI y el modo de publicación, para mi caso elegí el VI que contiene la **Interfaz Gráfica** creada y el **modo embebido(Embedded)**, para poder tener control completo de mi interfaz, véase la **Figura 49 Paso 1 selección del VI y Modo de Publicación**.

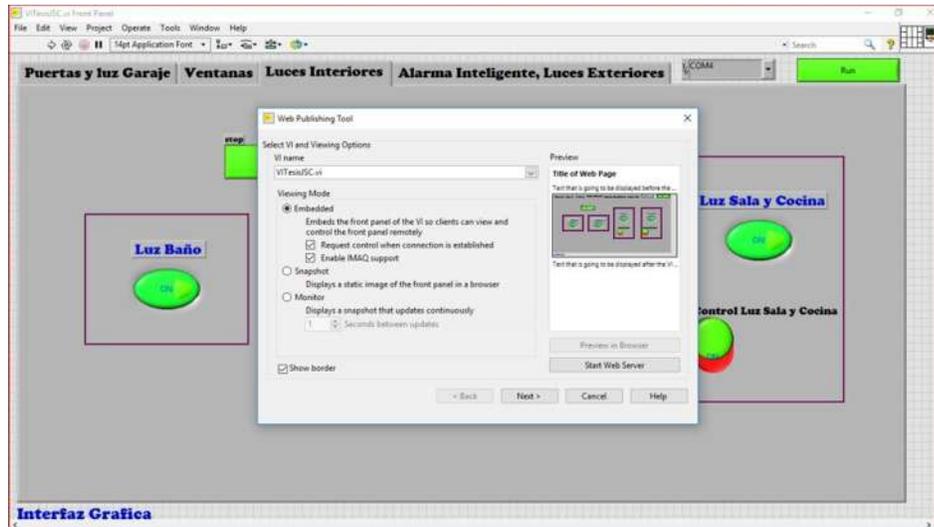


Figura 49 Paso 1 selección del VI y Modo de Publicación.

**Segundo paso:** Consiste en personalizar la página, para ello le coloque un título y algunos datos tal como se muestra en la **Figura 50 Personalización de la página web**.

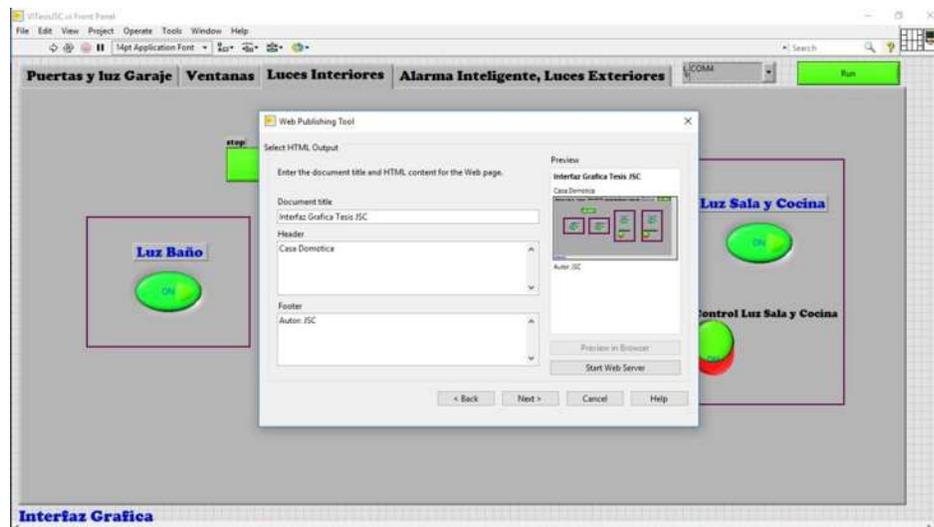


Figura 50 Personalización de la página web.

**Tercer paso:** Finalmente este solo es guardar el archivo o fichero, tal como se ve en la **Figura 51 Guardado del fichero**.

