

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE
TELECOMUNICACIÓN
UNIVERSIDAD POLITÉCNICA DE CARTAGENA



Proyecto Fin de Carrera

**Desarrollo de un sistema telemático para la gestión de la energía en
industrias agroalimentarias basado en PAC**



AUTOR: Alejandro González Moral

DIRECTORES: M^a Victoria Bueno Delgado, José Miguel Molina Martínez

CODIRECTOR: Francisco Javier Cánovas Rodríguez

Septiembre / 2013



Autor	Alejandro González Moral
E-mail del Autor	alejandrogonzalezmoral@gmail.com
Director(es)	M ^a Victoria Bueno Delgado, José Miguel Molina Martínez
E-mail del Director	Mvictoria.bueno@upct.es
Codirector(es)	Francisco Javier Cánovas Rodríguez
Título del PFC	Desarrollo de un sistema telemático para la gestión de la energía en industrias agroalimentarias basado en PAC
Descriptor(es)	
Resumen	
<p>En la Industria Agroalimentaria existe una necesidad de obtener información en tiempo real sobre lo que ocurre a su alrededor, como de los consumos de agua o de electricidad, por ejemplo, con el fin de tener una capacidad de gestión más ágil. Para esto, es indispensable el uso de las TICs.</p> <p>Tradicionalmente la recogida de esa información se ha realizado por personas, a través de registros y se ha constituido en archivos. Posteriormente se han informatizado dichos archivos posibilitando el almacenado y análisis de información más rápido. Incluso los equipos actuales almacenan dicha información para su posterior volcado en ordenadores. Pero esa información es imprescindible tenerla disponible en tiempo real si se quiere ser competitivo.</p> <p>El objetivo de este proyecto es realizar la comunicación entre los distintos equipos en funcionamiento de una empresa agroalimentaria y su posterior monitorización para la obtención de datos de interés que permitan gestionar y mejorar la eficiencia energética de la empresa. Para llevar a cabo este trabajo es necesario utilizar hardware y software adecuado: el entorno de programación gráfica LabView, analizadores de redes Siemens, tarjetas de comunicación, sensores y sistemas de comunicación, etc.</p>	
Titulación	Ingeniería Técnica de Telecomunicaciones, especialidad Telemática
Intensificación	
Departamento	Tecnologías de la Información y las Comunicaciones
Fecha de Presentación	Septiembre 2013

ÍNDICE

Capítulo 1	11
1.1 Antecedentes	11
1.2 Motivación y objetivos.....	12
1.3 Estructura del contenido	14
Capítulo 2.....	15
2.1 Introducción.....	15
2.2 Normativa.....	15
2.2.1 ISO 50001.....	15
2.2.2 UNE-EN 12464-1.....	16
2.3 Cálculos asociados	18
2.3.1 Conceptos previos.....	18
2.3.2 VEEI	18
2.3.3 Flujo luminoso radiante.....	18
2.4 Requisitos de iluminación.....	20
2.5 Ahorro energético.....	23
2.6 Impacto medioambiental.....	23
Capítulo 3	25
3.1 Introducción.....	25
3.2 Funcionamiento de Modbus	25
3.2.1 Medio físico.....	26
3.2.2 Acceso al medio.....	26
3.3 Variantes de Modbus	26
3.3.1 Modbus sobre bus serie.....	26
3.3.2 Modbus sobre Ethernet	27
3.4 Descripción del protocolo.....	28
3.4.1 Modbus RTU	28
3.4.2 Modbus ASCII.....	30
3.5 Formato de los datos.....	31
3.5.1 Modbus RTU	31
3.5.2 Modbus ASCII.....	31
3.5.3 Registros y funciones Modbus	32
3.6 Requisitos para la comunicación	33
3.7 Ventajas y desventajas de Modbus.....	34

Capítulo 4.....	35
4.1 Introducción.....	35
4.2 SCADA	36
4.2.1 Elementos comunes.....	36
4.2.2 Concepto y aplicación.....	37
4.3 Escenario de trabajo.....	38
4.3.1 Adquisición de los datos	38
4.3.2 Gestión de datos.....	39
4.3.3 Representación de los datos	40
4.4 Diseño del software.....	41
4.4.1 Diseño del proceso de adquisición.....	41
4.4.1.1 CompactRIO.....	42
4.4.1.2 PC antiguo	44
4.4.1.3 Mini-PC.....	44
4.4.2 Diseño del proceso de gestión	45
4.4.2.1 Configuración.....	45
4.4.2.2 Datos globales medidos.....	45
4.4.2.3 Datos de Iluminación.....	45
4.4.3 Diseño del proceso de representación	46
4.5 Conexión entre equipos.....	47
4.5.1 Analizadores de redes	47
4.5.2 Ordenador central	48
4.5.3 Equipo servidor.....	48
Capítulo 5.....	49
5.1 Introducción.....	49
5.2 Interfaz principal.....	49
5.3 Configuración.....	50
5.3.1 Dirección de esclavo Modbus	51
5.3.2 Parámetros de la comunicación	51
5.3.3 Selección de analizadores de redes	52
5.3.4 Protección con contraseña.....	52
5.4 Cuadro General.....	53
5.4.1 Resumen	54
5.4.2 Potencia.....	54
5.4.3 Tensión e Intensidad de corriente	55
5.4.4 Armónicos	57
5.4.5 Otros datos.....	57

5.4.5 Botones Borrar Máximos y Volver.....	58
5.5 Iluminación.....	58
5.5.1 Resumen	59
5.5.2 Tensiones.....	60
5.5.3 Intensidades.....	61
5.5.4 Potencias.....	62
5.5.5 Otros datos.....	63
5.5.6 Introducción de datos.....	64
5.5.6.1 Iluminación interior.....	64
5.5.6.2 Datos introducidos.....	65
5.6 Procesos transparentes.....	67
5.6.1 Lectura y propagación de datos	67
5.6.2 Respaldo en servidor.....	68
Capítulo 6.....	69
6.1 Análisis	69
6.1.1 Funcionamiento de la red Modbus.....	69
6.1.2 Módulo de Cuadro General.....	69
6.1.3 Módulo de Iluminación.....	70
6.2 Conclusiones	71
6.3 Líneas futuras de desarrollo	71
Apéndice 1	75
1.1 Introducción.....	75
1.2 Instrumentos Virtuales (VIs)	75
1.3 Panel Frontal y Diagrama de Bloques	76
1.4 Operaciones y estructuras	78
1.5 Ficheros ejecutables.....	85
Apéndice 2	87
2.1 Introducción.....	87
2.2 Descripción de códigos de función.....	87
2.2.1 Lectura de registros discretos (Código 1, 0x01)	87
2.2.2 Lectura de entradas discretas (Código 2, 0x02)	88
2.2.3 Lectura de registros de almacenamiento (Código 3, 0x03).....	89
2.2.4 Lectura de registros de entrada (Código 4, 0x04).....	89
2.2.5 Escritura de un único registro discreto (Código 5, 0x05)	90
2.2.6 Escritura de un único registro de almacenamiento (Código 6, 0x06)	90
2.2.7 Lectura del estado de las excepciones (Código 7, 0x07)	91
2.2.8 Escritura de n registros discretos (Código 15, 0x0F).....	91

2.2.9 Escritura de n registros de almacenamiento (Código 16, 0x10).....	92
2.3 Códigos de excepción	93
REFERENCIAS.....	95

ÍNDICE DE FIGURAS

Figura 1. 1: Esquema de un sistema SCADA	12
Figura 1. 2: Sistema de Gestión Energética	13
Figura 2. 1: Tabla de flujo luminoso en función de la tensión aplicada.....	19
Figura 2. 2: Tabla de constantes de la curva unificada de flujo radiante.....	19
Figura 2. 3: Tabla de requisitos de iluminación según UNE-EN 12464-1.	20
Figura 2. 4: Tabla de requisitos de iluminación según UNE-EN 12464-1 (cont.).....	21
Figura 2. 5: Límite de VEEI en las zonas de iluminación interior.	22
Figura 3. 1: Comunicación Modbus entre maestro y esclavos en un bus	25
Figura 3. 2: Esquema típico de comunicación Modbus en un bus serie	27
Figura 3. 3: Comunicación Modbus sobre Ethernet. Esquema lógico y físico	28
Figura 3. 4: Composición de mensaje Modbus con codificación RTU.....	28
Figura 3. 5: Cálculo del CRC en Modbus RTU	29
Figura 3. 6: Composición de un mensaje Modbus con codificación ASCII.....	30
Figura 3. 7: Composición de un baudio de información en Modbus RTU	31
Figura 3. 8: Composición de un baudio de información en Modbus ASCII.....	32
Figura 3. 9: Direcciones de registros	32
Figura 3. 10: Funciones Modbus y su código de operación	33
Figura 4. 1: Esquema de modelo básico de comunicaciones.....	35
Figura 4. 2: Ejemplo SCADA.	36
Figura 4. 3: Esquema de topología SCADA.....	37
Figura 4. 4: Ejemplo de sinóptico SCADA.	38
Figura 4. 5: Analizador de redes Sentron PAC3100.....	39
Figura 4. 6: Esquema equipo con servidor de respaldo.....	40
Figura 4. 7: Entorno de programación LabVIEW.	41
Figura 4. 8: Arriba: La programación en código gráfico de la aplicación. Izquierda: La interfaz de la aplicación resaltando el valor medido. Derecha: Display del analizador con el valor medido.....	43
Figura 4. 9: Primera aplicación programada en el CompactRIO.	43
Figura 4. 10: Mini-pc. Por comparación con el conector VGA se aprecian las dimensiones del equipo.	44
Figura 4. 11: Introducción de datos del usuario.....	46
Figura 4. 12: Interfaz primitiva.....	46
Figura 4. 13: Panel trasero del PAC4200.....	48
Figura 5. 1: Interfaz principal de la aplicación.	50
Figura 5. 2: Ventana de Configuración.	51
Figura 5. 3: Contraseña de acceso a Configuración.....	53
Figura 5. 4: Cuadro General, pestaña Resumen.....	54
Figura 5. 5: Cuadro General, pestaña Potencia.....	55
Figura 5. 6: Cuadro General, pestaña de Tensión e Intensidad.....	56
Figura 5. 7: Cuadro General, pestaña de Armónicos.....	57
Figura 5. 8: Cuadro General, pestaña de Otros datos.	58
Figura 5. 9: Iluminación, pestaña Resumen.....	59
Figura 5. 10: Iluminación, pestaña Tensiones.....	60
Figura 5. 11: Iluminación, pestaña Intensidades.....	61
Figura 5. 12: Iluminación, pestaña Potencias.....	62

Figura 5. 13: Iluminación, pestaña Otros.	63
Figura 5. 14: Introducción de datos, Iluminación interior.	64
Figura 5. 15: Introducción de datos, Datos introducidos.	66
Figura 5. 16: Código de la lectura de analizadores de redes.	68
Figura 6. 1: El tipo de bombilla es un factor determinante en el aprovechamiento energético.	70
Figura 6. 2: Futuros módulos a implementar.	72
Figura 7. 1: VI de un filtro paso bajo.....	76
Figura 7. 2: Ejemplo de un panel frontal	77
Figura 7. 3: Vista en el diagrama de bloques	78
Figura 7. 4: Operaciones aritméticas	78
Figura 7. 5: Operaciones lógicas	79
Figura 7. 6: Operaciones con cadenas de texto.	80
Figura 7. 7: Operaciones de comparación.....	81
Figura 7. 8: Operaciones con arrays.	82
Figura 7. 9: Operaciones con clústeres.	82
Figura 7. 10: Operaciones con archivos.	83
Figura 7. 11: Operaciones de temporización.	84
Figura 7. 12: Estructuras de ejecución de código.	85
Figura 8. 1: Códigos de excepción más recurrentes de Modbus.....	93

Capítulo 1

Introducción

1.1 Antecedentes

Hoy en día es indiscutible la importancia que tienen las Tecnologías de la Información y las Comunicaciones (TICs), y su uso se extiende desde la educación y el ocio, hasta la industria o la investigación. Uno de los mayores potenciales del uso de las TICs es el poder obtener información en tiempo real de todo lo que acontece a nuestro alrededor. En la Industria Agroalimentaria existe una necesidad de obtener en tiempo real información acerca de diferentes factores, como la temperatura o el consumo eléctrico, por lo que queda patente la necesidad del uso de las TICs en este sector.

Tradicionalmente, en la Industria agroalimentaria, la recogida de la información de interés se ha realizado por personas, a través de registros y se ha constituido en archivos. Posteriormente se han informatizado dichos archivos posibilitando el almacenado y análisis de información más rápido. Incluso los equipos actuales almacenan dicha información para su posterior volcado en ordenadores. Pero esa información es imprescindible tenerla disponible en tiempo real si se quiere ser competitivo.

Actualmente las TICs permiten la comunicación entre un equipo o una serie de equipos, dispositivos y sensores en funcionamiento de una empresa agroalimentaria y el ordenador central de la organización. Incluso si el equipo a seguir no tiene posibilidad de comunicarse directamente con la entidad central se pueden añadir elementos externos (sensores, autómatas,...) que nos aporten la información necesaria y almacenarla temporalmente para un posterior volcado. A este tipo de arquitectura de comunicación se le denomina SCADA (Supervisory Control and Data Acquisition) [SCADA].

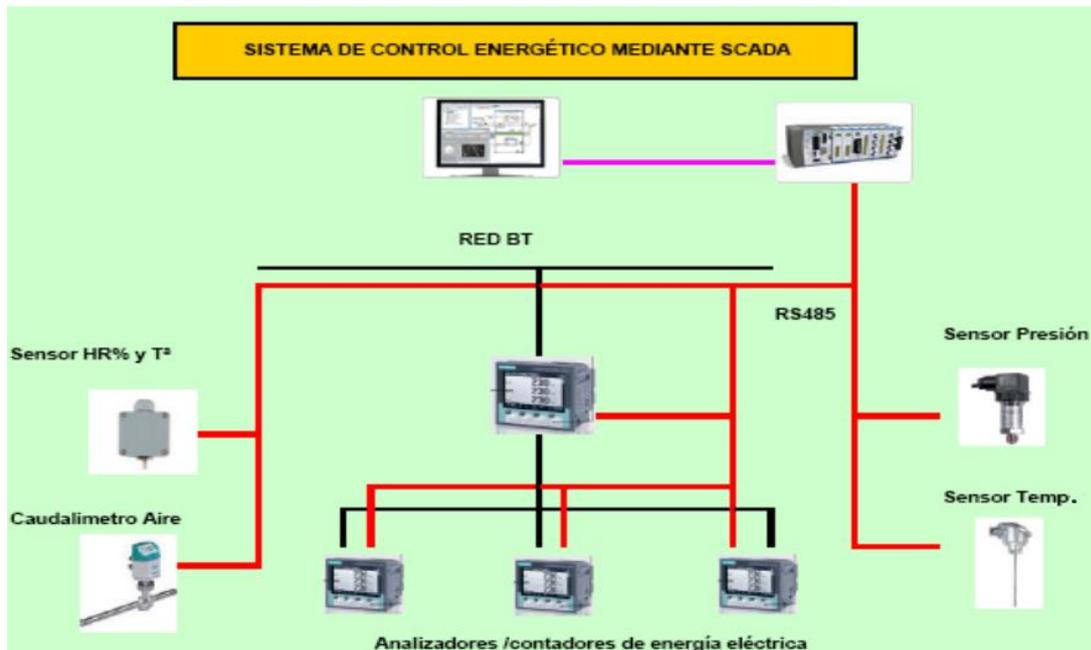


Figura 1. 1: Esquema de un sistema SCADA

1.2 Motivación y objetivos

El acceso remoto a los contenidos que se recogen o generan en otros equipos es la base fundamental de Internet. Y a su vez, éste se ha ido integrando cada vez más en la forma de pensar, trabajar e incluso de relacionarse de la sociedad. Un ejemplo de ello son las redes sociales, como Facebook, que tanto están revolucionando la forma en la que las personas interactúan. Otra de las aplicaciones del acceso remoto a contenidos a través de Internet es la posibilidad de ofrecer sistemas de monitorización que puedan ser accedidos remotamente desde cualquier lugar. De este modo podría conocerse desde un ordenador en el hogar o incluso desde una tablet o smartphone (teléfono inteligente) en todo momento el estado de cualquier sistema, como una caldera en una industria agroalimentaria, por ejemplo.

Por otra parte, aunque manteniéndonos en el ejemplo anterior, en una sociedad en la que prácticamente todo lo que concebamos está informatizado, resulta lógico buscar algún modo de aplicar esta informatización a las tareas de registro de datos, de modo que se pueda automatizar este proceso y tener un conjunto de datos actualizados, o dicho de otro modo, en **tiempo real**. Un ejemplo en el que este registro informatizado y automatizado de datos se podría aplicar con resultados significativos es la gestión del consumo de energía, dando lugar a un sistema de control de la eficiencia energética. Este ejemplo concreto es el que alcanza, finalmente, la motivación de este proyecto: el desarrollo de un sistema de gestión de la energía para la mejora de la eficiencia.

Dicho sistema implicaría una determinada inversión inicial en equipos específicos para el análisis de datos de las instalaciones así como en el software encargado de recoger, interpretar y registrar los citados datos. No obstante, se traduciría en la reducción de costes

energéticos (con el importante ahorro económico que ello supone) así como en la mejora de la sostenibilidad.

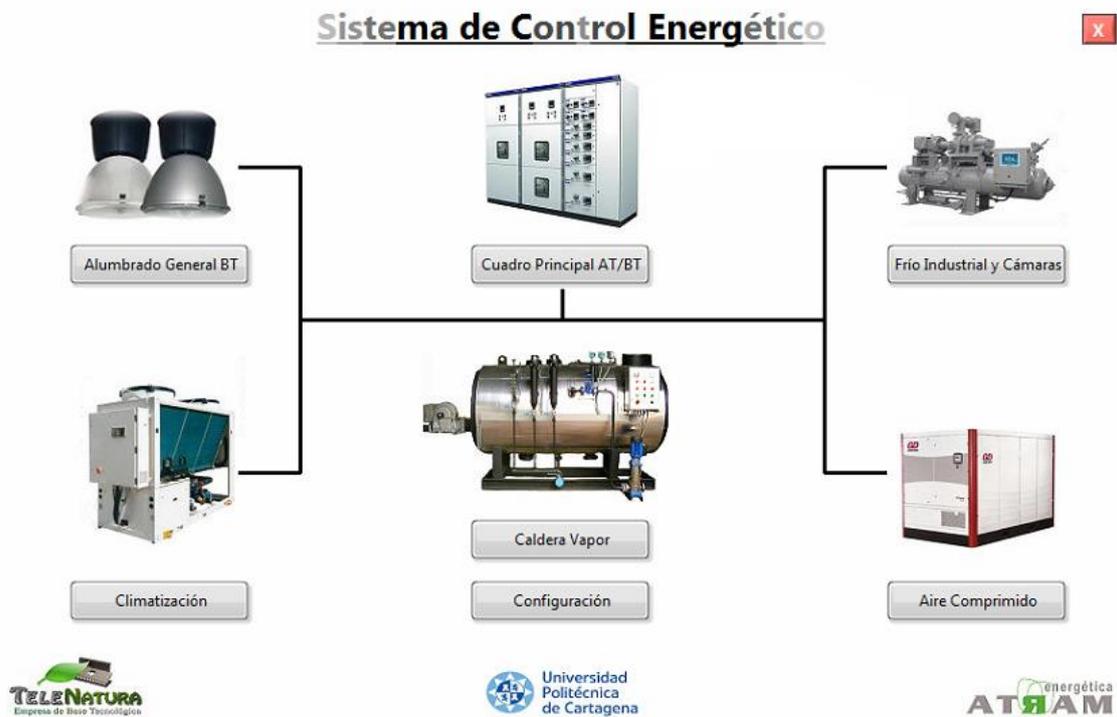


Figura 1. 2: Sistema de Gestión Energética

A grandes rasgos el sistema SCADA global se basaría en la recogida de diversas medidas en los distintos cuadros eléctricos de una industria, así como de otros datos recogidos por sensores de otros valores como temperatura o presión, de modo que se pueda obtener una continua evaluación en tiempo real del consumo energético, la eficiencia y las mejoras posibles, tanto como información instantánea como en forma de informes y de registros históricos replicados en un servidor. No obstante, cabe mencionar que este proyecto está centrado en el módulo de gestión energética en la **iluminación**.

Para que esto sea posible, es muy importante la comunicación tanto entre equipos como hacia el ordenador central y hacia el servidor de respaldo de los datos. Los distintos dispositivos de medida formarán parte de un bus en serie RS-485 operando con protocolo Modbus RTU, bus de esclavos del que el ordenador central será el Maestro. Por otra parte, la réplica de históricos se almacenará por FTP en el servidor remoto. La ejecución del programa se llevará a cabo por el ordenador central, pero no se interactuará con el mismo directamente, sino que el software se publicará como servicio web de forma que pueda ser accedido a través de un navegador de internet desde cualquier ordenador dentro de la misma red Ethernet local.

De modo resumido, podrían considerarse los siguientes objetivos o pasos que se han de realizar para llevar a cabo el proyecto:

- Construcción de una estructura en bus serie RS-485 que interconecte los equipos que se necesitarán en este sistema.
- Implementación de un servidor FTP para poder guardar los archivos históricos de los distintos valores que representen el consumo energético.
- Desarrollo de un software capaz de comunicarse con los equipos analizadores; capaz de analizar y procesar los datos, mostrarlos por pantalla de manera limpia, ordenada y fácil de interpretar, y capaz de guardar estos datos registrados en el servidor.
- Desarrollo de un software que, en base a los datos recogidos, realice los cálculos necesarios para deducir el consumo energético de iluminación y sea capaz de generar informes interpretables en procesadores de texto, así como indicar el índice de eficiencia energética.

1.3 Estructura del contenido

Los objetivos comentados en el punto anterior se desarrollan y explican a lo largo de esta memoria, y vienen estructurados en los siguientes capítulos:

- En el capítulo 2, se realiza una breve introducción sobre la eficiencia energética y las normativas que la rigen, así como los cálculos que serán de interés para este sistema.
- En el capítulo 3, se presenta el protocolo de comunicaciones Modbus, explicando su funcionamiento general, descripción del protocolo y formato de los datos, así como los requisitos para la comunicación Modbus.
- En el capítulo 4, se introduce el sistema de comunicación, explicando su escenario de trabajo y también de las conexiones entre los distintos equipos.
- En el capítulo 5 se presenta la interfaz de usuario del sistema, con sus distintos subprogramas dedicados a configuración, obtención de datos, representación, etc.
- En el capítulo 6 se resumen las conclusiones del proyecto realizado y las líneas futuras que se pretenden seguir.

Capítulo 2

Control de eficiencia energética

2.1 Introducción

Hoy en día es habitual escuchar alusiones a un concepto conocido, aunque quizá poco entendido, la **eficiencia energética**, término muy sonado en tanto que cada vez los recursos son más **limitados** y es primordial tratar de **ahorrar** energía tanto como sea posible. Pero, ¿a qué se le llama eficiencia energética?. La eficiencia energética consiste en una **práctica**, llevada a cabo durante el **consumo** de energía, que tiene como finalidad el intento de la **disminución** del uso de energía. Tradicionalmente se ha buscado abastecerse de energía o utilizar ésta misma de manera que no resulte escasa. Hoy en día, sin embargo, la necesidad es la de utilizar la energía **necesaria** para funcionar, pero buscando que ésta sea la **mínima** posible, de manera que no exista un desperdicio de la misma.

Los individuos y las organizaciones que son consumidores directos de la energía pueden desear el ahorro de esta a fin de **reducir costes** energéticos y promover una mayor **sostenibilidad** económica, política y ambiental. Por su parte, los usuarios industriales y comerciales pueden desear aumentar su **eficacia**, maximizando de esta manera **beneficios**.

Entre las preocupaciones actuales que caracterizan a la eficiencia energética está el **ahorro** de energía y **reducir** el efecto ambiental de la generación de energía eléctrica.

Un ejemplo actual de cuan implicados se encuentran los gobiernos con la eficiencia energética es la reciente obligatoriedad en España de que los inmuebles (viviendas y locales comerciales) a vender o alquilar cuenten con su certificado de eficiencia energética.

2.2 Normativa

El concepto de eficiencia energética no es una simple preocupación de un pequeño conjunto de individuos. De hecho, es un asunto que concierne a **todos**, motivo por el que las autoridades de los distintos países han tomado **medidas** con el objetivo de promover el aprovechamiento eficiente de la energía. Gracias a esto, hoy día existen tanto a nivel nacional como internacional, una serie de **normativas** que regulan determinados aspectos con el fin de **garantizar** la eficiencia energética. En este epígrafe se explica las normativas por las que se rige el Sistema de Gestión Energética que se pretende desarrollar, lo que implica **eficiencia energética** y, a su vez, **calidad** de la iluminación.

2.2.1 ISO 50001

La norma de **calidad** ISO 50001 [ISO] de Sistemas de Gestión Energética, certifica la existencia de un sistema **optimizado** para el uso **correcto** de la energía en cualquier **organización**, sea cual sea su naturaleza o tamaño, su actividad o su dedicación. Esta norma

sustituye a la anterior norma UNE-ISO 16001 [UNE-ISO 16001], recogiendo algunos cambios sustanciales respecto a la misma, como la introducción del concepto de **revisión energética** y cálculo de la línea base. Este giro hace a la norma ISO 50001 más técnica en comparación a su antecesora, acercándola al concepto de **auditoría** energética, equiparable con el de revisión energética. Además, introduce algunas especificaciones referidas a requisitos en la compra de energía.

