

**PONTIFICIA UNIVERSIDAD JAVERIANA**  
**FACULTAD DE INGENIERÍA**

**CONTROLADOR INTELIGENTE PARA AHORRO DE ENERGÍA POR DETECCIÓN DE  
RUTINAS**

**Autor:**

**ANGÉLICA JEANNETTE MORALES DOMINGUEZ**

**Director:**

**ING. PEDRO VIZCAYA GUARÍN, Ph.D**

**JUNIO DE 2011**

**BOGOTÁ**

## *AGRADECIMIENTOS*

Doy a gracias a quiénes durante este largo proceso de esfuerzo y grandes méritos me acompañaron y apoyaron sin dudar de mis capacidades.

Agradezco al Ingeniero Pedro Vizcaya por su confianza, apoyo y colaboración durante el desarrollo de mi proyecto. Del mismo modo quiero agradecerle por los conocimientos que me brindó.

A mis padres infinitas gracias por brindarme la oportunidad de ser profesional, por hacerme sentir fuerte en los momentos de debilidad así como para hacerme recobrar fortaleza para continuar con mi carrera.

A mi amigo Juan Camilo Fajardo, quién me apoyó y colaboró para lograr lo que pretendía con el proyecto.

## TABLA DE CONTENIDOS:

<b>AGRADECIMIENTOS</b> .....	<b>2</b>
<b>1. INTRODUCCIÓN</b> .....	<b>6</b>
<b>2. MARCO TEÓRICO</b> .....	<b>7</b>
<b>2.1 MICROCONTROLADOR</b> .....	<b>7</b>
2.1.1 MICROCONTROLADOR ARDUINO UNO .....	7
<b>2.2 SISTEMA DE COMUNICACIÓN</b> .....	<b>8</b>
2.2.1 MÓDULOS XBee.....	9
<b>2.3 SENSORES</b> .....	<b>9</b>
2.3.1 SENSOR DE MOVIMIENTO ePIR.....	9
2.3.2 SENSOR DE LUZ .....	10
<b>2.4 SERVIDOR DE BASE DE DATOS</b> .....	<b>10</b>
2.4.1 SERVIDOR DE BASE DE DATOS MySQL.....	10
2.4.2 SERVIDOR DE BASE DE DATOS PRTG .....	11
<b>2.5 SERVIDOR WEB DE LABVIEW</b> .....	<b>12</b>
<b>3. ESPECIFICACIONES</b> .....	<b>13</b>
<b>3.1 DESCRIPCIÓN GENERAL Y DIAGRAMA EN BLOQUES</b> .....	<b>14</b>
<b>3.2 SOFTWARE Y HARDWARE</b> .....	<b>15</b>
3.2.1 VARIABLES DE ENTRADA .....	16
3.2.2 VARIABLES DE SALIDA .....	16
<b>4. DESARROLLO</b> .....	<b>17</b>
<b>4.1 PROGRAMA DESARROLLADO EN ARDUINO</b> .....	<b>17</b>
4.1.1 ENVÍO DE DATOS.....	18
4.1.2 RECEPCIÓN DE DATOS.....	18
4.1.3 JUSTIFICACIÓN DEL DESARROLLO DEL PROGRAMA DE ARDUINO .....	18
<b>4.2 DESARROLLO DE INTERFAZ EN LABVIEW</b> .....	<b>19</b>
4.2.1 PANEL FRONTAL.....	19
4.2.2 PRESENTACIÓN DEL CONTROL .....	19
4.2.3 CONFIGURACIÓN DEL PUERTO COM .....	20
4.2.4 SELECCIÓN DE ARCHIVOS.....	20
4.2.5 MUESTREO .....	20
4.2.6 AUTO.....	21
4.2.7 MANUAL .....	21
4.2.8 RECOMENDACIONES .....	22
4.2.9 RANGOS.....	22
4.2.10 DATOS DEL SISTEMA .....	23
4.2.11 LECTURA DE LAS RECOMENDACIONES.....	23
4.2.12 CONTROLADOR VISA .....	24
4.2.13 ESCRITURA SERIAL .....	24
4.2.14 LECTURA SERIAL.....	24
4.2.15 ESCRITURA EN UNA BASE DE DATOS Y EN ARCHIVO .CSV.....	25
4.2.16 LECTURA DE UNA TABLA DE UNA BASE DE DATOS .....	25

<b>4.3</b>	<b><i>RECOMENDACIONES</i></b> .....	<b>26</b>
<b>4.4</b>	<b><i>COMUNICACIÓN INALÁMBRICA XBEE</i></b> .....	<b>27</b>
<b>5.</b>	<b>ANÁLISIS DE RESULTADOS</b> _____	<b>28</b>
<b>6.</b>	<b>CONCLUSIONES</b> _____	<b>31</b>
<b>7.</b>	<b>BIBLIOGRAFÍA Y FUENTES DE INFORMACIÓN</b> _____	<b>32</b>

## **TABLA DE ANEXOS**

ANEXO 1- Arduino User Manual

ANEXO 2-ArduinoProgramming notebook

ANEXO 3-Características de Arduino

ANEXO 4-Esquemático Arduino

ANEXO 5- Programa en Arduino del CIAEDR

ANEXO 6- XBee-Datasheet

ANEXO 7-Primeros Pasos XBee y Arduino

ANEXO 8- XBee Manual Configuración Modulo Serie 2

ANEXO 9-Configuración básica Módulos XBee 2-Importante

ANEXO 10- Manual X-CTU

ANEXO 11-LabVIEW - Connect to MySQL

ANEXO 12-Labview Database Connectivity Toolset User Manual

ANEXO 13-Costos

ANEXO 14- Informe Registros

ANEXO 15- Informe FAR y FRR- Casa

ANEXO 16- Informe FAR y FRR- Consultorio

ANEXO 17- Informe FAR y FRR- Oficina

ANEXO 18- Manual MySQL 5.0

ANEXO 19-Datsheet ePir

ANEXO 20- Manual PRTG

ANEXO 21- Sensor ePir Vs. Ultrasonido

ANEXO 22-Recomendaciones C++

ANEXO 23- Caso1 Domótica

ANEXO 24- Caso2 Consultorio

ANEXO 25- Caso3 Oficina

ANEXO 26- Esquemático y PCB Eagle

ANEXO 27- Programa LabView

ANEXO 28- Imágenes tarjeta

## 1. INTRODUCCIÓN

En la actualidad los temas de automatización y gestión energética han generado un gran interés, incrementando de este modo la integración de la tecnología en el diseño inteligente de espacios o recintos cerrados. [6]

La domótica es la automatización aplicada a la gestión técnica de viviendas y edificios, su objetivo principal es mejorar la calidad de vida (seguridad, comodidad, ahorro de energía).

Dentro de los servicios que ofrece la domótica, este proyecto se enfoca en el servicio de ahorro de energía, siendo un concepto al que se puede llegar de muchas maneras. Lo que se propone con este proyecto es lograr obtener el uso eficiente de energía sobre los espacios de prueba (domótica, consultorio, oficina) a partir de las recomendaciones que se brinden, favoreciendo así al usuario, ya que éste no tendrá que sustituir aparatos o sistemas del hogar por otros que consuman menos, por lo que el usuario podrá continuar haciendo uso de sus mismos equipos y encontrar una reducción en consumo de energía eléctrica. Con dichas recomendaciones el usuario tendrá la opción de aplicar medidas correctivas al entorno monitoreado.

El *Controlador Inteligente para Ahorro de Energía por Detección de Rutinas* -CIAEDR se desarrolla en la plataforma de programación Arduino, poniendo en funcionamiento el sistema diseñado y recopilando la información detectada por los sensores de reconocimiento de actividad en el servidor de base de datos MySQL y a su vez en Excel con el fin de entrar a analizar dicha información por medio de una aplicación desarrollada en C/C++. Una vez es analizada la información se generan las recomendaciones y el respectivo control para el ahorro de energía.

Con el CIAEDR el usuario tendrá la posibilidad de controlar de manera remota su recinto. Este dispositivo brinda la opción de encender y apagar equipos y a su vez manipular la intensidad lumínica artificial. Por otro lado, ofrece recomendaciones sobre qué hacer con los equipos y luz en momentos de detección y no detección.

El objetivo principal de este proyecto es desarrollar un programa basado en la plataforma de programación Arduino para detectar actividades en diferentes ámbitos mediante sensores de detección conectados a dicha plataforma. Se recopila y analiza la información recibida por los sensores de donde se realizan las recomendaciones a partir de que haya o no detección así como de la intensidad lumínica.

El proyecto fue puesto a prueba en tres casos diferentes en una habitación de hogar, una oficina y un consultorio odontológico, donde se evaluó que efectivamente hubiera reconocimiento de presencia en estos diferentes espacios. La información fue recopilada por un servidor de base de datos donde luego se analizó con el programa desarrollado en C/C++ anteriormente mencionado.

Por otro lado, las variables que intervinieron en este proyecto fueron: periodo de tiempo, horarios y control/recomendación del consumo de energía (para un uso eficiente de la misma); donde las dos primeras terminan siendo dependientes al finalizar el proceso de detección y la tercera dependiente de los resultados de detección.

Ahora bien, del CIAEDR se resaltan beneficios importantes con incidencia a nivel global, ya que al estar relacionado con temas como el ahorro de energía se pueden obtener beneficios tales como ecológicos y económicos.

Si bien día a día un gran porcentaje de la humanidad se ha sensibilizado con temas relacionados con el calentamiento global, el desarrollo de este proyecto busca aportar a la reducción de consumos energéticos. ¿Cómo? A partir de las recomendaciones que se brindan con base en los resultados obtenidos de los sensores y el control desarrollado para la manipulación de equipos. La finalidad del CIAEDR es lograr una disminución en el consumo de energía, racionando adecuadamente el uso de la misma. Al brindar recomendaciones para reducir consumos de energía se ofrece al mismo tiempo un beneficio de tipo económico ya que estas son variables directamente proporcionales, en las cuales si se reduce el consumo de energía aplicándose un uso eficiente, mínimo y correcto igualmente se reducirá la variable económica. Sin embargo el usuario tendrá la opción de tener o no en cuenta las recomendaciones que se le brinden.

De lo anterior, se observa que con el CIAEDR se persigue optimizar el uso de energía, obteniendo recíprocamente una reducción de costos en servicios energéticos, beneficiando así a usuarios que por su estrato social subsidian estratos inferiores.

Este proyecto sirve de base para futuros proyectos con enfoque a reducción de consumos energéticos.

## **2. MARCO TEÓRICO**

En los siguientes apartados se definen los conceptos básicos necesarios para el entendimiento y comprensión del desarrollo, diseño y funcionamiento del controlador inteligente para ahorro de energía por detección de rutinas.

### *2.1 MICROCONTROLADOR*

Un microcontrolador es un circuito integrado programable diseñado especialmente para controlar sistemas electrónicos, este realiza las operaciones de control a partir de las funciones ingresadas por el programador. Los microcontroladores disponen de: sistema de Memoria (RAM y/o ROM), CPU, Conversor Análogo-Digital, temporizadores, sistema de comunicaciones serial, entre otros.

A continuación se da una breve descripción del microcontrolador a utilizar en el CIAEDR.