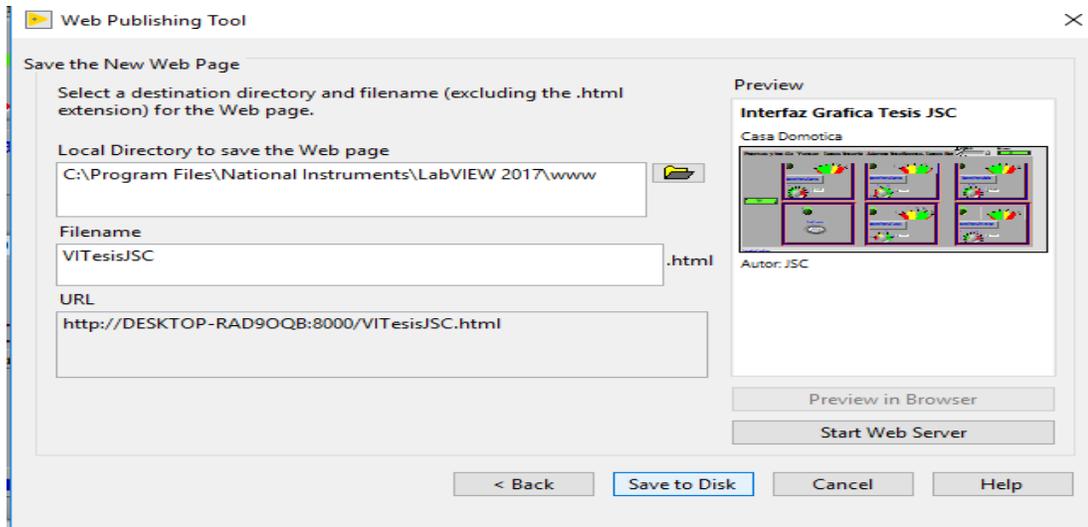


Figura 51 Guardado del fichero.

Una vez guardado nos arroja la información que se ilustra en la **Figura 52 Dirección de la página**, la cual es en la dirección de la página web para entrar al VI en mi caso la interfaz Gráfica creada, también podemos observar que nos da la opción de conectarnos desde esta ventana en *Connect*.

Como funciona en realidad el navegador pedirá la página en que se publica el VI al servidor, el servidor enviará la página al navegador, la página web mostrada requerirá un plug-in que debe estar instalado en el navegador y el plug-in utilizara las funciones de LabVIEW Run-Time para mostrar y controlar el Panel Frontal. [8]

Al cargar el link <http://DESKTOP-RAD9OQB:8000/VITesisJSC.html> o dar clic en conectar, se abre el VI desde en el navegador web en el dispositivo que deseemos, tal como se muestra en la **Figura 53. Comunicación vía remota por medio de la página web desde otro ordenador** con el cual podemos tener control y visualización de nuestra interfaz de manera remota desde cualquier parte si necesidad de estar presentes en la computadora local que alberga la interfaz. Esto nos da muchas ventajas ya que podemos monitorear y controlar nuestra casa prácticamente desde cualquier parte donde tengamos acceso al internet y a una computadora.

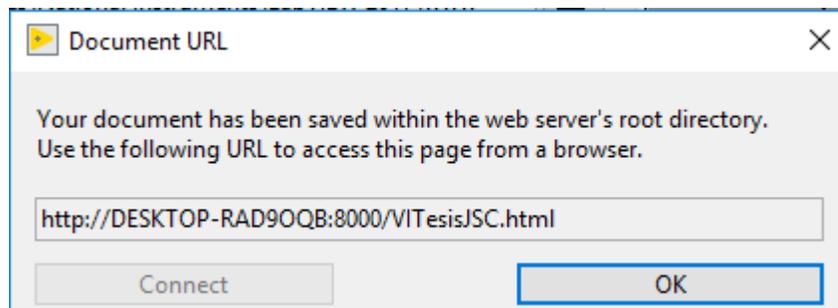


Figura 52 Dirección de la página web (URL)



Figura 53. Comunicación vía remota por medio de la página web desde otro ordenador

Se puede modificar que computador tiene el control el remoto o el local a través del menú que aparece en el rectángulo blanco de la esquina inferior izquierda del VI o Interfaz Gráfica, tal como se muestra:



Es importante que el navegador tenga activada la casilla **Habilitar** para descargar los controles **Active X**, que es el plug-in que el VI de LabVIEW, requiere para ejecutarse desde el navegador, tal como se muestra en la **Figura 54 Activación de descarga de controles Active X en el navegador Internet Explorer.**