La finalidad última de la norma ISO 50001 es la facilitación a las organizaciones, independientemente de su sector de actividad o su tamaño, de una **herramienta** que permita la reducción de los **consumos** de energía, así como los **costos** financieros asociados y consecuentemente las **emisiones** de gases de efecto invernadero. Basada en el principio “**medir para identificar, e identificar para mejorar**”, la implantación de un Sistema de Gestión Energética de acuerdo a la norma ISO 50001 permite a las administraciones públicas y empresas **ahorrar** energía, haciendo que cualquier inversión en esta línea tenga un **retorno** económico inmediato, al contrario con lo que sucede con los sistemas de gestión medio ambiental.

Son varios los beneficios derivados de la implantación de la norma ISO 50001. La naturaleza de los mismos son claros desde un punto de vista medio ambiental, económico y operativo y de imagen. Algunos de ellos son:

- **Ahorro** de energía en el corto, medio y largo plazo
- Toma de **conciencia** y **control** de la cantidad de energía consumida en cada proceso.
- Toma de conciencia de las **medidas** de ahorro energético para los procesos **consumidores** de energía en la organización.
- **Reconocimiento** e **imagen** de cara al exterior (clientes, proveedores, accionistas, opinión pública) de su compromiso con un consumo energético **sostenible**.

Así pues, se puede entender la naturaleza de la motivación de este PFC, que implica el desarrollo e implementación de un Sistema de Gestión Energética con el que perseguir los objetivos que certifiquen el cumplimiento de esta norma.

2.2.2 UNE-EN 12464-1

Si bien la norma anterior supone la puesta en práctica de medidas para mejorar la eficiencia energética, en un proyecto como el que se quiere desarrollar, cuyo epicentro es la iluminación, conviene, a la par que conseguir el ahorro de energía, que por otra parte esto se haga cumpliendo unas características de calidad.

La norma UNE-EN 12464-1 [UNE-EN 12464-1] especifica los **requisitos** mínimos de iluminación para espacios de trabajo en **interiores** (excepto instalaciones deportivas). La finalidad es proporcionar un ambiente de iluminación **agradable** en el que las personas puedan trabajar cómodamente. Al mismo tiempo, también pretende satisfacer las necesidades en cuanto a comodidad visual, seguridad y rendimiento.

Los principales factores que afectan a la calidad de la iluminación, recogidos en la norma UNE-EN 12464-1, son los siguientes:

- Reproducción del color
- Temperatura del color
- Niveles de iluminación
- Deslumbramientos

A continuación se adjunta una breve explicación sobre las indicaciones que la norma específica para los anteriores factores, indicaciones que luego se pretende que el software del sistema a desarrollar implementen.

Reproducción del color

El índice de **reproducción** del color Ra indica cuán **naturales** aparecen los objetos bajo la luz, en una escala del 0 al 100. La norma UNE 12464-1 establece unos niveles **mínimos** para prácticamente todos los espacios de trabajo. En los espacios donde se trabaja de forma ininterrumpida, la norma exige un Ra mínimo de 80. Sólo en algunos casos, por ejemplo en la industria metalúrgica pesada, se acepta 60 como mínimo. En consecuencia, algunos tipos de lámpara, conocidos más comúnmente como "estandar", ya no se pueden aplicar debido a sus insuficientes propiedades de reproducción del color.

Temperatura de color

El color de la luz es la **apariencia** de calidez o frialdad de la luz. Se caracteriza por la **temperatura** (T_{CP}) del color (en Kelvin). Para la iluminación general de interiores, la luz se divide en 3 clases de colores: blanco cálido (por debajo de 3.300 K), blanco neutro (de 3.300 a 5.000 K) y blanco frío (por encima de 5000 K). En las salas de atención médica y en las zonas en las que la se realicen trabajos de inspección de color o de precisión, se deberá utilizar una temperatura de color como mínimo de 4.000 K. En las consultas de odontología (igualación del blanco de los dientes) e inspección de color en laboratorios y farmacias, se precisa una temperatura de color de 6.000 K o superior.

Niveles de iluminación

Los **niveles** de iluminación se deben diseñar y medir sobre las zonas de **trabajo** de la sala. La norma establece unos valores **mínimos** para los niveles de iluminación en edificios **profesionales**. En las áreas en las que hay personas presentes, el nivel mínimo es de 200 lux. En las áreas en las que solamente hay personas presentes de forma ocasional, el nivel mínimo es de 20 lux. Estos niveles deben **aumentarse** cuando se trata de trabajos muy **visuales**, o bien la mayor **productividad** tiene mucha importancia, o si la capacidad visual del usuario está por debajo de lo normal, así como cuando las tareas se llevan a cabo durante un tiempo bastante más prolongado de lo habitual. Por el contrario, pueden **reducirse** cuando los detalles de las tareas tienen mucho **contraste** o cuando las tareas se llevan a cabo en muy **poco** tiempo.

Deslumbramientos

El índice de deslumbramiento unificado, también conocido como UGR, es una valoración del **deslumbramiento**, que es la sensación producida por áreas **brillantes** dentro del campo de visión y puede ser experimentado como deslumbramiento **molesto** o perturbador. En lugares de trabajo en **interiores**, el deslumbramiento molesto puede producirse **directamente** a partir de luminarias brillantes o ventanas.

2.3 Cálculos asociados

Para llevar a cabo el proyecto de desarrollo de un Sistema de Gestión Energética, en este caso centrado en Iluminación, de un modo que se ajuste a la normativa, se definen una serie de conceptos y datos que habrá que tener en cuenta.

2.3.1 Conceptos previos

La norma UNE-EN 12464-1 indica la existencia de una serie de factores, que en las siguientes líneas se detallan:

- **Iluminancia mantenida (E_m):** Es el valor por debajo del cual no se permite que caiga la iluminancia media en una superficie determinada.
- **Rendimiento del color (R_a):** Explicada en el epígrafe anterior, determina la calidad con la que se distinguen los colores.
- **Índice de deslumbramiento molesto (UGR):** El deslumbramiento que se produce por las luminarias en una instalación de iluminación interior se debe determinar según el método de tabulación del Índice de Deslumbramiento Unificado de la CIE (UGR, Unified Glare Rating), basado en la fórmula:

$$UGR = 8 \cdot \log_{10} \left(\frac{0,25}{L_b} \sum \frac{L^2 \cdot \omega}{p^2} \right) \text{ donde:}$$

- L_b es la iluminancia de fondo en $\text{cd} \cdot \text{m}^{-2}$, calculada como $E_{\text{ind}} \cdot \pi^{-1}$, en la que E_{ind} es la iluminancia indirecta vertical en el ojo del observador.
- L es la iluminancia de las partes luminosas de cada luminaria en la dirección del ojo del observador en $\text{cd} \cdot \text{m}^{-2}$.
- ω es el ángulo sólido (estereorradianes) de las partes luminosas de cada luminaria en el ojo del observador.
- p es el índice de posición de Guth para cada luminaria individual que se refiere a su desplazamiento de línea de visión.

2.3.2 VEEI

Se define el VEEI como el valor de eficiencia energética de la instalación. Este valor se determinará mediante la siguiente expresión $VEEI = \frac{P \cdot 100}{S \cdot E_m}$ donde:

- P es la potencia de la lámpara más el equipo auxiliar (W).
- S es la superficie iluminada (m^2).
- E_m es la iluminancia media mantenida (lux).

2.3.3 Flujo luminoso radiante

Se define el flujo radiante como la medida de la potencia de una radiación electromagnética (incluyendo los infrarrojos, ultravioletas y la luz visible). Es la energía que transportan las ondas por unidad de tiempo. Esta potencia puede ser la total emitida por una fuente o la que llega a una superficie determinada.

En la tabla que se adjunta a continuación, se puede observar cómo son las curvas de flujo luminoso radiante, en porcentaje, para cada uno de los tipos de lámpara más conocidos, en función de la cantidad de tensión en voltios (V) que alimenta la lámpara.

LÁMPARA	$\phi=f(v)$
1 INCANDESCENTE	$\phi=5,68*v - 1148,9$
2 INCANDESCENTE HALÓGENA	$\phi=2E-42v^{18,651}$
3 FLUORESCENCIA	$\phi= 0,25v + 45$
4 VAPOR DE MERCURIO ALTA PRESIÓN	$\phi=2,668v - 486,960$
5 MEZCLA	$\phi=2,6657v - 486,4$
6 HALOGENUROS	$\phi=2,46x - 441$
7 VAPOR DE SODIO ALTA PRESIÓN	$\phi=3,12v - 585,6$
8 VAPOR DE SODIO BAJA PRESIÓN	$\phi=a*\text{Sin}(b*(v-c))+d$

Figura 2. 1: Tabla de flujo luminoso en función de la tensión aplicada.

De la anterior tabla falta indicar el valor que adquieren las constantes **a**, **b**, **c** y **d**:

- **a = -1,9810505**
- **b = 0,99090702**
- **c = -0,66666937**
- **d = 98,1893343**

Para poder llevar a cabo una correcta medición del flujo luminoso en el posterior software del Sistema de Gestión Energética que se pretende crear, hay que hallar la manera de unificar todas las funciones de las curvas de flujo radiante indicadas. Para ello se ha llegado a la siguiente ecuación:

$$\phi = \alpha \cdot v + \beta \cdot v^{18,651} + \gamma + \delta \cdot [a * \text{Sin}(b * (v - c)) + d]$$

Para cada tipo de iluminación existen unas constantes α , β , γ y δ distintas, dando lugar a la curva correcta para cada uno de los tipos distintos de iluminación:

TIPO DE ILUMINACIÓN	α	β	Γ	δ
1 INCANDESCENTE	5,68	0	-1148,9	0
2 INCANDESCENTE HALÓGENA	0	$2 \cdot 10^{-42}$	0	0
3 FLUORESCENCIA	0,25	0	45	0
4 VAPOR DE MERCURIO ALTA PRESIÓN	2,668	0	-486,96	0
5 MEZCLA	2,6657	0	-486,4	0
6 HALOGENUROS	2,46	0	-441	0
7 VAPOR DE SODIO ALTA PRESIÓN	3,12	0	-585,6	0
8 VAPOR DE SODIO BAJA PRESIÓN	0	0	0	1

Figura 2. 2: Tabla de constantes de la curva unificada de flujo radiante.

2.4 Requisitos de iluminación

La normativa UNE-EN 12464-1 proporciona una serie de indicaciones para que la iluminación en una organización responda a unos criterios determinados de calidad.

El Sistema de Gestión Energética que se pretende desarrollar está ideado para ser implantando, fundamentalmente, en industrias agroalimentarias. Por ello, a continuación se indican en unas tablas los requisitos exigidos para cada una de las zonas de iluminación contempladas dentro del entorno agroalimentario.

Los datos recogidos serán **implementados** en el software del Sistema de Gestión Energética, de modo que se cumplan los **criterios** de calidad especificados por la misma.

1.1 Zonas de tráfico					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
1.1.1	Áreas de circulación y pasillos	100	28	40	1. Iluminancia al nivel del suelo. 2. R_a y UGR similares a áreas adyacentes. 3. 150 lux si hay vehículos en el recorrido. 4. El alumbrado de salidas y entradas debe proporcionar una zona de transición para evitar cambios repentinos de iluminancia entre interior y exterior de día o de noche. 5. Debería tenerse cuidado para evitar el deslumbramiento de conductor y peatones.
1.1.2	Escaleras, escaleras automáticas y cintas transportadoras	150	25	40	
1.1.3	Rampas, tramos de carga	150	25	40	
1.2 Zonas de descanso, sanitarias y de primeros auxilios					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
1.2.1	Cantinas, despensas	200	22	80	
1.2.2	Salas de descanso	100	22	80	
1.2.3	Salas para ejercicio físico	300	22	80	
1.2.4	Vestuarios, salas de lavado, cuartos de baño, servicios	200	25	80	
1.3 Salas de control					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
1.3.1	Salas de material, salas de mecanismos	200	25	60	
1.3.2	Salas de fax, correos, cuadro de contadores	500	19	80	
1.4 Salas de almacenamiento, almacenes fríos					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
1.4.1	Almacén y cuarto de almacén	100	25	60	200 lux si está ocupado en continuo
1.4.2	Áreas de manipulación de paquetes y de expedición	300	25	60	

Figura 2. 3: Tabla de requisitos de iluminación según UNE-EN 12464-1.

1.5 Área de almacenamiento con estanterías					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
1.5.1	Pasillos: sin guarnecer	20		40	Iluminancia al nivel del suelo
1.5.2	Pasillos: guarnecidas	150	22	60	Iluminancia al nivel del suelo
1.5.3	Estaciones de control	150	22	60	
2.1 Agricultura					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
2.1.1	Carga y operaciones con artículos, equipo de manipulación y maquinaria	200	25	80	
2.1.2	Edificios para ganadería	50		40	
2.1.3	Sala de veterinaria, establos para parir	200	25	80	
2.1.4	Preparación de alimentos; vaquería; lavado de utensilios	200	25	80	
2.2 Panaderías					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
2.2.1	Preparación y hornos de cocer	300	22	80	
2.2.2	Acabado, horneado, decorado	500	22	80	
2.7 Productos alimenticios e industria de alimentos de lujo					
Ref	Tipo de interior y actividad	E_m	UGR _L	R_a	Observaciones
2.7.1	Puestos de trabajo y zonas en: <ul style="list-style-type: none"> - Fábricas de cerveza, malta - Para lavado, llenado de barriles, limpieza, tamizado, descascarado - Cocción en fábricas de conservas y chocolates - Puestos de trabajo y zonas en azucareras - Para secar y fermentar el tabaco en rama, cueva de fermentación 	200	25	80	
2.7.2	Clasificación y lavado de productos: molienda, mezclado, envasado	300	25	80	
2.7.3	Puestos de trabajo y zonas críticas en mataderos, carnicerías, molinos de queserías, o zonas de filtrado en refinerías de azúcar	500	25	80	
2.7.4	Corte y clasificación de frutas y vegetales	300	25	80	
2.7.5	Fabricación de alimentos de delicatessen, trabajo en cocinas, fabricación de puros y cigarrillos	500	22	80	
2.7.6	Inspección de vidrios y botellas, control de productos, desbarbadura, clasificación, decoración	500	22	80	
2.7.7	Laboratorios	500	19	80	
2.7.8	Inspección de colores	1000	16	90	$T_{CP} \geq 4000 \text{ K}$

Figura 2. 4: Tabla de requisitos de iluminación según UNE-EN 12464-1 (cont.).

Por otra parte, se establece un valor **máximo** para el VEEI según el tipo de zona de iluminación que se presente, como se puede comprobar en la siguiente tabla. De este modo se distingue entre dos **grupos**, que engloban cada uno una serie de zonas de iluminación. Estos grupos son los siguientes:

- **Zonas de no representación:** espacios donde el criterio de diseño, la imagen o el estado anímico que se quiere **transmitir** al usuario con la iluminación, queda relegado a un **segundo plano** frente a otros criterios como el nivel de iluminación, el confort visual, la seguridad y la eficiencia energética;
- **Zonas de representación:** donde el criterio, la imagen o el estado anímico que se quiere **transmitir** al usuario con la iluminación, son **preponderantes** frente a los criterios de eficiencia energética.

Grupo	Zonas de actividad	VEEI
Zonas de no representación	Administrativo en general	3,5
	Andenes de estaciones de transporte	3,5
	Salas de diagnóstico	3,5
	Pabellones de exposición o ferias	3,5
	Aulas y laboratorios	4
	Habitaciones de hospital	4,5
	Zonas comunes	4,5
	Almacenes, archivos, salas técnicas, cocinas	5
	Aparcamientos	5
	Espacios deportivos	5
	Recintos interiores asimilables a este grupo, no citados	4,5
Zonas de representación	Administrativo en general	6
	Estaciones de transporte	6
	Supermercados, hipermercados y grandes almacenes	6
	Bibliotecas, museos y galerías de arte	6
	Zonas comunes en edificios residenciales	7,5
	Centros comerciales (excluidas tiendas)	8
	Hostelería y restauración	10
	Religioso en general	10
	Salones de actos, auditorios, salas de usos múltiples y convenciones, salas de ocio o espectáculo, salas de reuniones	10
	Tiendas y pequeño comercio	10
	Zonas comunes	10
	Habitaciones de hoteles, hostales, etc.	12
	Recintos interiores asimilables a este grupo, no citados	10

Figura 2. 5: Límite de VEEI en las zonas de iluminación interior.

Todos estos parámetros recogidos en esta sección son los que determinarán que una instalación de iluminación sea **eficiente**.

2.5 Ahorro energético

Conseguir una **mejora** de la eficiencia energética tiene una ventaja principal, que afecta directamente a los consumidores. Se trata del **ahorro** energético.

Cuando se decide llevar a cabo una optimización de este tipo, se persigue como objetivo final **reducir**, tanto como sea posible, el **consumo** de la energía que se lleva a cabo. Esto, en función de la cantidad que se consiga llegar a reducir de consumo energético, junto a otros factores como puede ser el tamaño de la empresa u organización en la que se pretenda llevar el control de eficiencia, puede suponer una gran cantidad de **dinero** ahorrado, lo cual es muy deseable para cualquier organización.

Mediante la adquisición y puesta en marcha de unas fáciles **costumbres** en cuanto a medidas de ahorro, se puede llegar a reducir **parcialmente** la cantidad de energía que se consume, tanto en el hogar como en empresas o industrias. No obstante, estas medidas pueden no ser conocidas, además de no ser suficientes para llevar a cabo una **optimización** importante de la energía.

El desarrollo de un Sistema de Gestión Energética se encarga de **facilitar** ese ahorro económico, gracias a su labor de **recogida** de datos y la **información** que se proporciona al respecto. Mediante el ofrecimiento al usuario de los datos que le ayuden a comprender el mayor o menor consumo de energía que ofrecen distintas variantes de un mismo elemento (como por ejemplo, distintos tipos de lámpara para una instalación de iluminación), éste puede tomar las **decisiones** pertinentes para conseguir un mayor **ahorro** energético y así aumentar **beneficios**, del mismo modo que se trabaja de un modo más respetuoso con el **medio ambiente**.

Bajo este planteamiento, resulta entendible que se pretenda desarrollar un Sistema de Gestión Energética para su implantación, principalmente, en industrias agroalimentarias, ya que el ahorro energético y, por tanto, **económico** que supone, **compensa** el gasto realizado en los **aparatos** de medida y **software** que conforman dicho sistema.

2.6 Impacto medioambiental

Como por todos es sabido, el consumo energético está altamente relacionado con las emisiones de CO₂, y por tanto el ahorro de energía es absolutamente fundamental para la reducción de dichas emisiones. Tradicionalmente se ha tratado de reducir el impacto ambiental de la generación de energía a base del establecimiento de determinadas cuotas de emisión de CO₂ o impuestos sobre las emisiones. De este modo, también se han tratado de frenar fenómenos como es el cambio climático. Sin embargo, estos métodos empleados hasta el momento resultan escasos en la pugna por conseguir su objetivo, lo que sugiere la conveniencia de favorecer el ahorro y eficiencia energética.

Aunque actualmente existen diferentes sistemas de generación de energía limpia, como las placas fotovoltaicas o los parques eólicos, la mayor parte de la energía proviene de las centrales térmicas y nucleares. Ambos tipos de centrales suponen determinados riesgos:

- Las centrales térmicas, por una parte, emplean para la generación de energía combustibles fósiles. Estos, al ser quemados, liberan emisiones de CO₂ a la atmosfera, favoreciendo el efecto invernadero y, con ello, el cambio climático. Además, los combustibles fósiles son una fuente de energía no renovable, con lo que conviene reducir su utilización en pos de frenar el agotamiento de estos.
- Por otra parte, las centrales nucleares suponen un grave riesgo para la población debido a la radiactividad de los residuos o al peligro tanto de fugas como, aunque más improbable, de una explosión nuclear.

Estas razones son solo un pequeño ejemplo de la necesidad del ahorro de la energía, gracias al cual reducir estos riesgos que la generación de energía conlleva por satisfacer una demanda de la misma hasta ahora elevada.

La eficiencia energética es el **camino** ineludible hacia un crecimiento **sostenible**, de modo que tanto el ser **humano** directamente, gracias al ahorro económico, como el **medio ambiente** se vean beneficiados.

Capítulo 3

Comunicación con Modbus®

3.1 Introducción

En la actualidad, los procesos industriales se encuentran informatizados en la medida de lo posible. Esto se debe principalmente a la mejora productiva que esta informatización significa, agilizando la toma de datos y automatizando tareas. En este contexto, coordinar todos los procesos es primordial, las comunicaciones han de realizarse de la manera que más se adecúe al entorno, valorando que se ofrezca una correcta transmisión de la información, de manera fiable y segura, lo más rápida posible y de la manera más económica. Para esta intención encontramos una serie de protocolos de comunicaciones capaces de cumplir este objetivo, no obstante es el protocolo Modbus [Modbus] el que mejor se adapta a estos propósitos. Es por ello que este capítulo está dedicado a su estudio, explicación y comprensión.

La designación Modbus Modicon corresponde a una marca registrada por Gould Inc. Como en tantos otros casos, la designación no corresponde propiamente al estándar de red, incluyendo todos los aspectos desde el nivel físico hasta el de aplicación, sino a un protocolo de nivel de enlace, nivel 2 de la pila de protocolos OSI (*Open Systems Interconnection*) [OSI]. Puede, por tanto, implementarse con diversos tipos de conexión física y cada fabricante suele suministrar un software de aplicación propio, que permite parametrizar sus productos. No obstante, se suele hablar de Modbus como un estándar de bus de campo, cuyas características esenciales son las que se detallan a continuación, en los siguientes epígrafes.

3.2 Funcionamiento de Modbus

El funcionamiento de Modbus sigue una arquitectura maestro-esclavo en forma de bus con múltiples esclavos. De este modo, el maestro tiene acceso a cada uno de los esclavos de manera que puede leer sus registros de lectura, y leer o escribir los registros de entrada.

En la siguiente figura se puede observar una representación de una posible topología de comunicación Modbus, en la que un ordenador ejerce de maestro y varios PLCs ejercerían como esclavos:

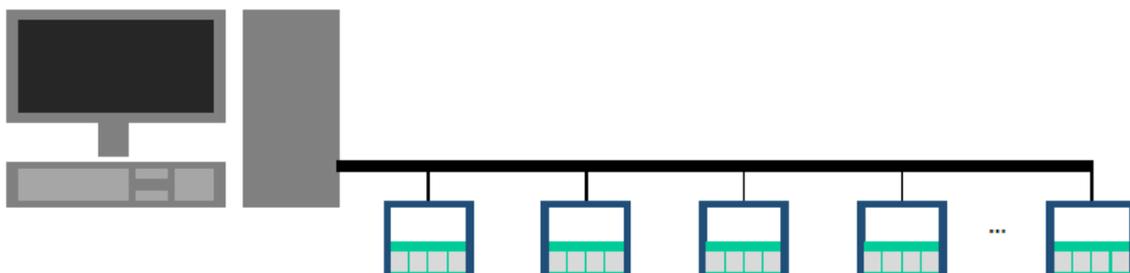


Figura 3. 1: Comunicación Modbus entre maestro y esclavos en un bus

3.2.1 Medio físico

El medio físico de conexión puede ser un bus semidúplex (half duplex) (RS-485 o fibra óptica) o dúplex (full duplex) (RS-422, BC 0-20mA o fibra óptica).

La comunicación es asíncrona y las velocidades de transmisión previstas van desde los 75 baudios a 19.200 baudios. La máxima distancia entre estaciones depende del nivel físico, pudiendo alcanzar hasta 1200 m sin repetidores.

3.2.2 Acceso al medio

Cada uno de los esclavos tiene asignada una dirección de esclavo desde el 1 hasta el 247. Esta dirección será el identificador unívoco que determine con quién establece la comunicación el maestro. Cabe destacar que, aunque puedan direccionarse hasta 247 dispositivos, es necesario contar con repetidores. De otro modo, el número máximo de equipos queda limitado a 32.

Para comunicarse con un esclavo del bus, el maestro debe indicar el número (dirección) de dicho esclavo, además de la función que desea realizar, siendo las más comunes las de lectura o escritura de registros *'holding'*, lectura o escritura de registros booleanos o *'coils'* o lectura de los registros de entrada. En el caso de realizar lecturas, debe indicar la dirección del registro desde la que se quiere leer, así como el número de registros consecutivos que se desea obtener. Por otra parte, en el caso de las escrituras, se indica la dirección del registro a escribir y el valor que se le desea dar.

Al enviar el maestro la trama de petición al bus todos los esclavos la reciben pero solo aquel cuya dirección de esclavo coincide con la de la petición será quien la procese y la ejecute.

La comunicación se realiza normalmente en modo en una sola dirección a la vez (*half duplex*), de modo que o bien el maestro está realizando alguna solicitud a los esclavos, o bien uno de los esclavos se halla respondiendo al maestro. Esto sucede así ya que no tendría sentido en estas circunstancias que los esclavos estuviesen dando una información que no les ha sido solicitada.

3.3 Variantes de Modbus

Modbus es un protocolo estándar para comunicaciones industriales. Sin embargo, su existencia presenta varias versiones, en función del medio sobre el que se transmita, por una parte, y dependiendo de cómo la información a transmitir es codificada, por otro. Para entender el protocolo que se ha utilizado, se procede a la clasificación de variantes.

3.3.1 Modbus sobre bus serie

Sobre este soporte físico, concretamente sobre bus en serie RS-232, RS-422 o RS-485, podemos encontrar los protocolos Modbus RTU y Modbus ASCII.

- Cuando los controladores están configurados para comunicarse mediante **Modbus ASCII**, cada byte de un mensaje es enviado como dos caracteres. Esto presenta la ventaja de mayor robustez frente a errores al permitir intervalos temporales de hasta un segundo entre caracteres sin que se produzca error.

- Por otra parte, si los controladores del bus funcionan mediante **Modbus RTU**, cada byte de un mensaje contiene dos caracteres hexadecimales. La principal ventaja de este modo de funcionamiento es que su mayor densidad de carácter presenta un mayor volumen de información que la variante ASCII para la misma tasa de baudios. Cada mensaje se debe transmitir en una cadena continua.