#### *2.1.1 MICROCONTROLADOR ARDUINO UNO*

Atmel es una industria líder especialmente en el mercado de las tarjetas seguras, debido a que sus microcontroladores resultan inmunes frente a ruidos y contaminación electromagnética. Esto es una gran ventaja frente a otras famosas marcas productoras de semiconductores. El Arduino Uno incluye el ATmega 328p que maneja la arquitectura AVR 32. Gracias a este microcontrolador esta placa (Arduino UNO) es robusta e ideal para el desarrollo de aplicaciones.

Es interesante destacar que existen otros microcontroladores con capacidades y características similares a las de la familia AVR tal es el caso de PIC, un microcontrolador de Microchip. Los microcontroladores PIC tienen dentro de sus méritos el ser muy populares y por lo tanto encontrar ejercicios y ejemplos en la

web, pero con la gran acogida mundial de la placa Arduino la documentación que se encuentra orientada a los microcontroladores de AVR es inmensa y está en continuo crecimiento.

Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en aplicaciones y dispositivos (hardware) flexibles y fáciles de usar. [7] Al ser hardware libre, tanto su diseño como su distribución es gratuito, por lo que puede utilizarse para el desarrollo de cualquier tipo de proyecto sin haber adquirido ninguna licencia.

La plataforma Arduino Uno es una de las ofrecidas por Arduino. Esta cuenta con 14 terminales de entrada/salida digital (de los cuales 6 pueden ser utilizados como salidas PWM), 6 entradas analógicas y una conexión USB la cual permite conexión serial con el computador u otros microcontroladores. A partir de la conexión USB se pueden transmitir los datos a un computador donde estos se podrán analizar por pantalla. Para este proyecto, la plataforma no irá conectada por puerto USB al computador, pues se cuenta con dos módulos XBee los cuales permiten el envío de datos de forma inalámbrica. Por otro lado, en caso de ser necesario algún tipo de modificación u otra función, ésta puede realizarse por esta misma conexión.

En el ANEXO 1 y 2 se incluyen manuales de Arduino. El ANEXO 1 manual del usuario y en el ANEXO 2 manual de programación de Arduino, los cuales fueron consultados para el desarrollo y programación del dispositivo.

## 2.2 SISTEMA DE COMUNICACIÓN

Un sistema de comunicación se refiere a la transferencia de datos desde un lugar (origen, fuente o transmisor) a otro (destino o receptor), a través de un medio o canal de transmisión. En el diagrama de la *ilustración 1* se representa esquemáticamente el proceso de comunicación. En la “fuente de información” se genera el mensaje o los datos que se van a transmitir, luego en el “transductor de entrada” se codifica el mensaje para poder ser transmitido. Una vez codificado el mensaje este se envía por el “transmisor” el cual lleva lo lleva por el canal o medio de transmisión para que este llegue a su destino. El mensaje es tomado por el “receptor” y luego enviado al “transductor de salida” el cual cumple la función inversa del transductor de entrada (decodificar el mensaje) para entregarlo a su “destino”.



**Ilustración 1. Diagrama de bloques de Sistema de Comunicación**

Existen varios medios para transmitir la información, estos pueden ser fibra óptica, par trenzado, cable coaxial e inalámbrico. Para este proyecto se utilizará la comunicación inalámbrica. Con el fin de lograr dicha comunicación se hace uso de los módulos XBee.



### 2.2.1 MÓDULOS XBee

Los módulos XBee son módulos de radio frecuencia que trabajan en la banda de 2.4 GHz basados en el estándar de comunicaciones IEEE 802.15.4 fabricados por MaxStream. Este realiza las comunicaciones a través de una única frecuencia (canal) y cuenta con una velocidad de transmisión aproximadamente de 256 kbps. El alcance de estos módulos con potencia de 1mW en exteriores es de 100 m y en interiores 30 m.



Ilustración 2. Módulo XBee

Estos módulos son comúnmente utilizados en automatización de casas (Domótica), sistemas de seguridad, monitoreo de sistemas remotos, aparatos domésticos, alarmas contra incendio, plantas tratadoras de agua, etc.

La comunicación de ZigBee (o XBee) puede ser punto a punto, punto a multipunto o peer-to-peer (P2P). En el presente proyecto se hace uso de dos módulos para la comunicación inalámbrica. La comunicación entre módulos es punto a punto, donde el modulo XBee conectado a la plataforma Arduino UNO tendrá comunicación serial con esta. Por otra parte, el segundo modulo conectado por puerto USB al computador será quien reciba los datos obtenidos por el modulo conectado directamente a la plataforma. Los datos obtenidos de los sensores por la plataforma Arduino UNO son visualizados por la interfaz diseñada en LabVIEW, dichos datos se ingresan a una base de datos en MySQL (un sistema de gestión de base de datos) de donde se entra a analizar lo obtenido para luego realizar las recomendaciones y control pertinente.

### 2.3 SENSORES

Los sensores son dispositivos capaces de detectar magnitudes físicas o químicas, llamadas variables de instrumentación, y ser transformadas en señales eléctricas. Las variables de instrumentación pueden ser: Temperatura, Lúmenes, Presión, etc.

Para el desarrollo de este proyecto se utilizarán los sensores mencionados a continuación.

#### 2.3.1 SENSOR DE MOVIMIENTO ePIR

El ePIR de Zilog es un dispositivo de detección de movimiento contenido en una sola tarjeta SBC (single-board computer). Está conformado por un XP MCU (microcontrolador XP) Z8 ENCORE, y un sensor infrarrojo pasivo Murata (PIR) el cual está protegido por un lente de Fresnel.



Ilustración 3. Sensor de Movimiento ePIR

Este sensor proporciona dos modos de interfaz: una interfaz de hardware simple y la otra con un método más avanzado serial. Si se usa la interfaz de hardware, el pin de detección de movimiento MD (motion detect) pasará de “1” a “0” cuando se detecte movimiento. En el modo serial el ePIR se lee y configura con el conjunto definido de comandos –todos enviados de forma asíncrona.

Para el CIAEDR el ePIR será utilizado en modo de interfaz de hardware para la detección de movimiento en el lugar bajo monitoreo. Este irá conectado a la plataforma Arduino UNO a una de sus terminales de entrada digital la cual a su vez irá conectada en modo serial al modulo XBee.

En el ANEXO 21 se especifica porque fue elegido este sensor en lugar de uno de ultrasonido.

### 2.3.2 *SENSOR DE LUZ*

El sensor de luz responde a los cambios de intensidad de luz en el espacio bajo monitoreo. Dicho sensor se usa para detectar el nivel de luz y producir una señal de salida representativa respecto a la cantidad de luz detectada. Un sensor de luz incluye un transductor fotoeléctrico que convierte la luz en una señal eléctrica.



**Ilustración 4. Sensor de Luz**

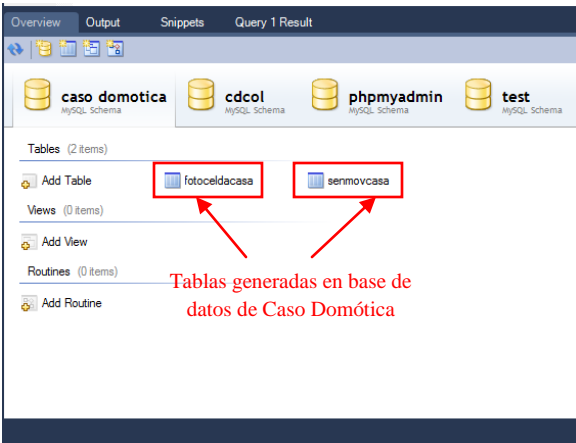
El sensor de luz cambia su resistencia dependiendo de la cantidad de luz a la que es expuesto por lo que sus valores son variables. A partir de la detección se genera una señal que luego es procesada por la plataforma Arduino UNO y expuesta en la interfaz diseñada para su visualización. Para visualizar los datos en la interfaz, la fotocelda es conectada a una de las terminales de entrada análogas del Arduino UNO. La plataforma envía los datos de manera inalámbrica al modulo XBee conectado al puerto USB del PC donde los datos son registrados en la base de datos MySQL y en la interfaz realizada en LabView.

## 2.4 *SERVIDOR DE BASE DE DATOS*

Los servidores de bases de datos surgen debido a la necesidad de almacenar y manejar grandes volúmenes de datos. Es una colección estructurada de datos. Puede ser cualquier cosa, desde una simple lista de compra a una galería de pintura o las más vastas cantidades de información en una red corporativa. [8] Muchas veces la información almacenada requiere de ser compartida con un conjunto de personas autorizadas (usuarios). El servidor de bases de datos brinda soluciones tales como fiabilidad, rentabilidad y alto rendimiento al ser dedicado. Su modelo está basado en el de cliente-servidor, ya que consiste básicamente en peticiones de un cliente a un servidor que le da respuesta. El servidor pone a disposición del cliente sus recursos.

### 2.4.1 *SERVIDOR DE BASE DE DATOS MySQL*

MySQL es el sistema de gestión de bases de datos de código abierto SQL más popular, lo desarrolla, distribuye y soporta MySQL AB. Es un sistema de gestión de bases de datos relacionales.



**Ilustración 5. Base de Datos MySQL**

Una base de datos relacional almacena datos en tablas separadas en lugar de poner todos los datos en un gran almacén. Esto añade velocidad y flexibilidad. La parte SQL de “MySQL” se refiere a “Structured Query Language”, lenguaje estandarizado más común para acceder a bases de datos y está definido por el estándar ANSI/ISO SQL.

Las bases de datos generadas se realizan a partir de los datos obtenidos de los sensores. La base de datos se divide en dos tablas, una corresponde a la del sensor de detección y la otra a la fotocelda. Ambas tablas constan de: Fecha, Hora y el valor obtenido.

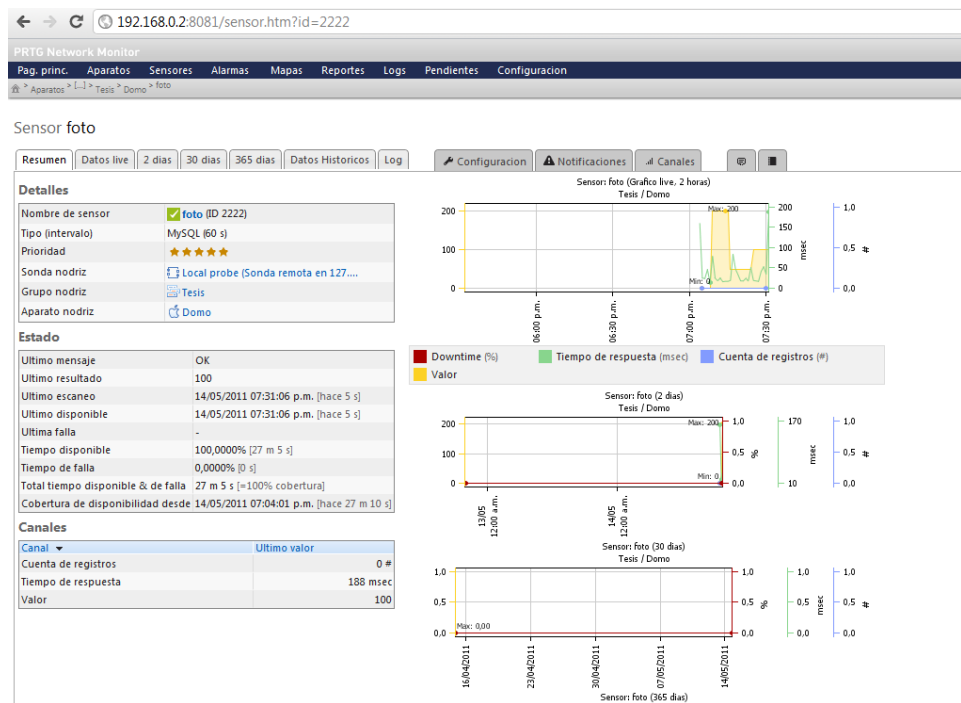
A partir de los datos almacenados se detecta la rutina del espacio monitoreado y de ahí las recomendaciones y control (de ser necesario) pertinentes.

