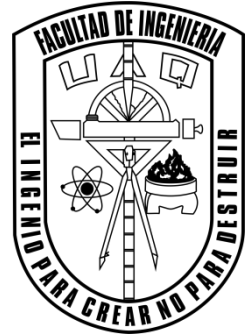


Universidad Autónoma de Querétaro

Campus San Juan del Río

Facultad de ingeniería



MONITOREO REMOTO VÍA ETHERNET DE LA CALIDAD DE
LA ENERGÍA EN SISTEMAS DE BAJA TENSIÓN

Tesis

Que como parte de los requisitos para obtener el título de
Ingeniero Electromecánico

Presenta:

Arturo Mejía Barrón

Dirigido por:

M. en I. Armando Guadalupe García Ramírez

Dr. Roque Alfredo Osornio Ríos

San Juan del Río, Qro. Septiembre del 2013

Resumen

Anteriormente la comunicación vía Ethernet era una tecnología disponible solo para computadoras, en la actualidad el uso de este tipo de comunicación se ha orientado a la aplicación de sistemas embebidos debido a la aparición de chips controladores que realizan la comunicación física entre un sistema de monitoreo y una o varias computadoras. Las redes Ethernet son un medio de comunicación estándar con el que actualmente todas las computadoras, desde personales hasta computadoras de escritorio cuentan, que permiten acceso a datos de un dispositivo conectado dentro de la misma red y en caso de contar con conexión a Internet funciona como un medio de comunicación remota que puede ser aprovechado para sistemas de monitoreo a distancia. En el presente proyecto se realizó la integración de una interfaz Ethernet que consta de un controlador Ethernet W5100 y una plataforma de hardware libre Arduino. Esta interfaz se integró a un sistema basado en arreglos de compuertas lógico programables FPGA (por sus siglas en inglés *Field Programmable Gate Array*) de monitoreo de calidad de la energía desarrollado dentro de la Universidad Autónoma de Querétaro con la finalidad de realizar monitoreo remoto de sistemas monofásicos de baja tensión o monitoreo en sistemas alimentados con corriente continua. Se estableció una comunicación serial RS-232 para la transferencia de datos entre Arduino-FPGA, donde la tarjeta Arduino contiene una página web embebida la cual se encarga de activar un monitoreo de 200 milisegundos, posteriormente los datos adquiridos son enviados a la tarjeta Arduino y guardados en un archivo de texto. Por otra parte se desarrolló una interfaz gráfica de usuario capaz de acceder a la página embebida para adquirir y procesar los datos calculando parámetros de calidad de la energía, el monitoreo continuo lo lleva a cabo el software seleccionando el intervalo de medición y el tiempo que se llevara a cabo el monitoreo. El software permite visualizar los parámetros mientras se actualizan y aguarda el historial de los parámetros que se deseen obtener para su posterior análisis. En este documento se describe el sistema a detalle y se muestran las pruebas realizadas que demuestran su funcionamiento.

Dedicatoria

“A mi padre por su apoyo incondicional,
A mi madre por sus sabios consejos,
A mis hermanas por motivarme a seguir adelante.”

Agradecimientos

A mis padres por brindarme su apoyo y motivación para realizar mis estudios en esta institución.

A mi asesor el M. en I. Armando Guadalupe García Ramírez y co-asesor Dr. Roque Alfredo Osornio Ríos, por su apoyo y motivación en este proyecto y a lo largo de mis estudios.

Al Dr. David Granados Lieberman y al M. en C. Arturo Yosimar Jaen Cuellar por su apoyo.

A la Universidad Autónoma de Querétaro, por permitirme realizar mis estudios en esta institución.

A mis profesores por compartirnos sus conocimientos.

A mis compañeros por sus motivación y enseñanzas.

Gracias!

Índice general

Resumen	I
Dedicatoria.....	II
Agradecimientos.....	III
Índice general	IV
Índice de figuras	VII
Índice de tablas	VIII
Capítulo 1	1
Introducción	1
1.1 Antecedentes	2
1.2 Justificación.....	5
1.3 Objetivo.....	7
1.3.1 Objetivos Particulares	7
1.4 Planteamiento General	8
Capítulo 2	11
Estado del arte.....	11
2.1 Ethernet	11
2.1.1 Componentes de una red.....	12
2.1.2 Cableado Ethernet.....	13
2.1.3 Encapsulación	13
2.2 Protocolos y capas.....	14
2.2.1 Modelo de referencia OSI.....	15
2.2.2 Modelo TCP/IP	16
2.3 Servidor web	17

2.4	Página web	17
2.4.1	Lenguaje HTML	18
2.5	Interfaz de usuario	18
2.5.1	Definición de interfaz gráfica de usuario	19
2.6	Microcontrolador	19
2.6.1	Periféricos	20
2.6.1.1	Interfaz UART	20
2.6.1.2	Interfaz SPI	20
2.7	Arduino	21
2.8	Chip Ethernet	22
2.9	Calidad de la energía	23
2.9.1	Índices de calidad de la energía en corriente alterna	24
2.9.2	Índices de calidad de la energía en corriente continua	26
2.10	Herramientas de diseño	27
2.10.1	IDE de Arduino	27
2.10.2	Matlab GUI	29
Capítulo 3	30
Metodología	30
3.1	Implementación de comunicación serial a sistema de monitoreo	32
3.1.1	Módulo de comunicación serial	33
3.1.2	Módulo de lectura de SDRAM	35
3.1.3	Módulo de control principal	36
3.2	Página web embebida	37
3.3	Interfaz gráfica de usuario	38
Capítulo 4	39
Pruebas y resultados	39
4.1	Página web embebida	40
4.1.1	Acceso por conexión punto a punto	42
4.1.2	Acceso por red local	43
4.1.3	Acceso por Internet	46
4.2	Pruebas preliminares de adquisición	47

4.3	Interfaz gráfica de usuario.....	49
4.4	Pruebas preliminares en casa-habitación.....	52
4.5	Prueba final en Fitotrón.....	61
4.6	Ventajas y desventajas	66
Capítulo 5	68
Conclusiones	68
Trabajo futuro	69
Bibliografía.....		70
ANEXOS		72

Índice de figuras

Figura 1.1. Diagrama a bloques de sistema de monitoreo de calidad de la energía.	9
Figura 1.2. Diagrama a bloques de interfaz Ethernet.	9
Figura 2.1. Componentes de una red.	12
Figura 2.2. Ejemplo de encapsulación.	14
Figura 2.3. Protocolos y capas.	15
Figura 2.4. Modelo de referencia OSI de siete capas.	15
Figura 2.5. Modelo de referencia TCP/IP.	16
Figura 2.6. Diagrama a bloques W5100.	22
Figura 2.7. IDE de Arduino.	28
Figura 3.1. Proceso para integración de interfaz Ethernet.	31
Figura 3.2. Diagrama general de bloques de sistema de monitoreo.	33
Figura 3.3. Diagrama de bloques de módulo de comunicación serial.	34
Figura 3.4. Diagrama de bloques de módulo de lectura de SDRAM.	35
Figura 3.5. Diagrama de bloques de módulo de control.	36
Figura 4.1. Página web embebida en tarjeta Arduino.	40
Figura 4.2. a) Gráfica de voltaje, b) Gráfica de corriente.	48
Figura 4.3. Parámetros eléctricos calculados.	49
Figura 4.4. Interfaz gráfica de usuario principal.	50
Figura 4.5. Interfaz gráfica de monitoreo de corriente continua.	51
Figura 4.6. Interfaz gráfica de monitoreo de corriente alterna.	52
Figura 4.7. Monitoreo de corriente alterna.	53
Figura 4.8. Prueba de monitoreo remoto a través de Internet.	57
Figura 4.9. Componentes utilizados para monitoreo en Fitotron.	61

Índice de tablas

Tabla 2.1. Tipos de cableado Ethernet más comunes.....	13
Tabla 4.1. Acceso a página web en conexión punto a punto.....	42
Tabla 4.2. Acceso a página web en red local.....	44
Tabla 4.3. Capturas de pantalla de acceso a página web por Internet.	46
Tabla 4.4. Monitoreo de vivienda en red local.	53
Tabla 4.5. Monitoreo a vivienda través de Internet.	57
Tabla 4.6. Monitoreo a Fitotrón en corriente alterna.....	62
Tabla 4.7. Monitoreo a Fitotrón en corriente continua.	65

Capítulo 1

Introducción

En la actualidad el monitoreo remoto en tiempo real a largo alcance es parte importante de los procesos industriales, ya que ofrece ventajas como el monitoreo en uno o varios ordenadores de las condiciones actuales del sistema, modificar parámetros de operación del equipo sin tener que ir hasta el sitio de medición, realizar arranques y paros a distancia, generar gráficos histogramas para su posterior análisis y tener vigilancia constante de un sistema o proceso desde cualquier ubicación con acceso a Internet.

Ethernet es una tecnología de redes usada en muchas oficinas y hogares que tiene como objetivo habilitar la comunicación entre computadoras y compartir recursos, ya sea en una red local o a través de Internet. Por muchos años esta tecnología estuvo disponible sólo para computadoras, y los sistemas de monitoreo fueron limitados a interfaces de baja velocidad, rango limitado o protocolos de aplicación normalizada. Hoy día, nuevos desarrollos en la tecnología y el mercado hacen posible la comunicación de cualquier sistema en redes locales e Internet, haciendo los sistemas de monitoreo más poderosos y sencillos de controlar.

Un factor que hace la comunicación vía Ethernet ideal para sistemas de monitoreo es la disponibilidad de circuitos integrados de control económicos para manejar los detalles de dicha comunicación. Por otro lado las unidades de control que proporcionan el nivel de procesamiento como por ejemplo los microcontroladores han adquirido más velocidad y más capacidad de manejo de las demandas, y aunque existen un gran número de dispositivos con la capacidad de controlar los circuitos integrados de comunicación e incluso en el mercado

existen sistemas de monitoreo industriales basados en controladores lógicos programables (PLC por sus siglas en inglés *Programmable Logic Controller*).

Debido a que la Universidad Autónoma de Querétaro destaca por la realización de proyectos basados en arreglos de compuertas programables en campo (FPGA por sus siglas en inglés *Field Programmable Gate Array*) en diversas aplicaciones se hace necesario contar con interfaces robustas y actualizadas para el desarrollo tecnológico, es por ello que en este proyecto se integrará una interfaz para monitoreo remoto vía Ethernet a un sistema basado en FPGA por medio de un microcontrolador el cual manejará el controlador Ethernet, albergará el sitio web y realizará la adquisición de datos del FPGA, esto ofrece ventajas como eliminar la dependencia a una computadora personal (PC por sus siglas en inglés *Personal Computer*) para hacer la transmisión dentro de una red local o a través de internet. La finalidad es obtener mediciones de los principales parámetros de calidad de la energía en sistemas de corriente continua y sistemas monofásicos de corriente alterna a través de internet o una red de área local, así como la obtención de índices eléctricos como:

- Valor pico de voltaje y corriente
- Valor eficaz (RMS por sus siglas en inglés *Root Mean Square*) de voltaje y corriente.
- Potencia efectiva, reactiva y potencia aparente
- Factor de potencia
- Distorsión total armónica (THD por sus siglas en inglés *Total Harmonic Distorsion*)

1.1 Antecedentes

El monitoreo remoto vía Ethernet no es una tecnología nueva, incluso existen sistemas industriales que cuentan con esta tecnología. Dentro de la Universidad Autónoma de Querétaro se han realizado una serie de proyectos referidos al monitoreo remoto y por otra parte al monitoreo de calidad de la energía.

Toledano (2006), diseñó un sistema de monitoreo remoto para un invernadero por medio tecnología inalámbrica para transmisión de datos adquiridos desde unidades remotas (ubicadas fuera y dentro del invernadero) hacia la unidad base, la cual se encuentra conectada a un ordenador a través del puerto de bus universal en serie (USB de sus siglas en inglés, *Universal Serial Bus*), donde se analizan los datos y se generan las gráficas en tiempo real de las variables climatológicas medidas, así como reportes en archivo de texto organizados por días, historial de gráficas del comportamiento del clima, etc. Adicionalmente, realizó el monitoreo a través de Internet que consiste en presentar una página web las gráficas de las mediciones que son tomadas actualizándose automáticamente cada 5 minutos. De esta manera logró el monitoreo de las variables climatológicas tanto dentro como fuera del invernadero para observar su comportamiento y generar las acciones correctivas necesarias.

Martínez (2009), implementó la comunicación en red de un equipo de cómputo que maneja una interfaz sistema integral de control contra incendios (SICCI) con una de las válvulas de control para aspersores de descarga de auto-tanques, haciendo uso de un controlador lógico programable (PLC de sus siglas en inglés *Programmable Logic Controller*) , en la parte de comunicación y monitoreo se formó una interfaz de control de supervisión y adquisición de datos (SCADA de sus siglas en inglés *Supervisory Control And Data Acquisition*) donde los datos obtenidos por el PLC son enviados a un ordenador y de ahí son transmitidos a un servidor web.

Guzmán (2009), desarrolló dos aplicaciones de monitoreo, la primera aplicación fue la implementación del accionamiento de un sistema de riego vía Ethernet con interfaz gráfica en LabView, por medio de un PLC se conectó un ordenador a la red en la que se encontraba configurado el sistema de riego, para establecer las horas de riego, los tiempos de encendido, apagado del sistema y la zona fuera ser regada.

Toledano (2010), desarrolló un sistema de telemetría para la medición remota de largo alcance para estaciones de sensores. El sistema fue probado en una aplicación práctica que consiste en el monitoreo remoto de las variables climatológicas en los altos de Chiapas, nombrando al sistema Meteo UAQ. Este sistema consta de tres partes principales: Unidades Terminales Remotas (RTU de sus siglas en ingles *Remote Terminal Unit*), Unidades Terminales Base (BTU de sus siglas en ingles *Base Terminal Unit*) y un servidor central (CS de sus siglas en ingles *Central Server*). La adquisición de datos la realizan las RTUs comunicándose entre ellas para formar una red de área local inalámbrica en la banda destinada por la comisión federal de

telecomunicaciones (COFETEL) para aplicaciones industriales, científicas y médicas (ISM - *Industrial, Scientific and Medical*). Los datos adquiridos son gestionados por las BTUs quienes a su vez envían esta información vía un enlace satelital a un servidor central ubicado en la Facultad de Ingeniería de la Universidad Autónoma de Querétaro. Estos datos son almacenados y presentados por al usuario a través de un portal web, en el que se muestran las gráficas de comportamiento diarias y en donde se pueden descargar historiales en formato de archivo de texto. Arroyo (2010), plantearon una propuesta para controlar el pH y la temperatura del proceso de la elaboración de yogurt artesanal utilizando una red Ethernet. La conexión se realizó mediante un PLC y controladores Ethernet para hacer la medición de pH y temperatura, para el control de las propiedades del yogurt además de hacer los análisis requeridos para llevar acabo el correcto desarrollo del proceso.

Por otro lado a nivel nacional se también se han realizado proyectos relacionados con el monitoreo remoto vía Ethernet. Macías (2007) realizó la implementación de tele-instrumentación con el fin de lograr la interconexión de equipos de nuestro sísmico, llamado SISMO1, con una estación remota mediante una red de alta velocidad vía Ethernet, para monitoreo en tiempo real. Donde el equipo SISMO1 entrega una señal serial con el protocolo RS-232 que es convertida mediante una tarjeta diseñada para este proyecto a Ethernet, la cual alberga un tranceptor de señales. Todo esto para la adquisición, almacenamiento y tráfico de datos de manera remota en tiempo real. Méndez (2009), desarrolló un sistema de comunicación que permite la interacción entre una interfaz basada en un servidor web y un sistema basado en un microcontrolador, dicho sistema utiliza el monitoreo vía Ethernet y el conjunto de protocolos para comunicación en Internet, en donde se cuenta con una base de datos para almacenar las acciones realizadas. Todo esto a través de un microcontrolador. López (2010), desarrolló un sistema para el monitoreo y control a distancia de los parámetros de un interruptor de circuito de distribución de energía eléctrica a través de Internet, realizando su implementación en la tarjeta de desarrollo Easyweb 3, que cuenta con un microcontrolador que gobierna una interfaz para conectarse a una red de área local. El sistema desarrollado consiste en monitorear y controlar un interruptor de distribución desde una página web, desde donde se pueden enviar mandos al interruptor, así como observar su estado en tiempo real.

Por otra parte a nivel internacional se han elaborado proyectos de comunicación Ethernet a través del FPGA.

Mohsenin (2004), diseñó una interfaz para una tarjeta de red Ethernet/PCI (por sus siglas en inglés *Peripheral Component Interconnect*) usando un FPGA. Desarrollando solo la descripción para enviar y recibir paquetes por medio de una capa física PHY (*Physical Layer*), y una capa de control de acceso al medio MAC (por sus siglas en inglés *Media Access Control*) comerciales. Bernspang, (2004), desarrolló una interfaz basada en FPGA usando un controlador Ethernet MAC/PHY externo en el procesador MicroBlaze desarrollado por Xilinx, para este proyecto se adquirió un CS8900A el cual no se logró probar de manera adecuada debido al tiempo de adquisición del controlador.

En el presente proyecto se pretende implementar una interfaz de monitoreo remoto vía Ethernet a un sistema de calidad de la energía desarrollado dentro de la Universidad Autónoma de Querétaro, esto debido a que la calidad de la energía es un tema que cada vez toma más importancia en la industria y en los hogares. Aunque, comercialmente ya existen sistemas de monitoreo de calidad de la energía en caso de las industrias o medidores inteligentes en el caso de los hogares que ofrecen monitoreo remoto y continuo, son relativamente caros, cuentan con tecnología extranjera y además en algunos casos este tipo de aparatos se limitan a la medición de consumo energético. Debido a esto se pretende tener un sistema con las ventajas de monitoreo remoto que actualmente ofrecen los dispositivos comerciales en un sistema de bajo costo y con tecnología propia.

1.2 Justificación

En los últimos años la importancia del monitoreo de calidad de la energía ha crecido de manera vertiginosa, debido al incremento de uso de cargas no lineales como: motores de corriente alterna, equipo electrónico, fuentes conmutadas, etc. Por ello es necesario contar con sistemas de monitoreo que sirvan como herramienta al usuario para la prevención de algunas fallas eléctricas. Por otra parte, el uso de energías alternativas ha incrementado y

para ello surge la necesidad de tener una distribución adecuada de cargas eléctricas para así tener un mejor aprovechamiento, y garantizar el uso adecuado de este tipo de sistemas. Por esta razón, se requiere de un monitoreo continuo para obtener un análisis completo de cualquiera de los dos sistemas. Por otra parte, el monitoreo remoto en tiempo real con sistemas de medición continua resulta una herramienta muy útil para el usuario, ya que le permite acceso a las mediciones realizadas desde cualquier parte con acceso Internet o a la red de área local donde se encuentre el sistema conectado.

En la Universidad Autónoma de Querétaro, se ha desarrollado cámara de crecimiento con el objetivo de contar con un sistema de tecnología propia que cumpla con las exigencias que su aplicación demanda, una de las características de este sistema es que cuenta con dos tipos de alimentación: suministro de red eléctrica de corriente alterna y suministro de energía solar renovable de corriente continua. Si bien las variables de relativa importancia en este tipo de sistemas son las climatológicas como: temperatura, humedad e iluminación, el estudio de calidad de la energía es parte importante y a la vez necesaria, ya que es uno de los primeros sistemas desarrollado dentro de la universidad que cuenta suministro a través de energías alternativas, y además deberá contar con un funcionamiento ininterrumpido, es de suma importancia conocer si la distribución de cargas eléctricas dentro del sistema es la adecuada para el funcionamiento y además conocer algunos parámetros eléctricos de corriente alterna.

Dentro de la universidad también se ha desarrollado un sistema de monitoreo de calidad de la energía de baja tensión basado en FPGA, que cuenta con sensores de corriente, tensión y una interfaz USB para transferencia de datos que posteriormente son interpretados por el usuario. La intención de este proyecto es desarrollar una interfaz Ethernet e integrarla al sistema de monitoreo, con la intención de realizar monitoreo continuo, de manera remota vía Ethernet.

1.3 Objetivo

Realizar la integración de una interfaz Ethernet al sistema de monitoreo de calidad basado en FPGA, por medio de un controlador Ethernet comercial y un microcontrolador encargado de manejar la comunicación entre el sistema de monitoreo de calidad de la energía y el controlador Ethernet. Así también, realizar pruebas de monitoreo continuo en fitotrón con la finalidad de verificar la distribución de las cargas eléctricas en la parte suministrada por energías alternativas y obtener índices de calidad de la energía en corriente alterna en la parte suministrada por corriente alterna, además analizar las ventajas y desventajas que ofrece trabajar con servidores web embebidos.

1.3.1 Objetivos Particulares

- 1 Seleccionar un microcontrolador y chip Ethernet, a partir de las características mínimas requeridas para la interacción FPGA-microcontrolador, para adquirir los dispositivos adecuados, que se ajusten a las necesidades.
- 2 Diseñar e implementar la página web embebida en el microcontrolador, a partir de las librerías que ofrece el fabricante para el uso del chip Ethernet, haciendo uso de herramientas y lenguajes de programación necesarios para dicha implementación, esta página web embebida servirá como medio de transferencia para los datos leídos del FPGA, a la interfaz de usuario.
- 3 Realizar prueba de conexión punto a punto, a través de la conexión directa a una computadora a través del puerto RJ-45, para verificar el funcionamiento del chip Ethernet.
- 4 Realizar pruebas de envío y recepción de datos a interfaz Ethernet a través de la página web embebida, con el objetivo conocer la programación necesaria para el envío y recepción en la página web embebida que servirá como base para mostrar los parámetros adquiridos del sistema de monitoreo.

- 5 Desarrollar una interfaz de usuario de sistema de monitoreo de calidad de la energía, la cual mostrara los índices de calidad de energía (voltaje y corriente RMS, potencia real, reactiva y aparente, distorsión total armónica y factor de potencia), que se actualizará continuamente para el monitoreo en tiempo real del sistema.
- 6 Implementar la comunicación serial entre FPGA y microcontrolador, por medio de protocolo RS-232, para recibir los datos adquiridos por el sistema y posteriormente mostrarlos en la página web.
- 7 Integrar el sistema de monitoreo de calidad de la energía con una interfaz Ethernet por medio del bloque de comunicación serial implementado en el FPGA.
- 8 Probar el monitoreo de calidad de la energía en Fitotrón, realizando monitoreo vía Ethernet en las partes alimentadas con corriente alterna y en partes de corriente continua, con la finalidad de obtener las ventajas, desventajas y posibles mejoras que puedan realizarse en trabajos futuros relacionados con el monitoreo remoto.

1.4 Planteamiento General

El Planteamiento general de este trabajo es la implementación de interfaz Ethernet a sistema de monitoreo de calidad de la energía, con la finalidad de obtener un sistema de monitoreo remoto accesible desde una red local vía Ethernet o en su defecto desde internet. En el diagrama a bloques de la figura 1.1 se muestran los componentes a utilizar, en primer lugar la cámara de cultivo Fitotrón, que cuenta con dos suministros de alimentación eléctrica, el sistema de monitoreo basado en FPGA y la interfaz de comunicación vía Ethernet.

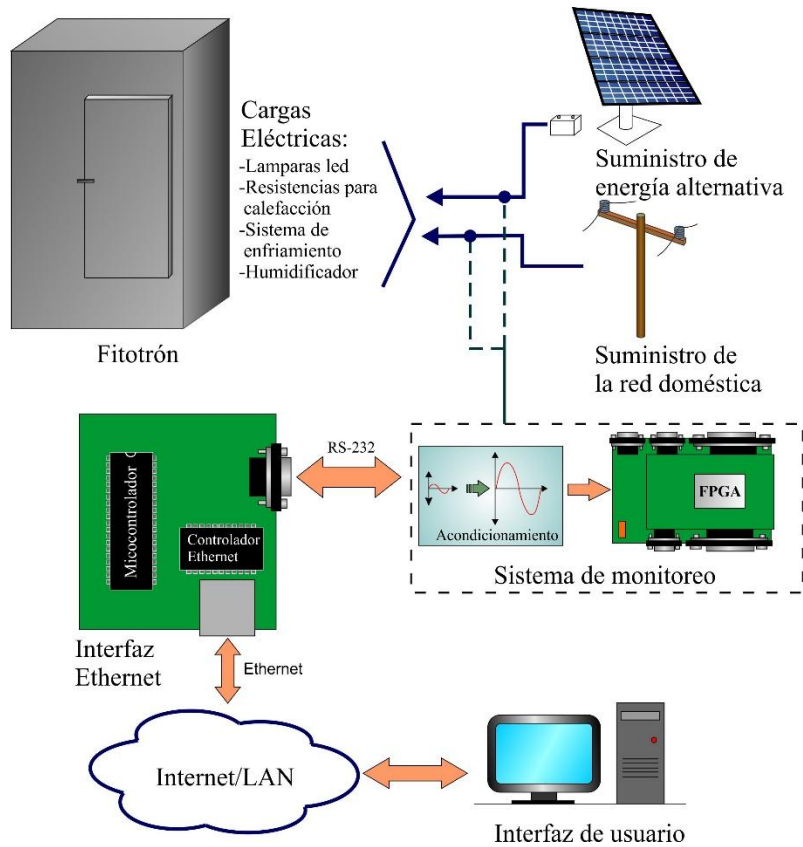


Figura 1.1: Diagrama a bloques de sistema de monitoreo de calidad de la energía.