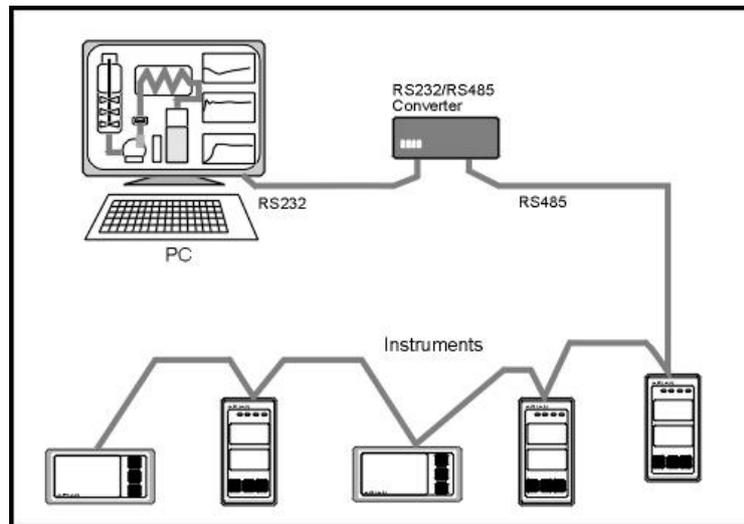


Figura 3. 2: Esquema típico de comunicación Modbus en un bus serie

3.3.2 Modbus sobre Ethernet

Por su parte, una variante de este protocolo se desarrolló para funcionar sobre Ethernet. Es el llamado Modbus TCP.

- El protocolo Modbus TCP se caracteriza por el envío de la información encapsulada en una trama TCP. En esta variante, los distintos esclavos y el maestro se hallan conectados en una misma red Ethernet a través de, al menos, un switch. Los sockets TCP a través de los que la información se transmite, se envían sobre el puerto registrado 502. En esta variante, la información se codifica del mismo modo que en Modbus RTU, con la adición de unas determinadas cabeceras.

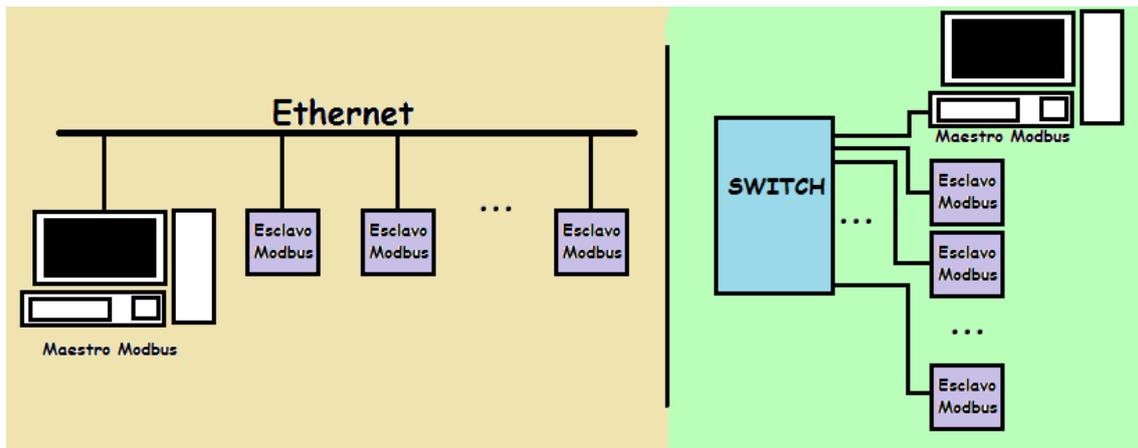


Figura 3. 3: Comunicación Modbus sobre Ethernet. Esquema lógico y físico

3.4 Descripción del protocolo

Como se ha explicado con anterioridad, sobre este soporte físico que se empleó para el desarrollo del proyecto, esto es, sobre bus en serie RS-485, podemos encontrar las variantes del protocolo **Modbus RTU** y **Modbus ASCII**. A continuación se van a detallar estas dos versiones de dicho protocolo.

3.4.1 Modbus RTU

El mensaje Modbus consiste en una trama con cuatro campos principales, que se pueden apreciar en la siguiente figura.

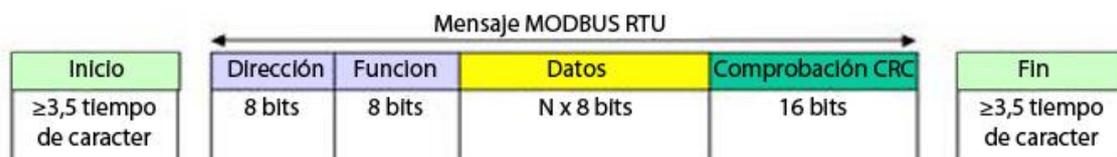


Figura 3. 4: Composición de mensaje Modbus con codificación RTU

Como se puede apreciar, los distintos campos tienen diferente tamaño y finalidad. A continuación se explica cada uno de los campos:

- **Dirección/Número de esclavo:** Es el número o dirección unívoca que determina a qué esclavo se refiere el mensaje. A este campo se le dedica 1 byte.
- **Código de operación o función:** Es un campo de 1 byte que permite indicar la función solicitada al esclavo. Estas funciones pueden consistir en la transmisión de datos u órdenes. A su vez, estas órdenes pueden ser de lectura/escritura de datos en los registros o en la memoria del esclavo, o por otra parte órdenes de control del esclavo y el propio sistema de comunicaciones (RUN/STOP, carga y descarga de programas, etc.).

- **Código de subfunciones/datos:** Este es un campo que suele contener, primeramente, los parámetros necesarios para ejecutar la función indicada por el byte anterior. Dichos parámetros pueden ser códigos de subfunciones de control, direcciones del primer bit o byte a leer o escribir, valor a escribir o cantidad de bytes para leer, entre otros ejemplos.
- **Palabra de control de errores (CRC):** Este campo se encarga de controlar la existencia de errores en el mensaje. En Modbus RTU este CRC se calcula con una fórmula polinómica según el siguiente algoritmo:

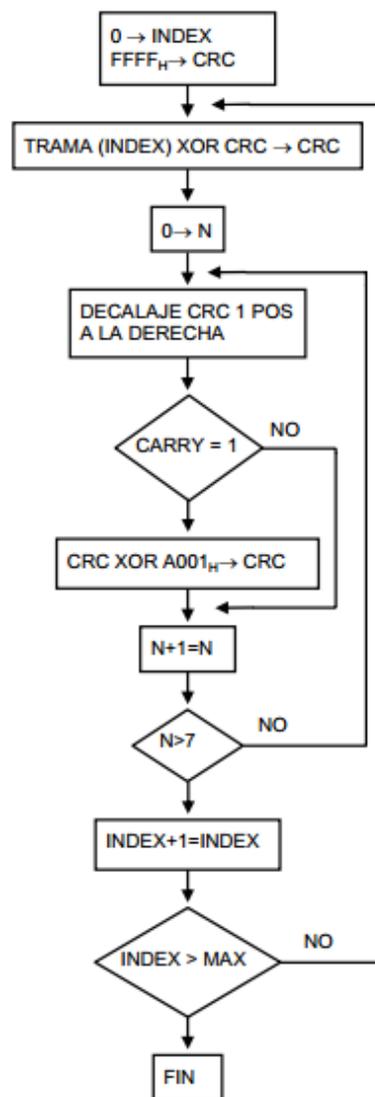


Figura 3. 5: Cálculo del CRC en Modbus RTU

3.4.2 Modbus ASCII

En esta variante, el mensaje consta de seis campos, a diferencia de su versión RTU en la que solo existen cuatro campos. Esta información adicional sirve para determinar el comienzo y el final del mensaje, como se puede observar en la siguiente figura.

Inicio	Dirección	Función	Datos	Comp. LRC	Fin del mensaje
1 car. :	2 caracteres	2 caracteres	Hasta 2 x 252 caracteres	2 caracteres	2 caracteres (Retorno y salto)

Figura 3. 6: Composición de un mensaje Modbus con codificación ASCII

Los campos que conforman un mensaje Modbus están destinados a una determinada función cada uno, ligeramente distinto a Modbus RTU. Al igual que en el caso anterior, se explican a continuación:

- **Inicio del mensaje:** Es una señal que indica la presencia de un nuevo mensaje. Consiste en el carácter ‘:’.
- **Dirección/Número de esclavo:** Es el número o dirección unívoca que determina a qué esclavo va dirigido el mensaje. Este campo tendrá un tamaño de dos caracteres.
- **Código de operación o función:** Este campo de dos caracteres permite indicar la función a realizar que, como en el caso de Modbus RTU, están determinadas.
- **Código de subfunciones/datos:** Este es un campo que suele contener, primeramente, los parámetros necesarios para ejecutar la función indicada por el campo anterior. Dichos parámetros pueden ser códigos de subfunciones de control, direcciones del primer bit o byte a leer o escribir, valor a escribir o cantidad de bytes para leer, por ejemplo, variando según el código de operación.
- **Palabra de control de errores (LRC):** Este campo se encarga de controlar la existencia de errores en el mensaje. En Modbus ASCII este control de errores se realiza por Comprobación de Redundancia Longitudinal (LRC), consistente en la suma de comprobación del mensaje o ‘*checksum*’ en módulo 16 expresado en ASCII. Consta de dos caracteres.
- **Fin del mensaje:** Se determina el final de mensaje mediante dos caracteres, concretamente el carácter especial **retorno de carro** seguido del **salto de línea**. Esto indica que el mensaje a leer ha concluido. Aporta la ventaja respecto a la variante Modbus RTU de que en este caso no se precisa dejar tiempo pasar para iniciar un nuevo mensaje, con lo que se puede transmitir información en menos tiempo para un determinado valor de velocidad de transmisión.

3.5 Formato de los datos

En esta sección se detalla el formato que presentan los datos en este protocolo, desde la manera de envío de cada baudio como el destino o las funciones más comunes.

3.5.1 Modbus RTU

Al comunicarse los equipos en un bus serie con protocolo Modbus utilizando el modo RTU, cada baudio de un mensaje contiene dos caracteres hexadecimales. Cada mensaje debe ser transmitido como una cadena continua de caracteres.

Cada baudio está formado por 11 bits. El formato para cada uno de los baudios transmitidos en el modo RTU es el siguiente:

- 1 byte corresponde a 8 bits de información binaria.
- 1 bit de comienzo.
- 8 bits de datos, enviando primero el bit menos significativo.
- 1 bit para completar la paridad.
- 1 bit de parada.

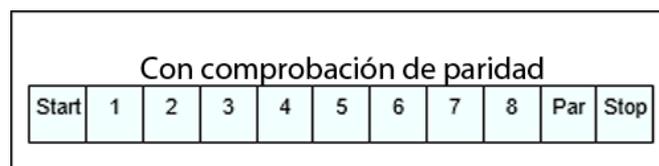


Figura 3. 7: Composición de un baudio de información en Modbus RTU

Aunque la paridad se requiera, otros modos pueden usarse también, de modo que para asegurar la máxima compatibilidad entre equipos es recomendado soportar el modo sin paridad. En el caso de no usarse paridad, se emplearán dos bits de parada.

3.5.2 Modbus ASCII

Cuando los dispositivos están configurados para comunicarse sobre un bus en serie y con protocolo Modbus mediante el modo ASCII, cada byte (8 bits) del mensaje es enviado como dos caracteres ASCII. Este modo se usa cuando el enlace físico de comunicación o el equipo no cumplen con los requisitos de temporización del modo RTU. Este modo es menos eficiente que dicho modo ya que cada byte necesita dos caracteres. Un ejemplo de ello es el siguiente: El byte 0x5B es codificado como dos caracteres, de la siguiente manera: 0x35 y 0x42 (ya que 0x35 es la codificación del carácter 5 en ASCII, y 0x42 corresponde al carácter B en dicha codificación).

Cada baudio está formado por 10 bits. El formato para cada uno de los baudios transmitidos en el modo ASCII es el siguiente:

- Codificación hexadecimal de los caracteres ASCII del 0 al 9 y de la A a la F.
- Cada carácter hexadecimal contiene cuatro bits de datos en cada carácter ASCII.

- 1 bit de comienzo.
- 7 bits de datos, enviándose primero el bit menos significativo.
- 1 bit de paridad.
- 1 bit de parada.



Figura 3. 8: Composición de un baudio de información en Modbus ASCII

Aunque la paridad se requiera, otros modos pueden usarse también, de modo que para asegurar la máxima compatibilidad entre equipos es recomendado soportar el modo sin paridad. En el caso de no usarse paridad, se emplearán dos bits de parada.

3.5.3 Registros y funciones Modbus

Los dispositivos que se comunican mediante Modbus tienen en común una determinada infraestructura interna a nivel de registros de información. Estos registros están distribuidos según el tipo de datos que contienen, que pueden ser binarios o palabras (16 bits). La posición de memoria de cada registro determinará su orden y, a su vez, el tipo de datos que contiene.

Para una mejor comprensión del reparto de los registros Modbus de memoria, y la correspondencia entre la dirección en un dispositivo y la dirección Modbus estándar para dichos registros, se adjunta la siguiente tabla:

Dirección dispositivo	Dirección Modbus	Descripción
00001...10000*	dirección -1	Registros de E/S discretos
10001...20000*	dirección -10001	Entradas discretas
30001...40001*	dirección -30001	Registros de entrada
40001...50001*	dirección -40001	Registros de almacenamiento
*El valor máximo depende del dispositivo Modbus		

Figura 3. 9: Direcciones de registros

Con los registros repartidos de esta manera el protocolo Modbus proporciona una serie de funciones para operar con los datos de modo que cada una de estas funciones se encarga de leer o escribir uno o varios registros de una zona de memoria concreta (zona de registros discretos, de entradas discretas, de registros de entrada o de registros de almacenamiento). Aquellas funciones que más se utilizan se describen en la siguiente tabla para su análisis y comprensión.

Código	Descripción
01	Lectura de registros discretos
02	Lectura de entradas discretas
03	Lectura de registros de almacenamiento
04	Lectura de registros de entrada
05	Escritura de un solo registro discreto
06	Escritura de un solo registro de almacenamiento
07	Lectura de excepciones
15	Escritura de n registros discretos
16	Escribir n registros de almacenamiento

Figura 3. 10: Funciones Modbus y su código de operación

En el Apéndice 2 que se adjunta, se explican en detalle las distintas funciones Modbus y los mensajes de error.

3.6 Requisitos para la comunicación

Para que la comunicación mediante el protocolo Modbus sea posible, previamente es necesario que se cumplan una serie de requisitos.

- Por una parte se han de dar las condiciones **físicas**, esto es, se necesita un medio físico que pueda servir como soporte al protocolo. Como se ha explicado con anterioridad, el medio físico puede ser un bus semidúplex o full dúplex. Cada uno de estos medios presentará unas distintas condiciones con lo que la elección dependerá de lo que mejor se ajuste a las circunstancias a nivel de velocidad, distancia o presencia de ruido.
- En segundo lugar, conectados todos los elementos en el mismo bus, se hace evidente la necesidad de que todos estos dispositivos estén **preparados** para operar mediante Modbus con la misma **codificación** (RTU/ASCII/TCP).
- Como en toda comunicación, para el correcto funcionamiento del sistema será necesario que **todos los dispositivos** hayan sido previamente configurados para funcionar a una misma velocidad de transmisión, y con una misma configuración en cuanto a paridad y control de flujo, además de tener correctamente asignado el puerto por el que se va a leer.
- Todos los dispositivos deben tener claramente determinado su *rol* en el bus, de modo que todos los esclavos han de tener correctamente configurada una dirección de esclavo que el usuario necesitará conocer para que el maestro pueda comunicarse con el dispositivo correspondiente.
- Aunque los anteriores requisitos permitan la correcta comunicación Modbus se hace importante también la necesidad de contar con un **software** en el equipo maestro que se encargue de **gestionar la información** (tanto las órdenes que se envían hacia los esclavos como los datos que éstos devuelvan).

3.7 Ventajas y desventajas de Modbus

Simple y robusto este protocolo de comunicaciones es ya un estándar de facto por las ventajas que ofrece, de las cuales se destacan las siguientes:

- Modbus es un protocolo público, con lo que su implementación y aprovechamiento no supone costes adicionales por permisos de uso.
- Se desarrolló con propósitos de aplicación industrial, y por ello es idóneo para este y otros proyectos de dicha índole, ya que se adapta a las circunstancias.
- Debido a su simpleza, es un protocolo muy fácil de implementar. Esto implica que, a su vez, sea también fácil de mantener.
- Maneja bloques de datos sin suponer restricciones a los vendedores. Esto se debe a que, desde el punto de vista del modelo OSI, Modbus es un protocolo de aplicación.
- Está reconocido y soportado por la mayoría de los dispositivos disponibles en el mercado.

Estas ventajas hacen de Modbus el candidato ideal a emplearse en este proyecto frente a otros protocolos similares. No obstante, existen también una serie de limitaciones que se indican a continuación:

- Debido a su antigüedad, ciertos tipos de datos no están soportados, limitándose a aquellos entendidos por los PLCs de la época.
- Por su condición de protocolo tipo maestro/esclavo, no hay forma de saber si ocurrió alguna excepción a menos que el maestro compruebe periódicamente cada esclavo. Esto consume ancho de banda y tiempo de ejecución.
- El direccionamiento Modbus está limitado a 247 dispositivos en el bus, limitando el número de dispositivos de campo que pueden conectarse a un maestro.
- El protocolo Modbus no proporciona seguridad contra interceptación de datos o comandos desautorizados.

La escasa medida en la que los inconvenientes mencionados suponen un problema, unida a las ventajas que aporta el protocolo, hacen de Modbus la mejor elección para poder desarrollar el proyecto frente a otras alternativas similares.

Capítulo 4

Desarrollo del sistema de comunicación

4.1 Introducción

Una vez expuestas las motivaciones (la mejora de la eficiencia energética) y el protocolo de comunicaciones (Modbus) utilizado para este proyecto, se explica en detalle el desarrollo del sistema que realizará las comunicaciones pertinentes para poder llevar a cabo el proyecto.

La **Comunicación** es la transferencia de información con sentido desde un lugar a otro. Por otra parte la **Información** es un patrón físico al cual se le ha asignado un significado comúnmente acordado. El patrón debe ser capaz de ser enviado por el transmisor, y capaz de ser detectado y entendido por el receptor.

Si la información es intercambiada entre comunicadores humanos, por lo general se transmite en forma de sonido, luz o patrones de textura en forma tal que pueda ser detectada por los sentidos primarios del oído, vista y tacto. El receptor asumirá que no se está comunicando información si no se reciben patrones reconocibles.



Figura 4. 1: Esquema de modelo básico de comunicaciones.

En la anterior figura se muestra un diagrama a bloques del modelo básico de un sistema de comunicaciones, en éste se muestran los principales componentes que permiten la comunicación, que son los siguientes:

- **Transmisor:** Es quien transmite el mensaje, o dicho de otro modo, es quien envía al medio o canal la información que desea transmitir.
- **Canal:** Es el medio por el que viaja la información hasta llegar a su destino, el receptor. Sirve como enlace entre transmisor y receptor, y puede ser desde el mismo aire hasta un alambre, un cable o fibra óptica.
- **Receptor:** Es quien recibe la información. Se encargará de interpretar dicha información para extraer el mensaje.

En base a esta breve explicación, se puede entender que el sistema que a continuación se presenta sigue de algún modo un patrón similar, aunque más complejo.

4.2 SCADA

Entrando de lleno en el campo de los sistemas de comunicación, podemos encontrar en la actualidad un amplio abanico de posibilidades, desde algunas topologías omnipresentes y con múltiples aplicaciones, hasta otros sistemas de comunicación menos conocidos, para fines más específicos y posiblemente más complejos.

En el caso de este proyecto, la monitorización de la energía que se consume para el aumento de la eficiencia energética, resulta muy inmediato pensar en un sistema SCADA (Supervisory Control and Data Acquisition) como solución eficaz para gestionar las comunicaciones.

SCADA es un tipo de sistema de control industrial. Los sistemas de este tipo se caracterizan por estar controlados por ordenador y sirven para monitorizar y controlar los procesos industriales que existen en el mundo real. Desde siempre, los sistemas SCADA se desmarcan de los otros sistemas de control industrial por su infraestructura de gran escala, que puede implicar presencia en múltiples lugares y largas distancias de separación.

Una de las aplicaciones principales de estos sistemas es la basada en instalaciones. Se presenta tanto en instalaciones públicas como privadas, tales como edificios, aeropuertos, barcos... Monitorizan y controlan temperatura, ventilación, acceso de personas, sistemas de aire acondicionado y, también, **consumo de energía**. Para este proyecto, es éste último el dato que interesa poder monitorizar de modo que, aun sin llevar a cabo labores de control, pueda contribuirse al ahorro en dicho consumo.



Figura 4. 2: Ejemplo SCADA.

4.2.1 Elementos comunes

Un sistema SCADA consta habitualmente de una serie de subsistemas, que a continuación se detallan:

- **Dispositivos de adquisición de datos:** Éstos recogen la información de distintas condiciones que se desean medir gracias a sus sensores (p.ej. temperatura, humedad, tensión...) y las transmiten al proceso central. Pueden ser terminales remotas específicas o, más habitualmente debido a su bajo

coste, PLCs. La información analógica captada por dichos sensores es convertida en valores digitales que luego pueden ser transmitidos.

- **Sistema supervisor o proceso servidor:** Es el proceso que se encarga de reunir los datos que le son transmitidos por los dispositivos de adquisición. Una vez reunidos dichos datos, los analiza, almacena y los transmite a las posibles interfaces de usuario. Además, en función de los datos recibidos, puede enviar órdenes para controlar el proceso de toma de datos.
- **Base de datos:** Es importante que exista un respaldo de la información para evitar la pérdida de los datos en caso de que se produzca cualquier problema, así como permitir examinar la evolución histórica de los datos. A su vez, la base de datos puede estar replicada para mayor seguridad.
- **Interfaz de usuario:** Así como los datos recibidos por el proceso central son almacenados como réplica en la base de datos, otra réplica de los mismos es enviada a las posibles interfaces de usuario. Estas pueden ser desde un panel físico en la misma industria hasta aplicaciones web a las que acceder desde cualquier navegador o aplicaciones que manejar desde un PC o un teléfono inteligente.

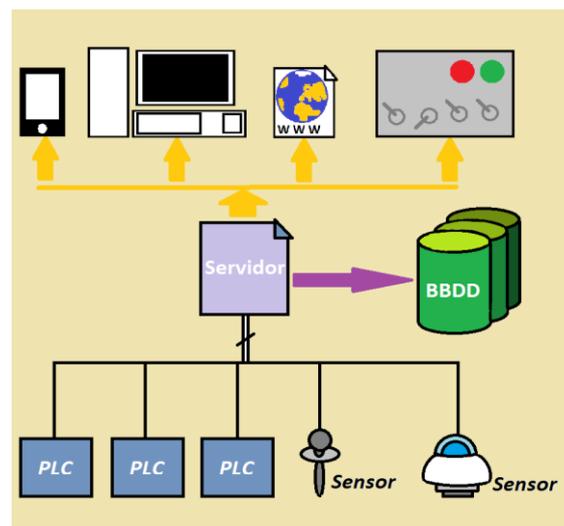


Figura 4. 3: Esquema de topología SCADA.

4.2.2 Concepto y aplicación

El término SCADA se suele referir a sistemas centralizados que **monitorizan** o **controlan** estancias enteras, o sistemas complejos que se distribuyen por grandes superficies. La mayoría de acciones de control son ejecutadas automáticamente por los terminales remotos o los PLCs, pero dentro de unas limitaciones concretas; es el propio sistema SCADA quien se encarga de gestionar o modificar las condiciones en las que estos equipos operan. Un ejemplo de esto consistiría en un sistema cuyos PLCs controlasen la refrigeración del agua como parte de un proceso industrial, no obstante el sistema SCADA sería quien permitiría a los operadores cambiar los valores críticos a medir y activar condiciones de alarma por, por ejemplo, temperatura excesiva.

Como se explicó de un modo más esquemático, la **adquisición** de datos comienza en los PLCs o terminales remotos con lectura de valores o medidas e informes de estado de los equipos que están comunicados al SCADA. Los datos son entonces **interpretados** y ajustados de modo que un operador pueda, gracias a la **representación** por la interfaz gráfica, tomar decisiones de supervisión. Estos datos a la vez son almacenados en un historial y, a menudo, insertados en una base de datos para permitir análisis más precisos.

Los sistemas SCADA son sistemas realmente importantes que además son utilizados en **infraestructuras nacionales** como tendidos eléctricos, suministro de agua y tuberías. En lo que se refiere a este proyecto, el objetivo que se persigue es un sistema SCADA de monitorización del consumo de la energía para conseguir una mejora de la eficiencia energética. Aunque el sistema se centra solo en el apartado de **Iluminación**, con la que la presencia de medidores o PLCs está limitada a simples analizadores de redes, forma parte de un sistema más complejo que también busca mejorar la eficiencia en cuanto a climatización, refrigeración, caldera o aire comprimido entre otras.

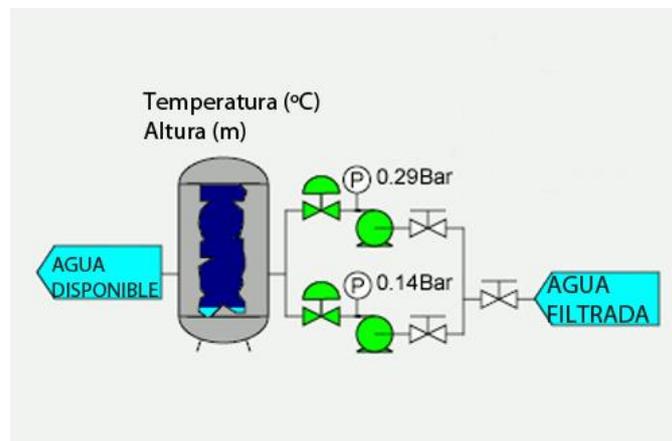


Figura 4. 4: Ejemplo de sinóptico SCADA.