#### 2.4.2 SERVIDOR DE BASE DE DATOS PRTG

El PRTG es utilizado principalmente para el monitoreo de redes, para el CIAEDR se permite la implementación de este software para brindarle al usuario la opción de visualizar en tiempo real los datos obtenidos por los sensores de manera remota.

Esta aplicación toma los datos cada minuto desde la base de datos generada en MySQL y los publica en red a través de un servidor

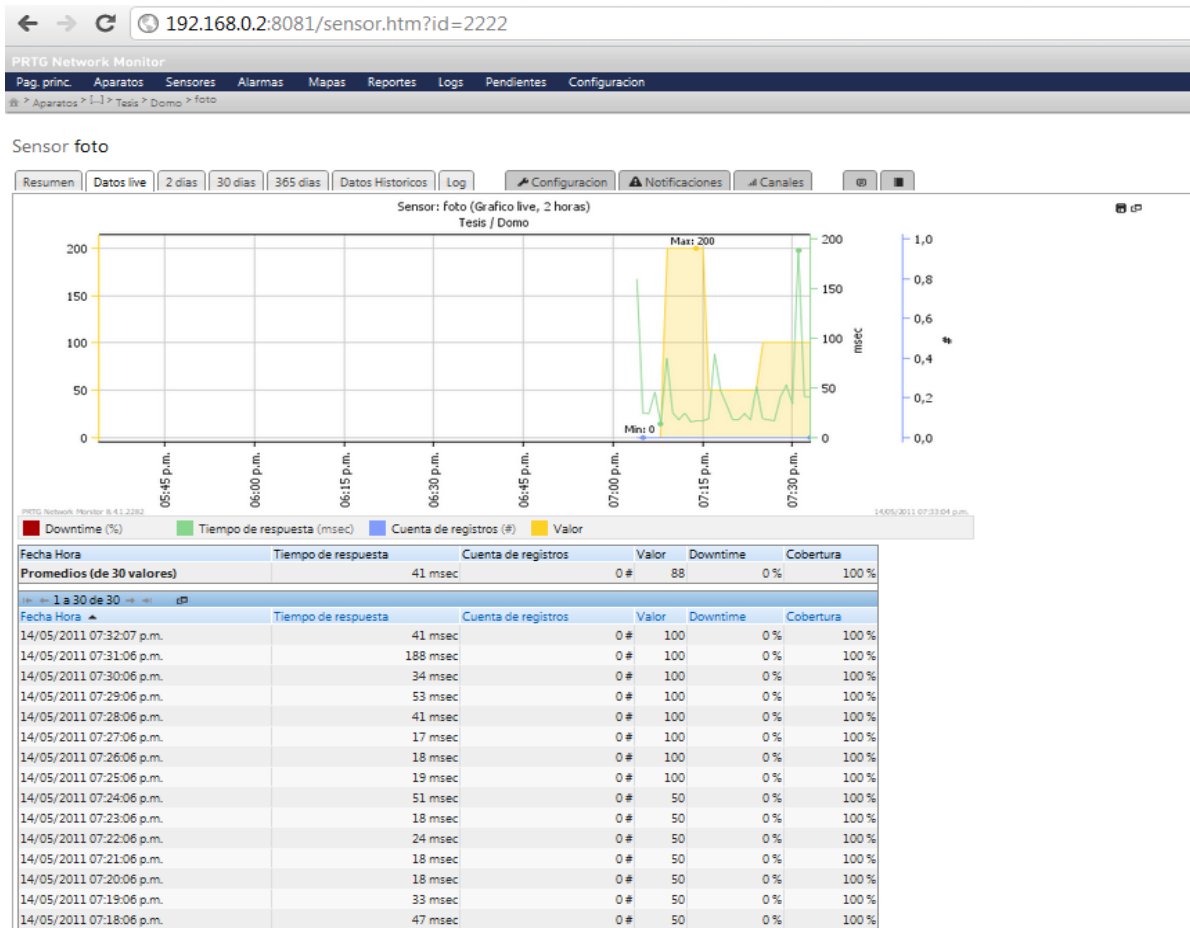
web el cual viene incorporado para proveer al usuario acceso remoto y el acceso a gráfico y tablas.



**Ilustración 6. Interfaz PRTG**

Con esta aplicación se logran crear grupos de sensores incorporados en un mismo sistema, por lo que se podrá visualizar la información de cada sensor, del mismo modo, al ser almacenados los datos se brinda la

opción de poder analizarlos con posterioridad. El usuario podrá configurar los parámetros bajo monitorización y añadir en su momento más dispositivos para ser monitorizados.



**Ilustración 7. Interfaz PRTG base de datos**

## 2.5 SERVIDOR WEB DE LABVIEW

El servidor web ofrecido por LabVIEW se le conoce como la herramienta *Web Publishing*. Este servidor se ejecuta continuamente en el computador, manteniéndose a la espera de peticiones de ejecución del cliente de manera remota, a través de la web. El servidor web es quien se encarga de contestar a las peticiones realizadas por el usuario así como ejecutar los comandos solicitados por el mismo, permitiéndole ver los resultados.

Este servidor web permite publicar el panel frontal de una aplicación en un documento HTML, para que el usuario pueda verla y controlarlo de manera remota, a través de un web browser. De acuerdo a la configuración seleccionada, el servidor web comienza a crear imágenes estáticas o animadas; así mismo como un reporte, que incluye tablas, listas, información del operador y gráficas con fecha y hora, entre otras.

Las imágenes que genera, en formato JPG y PNG, de los paneles frontales, permiten diferentes grados de compresión y detalle gráfico, en los reportes de monitoreo de la aplicación.

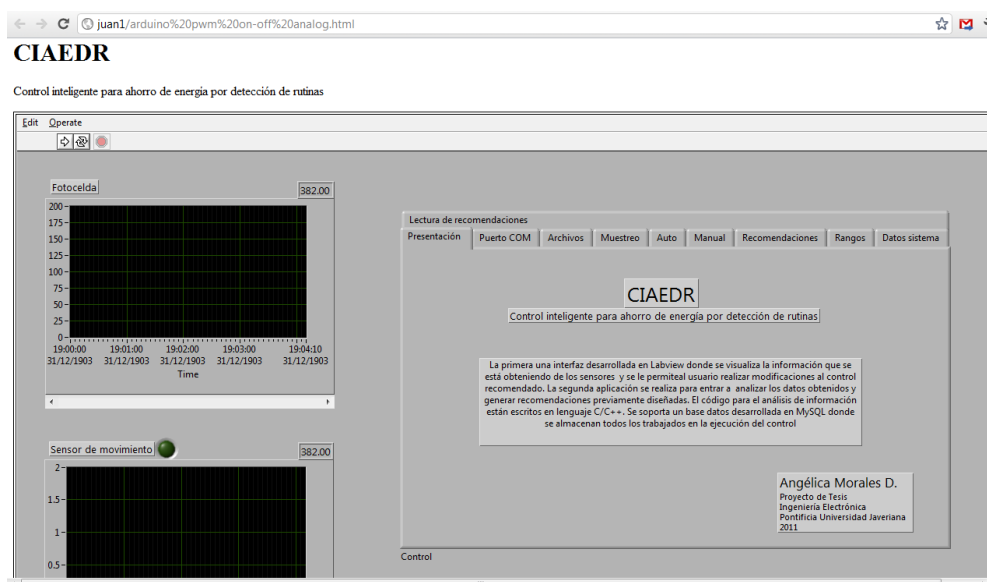


Ilustración 8. Interfaz de LabView visualizada por red.

### 3. ESPECIFICACIONES

Se diseñó y construyó un sistema para detectar presencia e intensidad lumínica en diferentes espacios con el fin de generar recomendaciones y control. Por otra parte se desarrolló una interfaz con el fin de brindarle al usuario un manejo completo y remoto del sistema.

A continuación en la Tabla 1 se presentan las especificaciones del sistema del enlace de comunicación, de la placa Arduino Uno y el tamaño del dispositivo CIAEDR. En ella se especifica la distancia de transmisión a línea de vista y en recinto cerrado, así como su canal y frecuencia de transmisión, del mismo modo se muestran especificaciones de la placa Arduino Uno. En los ANEXOS 5 y 1 se encuentran más especificaciones tanto del modulo XBee como de la placa Arduino Uno respectivamente.

MODULO XBEE-ENLACE DE COMUNICACIÓN		
ESPECIFICACIÓN	GENERAL	PARA EL CIAEDR
Alcance de Transmisión	A línea de vista	Hasta 100 m (300 ft)
	Recinto cerrado	Hasta 30 m (100 ft)
Número de canales de transmisión	16 Canales	Transmisión por el canal 13
Frecuencia de transmisión	$F_c = (2405 + 5 \cdot (k-11))$ MHz k= canal de transmisión	$F_c = 2415$ MHz
Topologías de red soportadas	Punto Punto, Punto a Multipunto y Peer-to-Peer (P2P)	Punto a Punto
Alimentación	2,8 – 3,4 V	3,3 V
PLACA ARDUINO		
Terminales	14 terminales de entrada/salida digital (6 pueden ser usadas como PWM) y 6 entradas analógicas	1 Entrada Análoga (Fotocelda), 1 Entrada Digital (ePIR), 2 Salidas PWM (Led y Alarma) y 1 Salida Digital (Relevo).
Alimentación		12 VDC

Tabla 1. Especificaciones de Módulo XBee y Plataforma Arduino

### 3.1 DESCRIPCIÓN GENERAL Y DIAGRAMA EN BLOQUES

La finalidad del CIAEDR es aportar al usuario recomendaciones válidas (que vayan acuerdo a lo que se detecta y visualiza se está obteniendo por la monitorización del recinto) para ser implementadas, así como la facilidad de controlar su sistema de manera remota haciendo las modificaciones que él considere pertinentes. De este modo lo que se logra es que él tenga un total manejo de su entorno (encendido y apagado de equipos, intensidad de luz, etc.).

Aportar al tema de ahorro de energía, es lo que se busca realizando el CIAEDR. El proyecto se ha diseñado con base en la plataforma de programación Arduino donde una de las principales ventajas que se le brinda al usuario es que maneja una aplicación libre, permitiendo de este modo que él tenga la libertad de re-acoplar o rediseñar el sistema a sus necesidades cuantas veces sea necesario sin ningún sobrecosto.