En la Figura 1.2 se muestran los bloques con los que contará la interfaz a implementar.

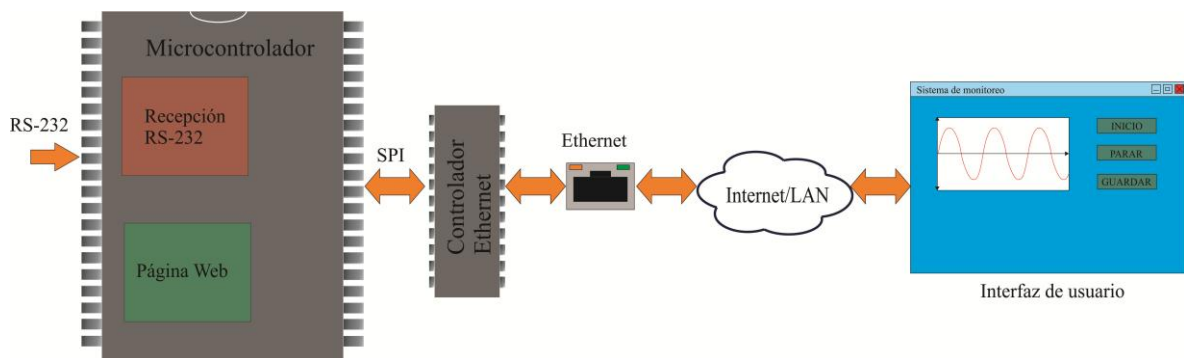


Figura 1.2: Diagrama a bloques de interfaz Ethernet.

- **Microcontrolador:** Encargado de realizar la adquisición de datos, contener una página web embebida, realizar la comunicación vía Ethernet a través del controlador Ethernet.
- **Comunicación microcontrolador-FPGA:** Su función es realizar la adquisición las mediciones realizadas por el sistema de monitoreo a través de comunicación serial RS-232.
- **Página web:** Se encarga de enviar activación y desactivación del sistema de monitoreo así también de mostrar los datos adquiridos por cada tiempo de adquisición.
- **Comunicación controlador Ethernet:** Encargado de llevar a cabo los comandos necesarios para el funcionamiento del controlador Ethernet, por medio de un protocolo de interfaz de periférico serie SPI.
- **Chip Ethernet:** Este bloque contiene la capa física y la capa de dirección MAC, será el encargado de establecer comunicación vía Ethernet.

Capítulo 2

Estado del arte

De acuerdo con [Axelon \(2003\)](#), Ethernet es una tecnología usada en casi todos los hogares y oficinas que habilita el compartimiento de información, y por mucho tiempo los sistemas embebidos y las redes Ethernet habrían existido en mundos distintos. Debido a que Ethernet era una tecnología disponible solo para computadoras los sistemas embebidos que requerían de un intercambio de información con otras computadoras eran limitados a interfaces de baja velocidad, con un rango limitado y falta de protocolos estandarizados. Pero desarrollos en la tecnología y en el mercado hacen posible la comunicación de sistemas embebidos en redes Ethernet locales y también en Internet. Donde la comunicación a través de redes puede hacer de un sistema embebido un sistema más poderoso y fácil de controlar y monitorear.

2.1 Ethernet

[Tanenbaum \(2003\)](#), describe el surgimiento de Ethernet, en la actualidad conocida como el estándar IEEE 802.3. Las redes Ethernet surgieron gracias al trabajo de Bob Metcalfe quien se basó en una de las primeras redes locales o LAN (por sus siglas en inglés *Local Area Network*) de la historia conocida como ALOHANET, dando como resultado notables mejoras en cuanto a velocidad de transmisión (inicialmente a 2.94 Mbps), pudiendo contener hasta

256 ordenadores y transmitiendo hasta 2.5km a través de cable coaxial (con repetidores a cada 500m), además de que cada computadora tenía que escuchar el cable para ver si había alguien más transmitiendo, lo que evitaba interferir con las transmisiones existentes, dando mayor eficiencia. Posteriormente, fueron surgiendo nuevas versiones de Ethernet como el llamado estándar DIX, con una velocidad de 10Mbps, hasta 1983 donde se convirtió en el estándar IEEE 802.3. Ethernet continuó su desarrollo y aun esta en desarrollo. Han salido nuevas versiones a 100 y 1000Mbps, e incluso más altas. También se ha mejorado en el cableado y se han agregado conmutación y otras características.

2.1.1 Componentes de una red

Para tener una idea más clara de cómo se lleva a cabo la comunicación entre computadoras a través de redes se muestra en la figura 2.1 que muestra las partes básicas de una red.

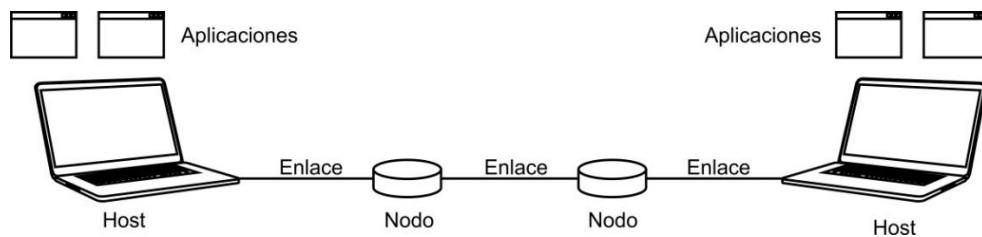


Figura 2.1. Componentes de una red.

- Aplicación: Se refiere al programa que usa la red.
- Host: Se encarga de soportar las aplicaciones.
- Nodos: Se encargan de retransmitir o pasar los mensajes.
- Enlace: Se encarga de conectar los nodos.

2.1.2 Cableado Ethernet

Dado que el nombre Ethernet se refiere al medio de transmisión a través de cable en la tabla 2.1 se muestran los cuatro tipos de cableado comúnmente utilizados:

Tabla 2.1. Tipos de cableado Ethernet más comunes.

Nombre	Cable	Seg. max.	Nodos/seg	Ventajas
10BASE5	Coaxial grueso	500 m	100	Cable original; ahora obsoleto
10BASE2	Coaxial delgado	185 m	30	No se necesita concentrador
10BASE-T	Par trenzado	100 m	1024	Sistema más económico
10BASE-F	Fibra óptica	200 m	1024	Mejor entre edificios

Para comprender la notación que se le da al nombre los primeros números indican la velocidad en Mbps (Mega bits por segundo), después de esto viene la palabra “Base” (o algunas veces “BASE”) para indicar transmisión de banda base. Por último si el medio es coaxial, su longitud máxima se redondea a unidades de 100 metros después de “Base”.

2.1.3 Encapsulación

La encapsulación es el mecanismo usado en efecto de los protocolos de cada capa, donde cada capa agrega su propia información para hacer una nueva entrega de mensaje. En la figura 2.2 se muestra un ejemplo de cómo se lleva el proceso de encapsulación, en este ejemplo la capa superior se conoce como la capa de aplicación donde a partir del protocolo que se utiliza en esta capa se genera un mensaje que es enviado a través de las demás capas que se encargan de añadir información que servirá para la transferencia de este mensaje, la última capa se encarga de transferir el paquete creado por las capas superiores. Al llegar el paquete a su destino el paquete atraviesa las mismas capas que se utilizaron para su encapsulación y así elimina la información añadida por cada protocolo, llegando a la capa de aplicación del Host 2 solo la información de la capa de aplicación del Host 1. La

encapsulación permite que los mensajes enviados lleven consigo información con la cual se asegura que los mensajes lleguen de manera correcta.

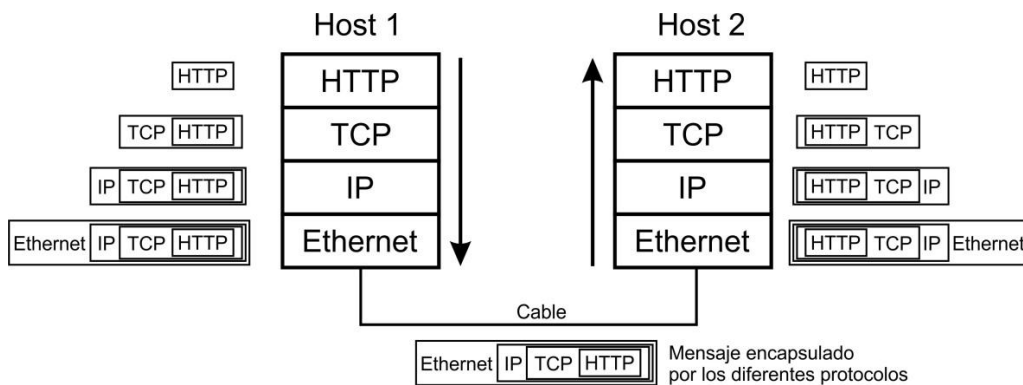


Figura 2.2. Ejemplo de encapsulación.

2.2 Protocolos y capas

Debido al gran número de operaciones que se realizan en las redes de computadoras, como establecer y finalizar conexiones, encontrar rutas de envío a través de la red, transferir información de manera segura, enviar tan rápido como la red lo permita, compartir el ancho de banda entre los usuarios, permitir que nuevos host sean agregados entre otras operaciones, se requiere de un diseño modular, el cual se compone de capas y protocolos. Los cuales son el principal método estructural para manejar las funcionalidades de una red. En donde cada instancia de un protocolo habla individual mente con su igual por medio de un protocolo, como muestra la figura 2.3 los protocolos se encuentran de manera horizontal y las capas de manera vertical. Algunos ejemplos de protocolos pueden ser: TCP (protocolo de control de transmisión por sus siglas en inglés *Transmission Control Protocol*), IP (protocolo de Internet por sus siglas en inglés *Internet Protocol*), IEEE 802.11, Ethernet, HTTP (protocolo de transferencia de hipertexto por sus siglas en inglés *Hypertext Transfer Protocol*), DNS (sistema de nombres de dominio por sus siglas en inglés *Domain Name System*), etc.

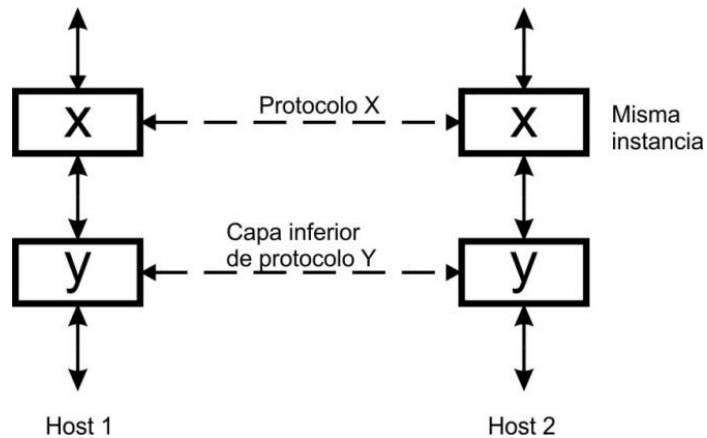


Figura 2.3. Protocolos y capas.

2.2.1 Modelo de referencia OSI

Una pregunta clave en el diseño basado en capas es, ¿Qué funcionalidad debe tener cada capa? Para esto existen modelos de referencia que proporcionan la estructura que se debe tener. El modelo de referencia OSI (interconexión de sistemas abiertos por sus siglas en ingles *Open System Interconnection*) mostrado en la figura 2.4 de siete capas, en un principio un estándar internacional para conectar sistemas, fue de gran influencia pero nunca se usó en la práctica.



Figura 2.4. Modelo de referencia OSI de siete capas.

Cada capa posee una funcionalidad específica como se muestra a continuación:

Capa de aplicación: Proporciona las funciones requeridas por el usuario.

Capa de presentación: Convierte diferentes representaciones.

Capa de sesión: Administra diálogos

Capa de transporte: Proporciona entregas de extremo a extremo de la red.

Capa de red: Envía paquetes en enlaces múltiples.

Capa de enlace de datos: Envía tramas de información.

Capa física: Envía los bits en forma de señal.

2.2.2 Modelo TCP/IP

Se le denomina modelo TCP/IP en referencia a los dos protocolos más importantes que la componen: TCP e IP, que fueron dos de los primeros en definirse, y que son los más utilizados de la familia, aunque actualmente existen tantos protocolos en este conjunto que pueden llegar a ser más de 100 diferentes. El modelo TCP/IP es la base de Internet y sirve para enlazar computadoras que utilizan diferentes sistemas operativos, incluyendo PC, minicomputadoras y computadoras centrales sobre redes LAN y WAN (redes de área extensa por sus siglas en inglés Wide Area Network) en la figura 2.5 se muestra el modelo de cuatro capas TCP/IP y algunos ejemplos de los protocolos que maneja cada capa.

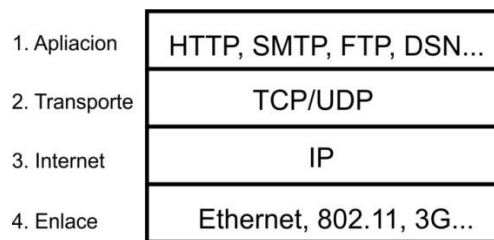


Figura 2.5. Modelo de referencia TCP/IP.

Capa de aplicación: Donde se encuentran los programas que usan los servicios de red.

Capa de transporte: Proporciona entregas extremo a extremo de los datos.

Capa de Internet: Envía paquetes sobre múltiples redes.

Capa enlace: Envía los marcos de datos en un enlace.

2.3 Servidor web

El servidor web es un programa que está esperando permanentemente las solicitudes de conexión mediante el protocolo HTTP por parte de los clientes web, (Lujan, 2001).

La parte del servidor de las aplicaciones web está formada por:

- Páginas estáticas (documentos HTML) que siempre muestran el mismo contenido
- Recursos adicionales que se pueden emplear dentro de las páginas o estar disponibles para ser descargados y ejecutados en la parte del cliente
- Programas o script que son ejecutados por el servidor web cuando el navegador del cliente solicita algunas páginas. Tradicionalmente este programa o script que es ejecutado se basa en tecnología de interfaz de entrada común (CGI por sus siglas en inglés *Common Gateway Interface*).

2.4 Página web

Es el nombre de un documento o información adaptada para la red informática mundial y puede ser accedida mediante un navegador. Esta información se encuentra en formato HTML o XHTML (lenguaje de marcado de hipertexto por sus siglas en inglés HyperText Markup Language), y puede proporcionar navegación a otras páginas web mediante enlaces de hipertexto. Una página web está compuesta principalmente por información (solo texto y/o multimedia) así como por hiperenlaces; además puede contener o asociar una hoja de estilo, datos de estilo para especificar como debe visualizarse, y también aplicación informática como aplicaciones embebidas para así hacerla interactiva. Las páginas web son escritas en un lenguaje marcado que provee la capacidad de manejar e insertar hiperenlaces,

generalmente HTML. El contenido de la página puede ser predeterminado página web estática o generado al momento de visualizarla o solicitarla a un servidor web página web dinámica. Las paginas dinámicas que se generan al momento de visualización, se especifican a través de algún lenguaje interpretado, y la aplicación encargada de visualizar el contenido es la que realmente debe generarlo. Las páginas dinámicas que se generan al ser solicitadas, son creadas por una aplicación en el servidor web que alberga a las mismas, (Lujan, 2001).

2.4.1 Lenguaje HTML

HTML, hace referencia al “lenguaje de marcado” predominante para la elaboración de “páginas web” que se utiliza para describir, traducir la estructura y la información en forma de texto, así como para complementar el texto con objetos tales como imágenes. El HTML se escribe en forma de “etiquetas”, rodeadas por “corchetes angulares” (<,>). HTML también puede describir, hasta un cierto punto, la apariencia de un documento, y puede incluir un “script”(por ejemplo JavaScript), el cual puede afectar el comportamiento de navegadores web y otros procesadores de HTML.

2.5 Interfaz de usuario

La interacción entre las personas y las computadoras es realizada principalmente a través de una interfaz gráfica de usuario, un tipo de interfaz de usuario, compuesto por metáforas graficas inscritas en una superficie de contacto en adición de otros elementos semánticos necesarios para posibilitar dicha interacción con los signos-objetos en la interfaz gráfica.

2.5.1 Definición de interfaz gráfica de usuario

La interfaz gráfica de usuario GUI (por sus siglas en inglés *Graphical User Interface*) es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes, objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso es proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una computadora.

2.6 Microcontrolador

Angulo (2003), menciona que un microcontrolador es un computador completo, aunque con limitadas prestaciones, que está contenido en el chip de un circuito integrado y se destina a gobernar una sola tarea. Las partes principales de un microcontrolador son:

1. Procesador
2. Memoria no volátil para contener el programa
3. Memoria de lectura y escritura para guardar los datos
4. Líneas de E/S para controladores de periféricos:
 - a) Comunicación paralelo
 - b) Comunicación serie
 - c) Diversas puertas de comunicación
5. Recursos auxiliares:
 - a) Circuito de reloj
 - b) Temporizadores
 - c) Perro guardián (watchdog)
 - d) Conversores AD y DA
 - e) Comparadores Analógicos
 - f) Protección ante fallos de alimentación

- g) Estado de reposo o de bajo consumo

2.6.1 Periféricos

Un periférico es un dispositivo electrónico especial, incluido en un microcontrolador que realiza una función o tarea específica. Los periféricos de un microcontrolador permiten ahorro de espacio y dinero en muchas aplicaciones electrónicas.

2.6.1.1 Interfaz UART

La interfaz UART (transmisor-receptor asíncrono universal por sus siglas en inglés *Universal Asynchronous Receiver-Transmitter*) permite al microcontrolador establecer comunicaciones serial RS-232 con otros dispositivos. Debido a que la comunicación serial es asíncrona no depende de una señal de reloj maestra entre el transmisor y receptor.

2.6.1.2 Interfaz SPI

El protocolo SPI (interfaz periférica serial por sus siglas en inglés *Serial Peripheral Interface*) es un protocolo de comunicación serie síncrono desarrollado por Motorola y, más tarde, adoptado por una gran mayoría de fabricantes de circuitos integrados. Permite la comunicación serie entre microcontroladores y varios dispositivos externos de alta velocidad (hasta 180MHz). El protocolo no dispone de especificaciones oficiales y no está regulado por ningún comité internacional, esto da cierta flexibilidad al crear un protocolo SPI. La conexión

del hardware es relativamente simple puede tener 3 o más líneas, dependiendo de si solo hay un esclavo o más, o si el esclavo puede recibir y enviar datos o solo una de las dos.

Las líneas de configuración más completas son:

- CS (por sus siglas en inglés *Chip Select*), permite seleccionar el dispositivo esclavo con el que el microcontrolador se quiere comunicar.
- SCK o SCLK (por sus siglas en inglés *Serial Clock*), La línea de reloj generada por el maestro para la sincronizar los datos.
- SDI (por sus siglas en inglés *Serial Data In*). Salida de datos del dispositivo maestro, y entrada de datos al dispositivo esclavo
- SDO (por sus siglas en inglés *Serial Data Out*). Salida de datos del dispositivo esclavo y entrada de datos al dispositivo maestro.

2.7 Arduino

Arduino es una plataforma de electrónica abierta para la creación de prototipos basada en software y hardware flexibles y fáciles de usar. El microcontrolador en la placa Arduino se programa mediante lenguaje de programación Arduino (basado en *Wiring*), y entorno de desarrollo Arduino (basado en *Processing*). Las placas pueden ser hechas a mano o montadas de fábrica; el software puede ser descargado de forma gratuita. Los ficheros de diseño de referencia CAD (por sus siglas en inglés *Computer Asisted Desing*) están disponibles bajo licencia abierta. El hardware consiste en una placa con un microcontrolador Atmel AVR y puertos de entrada/salida. Los microcontroladores más usados son el Atmega168, Atmega328, Atmega1280, Atmega8 por su sencillez y bajo costo permiten el desarrollo de múltiples diseños, Por otro lado es software consiste en un entorno de desarrollo que implementa el lenguaje de programación *Processing/Wiring* y el cargador de arranque (*bootloader*) que corre en la placa. Al ser open-hardware, tanto su diseño como su distribución son libres. Es decir, puede utilizarse libremente para el desarrollo de cualquier tipo de proyecto sin haber adquirido ninguna licencia, (Smith, 2006).

2.8 Chip Ethernet

En la actualidad existe una gran variedad de circuitos integrados Ethernet, en el presente proyecto se selecciona el chip Ethernet W5100 de WIZnet, el cual cuenta la capa de transporte, internet y enlace, soporta los protocolos: TCP, UDP, ICMP, IPv4 ARP, IGMP, PPPoE, Ethernet. Soporta conexiones 10BASE-T y 100BASE-TX, además de soporta 4 sockets de manera simultánea, Para su control cuenta con una interfaz de comunicación SPI, En la figura 2.6 se muestra el diagrama a bloques del chip W5100, [WIZnet \(2009\)](#).

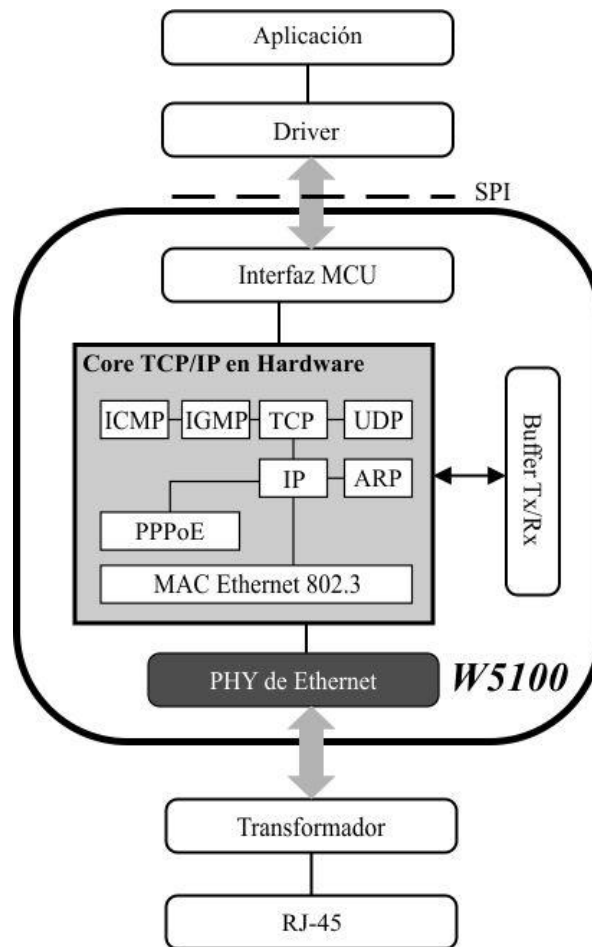


Figura 2.6. Diagrama a bloques W5100.

2.9 Calidad de la energía

Tanto las empresas suministradoras de energía eléctrica como los usuarios finales del servicio de energía eléctrica, han estado insistiendo cada vez más en el concepto de calidad de la energía. Esto se inició en la década de los 80's y se ha convertido en una especie de concepto general, alrededor del cual se puede ubicar una multitud de distintos disturbios y problemas que se pueden presentar en un sistema eléctrico, (Enríquez, 1999).

De acuerdo con Enríquez (1999), las principales razones por las que es necesario estudiar los conceptos relacionados con la calidad en suministro de la energía eléctrica son las siguientes:

1. Las cargas cada día son más sensibles a las variaciones de ciertos tipos de parámetros o cantidades en los sistemas del suministro de energía eléctrica, esto debido al aumento de cargas tanto industriales como residenciales y comerciales basados tecnología digital, como lo es la robótica, las computadoras personales, aparatos del hogar, etc.
2. Se ha incrementado el concepto de mayor eficiencia en los sistemas eléctricos, lo cual ha traído como resultado un incremento continuo en la aplicación de dispositivos de alta eficiencia, tales como: controladores de velocidad en motores eléctricos, el uso de capacitores en paralelo para la corrección de factor de potencia y reducir pérdidas, esto trae como consecuencia un incremento en los niveles de armónicas en los sistemas eléctricos.
3. Una mayor atención por parte de los usuarios finales a problemas con la calidad de suministro de energía eléctrica, que pueden afectar a las cargas como son: las interrupciones de servicio, los transitorios por maniobras, las depresiones y elevaciones de voltaje, etc.
4. La creciente tendencia a la interconexión de los sistemas eléctricos al nivel de sistemas de potencia y de instalaciones industriales, trae como resultado mayor cantidad de procesos integrados lo cual significa que una falla en cualquier componente tiene consecuencias más importantes.

2.9.1 Índices de calidad de la energía en corriente alterna

Los índices de calidad de la energía eléctrica en corriente alterna de primordial importancia:

Valor pico de voltaje y corriente

El valor pico de un voltaje o corriente es su valor máximo con respecto a cero.

Valor eficaz de voltaje y corriente

Un valor eficaz es un valor de corriente directa equivalente: expresa cuantos volts o amperes de corriente directa equivalen a una forma directa que varía con el tiempo en términos para producir potencia promedio, para obtener el valor eficaz o rms (valor cuadrático medio por sus siglas en inglés *Root Mean Square*) de una onda sinusoidal se realiza la operación de mostrada en la ecuación 1.

$$V_{rms} = \frac{V_m}{\sqrt{2}} = 0.707 * V_m$$

Donde V_m es el valor medio o valor pico de la corriente o voltaje.