4.3 Escenario de trabajo

El escenario de trabajo que se presenta para el proyecto consta de tres partes fundamentales, que corresponden a: la **adquisición** de datos reales, la gestión de los datos, y la presentación al usuario.

4.3.1 Adquisición de los datos

Uno de los primeros y principales pasos a dar es la **obtención de datos reales** del entorno físico. Este proceso es muy importante ya que será gracias a los datos obtenidos como el sistema realice cálculos, genere nuevos datos y regule el comportamiento.

Esta labor requiere de un equipo que sirva de **punto** entre los **datos físicos** y la **información digital**. En este caso, este punto consistirá en un analizador de redes. Un analizador de redes es un equipo dotado de una serie de entradas de tensión y de corriente eléctrica, gracias a las cuales se obtienen los datos de dicha red eléctrica, tales como la tensión a la que está sometida, la intensidad de corriente que la recorre, las potencias de distinto tipo (activa, reactiva, aparente), energías... y en analizadores algo más sofisticados, tenemos otros

parámetros más avanzados como los armónicos de tensión e intensidad que existen en la línea o la distorsión total que producen en la señal eléctrica.

El analizador de redes Sentron PAC3100 [PAC3100] de Siemens ha sido el equipo empleado para obtener los datos de la red eléctrica del centro de consumo encargado de la iluminación.



Figura 4. 5: Analizador de redes Sentron PAC3100.

En el sistema global que se quiere crear (más allá de este proyecto) presentaría un analizador de redes del mismo tipo por cada uno de los cuadros eléctricos encargados de los distintos consumos principales (climatización, frío industrial, aire comprimido, etc.). No obstante, en el cuadro eléctrico principal se instalaría un modelo más avanzado de analizador de redes, el Sentron PAC4200 [PAC4200] del mismo fabricante. Este modelo permite funciones avanzadas como distintos contadores para evaluar la energía eléctrica consumida en un período determinado por el usuario, o lo que es más importante, análisis de los armónicos que existen en la línea y cómo estos perturban la señal eléctrica. También se ha de aclarar que en algunos de los centros de consumo existirán, además de un analizador de redes, distintos tipos de **sensores**, como podría ser un sensor de presión (en el módulo de Aire Comprimido, o termopar en los módulos de Climatización y Frío Industrial).

De este modo, con un analizador de redes en el cuadro eléctrico principal, y otro en cada uno de los distintos cuadros de consumo, se mantendrían medidos los distintos consumos eléctricos para una mejor información del gasto energético y su distribución.

Por otra parte, es necesario indicar que cada analizador de redes cuenta con una **salida digital** para que la información que miden (la cuál puede ser consultada en su pantalla) pueda ser también **transmitida** a otro equipo para su análisis y gestión.

4.3.2 Gestión de datos

El proceso de gestión de los datos tiene su importancia en el **cálculo** de valores de diferentes medidas a raíz de la información recogida del analizador de redes en este proyecto, y del conjunto de analizadores de redes y distintos sensores en el sistema global.

En la manera en que este proyecto está planteado, la gestión de datos se lleva a cabo por software, sin la utilización de equipos específicos más allá de ordenadores corrientes. Se pueden encontrar principalmente estas labores de gestión de los datos:

- **Introducción de nuevos datos:** El software que se quiere diseñar necesita, en determinadas ocasiones, la **intervención del usuario** para aportar datos que, pese a referirse a la red eléctrica, no pueden ser determinados por los analizadores de redes o sensores. Estos datos se introducirán al programa por la **entrada estándar** (teclado y ratón). Para su introducción, el software consta de diversos campos de numéricos, de texto y botones.
- **Cálculo de nuevos datos:** A raíz de la gran cantidad de datos que se reciben del analizador de redes, así como de los datos introducidos por el usuario, existen una serie de valores o nuevos datos que han de obtenerse como resultado de **operaciones** entre ellos. Es el caso, por ejemplo, de valores como la potencia disipada en los conductores o el índice de eficiencia energética de iluminación.
- **Replicación y respaldo de los datos:** Los datos que recibe el software por parte del analizador de redes, así como aquellos guardados posteriormente incluyendo los nuevos datos obtenidos como resultado de las operaciones entre éstos y los datos introducidos por el usuario, son almacenados en un **servidor remoto**. Esto es muy importante ya que proporciona al sistema una robustez ante posibles fallos en el hardware del equipo local. En el equipo servidor se encontrarán los datos diarios guardados de cada entidad donde el sistema haya sido implementado, debidamente aislados unos de otros y con acceso únicamente para su propietario.

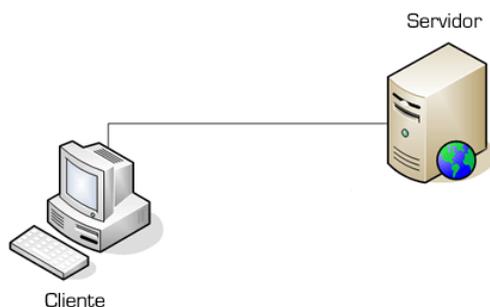


Figura 4. 6: Esquema equipo con servidor de respaldo.

4.3.3 Representación de los datos

La **representación** en un **medio físico** de los datos puede realizarse en un panel de control o una pantalla de ordenador o teléfono inteligente, a través de una aplicación o de una página web.

En este escenario de trabajo, la representación de los datos está principalmente pensada para producirse a través de una **pantalla de ordenador**. Este ordenador puede ser el mismo ordenador central que ejecuta todo el software y sus procesos, o puede ser, de un modo más conveniente, otro pc que acceda al primero a través de acceso remoto.

Todo el conjunto de datos que se han obtenido y procesado se muestran en distintas ventanas o pestañas del software diseñado, de modo que la información se presente de la forma más clara, ordenada e intuitiva posible. En el ordenador central, esta información se encuentra en continuo cambio ya que los datos se reciben de manera **constante**, y del mismo modo se podrá reflejar en cualquier ordenador que acceda remotamente al mismo. Por otra parte, la réplica de la información de forma diaria en el servidor permite que desde **otras plataformas**, como ordenadores externos o teléfonos inteligentes (a través del web o de apps) se puedan, aunque con menor precisión, **consultar** los datos gestionados.

4.4 Diseño del software

A la hora de diseñar el software, es muy importante tener en cuenta las funciones que se pretende llevar a cabo, de modo que pueda implementarse un software que responda a todas estas tareas, aunque éste a su vez se encuentre subdividido en varios **procesos**. Otro dato a tener en cuenta es la manera o las **herramientas** con las que llevar a cabo el **diseño** y la **implementación** del software. En la actualidad existen una gran cantidad de posibilidades, en lo que a lenguajes de programación y entornos de desarrollo se refiere. Pese al **total desconocimiento** del **entorno de programación LabVIEW** [LabVIEW] y su **código G** de programación gráfica al momento de comenzar a abordar este proyecto, el tipo de sistema que se quiere diseñar sugería a LabVIEW como la mejor herramienta de programación, con lo que la realización de este proyecto ha conllevado y a su vez supuesto un intenso, concienzudo y presto aprendizaje acerca de este entorno, su funcionamiento, sus posibilidades y el concepto de la programación gráfica. Este desconocimiento ha propiciado que parte del desarrollo del proyecto avanzase con bastante detenimiento, debido a ocasionales prácticas de programación mediante **ensayo-error**.

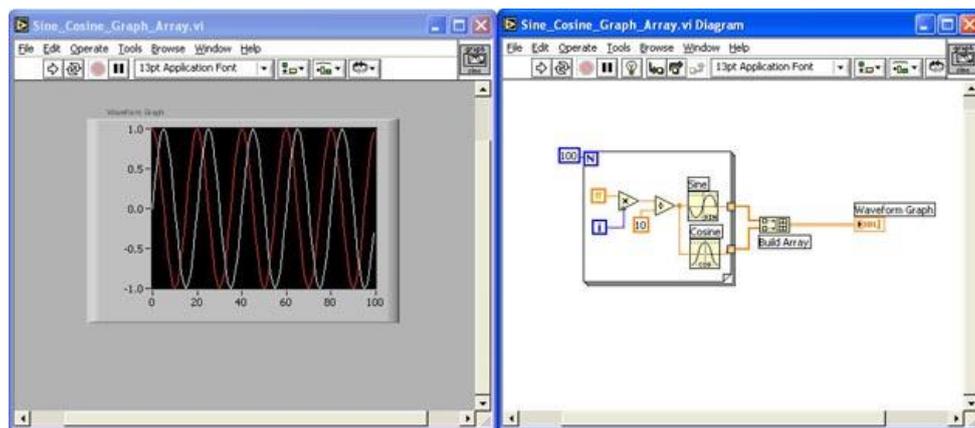


Figura 4. 7: Entorno de programación LabVIEW.

4.4.1 Diseño del proceso de adquisición

Esta etapa fue la que más se retrasó a la hora de ser realizada. Poder llevar a cabo la adquisición de los datos, en el escenario y con las tecnologías que se han planteado, consiste en permitir la comunicación de los dispositivos Modbus con el ordenador central, lo cual

supone que todos los equipos “**se entiendan**” (es decir, que estén configurados para operar de la misma manera) y que el ordenador sea capaz de recoger e interpretar la información, así como de solicitarles nueva información a los demás dispositivos. Para que este proceso se realizara de la mejor manera posible, se plantearon distintos escenarios y se intentó trabajar sobre ellos.

4.4.1.1 CompactRIO

En un principio, en la búsqueda de una **robustez** industrial idónea para que el sistema no se encontrara demasiado expuesto a las **condiciones adversas**, así como por facilidad de conexiones, se planteó el funcionamiento del software sobre un **controlador industrial** embebido de tiempo real, el CompactRIO 9076[CompactRIO 9076] de National Instruments. Éste es un dispositivo fácilmente programable y capaz de trabajar y ejecutar su software de manera **autónoma** solo con estar debidamente alimentado. El modelo 9076 en concreto se presentaba como ideal por su puerto RS-232. El modo de trabajo consistiría en programar desde un PC auxiliar para, una vez terminado el software, **implementarlo** sobre el CompactRIO (con el que la comunicación se realiza via Ethernet) y que éste se encargue de ejecutar el software. De este modo y bajo este planteamiento, al ser este el primer paso fundamental para el proyecto a realizar, los primeros avances y la primera toma de contacto con el entorno de programación LabVIEW consistió en desarrollar un cliente Modbus simple, que **comunicara** con uno de los dispositivos del bus (se probó con uno de los analizadores de redes) para obtener en el sistema software el contenido de uno de sus **registros** (es decir, uno de los valores que se está midiendo y representando en su display).

Esta pequeña aplicación es la base desde la que se partió para, mediante **continuas modificaciones** que permitieran ajustarla a cada necesidad concreta, llevar a cabo cualquier recogida de datos o cualquier tipo de comunicación Modbus.

Empezando a familiarizarse con la comunicación Modbus para toma de datos, el siguiente avance (en este caso sobre una aplicación más sofisticada capaz de leer tres campos consecutivos de datos del analizador de redes) fue ejecutar por primera vez el código sobre el controlador industrial, de forma **continua**.

El puerto serie del que consta el controlador es, como se especificó, del tipo RS-232, con lo que entre el bus y el puerto se utilizó un pequeño **convertidor** de RS-485 al tipo del que el CompactRIO incorpora.

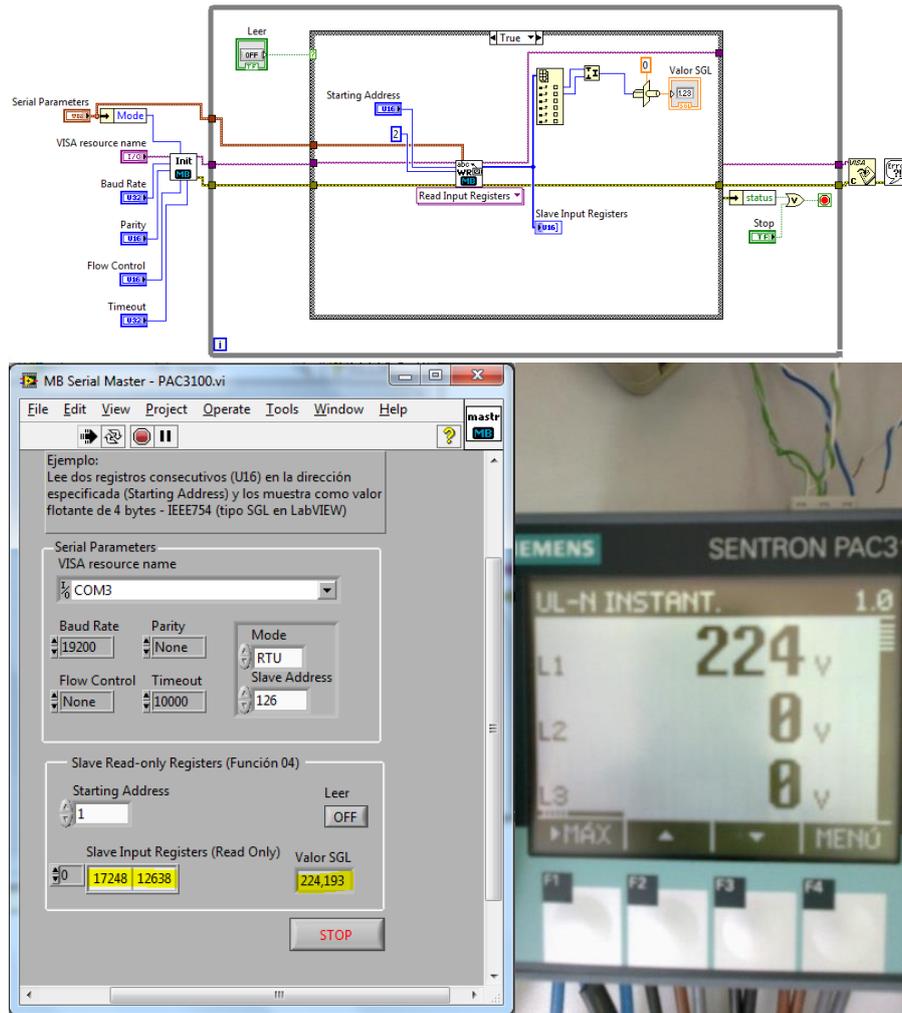


Figura 4. 8: Arriba: La programación en código gráfico de la aplicación. Izquierda: La interfaz de la aplicación resaltando el valor medido. Derecha: Display del analizador con el valor medido.

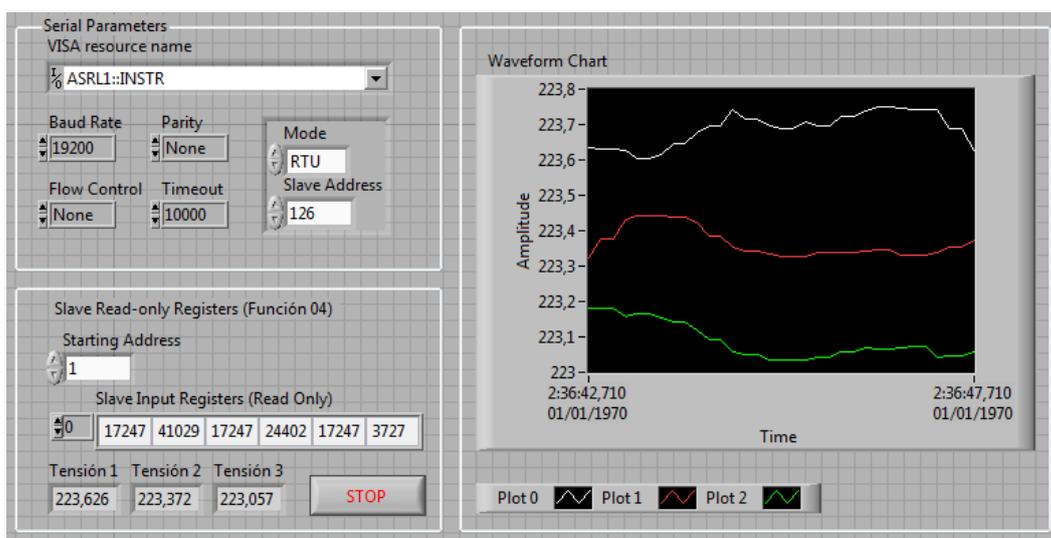


Figura 4. 9: Primera aplicación programada en el CompactRIO.

4.4.1.2 PC antiguo

Cuando la cantidad de datos a recoger se vio muy incrementada, la capacidad que el CompactRIO presentaba se vio **insuficiente**. Además, la programación más compleja con referencia a archivos y rutas se complicó debido al distinto formato de rutas entre un sistema Windows (el pc desde el cuál se programa) y el formato de ruta del CompactRIO. Fue por ello que, pensando principalmente en la conectividad sin necesidad de incluir demasiados conversores intermedios, se optó por ejecutar el software sobre un antiguo **pc** que constara de **puerto serie**.

No obstante, y pese a que el software funcionaba correctamente, no poder contar con el mismo tipo de ordenador para cada caso al ser equipos descatalogados, así como la incompatibilidad de los sistemas antiguos con un solo procesador **mononúcleo** con ciertas funciones avanzadas de programación de LabVIEW, específicas para trabajar más eficientemente haciendo uso de los **múltiples núcleos** de los procesadores, supuso el abandono de esta plataforma.

4.4.1.3 Mini-PC

La solución definitiva sobre la que implementar el software consiste en un **mini-pc**. Con bajo coste, gran versatilidad en prestaciones y tamaño reducido, resulta la opción ideal.

Pese a no contar con un puerto serie RS-232, sus características son similares a las de cualquier pc de gama media, lo que hace que el desempeño sea totalmente **eficiente**. Para la conexión de los distintos dispositivos del bus al mini-pc, se emplea un **convertidor** USB a RS-485.



Figura 4. 10: Mini-pc. Por comparación con el conector VGA se aprecian las dimensiones del equipo.

Con la plataforma idónea encontrada, la etapa de adquisición de datos quedó acabada como un **proceso** al que previamente se le indicasen los analizadores presentes, para intentar comunicarse uno a uno con los que se encuentran presentes vía Modbus, contemplando además que tanto los analizadores PAC4200 y PAC3100 como la cantidad de datos que han

de recoger son distintos. Los datos son almacenados en variables compartidas, gracias a las cuales otros procesos del software, como el de gestión o el de representación pueden recoger dichos datos y operar con ellos.

4.4.2 Diseño del proceso de gestión

Sentada la base de la recogida de datos, así como la plataforma definitiva sobre la que funcionar, el siguiente paso a dar es programar la gestión de la información.

Que el proyecto esté centrado en el apartado de la **Iluminación** implica que los datos a gestionar sean referidos a este módulo. No obstante, también se recogen otros datos menos evidentes como los datos de la instalación **general** o la **configuración**.

4.4.2.1 Configuración

Casi más importante que el propio proceso de gestión de los datos, es la gestión de la configuración, ya que será la que determine cómo ha de comportarse el software.

Esta es introducida por el usuario, que determinará cuál será la dirección de esclavo Modbus del analizador de redes **principal** (PAC4200) (de modo que el o los siguientes analizadores de redes tendrán como dirección un número consecutivo menor sucesivamente).

Otros datos que también se han de determinar son la velocidad de bit, paridad, control de flujo y bits de parada. Por último, se obtiene la información sobre cuántos y cuáles analizadores de redes están presentes, de modo que durante la ejecución, el programa no malgaste tiempo en intentar comunicarse con analizadores no están presentes.

Cuando la configuración ha sido elegida y aceptada, es guardada tanto en una variable compartida, para que el software la lea y la respete, como en un archivo de tipo XML para que, en futuras ejecuciones, siga siendo la misma.

4.4.2.2 Datos globales medidos

Todo el conjunto de datos que se reciben del analizador de redes PAC4200 que es instalado en el cuadro principal de la instalación eléctrica, es transmitido como una gran **variable compartida** al programa principal. Al ser datos muy específicos y muy concretos, no se opera de ninguna manera con ellos, salvo para **representar** los más comunes en pantalla mediante indicadores. No obstante, es importante que se recojan para posibles operaciones por parte de otros procesos del software.

4.4.2.3 Datos de Iluminación

Los datos para el módulo de Iluminación vienen dados tanto por el **analizador** de redes conectado al cuadro eléctrico dedicado a la iluminación (PAC3100), como por el **usuario**, que introduce ciertos datos por la entrada estándar.

Los datos que recoge el analizador determinan parámetros como la tensión y la intensidad de la corriente, la potencia total que cae en cada fase y por tanto la potencia y la energía que se consume en Iluminación. Por otra parte, la introducción de datos por parte del usuario determinará más **específicamente** cómo se realiza el consumo, introduciendo cada zona de iluminación con su superficie, longitud y sección de los conductores, número de lámparas y de luminarias.

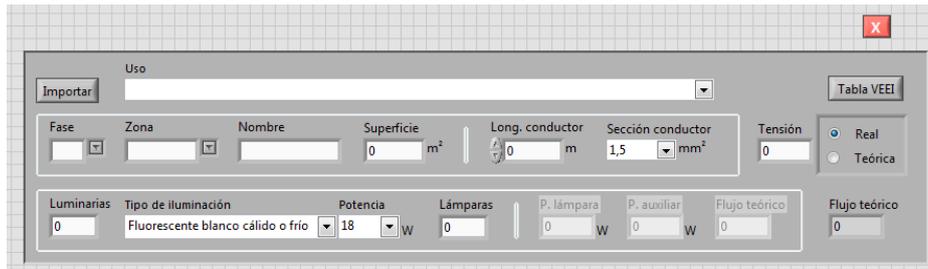


Figura 4. 11: Introducción de datos del usuario

El conjunto de datos de iluminación es guardado en un archivo con formato XML, que se lee en cada ejecución del programa de modo que éste no pierda los datos que se introdujeron en anteriores ocasiones.

4.4.3 Diseño del proceso de representación

El proceso de representación de los datos va estrechamente ligado a los anteriores procesos de adquisición y de gestión. Esto se debe a que el entorno de programación gráfica LabVIEW está muy orientado a **interfaces**, con lo que cada unidad de datos suele tener una representación mediante un indicador.

En un principio la importancia de este proceso quedó relegada a un segundo plano ya que interesaba obtener los datos y que la ejecución del software fuera correcta. Es por ello que las primeras interfaces de la aplicación fueran muy **monótonas** y a su vez detalladas en exceso, a partir de indicadores numéricos en abundancia.

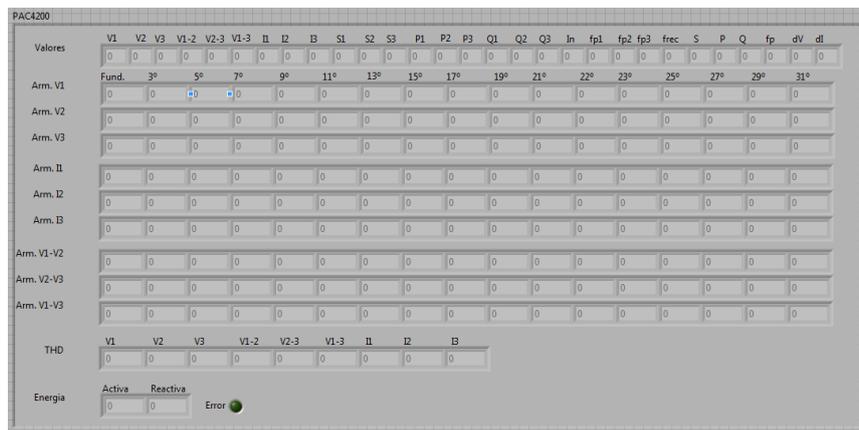


Figura 4. 12: Interfaz primitiva.

Con la funcionalidad del software conseguida, el siguiente paso a dar fue conseguir una **interfaz cuidada**, que fuera eficiente e intuitiva para el usuario. Además, se buscó que en la propia interfaz del programa se hiciera referencia a las entidades que intervienen en algún punto de la realización, implementación o gestión de este software, mediante incorporación de los logotipos de dichas entidades.

LabVIEW es un entorno con el que programar está, como se ha dicho anteriormente, muy orientado a interfaces. Para representar los datos se basa en indicadores que pueden ser distintos según el **tipo** de dato que representan: por ejemplo, para representar datos de texto facilita campos de texto, para representar datos booleanos ofrece indicadores que simulan LEDs, para datos numéricos existen desde el simple indicador numérico hasta indicadores de tipo termómetro o cuenta-revoluciones. Más allá, LabVIEW cuenta con distintas librerías, lo que permite que estos indicadores puedan tener diferente aspecto, desde una apariencia clásica con colores planos hasta otra más moderna. Gracias a la librería '*Silver*' de LabVIEW, se ha conseguido dotar al programa de una apariencia **sencilla** y a la vez **moderna**.

4.5 Conexión entre equipos

Para el correcto funcionamiento del sistema, debe existir **conexión** entre todos los equipos que lo conforman. Encontramos entonces un sistema principal de comunicaciones que ya se explicó con anterioridad, el bus de dos hilos RS-485 con protocolo Modbus. No obstante, también encontramos otro tipo de conexiones.