Así mismo, el proyecto se fundamenta en la detección de rutinas en tres diferentes recintos mediante sensores basados en reconocimiento de actividad. Los entornos en los que se realiza la detección y recopilación de datos son: habitación de hogar, consultorio odontológico y oficina de trabajo. Antes de efectuar las recomendaciones o controles pertinentes a los diferentes casos de trabajo se realizó un previo análisis del espacio. Para la detección se utilizaron diferentes sensores, tales como: ePIR (sensor de detección) y fotocelda. Una vez detectadas las señales obtenidas por los sensores, se procedió a la recopilación de las mismas. La información obtenida se almacenó en MySQL una base de datos con el propósito de luego ser analizada para realizar recomendaciones y control sobre el consumo de energía. Adicionalmente, los datos obtenidos son visualizados por un monitor de red conocido como PRTG, este se especializa en la adquisición de información brindada por los sensores conectados a un sistema con la ventaja de ser obtenidos en tiempo real, por lo que le permite al usuario mantenerse informado sobre su espacio bajo monitoreo.

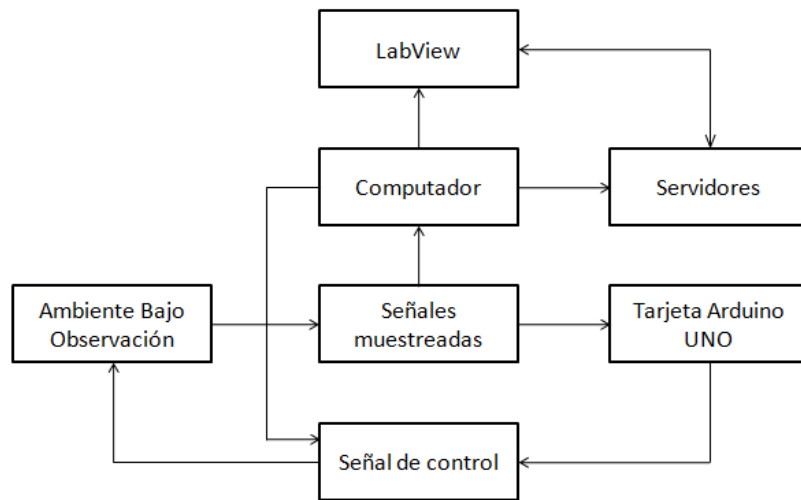
El análisis de la información recopilada se realiza mediante un programa diseñado en C/C++. A partir de la recopilación datos se puede saber la continuidad con que cierto lugar es utilizado, al igual que la hora en que este se utiliza, etc. ofreciendo luego unas recomendaciones para que el usuario a partir de ellas realice y aplique medidas de control (manipular el encendido y apagado de los equipos que tenga conectados al microcontrolador, variar la intensidad de luz en el recinto y modificar el muestreo con el que se están tomando los datos). De igual manera que no tenga que preocuparse por el encendido y apagado de los equipos que tenga conectados al dispositivo Arduino UNO (ej. Aire Acondicionado: si no se encuentra alguien en casa, lo ideal sería tenerlo apagado. Sin embargo para el momento en que se llegue a casa este se encuentre encendido, por lo tanto el CIAEDR encendería automáticamente el A/C previamente a su llegada, así encontrará su hogar climatizado).

La tarjeta Arduino UNO posee 14 terminales de entrada/salida digital y 6 entradas análogas donde se encuentran conectados los sensores a ella los cuales son programados en lenguaje de programación Arduino. Sin embargo, cabe resaltar que esta tarjeta soporta varios lenguajes de programación tales como Java, Python, Puredata, etc. Cada sensor tiene asignado una terminal de entrada, por donde se reciben las señales de cada uno. El ePIR se encuentra en la terminal digital 7, mientras que el relé en la 3. La fotocelda al brindar una señal análoga se encuentra en la terminal análoga A0, por otro lado para el lado de aviso y control se encuentran la alarma se en la terminal PWM 6 y el led en la 5.

Tras asignar los terminales de entrada/salida y poner a prueba el circuito con el programa diseñado se inició la adquisición y almacenamiento de información. Dicha información se recibió de forma inalámbrica, donde la comunicación se efectuó mediante las tarjetas XBee.

Se encuentra una tarjeta XBee conectada al puerto USB del computador de monitoreo por donde se reciben las señales de los sensores y envían las señales de control. Por otro lado, a través del computador de monitoreo se permite la visualización de las señales obtenidas por los sensores al igual que las recomendaciones diseñadas y el manejo de modificaciones por el usuario. Por medio de la aplicación desarrollada en LabView se permite el control del sistema mediante las terminales de salida del microcontrolador.

A continuación se muestra un diagrama en bloques para un mejor entendimiento de lo que se va a realizar en el proyecto.



**Ilustración 9. Diagrama en Bloques del CIAEDR**

### 3.2 SOFTWARE Y HARDWARE

El hardware consiste en las plataformas físicas. El dispositivo que se desarrolló para la detección de presencia e intensidad lumínica está compuesto por el ePir de Zilog, una fotocelda, un relevo, una alarma y dos led. Este circuito se acopla a la placa del XBee que a su vez está conectada al Arduino UNO. Los dos módulos XBee componen el hardware del enlace de comunicación. El sistema de alimentación del circuito está compuesto por un adaptador con entrada de 100-240 VAC y salida de 12 VDC/600 mA.

Para la visualización y análisis de información se han desarrollado dos aplicaciones. La primera una interfaz desarrollada en Labview donde se visualiza la información que se está obteniendo de los sensores y se le permite al usuario realizar modificaciones al control recomendado. La segunda aplicación se realiza para entrar a analizar los datos obtenidos y generar recomendaciones previamente diseñadas. El código para el análisis de información está escritos en lenguaje C/C++. A continuación se presentan las variables de entrada y salida del sistema.

### 3.2.1 VARIABLES DE ENTRADA

Puesto que es necesaria la detección de presencia al igual que la intensidad de luz para este proyecto, la detección se logra por medio de un sensor de ePIR y una fotocelda para la detección de intensidad de luz. A partir de estos se obtienen valores de detección (0- hay presencia ó 1- no hay presencia) y de intensidad lumínica.

#### 3.2.1.1 SENSOR DE LUZ

La Fotocelda es la encargada de capturar la intensidad de luz en el ambiente bajo monitoreo, cuenta con un circuito operado por un amplificador operacional con la función de linealizar el resultado adquirido por esta. Es una variable de entrada al Arduino, el valor que esta entrega es de 0 a 1023 considerándose como una entrada análoga.

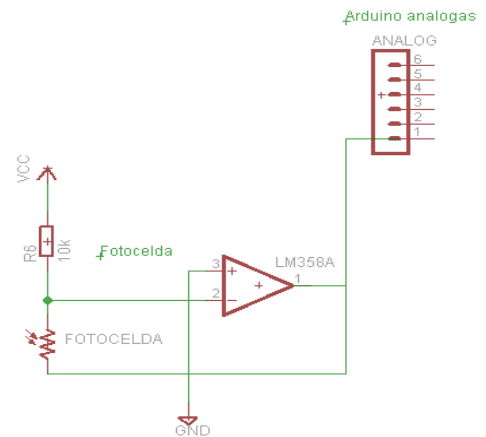


Ilustración 10. Esquemático de Fotocelda

#### 3.2.1.2 SENSOR DE MOVIMIENTO ePIR

Este sensor de movimiento maneja un estado discreto que lo hace ser una señal digital de entrada al sistema, cuenta con un circuito que mantiene el estado de activación por cierto tiempo, previamente este debe ser configurado por medio de voltajes mediante la conexión de estos dos pines que le garantizan el tiempo de activación.

### 3.2.2 VARIABLES DE SALIDA

Debido a que se genera un control para lograr el ahorro de energía, se hizo necesaria la implementación de un relevo así como un led de intensidad variable con el fin de mostrar como funcionaría el circuito y a lo que se llega con el proyecto. El led de intensidad variable se implementa ya que una de las cosas que se busca con el fin de ahorrar energía es poder variar la intensidad lumínica a la realmente necesaria cuando se presenta el caso de alta intensidad lumínica. Por otro lado, la implementación de la alarma se hace con el fin de alertar al usuario de que no se está detectando movimiento por lo que entrará a hibernar el equipo sobre el cual él quiere que haya control. La alarma se genera puesto que puede existir la posibilidad de que el sensor de movimiento deje de detectar al usuario, entonces, para que el sistema no apague o ponga a hibernar inmediatamente el equipo se genera la alarma sonora con el fin de llamar la atención del usuario para que este se mueva y el sistema vuelva a detectarlo. (**Nota:** La idea de hibernar el equipo se podría



desarrollar en futuros proyectos, cabe resaltar que el poner a hibernar un equipo no se realiza en este proyecto).

#### *3.2.2.1 LED DE INTENSIDAD VARIABLE*

Es una salida digital del sistema controlada por PWM (Modulación del ancho de pulso), esta se ha dispuesto con el propósito de mostrar el ajuste en la intensidad de una señal lumínica. Puede recibir valores de 0 a 255, donde 0 es apagado y 255 es el máximo brillo logrado por este dispositivo emisor de luz.

#### *3.2.2.2 ALARMA*

Salida digital del sistema controlada por PWM (Modulación del ancho de pulso), esta se ha dispuesto con el propósito de poder generar una alarma audible a determinada reacción en el sistema. Puede recibir valores de 0 a 255, donde 0 es apagado y 255 es el máximo tono logrado por este zumbador.

#### *3.2.2.3 RELÉ*

Salida digital con el propósito de simular el encendido/apagado de una dispositivo de potencia. Mediante la activación de la bobina interna del relé es posible generar un campo electromagnético que conmuta el contacto interno produciendo dos estados abrir/cerrar el contacto.

### **4. DESARROLLO**

En este apartado se presenta la descripción de lo que se desarrolló para el proyecto. Se muestra el desarrollo para el diseño del código a implementar en la placa Arduino, así como el de la interfaz en Labview. También se muestra como se realiza la comunicación y configuración entre módulos XBee. La justificación de los componentes elegidos, una breve comparación entre un microcontrolador Arduino y un PIC, el código del software para análisis de datos y un diagrama de flujo de cómo se realizan las recomendaciones.

#### *4.1 PROGRAMA DESARROLLADO EN ARDUINO*

El desarrollo implementado en el código de la placa Arduino consiste en dos funciones básicas, enviar y recibir datos a través del puerto serial, con ellas es posible comunicarlo con cualquier programa que se

encuentre en la capacidad de enviar y recibir una trama de datos que corresponda a los parámetros esperados por programa, a continuación se presentan las dos funciones básicas:

#### 4.1.1 ENVÍO DE DATOS

Consiste en la transferencia de los datos captados por los sensores conectados al Arduino, por ello se transfiere los valores de la fotocelda y el estado del sensor de movimiento, esta información es recibida por LabView donde es analizada y almacenada en una base de datos.

El programa de Arduino envía los datos de este modo:

a	a	a	a		b	/0
---	---	---	---	--	---	----

aaaa = Valor entero de la Fotocelda de 0 a 1023

\_ = Espacio en blanco para separar los datos

b = Valor entero del sensor de movimiento 0 ó 1

/0 = Indica el fin de ese paquete de datos

#### 4.1.2 RECEPCIÓN DE DATOS

Capta la información enviada por LabView. Mediante una trama codificada se reciben los valores que ajustan el estado de los periféricos de salida conectados al Arduino.