Potencia aparente

La potencia aparente se llama así porque aparentemente la potencia debería ser el producto voltaje-corriente, por analogía con los circuitos resistivos de corriente continua. Esta potencia se mide en volt-amperes o V_a para distinguirla de la potencia promedio o real, la cual se mide en watts. La potencia aparente está definida como el producto de los valores rms de voltaje y corriente.

$$|S| = V_{rms} I_{rms}$$

Potencia real

Es la potencia capaz de transformar la energía eléctrica en trabajo. Los diferentes dispositivos eléctricos existentes convierten la energía eléctrica en otras formas de energía

tales como: mecánica, térmica, química, etc. Esta potencia es, por lo tanto, la realmente consumida por los circuitos y, en consecuencia, cuando se habla de demanda eléctrica, es esta potencia la que se utiliza para determinar dicha demanda y está definida como:

$$P = V_{rms}I_{rms}\cos\phi$$

Donde ϕ es el ángulo entre la potencia real P y el valor absoluto de la aparente S.

Potencia reactiva

La potencia reactiva Q es la componente imaginaria de la potencia compleja y constituye una medida de la tasa de flujo de energía hacia o desde las componentes reactivas de su carga; su unidad de medida es volt-ampere-reactivo (VAR), y está definida como:

$$Q = V_{rms}I_{rms}\sen\phi$$

Donde ϕ es el ángulo entre la potencia real P y el valor absoluto de la aparente S.

Factor de potencia

Se define factor de potencia, f.p., de un circuito de corriente alterna, como la relación entre la potencia activa, P, y la potencia aparente, S. Da una medida de la capacidad de una carga de absorber potencia real. Por esta razón el factor de potencia es igual a la unidad en cargas puramente resistivas en elementos inductivos y capacitivos ideales sin resistencia el factor de potencia es igual a cero. Se define el factor de potencia como:

$$\text{f. p.} = \frac{P}{|S|} = \cos \phi$$

Donde ϕ es el ángulo entre la potencia real P y el valor absoluto de la aparente S.

Distorsión total armónica THD

La distorsión armónica total o THD (por sus siglas en ingles *Total Harmonic Distortion*) de una señal es una medida de señales con frecuencias múltiples a la frecuencia

fundamental de la señal original. A la señal original se le conoce como primera armónica se llama frecuencia fundamental. Dos por la frecuencia original de la señal es igual a la segunda armónica, tres origina la tercera, etcétera. La distorsión armónica total es la amplitud rms combinada de las armónicas superiores, dividida entre la amplitud rms de la frecuencia fundamental. La ecuación matemática de la distorsión armónica total es.

$$\%THD = \frac{\sqrt{V_2^2 + V_3^2 + \dots + V_n^2}}{V_{fundamental}} \times 100$$

Una herramienta útil para obtener el valor de distorsión armónica total es la transformada de Fourier que determina la magnitud y fase de cada componente de frecuencia de una señal. La composición de frecuencias de una señal, expresada por la transformada de Fourier se llama espectro de frecuencias de la señal. Ese espectro de frecuencias de onda se puede graficar con las frecuencias armónicas en el eje de las abscisas y sus amplitudes en el eje de las ordenadas.

2.9.2 Índices de calidad de la energía en corriente continúa

En cuanto a corriente directa se requieren menos parámetros ya que en este proyecto se busca evaluar consumo en la parte de corriente directa que es suministrada por medio de energía alternativa, los índices que se requieren en este caso son:

Voltaje

Es una magnitud física que cuantifica la diferencia de potencial eléctrico entre dos puntos. También se puede definir como el trabajo por unidad de carga ejercido por el campo eléctrico sobre una partícula cargada para moverla entre dos posiciones determinadas.

Corriente

Es el flujo de carga por unidad de tiempo que recorre un material. Se debe al movimiento de los electrones en el interior del material. En el Sistema Internacional de Unidades se expresa en C/s (culombios sobre segundo), unidad que se denomina amperio.

Potencia

Cuando se trata de corriente continua la potencia eléctrica desarrollada en un cierto instante por un dispositivo de dos terminales, es el producto de la diferencia de potencial entre dichos terminales y la intensidad de corriente que pasa a través del dispositivo. Por esta razón la potencia es proporcional a la corriente y a la tensión. Esto es,

$$P = V * I \quad (3)$$

2.10 Herramientas de diseño

2.10.1 IDE de Arduino

El entorno de desarrollo integrado IDE (por sus siglas en inglés *Integrated Development Environment*), contiene un editor de texto para la escritura de texto, área de mensajes, a una consola de texto, barra de herramientas con botones para funciones comunes, y una serie de menús, Este se conecta con cualquier tarjeta Arduino para cargar los programas y comunicarse con la tarjeta a través de un monitor serial. La escritura en software usando Arduino son llamados bocetos. Estos bocetos son escritos en el editor de texto y se guardan con una extensión *.ino*. Este software de Arduino tiene soporte para Windows, Mac y Linux y puede ser descargado de la página oficial de Arduino, en la figura 2.7 se muestra el entorno del IDE de Arduino, el cual cuenta con algunas librerías y ejemplos para el desarrollo o edición de proyectos, (Margolis, 2011).

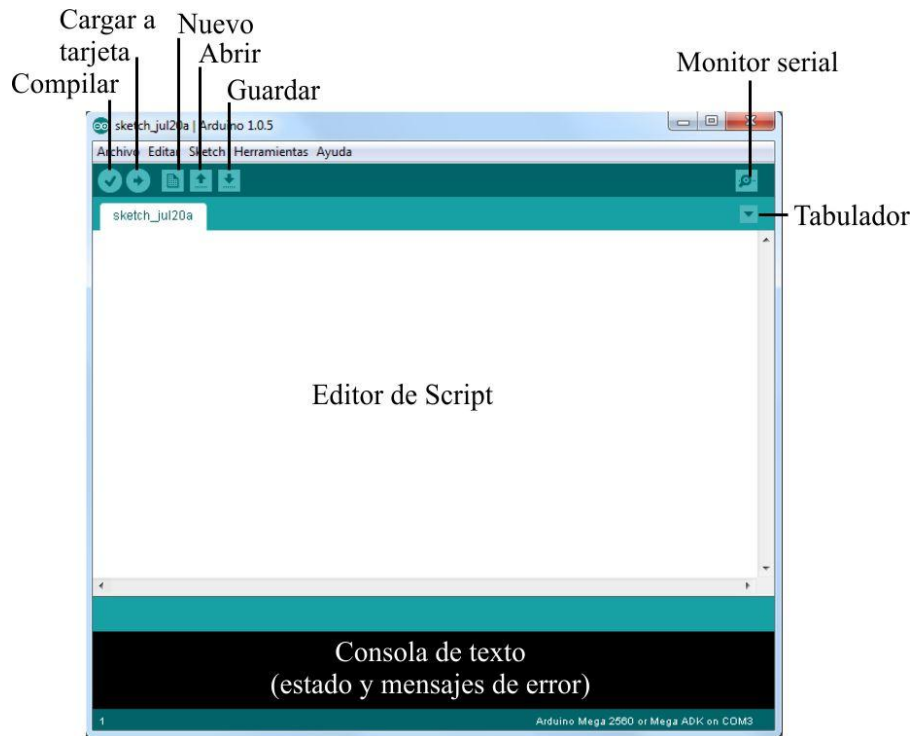


Figura 2.7. IDE de Arduino.

La plataforma Arduino se programa mediante el uso de un lenguaje propio basado en el popular lenguaje de programación de alto nivel *Processing*. Sin embargo, es posible utilizar otros lenguajes de programación populares en Arduino. Algunos ejemplos son:

- Java
- Flas (Mediante ActionScript)
- Pure Data
- MaxMSP (entorno grafico de programación para aplicaciones musicales, de audio y multimedia)
- VVVV (Síntesis de video en tiempo real)
- Python
- C
- C++
- C#
- Cocoa/Objective-C
- Linux TTY

- 3DVIA Virtools (aplicaciones interactivas y de tiempo real)
- SuperCollider
- Matlab
- Mathematica
- Isadora (Interactividad audiovisual en tiempo real)
- Visual Basic .NET
- Gambas
- Php

Esto es posible debido a que Arduino se comunica mediante la transmisión de datos en formato serie que es algo que la mayoría de los lenguajes anteriormente citados soportan. Para los que no soportan el formato serie de forma nativa, es posible utilizar software intermediario que traduzca los mensajes enviados por ambas partes para permitir la comunicación fluida. Arduino está basado en C y soporta todas las funciones del estándar C y algunas de C++.

2.10.2 Matlab GUI

La primera versión de Matlab fue desarrollada en 1977 por Cleve Moler, diseñado para resolver sistemas de ecuaciones lineales y desempeñar cálculos con matrices. En ese tiempo Matlab fue una aplicación relativamente simple con una línea de comandos con alrededor de 80 funciones. Entre los años 1980 y 1990 se integraron conjuntos de herramientas con miles de funciones gráficas y a la mitad de 1990 se incorporó la interfaz gráfica de usuario GUI (por sus siglas en inglés *Graphical User Interface*), en donde versiones recientes incluyen GUI's avanzadas con capacidades que incluyen graficas en tiempo real, (Smith, 2006).

Capítulo 3

Metodología

La metodología de este proyecto se muestra en la figura 3.1. Donde se describen todos los componentes y las herramientas necesarias para el desarrollo de la interfaz Ethernet, como:

- Arduino Mega2569
- Ethernet Shield con chip W5100
- Sistema de monitoreo
- Protocolo de comunicación serial
- Sitio web
- Interfaz gráfica de usuario
- PC

Este trabajo se basa en la combinación de un sistema basado en FPGA y un microcontrolador, lo que facilita la integración de la interfaz Ethernet a un sistema basado en FPGA ya que aunque el chip Ethernet cuenta con algunos protocolos desarrollados en hardware, la complejidad para realizar aplicaciones Ethernet como una página web es alta. Por lo tanto, gracias al microcontrolador se puede hacer la integración del sistema de manera sencilla ya que el software arduino cuenta con las librerías necesarias para el manejo del chip Ethernet W5100.

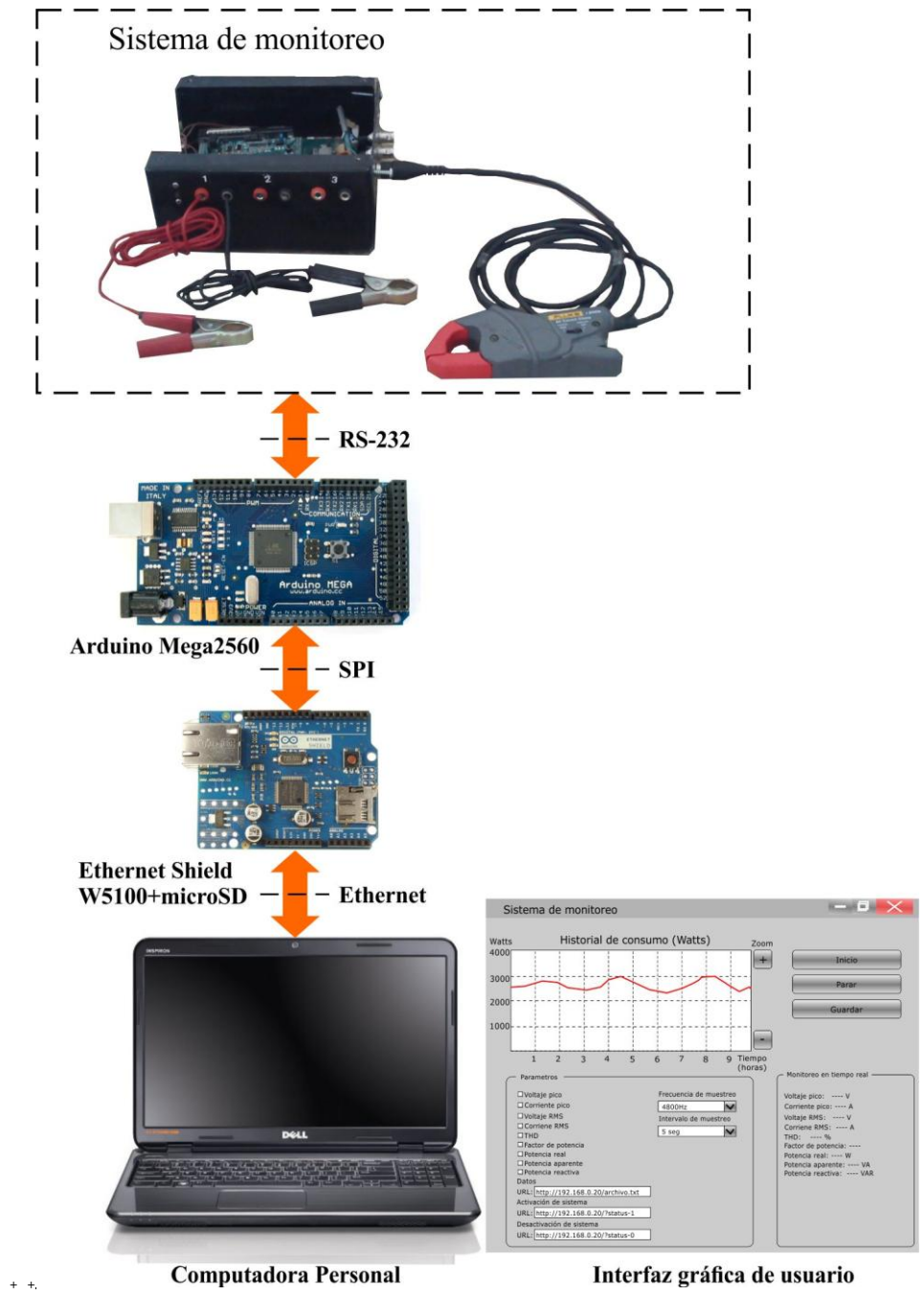


Figura 3.1. Proceso para integración de interfaz Ethernet.

A continuación se presentan las partes en las que se divide la metodología de este trabajo:

1. Implementación de comunicación serial (RS-232) a sistema de monitoreo.
2. Página web embebida
3. Interfaz gráfica de usuario.

3.1 Implementación de comunicación serial a sistema de monitoreo

El sistema de monitoreo, cuenta actualmente con una interfaz USB que le permite hacer monitoreos por un cierto tiempo que el usuario decide y posteriormente es enviado a la computadora, para fines de establecer una comunicación entre el FPGA y la tarjeta Arduino, se implementará en el sistema el módulo en VHDL para comunicación serial, así como bloques de control que se encargen de controlar la comunicación entre FPGA-Arduino. El sistema cuenta con una unidad de gestión de memoria MMU (por sus siglas en inglés *Memory Management Unit*), que se encarga de realizar las lecturas y escrituras en una memoria RAM dinámica, asu vez cuenta con un bloque de control de escritura que como su nombre lo indica se encarga de llevar a cabo de forma correcta la secuencia para la escritura de los datos obtenidos, así también cuenta con el core para el control del convertidor analógico digital ADC (por sus siglas en inglés *Analog to Digital Converter*) de 12bits y 4 canales ADS7841. Para la integración de la comunicación serial se propone la integración de 3 módulos:

1. Módulo de comunicación serial
2. Módulo de control de lectura de memoria RAM
3. Módulo de control principal

En la figura 3.2 se muestra como se encuentra integrado el sistema, asimismo se señalan los bloques existentes y los bloques que serán añadidos al sistema.

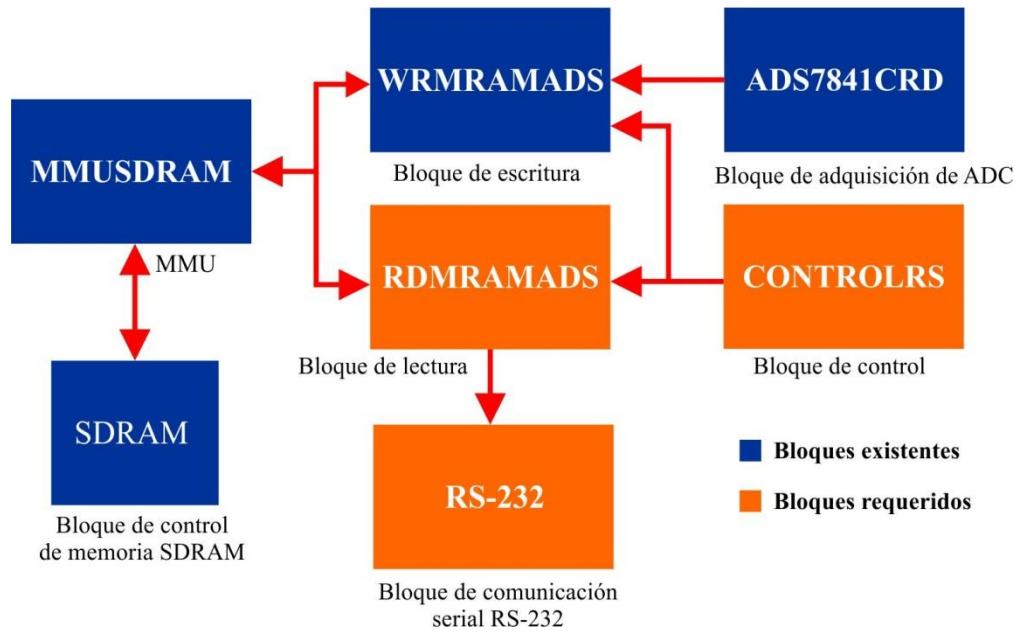


Figura 3.2. Diagrama general de bloques de sistema de monitoreo.

3.1.1 Módulo de comunicación serial

En la figura 3.3 se muestra el diagrama del módulo de comunicación serial, el cual tiene los elementos para transmisión serial, como se vio en el capítulo anterior la comunicación serial soporta datos de 8 bits que son enviados a este módulo a través del bloque de lectura, y a su vez el bloque de lectura se encarga de activar la salida del dato por medio de la entrada “TxValid”, posteriormente el bloque envía una señal que indica que el dato fue transmitido correctamente “TxReady”, cuya señal se usa como salida digital del sistema para indicar que el dato fue enviado, la salida del dato serial es a través de “TxD” que se encarga de enviar el dato de forma serial.

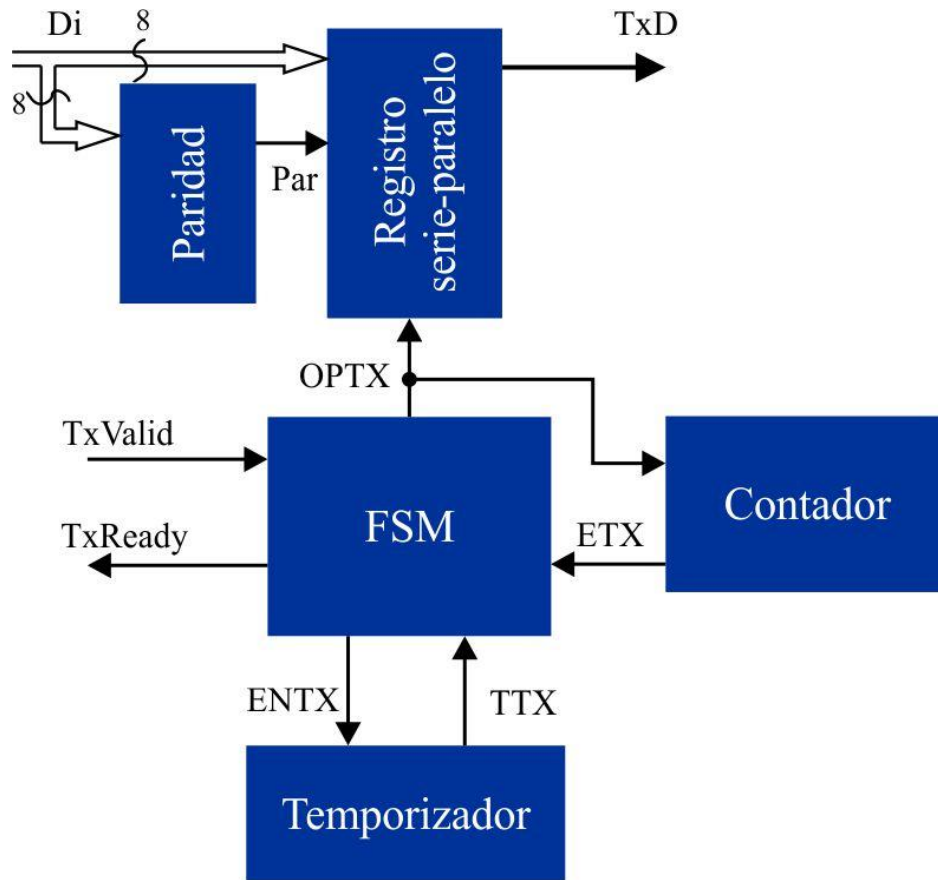


Figura 3.3. Diagrama de bloques de módulo de comunicación serial.

Este módulo está integrado por un temporizador el cual es activado por la máquina de estados cuando un envío se está llevando a cabo su función es mantener la comunicación a una frecuencia estándar de comunicación serial RS-232, la cual por objeto de este proyecto se define en 115200 baudios, por lo tanto la constante de este temporizador está definida por el cociente entre la frecuencia deseada (115200) y la frecuencia de operación del FPGA, en este caso 48MHz, por otra parte se cuenta con un contador cuya función es asegurarse de que se envíen solo 11bits que es el estándar para comunicación serial, el registro paralelo-serie como su nombre lo indica se encarga de transmitir el dato junto con el bit de paridad, las señales de inicio y la señal de fin. La transmisión se llevará a cabo siempre y cuando se mande la activación de la señal “TxValid” y al finalizar el envío devolverá una bandera llamada “TxReady” que indica que el dato fue enviado.

3.1.2 Módulo de lectura de SDRAM

En la figura 3.4 se muestra el diagrama a bloques del módulo de lectura de la SDRAM, el cual se encarga de leer los datos almacenados en la memoria SDRAM y a su vez ordenarlos ya que los datos almacenados de las lecturas son de 12 bits y el envío permitido por el protocolo serial es de 8 bits por lo cual este módulo se encargara de hacer paquetes de 8 bits.

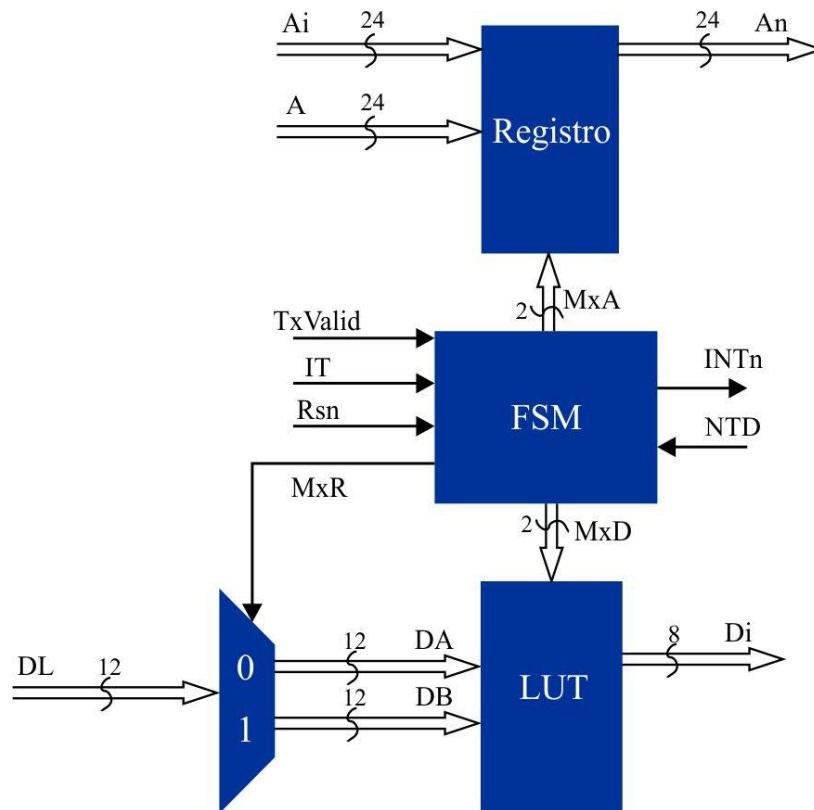


Figura 3.4. Diagrama de bloques de módulo de lectura de SDRAM.

Este módulo cuenta con una máquina de estados encargada de controlar la dirección a acceder a la memoria RAM, así como la adecuada distribución de los datos leídos de la memoria RAM, a su vez cuenta con una activación principal “IT”, la cual es activada cuando el módulo de control termina de hacer las adquisiciones requeridas. La señal “Rsn” es una bandera proveniente del módulo MMU, que es activada cuando un dato fue leído de la

memoria RAM. La máquina de estados también cuenta con la bandera “INTn”, la cual es la activación principal para que el módulo de MMU comienza con una lectura o escritura, finalmente se cuenta con la bandera “NTD” que se activa cuando se han leído las direcciones de la memoria RAM requeridas, finalizando así la lectura de datos de memoria.

3.1.3 Módulo de control principal

El módulo de control principal es el encargado de activar el inicio de muestreo, establecer el periodo de muestreo y número de muestras a adquirir, en este caso se propone una frecuencia de muestreo de 4.8KHz y un número de muestras de 1000 que equivale a 200ms, lo cual es de acuerdo a la Norma Internacional de la Comisión Electrotécnica Internacional IEC (por sus siglas en inglés *International Electrothechnical Commission*) 61000-4-30, la cual establece que para una medición de clase B (usada para estudios estadísticos, aplicaciones para resolución de problemas entre otras aplicaciones) es el intervalo de tiempo de medición básico para magnitudes como: voltaje, armónicos y desbalance. En la figura 3.5 se muestra el diagrama de bloques del módulo de control.