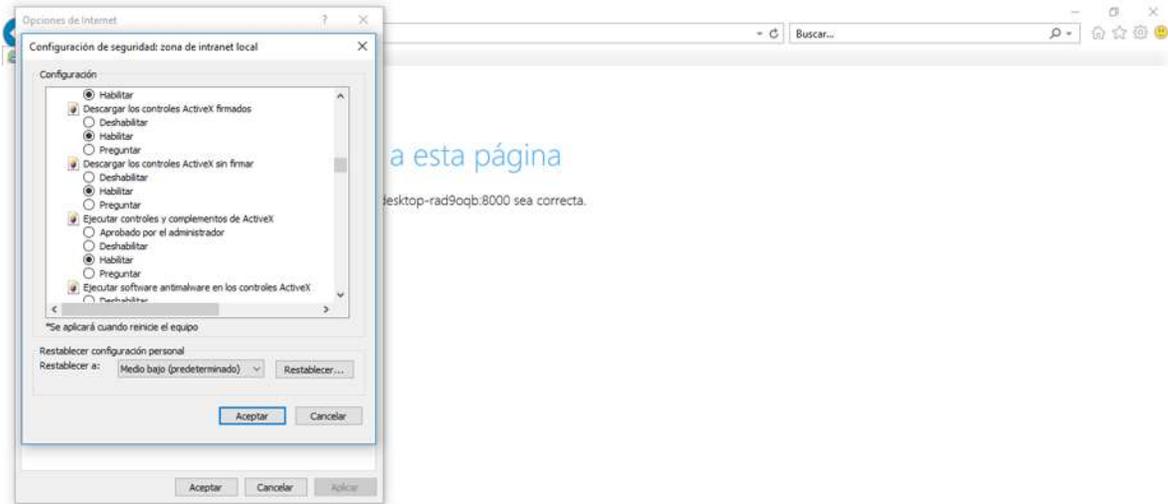


Figura 54 Activación de descarga de controles Active X en el navegador Internet Explorer.

## 4.4 Propuestas de actuadores (motores) y sensores en aplicación real.

### Servo-motores como actuadores.

Actualmente se puede encontrar una amplia gama de servo-motores de todo tipo y elegir entre alguno de ellos depende del uso específico que se le dará

Se proponen dos tipos de servomotores, que nos pueden ser útiles en la automatización de apertura y cierre de puertas y ventanas en una casa domótica los cuales son el servo rotatorio **SGMJV** o el lineal **SGLGW** ambos de la marca **YASKAWA** de los cuales podemos encontrar su información en la liga <http://www.pillar.com.mx/Varios/AC%20SERVOS%20DRIVES.pdf>

En la Figura 55 Especificaciones de los servomotores tenemos una descripción de las principales características de acuerdo a capacidad de cada servo motor.

| Rotary Servomotor Model  | Capacity  | SERVOPACK Model SGDV-□□□□ |                     |                     | Stock Status |   |
|--|-----------|---------------------------|---------------------|---------------------|--------------|---|
|  |           | Single-phase 100 VAC      | Three-phase 200 VAC | Three-phase 400 VAC |              |   |
| SGMJV<br>(Medium Inertia, Small Capacity)<br>3000min <sup>-1</sup> | SGMJV-A5A | 50 W                      | R70F                | R70A                | —            | S |
|  | SGMJV-01A | 100 W                     | R90F                | R90A                |              |   |
|  | SGMJV-C2A | 150 W                     | 2R1F                | 1R6A                |              |   |
|  | SGMJV-02A | 200 W                     | 2R1F                | 1R6A                |              |   |
|  | SGMJV-04A | 400 W                     | 2R8F                | 2R8A                |              |   |
|  | SGMJV-06A | 600 W                     | —                   | 5R5A                |              |   |
|  | SGMJV-08A | 750 W                     | —                   | 5R5A                |              |   |



SGMJV  
(Medium Inertia, Small Capacity)  
50 W to 750 W

| Linear Servomotor Model                              | Rated Force N | Peak Force N | SERVOPACK Model SGDV-□□□□ |                     |                     | Stock Status |   |
|--|---------------|--------------|---------------------------|---------------------|---------------------|--------------|---|
|  |               |              | Single-phase 100 VAC      | Three-phase 200 VAC | Three-phase 400 VAC |              |   |
| SGLGW<br>(Coreless Type, With standard magnetic way) | SGLGW-30A050C | 12.5         | 40                        | R70F                | R70A                | —            | S |
|  | SGLGW-30A080C | 25           | 80                        | R90F                | R90A                |              |   |
|  | SGLGW-40A140C | 47           | 140                       |                     |                     |              |   |
|  | SGLGW-40A253C | 93           | 280                       | 2R1F                | 1R6A                |              |   |
|  | SGLGW-60A140C | 70           | 220                       |                     |                     |              |   |
|  | SGLGW-40A365C | 140          | 420                       | 2R8F                | 2R8A                |              |   |
|  | SGLGW-60A253C | 140          | 440                       |                     |                     |              |   |
|  | SGLGW-60A365C | 210          | 660                       | —                   | 5R5A                |              |   |
|  | SGLGW-90A200C | 325          | 1300                      |                     | 120A                |              |   |
|  | SGLGW-90A370C | 550          | 2200                      |                     | 180A                |              |   |
|  | SGLGW-90A535C | 750          | 3000                      |                     | 200A                |              |   |