El programa desarrollado en Arduino espera recibir los datos de la siguiente manera:

c	c	c	d	e	e	e	f
---	---	---	---	---	---	---	---

ccc = Valor entero para el ajuste del brillo al Led PWM 0 a 255

d = Valor entero para encender/apagar el relevo 0 ó 1

eee = Valor entero para el ajuste del tono de la alarma 0 a 255

f = Diversas pruebas y lecturas del puerto serie, indican que al enviar Labview la información al Arduino en el final del paquete se envía una “,” (coma), con ello surge la necesidad de reservar un espacio para la lectura de este carácter.

#### 4.1.3 JUSTIFICACIÓN DEL DESARROLLO DEL PROGRAMA DE ARDUINO

El modelo enunciado anteriormente es de gran utilidad puesto que al realizar una comunicación serial se debe garantizar que se envíe o reciba un dato a la vez, debido a la distribución y el número de periféricos

de entrada y salida del sistema conectado al Arduino se requiere manejar el envío y recepción de paquetes de datos. El envío o recepción de un paquete de datos hace posible el manejo de un gran volumen de información que hacen que la ejecución de órdenes dentro del sistema sea rápida, a pesar de que se esté trabajando sobre una comunicación serial. El ANEXO 5 describe y especifica paso a paso el programa diseñado en Arduino Uno.

## 4.2 DESARROLLO DE INTERFAZ EN LABVIEW

### 4.2.1 PANEL FRONTAL

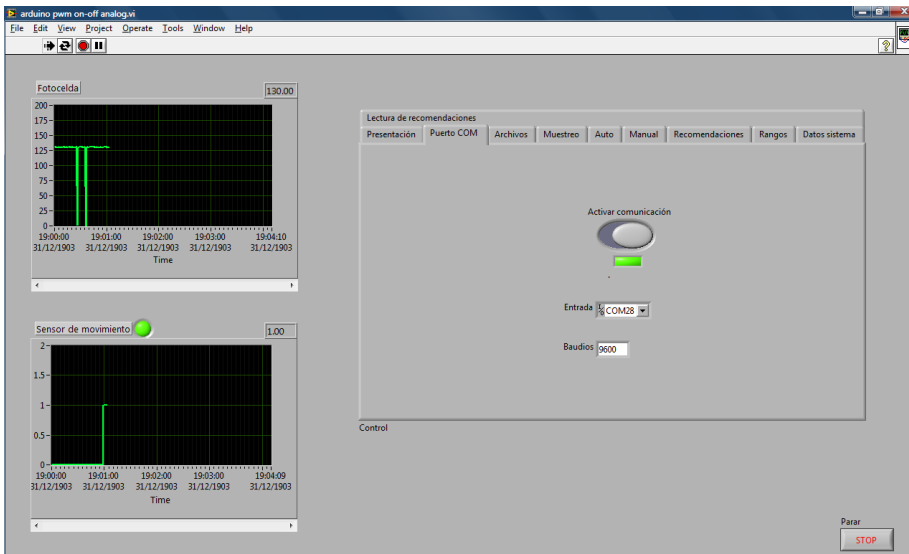


Ilustración 11. Panel de Control LabView

gráficas que corresponden a la Fococelda y al sensor de movimiento, en las próximas imágenes podremos ver más en detalle el control del software.

Programa desarrollado con la programación gráfica de LabView, dicho programa tiene varios bloques de trabajo tales como:

- Conexión serial.
- Escribir y leer el puerto serie.
- Escribir y leer tablas de una base de datos.

A continuación se presenta el panel frontal, en donde podemos visualizar las

### 4.2.2 PRESENTACIÓN DEL CONTROL



Ilustración 12. Presentación del Control del CIAEDR

### 4.2.3 CONFIGURACIÓN DEL PUERTO COM

Permite la configuración del puerto COM a usar así como la velocidad de transferencia en baudios, cuenta con un botón de activación de comunicación, para un mayor control del usuario.

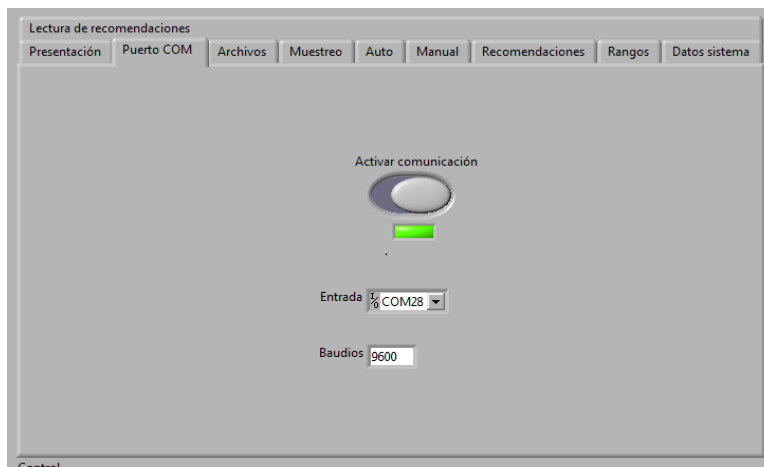


Ilustración 13. Configuración del puerto COM

### 4.2.4 SELECCIÓN DE ARCHIVOS

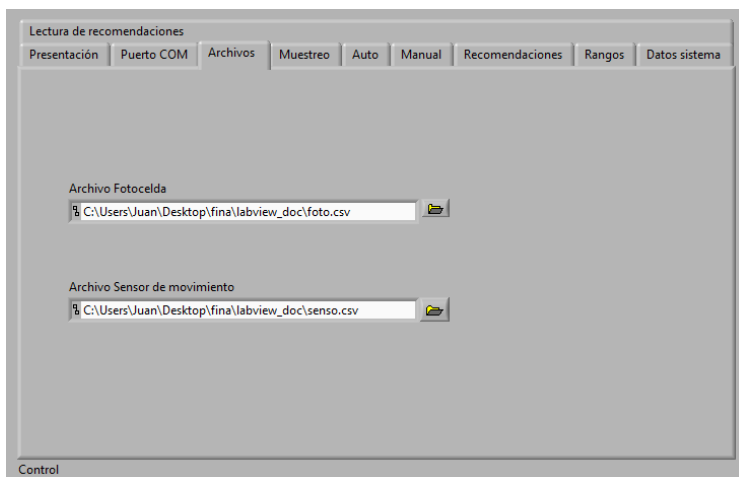


Ilustración 14. Pestaña Archivos, para seleccionar ruta de archivo

El valor del sensor de movimiento así como el de la fotocelda pueden ser almacenados en unos archivos separados por comas (.csv), el usuario cuenta con la posibilidad de escoger la ruta del archivo a guardar.

### 4.2.5 MUESTREO

Un dinámico y amigable control permite escoger que tantos segundos se desean para hacer el muestreo y el posterior almacenamiento en la base de datos y en el archivo .csv.

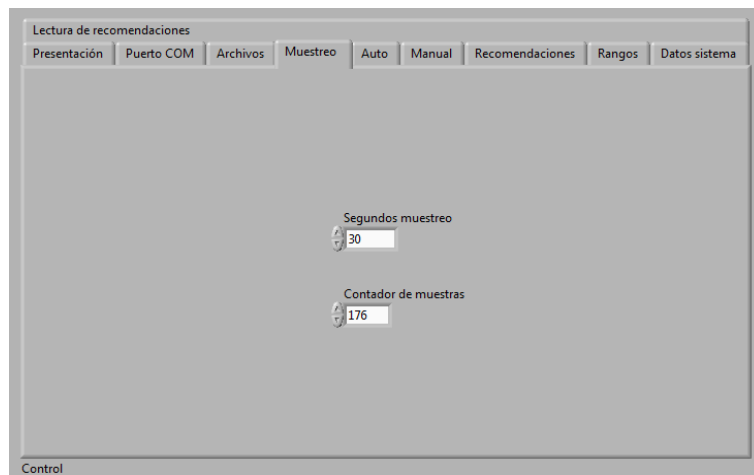


Ilustración 15. Pestaña Muestreo

#### 4.2.6 AUTO

En esta pestaña se encuentran funciones dedicadas al usuario, al cual se le permite la opción de manipular el sistema en caso tal que desee dejar su sistema inactivo luego de cierto tiempo, se cuenta con las siguientes funcionalidades:

**Apagar el sistema en:** 5min, 10min, 30min, 1hora, 4horas y nunca son las posibilidades del usuario para dejar el sistema inactivo luego de ese tiempo. Al

seleccionar el periodo que desea el usuario pase una vez transcurrido se apagan los equipos asociados al CIAEDR.

**Tiempo para hibernar:** Luego de que el sistema detecte inactividad por 30 min, tiempo durante el cual no hay detección de movimiento por parte del sensor de movimiento, el sistema ingresaría en periodo de hibernación, donde se apagarían los periféricos asociados. (**Nota: Como se mencionó anteriormente, la hibernación de equipos no se desarrolló para este proyecto, se deja para futuros proyectos**).

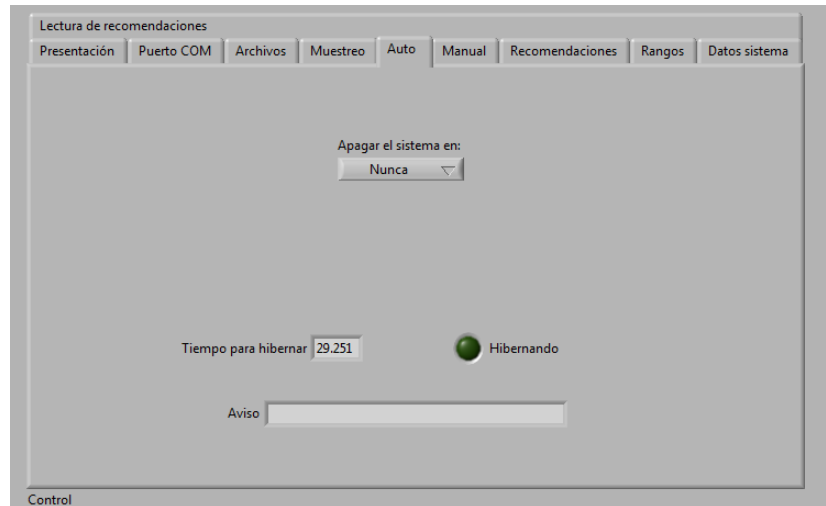


Ilustración 16. Pestaña Auto

#### 4.2.7 MANUAL

En esta pestaña el usuario encuentra el control para encender/apagar el relevo de manera remota o para ajustar el brillo del LED variable. Con esto lo que se logra es que el usuario manipule de manera inmediata y remota los equipos controlados por el CIAEDR.