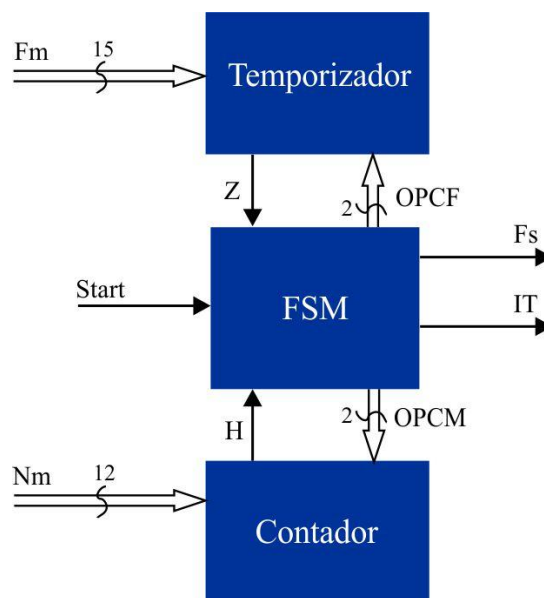


Figura 3.5. Diagrama de bloques de módulo de control.

El módulo de control cuenta con una máquina de estados que se encarga de habilitar la adquisición de datos del ADC mediante la bandera “Fs” la cual es activada dependiendo el periodo de muestreo, a su vez el contador indica cuando el número de muestras llega al valor deseado, activando la señal “IT” que indica el fin de la adquisición, la máquina de estados cuenta con una activación principal Start, la cual será enviada mediante una salida digital de la tarjeta Arduino.

3.2 Página web embebida

El medio que servirá para la comunicación a través de una red LAN o a través de Internet del sistema de calidad de la energía será por medio de una página web embebida en la tarjeta Arduino, la cual será una página web estática encargada de mostrar los datos que fueron adquiridos. Esta página se integrará al sistema por medio de las librerías con las que cuenta Arduino para comunicación Ethernet.

A continuación se enlistan las características principales para la página web:

1. **ACTIVACIÓN DE SISTEMA:** La página web embebida debe permitir la activación del sistema de monitoreo por medio de una salida digital.
2. **MOSTRAR DATOS:** Deberá mostrar los datos adquiridos durante el periodo programado en el FPGA. Debido a que la memoria de la tarjeta Arduino es limitada, los datos serán almacenados en una tarjeta micro-SD con la que cuenta el Ethernet Shield. Los datos serán guardados en la tarjeta micro-SD en un archivo de texto y posteriormente deberán ser mostrados en una página web alterna donde serán descargados por la interfaz gráfica de usuario.

3.3 Interfaz gráfica de usuario

La interfaz gráfica de usuario será la encargada de acceder a la página web embebida mediante una conexión Ethernet local o a través de Internet por medio de la dirección URL (localizador de recurso uniforme por sus siglas en inglés *Uniform Resource Locator*) con la que cuente el dispositivo, la interfaz gráfica de usuario será desarrollada en Matlab y deberá contar con las siguientes características:

1. **ACTIVACIÓN REMOTA DEL SISTEMA:** Capacidad para enviar activación de inicio de monitoreo al sistema de manera remota (a través de una red LAN o Internet).
2. **SELECCIÓN DE PARÁMETROS ELÉCTRICOS:** El usuario debe ser capaz de seleccionar los parámetros eléctricos que serán mostrados y almacenados.
3. **GRAFICAR:** Realizar la gráfica de consumo en tiempo real, calculando el consumo a partir de los parámetros obtenidos del sistema (voltaje y corriente).
4. **CALCULAR PARÁMETROS ELÉCTRICOS:** Realizar los cálculos necesarios para obtener los parámetros eléctricos a mostrar.
5. **MOSTRAR PARÁMETROS ELÉCTRICOS:** Mostrar, en tiempo real, los parámetros previamente seleccionados.
6. **GENERAR HISTORIAL:** Deberá generar un archivo de texto el cual almacenará los parámetros calculados cada intervalo de muestreo, este intervalo de muestreo se establece en la interfaz gráfica y puede ser de 10s, 5min, 10min, y 1 hora. Por otro lado la duración de monitoreo también se establece en la interfaz gráfica donde pide un número y la selección de las unidades de tiempo para la duración ya sea minutos, horas, días etc. Estos datos se almacenan en archivos los cuales una vez que termina la adquisición los datos anteriores se conservan y solo adjunta los datos en caso de que otra medición se lleve a cabo. En otro caso se puede mover los archivos a otra carpeta para su posterior análisis y automáticamente se crea un nuevo archivo.

Capítulo 4

Pruebas y resultados

Los resultados obtenidos en este proyecto fue la implementación de una interfaz Ethernet a un sistema de monitoreo de calidad de la energía así como su interfaz gráfica de usuario diseñada para mostrar los datos adquiridos en tiempo real y generar historiales de los parámetros eléctricos parámetros eléctricos mencionados en el Capítulo 1.

El sistema funciona mandando una activación a través de la tarjeta Arduino, dicha activación le indica al sistema de monitoreo que se llevará a cabo una adquisición, el sistema de monitoreo adquiere los datos de dos canales del convertidor analógico digital, un canal dedicado al monitoreo de corriente y otro canal dedicado al monitoreo de voltaje, estos datos son adquiridos a una frecuencia de muestreo fija de 4.8kHz y por un tiempo de 200ms, lo cual representa 12 ciclos de forma de onda de voltaje y corriente, que conforme a la Norma Internacional IEC 61000-4-30 se establece como el tiempo de muestreo básico para la medición de parámetros como distorsión total armónica, magnitudes de corriente y voltaje. Posteriormente, estos datos son almacenados en una memoria micro-SD incluida en el Ethernet Shield de Arduino y lo que facilita el manejo de la cantidad de datos, una vez actualizado, el archivo es adquirido por medio de una interfaz gráfica a través de una computadora con acceso a la red local donde se encuentre el sistema o bien con acceso a internet. Los datos son procesados en la interfaz donde se obtienen los parámetros seleccionados por el usuario, los cuales son mostrados en tiempo real a través de la interfaz.

Se realizaron las pruebas necesarias para demostrar la funcionalidad de la interfaz Ethernet con el sistema de monitoreo, así como pruebas preliminares que se realizaron en la tarjeta Arduino, para el desarrollo de la página web embebida.

4.1 Página web embebida

En la figura 4.1 se muestra la página web desarrollada, fue elaborada a través de lenguaje de programación HTML5, una de las últimas versiones de programación HTML para páginas web, y CSS3 lenguaje que se usa para estilizar las páginas web. En este caso se trata de una página web estática cuya función principal es servir como medio de activación del sistema, y enlazar una página alterna que albergara los datos adquiridos por el sistema.

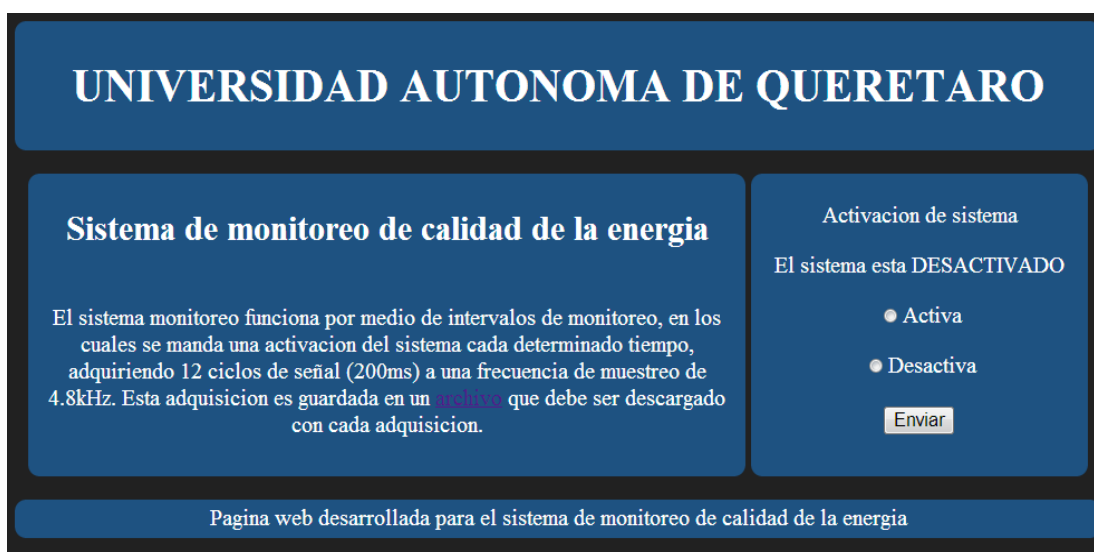


Figura 4.1. Página web embebida en tarjeta Arduino.

A continuación se describe la funcionalidad de acuerdo a las características de la página web.



1. Activar adquisición de sistema de monitoreo.

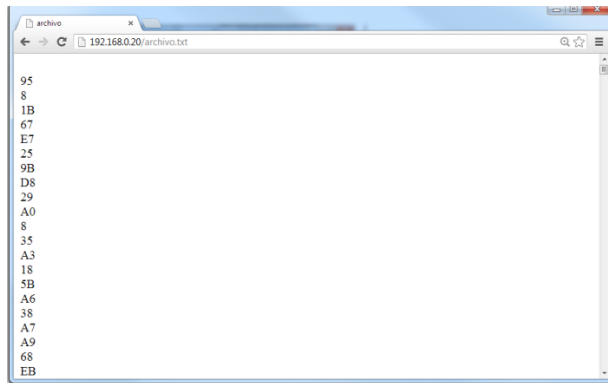
- Botón radio Activa. Realiza la activación del sistema de monitoreo por medio de una entrada digital del microcontrolador. La activación del sistema indica que el inicio de una medición de 200ms.
2. Desactivar adquisición de sistema de monitoreo.
 - Botón radio Desactiva. Realiza la desactivación del sistema de monitoreo a través de la misma salida digital para la activación, es necesario que para cualquier monitoreo que se realice, enseguida se desactive el sistema para, posteriormente realizar otra adquisición.
 3. Envío de estados al sistema.
 - Botón Enviar. Es el encargado de enviar cualquiera de las opciones seleccionadas en los botones radio.
 4. Mostrar estado de sistema.
 - Estado de pin de activación. La página muestra el estado del sistema en el área de activación de sistema en letras mayúsculas ACTIVADO o DESACTIVADO, dependiendo el estado en que se encuentre el sistema.
 5. Actualizar archivo de adquisición de datos:
 - Enlace de archivo. Este enlace permite acceder a una página alterna que se crea a partir del contenido del archivo del cual se realizó la adquisición.
 6. Características generales.
 - La página web embebida puede ser accedida desde una red local, ya sea conexión punto a punto, o a través de un enrutador.
 - La página web embebida puede ser accedida de desde Internet por medio de la conexión a un enrutador que cuente con servicio de Internet.

4.1.1 Acceso por conexión punto a punto

La conexión punto a punto se realiza por medio de la conexión de solo dos nodos, para obtener el acceso través de una red punto a punto se requiere definir la dirección IP de estática que tendrá la tarjeta Arduino, con la que se accederá en el navegador. En la tabla 4.1 se muestran los accesos realizados a través de la página web por medio de la conexión punto a punto, por medio de la dirección IP.

Tabla 4.1. Acceso a página web en conexión punto a punto

Captura de pantalla	Descripción
 A screenshot of a web browser displaying a page for the 'UNIVERSIDAD AUTONOMA DE QUERETARO'. The page title is 'Sistema de monitoreo de calidad de la energia'. The main content area is split into two columns. The left column contains text: 'El sistema monitoreo funciona por medio de intervalos de monitoreo, en los cuales se manda una activacion del sistema cada determinado tiempo, adquiriendo 12 ciclos de señal (200ms) a una frecuencia de muestreo de 4.8kHz. Esta adquisicion es guardada en un archivo que debe ser descargado con cada adquisicion.' The right column is titled 'Activacion de sistema' and shows 'El sistema esta DESACTIVADO'. Below this, there are two radio buttons: 'Activa' (selected) and 'Desactiva'. An 'Enviar' button is at the bottom. A footer reads 'Pagina web desarrollada para el sistema de monitoreo de calidad de la energia'.	<p>Se accede a la página web embebida en la tarjeta Arduino por medio de la dirección IP definida en el programa. Dirección IP: 192.168.0.20 URL: http://192.168.0.20</p>
 A screenshot of the same web browser, but the URL in the address bar is '192.168.0.20/?status=1'. The page content is identical to the first screenshot, but the status in the 'Activacion de sistema' section has changed to 'El sistema esta ACTIVADO'. The 'Desactiva' radio button is now selected, and the 'Activa' button is unselected. The 'Enviar' button remains at the bottom.	<p>Al realizar la activación del sistema se muestra la página del sistema con el estado actualizado en que se encuentra el sistema, y a la dirección URL se le agrega la acción que se envió. URL: http://192.168.0.20/?status=1</p>



Al realizar la desactivación del sistema se muestra la página del sistema con el estado actualizado en que se encuentra el sistema, y a la dirección URL se le agrega la acción que se envió.

URL:

<http://192.168.0.20/?status=0>

Al abrir el enlace del archivo se muestra una página web alterna donde se encuentran los datos de la adquisición en formato hexadecimal, así mismo se agrega la acción realizada que en este caso fue la de acceder al archivo en la dirección URL.

URL:

<http://192.168.0.20/archivo.txt>

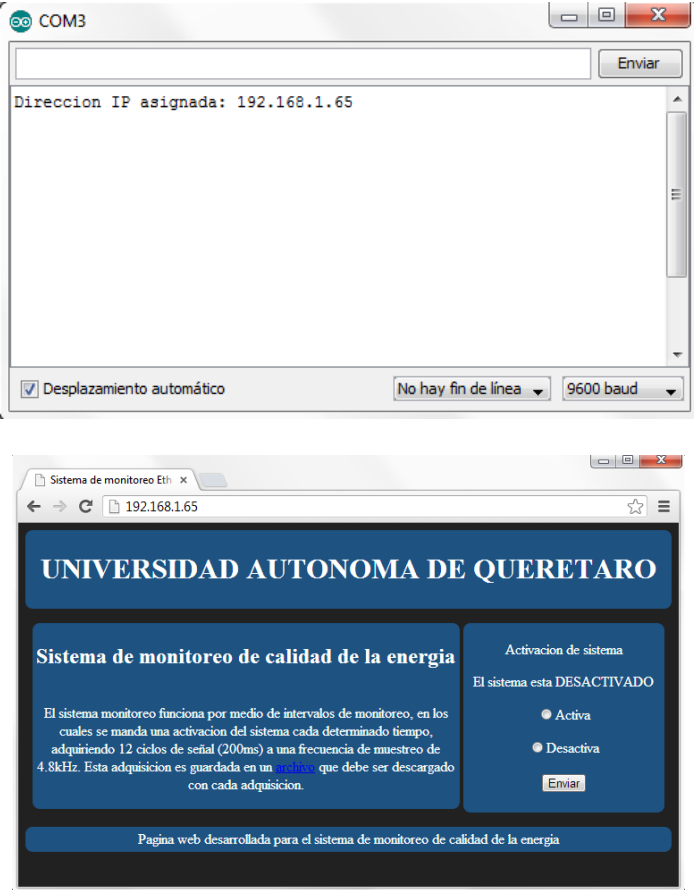
Como se observa cada acción realizada en la página genera una URL diferente, lo que es útil para manejar la adquisición por medio de la interfaz gráfica de usuario.

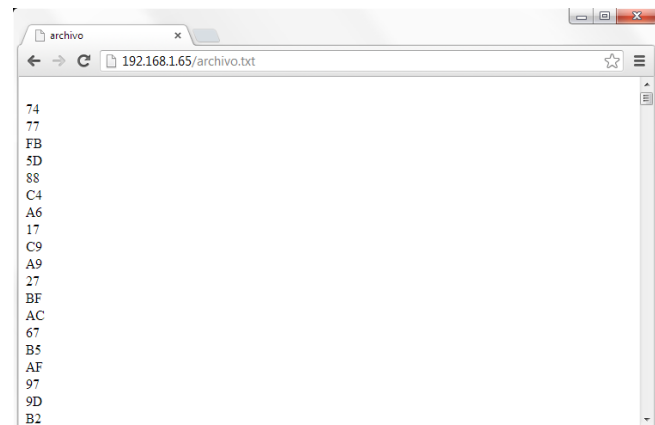
4.1.2 Acceso por red local

El acceso por medio de un enrutador o modem, se lleva a cabo de la misma manera cuando una computadora es conectada a este. Esto es el modem a través del protocolo de configuración dinámica de host DHCP (por sus siglas en inglés *Dynamic Host Configuration Protocol*), asigna una dirección IP a la computadora o dispositivo que se conecta. Esto es de gran ayuda ya que facilita a los usuarios la conexión a una red, sin necesidad de realizar alguna configuración. En la tabla 4.2 se muestran los ejemplos de acceso a la página web embebida a través de una red local, por medio de un enrutador marca HUAWEI. Para realizar la esta conexión se requiere implementar una clase propia de la librería Ethernet de Arduino

la cual recibe una dirección IP del enrutador, que será la dirección URL para acceder a la página web. Asimismo, para que el usuario conozca la dirección IP asignada por el enrutador se debe conectar la tarjeta Arduino al ordenador por su puerto de programación y a través del monitor serial se mostrará la dirección IP asignada.

Tabla 4.2. Acceso a página web en red local.

Captura de pantalla	Descripción
 <p>The top screenshot shows a serial monitor window titled 'COM3'. The text displayed is 'Direccion IP asignada: 192.168.1.65'. The bottom screenshot shows a web browser window with the address '192.168.1.65'. The page content includes the header 'UNIVERSIDAD AUTONOMA DE QUERETARO' and a section titled 'Sistema de monitoreo de calidad de la energia'. It contains a paragraph about the monitoring system, a control panel for 'Activacion de sistema' (currently 'DESACTIVADO') with 'Activa' and 'Desactiva' radio buttons and an 'Enviar' button, and a footer 'Pagina web desarrollada para el sistema de monitoreo de calidad de la energia'.</p>	<p>Monitor serial incluido en el IDE de Arduino, si el enrutador cuenta con el protocolo DHCP se mostrará la dirección IP asignada de lo contrario mandará un mensaje de error.</p> <p>Se accede a la página web a través de la dirección obtenida previamente. URL: http://192.168.1.65</p>



Las acciones realizadas como en el caso anterior, pueden ser vistas o realizadas a través de la URL correspondiente a cada acción. En este caso se muestra una activación del sistema.

URL:
<http://192.168.1.65/?status=1>

En este caso se muestra la URL correspondiente a la desactivación del sistema.

URL:
<http://192.168.1.65/?status=0>



Finalmente se muestra la página y la URL correspondiente al enlace del archivo de texto.

URL:
<http://192.168.1.65/archivo.txt>

4.1.3 Acceso por Internet

Además de permitir el acceso a través de cualquier red local, es posible tener acceso a través de cualquier computadora con acceso a Internet. Actualmente existe una librería proporcionada por Yaler, un servidor que permite acceso fácil y seguro a través de Internet, solo es necesario agregar la librería proporcionada, y crear una cuenta gratuita en la página de Yaler, en donde genera un nombre de dominio al que se accederá a través de Internet, esto representa una modificación mínima al programa para conexión a través de una red local, en la tabla 4.3 se muestran algunas capturas de pantalla de acceso a la página web a través de Internet.

Tabla 4.3. Capturas de pantalla de acceso a página web por Internet.

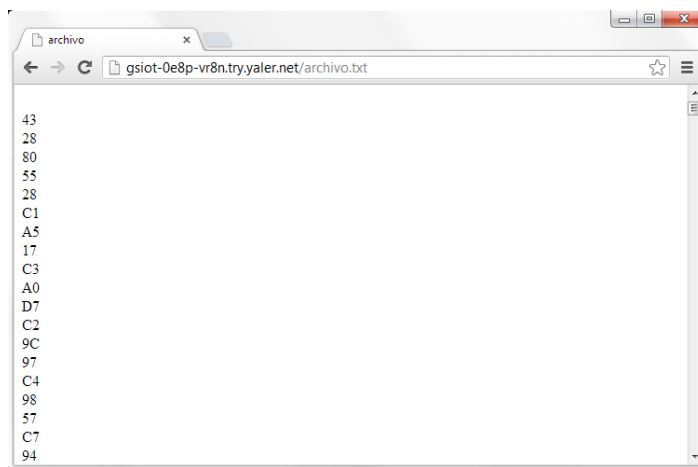
Captura de pantalla	Descripción
 A screenshot of a web browser displaying the 'Sistema de monitoreo de calidad de la energía' page. The page header is 'UNIVERSIDAD AUTONOMA DE QUERETARO'. The main content area is split into two columns. The left column contains text about the monitoring system's operation. The right column, titled 'Activación de sistema', shows 'El sistema esta DESACTIVADO' and two radio buttons for 'ON' and 'OFF', with 'OFF' selected. A 'Submit' button is at the bottom of this section. A footer note says 'Pagina web desarrollada para el sistema de monitoreo de calidad de la energia'.	Acceso a página a través de Internet, por medio de la librería y el nombre de dominio proporcionados por Yaler.
 A screenshot of the same web browser, but the URL now includes '?status=1'. The page content is identical to the previous screenshot, but the status in the 'Activación de sistema' section is now 'El sistema esta ACTIVADO' and the 'ON' radio button is selected.	Tal como en los casos anteriores se agrega información a la dirección URL con forme se llevan a cabo las acciones. En este caso se envió una activación al sistema
	URL: http://gsiot-0e8p-br8n.try.yaler.net/?status=1



En este caso se envía una desactivación al sistema.

URL:

<http://gsiot-0e8p-br8n.try.yaler.net/?status=0>



Finalmente se envía una petición del archivo.

URL:

<http://gsiot-0e8p-br8n.try.yaler.net/archivo.txt>

4.2 Pruebas preliminares de adquisición

Para verificar la correcta comunicación del sistema de monitoreo con la tarjeta Arduino y con la finalidad de comprobar los métodos propuestos en Matlab para el cálculo de los índices necesarios se llevaron a cabo algunas pruebas de adquisición. Que servirían posteriormente para la implantación de la interfaz gráfica de usuario. La pruebas consistieron en realizar adquisiciones por medio de red local a través de Matlab que cuenta con una función para la lectura de direcciones URL, devolviendo una cadena de caracteres que

representa el código HTML de la página web. Las acciones que realiza este programa se describen a continuación

1. Accede a la activación y a la desactivación a través de las direcciones URL con que cuenta dependiendo del tipo de conexión que se tenga.
2. Acceder al archivo de texto que contiene los datos adquiridos (corriente y voltaje).
3. Realizar la interpretación de los datos adquiridos.
4. Calcular los parámetros de calidad de la energía previstos

En la presente prueba se realizó una adquisición en una carga de prueba, que consistía en un foco de 60W, en la figura 4.2 se muestran las adquisiciones realizadas de corriente y voltaje por 200ms, para el cálculo de los parámetros necesarios.

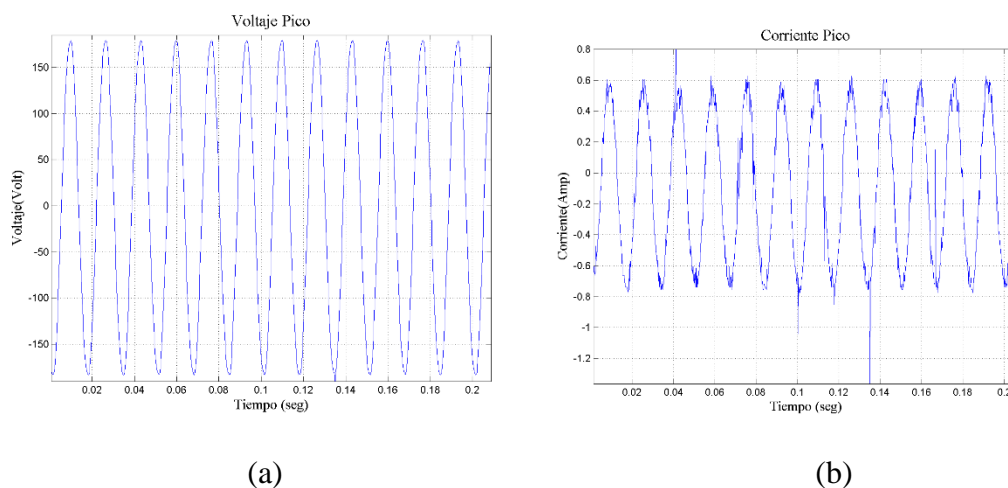
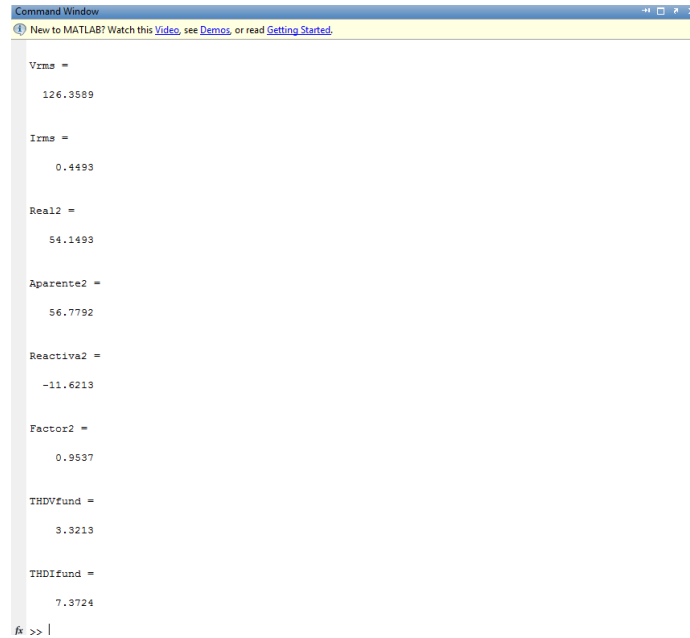


Figura 4.2. a) Gráfica de voltaje, b) Gráfica de corriente.