4.5.1 Analizadores de redes

Los analizadores de redes cuentan con un panel trasero en el que se encuentran las entradas y salidas analógicas y digitales. Tales entradas y salidas son las siguientes:

- **IL1(↑), IL2(↑) e IL3(↑)** son las entradas de corriente. Cada una de las fases del cuadro correspondiente de la instalación eléctrica se han de conectar a estas entradas.
- **IL1(↓), IL2(↓) e IL3(↓)** son las salidas de corriente. Como por todos es sabido, la intensidad de la corriente es un factor que se mide en **serie**, por ello las mismas líneas que entran al equipo deben salir de él.
- **V₁, V₂, V₃ y V_N** son las entradas para medir tensión de las tres fases con respecto al neutro.
- Por otra parte, los analizadores de redes cuentan con la interfaz RS-485 para la transmisión de datos al bus. Esta interfaz cuenta de tres terminales, a saber **Com, +/B y -/A**, de los cuales utilizamos los dos últimos.

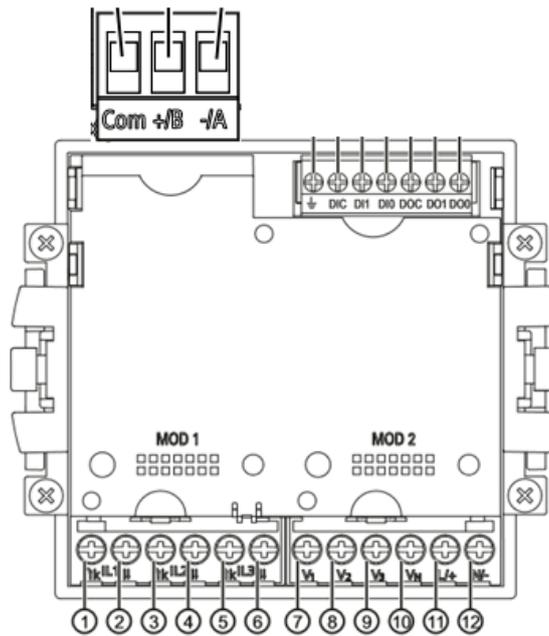


Figura 4. 13: Panel trasero del PAC4200.

4.5.2 Ordenador central

Recibe la información procedente de los analizadores de redes y, en caso de existir, otros dispositivos. Para ello debe formar de la misma red de comunicaciones. Es por ello que consta de un convertidor externo **USB a RS-485** que permite que reciba los datos y que envíe instrucciones al bus a través del puerto USB.

Por otra parte, cuenta también con conectividad tanto Wi-Fi como Ethernet para la comunicación a través de internet con el servidor, así como para que otros equipos puedan conectarse remotamente a éste.

4.5.3 Equipo servidor

Este equipo recoge y guarda la información que se le envía desde el ordenador central. Para ello hace uso de su conexión a Internet mediante su puerto Ethernet.

Capítulo 5

Interfaz de usuario del sistema

5.1 Introducción

La **interfaz de usuario** es el medio con que el usuario puede comunicarse con una máquina, un equipo o una computadora, y comprende todos los puntos de contacto entre el usuario y el equipo. Normalmente suelen ser fáciles de entender y fáciles de accionar.

Las interfaces básicas de usuario son aquellas que incluyen **elementos** como menús, ventanas, teclado, ratón, los *beeps* y algunos otros sonidos que la computadora hace, y en general, todos aquellos canales por los cuales se permite la **comunicación** entre el ser humano y la computadora. La mejor interacción humano-máquina a través de una adecuada interfaz, que le brinde tanto comodidad, como eficiencia.

Todo software orientado a interactuar con el usuario ha de contar con una interfaz. Si bien antiguamente éstas estaban basadas en **texto**, con la llegada de los primeros sistemas operativos basados en entorno gráfico llegó la tendencia y prácticamente obligación de crear aplicaciones cuyas interfaces fueran **gráficas** y fáciles de entender y manejar.

En este capítulo se explicará la **interfaz** de usuario del software que se ha diseñado para este Sistema de Gestión Energética y, a grandes rasgos, el **funcionamiento** del programa. También se explicarán algunos **procesos** que funcionan de manera transparente al usuario, cuya interfaz innecesaria existe únicamente por la manera en la que el entorno LabVIEW de programación funciona.

5.2 Interfaz principal

Al iniciar la aplicación, lo primero que se mostrará es la **interfaz principal**. De este modo, el usuario tendrá ante sí esta pantalla en la que los botones, acompañados de pequeñas imágenes descriptivas, le permitirán acceder a los distintos **módulos** del sistema. En este proyecto solo los módulos de Configuración, Cuadro General e Iluminación serán los que se encuentren disponibles.

Mientras la interfaz principal brinda al usuario las opciones de manejo del programa (los diferentes módulos a los que acceder), **internamente** el software se encuentra recogiendo la información procedente de analizadores de redes, procesándola y efectuando otra serie de operaciones que no resultan tan evidentes. Se cuenta con los siguientes botones/acciones:

- **Botón de Cerrar:** Detiene todos los procesos y cierra la aplicación.
- **Botón Configuración:** Lleva a la interfaz de configuración del software.

- **Botón Cuadro General:** Lleva a la ventana de información referente a los datos obtenidos del cuadro eléctrico principal, recogidos por el analizador de redes PAC4200.
- **Botón Iluminación:** Abre la interfaz de Iluminación, que muestra los datos que son recogidos en el cuadro eléctrico dedicado a iluminación, así como los que introduce el usuario. Además, permite dicha introducción de información sobre la iluminación de los distintos habitáculos del lugar en el que se quiere implantar el sistema.

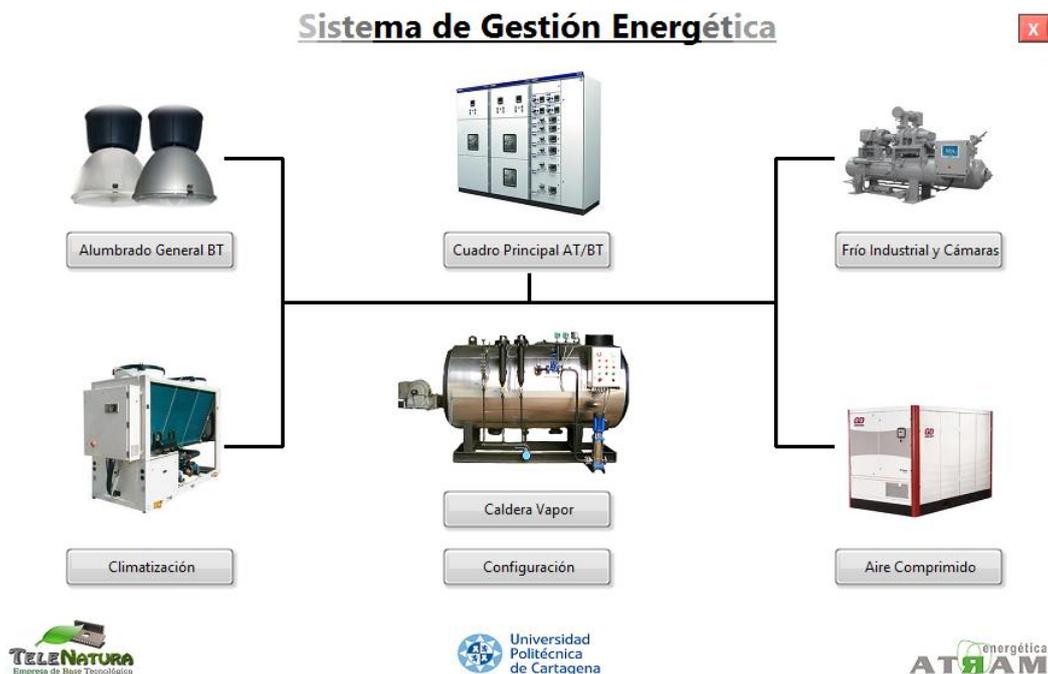


Figura 5. 1: Interfaz principal de la aplicación.

5.3 Configuración

La interfaz de configuración se encarga de recoger, por parte del usuario, así como de mostrar la **configuración** bajo la cual el software debe de funcionar. Esta ventana dedicada a la configuración puede aparecer bajo dos motivos:

- Se **presiona** el botón de Configuración en la interfaz principal de usuario. Esto provoca la llamada a la correspondiente ventana.
- Se inicia el software por primera vez. Al inicio, el programa comprueba la **existencia** del archivo de configuración. Si no existe, la aplicación llama a las funciones de configuración, de modo que el funcionamiento no sea errático.



Figura 5. 2: Ventana de Configuración.

5.3.1 Dirección de esclavo Modbus

Gracias a este campo, se selecciona la dirección del **esclavo** Modbus que corresponde al analizador de redes **principal**. Esta dirección se puede conocer comprobando en el propio analizador de redes su dirección. Por defecto, esta dirección es la 127. El funcionamiento del programa está previsto para que el resto de analizadores de redes sigan un **orden** en cuanto a la **numeración** de la dirección Modbus, de modo que el siguiente analizador tenga como dirección una unidad menos que el anterior. De este modo, el equipo encargado de recoger la información del cuadro eléctrico principal tiene por dirección Modbus la 127, mientras el equipo encargado de la **Iluminación** tiene la dirección 126. En este campo se **determina** la primera dirección con la que el programa intentará comunicarse para recoger la información.

5.3.2 Parámetros de la comunicación

Para que la comunicación con la red Modbus pueda ser llevada a cabo correctamente, es necesario que todos los dispositivos se encuentren **configurados** para trabajar de la misma manera. Es por ello que la ventana de configuración cuenta con los siguientes campos para modificar los parámetros de la comunicación:

- **Baud rate (tasa de baudios):** Es la velocidad con la que se transmitirán las unidades de señal (baudios). Por defecto está configurado a 19200 bps.
- **Parity (paridad):** Determina el tipo de paridad (en caso de existir) que se le añade a la información. Por defecto no se utiliza paridad en la transmisión.
- **Flow control (control de flujo):** Determina si se llevará a cabo algún tipo de control de flujo de la información. Por defecto está desactivado.
- **Timeout:** Determina cuántos milisegundos han de pasar sin obtener ninguna respuesta en una comunicación para tomarla como fallida. Por defecto está configurado a 10 segundos.

Estos parámetros de comunicación que se especifican, junto al número de dirección del primer esclavo Modbus, son encapsulados, guardados junto al resto de opciones, y determina, en los procesos de recogida de datos de la red Modbus, cómo han de producirse las lecturas.

5.3.3 Selección de analizadores de redes

El **tiempo** que se tarda en ejecutar las lecturas depende de muchos factores, como el número de campos de datos que se reciben, o la cantidad de analizadores de redes con los que hay que comunicarse. Sin embargo, el factor que más afecta al tiempo de lectura es el **intento de comunicación** con un analizador que no se encuentra presente en la red. Por esa razón, la ventana de configuración brinda una serie de botones con los que **seleccionar** cuántos y cuáles analizadores de redes se encuentran presentes. La selección de cada uno de estos analizadores implica, como dato de comprobación, que se **ilumine** el LED correspondiente al analizador seleccionado, de modo que se sepa cuáles están seleccionados y cuáles no.

Esta configuración, junto a la anteriormente mencionada, se encapsula y se guarda, y en los procesos de recogida de datos **determina** con qué analizadores se intentará realizar una conexión para lectura de datos y cuáles se **omitirán**.

Como una aclaración más detallada, se puede explicar que el conjunto de datos de la configuración elegida, se engloban como un **clúster** de información que pasarán a ser una variable compartida y, a su vez, serán escritos en un fichero con formato XML, desde el mismo momento en que el usuario presione el botón **Aceptar**. Tal acción, realiza las citadas labores de guardado y cierra la interfaz de Configuración, devolviendo al usuario a la interfaz principal.

Una vez de vuelta a la interfaz principal, la aplicación leerá la variable compartida referente a la configuración de modo que los procesos internos sepan de qué manera llevar a cabo las comunicaciones.

5.3.4 Protección con contraseña

Acceder a la interfaz de Configuración del software no es una labor que convenga permitir a cualquier usuario, ya que éste podría no entender el funcionamiento del software o los campos de configuración que se brindan. Una mala configuración podría llevar a un funcionamiento errático o inesperado, pero sobre todo **incorrecto**, por parte del programa.

Es muy importante que esta ventana sea solamente accedida por parte de personal **cualificado**, de modo que se eviten posibles imprudencias que den lugar al incorrecto funcionamiento de la aplicación. Por este motivo, el acceso a la interfaz de Configuración se encuentra protegido mediante la necesidad de introducción de una **contraseña**.

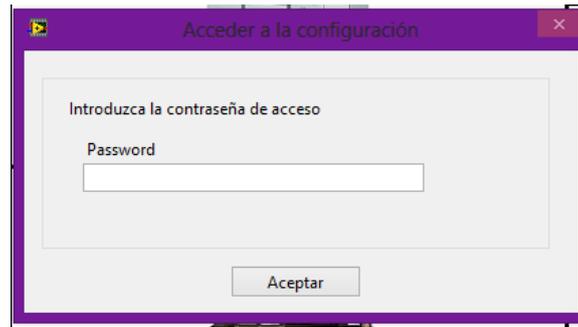


Figura 5. 3: Contraseña de acceso a Configuración.

La acción del usuario, desde la interfaz principal, de presionar sobre el botón de Configuración, así como la ejecución del software sin la existencia previa de archivo de datos de guardado, no traen a la interfaz de Configuración de por sí, sino que llaman a un **proceso intermedio** que muestra una ventana de introducción de **contraseña** como la que se puede observar en la anterior figura.

En presencia de dicha ventana, la ejecución del programa se encuentra paralizada a la espera de la contraseña. Se encuentran previstas dos posibles situaciones:

- **El usuario introduce la contraseña correctamente:** En ese caso, dicha ventana desaparece dando lugar a la aparición de la ventana de Configuración, a la cual se pretendía acceder.
- **El usuario introduce la contraseña incorrectamente:** En este supuesto, la ventana de introducción de contraseña reaparece para **reintentar** la introducción de contraseña. En caso de **tres intentos** fallidos consecutivos, la aplicación devolverá al usuario a la interfaz principal.

5.4 Cuadro General

Como se ha explicado en capítulos anteriores, el Sistema de Gestión Energética está provisto de una serie de analizadores de redes. Cada módulo del sistema cuenta con uno de los equipos Siemens PAC3100, mientras que el **cuadro principal** de la instalación eléctrica del lugar en el que se quiera implantar el sistema cuenta con un Siemens PAC4200, un equipo más sofisticado.

La información que se recoge en el cuadro principal de la instalación gracias al equipo PAC4200 es transmitida desde su interfaz RS-485 al ordenador principal, recogida por el software diseñado e **interpretada**.

La interfaz de Cuadro General del software para el Sistema de Gestión Energética, a la cual se accede oprimiendo el botón Cuadro General desde la interfaz principal del sistema, es la encargada de **mostrar** el resultado de la interpretación de los datos recogidos con anterioridad. Como puede apreciarse, se encuentra dividida en varias pestañas, que se detallan a continuación.



Figura 5. 4: Cuadro General, pestaña Resumen.

5.4.1 Resumen

La pestaña que se muestra por defecto al llamar a la interfaz o módulo de Cuadro General, muestra la información más básica y, a su vez, resumida. De este modo, se muestran los siguientes campos:

- **Tensiones** de cada una de las líneas, dadas en voltios (V). También se dan las tensiones existentes entre líneas.
- **Intensidad de corriente** de las tres líneas y de la línea neutro, dadas en amperios (A).
- **Potencias**, dadas respectivamente en voltio-amperios (VA), vatios (W) y voltio-amperios reactivos (VAr).
- **Energías**, dadas en kilowatios hora (KWh) y kilovoltio-amperios reactivos hora (KVARh) respectivamente.
- **Factor de potencia** de la red eléctrica.
- **Frecuencia** de la red eléctrica, expresada en Hertzios (Hz).

5.4.2 Potencia

En esta pestaña se detallan las potencias instantáneas y máximas existentes en cada una de las líneas y del total de las mismas, así como la naturaleza de cada potencia.

Figura 5. 5: Cuadro General, pestaña Potencia.

Como se puede observar, los distintos campos que se muestran en esta pestaña son los que a continuación se citan:

- **Potencia instantánea:** Muestra cada una de las potencias de los distintos tipos (Aparente, Activa y Reactiva) y por cada una de las fases, dadas en voltio-amperios (VA), vatios (W) y voltio-amperios reactivos (VAr) respectivamente.
- **Potencia máxima:** Recoge el máximo valor alcanzado por cada uno de los distintos tipos de potencia que existen en cada una de las fases. Cada uno de estos valores tomados es permanente, y solo es reemplazado por un valor mayor.
- **Potencia instantánea Total:** Muestra la potencia de cada naturaleza (a saber Aparente, Activa y Reactiva) del conjunto total de las tres fases de forma instantánea.
- **Potencia máxima Total:** Muestra el mayor valor de las distintas potencias en el conjunto total de las tres fases.

5.4.3 Tensión e Intensidad de corriente

Aunque ya se pudieron comprobar a grandes rasgos en la pestaña Resumen, en esta pestaña se aprecian con mayor detalle las distintas tensiones e intensidades de corriente del cuadro general eléctrico.

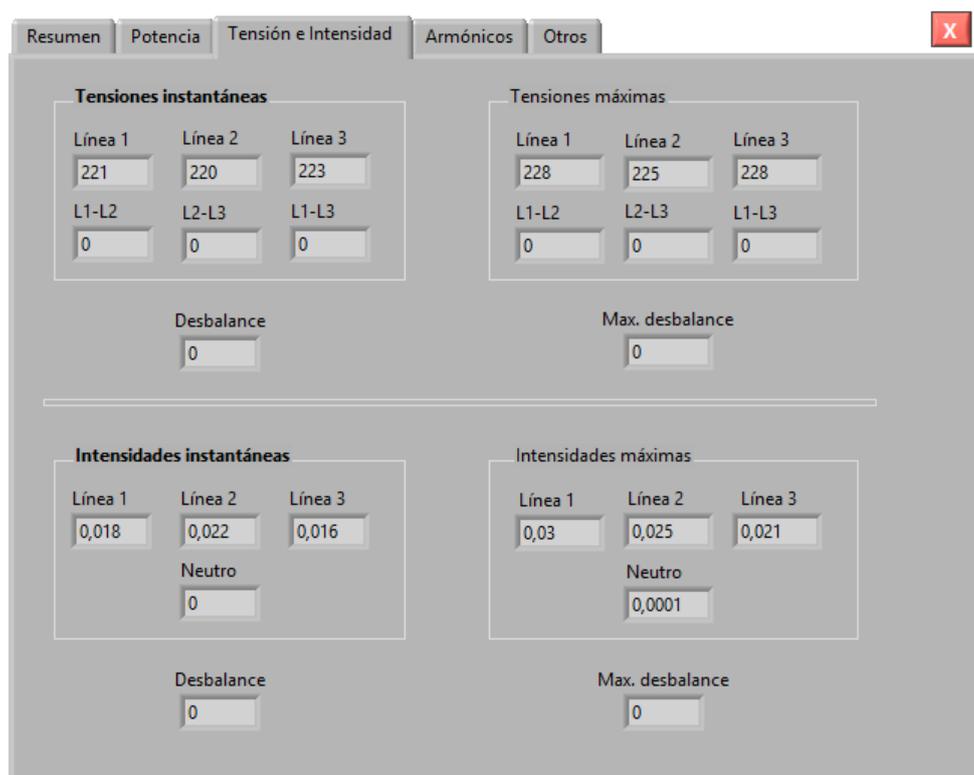


Figura 5. 6: Cuadro General, pestaña de Tensión e Intensidad.

Como se puede apreciar en la figura previa, la pestaña ofrece indicadores numéricos con la información siguiente:

- **Tensiones instantáneas:** Del mismo modo que en la pestaña Resumen se podía apreciar, tanto la tensión instantánea entre cada una de las fases y el neutro como la tensión entre las distintas fases se muestra, en voltios, en esta pestaña.
- **Tensiones máximas:** Estos indicadores almacenan el mayor valor de tensión que se produjese por cada una de las fases y entre fases.
- **Desbalance de tensiones:** Tanto en su versión instantánea como máxima, este indicador muestra el porcentaje actual o máximo respectivamente de los desbalances entre tensiones de las distintas fases.
- **Intensidades instantáneas:** Muestra las intensidades de la corriente que recorre cada una de las fases y el neutro, en amperios.
- **Intensidades máximas:** Muestra el máximo valor de intensidad de corriente de cada una de las líneas eléctricas del cuadro general.
- **Desbalance de intensidades:** Al igual que su homólogo en tensiones, indica el porcentaje de desbalance de intensidades tanto instantáneas como en valor máximo.

5.4.4 Armónicos

Esta pestaña muestra de una forma gráfica con **barras** el valor de los **armónicos** que se producen en la línea. Este es un valor **exclusivo** del analizador PAC4200. Por cada una de las líneas, muestra los armónicos impares del 3º al 31º clasificándose, según pestañas, en armónicos de tensión en cada fase, intensidad de corriente en cada fase y tensión entre fases.

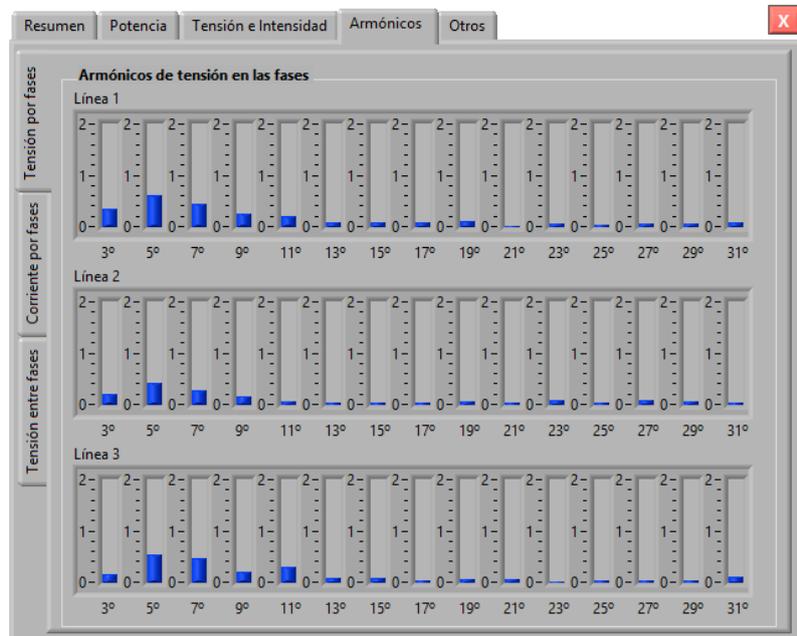


Figura 5. 7: Cuadro General, pestaña de Armónicos.

5.4.5 Otros datos

La última pestaña muestra al usuario una serie de valores de distinta naturaleza que, en algún momento, pueden resultar de gran utilidad. Los datos que provee esta pestaña son los siguientes:

- **Frecuencia:** Se muestra la frecuencia actual y su máximo y mínimo.
- **Factor de potencia:** De forma instantánea, máxima y mínima, tanto por fases como total, muestra los valores del factor de potencia, acompañado de un LED que avisa si tal factor es menor a 0,95.
- **THD (Distorsión Total por Armónicos):** Determina la distorsión total que se produce en la señal eléctrica por cada fase tanto por armónicos de tensión, como de intensidad y de tensión entre fases.

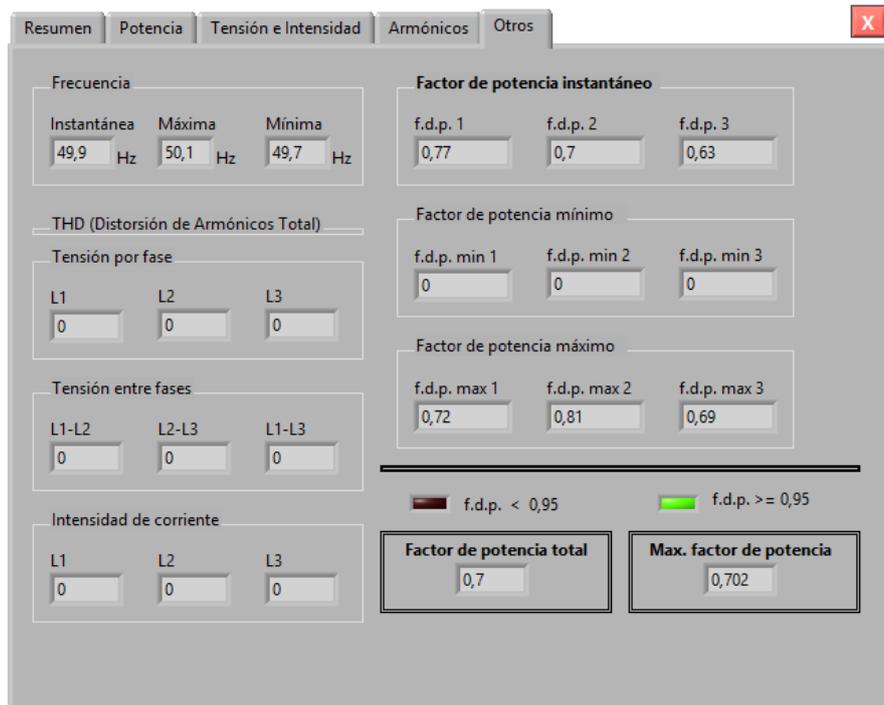


Figura 5. 8: Cuadro General, pestaña de Otros datos.

5.4.5 Botones Borrar Máximos y Volver

El botón de **Borrar Máximos** tiene como función anular los valores máximos almacenados, de modo que el sistema **olvide** todos los valores máximos de cada dato y los reemplace con el mayor valor obtenido tras el borrado. Por su parte, tanto **Volver** como el clásico botón de **cierre** harán desaparecer esta ventana y **regresar** a la interfaz principal.

5.5 Iluminación

Cuando se hace *click* sobre el botón de Iluminación se llama a este módulo y se muestra por pantalla la interfaz.