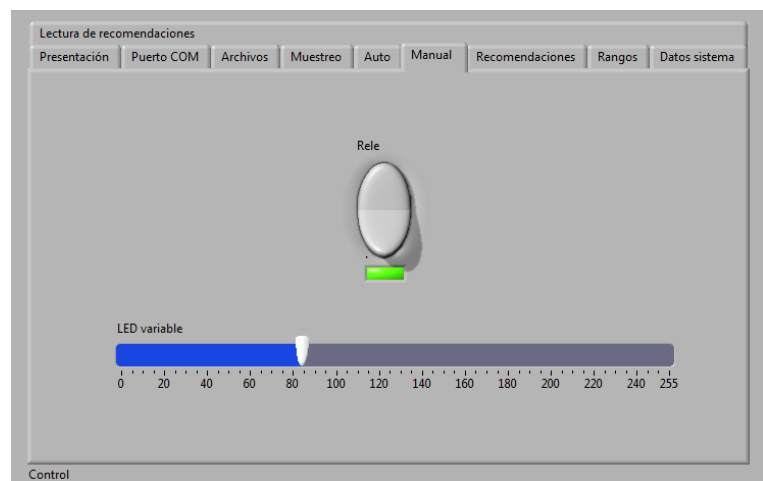


Ilustración 17. Pestaña Manual

#### 4.2.8 RECOMENDACIONES

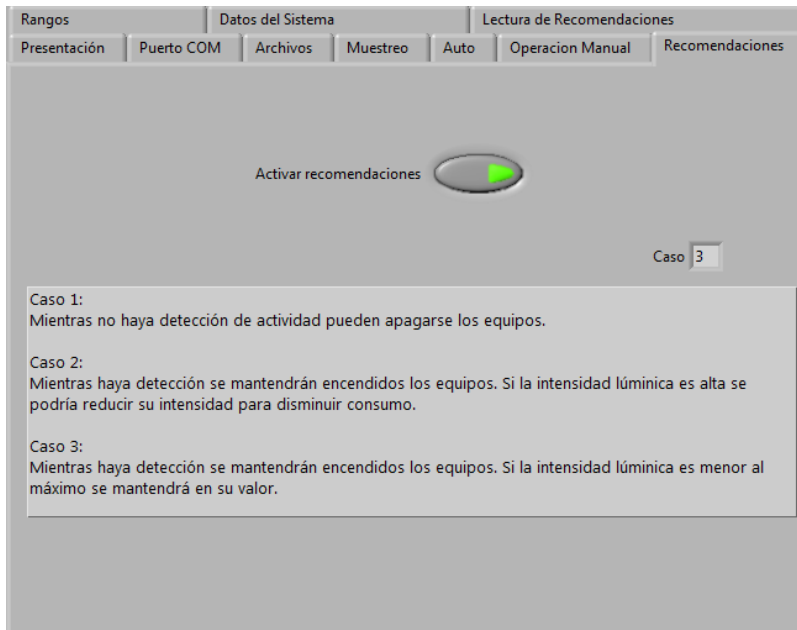


Ilustración 18. Pestaña de Recomendaciones

En esta pestaña la posibilidad de activar las recomendaciones generadas por el sistema, que evalúa los datos recibidos para ofrecer una recomendación en tiempo real, ello se llama “Caso n” que el usuario puede atender.

#### 4.2.9 RANGOS

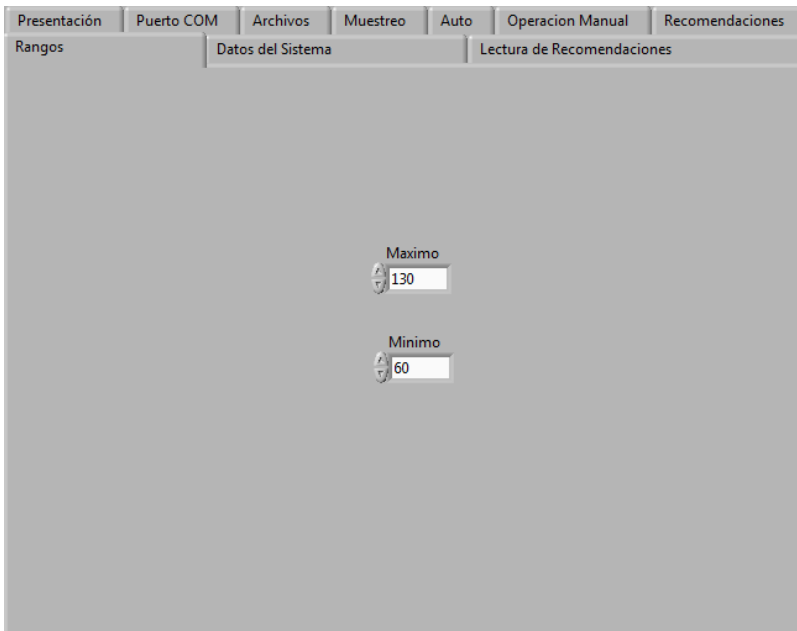


Ilustración 19. Rangos para ajuste de umbrales (Unidades en Lúmenes).

Valores proporcionados por el usuario para ajustar los umbrales y acciones que se van a ejecutar al generarse las recomendaciones.

#### 4.2.10 DATOS DEL SISTEMA

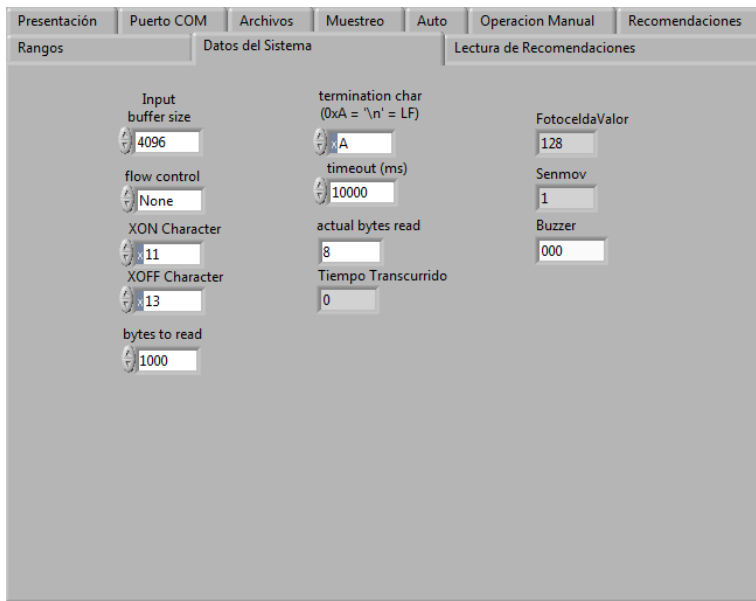


Ilustración 20. Pestaña Datos del Sistema

Valores de las variables usadas, con propósito de informar al administrador del sistema las entradas de los valores y escritura de las mismas.

#### 4.2.11 LECTURA DE LAS RECOMENDACIONES

Lectura de la base de datos para observar las recomendaciones realizadas a los diferentes casos (casa, oficina y consultorio), en ellas se puede apreciar el día de la semana junto con la hora y el caso evaluado en ese instante. Estas son de propósito informativo para que el usuario tenga conocimiento de las recomendaciones ofrecidas para ese día y a la hora que se muestra.

Las anteriores imágenes indican el control y la interfaz dispuesta al usuario, a continuación se presenta la programación gráfica destacando los módulos más importantes.

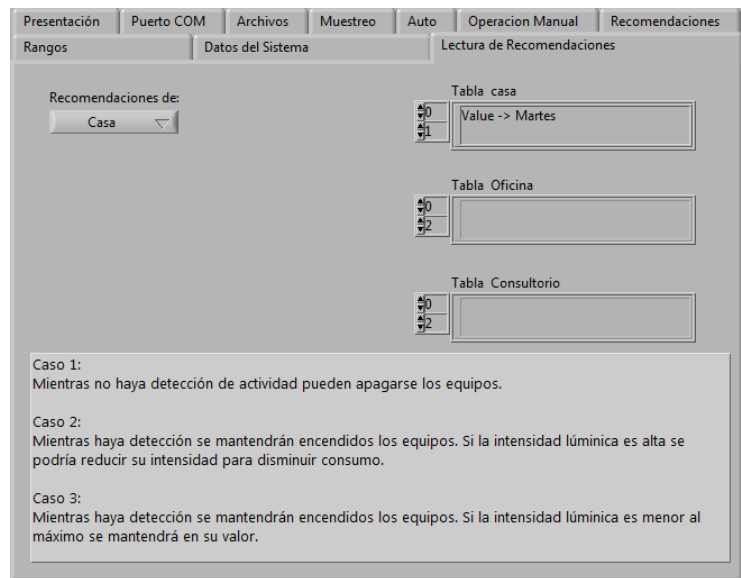


Ilustración 21. Lectura de Recomendaciones

#### 4.2.12 CONTROLADOR VISA

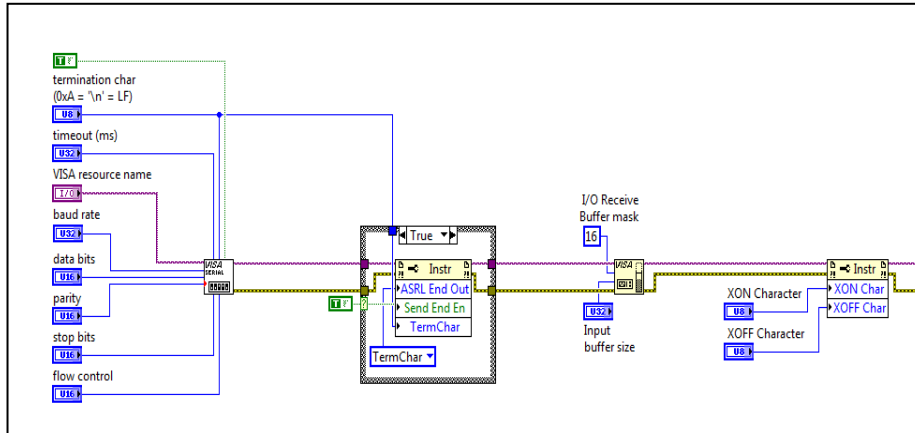


Ilustración 22. Bloque Controlador Visa

Bloque dedicado a la configuración de los parámetros para iniciar una comunicación serial con el puerto COM identificado.

#### 4.2.13 ESCRITURA SERIAL

La escritura serial se realiza a través de este bloque donde se envía una trama codificada que corresponde a los valores ajustados al Led de intensidad variable, al Relévo y a la alarma.

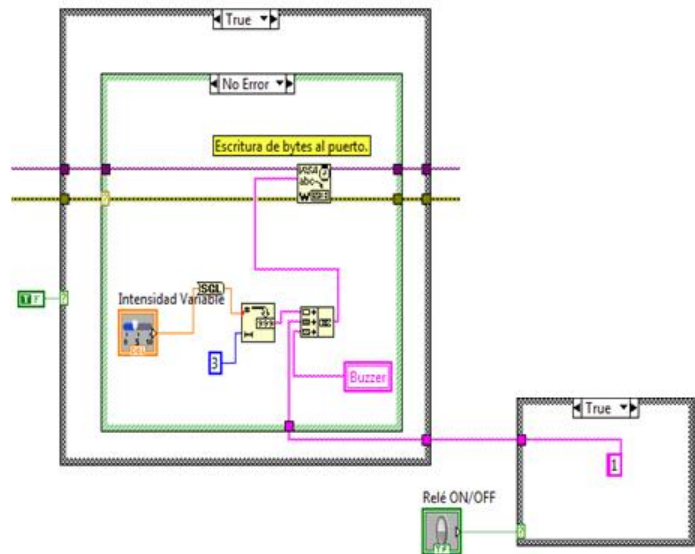


Ilustración 23. Bloque de Escritura Serial

#### 4.2.14 LECTURA SERIAL

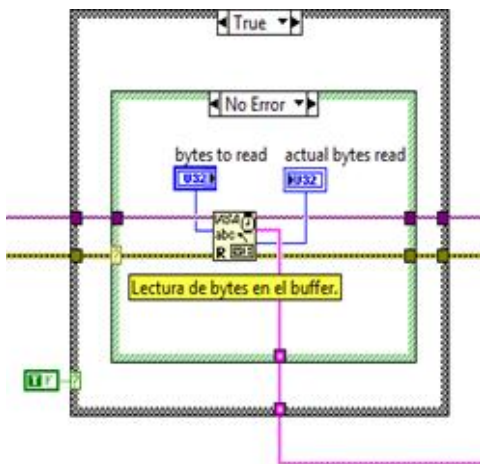


Ilustración 24. Bloque de lectura Serial

Este bloque está dedicado a escuchar lo que se recibe por el puerto serie, para este proyecto se recibe el estado de los sensores del sistema, tal es el caso de la fotocelda como el del sensor de movimiento.