En la figura 4.3 se muestran los resultados obtenidos de los cálculos de los parámetros que debe mostrar la interfaz de usuario.



```
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

Vrms =
    126.3589

Irms =
    0.4493

Real2 =
    54.1493

Aparente2 =
    56.7792

Reactiva2 =
   -11.6213

Factor2 =
    0.9537

THD Vfund =
    3.3213

THD Ifund =
    7.3724

fx >> |
```

Figura 4.3. Parámetros eléctricos calculados.

4.3 Interfaz gráfica de usuario

Para la adquisición de los datos vía Ethernet se desarrolló una interfaz en Matlab consiste en una ventana principal que se muestra en figura 4.4, en la cual se selecciona el sistema a monitorear, para monitoreo de corriente continua se pulsa el botón DC, y para monitoreo de corriente alterna se pulsa el botón AC.

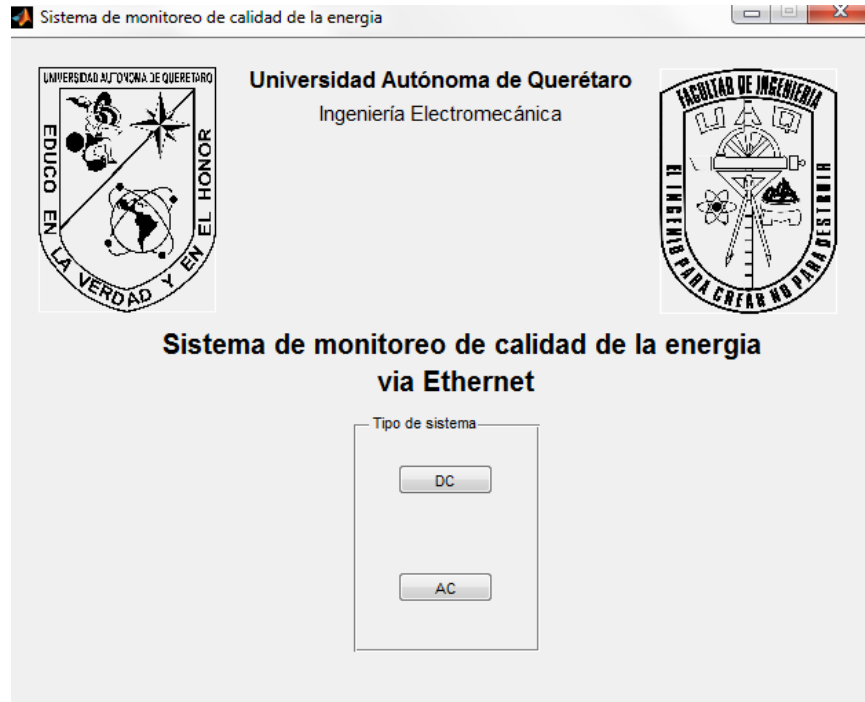


Figura 4.4. Interfaz gráfica de usuario principal.

Al seleccionar el botón DC se abre la ventana de la figura 4.5, como se muestra la ventana contiene un área donde se gráfica el consumo en watts, además cuenta con un área para establecer los parámetros, donde se seleccionan por medio de un “checkbox” los parámetros que se desean guardar en el archivo de historial, además de establecer las direcciones URL para el acceso a la página web y los parámetros de tiempo como el intervalo de muestreo y la duración del monitoreo, en el panel de monitoreo se observan los parámetros que se actualizan cada intervalo seleccionado, además cada parámetro cuenta con un botón de acceso a grafica que muestra los datos adquiridos hasta entonces. En el panel de acciones se selecciona el botón iniciar para comenzar con el monitoreo, en caso de no tener acceso al sistema envía un mensaje de error de conexión.

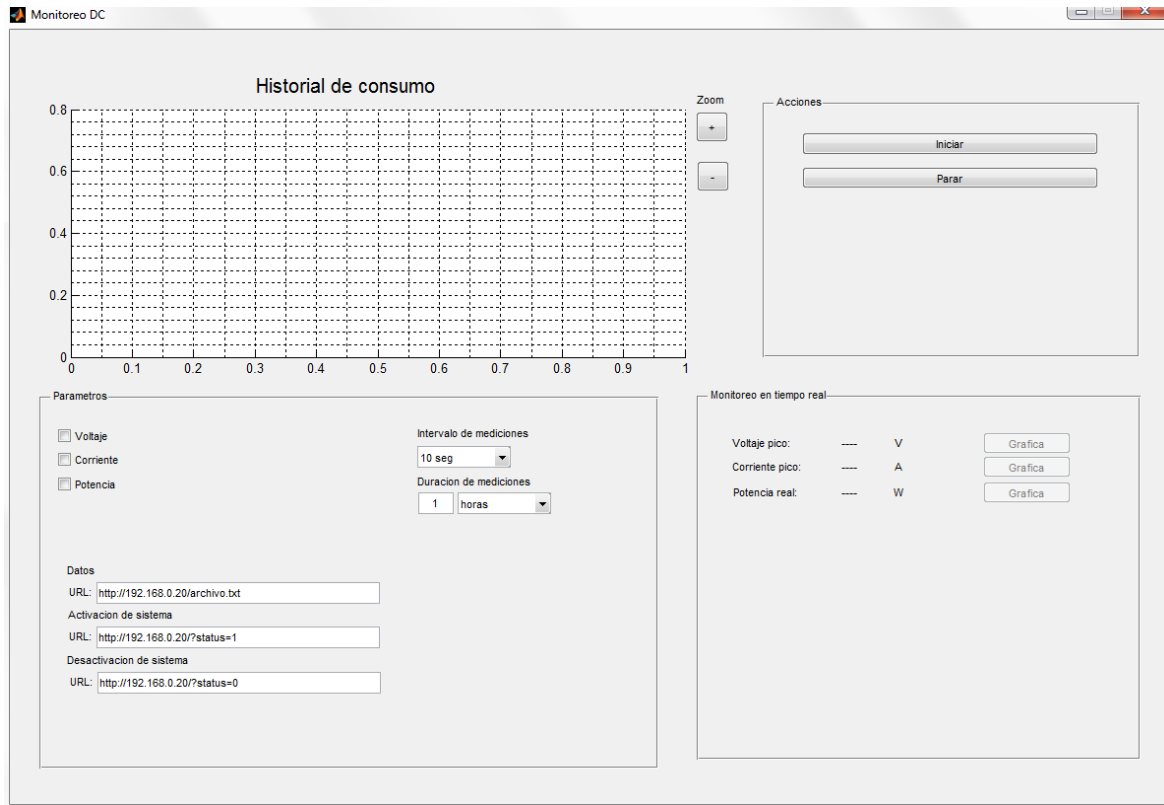


Figura 4.5. Interfaz gráfica de monitoreo de corriente continua.

Al seleccionar el botón AC se abre la ventana para monitoreo en sistemas de corriente alterna pudiendo realizar el monitoreo de corriente y voltaje en sistemas monofásicos, al igual que la ventana de monitoreo de corriente continua se muestra una sección para graficar el consumo en watts, además en el panel de parámetros cuenta con un listado de parámetros en “checkbox” para seleccionar los parámetros que se deseen guardar y cuenta con los cuadros de texto para insertar las direcciones URL requeridas para el acceso a la página web que como se estableció anteriormente puede ser por medio de una red local o a través de Internet, se establece el intervalo de muestreo y el tiempo de duración del muestreo, en el panel de acciones cuenta con el botón iniciar y parar que sirven para comenzar y pausar el monitoreo, respectivamente. En panel de parámetros en tiempo real se muestran los parámetros que se actualizan cada intervalo de tiempo y además cada parámetro cuenta con un botón de gráfica que muestra la gráfica de los datos obtenidos hasta el momento.

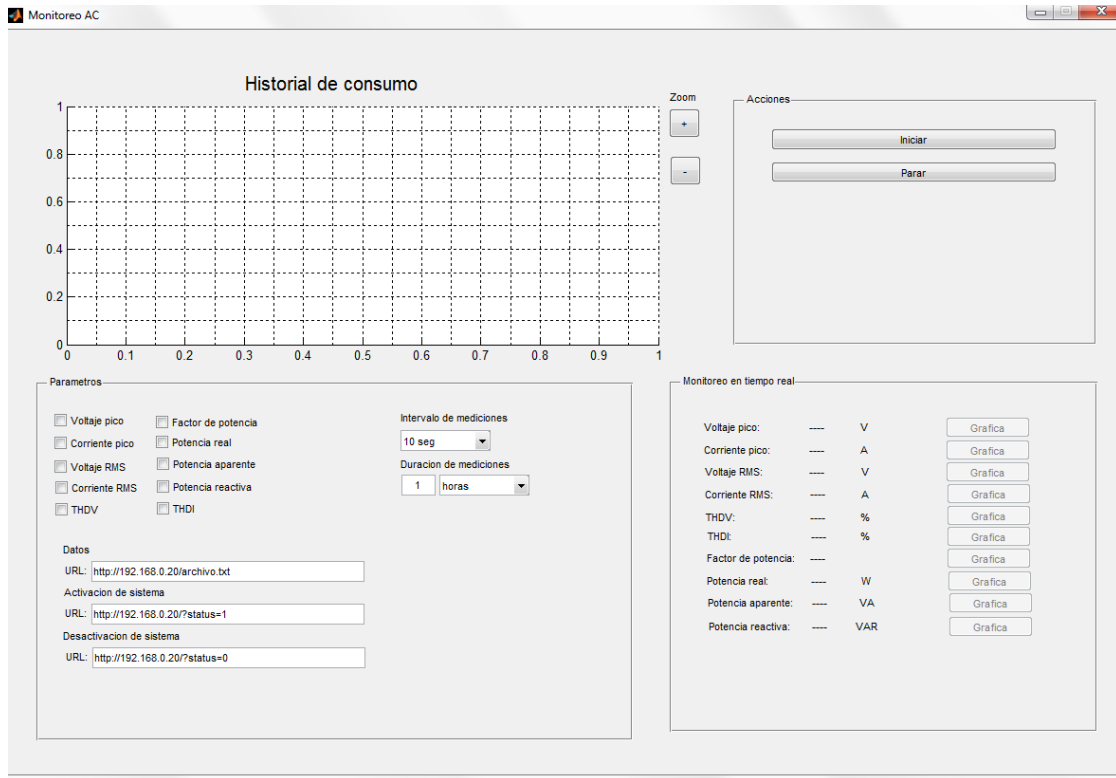


Figura 4.6. Interfaz gráfica de monitoreo de corriente alterna.

4.4 Pruebas preliminares en casa-habitación

Como prueba preliminar del sistema se realizó el monitoreo por medio de una red local y por medio de Internet, a través de un enrutador que le permite al usuario acceso inalámbrico al sistema dentro de la red local y a su vez acceso desde cualquier otra red con acceso a Internet. El monitoreo se realizó a una casa habitación, colocando el sensor de corriente antes de la primer protección de los circuitos lo que permite el monitoreo de todas las cargas disponibles en la vivienda. La primera prueba consistió en un monitoreo de 17 horas seleccionando todos los parámetros de calidad de la energía disponibles para el monitoreo con un intervalo de cada 10 segundos. En la figura 4.8 se muestra una captura del funcionamiento del sistema, en la cual como se mencionó anteriormente el usuario puede visualizar el consumo, y los parámetros actualizándose cada intervalo.

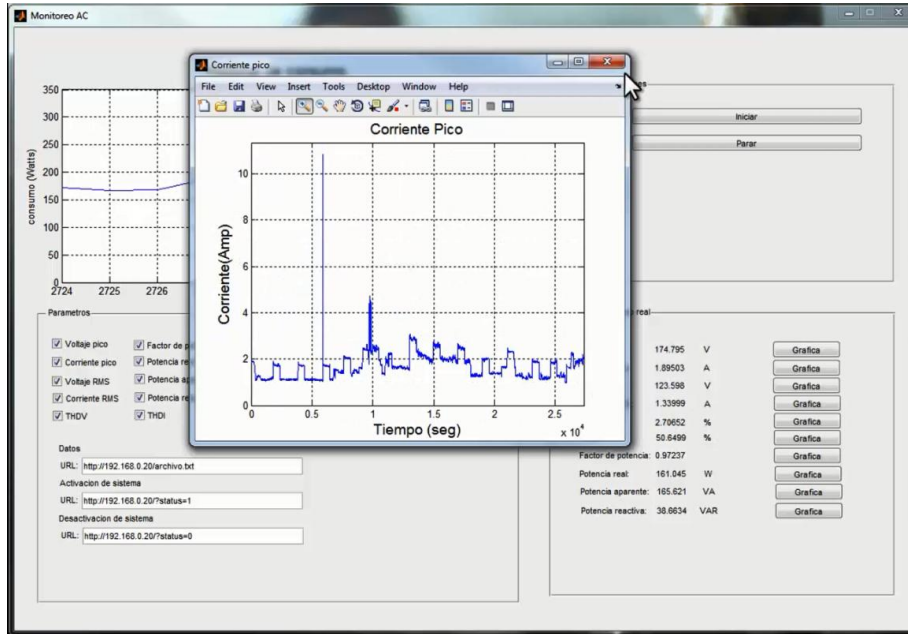
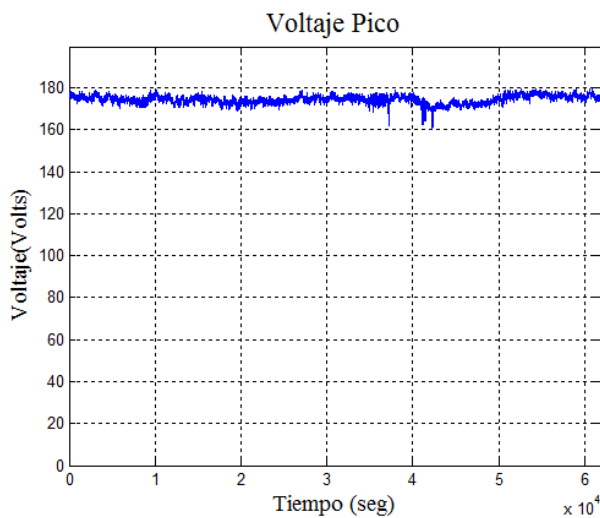


Figura 4.7. Monitoreo de corriente alterna.

Además de visualizar los datos que se actualizan cada intervalo el usuario puede acceder en cualquier momento a las gráficas de cada parámetro, donde se muestran los datos obtenidos hasta el momento, en la figura 4.4 se muestran las gráficas de los parámetros obtenidos al finalizar el monitoreo.

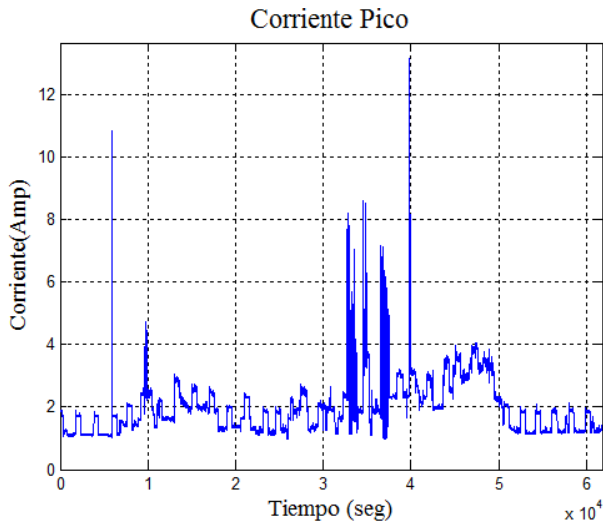
Tabla 4.4. Monitoreo de vivienda en red local.

Gráfica

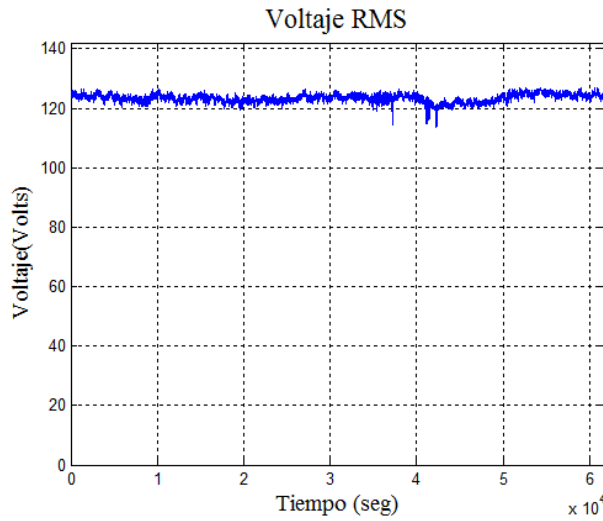


Descripción

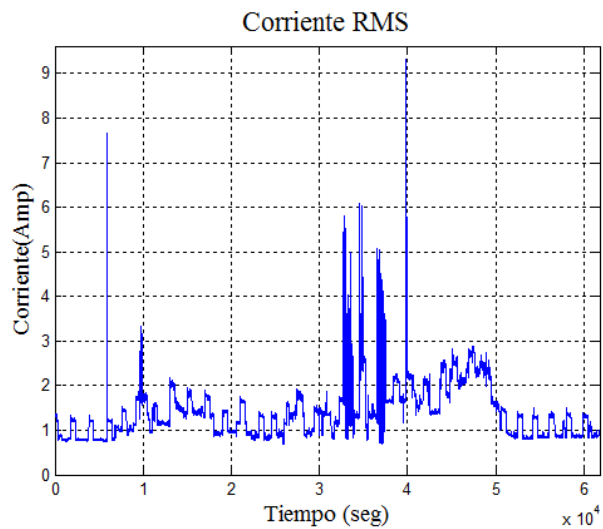
Gráfica de monitoreo de voltaje pico de 17 horas con intervalos de 10 segundos.
 Fecha de inicio
 26/07/2013 4:00am
 Fecha de terminación
 26/07/2013 9:00pm



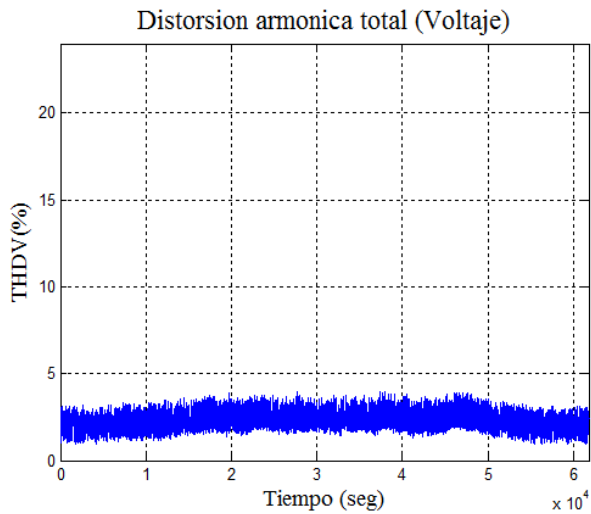
Gráfica de monitoreo de corriente pico de 17 horas con intervalos de 10 segundos.
 26/07/2013 4:00am
 Fecha de terminación
 26/07/2013 9:00pm



Gráfica de monitoreo de voltaje RMS de 17 horas con intervalos de 10 segundos.
 26/07/2013 4:00am
 Fecha de terminación
 26/07/2013 9:00pm



Gráfica de monitoreo de corriente RMS de 17 horas con intervalos de 10 segundos.
 26/07/2013 4:00am
 Fecha de terminación
 26/07/2013 9:00pm

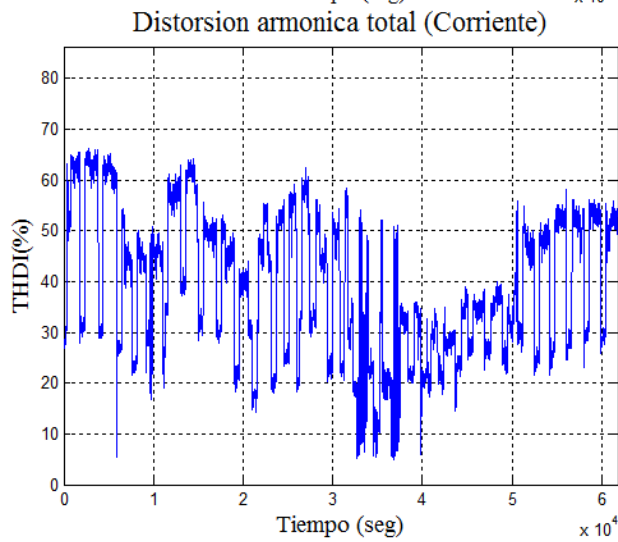


Gráfica de monitoreo de distorsión total armónica de voltaje de 17 horas con intervalos de 10 segundos.

26/07/2013 4:00am

Fecha de terminación

26/07/2013 9:00pm

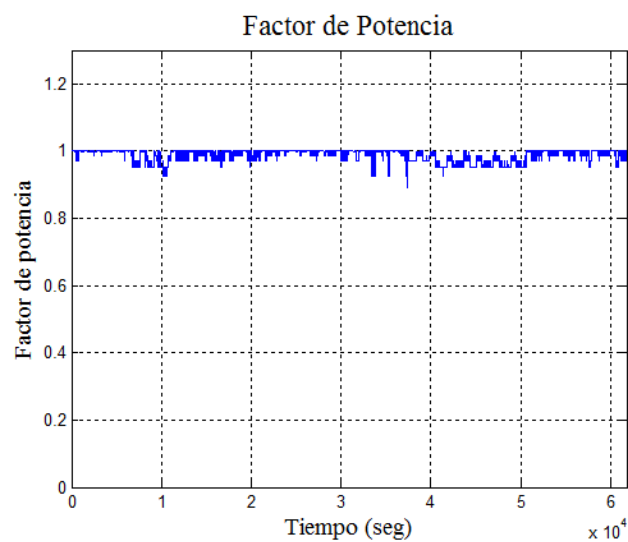


Gráfica de monitoreo de distorsión total armónica de corriente de 17 horas con intervalos de 10 segundos.

26/07/2013 4:00am

Fecha de terminación

26/07/2013 9:00pm

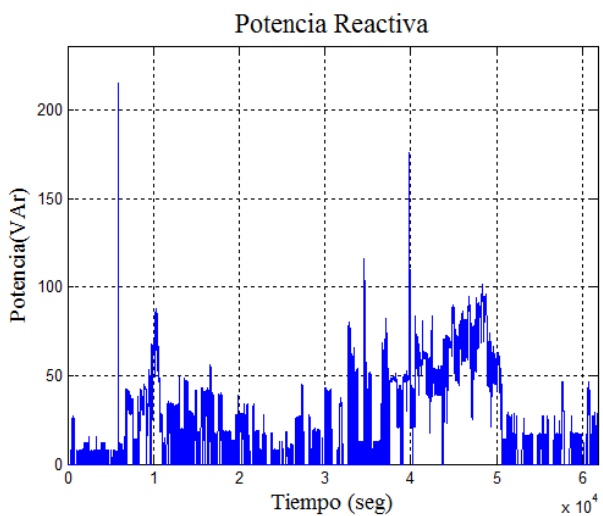
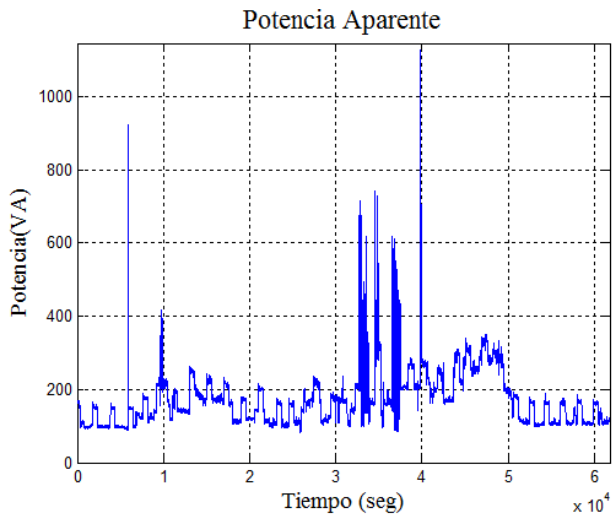
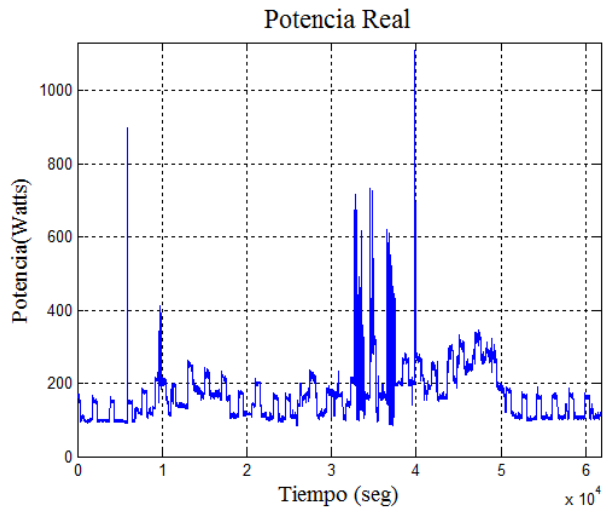


Gráfica de monitoreo de factor de potencia de 17 horas con intervalos de 10 segundos.