Como se puede observar en la siguiente figura, la interfaz está dividida en una serie de pestañas que se encargan de representar distintos datos. Consta, además, de un botón de **cierre** y otro de borrado de valores **máximos** como los que se han explicado anteriormente. Por último, el botón de **Introducir datos** se encarga de llevar al usuario a la interfaz de **gestión** de la información de la iluminación.



Figura 5. 9: Iluminación, pestaña Resumen.

5.5.1 Resumen

La pestaña Resumen muestra una serie de datos que resumen el consumo que se realiza de potencia en iluminación. Consta de los siguientes campos:

- **Mostrar por:** Cuando un usuario introduce un nuevo recurso de iluminación, lo hace especificando un nombre, una zona de referencia y la fase eléctrica en la que se encuentra dicho recurso. En la pestaña Resumen, este campo permite seleccionar cómo resumir el consumo de los recursos, mostrándolos por nombre, por zona o por fase.
- **Seleccionar:** Dependiente del anterior campo, mostrará los distintos nombres, las diferentes zonas predeterminadas o las tres fases. De este modo, la pestaña Resumen mostrará la información de resumen del consumo de tal recurso, zona o fase.
- **Potencia total de iluminación:** Es la potencia que consume el conjunto de los elementos de iluminación para un determinado recurso, zona o fase, como el total del siguiente cálculo:

$$n^{\circ}lámparas(P. lámpara + P. equipo aux.) \cdot n^{\circ}luminarias$$

- **Número de lámparas:** Es el número de lámparas de las que consta cada una de las luminarias que cuelgan del recurso, zona o fase especificada.

- **Número de luminarias:** Indica el número de luminarias presentes en cada recurso, zona o fase.
- **Potencia disipada en el cable:** Especifica la potencia que se consume por el simple hecho de ser recorridos por la corriente los conductores eléctricos.
- **VEEI máximo superado:** Indica si, para el elemento seleccionado, el VEEI máximo especificado ha sido rebasado.
- **Tensión:** Es la tensión que cae en el conductor desde el cuadro eléctrico hasta la primera luminaria.
- **Consumo total:** Es la suma entre la potencia de iluminación total y la potencia que se disipa en los conductores. En función de la cantidad de potencia, el indicador presenta un color distinto, siendo verde cuando el consumo es reducido, amarillo cuando es medio y rojo cuando es elevado. Junto a la barra que indica color y cantidad se encuentra un indicador numérico para que resulte más fácil determinar el consumo total realizado.

5.5.2 Tensiones

Esta pestaña representa la información proveniente de la recogida de información del analizador de redes PAC3100 que se halla conectado al cuadro eléctrico dedicado a la iluminación. Como se puede comprobar en la siguiente figura, estos datos se presentan de una forma ordenada y muy visual, con los indicadores numéricos acompañados de una gráfica de evolución temporal que determina cómo va variando la tensión a cada momento.

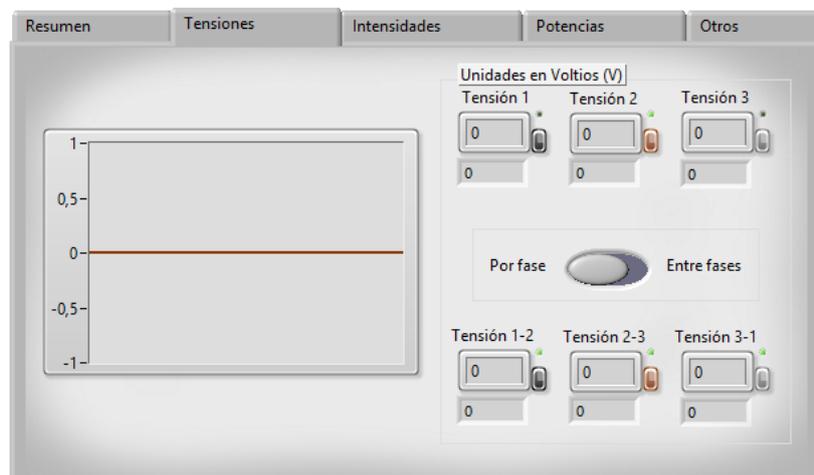


Figura 5. 10: Iluminación, pestaña Tensiones.

Los campos que se muestran en la pestaña actual se son:

- **Tensión (1, 2 y 3):** Muestran la tensión instantánea de cada fase. Aparecen acompañados de un segundo indicador que almacena el máximo valor de cada uno de estos, además de un pequeño interruptor del color de cada fase para activar o desactivar la representación de cada valor en la gráfica.

- **Tensión entre fases:** Indica la tensión entre fases. Del mismo modo, en otro indicador almacena el valor mayor de tensión entre fases, y consta también de un interruptor para activar o desactivar cada valor en la representación gráfica.
- **Interruptor de selección de gráfica:** Este interruptor seleccionará cuál ha de ser la gráfica que se muestre en la zona izquierda de la interfaz. De este modo, el usuario elegirá mostrar los datos de tensión de cada fase respecto al neutro, o en su lugar mostrar los datos de tensiones entre fases.

Cabe destacar que los datos de tensión que se muestran vienen dados en voltios.

5.5.3 Intensidades

La siguiente pestaña que encuentra el usuario sirve para mostrar la información de la intensidad de corriente que atraviesa las líneas del cuadro eléctrico de iluminación.

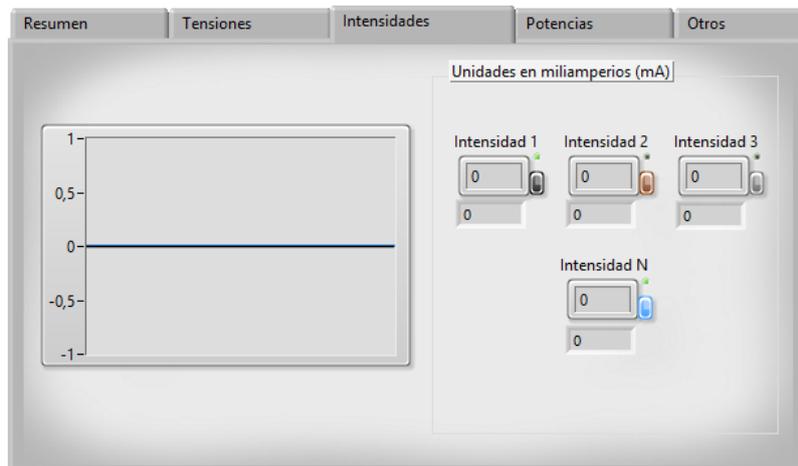


Figura 5. 11: Iluminación, pestaña Intensidades.

De un modo también muy visual, se encuentran los indicadores referentes a cada una de las líneas, junto con una representación gráfica que evoluciona temporalmente. Los indicadores son:

- **Intensidad (1, 2 y 3):** Muestra los datos de intensidad de cada fase. Cada indicador incluye un segundo indicador que muestra el máximo valor y un botón para activar o desactivar la representación gráfica.
- **Intensidad por Neutro:** Muestra los datos de la corriente que circula por el neutro. Al igual que los anteriores campos, posee su propio interruptor para activar o desactivar representación gráfica, así como un indicador de valor máximo.

5.5.4 Potencias

La siguiente pestaña es la que hace referencia a la potencia total del cuadro eléctrico, en sus distintas fases y distintos tipos.

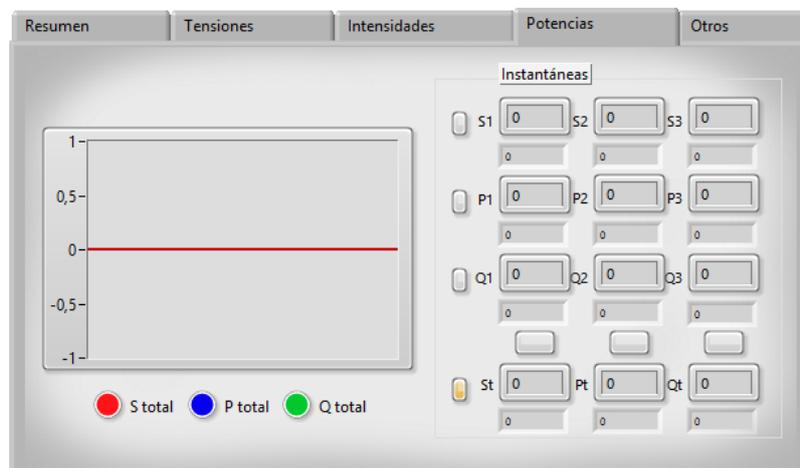


Figura 5. 12: Iluminación, pestaña Potencias.

Los indicadores que se presentan en esta interfaz, mostrados mediante su nombre abreviado, son los siguientes:

- **Potencia aparente (S):** Muestra la potencia aparente medida en el analizador de redes, para cada una de las fases del cuadro eléctrico. Cada uno de estos indicadores de potencia por fase incluye un segundo indicador con el valor máximo alcanzado para cada uno.
- **Potencia activa (P):** Muestra la potencia activa que el analizador de redes mide en el cuadro eléctrico de iluminación para cada fase, en cada indicador correspondiente. Cada uno de estos indicadores se acompaña de un indicador secundario para mostrar el valor máximo por cada uno.
- **Potencia reactiva (Q):** Indica la potencia reactiva medida en el analizador de redes para cada fase del circuito, incluyendo por cada valor un segundo indicador que muestre el máximo valor recogido de potencia reactiva en la fase.
- **Potencia aparente total (St):** Muestra el total de potencia aparente del cuadro eléctrico, acompañado por un segundo indicador que representa el valor máximo de tal potencia.
- **Potencia activa total (Pt):** Indica el total de potencia activa que es medida en el analizador de redes. A su vez, muestra el valor máximo de la misma.
- **Potencia reactiva total (Qt):** Indica la potencia reactiva total que el medidor de redes determina, junto con un segundo indicador para el valor máximo.

En la parte izquierda de la interfaz se encuentra un marco de representación gráfica, acompañado de una leyenda de colores inferior. Los datos que se muestran tanto en la gráfica como en la leyenda pueden variar según el interruptor que se seleccione:

- El interruptor junto a las potencias aparentes, mostrará por pantalla las potencias aparentes de cada una de las fases, cada una en un color distinto.
- El interruptor junto a las potencias activas representará en la gráfica el valor temporal de la potencia activa por cada fase, cada uno en un color distinto.
- El interruptor junto a las potencias reactivas mostrará en la gráfica temporal la potencia reactiva para cada una de las fases, en distinto color.
- El interruptor situado bajo las potencias de la fase 1 hará aparecer en pantalla las distintas potencias (Aparente, Activa y Reactiva) para la fase especificada.
- El interruptor situado bajo las potencias de la fase 2, representará en pantalla los tres tipos de potencia para la fase 2 en diferentes colores.
- El interruptor bajo las potencias de la fase 3, hará que se muestren en pantalla las distintas potencias para la fase 3 en distinto color.
- El interruptor junto a las potencias totales hará que en pantalla se muestren las potencias totales Aparente, Activa y Reactiva en distinto color.

Es importante destacar que la activación de cada uno de estos botones implicará la desactivación automática de los demás, de modo que solo existan en pantalla tres datos.

5.5.5 Otros datos

La última pestaña que se muestra en la siguiente figura recoge los siguientes cuatro parámetros que se detallan a continuación:

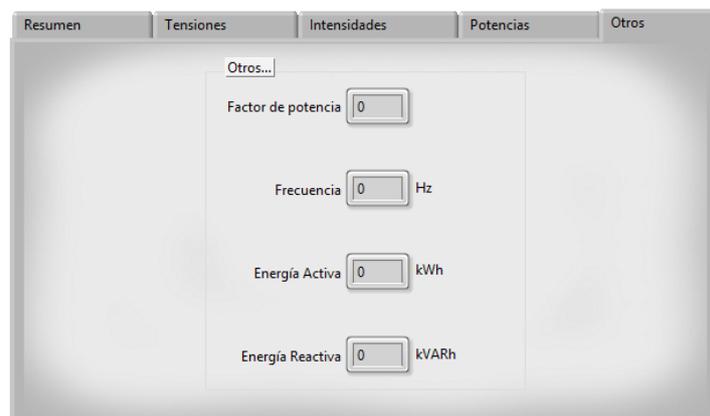


Figura 5. 13: Iluminación, pestaña Otros.

- **Factor de potencia:** Muestra el factor de potencia total medido por el analizador de redes en el cuadro de iluminación.
- **Frecuencia:** Indica la frecuencia de la red eléctrica.

- **Energía Activa:** Muestra el contador de energía activa consumida.
- **Energía Reactiva:** Muestra la energía reactiva de iluminación.

5.5.6 Introducción de datos

Al accionar este botón aparece una nueva interfaz para gestionar la información de Iluminación. Ésta consta de dos pestañas, es la que permite que el usuario pueda añadir, de una manera ordenada y detallada, información para cada habitáculo, del recinto en el que se quiere aplicar el Sistema de Gestión Energética, en el que exista tipo alguno de iluminación.

Común a las dos pestañas de esta nueva interfaz, existe un botón clásico de cierre, que hará regresar al usuario a la interfaz de Iluminación sin realizar cambio alguno, y otro interruptor rotulado Guardar y Salir, que sirve para guardar la información introducida junto a la ya existente, tanto en una variable compartida del programa como en un fichero en formato XML, de modo que no solo durante la actual ejecución del software sino en futuras ejecuciones, la aplicación recuerde los datos introducidos. Este último interruptor, además del guardado, cierra la ventana y devuelve al usuario a la misma interfaz de Iluminación que el botón de cierre.

5.5.6.1 Iluminación interior

Se presenta la siguiente interfaz con una serie de campos y de controles de tipo lista y numéricos para la detallada introducción de valores.

The screenshot shows a software window titled 'Iluminación interior' with a sub-tab 'Datos introducidos'. The interface is organized into several sections:

- Normativa:** Includes an 'Importar' button and a 'Uso:' dropdown menu set to '2.7.8 Inspección de colores'.
- Technical Parameters:** Fields for 'Em' (100), 'UGr' (16 lux), 'Ra' (90), 'Max. VEEI' (4), and 'Observaciones' (Tcp > 4000 K).
- Datos:** Fields for 'Fase' (1), 'Zona' (Nave), 'Nombre' (Lab.3), 'Superficie' (200 m²), 'Conductor (cable) Longitud' (40 m), and 'Sección' (1,5 mm²).
- Luminaria:** A 'Luminarias' spinner set to 12.
- Lámpara:** Fields for 'Lámparas' (4), 'Tipo de iluminación' (Fluorescente compacta), 'Potencia' (18 W), 'P. lámpara' (0 W), and 'P. auxiliar' (0 W). An image of a compact fluorescent lamp is shown.
- Flujo luminoso:** Fields for 'Tensión' (220 V), radio buttons for 'Real' and 'Teórica', 'Flujo' (0), and 'Flujo total' (100).

At the bottom right, there is a green 'Añadir' button. At the very bottom of the window, there are logos for 'TELE NATURA', 'Universidad Politécnica de Cartagena', 'energética ATRIAM', and a 'Guardar y salir' button.

Figura 5. 14: Introducción de datos, Iluminación interior.

A continuación se detallan los distintos campos que conforman la interfaz de introducción de datos:

- **Normativa:** Este campo muestra los datos a los que se ha de ajustar la zona de iluminación que se va a introducir para ser eficiente. El botón Importar actualizará la lista de los usos posibles para cualquier habitáculo según la normativa y, una vez seleccionado, se mostrarán en indicadores las diferentes características.
- **Datos:** Consta de tres descriptores que añadir a cada recurso. En primer lugar, la fase a la que se encuentra conectada la iluminación del habitáculo que se está comenzando a detallar. El segundo descriptor es el tipo de zona que es, en base a unos tipos determinados de habitáculo. El tercer descriptor es un nombre que se le da al recurso para poderlo identificar fácilmente. Por último, se añade otro dato, que es la superficie a iluminar.
- **Conductor:** Son los datos referentes al conductor que lleva la corriente desde el cuadro eléctrico hasta la primera luminaria. Consiste en los campos Longitud y Sección, este **último** estandarizado a los valores más comunes.
- **Luminaria:** Se indicarán el número de luminarias de las que consta el habitáculo.
- **Lámparas:** Se indican los datos referentes a las lámparas. El primer dato es el número de lámparas existentes en cada luminaria. Otro es el tipo de iluminación, del que existe una lista con los tipos más comunes. La potencia nominal de la lámpara se debe especificar. En caso de elegir un tipo de iluminación personalizado, se activarán los campos de potencia de lámpara y potencia del equipo auxiliar (o balastro) para indicarlos al sistema. Además, la elección del tipo de iluminación hará variar la imagen mostrada.
- **Flujo luminoso:** Indicando una tensión real, o seleccionando que la tensión es teórica, calcula y determina el flujo total luminoso de la lámpara.

El botón Añadir producirá que se inserte un nuevo elemento clúster en el array de clústeres que conforman la variable compartida y fichero XML de datos a guardar.

5.5.6.2 Datos introducidos

Los datos que se han introducido previamente en la pestaña anterior, pueden ser en cualquier momento consultados y, en caso de ser necesario, eliminados. Para ello se muestra, apreciable en la siguiente figura, la siguiente interfaz.

Figura 5. 15: Introducción de datos, Datos introducidos.

Los campos que se muestran en la pestaña actual, son de distinto tipo y se explican a continuación:

- **Lista de recursos:** Es la lista con los recursos o habitáculos introducidos, de modo que sea seleccionado uno para su revisión o eliminación.
- **Uso y denominación:** Este campo muestra una serie de datos. Por una parte, el uso dado al habitáculo. Por otra parte, los identificadores explicados previamente como Fase, Zona y Nombre. Además, se indica la superficie de la zona a iluminar.
- **Características:** Detalla las características de la zona de iluminación. Por ello, se indica primeramente el total de potencia consumida por el recurso. Además, se adjuntan datos como la potencia consumida por cada balastro o equipo auxiliar para cada lámpara, la potencia consumida por cada luminaria así como por cada lámpara. Se indica también el tipo de iluminación que se ha empleado.
- **VEEI y Rendimiento:** Se muestran de distinta manera una serie de valores:
 - **VEEI:** El valor de VEEI calculado para los datos de iluminación que se han introducido. Este valor, además de indicador numérico, se acompaña de un indicador gráfico de aguja.
 - **Máximo VEEI:** El valor máximo que la normativa considera, para el uso seleccionado, para que la instalación sea eficiente. En caso de que se sobrepase este máximo, el piloto LED rotulado como Exceso se iluminará con luz roja. En caso contrario se mantendrá como una luz verde.

- **Porcentaje de VEEI:** Es el valor, en tantos por ciento, del VEEI que se presenta frente al máximo permitido.
- **Resistencia del conductor:** Muestra el valor de resistencia que el conductor presenta debido a sus factores de longitud, sección y de la resistividad del cobre.

Por otra parte, encontramos un par de interruptores en esta nueva pestaña. Su uso se explica a continuación:

- **Eliminar:** Se encarga de eliminar el recurso que actualmente se encuentra seleccionado. La eliminación se hará efectiva cuando el usuario haga click sobre el botón general Guardar y Salir.
- **Tabla VEEI:** Este interruptor llama a una nueva ventana que permitirá consultar los datos que relacionan cada uno de los posibles usos de habitáculo con el valor límite del VEEI para cada uno de ellos. Gracias a estos valores es posible determinar cuándo una instalación de iluminación es eficiente o no lo es.

5.6 Procesos transparentes

Hasta ahora, todo lo que se ha explicado hace referencia a procesos que muestran una interfaz gráfica para interactuar con el usuario. No obstante, el software es un conjunto mucho más complejo, que incluye procesos que no se encuentran a la vista de los usuarios y funcionan de manera transparente.

El software implementado para conformar el Sistema de Gestión Energética incluye una serie de labores que suceden automáticamente, y para las cuales no es precisa la existencia de una interfaz.

5.6.1 Lectura y propagación de datos

Los datos que circulan por el programa son fruto de las continuas y rápidas lecturas de la red Modbus que realiza el propio software. Uno por uno, de forma cíclica, los distintos analizadores de redes son consultados de manera que, aquellos que se encuentren presentes y configurados para ser accedidos, éstos vuelcan la información al bus y el ordenador los procesa y los recoge. Cuando el software tiene todos los datos de un analizador de redes, compone un clúster que actualizará la variable compartida correspondiente, de modo que el resto del programa, como por ejemplo los módulos Cuadro General o Iluminación, tengan acceso a los datos que alberga la misma. El ciclo termina una vez leído el último analizador de redes. Posteriormente, tras un inapreciable período de tiempo, el ciclo de lectura vuelve a comenzar.

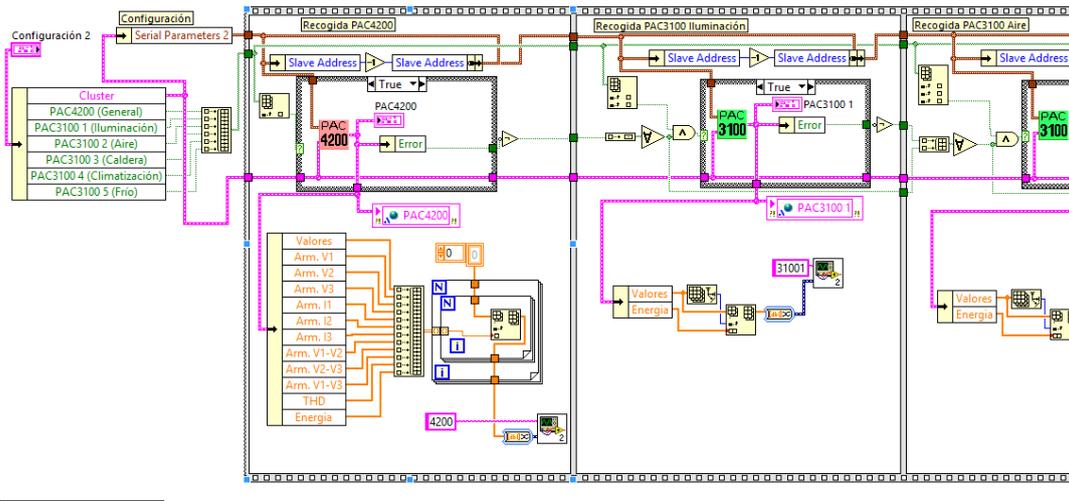


Figura 5. 16: Código de la lectura de analizadores de redes.

5.6.2 Respaldo en servidor

Del mismo modo, otro proceso que se realiza de manera transparente al usuario es el respaldo en el servidor de los datos. Cada día, a las 0.00h se comienza un nuevo archivo de datos. El software comprueba periódicamente la existencia de un nuevo archivo, en cuyo caso, carga en el servidor el archivo histórico del día anterior, previa comprobación de que no se encuentre ya en dicho servidor.

Capítulo 6

Conclusiones y líneas futuras

6.1 Análisis

El hecho de llevar a cabo un Proyecto Final de Carrera de tal envergadura, basado en la creación de un Sistema de Gestión Energética para la mejora de la **eficiencia** en cualquier escenario empresarial, industrial o particular (aunque principalmente ideado para la industria **agroalimentaria**) tiene como objetivo la obtención de resultados concretos, que reflejen la viabilidad y la **conveniencia** tanto del desarrollo del proyecto como de la subsiguiente implantación del sistema desarrollado en los escenarios indicados.

6.1.1 Funcionamiento de la red Modbus

La instalación del sistema que se ha diseñado en una industria o gran empresa resulta más **laboriosa**, a la hora de ser llevada a cabo, en comparación con la facilidad de montaje propia del periodo de pruebas y desarrollo, ya que en el momento de la programación y comprobación los dispositivos para el análisis de redes y la red eléctrica a analizar se encontraban a muy poca **distancia** del propio ordenador de programación y entre sí.

Llevar a cabo el montaje del sistema en una industria implica cubrir distancias relativamente **largas**, ya que tanto el lugar donde se quede instalado y protegido el ordenador central como la ubicación de cada cuadro eléctrico pueden distar mucho, en función de las **dimensiones** del lugar.

Tanto los analizadores de redes como el convertidor de USB a RS-485 del que consta el ordenador cuentan con una serie de diodos LED que parpadean cuando emiten y reciben datos. Ello supone que la simple ejecución del programa, aun sin desplegar ninguno de los módulos, y la correcta configuración del mismo, debe producir el parpadeo de los LED de los equipos.

El análisis concluyó satisfactoriamente: la activación de la lectura de ambos equipos, configurados adecuadamente los parámetros de la conexión, provocó el parpadeo previsto de los indicadores LED, así como el normal funcionamiento del programa, que en ningún momento desplegó mensajes de error de lectura. El posterior funcionamiento correcto de los módulos Cuadro General e Iluminación terminó de corroborar el éxito del análisis.

6.1.2 Módulo de Cuadro General

La comprobación del correcto funcionamiento de la red de comunicaciones Modbus sobre RS-485 implica que el tránsito de datos entre el **ordenador** y los **analizadores** se realiza correctamente y sin problemas debidos a las distancias.

La más importante **implicación** del funcionamiento de la red de comunicación es que, al recibir correctamente los datos del programa, el **módulo** de Cuadro General **funciona**.

Como se explicó con anterioridad, este módulo se encarga de **interpretar** los datos que el analizador de redes conectado al cuadro eléctrico principal mide y emite. La manera de analizar el **correcto** funcionamiento del módulo Cuadro General consiste en la **ejecución** de dicho módulo y **contraste** de datos.

Como el módulo en cuestión representa datos sin procesar, que fácilmente pueden ser consultados en el correspondiente analizador de redes, el análisis se llevó a cabo mediante la **comprobación**, dato a dato, de los parámetros medidos por el **analizador** de redes y su representación en el **software**. Cada dato debía de **coincidir** con su homólogo en el software.

El resultado del análisis concluyó **satisfactoriamente**: todos los campos del software **coincidieron** con los mostrados en el analizador de redes, si bien los datos emitidos a través del bus y posteriormente mostrados en la aplicación presentan mayor **complejidad** a nivel de **decimales** que aquellos mostrados en el display del analizador.