#### 4.2.15 ESCRITURA EN UNA BASE DE DATOS Y EN ARCHIVO .CSV

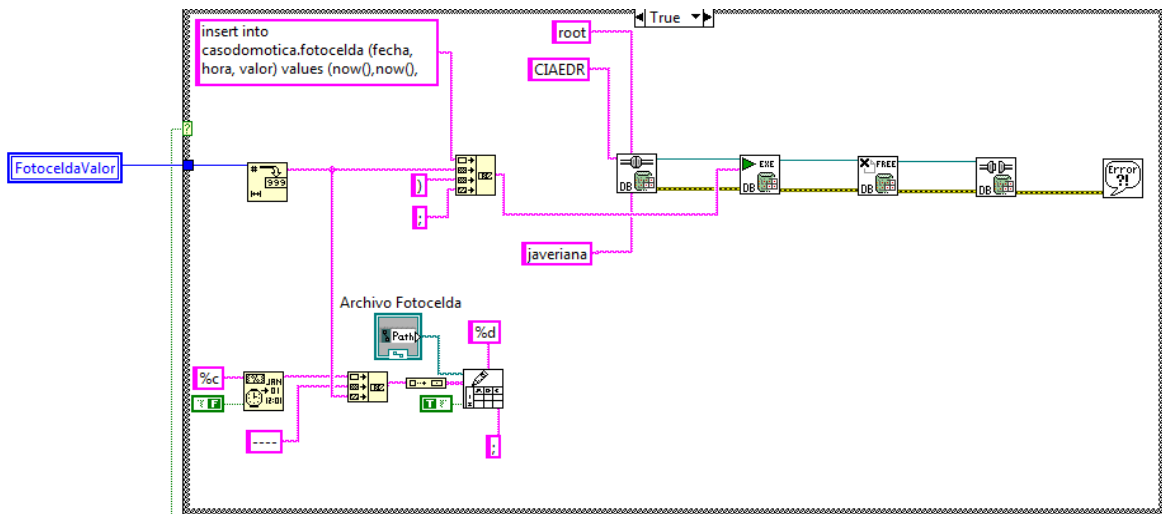


Ilustración 25. Bloque de escritura en una base de datos

En este bloque se realiza la configuración de los módulos para guardar los datos adquiridos en una tabla de una base de datos, de la misma manera para tener un respaldo de la información almacenando los datos en un archivo separado por comas .csv.

El formato de la tabla de la base de datos MySQL donde se almacena la información del estado de la fotocelda es de la siguiente forma, ver ilustración 26:

	idfotocelda	fecha	hora	valor
▶	601	2011-05-28	21:27:52	634
	600	2011-05-28	21:27:13	634
	599	2011-05-28	21:26:35	634
	598	2011-05-28	21:25:56	634
	597	2011-05-28	21:25:17	634
	596	2011-05-28	21:24:38	634
	595	2011-05-28	21:24:00	634
	594	2011-05-28	21:23:21	634
	593	2011-05-28	21:22:22	634
	592	2011-05-28	21:21:43	634
	591	2011-05-28	21:20:55	634

Ilustración 26. Tabla Fotocelda en la base de datos MySQL

#### 4.2.16 LECTURA DE UNA TABLA DE UNA BASE DE DATOS

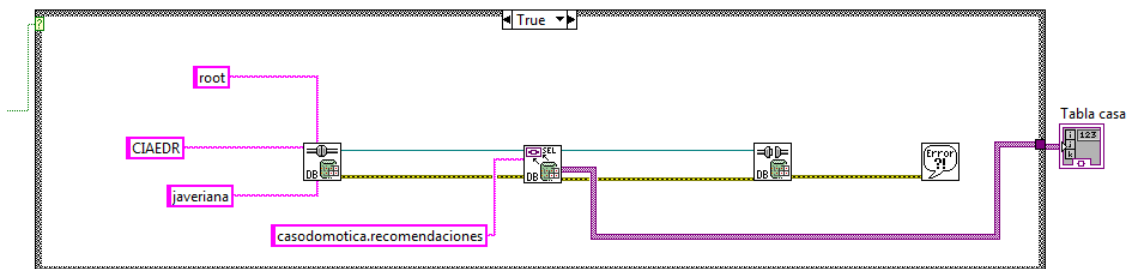
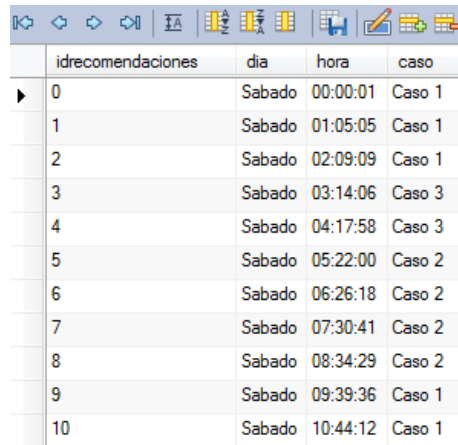


Ilustración 27. Bloque de lectura de una base de datos

Especificación de las conexiones para cargar en una tabla en LabView el contenido de una base de datos.

La apariencia de la tabla de la base de datos encargada de suministrar las recomendaciones del sistema tiene el siguiente formato, ver ilustración 28:



	idrecomendaciones	dia	hora	caso
▶	0	Sabado	00:00:01	Caso 1
	1	Sabado	01:05:05	Caso 1
	2	Sabado	02:09:09	Caso 1
	3	Sabado	03:14:06	Caso 3
	4	Sabado	04:17:58	Caso 3
	5	Sabado	05:22:00	Caso 2
	6	Sabado	06:26:18	Caso 2
	7	Sabado	07:30:41	Caso 2
	8	Sabado	08:34:29	Caso 2
	9	Sabado	09:39:36	Caso 1
	10	Sabado	10:44:12	Caso 1

**Ilustración 28.** Tabla encargada de suministrar recomendaciones.

LabView presenta un gran ventaja al interactuar con la placa Arduino Uno ya que sus bloques orientados a la lectura/escritura del puerto serie son de gran utilidad, consisten en establecer la comunicación a través del puerto COM para luego comenzar a manipular datos.

En los ANEXOS 11 Y 12 se proporcionan los manuales que se consultaron para lograr la lectura y toma de datos de la base de datos a través de Labview.

#### 4.3 RECOMENDACIONES

El propósito de las recomendaciones es formular una posible acción de cambio además de una información al usuario, con la finalidad de fortalecer los recursos del entorno, tal como lo es el ahorro de energía. Diversas pruebas con la fotocelda en el entorno indican un valor máximo y mínimo, estas variables son plenamente controladas por el usuario, en la pestaña Rangos de la intefaz de LabView. A partir de ellos se han formulado unas recomendaciones que corresponden a:

Fotocelda	Valor lógico
fotocelda > máximo	7
mínimo < Fotocelda <= máximo	5
fotocelda <= mínimo	3

**Tabla 2.** Tabla de valor de fotoceldas realizada para considerar casos para recomendaciones.

Sensor de movimiento	Valor lógico
Sensor de movimiento = 1 Activado	1
Sensor de movimiento = 0 Desactivado	0

**Tabla 3.** Tabla de valor de fotoceldas realizada para considerar casos para recomendaciones.

Suma de los valores lógicos	Suma de Valores lógicos	Caso
8	(Fotocelda = 7) + (Sen. Mov = 1) = 8	Caso 1
7	(Fotocelda = 7) + (Sen. Mov = 0) = 7	Caso 2
6	(Fotocelda = 5) + (Sen. Mov = 1) = 6	Caso 1
5	(Fotocelda = 5) + (Sen. Mov = 0) = 5	Caso 3
4	(Fotocelda = 3) + (Sen. Mov = 1) = 4	Caso 1
3	(Fotocelda = 3) + (Sen. Mov = 0) = 3	Caso 3

Tabla 4. Casos generados para recomendaciones.

**Caso 1:**

Mientras no haya detección de actividad pueden apagarse los equipos.

**Caso 2:**

Mientras haya detección se mantendrán encendidos los equipos. Si la intensidad lumínica es alta se podría reducir su intensidad para disminuir consumo.

**Caso 3:**

Mientras haya detección se mantendrán encendidos los equipos. Si la intensidad lumínica es menor al máximo se mantendrá en su valor.

A partir de la evaluación y ejecución de los adecuados casos se puede establecer estrategias de ahorro de energía mediante el análisis de las rutinas y la adecuada atención a las recomendaciones prescritas por el sistema.

4.4 *COMUNICACIÓN INALÁMBRICA XBEE*

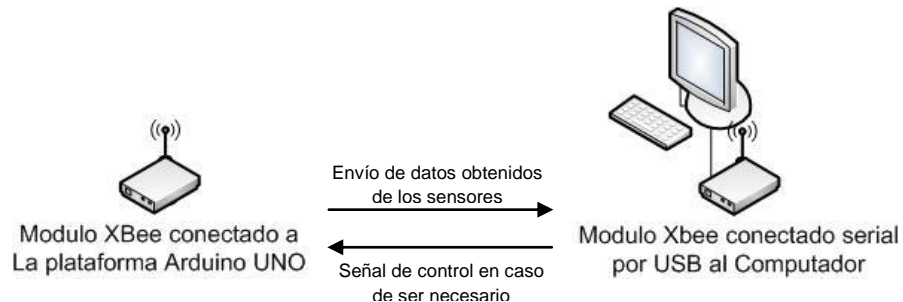
El CIAEDR está conformado por dos módulos XBee, uno conectado directamente al computador y el otro a la plataforma Arduino UNO. El módulo XBee conectado de forma serial a la plataforma toma los datos entregados por los sensores al Arduino UNO y los transmite inalámbricamente al módulo XBee conectado al computador. El enlace entre módulos XBee es punto a punto por lo que únicamente estarán en contacto estos dos nodos. En este tipo de comunicación la relación esclavo/maestro puede cambiar, por lo que en un momento uno de los módulos puede estar actuando como maestro y el otro como esclavo y viceversa.

Los enlaces que interconectan los nodos de una red punto a punto se pueden clasificar en tres tipos según el sentido de la comunicación:

- **Simplex:** El envío de datos solo es enviada en un sólo sentido.
- **Half-dúplex:** El envío de paquetes o información se realiza en ambos sentidos de forma alternativa, es decir, solo uno transmite en un momento dado, una vez este termina de transmitir el otro puede empezar, por lo que ambos no pueden transmitir al mismo tiempo.
- **Full-Dúplex:** El envío de información o paquetes se puede llevar a cabo en ambos sentidos simultáneamente.