26/07/2013 4:00am

Fecha de terminación

26/07/2013 9:00pm



Gráfica de monitoreo de potencia real de 17 horas con intervalos de 10 segundos.

26/07/2013 4:00am

Fecha de terminación

26/07/2013 9:00pm

Gráfica de monitoreo de potencia aparente de 17 horas con intervalos de 10 segundos.

26/07/2013 4:00am

Fecha de terminación

26/07/2013 9:00pm

Gráfica de monitoreo de potencia reactiva de 17 horas con intervalos de 10 segundos.

26/07/2013 4:00am

Fecha de terminación

26/07/2013 9:00pm

Posteriormente se realizó una prueba de monitoreo remoto en la misma vivienda a través de Internet, esta prueba se llevó a cabo 8 horas y con un intervalo de 5 minutos desde la Universidad Autónoma de Querétaro campus San Juan del Río a una distancia aproximada de 0.97Km , en la figura 4.5 se muestra las localizaciones.

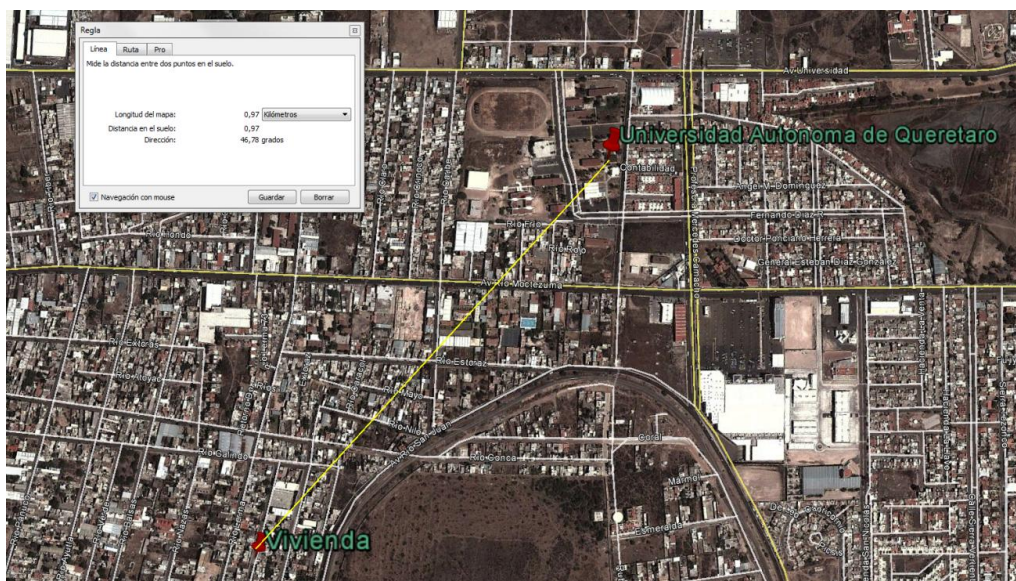
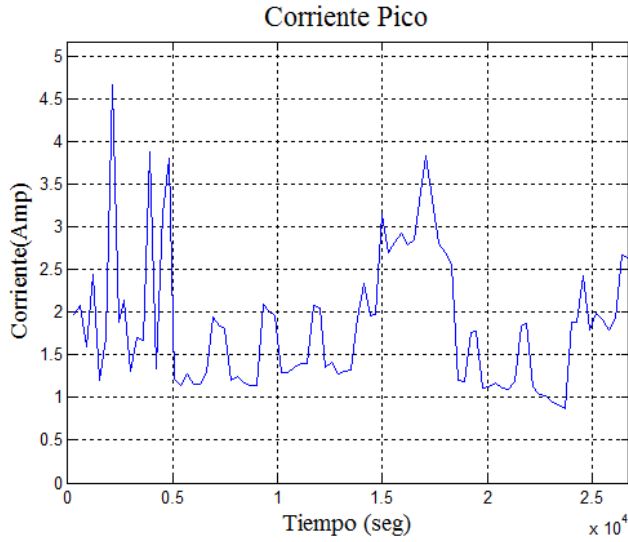


Figura 4.8. Prueba de monitoreo remoto a través de Internet.

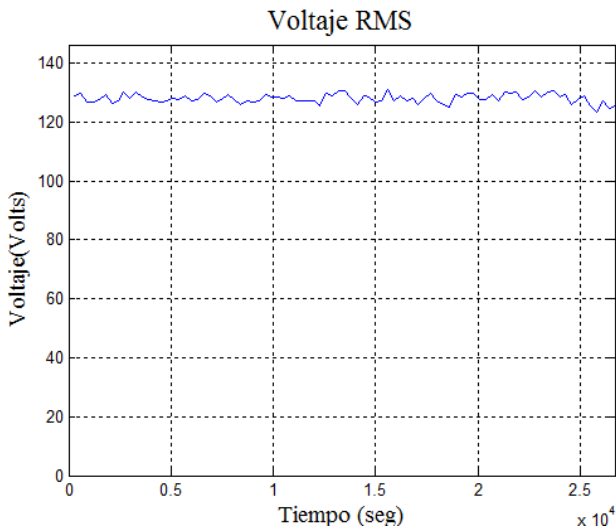
En la tabla 4.6 se muestran las gráficas de los parámetros obtenidos.

Tabla 4.5. Monitoreo a vivienda través de Internet.

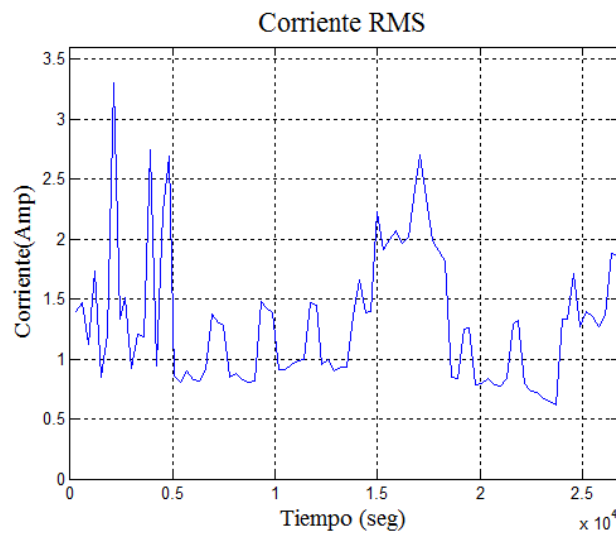
Gráfica	Descripción
<p style="text-align: center;">Voltaje Pico</p>	<p>Gráfica de monitoreo de voltaje pico de 8 horas con intervalos de 5 minutos. Fecha de inicio 27/07/2013 1:00pm Fecha de terminación 27/07/2013 8:00pm</p>



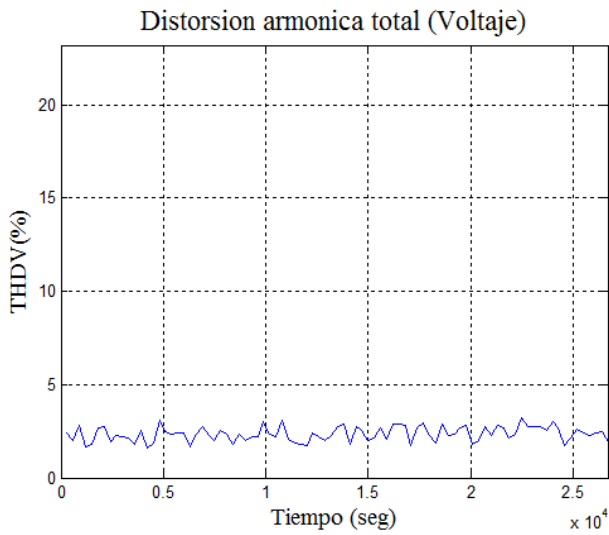
Gráfica de monitoreo de corriente pico de 8 horas con intervalos de 5 minutos.
 Fecha de inicio
 27/07/2013 1:00pm
 Fecha de terminación
 27/07/2013 8:00pm



Gráfica de monitoreo de voltaje RMS de 8 horas con intervalos de 5 minutos.
 Fecha de inicio
 27/07/2013 1:00pm
 Fecha de terminación
 27/07/2013 8:00pm



Gráfica de monitoreo de corriente RMS de 8 horas con intervalos de 5 minutos.
 Fecha de inicio
 27/07/2013 1:00pm
 Fecha de terminación
 27/07/2013 8:00pm



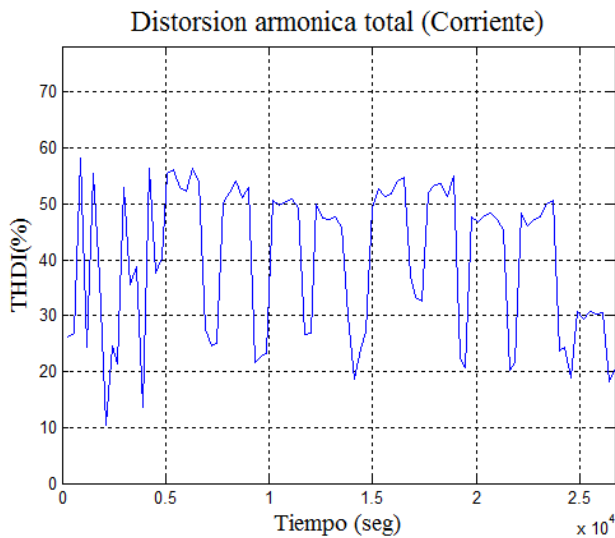
Gráfica de monitoreo de distorsión total armónica de voltaje de 8 horas con intervalos de 5 minutos.

Fecha de inicio

27/07/2013 1:00pm

Fecha de terminación

27/07/2013 8:00pm



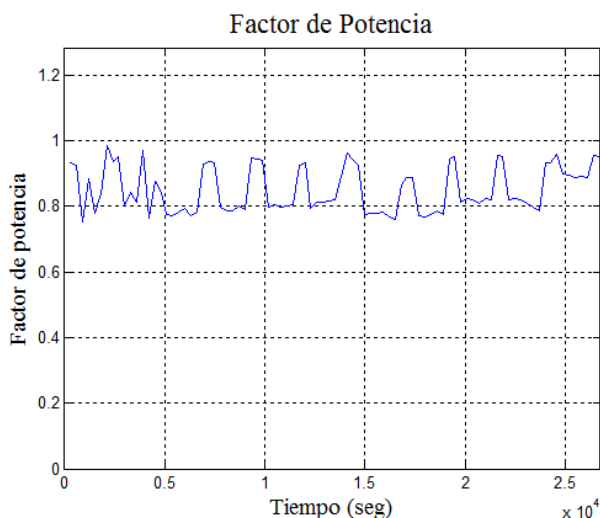
Gráfica de monitoreo de distorsión total armónica de corriente de 8 horas con intervalos de 5 minutos.

Fecha de inicio

27/07/2013 1:00pm

Fecha de terminación

27/07/2013 8:00pm



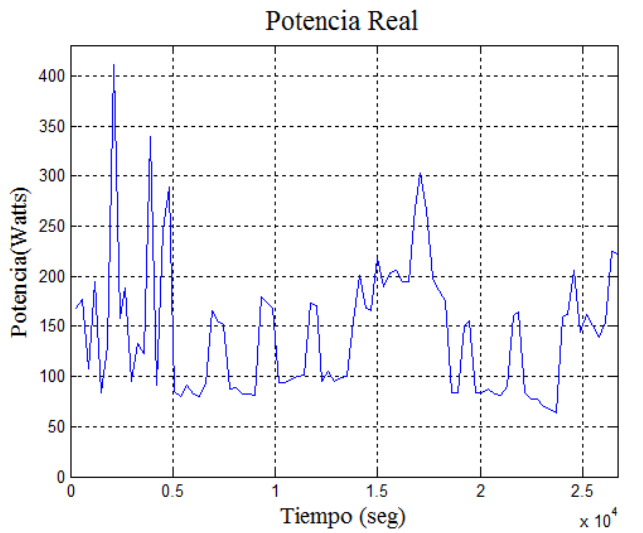
Gráfica de monitoreo de factor de potencia de 8 horas con intervalos de 5 minutos.

Fecha de inicio

27/07/2013 1:00pm

Fecha de terminación

27/07/2013 8:00pm



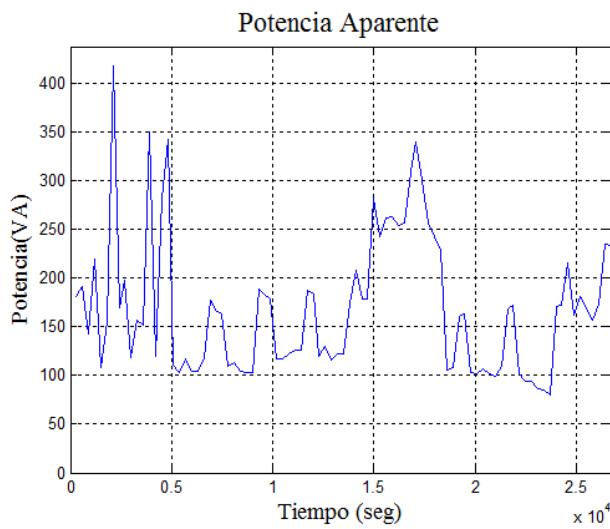
Gráfica de monitoreo de potencia real de 8 horas con intervalos de 5 minutos.

Fecha de inicio

27/07/2013 1:00pm

Fecha de terminación

27/07/2013 8:00pm



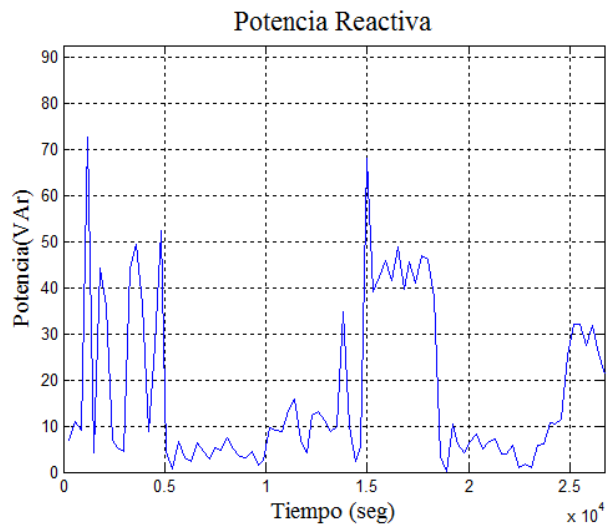
Gráfica de monitoreo de potencia aparente de 8 horas con intervalos de 5 minutos.

Fecha de inicio

27/07/2013 1:00pm

Fecha de terminación

27/07/2013 8:00pm



Gráfica de monitoreo de potencia reactiva de 8 horas con intervalos de 5 minutos.

Fecha de inicio

27/07/2013 1:00pm

Fecha de terminación

27/07/2013 8:00pm

4.5 Prueba final en Fitotrón

Como prueba final se realizó el monitoreo en la cámara de crecimiento Fitotrón desarrollada en la facultad de ingeniería de la Universidad Autónoma de Querétaro, figura 4.5.1, esta prueba se realizó en dos puntos de alimentación del Fitotrón, el primer monitoreo se realizó en la alimentación de corriente alterna con todas las cargas disponibles operando a su máxima capacidad, posteriormente se realizó el monitoreo en la alimentación con corriente directa cuyas cargas correspondían a la iluminación por lámparas LED. Esta prueba se realizó a través de una conexión punto a punto. En la tabla 4.5 se muestran las gráficas obtenidas de monitoreo de 1 hora y media a las cargas alimentadas con corriente alterna (humidificador y ventilador).

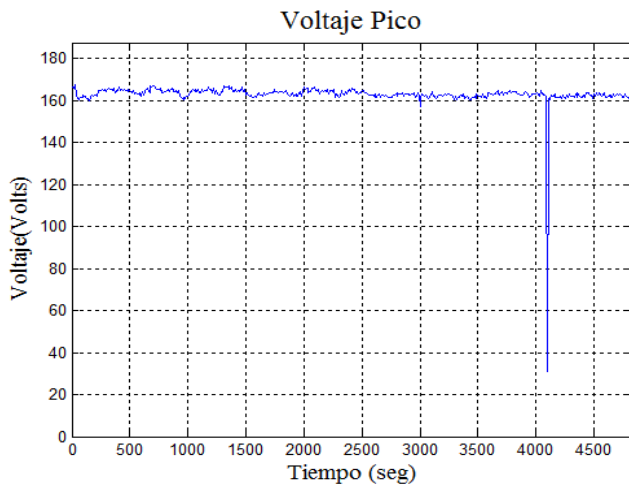


Figura 4.9. Componentes utilizados para monitoreo en Fitotrón.

Tabla 4.6. Monitoreo a Fitotrón en corriente alterna.

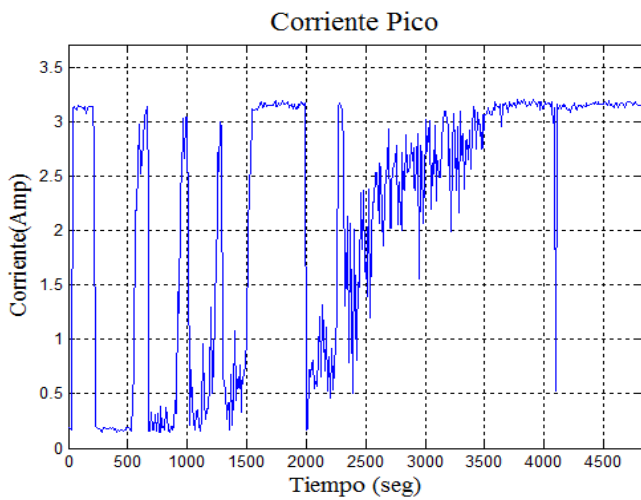
Gráfica

Descripción



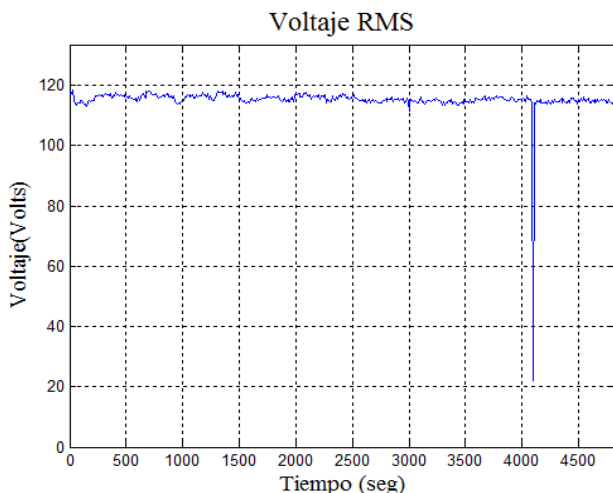
Gráfica de monitoreo de voltaje pico de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



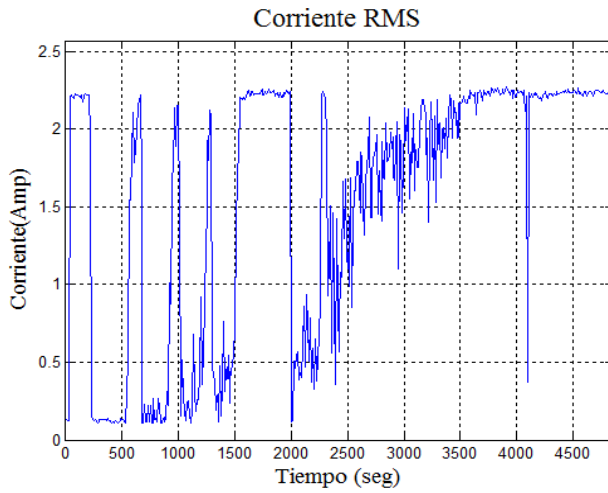
Gráfica de monitoreo de corriente pico de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



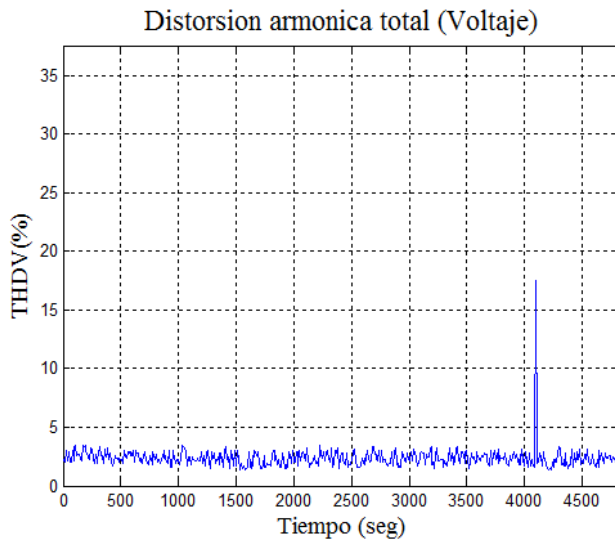
Gráfica de monitoreo de voltaje RMS de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



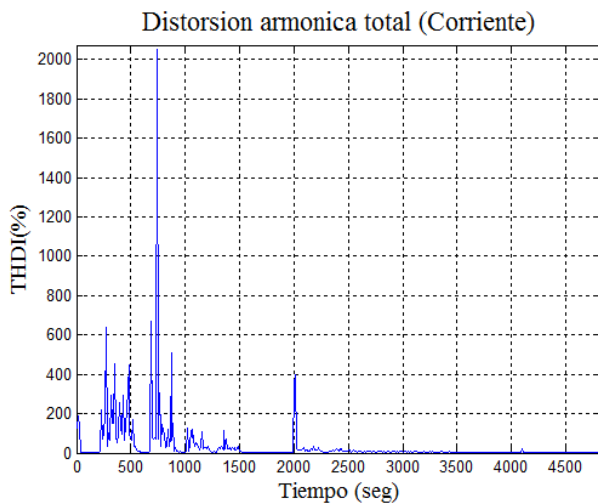
Gráfica de monitoreo de corriente RMS de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



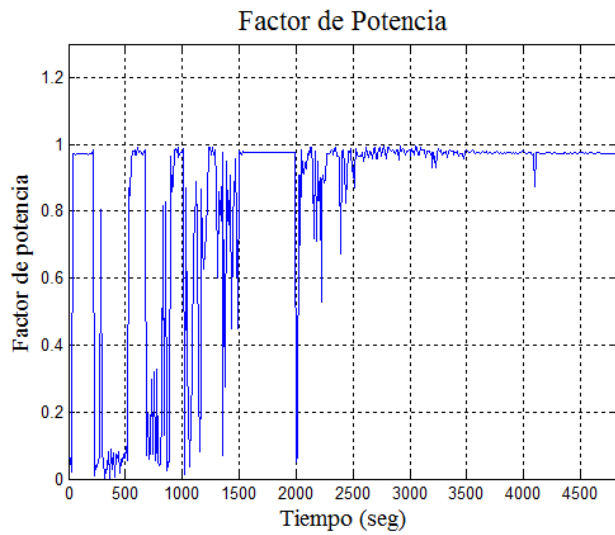
Gráfica de monitoreo de distorsión total armónica de voltaje de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



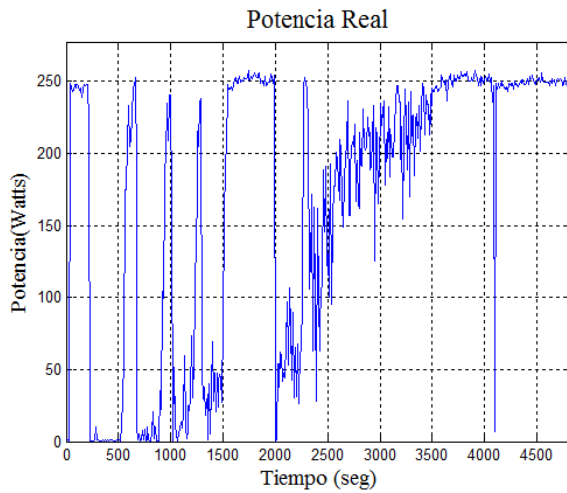
Gráfica de monitoreo de distorsión total armónica de corriente 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



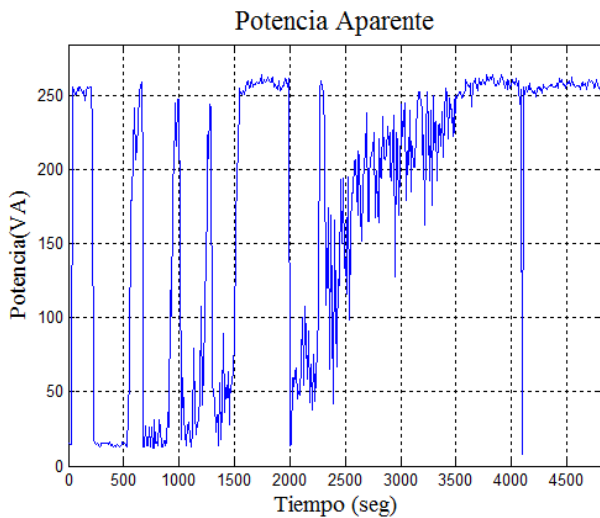
Gráfica de monitoreo de factor de potencia de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



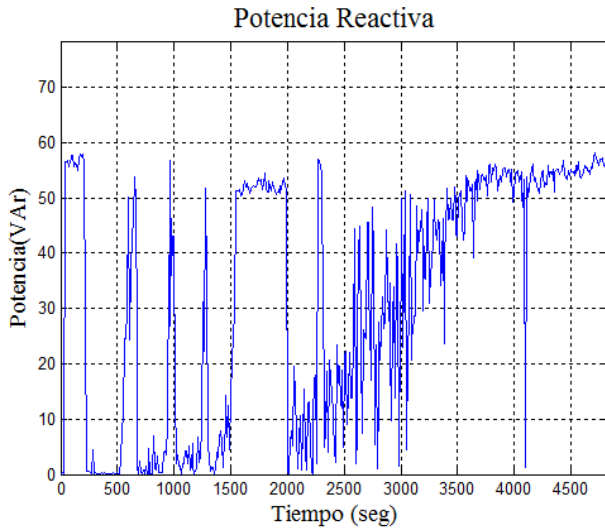
Gráfica de monitoreo de potencia real de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



Gráfica de monitoreo de potencia aparente de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm



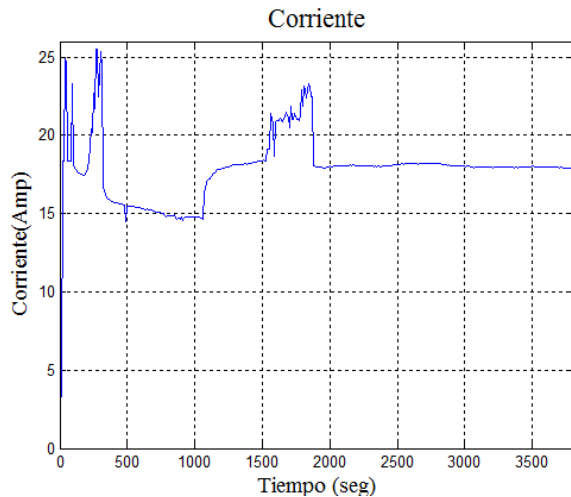
Gráfica de monitoreo de potencia reactiva de 1 hora y media con intervalos de 10 segundos.