6.1.3 Módulo de Iluminación

En cuanto al módulo de Iluminación, son dos los tipos de análisis a realizar. Por una parte, los análisis de datos no-procesados, aquellos que son un reflejo del resultado de la lectura del **analizador** de redes conectado al cuadro eléctrico de iluminación. Por otro lado, la comprobación de que los datos **introducidos** al sistema y los **cálculos** derivados de los mismos son correctos de una forma bastante aproximada.

La ejecución el módulo de Iluminación y comprobación los **datos** desde la pestaña **Tensiones** hasta la pestaña **Otros**, y sus **homólogos** en el analizador de redes del cuadro de iluminación, corroboró la **correspondencia** de aquellos datos que no son procesados.

Por otra parte, la introducción manual de datos al sistema y eliminación de los mismos sirvió para comprobar que dichos datos se **insertaban** y **eliminaban** sin problema, además de servir como reflejo de que **variando** determinados factores, como el número de luminarias en combinación con el tipo de lámparas, tiene como resultado una **mejora** de la eficiencia energética, como también se reflejó correctamente en la pestaña Resumen.

Finalmente, de la introducción de datos se desprenden ciertos **cálculos**, como la potencia por cada luminaria, o la potencia total de iluminación de un habitáculo, cuya **veracidad** fácilmente fue **corroborada** con ayuda de una calculadora.

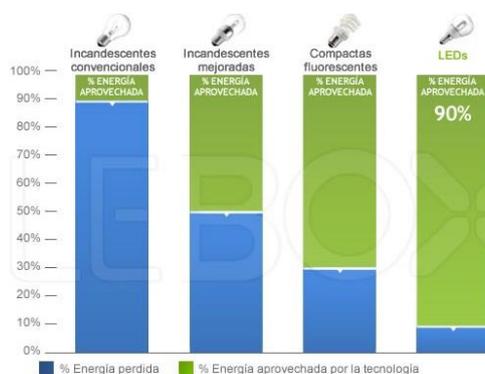


Figura 6. 1: El tipo de bombilla es un factor determinante en el aprovechamiento energético.

6.2 Conclusiones

Tras los resultados que se han obtenido gracias al análisis del sistema, se desprenden las siguientes **conclusiones**:

1. El sistema de comunicaciones resulta **eficaz**, presentando la debida **resistencia** a las condiciones adversas gracias a su **sencillez**, y permitiendo además unas **distancias** relativamente grandes para la transmisión a alta velocidad de datos.
2. La replicación de la información en el **servidor** resulta de gran utilidad para el **respaldo** de los datos históricos, con lo que una hipotética alteración del funcionamiento del sistema, como podría ser un apagón eléctrico, no supone la **pérdida** de todos los datos obtenidos, sino únicamente de aquellos más recientes.
3. El **correcto** funcionamiento del módulo Cuadro General supone la correcta monitorización del consumo energético **global**, lo que permite llevar a cabo **modificaciones** en la instalación para hacerla más **eficiente**.
4. El módulo de Iluminación, al proveer al usuario de los distintos **datos** de consumo energético en dicho concepto, permite que se monitoricen de forma correcta el **aprovechamiento** de la energía, qué porcentaje del consumo total energético recae en la **iluminación** y cuál en **otros** cuadros eléctricos. De la misma manera, permite medir con gran detalle la eficiencia energética de cada **habitáculo**, con lo que facilita el estudio de posibles **cambios** para conseguir una instalación más **eficiente**.
5. El simple detalle del tipo de **lámpara** que se utilice para la iluminación de una zona puede ser determinante para conseguir un mayor **aprovechamiento** de la energía.
6. En el tiempo **actual** en el que las posibilidades de mejora de la productividad ya han sido desarrolladas, así como en la actual coyuntura **económica**, el paso a dar consiste en la mejora del **ahorro** económico, y un fácil camino para la consecución del mismo es la mejora de la **eficiencia energética**. Esta, a su vez, se traduce en una mayor **sostenibilidad**.
7. Resulta **conveniente** llevar a cabo la implantación del sistema en cualquier escenario de tipo **industrial** o a nivel de **empresa** mediana o grande, ya que el coste asociado a los equipos necesarios es traducido en un gran conjunto de **posibilidades** para la mejora de la **eficiencia** energética que se desprenden de la monitorización de datos gracias al software. Esta mejora, a su vez, se traduce en un mayor **ahorro** económico.

6.3 Líneas futuras de desarrollo

Ha quedado patente en esta memoria que el Sistema de Gestión Energética desarrollado es satisfactorio, ya que cumple con las **expectativas** para las que fue llevado a

cabo. No obstante, se trata de una primera versión con lo que resulta posible y recomendable que siga siendo **desarrollado**.

Aunque a priori algunas posibilidades de **mejora** del programa resulten evidentes, a continuación se detallan las futuras **líneas de desarrollo** con las que se pretende que este sistema cumpla en un **futuro**:

- Como se explicó con anterioridad, aunque los módulos de Iluminación y del Cuadro General han sido **finalizados** con éxito, estos son, aunque de manera fundamental, solamente una **parte** de un Sistema de Gestión Energética más ambicioso. Por ello, una de las líneas futuras de desarrollo consistiría en el diseño e implementación del resto de **módulos** sobre cuyo consumo mejorar la eficiencia energética. De esta manera, el gran ahorro que se produce en la Iluminación, se produciría de igual forma en consumo de **aire comprimido**, **frío industrial y de cámara**, **calderas de vapor** y **aire acondicionado**.



Figura 6. 2: Futuros módulos a implementar.

- Por otra parte, aunque se ha buscado que la aplicación contara con unas interfaces limpias y vistosas, durante el desarrollo ha primado la **funcionalidad** antes que la apariencia. Es por ello que otra vía de desarrollo consistiría en la mejora de las interfaces, de modo que entre sí cada ventana del software produzcan sensación de **homogeneidad**, y a su vez una impresión de **sencillez**, de cara al usuario final.
- En determinados escenarios en los que las condiciones lo permitan, podría resultar interesante el estudio y posterior implementación de un sistema de comunicaciones basado en **Modbus TCP**, sustituyendo la actual red serie basada en RS-485 por una red **Ethernet**, y mejorando las prestaciones como la **velocidad** de comunicación.
- Otra vía de desarrollo futuro que resulta interesante consiste en el desarrollo e implementación de las versiones **móviles** y **web** del software para terminar de convertirlo en una aplicación **multiplataforma**.
- Una última línea de desarrollo que puede resultar también evidente: una vez que se ha entrado de lleno en la utilización de un sistema **SCADA** para llevar a cabo este proyecto, resultaría interesante no limitar el sistema a la simple **monitorización**. En su lugar, convendría avanzar al siguiente nivel, el

referido a **control**. De este modo se incrementaría el ahorro energético, ya que en lugar de **ofrecer** posibilidades de mejora al usuario para que tome decisiones, éstas podrían bien ser tomadas **automáticamente** por el **sistema** mediante la autoregulación, como llevadas a cabo **remotamente** por el usuario desde cualquier ordenador o teléfono inteligente.

Apéndice 1

LabVIEW

1.1 Introducción

A la hora de desarrollar soluciones de software, nos encontramos con una gran variedad de opciones. Por una parte, existen distintos niveles de programación, en función de la sencillez con la que se quiera trabajar o de la eficiencia que se quiera conseguir. De estos, lo más común es trabajar en lenguajes de alto nivel, ya que estos son más versátiles, en contrapunto con los lenguajes de bajo nivel, específicos del procesador.

Del mismo modo, dentro de la programación de alto nivel se nos brindan una serie de lenguajes distintos, como pudieran ser JAVA, C o sus variantes orientadas a objetos como C++ o C#, etc. Estos lenguajes citados son basados en texto, es decir, la programación se realiza mediante sentencias de código escrito que determinan el comportamiento que nuestro programa debe presentar.

Es en este punto cuando se presenta a LabVIEW, entorno de desarrollo de software mediante programación gráfica en lenguaje G, que permite el desarrollo más ágil de nuestro software al resultar más visual e intuitivo.

El nombre de LabVIEW es un acrónimo de Laboratory Virtual Instrument Engineering Workbench, o lo que es lo mismo, un entorno de programación de instrumentos virtuales (comúnmente denominados VIs) recomendado para programación de hardware y software de pruebas, control y diseño, simulado o real y embebido. Ofrece una amplia cantidad de soluciones para desarrollo de software no solo para ordenadores tradicionales, sino también para dispositivos móviles, computadores industriales u otro tipo de microcontroladores de uso específico.

1.2 Instrumentos Virtuales (VIs)

El método de trabajo con LabVIEW se basa en la creación de los llamados Instrumentos Virtuales, que cumplen una función concreta para la que los hayamos creado.

Estos VIs pueden a su vez ser parte de otro VI más grande, operar entre ellos o, por el contrario, estar formados por otros VIs internos (llamados subVIs). Dado que LabVIEW está orientado a la programación desde un punto de vista gráfico, un VI siempre constará de dos partes fundamentales:

- **Panel frontal**, o lo que es lo mismo, la interfaz que ese programa o subprograma va a presentar al usuario para que éste pueda interactuar con él.

- **Diagrama de bloques**, que se corresponde con la programación interna que determina el funcionamiento del VI o subVI.

Un instrumento virtual siempre constará de estas dos partes, no obstante, en función del uso más general o específico que se le quiera dar, es posible que no se muestre su panel frontal ya que no precise de interacción con el usuario, sino con otros procesos.

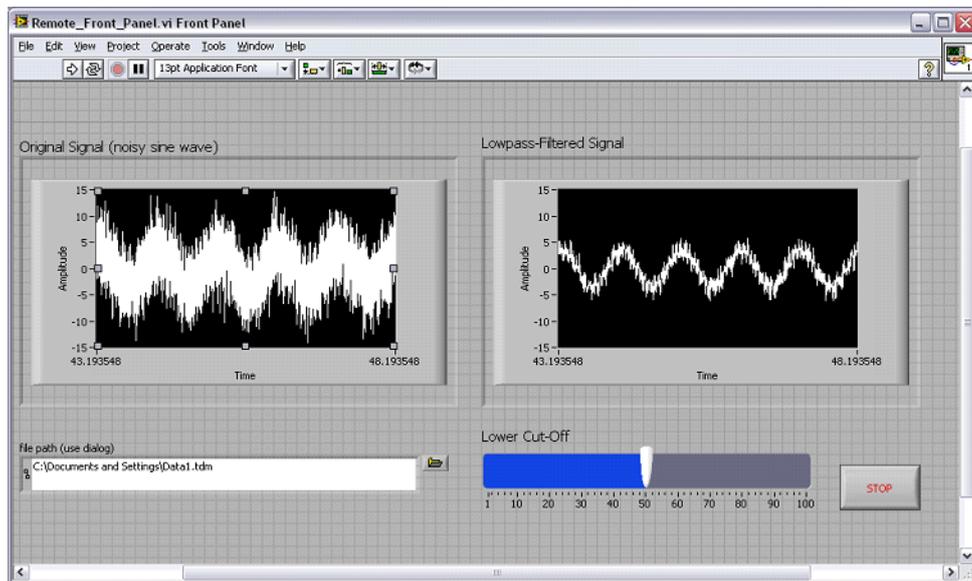


Figura 7. 1: VI de un filtro paso bajo

1.3 Panel Frontal y Diagrama de Bloques

LabVIEW nos presenta una interfaz con dos pantallas distintas pero relacionadas entre sí, con las que poder llevar a cabo el desarrollo de nuestro software. Estas son, por un lado, el Panel Frontal, y por otro, el Diagrama de Bloques. Por cada unidad de software o VI se nos presentarán estas dos ventanas.

Panel frontal

Como se ha explicado previamente para el caso de un VI, el Panel Frontal es el entorno en el que se diseñará la interfaz que el instrumento virtual va a presentar. Éste pone a disposición del usuario todos los elementos clásicos para una interfaz de usuario. En este aspecto, podemos encontrar, fundamentalmente:

- **Indicadores:** Muestran por pantalla datos resultantes de las operaciones que se hayan programado previamente en el diagrama de bloques, o que procedan de otros subVIs. Podemos encontrarlos de distinto tipo, como numéricos, de array, gráficos a modo de barra de progreso o de tipo termómetro o como campos de texto, por ejemplo.
- **Controles:** Son los que permiten al usuario interactuar con la interfaz y modificar el comportamiento del programa. Aparecen de distinta forma en función del tipo al que correspondan. Pueden ser campos para tipo numérico

o texto. Por otra parte también pueden ser controles deslizantes. Para el caso de los tipos booleanos, podemos encontrar distinto tipo de botones.

- **Etiquetas:** Son rótulos de texto que podemos colocar en cualquier parte del programa para diferenciar campos, especificar unidades de magnitud, o en general, aportar información.
- **Decoraciones:** Tales como marcos, controles de pestañas, o zonas elevadas o hundidas para agrupar conjuntos de controles y/o indicadores.

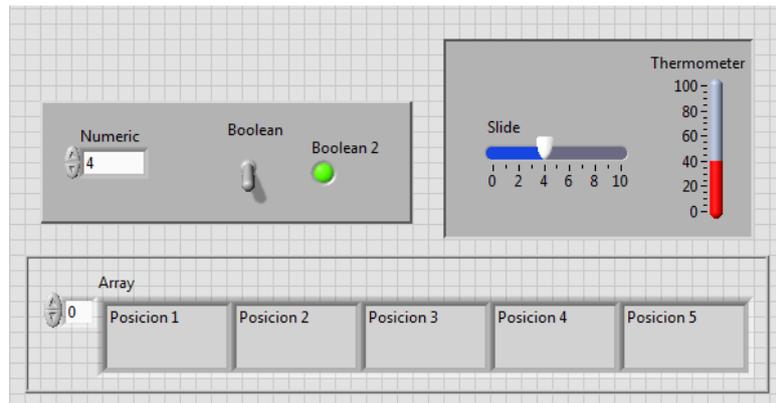


Figura 7. 2: Ejemplo de un panel frontal

En la figura 7.2 se puede distinguir un ejemplo de los distintos tipos de controles (numéricos de campo y deslizante, booleano de tipo interruptor), indicadores (de array de campos de texto, numérico de tipo termómetro, booleano de tipo LED) y de decoraciones (marco, caja elevada y caja hundida).

Diagrama de bloques

Tras la interfaz brindada por el panel frontal, se encuentra toda la programación del software que se relaciona con dicha interfaz y, del mismo modo, interrelaciona los distintos controles e indicadores mostrados por el panel frontal con sus indicadores, así como, opcionalmente, con parámetros de salida que puedan actuar sobre otros VIs.

Estos controles e indicadores se muestran también en el diagrama de bloques, de una forma representativa (Figura 7.3). Allí se relacionarán con distintos elementos aritméticos, lógicos o de una amplia gama de tipos, uniéndolos mediante cables.

Las operaciones que en el diagrama de bloques se pueden realizar son numerosas, son todas aquellas que cualquier lenguaje de programación permitiría. LabVIEW además, permite aumentar sus posibilidades para desarrollo de software mediante la inclusión de distintas librerías.

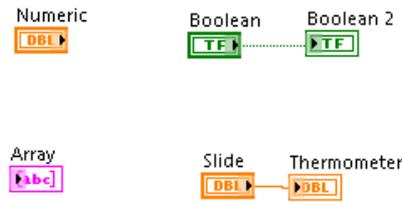


Figura 7. 3: Vista en el diagrama de bloques

A pesar del elevado número de operaciones posibles, en el siguiente epígrafe se mostrarán las operaciones más comunes para programación básica.

1.4 Operaciones y estructuras

Todo lenguaje de programación ofrece una serie de funciones, operaciones y estructuras gracias a los cuales determinar la manera en la que el programa o aplicación que se esté diseñando deba comportarse. Ejemplos de esto son operaciones aritméticas (sumas, restas...), funciones de entrada/salida (como *scanf()* en C o *println()* en JAVA), y estructuras de control (bucles while, do... while o for).

En LabVIEW se presentan estas funciones de forma visual, como dibujos con al menos una entrada o una salida, que se relaciona con otros elementos mediante líneas o cables. A continuación se detallan las más importantes:

Operaciones aritméticas

Las operaciones más básicas en lo que a programación se refiere. LabVIEW cuenta con los elementos para realizar este tipo de operaciones, como sumar, restar, multiplicar, calcular el cociente y resto de una división y otra gran variedad de opciones.

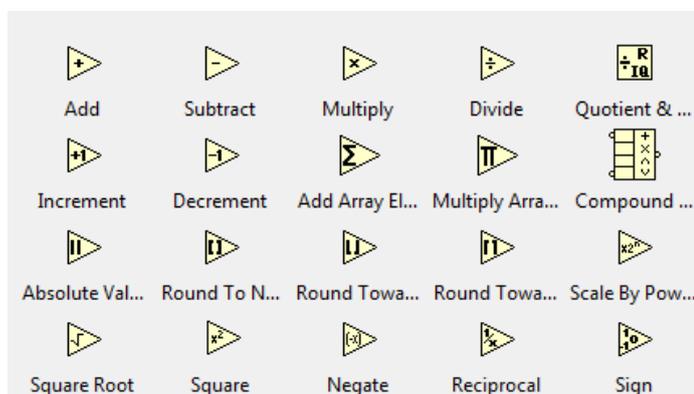


Figura 7. 4: Operaciones aritméticas

Operaciones lógicas

Del mismo modo, existen operaciones lógicas o booleanas, indispensables en el campo de la electrónica y la informática, e igualmente importantes en el campo de la programación. Estas son, por ejemplo, funciones como la OR, la NOT o la NAND entre otras. Se detallan las más importantes.

- **And:** Devuelve FALSE si existe un 0 lógico en alguna de las entradas, y TRUE en caso contrario. El funcionamiento es a la inversa en el caso de Not And.
- **Or:** Devuelve TRUE si existe un 1 lógico en alguna entrada, y FALSE en caso contrario. Funciona a la inversa en el caso de Not Or.
- **Exclusive Or:** Devuelve FALSE si ambas entradas poseen el mismo valor, y TRUE en caso contrario. El funcionamiento es inverso para Not Exclusive Or.
- **Not:** Niega la entrada, convirtiendo un FALSE en TRUE, o viceversa.
- **And/Or Array Elements:** Realiza las funciones explicadas para And/Or con todos los elementos de un array.
- **True/False Constant:** Crea una constante con valor TRUE o FALSE que se puede utilizar como entrada para otra operación o con otro propósito.

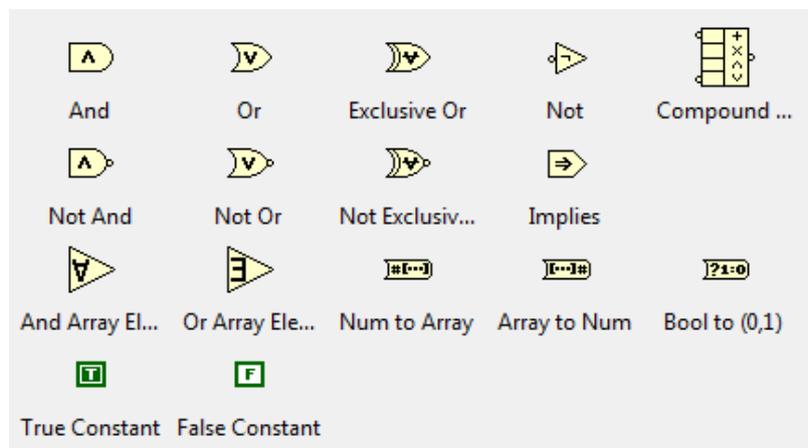


Figura 7. 5: Operaciones lógicas

Operaciones sobre cadenas de texto o string

LabVIEW brinda una serie de funciones como búsqueda de caracteres, concatenación de cadenas o conversión de éstas a otros tipos de datos, como numéricos o de fecha.

- **String Length:** Devuelve el número de caracteres de la cadena.
- **Concatenate:** Toma dos cadenas de texto como entrada y devuelve la unión de las mismas.

- **String subset:** Extrae, dadas una posición y longitud, una sub-cadena desde una cadena de texto.
- **Replace subset:** Modifica una zona determinada de una cadena de texto.
- **Match pattern:** Busca una sub-cadena en una cadena de texto y devuelve los datos previos, posteriores, dicha sub-cadena y la posición donde ésta acaba.
- **Spreadsheet to array:** Toma como entrada una hoja de datos y la convierte en un array. La función Array to spreadsheet realiza la acción opuesta.
- **String constant:** Una constante de tipo string con un valor dado por el usuario para poder utilizarse en otra parte del código. Junto a ésta, existen otra serie de constantes como retorno de carro, carácter fin de línea o tabulación.

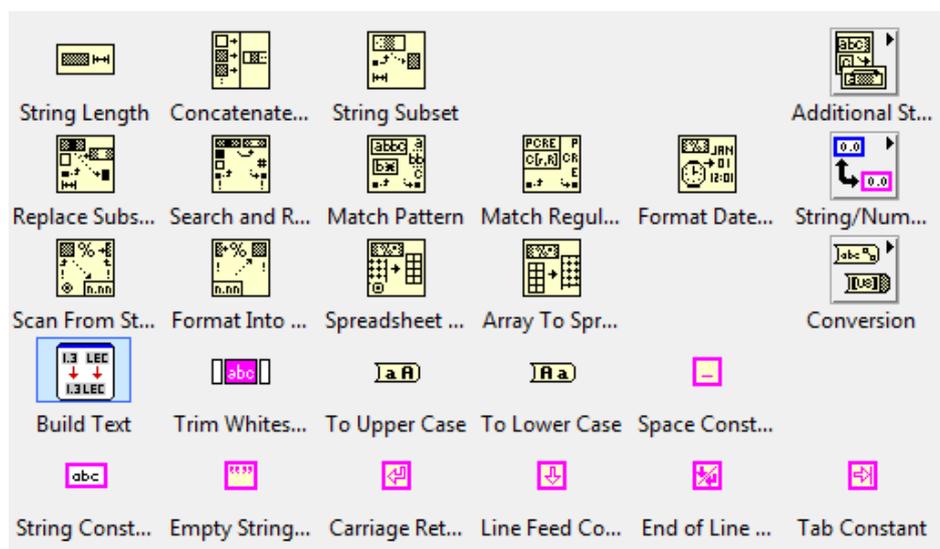


Figura 7. 6: Operaciones con cadenas de texto.

Operaciones de comparación de datos

Operaciones muy básicas y muy utilizadas no ya exclusivamente en el ámbito de la programación sino en la vida cotidiana, son las comparaciones. LabVIEW permite múltiples posibilidades para comparación de datos.

- **Equal:** Compara entre dos números, con una salida booleana que vale TRUE en el caso de que sean iguales o FALSE si no lo son. Las operaciones Not equal, Greater, Less, etc., realizan las demás comparaciones (números desiguales, mayor que, menor que, respectivamente).
- **Equal to 0:** Compara si la entrada es igual a 0. Del mismo modo, existen otras funciones relacionadas, como Not equal to 0, Greater or equal than 0 o less than 0 que comparan si el número es distinto de cero, mayor o igual a cero, o menor que cero, respectivamente.
- **Select:** Esta interesante función realiza las veces de multiplexor, de modo que toma dos entradas de datos y una booleana, y en función del valor TRUE o FALSE que reciba, volcará a la salida una de las dos entradas.

- **Max & min:** Devuelve para una señal de entrada cuáles son sus valores máximo y mínimo.



Figura 7. 7: Operaciones de comparación.

Operaciones con arrays

En cualquier escenario de desarrollo de software es siempre necesario organizar datos de un mismo tipo en arrays. Este entorno de programación ofrece también operaciones para trabajar con estas unidades de datos.

- **Array size:** Toma como entrada un array y devuelve como salida su tamaño.
- **Index array:** Toma un array y, opcionalmente, valores a las entradas para devolver los datos almacenados en dichas posiciones. Si no se indican valores de entrada, devuelve los datos de las consecutivas posiciones en cada salida.
- **Insert into array:** Inserta un elemento en un array, en el lugar indicado como parámetro.
- **Build array:** Forma un array a partir de las entradas de datos que le sean conectadas.
- **Reshape array:** Reordena el array en base a unas dimensiones indicadas.
- **Transpose 2D array:** transpone un array bidimensional, intercambiando el número de filas por el número de columnas.
- **Array constant:** crea un array de constantes, cuyo tipo será determinado por el tipo de constante que se arrastre a su interior.

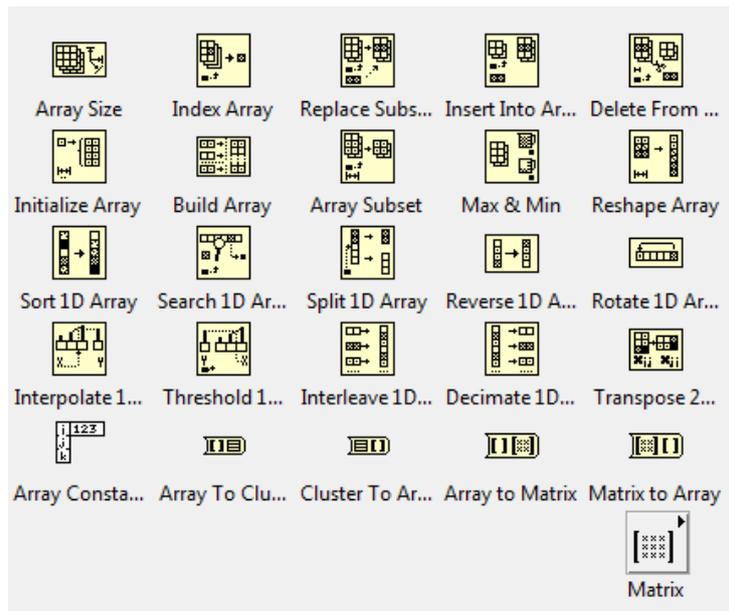


Figura 7. 8: Operaciones con arrays.