El enlace para la comunicación punto a punto del CIAEDR es Half-Dúplex, por lo que en el momento en que se están enviando los datos obtenidos de los sensores, el módulo XBee conectado al computador está en modo recepción. Una vez se finaliza el envío de datos el módulo conectado al computador puede

iniciar un proceso de transmisión quedando de este modo el módulo conectado a la plataforma Arduino UNO en modo de recepción. Con este orden de ideas, el módulo conectado al computador enviará alguna señal de control en caso de ser necesario a partir de los datos que este reciba de los sensores (ver ilustración 8).



**Ilustración 29. Comunicación Punto a Punto, Half-Dúplex**

En los ANEXOS 8 y 9 se brinda información para la configuración de los módulos XBee Serie 2, con base en lo que se encuentra aquí y en ambos anexos se configuró de forma correcta los módulos.

## 5. ANÁLISIS DE RESULTADOS

Durante el desarrollo del proyecto se logró identificar que en la transmisión y recepción por puerto serie, se encuentran valores que dañan la trama de datos, siendo necesario la utilización de códigos por programación para salvaguardar el paquete.

La implementación de una base de datos para el almacenamiento de la información de acuerdo a unos parámetros prescritos, ayuda a establecer relaciones de asociación que generan un análisis más claro.

Incorporar un sistema de análisis de rutinas es vital para establecer las tendencias de comportamiento de los estados de los periféricos dentro de un entorno establecido.

Por otro lado, tras poner a prueba el CIAEDR en diferentes recintos y variar sus umbrales, se tomaron los datos almacenados en las bases de datos generadas en cada caso con el fin de analizar los datos. Estos datos obtenidos por los sensores se analizaron de la siguiente manera.

Se contó el número total de falsos rechazos (FRR) y falsas aceptaciones (FAR) en detección teniendo en cuenta el umbral en que estos se generaron. Una vez se obtienen dichos datos se calcula el FAR y FRR para cada umbral del sensor de movimiento esto se hace mediante la siguiente ecuación:

$$FAR = \frac{\# \text{ total de posibles falsas aceptaciones}}{\text{número total de muestras}}$$

$$FRR = \frac{\# \text{ total de posibles falsos rechazos}}{\text{número total de muestras}}$$

Con los datos obtenidos tras calcular el FAR y FRR se procede a graficar cada uno de ellos con el fin de obtener el EER (Equal Error Rate).

A continuación se muestran los puntos donde se presentaron para cada caso el EER (punto donde se presenta el mismo número de falso rechazo como falsa aceptación):

**Caso Domótica:**

	Umbral	FAR/FRR
<b>EER</b>	0,5242	0,1446

**Caso Consultorio:**

	Umbral	FAR/FRR
<b>EER</b>	0,1947	0,202

**Caso Oficina:**

	Umbral	FAR/FRR
<b>EER</b>	0,3396	0,1462

Luego de tener estos datos, se procedió a realizar funciones de costo.

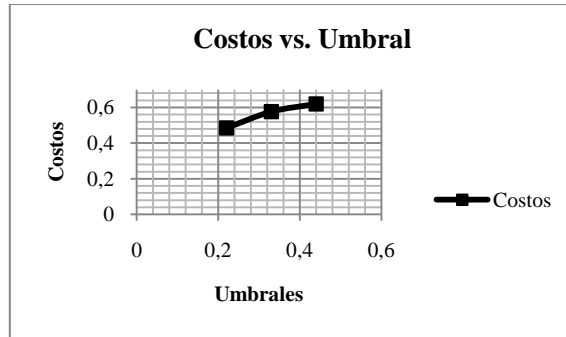
Esto se realizó del siguiente modo, se dio un costo a cada evento que se podía dar en el proceso de detección. Estos eventos son VAR (verdadera aceptación) = \$ 0,2, VRR (verdadero rechazo) = \$0,2, FRR (falso rechazo) = \$0,9 y FAR (falsa aceptación) = \$ 1.

De este modo con las probabilidades calculadas se procedió a calcular el costo total de cada umbral. Este costo total se halló de la siguiente manera:

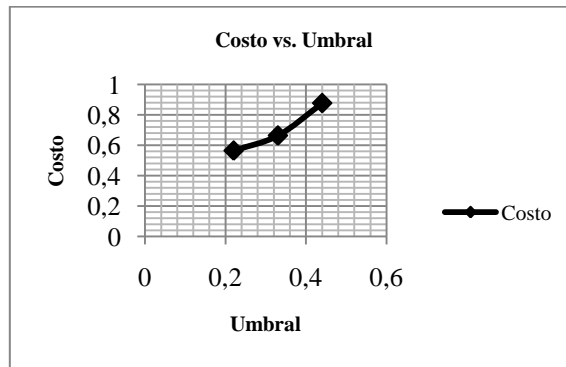
$$C_T = \sum_{i,j} p_{ij} * c_{ij}$$

A continuación se muestran las gráficas obtenidas para cada caso:

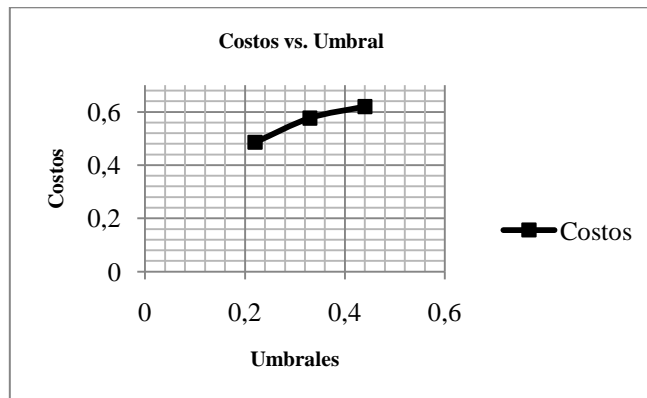
**Caso Domótica:**



**Caso Consultorio:**



**Caso Oficina:**

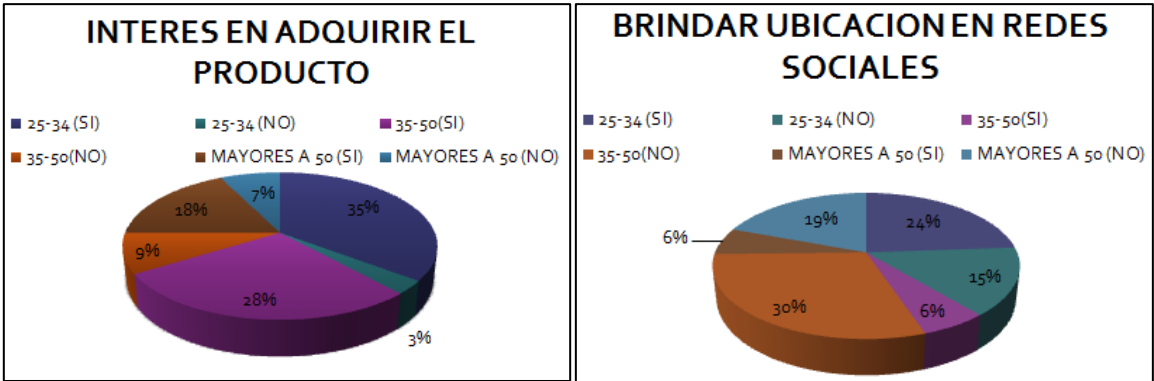


Lo que se analiza a partir de los datos obtenidos por costos es que el menor costo se da cuando el umbral del circuito está en 0,22 es decir 0,4 V en el pin 3 del sensor de detección, esto se concluye para todos los casos. Se observa del análisis realizado que a menor voltaje en el pin de detección del ePir (pin 3) menor falso rechazo y mayor falsas aceptaciones, por lo que se decide trabajar el umbral para detección del sensor de movimiento en voltaje de 0,4 ya que se prefiere tener mayor falsas aceptaciones para no correr el riesgo de no detectar y apagar un equipo cuando verdaderamente se encuentra alguien. De igual forma, al manejar el circuito a este umbral se vuelve más sensible el circuito a detección de movimiento.

En los ANEXOS 14, 15, 16 y 17 se encuentran con mayor detalle el desarrollo que se realizó para el análisis de FAR, FRR y costos.

**6. CONCLUSIONES**

- A partir del monitoreo y análisis de información obtenidos para cada uno de los casos que se pusieron bajo prueba para el desarrollo de este proyecto, se lograron establecer estrategias para un ahorro en el consumo de energía mediante el análisis de las rutinas y sus recomendaciones.
- El desarrollo de sistemas de adquisición de datos requiere del trabajo de las diversas tecnologías informáticas para el adecuado análisis, con el fin de garantizar el manejo adecuado de datos y la obtención de resultados satisfactorios. Es decir, garantizar con el uso de las tecnologías implementadas que las recomendaciones y control desarrollado sean los pertinentes para cada caso.
- El análisis y obtención de la recomendación adecuada requiere un proceso de detección de rutinas que hagan del sistema un proceso inteligente para formular la acción más oportuna e ideal para el ahorro de energía.
- Tras presentarle el proyecto a una pequeña población (población de estrato 4,5 y 6) y mostrar los servicios que se podrían brindar al tener un proyecto de estos se obtuvo información sobre el interés en este proyecto. A partir de los resultados se observa que varias personas adquirirán un dispositivo de estos en caso de estar a la venta, por otro lado, se puede observar que la mayoría de las personas no se encontrarían interesados en comunicar su ubicación en redes sociales.



## 7. BIBLIOGRAFÍA Y FUENTES DE INFORMACIÓN

1. *A Generic Formal Plan Recognition Theory*. **Camilleri, Guy**.
2. *Activity Analysis, Summarization, and Visualization for Indoor Human Activity Monitoring*. **Zhongna Zhou, Xi Chen, Yu-Chia Chung, Zhihai He, Senior Member, IEEE, Tony X. Han, and James M. Keller, Fellow, IEEE**.
3. *A Pattern Mining Approach to Sensor-based Human Activity Recognition*. **Tao GU, Member, IEEE, Liang Wang, Student Member, IEEE, Zhanqing Wu, Xianping Tao, Memeber, IEEE, and Jian Lu**.
4. *On-Body Recognition in a Dynamic Sensor Network*. **C. Lombriser, N.B. Bharatula, D. Roggen, and G. Tröster**.
5. *Activity Recognition in the Home Using Simple and Ubiquitous Sensors*. **E.M. Tapia, S.S. Intille, and K. Larson**.
6. *VISIR, A Simulation Software for Domotics Installations to Improve Laboratory Training*. **Victor M. Gonzalez, Felipe Mateos, Antonio M. López, José M. Enguita, Marta Garcia and Rosana Olaiz**.
7. [www.capta.org.es/wp-content/uploads/2010/12/seminario\\_arduino](http://www.capta.org.es/wp-content/uploads/2010/12/seminario_arduino). [En línea]
8. *MySQL 5.0 Reference Manual*. s.l. : Copyright © 1997, 2011, Oracle and/or its affiliates. All rights reserved., 1997, 2011.
9. *General Discussion on Energy Saving*. **Wang, D.H., Cheng, K.W.E., Dept. of EE, Hong Kong Polytech. University, China**.
10. *Adaptive Energy Saving Monitoring Technology for Air Condition in Telecommunication Equipment Room*. **Tao, D., Wei, P., Guoping, L., Jiang, H., Xiyu, L.** 2007.