Fecha de inicio
08/08/2013 12:00pm
Fecha de terminación
08/08/2013 1:30pm

Finalmente se realizó el monitoreo de uno de los paneles solares que suministran corriente directa al inversor que alimenta las cargas destinadas a la alimentación por energías alternas, actualmente se alimentan las lámparas LED usadas para el control de iluminación del Fitotrón, el monitoreo en este caso se realizó por una hora con intervalos de 10 segundos. En la tabla 4.6 se muestran los parámetros obtenidos.

Tabla 4.7. Monitoreo a Fitotrón en corriente continua.

Gráfica	Descripción
<p>Este gráfico muestra el voltaje en Volts durante un periodo de 3500 segundos. El voltaje comienza con fluctuaciones entre 15 y 25 Volts, se estabiliza en unos 15 Volts hasta los 1000 segundos, luego sube a unos 20 Volts hasta los 1800 segundos, y finalmente se estabiliza en unos 18 Volts hasta el final del periodo.</p>	<p>Gráfica de monitoreo de voltaje de una hora con intervalos de 10 segundos.</p> <p>Fecha de inicio 08/08/2013 2:00pm Fecha de terminación 08/08/2013 3:00pm</p>



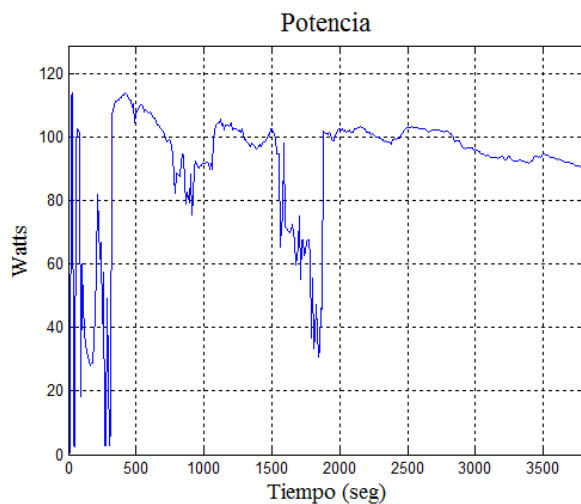
Gráfica de monitoreo de corriente de una hora con intervalos de 10 segundos.

Fecha de inicio

08/08/2013 2:00pm

Fecha de terminación

08/08/2013 3:00pm



Gráfica de monitoreo de potencia de una hora con intervalos de 10 segundos.

Fecha de inicio

08/08/2013 2:00pm

Fecha de terminación

08/08/2013 3:00pm

Como resultado se ha logrado la integración de monitoreo vía Ethernet a un sistema basado en FPGA, que ofrece monitoreo remoto del sistema a través de una red local o a través de Internet, además de que se cuenta con una interfaz que puede ser fácilmente integrada para cualquier otra aplicación que se desee.

4.6 Ventajas y desventajas

Este apartado pretende mencionar algunas de las ventajas que tiene el monitoreo vía Ethernet con sistemas embebidos y a su vez las desventajas que presenta el sistema.

Ventajas

- Ubicuidad: La tecnología de comunicación vía Ethernet está presente en todas partes.
- Conexión a Internet: Permite comunicación remota si se cuenta con acceso a Internet.
- Bajo costo: La comunicación vía Ethernet continúa ganando aceptación en aplicaciones de medición y control debido a su bajo costo.

Desventajas

- Dependencia de software: Para la operación del monitoreo se requiere que el usuario cuente con el software desarrollado para el monitoreo.
- Gran cantidad de datos a transferir: En el caso específico de monitoreo de calidad de la energía se requiere de una cantidad de datos muy grande lo cual limita a este proyecto en cuanto a los intervalos de medición, donde para monitoreo a través de red local se define un intervalo mínimo de 10 segundos y para monitoreo a través de Internet se tiene un intervalo mínimo de 5 minutos.

Capítulo 5

Conclusiones

La aportación práctica de este proyecto es la implementación de una interfaz de monitoreo remoto a un sistema de monitoreo de calidad de la energía, que no solo puede ser usado para esta aplicación sino para aplicaciones futuras que requiera la universidad. Con el desarrollo de este proyecto se logró implementar comunicación remota vía Ethernet a un sistema basado en FPGA de monitoreo de calidad de la energía, que permite al usuario realizar monitoreo a través de una red local o en su defecto a través de Internet en sistemas monofásicos de baja tensión tanto en corriente alterna como en corriente directa.

En este proyecto se utilizó la plataforma Arduino, de hardware libre basada en un microcontrolador ATmega2560, que además de su facilidad para programar y su amplia gama de librerías existentes, proporciona una interfaz de bajo costo para implementación de comunicación remota vía Ethernet. Una de las ventajas de la tarjeta Ethernet que se utilizó en este proyecto es que cuenta con un módulo para memoria micro-SD, lo cual en el caso de este proyecto resultó de mucha utilidad por la cantidad de datos que se transferían por cada intervalo de medición.

El sistema de monitoreo remoto de calidad de la energía favorece al usuario en proporcionar en cualquier momento acceso a la información que se tiene desde cualquier parte con acceso a Internet o a la red local, y en el caso de la Universidad Autónoma de Querétaro pudiera ser de utilidad en la línea terminal de energías alternativas para estudios de medición continua.

Trabajo futuro

Como se menciona una de las principales desventajas de este trabajo es la gran cantidad de datos que tienen que ser transferidas para posteriormente por el software ser procesadas para obtener los parámetros de calidad de la energía seleccionados.

Un trabajo a futuro para este proyecto sería el desarrollo de un módulo procesador digital de señales DSP (por sus siglas en inglés *Digital Signal Processor*) en hardware, el cual proporcione los parámetros de calidad de la energía en lugar de la señal completa, esto reduciría el tiempo de transferencia y a su vez eliminaría la dependencia del software para la adquisición y el procesamiento, pudiendo generar el historial de los parámetros dentro de la misma Interfaz gracias a que se cuenta con un módulo micro-SD provee la capacidad de almacenamiento en la misma interfaz.

Bibliografía

- Arroyo González, Abdiel y Duran Balderas, Erick Martin 2010. Desarrollo de tecnología Ethernet y sus aplicaciones. Universidad Autónoma de Querétaro.
- Alexon, Jan 2003. Embedded Ethernet and Internet complete. Laview Research LLC.
- Angulo Usategui, José Ma. 2003. Microcontroladores diseño práctico de aplicaciones. Editorial McGrawHill.
- Bernspang, Johan 2004. Interfacing an external Ethernet MAC/PHY to a MicroBlaze system on a Virtex-II FPGA. Universidad de Queensland.
- Downs, R. J. 1980. “The Botanical Review”. Publisher: Springer New York.
- Enríquez Harper, Gilberto.1999. El ABC de la calidad de la energía. Editorial Limusa
- Guzmán Gallegos, Ricardo 2009. Monitoreo y control de riego mediante una red Ethernet con interfaz en labview: Adquisición de datos en una red modbus plus a través de una red interbus. Universidad Autónoma de Querétaro, Facultad de Ingeniería.
- H. Robbins, Allan y C. Miller Wilhelm.2008. Análisis de circuitos. Teoría y práctica. Editorial Cengage Learning Editores, S.A. de C.V.
- López Gaudencio, Silvia 2010. Monitoreo y control a distancia de un interruptor de distribución mediante el microcontrolador MSP430 de Texas Instruments. Universidad Tecnológica de la Mixteca.
- Macías Fernández, Ricardo 2007. Sistema en tiempo real para las estaciones de monitoreo SISMO1. Universidad de Colima.
- Margolis Michael, 2011. Arduino Cookbook. O’Reilly Media.
- Martínez Velázquez, Gabriela 2009. Comunicación de elemento final de control con red Ethernet. Universidad Autónoma de Querétaro, Facultad de Ingeniería.
- Méndez Bautista, David 2009. Sistema de comunicaciones basado en Ethernet para el control de sistemas empotrados. Universidad Tecnológica de la Mixteca.

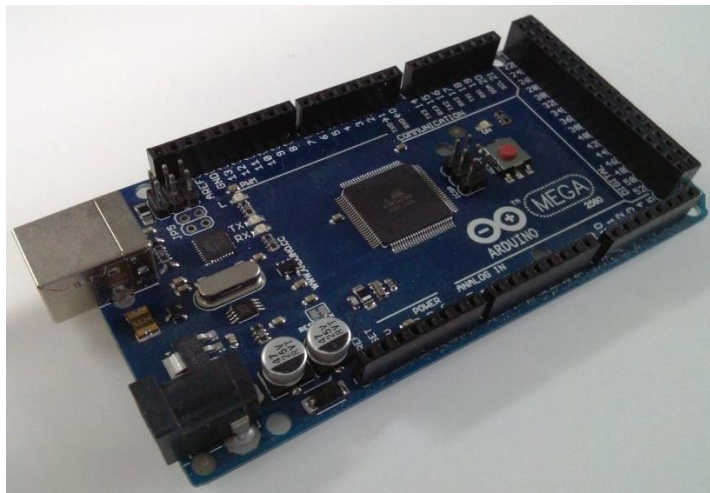
- Mohsenin, Tinoosh 2004. Design and evaluation of FPGA-Based Gigabit-Ethernet/PCI Network Interface Card. Universidad Rice.
- Lujan Mora, Sergio. 2001. Programación en Internet: Clientes Web. Editorial club universitario
- Parker, S. 2002. Dictionary of Scientific and Technical Terms. McGraw-Hill.
- Smith Scott T. 2006. *MATLAB*: Advanced GUI development. Dog Ear Publishing
- Tanenbaum, Andrew S. 2003. Redes de computadoras. Editorial Pearson.
- Toledano Ayala, Manuel 2006. Diseño e implementación de un sistema de monitoreo remoto para un invernadero. Universidad Autónoma de Querétaro, Facultad de Ingeniería.
- Toledano Ayala, Manuel 2010. Arquitectura híbrida basada en sistemas embebidos para el monitoreo remoto de largo alcance de estaciones de sensores. Universidad Autónoma de Querétaro, Facultad de Ingeniería.
- WIZnet, 2009, *W5100* Datasheet

ANEXOS

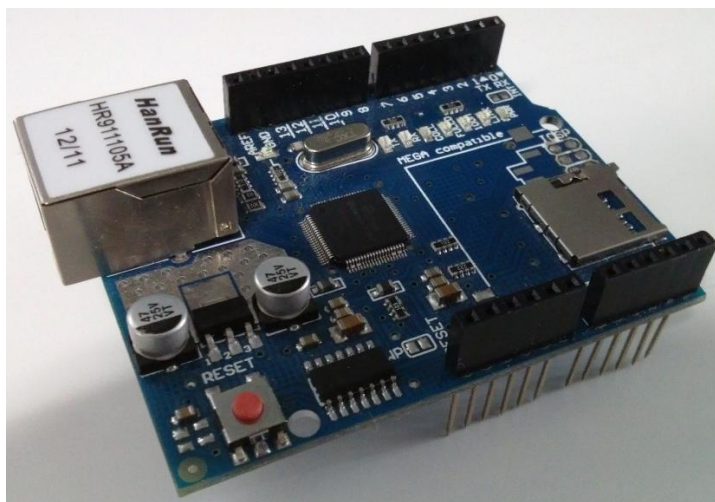
Anexo1. Manual de procedimientos para la instalación de sistema

Antes de comenzar a usar el sistema se debe tener previamente definido el tipo de monitoreo a realizar (punto a punto, red local o Internet), ya que de eso depende el programa que se cargara a la tarjeta Arduino y los pasos a realizar para realizar el monitoreo, a continuación se muestra una lista con los elementos del sistema:

- Tarjeta Arduino Mega2560



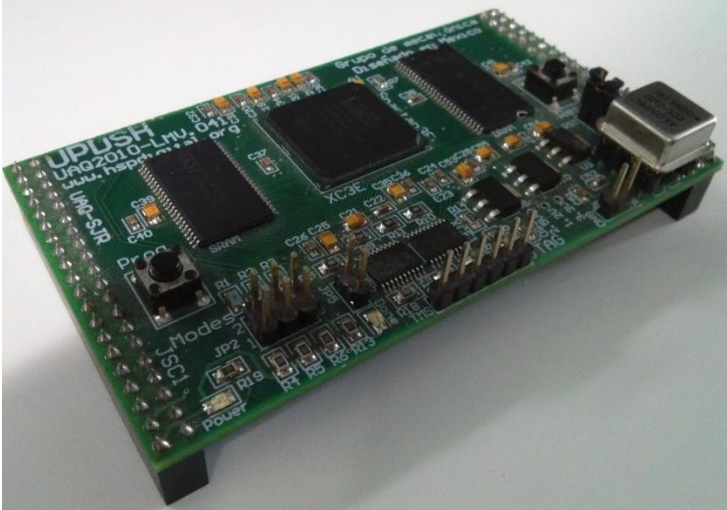
- Ethernet Shield W5100 de Arduino (con ranura para micro-SD)



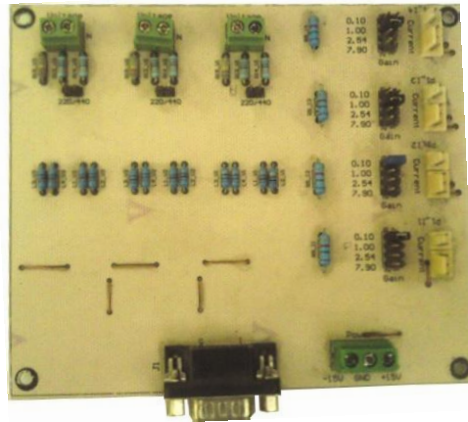
- Tarjeta PLCUAQ (cuenta con los convertidores analógicos digitales).



- Tarjeta UDPSH (FPGA)



- Tarjeta de acondicionamiento de sensores



- Sensores de corriente (sensores efecto Hall o sensores de gancho)



Para realizar la conexión física de los componentes es importante conocer la ubicación y la función que desempeña cada puerto, en la figura 1 se muestran indican los puertos con los que cuenta la tarjeta PLCUAQ, junto con la tarjeta UPDSH, para indicar como se debe colocar. En este caso solo se usaran los puertos de salidas digitales, entradas digitales y entradas analógicas.

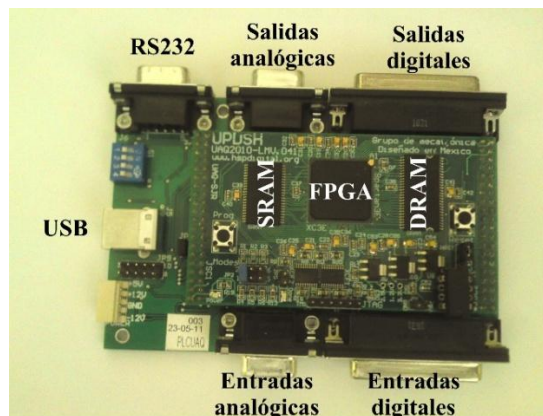


Figura 1. Tarjeta PLCUAQ y UPDSH.

La comunicación FPGA-Arduino se lleva a cabo a partir de salidas digitales a niveles TTL de 5V, a continuación se muestran las conexiones entre los pines de los puertos de la tarjeta PLCUAQ con los pines de la tarjeta Arduino Mega y con la tarjeta de acondicionamiento de los sensores.

Salidas digitales

Puerto DB25

Pin 5 → Pin 19(Arduino)

Pin 18 → Pin 7(Arduino)

Entradas digitales

Puerto DB25

Pin 11→Pin 8(Arduino)

Entradas analógicas

Puerto DB9

Pin 1 → Pin 2(Tarjeta de acondicionamiento puerto DB9)

Pin 4 → Pin 5(Tarjeta de acondicionamiento puerto DB9)

Pin 5 → Pin 1(Tarjeta de acondicionamiento puerto DB9)

En este caso para las conexiones indicadas se usan los conectores mostrados en la figura 2. Para la medición de voltaje solo es necesario realizar la conexión en paralelo al voltaje que se quiere medir, en caso de medición de corriente, la tarjeta esta adecuada para la conexión directa con sensores de efecto Hall.

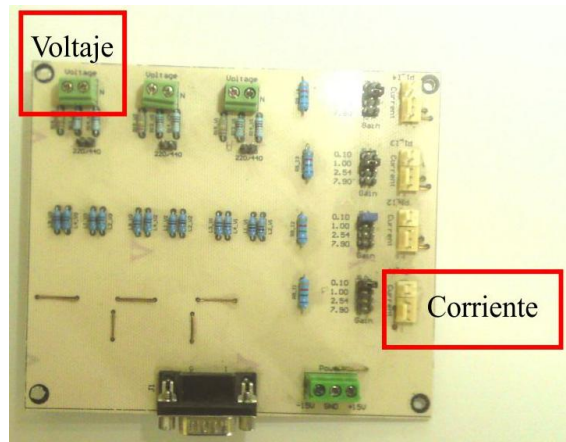


Figura 2. Tarjeta de acondicionamiento.

Una vez realizada la conexión es necesario una fuente para alimentar la tarjeta PLCUAQ, la tarjeta Arduino y la tarjeta de acondicionamiento, en general se requiere de los siguientes niveles de voltaje: 5V, 12V, -12V, 15V, -15V. Posteriormente realizar la implementación del programa en VHDL del sistema de monitoreo. En caso de la tarjeta Arduino como se vio anteriormente se requiere saber qué tipo de conexión que se va a realizar, a continuación se muestran los pasos a realizar para los tipos de conexión disponibles.

Monitoreo punto a punto

Se entiende por conexión punto a punto, la comunicación entre dos dispositivos en este caso el sistema de monitoreo conectado directamente a una computadora a través del puerto RJ-45 usado para comunicación Ethernet. Para realizar la programación de la tarjeta Arduino se requiere descargar e instalar el IDE de Arduino disponible en la página oficial. Posteriormente como se muestra en la figura 3 se selecciona la tarjeta Arduino Mega, en la sección de Herramientas → Tarjeta → Arduino Mega 2560 or Mega ADK.

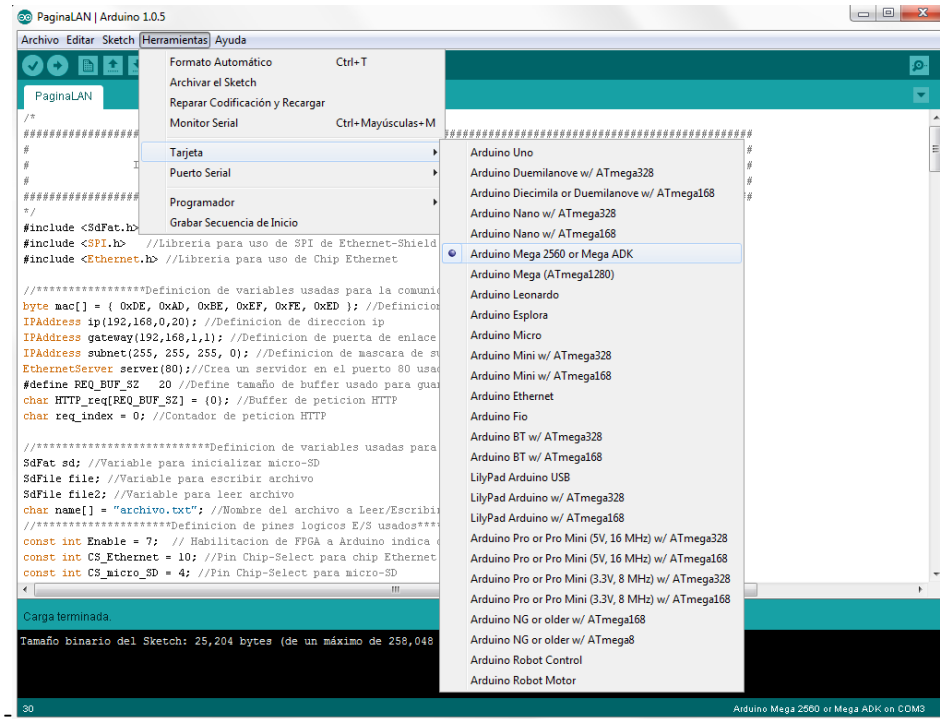


Figura 3. Selección de tarjeta en IDE de Arduino.

Una vez seleccionada la tarjeta se abre el archivo del código del programa para acceso punto a punto, en el cual es posible configurar la “ip”, que servirá para acceder a la página web.

Para cargar el programa a la tarjeta como se muestra en la figura 4 se tiene un icono que directamente compila y carga el programa a la tarjeta.

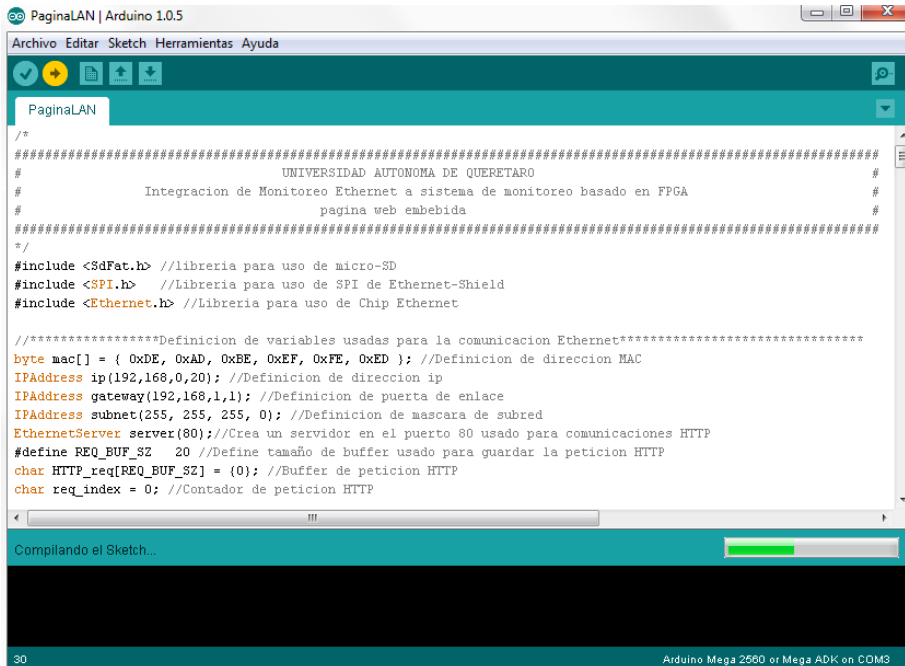


Figura 4. Compilación e implementación de programa en tarjeta Arduino.

Después de haber cargado el programa a la tarjeta con la tarjeta Ethernet Shield conectada se realiza la conexión vía Ethernet con la computadora. Para esto se realiza la configuración en el apartado de centro de redes y recursos compartidos para computadoras con Windows 7 como se muestra en la figura 5, posteriormente se accede al apartado de Conexión de área local.