SGLGW (Coreless Type)  
12.5 to 750 N

Figura 55 Especificaciones de los servomotores SGMJV SGLGW

## Sensores de movimiento

Los sensores de movimiento para llevar acabo la detección de intrusos y la activación de la luces en un hogar domótico pueden ser cualquier tipo de sensores comerciales existentes, actualmente existen variedades de marcas y modelos que se pueden usar dependiendo del cliente, y de manera específica yo propongo el uso de los modelos **PDM-IXx12/T y PDM-IXx18/T** de la marca **siemens** los cuales son detectores de movimientos de doble tecnología sus características se pueden ver en [línea en la liga https://es.spiap.com/uploads/master/product/documents/v54531f113a100\\_pdmixx12ixx12t\\_023\\_01\\_es.pdf](https://es.spiap.com/uploads/master/product/documents/v54531f113a100_pdmixx12ixx12t_023_01_es.pdf)

*Acerca de estos sensores*

*Los sensores de movimiento PDM-IXx12/T y PDM-IXx18/T, ofrecen una fiable detección de intrusos y una alta inmunidad contra falsas alarmas. Sus prestaciones se basan en una versión mejorada del algoritmo Matchtec de Siemens, que combina canales de infrarrojo pasivo (PIR) con microondas (MW) para tomar decisiones más precisas en la detección de movimiento en su zona de influencia.*



Figura 56 Sensor siemens PDM-IXx12/T y PDM-IXx18/T

*Asimismo, estos detectores proporcionan una alta fiabilidad en la detección y reducción de falsas alarmas gracias a las rutinas de procesamiento inteligente que incluye el algoritmo mejorado Matchtec. Un sistema que analiza el nivel de señal recibido por parte de los sensores IR y MW de un objeto en movimiento para tomar la decisión más adecuada en cada caso. Es posible incluso utilizar varios detectores muy cercanos unos de otros, ya que el algoritmo reduce la interferencia entre los módulos de microondas, lo que incrementa la flexibilidad a la hora de seleccionar la ubicación de las instalaciones [15]*

## Capítulo 5. Conclusiones y Trabajos Futuros

### 5.1 Conclusiones

#### 5.1.1 Conclusiones Generales

En esta tesis se logró comprender más acerca de la **domótica**, su **objetivo**, así como las amplias mejoras que esta puede ofrecer a un Hogar, como lo son **Ahorro Energético, Accesibilidad, Seguridad, Confort etc.**

También se logró una aplicación de los **conocimientos adquiridos a lo largo de la carrera**, para realizar el diseño, creación e implementación de una interfaz basada en la plataforma LabVIEW con un protocolo de comunicación (USB) con la plataforma Arduino, la cual se comprobó su funcionalidad en una práctica maqueta y se concluye que esta **interfaz gráfica** puede **usarse de manera práctica en la vida real ayudando a facilitar las tareas del hogar.**

Así mismo se observó que la **domótica** es un **área**, que está teniendo mucho **impacto en la actualidad**. Cabe destacar que con los conocimientos adquiridos en el transcurso de la carrera y en el desarrollo de esta tesis, se **logró** obtener; las bases necesarias para tener una **buena incursión al sector de la domótica** y así poder brindar una mejor calidad de vida a los usuarios, mediante la **solución** de sus requerimientos y necesidades.

#### 5.1.2 Conclusiones Particulares

Puedo concluir que se logró diseñar una interfaz gráfica en la plataforma LabVIEW mediante programación gráfica, la cual tiene una comunicación (Software-Hardware) mediante el uso de la interfaz LabVIEW para Arduino (LIFA) *Toolkit*, también se logró tener una comunicación vía inalámbrica mediante la publicación del VI en la red, con lo cual se puede tener un acceso y control de manera local y de forma remota.