Operaciones con clústeres

En ocasiones se necesita trabajar con distintos tipos de datos, y resulta práctico agruparlos en una sola unidad o clúster, de modo que se reduzcan drásticamente el número de entradas o salidas que un determinado programa requiera. En LabVIEW existen una serie de funciones para trabajar con estas unidades de datos.

- **Unbundle by name:** Dado un clúster de datos, lo descompone en sus distintas variables mostrando el nombre de las mismas.
- **Bundle by name:** Inserta nuevas variables en un clúster, obteniéndose a la salida el nuevo conjunto de datos.
- **Build cluster array:** Construye un array de clústeres.
- **Cluster to array:** Convierte un cluster en un array.
- **Cluster constant:** Crea un clúster vacío en el que pueden introducirse constantes de diferentes tipos.

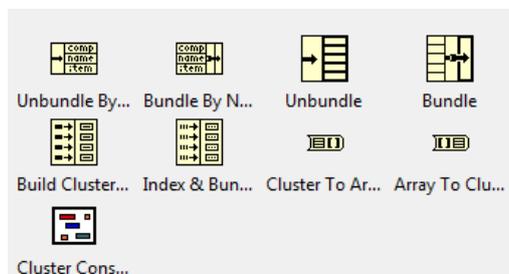


Figura 7. 9: Operaciones con clústeres.

Operaciones con archivos

En cualquier software no ya cotidiano sino mínimamente profesional, se trabaja en algún momento con archivos. A continuación se detallan algunas funciones que LabVIEW proporciona para trabajar con ellos:

- **Write spreadsheet file:** Crea un fichero de texto como hoja de datos, con valores separados por un carácter especificado y con una ubicación y nombre dados. Su opuesto es Read spreadsheet file, que lee de una hoja de datos y almacena una fila o varias en un array.
- **Open/create file:** Abre un archivo referenciado o, en caso de no existir, lo crea. Por su parte, Close file cierra el archivo.
- **Build path:** Dada una ruta y una cadena de texto, construye la ruta resultante de concatenar los anteriores datos. Por el contrario, Strip path secciona una ruta a la altura de la última subcarpeta, devolviendo ésta por un lado y la ruta resultante por otro.

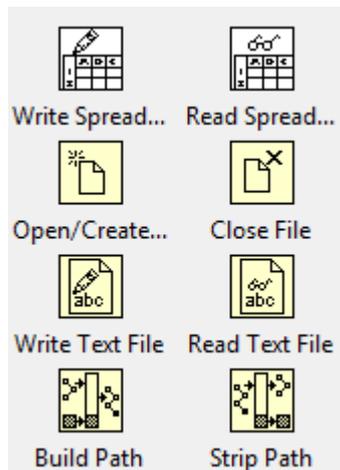


Figura 7. 10: Operaciones con archivos.

Operaciones de temporización

Cualquier código programado tiende a ejecutarse lo más rápidamente posible, adquiriendo la atención del procesador. En muchas ocasiones, esta no es una condición deseable, ya que puede provocar la inestabilidad del sistema o de otras secciones de código. Por otra parte, en alguna ocasión, el usuario quiere provocar retardos intencionados en la ejecución del programa, que permitan, por ejemplo, leer un mensaje de texto antes de que éste desaparezca a causa de las siguientes funciones. Pueden encontrarse en LabVIEW las siguientes operaciones para trabajar con temporización:

- **Wait (ms):** Espera el determinado número de milisegundos antes de continuar (espera asíncrona).
- **Wait until next (ms):** Espera un determinado número de milisegundos para proseguir en el siguiente ciclo (espera síncrona).
- **To time stamp:** Convierte un número a formato fecha-hora.

- **Get date/time string:** Obtiene la fecha y hora y la convierte a una cadena de texto según los parámetros que le sean indicados.
- **Get date/time in seconds:** Obtiene la fecha y hora en segundos.

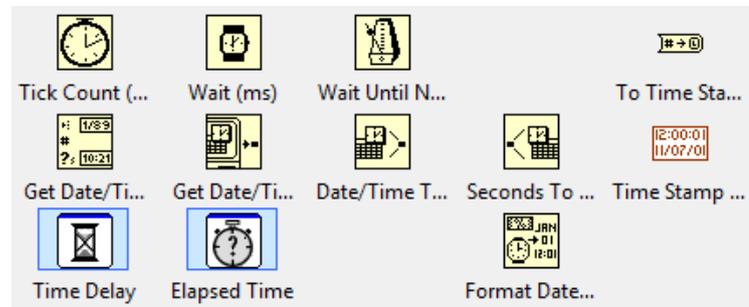


Figura 7. 11: Operaciones de temporización.

Estructuras de ejecución de código

Una herramienta básica a la hora de programar es la utilización de determinados elementos que son los que realmente determinen cómo ha de comportarse el código en las diversas casuísticas. Estos elementos, como bucles y condicionales, podemos encontrarlos en este entorno de programación:

- **For loop:** Repite el código que exista en su interior un número determinado de veces.
- **While loop:** Repite el código de su interior hasta que se cumpla o se deje de cumplir una condición determinada. Timed structure funciona de manera similar, pero permitiendo elegir una duración determinada para los ciclos de repetición.
- **Case structure:** Toma una variable de entrada que puede ser booleana (en tal caso funciona como un if) o tener múltiples valores. Para cada uno de los valores de dicha variable de entrada, ejecuta un código distinto, programado en su interior.
- **Flat sequence:** Permite escribir código por etapas que se sucederán secuencialmente. Flat sequence se diferencia de Stacked sequence en la representación, ya que mientras la primera muestra todas las secuencias del código consecutivamente, la segunda muestra las secuencias de una en una en un único marco.
- **Diagram disable:** Permite elegir una parte del código para deshabilitarlo, e incluso habilitar otro código en su lugar. Es muy útil para funciones de depuración de errores.
- **Formula node:** Permite realizar fórmulas u operaciones matemáticas de modo escrito, tomando variables de entrada y generando variables de salida, que han de estar conectadas a variables existentes en el programa. MathScript permite un uso similar pero a nivel de script.
- **Local variable/global variable:** Permiten crear o llamar a una variable local/global del programa.

- **Shared variable:** Cuando existen proyectos, es decir, un conjunto de VIs diferentes que forman parte de un mismo programa, y es preciso el acceso a determinadas variables en dos o más VIs, esta herramienta nos permite crear variables compartidas.

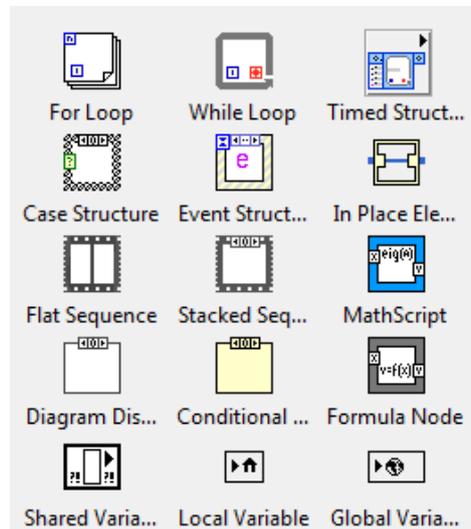


Figura 7. 12: Estructuras de ejecución de código.

1.5 Ficheros ejecutables

El destino final para el código programado es la creación de nuevo software, o al menos la optimización o mejora del ya existente. Es por ello que deben existir uno o más archivos ejecutables.

Durante el proceso de desarrollo, programación y pruebas, se ejecuta el software desde el mismo entorno de programación. No obstante, una vez está finalizado el proceso de desarrollo, lo que se pretende es compilar ese código generando un archivo que pueda ejecutarse en cualquier equipo, evidentemente dadas unas características que lo permitan.

Para un proyecto en LabVIEW, además de los VIs y las operaciones para trabajar con ellos, se muestran al usuario una serie de herramientas de compilación, siendo las más importantes dos: build Application (EXE), y build Installer.

- Build Application (EXE) presenta un menú con múltiples funciones y submenús. En éste, se pueden especificar los archivos que se incluyen en la compilación, aquellos que deberían iniciarse al arrancar la aplicación, si se debería optimizar el código para trabajar con microprocesadores actuales, el nombre que se le quiere dar a la aplicación, etc. El resultado de la compilación será un archivo ejecutable .exe con el nombre dado por el usuario.
- Build Installer, una vez que existe un compilado ejecutable, permite crear un instalador para que dicha aplicación sea implantada en un equipo en base a unos criterios. Este menú permite elegir los archivos que han de instalarse, la ruta donde han de hacerlo, los sistemas operativos Windows en los que ha de

funcionar así como la posibilidad de incluir opcionalmente otro software de National Instruments. El resultado es un paquete de instalación transportable a cualquier entorno Windows que se haya contemplado durante el proceso de creación del instalador.

Existe un detalle que se debe tener en cuenta a la hora de crear una aplicación con LabVIEW. Este entorno de programación incluye su propio lenguaje y funciones, con lo que un archivo compilado con una determinada versión de LabVIEW requiere que exista su correspondiente versión del motor de ejecución LabVIEW Runtime Engine. Una práctica aconsejable a la hora de desarrollar software en LabVIEW es la creación de un instalador que copie, además de la aplicación principal, el motor de ejecución en el sistema destino.

Apéndice 2

Funciones Modbus

2.1 Introducción

Para la realización de este Sistema de Gestión Energética se han empleado diferentes **funciones** del protocolo de comunicaciones Modbus. Este es un protocolo muy utilizado en el ámbito de las comunicaciones industriales, debido a su sencillez y a otros factores como son la facilidad de uso y que éste sea un protocolo de uso público.

Si bien dicho protocolo de comunicaciones ha sido explicado en profundidad en uno de los capítulos anteriores, la explicación de sus funciones puede haber resultado escueta y demasiado ceñida a su uso para llevar a cabo el **proyecto**.

En este apéndice se explican con mayor **detalle** las funciones Modbus cuyo uso resulta más común en el ámbito de las comunicaciones industriales.

2.2 Descripción de códigos de función

Como todo protocolo, cada mensaje se compone de una serie de campos. En el caso de Modbus, uno de los campos más importantes es el de **función**, ya que es el que determina la operación que quiere realizarse con los datos que contiene, así como el significado que estos adquieren.

A lo largo de este epígrafe se explican los distintos códigos de función, diferenciando entre mensajes de **petición**, **respuestas** y mensajes de **error**.

2.2.1 Lectura de registros discretos (Código 1, 0x01)

Este código de función se utiliza para **leer** el estado de los **registros discretos** (de 1 a 2000) de un equipo. El mensaje de solicitud especifica la dirección inicial (la dirección del primer registro especificado) y el número de registros discretos. En el mensaje, las direcciones de los registros están numeradas empezando desde cero, de modo que los registros numerados del 1 al 16 tienen como direcciones de la 0 a la 16.

Los registros discretos en el mensaje de respuesta quedan encapsulados como un registro por bit del campo de **datos**. Su estado queda indicado como un 1 cuando están activos y como un 0 cuando no lo están. El bit menos significativo del primer byte de datos contiene la dirección de origen de la petición previa. Los demás registros discretos se encuentran a continuación en los siguientes bits de menor a mayor de este byte, y de los siguientes bytes también de menor a mayor si son necesarios.

Si la cantidad de registros devuelta no es un **múltiplo** de 8, los bits que quedan al final del último byte de datos son **rellenados** con ceros. El campo de Número de Bytes determina la cantidad de bytes completos de datos.

Mensaje de solicitud

Código de función	1 Byte	0x01
Dirección inicial	2 Bytes	De 0x0000 a 0xFFFF
Cantidad de registros	2 Bytes	De 1 a 2000 (0x7D0)

Mensaje de respuesta

Código de función	1 Byte	0x01
Número de Bytes	1 Byte	N*
Estado del registro	n Bytes	n = N ó N+1

(N es la cantidad de registros entre 8. Si el resto no es cero, n valdrá N+1)

Mensaje de error

Código de error	1 Byte	0x81
Código de excepción	1 Byte	01, 02, 03 o 04

2.2.2 Lectura de entradas discretas (Código 2, 0x02)

Este código de función se utiliza para **leer** el estado de las **entradas discretas** (de 1 a 2000) de un dispositivo. El mensaje de solicitud especifica la dirección inicial (como la dirección de la primera entrada a leer) y el número de entradas. En el mensaje, las entradas discretas tienen como dirección más pequeña el cero. De este modo, las entradas discretas de la 1 a la 16 están representadas por las direcciones de la 0 a la 15.

Las entradas discretas en el mensaje de respuesta son empaquetadas como una entrada por bit del campo de **datos**. Los estados de las entradas se indicarán como un 1 cuando se encuentran activas y un 0 en el caso de estar inactivas. El bit menos significativo del primer bit de datos contiene la dirección de la entrada inicial especificada en la solicitud. El resto de entradas siguen en orden ascendente en el resto de bits del mismo byte de datos y, en caso de ser necesario, en el resto de bytes de datos, también de manera ascendente.

Si la cantidad de entradas devuelta no es un múltiplo de 8, el resto de bits del último byte de datos se **rellenarán** con ceros (siguiendo el orden ascendente). El campo de Número de Bytes especifica la **cantidad** de bytes de datos completos.

Mensaje de solicitud

Código de función	1 Byte	0x02
Dirección inicial	2 Bytes	De 0x0000 a 0xFFFF
Cantidad de entradas	2 Bytes	De 1 a 2000 (0x7D0)

Mensaje de respuesta

Código de función	1 Byte	0x02
Número de Bytes	1 Byte	N*
Estado de la entrada	n Bytes	n = N ó N+1

(N es la cantidad de entradas entre 8. Si el resto no es cero, n valdrá N+1)

Mensaje de error

Código de error	1 Byte	0x82
Código de excepción	1 Byte	01, 02, 03 o 04

2.2.3 Lectura de registros de almacenamiento (Código 3, 0x03)

Este código de función es utilizado para **leer** los contenidos de todo un bloque de **registros de almacenamiento contiguos** en un dispositivo. El mensaje de solicitud indica la dirección del primer registro a leer y el número de registros. En el mensaje, la dirección de los registros comienza desde 0. Por ello, los registros numerados del 1 al 16 tienen como dirección de la 0 a la 15.

Los datos del registro en el mensaje de respuesta vienen encapsulados como dos bytes por registro, con el contenido binario justificado en cada byte. Por cada registro, el **primer** byte contiene los bits de **mayor** peso y el **segundo** contiene los bits de **menor** peso.

Mensaje de solicitud

Código de función	1 Byte	0x03
Dirección inicial	2 Bytes	De 0x0000 a 0xFFFF
Cantidad de registros	2 Bytes	De 1 a 125 (0x7D)

Mensaje de respuesta

Código de función	1 Byte	0x03
Número de Bytes	1 Byte	2 x N*
Valor de registro	N* x 2 Bytes	

(N es la cantidad de registros)

Mensaje de error

Código de error	1 Byte	0x83
Código de excepción	1 Byte	01, 02, 03 o 04

2.2.4 Lectura de registros de entrada (Código 4, 0x04)

Este código de función se utiliza para **leer** los **registros de entrada contiguos** (de 1 hasta 125) en un dispositivo. El mensaje de solicitud especifica la dirección del primer registro a leer y el número de registros. La dirección de los registros comienza desde cero, con lo que los registros del 1 al 16 tienen como dirección de la 0 a la 15.

Los datos del registro en el mensaje de respuesta se encapsulan como dos bytes por registro, con el contenido binario justificado en cada byte. Para cada registro, el **primer** byte contiene los bits de **mayor** peso y el **segundo** contiene los de **menor** peso.

Mensaje de solicitud

Código de función	1 Byte	0x04
Dirección inicial	2 Bytes	De 0x0000 a 0xFFFF
Nº de registros de entrada	2 Bytes	De 1 a 125 (0x7D)

Mensaje de respuesta

Código de función	1 Byte	0x04
Número de Bytes	1 Byte	2 x N*
Registros de entrada	N* x 2 Bytes	

(N es la cantidad de registros de entrada)

Mensaje de error

Código de error	1 Byte	0x84
Código de excepción	1 Byte	01, 02, 03 o 04

2.2.5 Escritura de un único registro discreto (Código 5, 0x05)

Este código de función se utiliza para la **escritura** de un **único registro discreto** (de salida) de modo que quede activo o inactivo en un dispositivo. El estado de activación o desactivación solicitado es indicado por una constante en el campo de **Valor del Registro Discreto** de la solicitud. Un valor de **0xFF00** solicita que la salida quede **activa**. Un valor de **0x0000** determinará que la salida quede **inactiva**. Cualquier otro valor será **ilegal** y no afectará al registro discreto.

El mensaje de solicitud especifica la dirección del registro discreto a ser escrito. Los registros tienen como dirección inicial 0, lo que implica que el registro 1 tiene como dirección la 0, y lo mismo le sucederá al resto.

La respuesta normal es un **eco** de la solicitud, devuelta una vez que el estado del registro discreto ha sido escrito.

Mensaje de solicitud

Código de función	1 Byte	0x05
Dirección de registro d.	2 Bytes	De 0x0000 a 0xFFFF
Valor de registro discreto	2 Bytes	0xFF00 (ON) ó 0x0000 (OFF)

Mensaje de respuesta

Código de función	1 Byte	0x05
Dirección de registro d.	2 Bytes	De 0x0000 a 0xFFFF
Valor de registro discreto	2 Bytes	0xFF00 (ON) ó 0x0000 (OFF)

Mensaje de error

Código de error	1 Byte	0x85
Código de excepción	1 Byte	01, 02, 03 o 04

2.2.6 Escritura de un único registro de almacenamiento (Código 6, 0x06)

Este código de función se utiliza para llevar a cabo la **escritura** de un **único registro de almacenamiento** en un dispositivo.

El mensaje de solicitud especifica la dirección del registro a escribir. Éstos registros son direccionados desde la dirección 0, con lo que cada registro tiene como dirección una unidad menos, siendo el registro 1 correspondido con la dirección 0, por ejemplo. La respuesta normal es un **eco** de la solicitud una vez que el registro ha sido escrito.

Mensaje de solicitud

Código de función	1 Byte	0x06
Dirección de registro d.	2 Bytes	De 0x0000 a 0xFFFF
Valor de registro discreto	2 Bytes	De 0x0000 a 0xFFFF

Mensaje de respuesta

Código de función	1 Byte	0x06
Dirección de registro d.	2 Bytes	De 0x0000 a 0xFFFF
Valor de registro discreto	2 Bytes	De 0x0000 a 0xFFFF

Mensaje de error

Código de error	1 Byte	0x86
Código de excepción	1 Byte	01, 02, 03 o 04

2.2.7 Lectura del estado de las excepciones (Código 7, 0x07)

Este código de función se utiliza para **leer** los contenidos de los ocho **registros de Excepción** en un dispositivo.

La función proporciona un método simple para el acceso a esta información, ya que las **referencias** de las posibles excepciones son **conocidas** (no se necesita referencia para el registro en la función). La respuesta normal contiene el **estado** de los ocho **registros de Excepción**. Las salidas son encapsuladas en un byte de datos, con un bit por salida. El estado de la referencia de salida más baja está contenido en el bit menos significativo del byte de datos.

Los **contenidos** de los registros de las ocho Excepciones son **específicos** para cada dispositivo.

Mensaje de solicitud

Código de función	1 Byte	0x07
-------------------	--------	------

Mensaje de respuesta

Código de función	1 Byte	0x07
Datos de salidas	1 Byte	De 0x00 a 0xFF

Mensaje de error

Código de error	1 Byte	0x87
Código de excepción	1 Byte	01 o 04

2.2.8 Escritura de *n* registros discretos (Código 15, 0x0F)

Este código de función se utiliza para **establecer** cada **registro discreto** de una **secuencia** de éstos a activo o inactivo en un dispositivo. El mensaje de solicitud indica las referencias de los registros discretos a ser escritos. Éstos se direccionan desde la dirección mínima 0, con lo que el registro número 1 tiene como dirección 0.

Los estados de activo/inactivo solicitados se especifican en el contenido del campo de **datos** de la petición. Un 1 lógico en una **posición** de un bit de dicho campo implica la

solicitud de **activación** de la salida **correspondiente**. Un 0 lógico solicitará su desactivación. La respuesta normal devuelve el código de función, dirección de comienzo y cantidad de registros discretos a escribir.

Mensaje de solicitud

Código de función	1 Byte	0x0F
Dirección de inicio	2 Bytes	De 0x0000 a 0xFFFF
Cantidad de registros d.	2 Bytes	De 0x0000 a 0x07B0
Número de bytes	1 Byte	N*
Valor de los registros d.	N* x 1 Byte	

(N es la cantidad de registros entre 8. Si el resto es distinto de 0, vale N+1).

Mensaje de respuesta

Código de función	1 Byte	0x0F
Dirección de inicio	2 Bytes	De 0x0000 a 0xFFFF
Cantidad de registros d.	2 Bytes	De 0x0000 a 0x07B0

Mensaje de error

Código de error	1 Byte	0x8F
Código de excepción	1 Byte	01, 02, 03 o 04

2.2.9 Escritura de n registros de almacenamiento (Código 16, 0x10)

Este código de función se utiliza para **escribir** un **conjunto** indicado de **registros de almacenamiento** contiguos (desde 1 a 123) en un dispositivo. Los valores a escribir solicitados son indicados en el campo de **datos** de la petición. Los datos son encapsulados como dos bytes por cada registro. La respuesta normal devuelve el código de función, así como la dirección de inicio y cantidad de registros escritos.

Mensaje de solicitud

Código de función	1 Byte	0x10
Dirección de inicio	2 Bytes	De 0x0000 a 0xFFFF
Cantidad de registros	2 Bytes	De 0x0000 a 0x007B
Número de bytes	1 Byte	2 x N*
Valor de los registros	N* x 2 Bytes	(valor especificado)

(N es la cantidad de registros).

Mensaje de respuesta

Código de función	1 Byte	0x10
Dirección de inicio	2 Bytes	De 0x0000 a 0xFFFF
Cantidad de registros	2 Bytes	De 0x0000 a 0x007B

Mensaje de error

Código de error	1 Byte	0x90
Código de excepción	1 Byte	01, 02, 03 o 04

2.3 Códigos de excepción

A lo largo de este apéndice se han detallado las funciones Modbus más utilizadas en comunicaciones para uso industrial. A lo largo de tal explicación, se puede apreciar que ha sido **recurrente** la aparición de determinados códigos de **excepción**.

Procede a continuación explicar estos **cuatro** códigos de excepción más frecuentes, lo que implican y sus posibles causas. La siguiente tabla proporcionará tal información:

CÓDIGOS DE EXCEPCIÓN DE MODBUS		
Código	Nombre	Significado
01	FUNCION ILEGAL	El código de función recibido en la solicitud no es una acción permitida para el servidor. Esto puede deberse a que el código solicitado solo sea aplicable a dispositivos más modernos, y no esté implementado en el equipo actual. Puede indicar también que el servidor está en un estado erróneo para procesar una petición de este tipo, por ejemplo porque no se encuentre configurado y le esté siendo solicitado el retorno de valores de registros.
02	DIRECCION DE DATOS ILEGAL	La dirección de datos especificada en la petición no es una dirección permitida en el servidor actual. Más específicamente, la combinación del número de referencia y la longitud de la transferencia es inválida. Para un controlador con 100 registros, el mensaje direcciona el primer registro como 0 y el último como 99. Si una petición se envía con un valor de dirección de inicio de 96 y una cantidad de 4 registros, operará correctamente en los registros 96, 97, 98 y 99. Si por el contrario se envía una petición con valor de dirección de inicio de 96 y una cantidad de 5 registros, entonces fallará con este Código de Excepción, ya que no existe un registro 100.
03	VALOR DE DATOS ILEGAL	Un valor contenido en el campo de datos de la solicitud no es un valor permitido para el actual servidor. Esto indica un fallo en la estructura del resto de una petición compleja, como en las que la longitud implicada no es correcta. Esto NO significa que un dato enviado para almacenar en un registro tenga un valor fuera de lo que el programa espera, ya que Modbus no se hace eco del significado de ningún valor particular para ningún registro.
04	FALLO DEL EQUIPO SERVIDOR	Un error irrecuperable ocurrió cuando el servidor intentaba llevar a cabo la acción solicitada.

Figura 8. 1: Códigos de excepción más recurrentes de Modbus.

REFERENCIAS

[SCADA]: Introducción a SCADA:

<http://www.uco.es/grupos/eatco/automatica/ihtm/descargar/scada.pdf>

[ISO]: Web oficial: <http://www.iso.org>

[UNE-EN 16001]: Web AENOR – SERCOBE:

<http://www.aenor.es/aenor/normas/ctn/fichactn.asp?codigonorm=AEN/CTN%20115&pagina=1#.Uib199JT5q0>

[UNE-EN 12464-1]: Web AENOR – ANFALUM:

<http://www.aenor.es/aenor/normas/ctn/fichactn.asp?codigonorm=AEN/CTN%2072&pagina=1#.Uib0-9JT5q0>

[Modbus]: Web oficial: <http://www.modbus.org/>

[OSI]: Definición y explicación del modelo: <http://support.microsoft.com/kb/103884/es>

[PAC3100]: Ficha del producto:

https://www.swe.siemens.com/spain/web/es/ic/mvkv/low_voltage/analizadores_sen tron/sen tronpac3100/pages/pac3100.aspx

[PAC4200]: Ficha del producto:

https://www.swe.siemens.com/spain/web/es/ic/mvkv/low_voltage/analizadores_sen tron/SENTRONPAC4200/Pages/PAC4200.aspx

[LabVIEW]: Web oficial del software: <http://www.ni.com/labview/esa/>

[CompactRIO 9076]: Ficha del producto:

<http://sine.ni.com/nips/cds/view/p/lang/es/nid/209758>