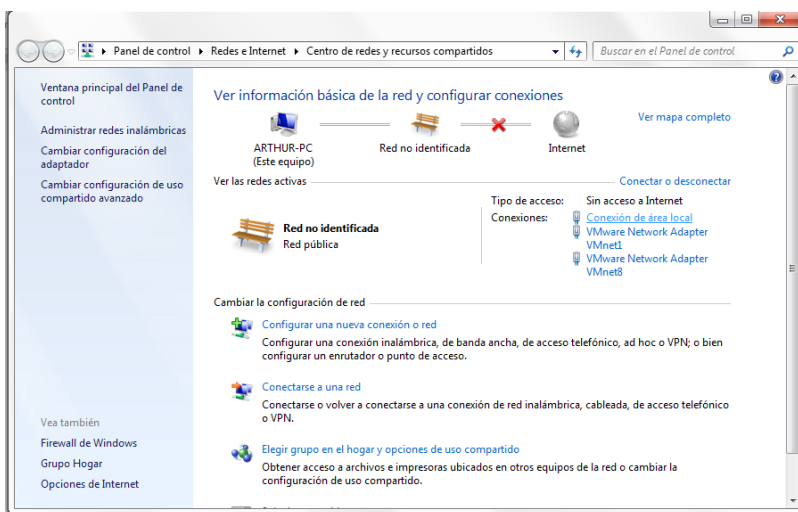


Figura 5. Centro de redes y recursos compartidos Windows 7.

Después de haber presionado en el apartado de conexión de área local se abre la ventana que se muestra en la figura 6, donde se accederá al botón de propiedades que se encuentra en la parte inferior izquierda.

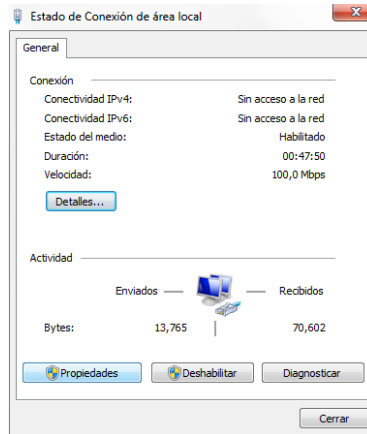


Figura 6. Estado de conexión de área local.

Al presionar el botón de propiedades se abrirá la ventana que se muestra en la figura 7, donde aparece una lista de las propiedades se localiza la que tenga el nombre de Protocolo de Internet versión 4 (TCP/IPv4), y se accede dando doble click.

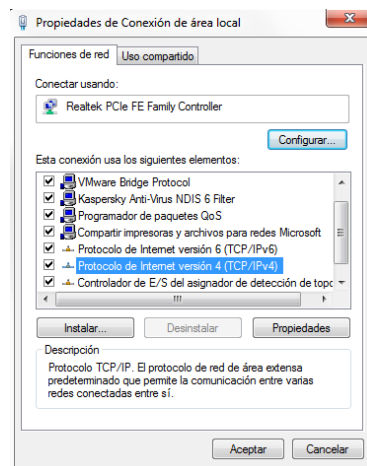


Figura 7. Propiedades de conexión de área local.

Del paso anterior se abrirá la ventana que se muestra en la figura 8, donde inicialmente estará seleccionado el botón de radio de "Obtener una dirección IP automáticamente", esta parte se cambiara a "Usar la siguiente IP", y se escribirá una dirección IP donde los tres primeros

números deben coincidir con los que se designen en el programa previamente cargado en la tarjeta Arduino, y el ultimo numero deberá ser un numero entre 0 y 255 diferente al que tiene se asignó en el programa de la tarjeta.

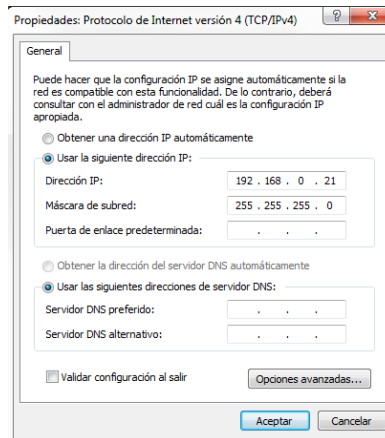


Figura 8. Propiedades Protocolo de Internet versión 4(TCP/IPv4).

Una vez realizada esta configuración se debe presionar aceptar en todas las ventanas que se abrieron anteriormente. Por ultimo para verificar que se tiene conexión con la tarjeta, se abre el Procesador de comandos en Windows “cmd.exe” y se ejecuta el comando ping “dirección IP de la tarjeta” en la figura 9 se muestra los resultados que se deben obtener, en caso de no obtener resultados parecidos, es probable que existan problemas con el cable con que se está usando o no se llevaron a cabo todos los pasos mencionados anteriormente.

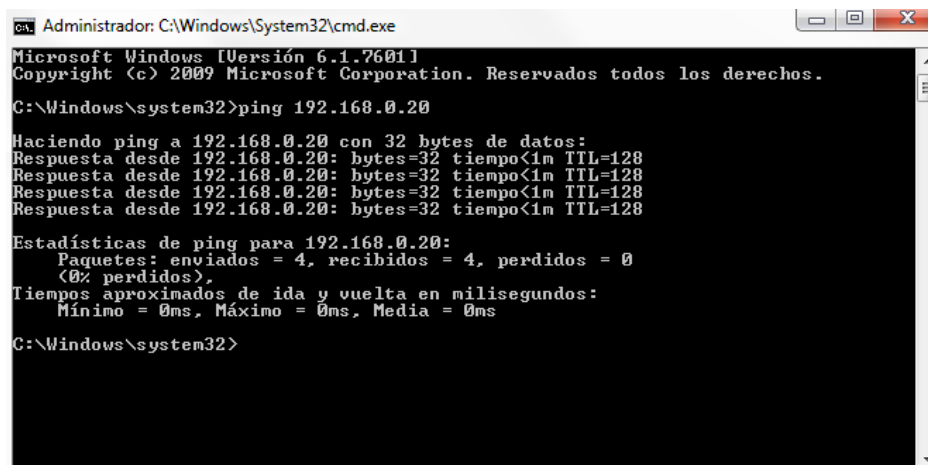


Figura 9. Prueba de conexión con tarjeta Arduino vía Ethernet

Una vez verificada la conexión entre la tarjeta Arduino y la computadora se puede acceder a la página web a través de la dirección IP de la tarjeta en cualquier navegador web. En la figura 10 se muestra la página web a través de un acceso punto a punto.



Figura 10. Acceso a página web.

Una vez consiguiendo el acceso a la página web de la tarjeta Arduino se puede comenzar con el monitoreo a partir de la interfaz gráfica desarrollada en Matlab. En la figura 11 se muestra la ventana principal donde se muestran los botones que acceden a ventanas de monitoreo de corriente alterna “AC” o corriente continua “DC”.

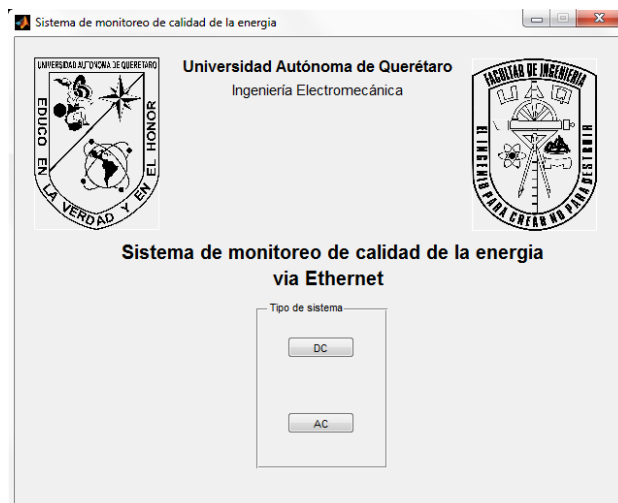


Figura 11. Acceso a página web.

Una vez seleccionando el monitoreo que se desea realizar aparecerá otra ventana en donde se seleccionan los parámetros que se desean obtener, además de la dirección URL de la página web. En el caso de conexión punto a punto será la dirección con la que cuente la tarjeta en los casos de las cajas de texto de activación y desactivación se agrega a la dirección “/?status=1” y “/?status=0”, respectivamente. Posteriormente, se establece el intervalo entre muestras y la duración de la medición. Por último se pulsa el botón iniciar, para comenzar con el monitoreo.

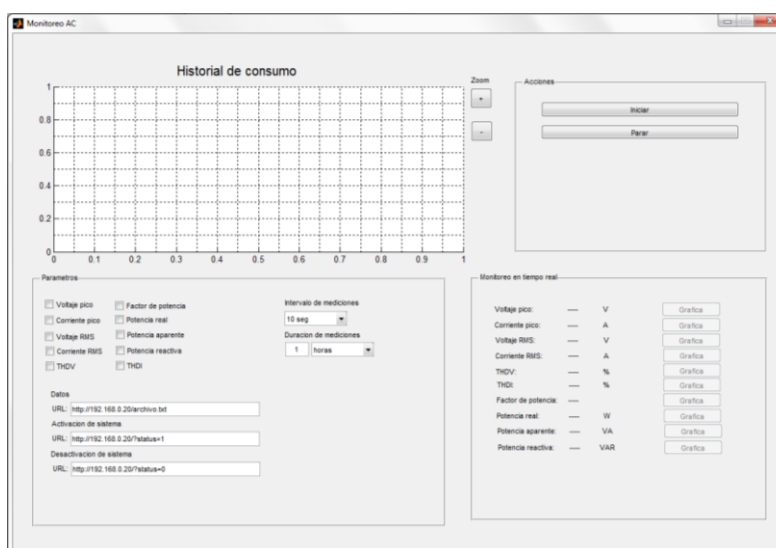


Figura 11. Interfaz gráfica para monitoreo de corriente alterna.

Para monitoreo de corriente continua se cuenta con una interfaz parecida, lo único que cambian son los parámetros que se pueden seleccionar en la medición.

Monitoreo en red local

Para realizar una conexión a través de una red de área local se requiere de un enrutador que cuente con protocolo de configuración dinámica de host DHCP, que en la actualidad son la mayoría que cuentan con este protocolo, este protocolo permite que los clientes de una red

IP obtengan sus parámetros automáticamente, para realizar esta conexión se deben hacer algunos cambios, por ejemplo en la definición de la dirección IP del código original se tiene la definición:

```
IPAddress ip(192,168,0,20);
```

Esta definición se cambia por:

```
IPAddress ip;
```

También dentro de la función Setup de Arduino se tiene la inicialización del servidor Ethernet de la siguiente manera:

```
Ethernet.begin(mac, ip);  
server.begin();
```

Este código deberá ser sustituido por el las siguientes líneas:

```
if (Ethernet.begin(mac) == 0) {  
    Serial.println("ERROR DHCP");  
}  
else {  
    ip=Ethernet.localIP();  
    Serial.print("Direccion IP asignada: ");  
    Serial.println(ip);  
    server.begin();  
}
```

Después de estas modificaciones se procede a programar la tarjeta Arduino, en donde para obtener la dirección a la que debemos acceder se debe tener la tarjeta Arduino conectada a la computadora por medio del cable USB y la conexión Ethernet se realiza por medio del enrutador. En el monitor serial del IDE de Arduino se mostrara la dirección IP asignada a la tarjeta Arduino para acceder a la página web, en la figura 12 se muestra la ventana del monitor serial de Arduino.

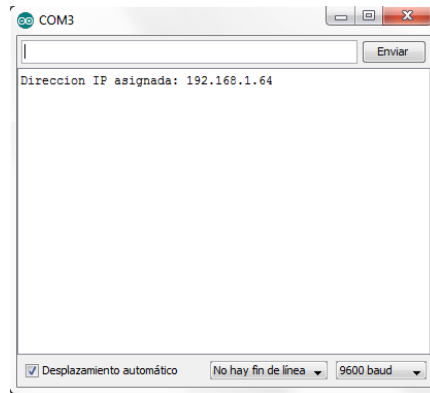


Figura 12. Dirección IP asignada por enrutador.

Una vez obtenida la dirección IP se puede desconectar la tarjeta Arduino del puerto de programación USB y es posible acceder a través de la red de área local. En la figura 13 se muestra un ejemplo de acceso a la página web, para realizar el monitoreo al igual que el caso anterior se ejecuta la interfaz gráfica en Matlab para comenzar con las adquisiciones.



Figura 13. Acceso a página web a través del área local.

Monitoreo a través de Internet

En caso de que se desee realizar monitoreo a través de Internet se sugiere que los intervalos entre mediciones sean mínimo de 5 minutos debido a que es una gran cantidad de datos la que tiene que ser transmitida. Entonces para realizar monitoreo a través de Internet se usa una librería proporcionada por Yaler.net, un servidor que proporciona acceso seguro de páginas web a dispositivos embebidos, además de la librería se requiere un registro para obtener un nombre de dominio gratuito que proporciona esta misma página. Una vez descargada la librería se debe copiar en la ruta de librerías en la dirección donde se instaló el IDE de Arduino. Para acceso a través de internet se hacen algunos cambios al igual que para el monitoreo a través de una red local, lo primero es incluir la librería en la declaración de librerías:

```
#include <YalerEthernetServer.h>
```

Posteriormente se elimina la declaración de la dirección IP y la declaración del puerto de comunicación que aparece a continuación:

```
IPAddress ip(192,168,0,20);  
EthernetServer server(80);
```

En lugar de las líneas anteriores se declara un servidor, en donde se incluirá el nombre de dominio generado previamente:

```
YalerEthernetServer server("try.yaler.net", 80, "NOMBRE DE DOMINIO");
```

Por último dentro de la función Setup se sustituyen las líneas de código:

```
Ethernet.begin(mac, ip);  
server.begin();
```

Por las líneas de código que se muestran debajo:

```
if (Ethernet.begin(mac) == 0) {  
    Serial.println("Failed.");  
} else {  
    Serial.println(Ethernet.localIP());  
}
```

```
server.begin();  
}
```

Para acceder a la página web como se muestra en la figura 14 para acceder a la página a través de Internet se escribe el nombre de dominio que se proporciona más el nombre del servidor que en este caso es “.try.yaler.net”. Con esto al igual que en los casos anteriores solo se deben sustituir las direcciones URL en la interfaz de grafica de Matlab dependiendo de cómo se desee acceder a las mediciones.



Figura 14. Acceso a página web a través de Internet.

Por ultimo para descartar fallas en el hardware se recomienda realizar antes que nada una prueba de comunicación punto a punto.

Anexo 2. Código de servidor embebido en Arduino

```
/*
#####
#                                UNIVERSIDAD AUTONOMA DE QUERETARO                                #
#Integracion de Monitoreo Ethernet a sistema de monitoreo basado en FPGA#
#                                página web embebida                                        #
#####
*/
#include <SdFat.h> //libreria para uso de micro-SD
#include <SPI.h> //Libreria para uso de SPI de Ethernet-Shield
#include <Ethernet.h> //Libreria para uso de Chip Ethernet

//Definicion de variables usadas para la comunicacion Ethernet
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED }; //Definicion de
direccion MAC
IPAddress ip(192,168,0,20); //Definicion de direccion ip
IPAddress gateway(192,168,1,1); //Definicion de puerta de enlace
IPAddress subnet(255, 255, 255, 0); //Definicion de mascara de subred
EthernetServer server(80); //Crea un servidor en el puerto 80 usado para
comunicaciones HTTP
#define REQ_BUF_SZ 20 //Define tamaño de buffer usado para guardar la
peticion HTTP
char HTTP_req[REQ_BUF_SZ] = {0}; //Buffer de peticion HTTP
char req_index = 0; //Contador de peticion HTTP

//Definicion de variables usadas para micro-SD
SdFat sd; //Variable para inicializar micro-SD
SdFile file; //Variable para escribir archivo
SdFile file2; //Variable para leer archivo
char name[] = "archivo.txt"; //Nombre del archivo a Leer/Escribir
//Definicion de pines logicos E/S usados
const int Enable = 7; // Habilitacion de FPGA a Arduino indica que esta
listo para transmitir los datos
const int CS_Ethernet = 10; //Pin Chip-Select para chip Ethernet W5100
const int CS_micro_SD = 4; //Pin Chip-Select para micro-SD
const int Flag1 = 8; //Bandera para escritura de datos via Serial
const int Start = 2; //Habilitacion de Arduino a FPGA que inicia que se
debe hacer una adquisicion

//Funcion de inicializacion de configuracion setup propia de Arduino
void setup()
{
    //Configuracion de E/S
    pinMode(CS_Ethernet, OUTPUT);
    digitalWrite(CS_Ethernet, HIGH);
    pinMode(CS_micro_SD, OUTPUT);
    pinMode(Enable, INPUT);
    pinMode(Flag1, OUTPUT);
    pinMode(Start, OUTPUT);
    //Configuracion de comunicacion serial
    Serial.begin(9600);
    Serial1.begin(115200);
    //Inicializacion de dispositivo Ethernet
    Ethernet.begin(mac, ip);
}
```

```

    server.begin(); //Inicializa servidor, listo para solicitud de
cliente
    //Inicializacion de tarjeta micro-SD
    sd.begin(CS_micro_SD,SPI_HALF_SPEED);
}

//Funcion de lectura de datos de FPGA
void LECTURA_SISTEMA()
{
    int i=0; //Variable para contador
    digitalWrite(Flag1, LOW);
    file.open(name, O_CREAT | O_WRITE); //Crea un archivo cada nueva
adquisicion
    while(i<3000){ //Guarda 3000 datos por cada adquisicion
        digitalWrite(Flag1, HIGH); //Activa bandera que indica al FPGA que
Arduino esta listo para recibir dato
        if(Serial1.available()>0){ //Si hay un dato que recibir
            digitalWrite(Flag1,LOW); // desactiva la bandera
            file.write(Serial1.read()); //y escribe el dato en el archivo
            i++;
        }
    }
    //Serial.println("Se termino de escribir");
    file.close(); //cierra el archivo
}

void loop()
{
    int TX_EN = digitalRead(Enable); //mientras se este escribiendo no es
posible acceder a la pagina web
    if(TX_EN)
    {
        digitalWrite(Flag1, LOW);
        LECTURA_SISTEMA(); // Accede a funcion de lectura
    }
    else
    {
        //En caso de que no se este escribiendo se puede acceder a la pagina
web
        EthernetClient client = server.available(); //Guarda estado del
servidor
        if (client) { //Verifica si hay un cliente
            boolean currentLineIsBlank = true; //Define variable para
identificar fin de requisicion la cual termina con una linea en blanco
            while (client.connected()) {
                if (client.available()) { //Si los tados del cliente estan
disponibles
                    char c = client.read(); //Lee 1 byte o caracter desde el
cliente
                    if (req_index < (REQ_BUF_SZ - 1)) {
                        HTTP_req[req_index] = c; // Salva los carateres del
requerimiento HTTP
                        req_index++;
                    }
                    //Serial.print(c); // imprime peticion HTTP

```

```

//La ultima linea de la peticion HTTP termina con una
linea en blanco la respuesta se envia solo hasta que se termino de leer
la requisicion
    if (c == '\n' && currentLineIsBlank) {
        //Envia respuesta HTTP
        client.println("HTTP/1.1 200 OK");
        client.println("Content-Type: text/html");
        //client.println("Connection: close");
        client.println();
        if (StrContains(HTTP_req, "GET /?status=1"))
            {
                digitalWrite(Start, HIGH); //Activacion
de sistema por parte de la pagina web
            }
            if (StrContains(HTTP_req, "GET /?status=0"))
            {
                digitalWrite(Start, LOW); //Desactivacion
de sistema por parte de la pagina web
            }
            if (StrContains(HTTP_req, "GET / ") ||
StrContains(HTTP_req, "GET /?status=1") || StrContains(HTTP_req, "GET
/?status=0")) {
                page1(client); //Envia Pagina 1
            }
            else if (StrContains(HTTP_req, "GET /archivo.txt"))
{
                archivo(client); //Envia contenido del archivo
            }
            req_index = 0;
            StrClear(HTTP_req, REQ_BUF_SZ); //Limpia los buffers
usados para almacenar las peticiones HTTP
            break;
        }
        //Cada linea recibida termina con \r\n
        if (c == '\n') {
            currentLineIsBlank = true;
        }
        else if (c != '\r') {
            currentLineIsBlank = false;
        }
    } //Finaliza if (client.available())
} //Finaliza while (client.connected())
delay(1); //Da tiempo al cliente de recibir los datos
client.stop(); //Cierra comunicacion
} //Finaliza if (client)
}}

// Limpia buffers utilizados
void StrClear(char *str, char length)
{
    for (int i = 0; i < length; i++) {
        str[i] = 0;
    }
}
// Busca la cadena sfind en una cadena str
// Retorna 1 si la cadena se encontro
// Retorna 0 si la cadena no se encontro

```

```

char StrContains(char *str, char *sfind)
{
    char found = 0;
    char index = 0;
    char len;
    len = strlen(str);
    if (strlen(sfind) > len) {
        return 0;
    }
    while (index < len) {
        if (str[index] == sfind[found]) {
            found++;
            if (strlen(sfind) == found) {
                return 1;
            }
        }
        else {
            found = 0;
        }
        index++;
    }
    return 0;
}

//***** Pagina principal*****
*****
void pagel(EthernetClient client)
{
    //Serial.println("Solicitud de archivo de pagina1"); //Indica que
    se accedio a la pagina 1
    //La funcion client.println imprime la respuesta http que es un
    documento html o pagina web
    client.println("<!DOCTYPE html>");
    client.println("<meta charset = \"utf-8\">");
    client.println("<head>");
    client.println("<title>Sistema de monitoreo Ethernet</title>");
    client.println("<style>");
    client.println("body{");
    client.println("background: #212121;");
    client.println("color: #FFF;");
    client.println("text-align: center;");
    client.println("}");
    client.println("header, footer{");
    client.println("background: #1E5281;");
    client.println("border-radius: 0.5em;");
    client.println("margin: 0.5em auto;");
    client.println("max-width: 960px;");
    client.println("padding: 0.25em;");
    client.println("text-align: center;");
    client.println("}");
    client.println("section#contenedor{");
    client.println("margin: 0.5em auto;");
    client.println("max-width: 960px;");
    client.println("padding: 0;");
    client.println("text-align: center;");
    client.println("}");
    client.println("section#principal, aside{");

```

```

client.println("background: #1E5281;");
client.println("border-radius: 0.5em;");
client.println("display: inline-block;");
client.println("margin: 0.5em auto;");
client.println("max-width: 960px;");
client.println("min-height: 200px;");
client.println("padding: 0.25em;");
client.println("text-align: center;");
client.println("vertical-align:top;");
client.println("width: 65%;");
client.println("{}");
client.println("aside{");
client.println("width: 30%;");
client.println("}");
client.println("</style>");
client.println("</head>");
client.println("<body>");
client.println("<header>");
client.println("<h1>UNIVERSIDAD AUTONOMA DE QUERETARO<h1>");
client.println("</header>");
client.println("<section id=\"contenedor\">");
client.println("<section id=\"principal\">");
client.println("<h2>Sistema de monitoreo de calidad de la
energia</h2>");
client.println("<br />");
client.println("El sistema monitoreo funciona por medio de
intervalos de monitoreo, en los cuales se manda una activacion del
sistema cada determinado tiempo, ");
client.println("adquiriendo 12 ciclos de señal (200ms) a una
frecuencia de muestreo de 4.8kHz. Esta adquisicion es guardada en un");
client.println("<a href=\"http://192.168.0.20/archivo.txt\">
archivo</a> que debe ser descargado con cada adquisicion.");
client.println("</section>");
client.println("<aside>");
client.println("<p>Activacion de sistema</p>");
//Dependiendo del estado del pin Start muestra el estado en el
que se encuentra el sistema
if (digitalRead(Start)){
client.print(" El sistema esta ACTIVADO");
}else{
client.print("El sistema esta DESACTIVADO");
}
client.println("<br />");
client.print("<FORM action=\"http://192.168.0.20/\">");
client.print("<P> <INPUT type=\"radio\" name=\"status\"
value=\"1\">Activa");
client.print("<P> <INPUT type=\"radio\" name=\"status\"
value=\"0\">Desactiva");
client.print("<P> <INPUT type=\"submit\" value=\"Enviar\">
</FORM>");
client.println("</aside>");
client.println("</section>");
client.println("<footer>");
client.println("Pagina web desarrollada para el sistema de
monitoreo de calidad de la energia");
client.println("</footer>");
client.println("</body>");

```

```

        client.println("</html>");
    }

    // Pagina web alterna creada a apartir de los datos adquiridos en el arch
    ivo de texto
    void archivo(EthernetClient client)
    {
        //Serial.println("Solicitud de archivo de texto"); //Indica que se
    solicito la pagina alterna
        if(file2.open(name,O_READ))//Abre archivo de texto guardado
        {
            //Manda cabecera de archivo html
            client.println("<html>");
            client.println("<title>archivo</title>");
            client.println("<body>");
            //Serial.println("El archivo se abrio correctamente");
            while(file2.available()){
                client.write("<br />");
                client.print(file2.read(),HEX);//añade los datos en formato HEXADECIMAL
            }
            file2.close(); //Cierra archivo
            client.println("</body>");
            client.println("</html>"); //Indica fin de la pagina web
        }
        else{

            //Serial.println("No se puede abrir archivo");//Si no existe o no se
        puede encontrar el archivo
        }
    }
}

```