### 5.2. Trabajos Futuros

Los trabajos futuros a desarrollarse son los que se numeran a continuación:

#### 5.2.1 Control remoto.

Se pretende hacer un control vía remota mediante una aplicación ya existente y por otro lado se pretende realizar una aplicación personalizada para cada usuario según sus necesidades

#### 5.2.1.1 Aplicación Data Dashboard

Mediante el uso de esta aplicación se pretende tener una visualización del estado del hogar domótico desde cualquier lugar donde se encuentre el usuario, es decir mediante una conexión vía wifi tener acceso a nuestra casa domótica para control y vigilancia de la misma. “El Data Dashboard permite crear **vistas** personalizadas y portátiles de las aplicaciones del software LabVIEW de National Instruments. Con esta aplicación se pueden crear paneles de control de datos para mostrar los valores de las variables compartidas publicados en la red y desplegar los servicios de Web de LabVIEW sobre indicadores, tales como gráficos, cuadros de texto, LEDs, etc.[14]

#### 5.2.1.1 Aplicaciones personalizadas en Android.

Se pretende que también esta interfaz gráfica creada se pueda manipular mediante una aplicación para Android creada mediante Android Studio.



Figura 57 Logo entorno de desarrollo de Android Studio.

Android Studio es el entorno de desarrollo integrado (IDE) oficial para el desarrollo de aplicaciones para Android y se basa en IntelliJ IDEA. Además del potente editor de códigos y las herramientas para desarrolladores de IntelliJ, Android Studio ofrece aún más funciones que aumentan tu productividad durante la compilación de apps para Android, como las siguientes:

- Un sistema de compilación basado en *Gradle* flexible
- Un emulador rápido con varias funciones
- Un entorno unificado en el que puedes realizar desarrollos para todos los dispositivos Android
- *Instant Run* para aplicar cambios mientras tu app se ejecuta sin la necesidad de compilar un nuevo APK
- Integración de plantillas de código y GitHub para ayudarte a compilar funciones comunes de las apps e importar ejemplos de código
- Gran cantidad de herramientas y *frameworks* de prueba
- Herramientas *Lint* para detectar problemas de rendimiento, usabilidad, compatibilidad de versión, etc.
- Compatibilidad con C++ y NDK
- Soporte incorporado para Google Cloud *Platform*, lo que facilita la integración de Google Cloud *Messaging* y *App Engine*. [15]

## Bibliografía

- [1] D. Ponce León and S. Santillán Aguirre, Artists, *Diseño e implementación de un sistema de transferencias de datos a través de la red eléctrica de baja tensión con la interfaz Labview-arduino empleando la tecnología power line communications (PLC)*. [Art]. Universidad Politécnica Salesiana, 2016.
- [2] R. Alonso Rodríguez, Artist, *Generador virtual de funciones de onda predefinida y arbitraria con herramientas hardware y software ARDUINO y LABVIEW*. [Art]. Universidad Valladolid, 2014.
- [3] H. X. Muñoz Zatzabal and G. F. Garcia Salazar, Artists, *CONSTRUCCIÓN Y ENSAMBLAJE DE UN BRAZO ROBOT A ESCALA*. [Art]. UNIVERSIDAD DE GUAYAQUIL, 1 marzo 2016.
- [4] F. A. Cerver, *Enciclopedia Atrium de la Plomeria Volumen 5 Domotica*, España: Axis Books,S.A., 1993.
- [5] "Real Academia Española," [Online]. Available: <http://www.rae.es/>. [Accessed 15 octubre 2017].
- [6] J. R. V. Orte, Artist, *Mejora de la eficiencia energética en viviendas domóticas*. [Art]. Universidad de la Rioja, 2014.
- [7] S. S. Aguirre Quiroz and . E. D. Mogollón Flores, Artists, *Diseño e implementación del sistema inmótico para el control de iluminación en el aeropuerto de Latacunga basado en la tecnología lonworks*. [Art]. ESCUELA POLITÉCNICA DEL EJÉRCITO, 2011.
- [8] J. R. Lajara Vizcaíno and J. Peligrí Sebastiá, *Lab VIEW Entorno gráfico de programación*, España: Alfaomega Grupo Editor, 2007.
- [9] LabVIEW, "National Instruments," [Online]. Available: <http://www.ni.com/es-mx/shop/labview.html>.
- [10] F. Reyes Cortés and J. Cid Monjaraz, *Arduino Aplicaciones en Robotica Mecatronica e Ingenierias.*, Mexico D.F.: Alfaomega Grupo editor, S.A. de C.V., 2015.
- [11] C. Sánchez, "Arduino, la revolución de las placas azules que se gestó en un bar italiano".*El diario.es*.
- [12] "Pagina oficial Arduino," [Online]. Available: <https://www.arduino.cc/>.

- [13] "Interfaz LabVIEW para Arduino (Lifa)," [Online]. Available: <https://forums.ni.com/t5/LabVIEW-Interface-for-Arduino/ct-p/7008>. [Accessed 26 septiembre 2017].
- [14] "Pagina Oficial de National Instruments," [Online]. Available: <http://www.ni.com/es-mx.html>.
- [15] [Online]. Available: <http://www.prensa.siemens.biz/index.php/notas-de-prensa/especializadas/539-siemens-lanza-detectores-de-movimiento-de-doble-tecnologia-para-reducir-falsas-alarmas>.
- [16] "Andriod Studio," [Online]. Available: <https://developer.android.com/studio/intro/index.html?hl=es-419>.
- [17] S. M. Anglés, Artist, *Metodología y criterios para evaluar la influencia de la domótica y su preinstalación en los edificios en función de los condicionantes constructivos y de la envolvente interior*. [Art]. Universidad Politécnica de Madrid, 2014.
- [18] "Electronica Embajadores," [Online]. Available: <https://www.electronicaembajadores.com/es/Productos/Detalle/LCA1008/modulos-electronicos/modulos-arduino/modulo-arduino-leonardo-r3>.
- [19] "Microcontroladores y Electrónica," 06 02 2014. [Online]. Available: <http://mikroe.es/labview-para-arduino/>.

## Apéndices. Conexiones físicas en la maqueta desarrollada.

### Apéndice A.

| Número físico para identificación | Descripción  | Numero de pin en la placa electrónica arduino |
|-----------------------------------|--|---|
| 1                                 | <b>Servo</b> que controla la <b>ventana de la cocina</b>         | <b>25 y 26</b>                                |
| 2                                 | <b>Servo</b> que controla la <b>puerta del baño</b>              | <b>4 y 11</b>                                 |
| 3                                 | <b>Servo</b> que controla la <b>puerta principal</b>             | <b>6 y 10</b>                                 |
| 4                                 | <b>Sensor</b> que controla la lampara de la <b>cocina y sala</b> | <b>38</b>                                     |
| 5                                 | <b>Sensor</b> que controla la lampara <b>externa</b>             | <b>44</b>                                     |
| 6                                 | Conexión de la <b>bocina</b> de la alarma                        | <b>45</b>                                     |
| 7                                 | Conexión de la <b>lampara del baño</b>                           | <b>34</b>                                     |
| 8                                 | Conexión de la <b>lampara externa</b>                            | <b>46</b>                                     |
| 9                                 | Conexión de la <b>lampara del cuarto 2.</b>                      | <b>35</b>                                     |
| 10                                | Conexión de la <b>lampara de la cocina y sala.</b>               | <b>39</b>                                     |

*Tabla 5 Conexiones físicas en el bus de datos del Baño*

## Apéndice B.

| Número físico para identificación | Descripción  | Numero de pin en la placa electrónica arduino |
|-----------------------------------|--|---|
| 1                                 | Conexión a tierra <b>(GND)</b> .                           | <b>(GND)</b>                                  |
| 2                                 | Conexión a voltaje de alimentación +5v <b>(VCC)</b> .      | <b>(VCC)</b>                                  |
| 3                                 | <b>Servo</b> que controla la <b>ventana del cuarto 2.</b>  | <b>24 y 27</b>                                |
| 4                                 | <b>Servo</b> que controla la <b>puerta del cuarto 2.</b>   | <b>3 y 9</b>                                  |
| 5                                 | <b>Servo</b> que controla la <b>puerta del cuarto 1.</b>   | <b>2 y 7</b>                                  |
| 6                                 | <b>Servo</b> que controla la <b>ventana de la sala.</b>    | <b>28 y 29</b>                                |
| 7                                 | <b>Servo</b> que controla la <b>puerta del garaje.</b>     | <b>5 y 8</b>                                  |
| 8                                 | <b>Servo</b> que controla la <b>ventana del cuarto 1.</b>  | <b>22 y 23</b>                                |
| 9                                 | Conexión de la <b>lampara del cuarto 1.</b>                | <b>37</b>                                     |
| 10                                | <b>Sensor</b> que controla la <b>lampara del cuarto 1.</b> | <b>36</b>                                     |
| 11                                | Conexión de la <b>lampara del garaje.</b>                  | <b>12</b>                                     |

*Tabla 6 Conexiones físicas en el bus de datos del garaje.*