

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA DE INGENIERÍA ELÉCTRICA



**Diseño, configuración y supervisión de la red de
medidores de energía eléctrica del campus central de
la Universidad de El Salvador.**

PRESENTADO POR:

JUAN JOSÉ BONILLA PERLA

PARA OPTAR AL TÍTULO DE:

INGENIERO ELECTRICISTA

CIUDAD UNIVERSITARIA, FEBRERO DEL 2014

UNIVERSIDAD DE EL SALVADOR

RECTOR :

ING. MARIO ROBERTO NIETO LOVO

SECRETARIA GENERAL :

DRA. ANA LETICIA ZAVALA DE AMAYA

FACULTAD DE INGENIERIA Y ARQUITECTURA

DECANO :

ING. FRANCISCO ANTONIO ALARCÓN SANDOVAL

SECRETARIO :

ING. JULIO ALBERTO PORTILLO

ESCUELA DE INGENIERÍA ELÉCTRICA

DIRECTOR :

ING. JOSÉ WILBER CALDERÓN URRUTIA

UNIVERSIDAD DE EL SALVADOR
FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA DE INGENIERÍA ELÉCTRICA

Trabajo de Graduación previo a la opción al Grado de:

INGENIERO ELECTRICISTA

Título :

Diseño, configuración y supervisión de la red de medidores de energía eléctrica del campus central de la Universidad de El Salvador.

Presentado por :

JUAN JOSÉ BONILLA PERLA

Trabajo de Graduación Aprobado por :

Docente Director :

Dr. CARLOS EUGENIO MARTÍNEZ CRUZ

San Salvador, Febrero de 2014

Trabajo de Graduación Aprobado por:

Docente Director :

Dr. CARLOS EUGENIO MARTÍNEZ CRUZ

ACTA DE CONSTANCIA DE NOTA Y DEFENSA FINAL

En esta fecha, 27 de noviembre de 2013, en la Sala de Lectura de la Escuela de Ingeniería Eléctrica, a las 5:30 horas, en presencia de las siguientes autoridades de la Escuela de Ingeniería Eléctrica de la Universidad de El Salvador:

1. Ing. José Wilber Calderón Urrutia
Director

Firma: Wilber Calderón

2. Ing. Salvador de Jesús Germán
Secretario

Firma: [Firma]



Y, con el Honorable Jurado de Evaluación integrado por las personas siguientes:

1- Ing. Mauricio Amílcar Ayala Arévalo

2- Ingra. Fantina Deruchette Orellana Alvarado

3- Ing. Jorge Alberto Zetino Chicas

Firma: [Firma]

Se efectuó la defensa final reglamentaria del Trabajo de Graduación.

Diseño, configuración y supervisión de la red de medidores de energía eléctrica del campus central de la Universidad de El Salvador.

A cargo del Bachiller:

- Bonilla Perla Juan José

Habiendo obtenido en el presente Trabajo una nota promedio de la defensa final, de: 9.3

(NUEVE PUNTO TRES)

Agradecimientos.

Primeramente quiero agradecer a Dios por permitirme llegar hasta el fin de mi carrera universitaria y por las muchas bendiciones que he disfrutado en mi vida.

A mis padres Rafael Antonio Bonilla y María Vicenta Perla de Bonilla, por su amor, animo, ejemplo y por el sacrificio que han hecho para poder terminar mi carrera universitaria.

A mis hermanos Rafael Bonilla y Keren Bonilla, por su cariño y palabras de ánimo en todo momento.

A mi novia Silvia Hernandez, por su paciencia y comprensión durante el desarrollo de este trabajo.

A mis amigos y compañeros de la Escuela de Ingeniería Eléctrica, gracias por los buenos momentos, exámenes y horas de estudio.

Al Dr. Carlos Martínez por su dirección en el desarrollo de este trabajo de graduación, sin sus consejos y guía no hubiera sido posible llevarlo a cabo.

A la Universidad de El Salvador en especial a la Escuela de Ingeniería Eléctrica, por todos los conocimientos compartidos a través sus catedráticos.

A mis compañero de trabajo en ETESAL y EOR, por sus amistad y consejos.

En fin a todos los que han hecho posible estar en terminar esta etapa de mi vida muchas gracias.

Que Dios los bendiga siempre.
Juan José Bonilla Perla.

Acrónimos

ADU:	Unidad de Aplicaciones de Datos.
ATA:	Analog Telephone Adapter.
B.A.T.M.A.N:	Better Approach To Mobile Ad-hoc Networking
BSSID:	Basic Service Set Identifier.
DNS:	Domain Name System.
WLAN:	Wireless Local Area Network.
EIE:	Escuela de Ingeniería Eléctrica.
FIA:	Facultad de Ingeniería y Arquitectura.
HMI:	Human Machine Interface.
HTML:	Hiper Text Markup Language.
IEEE:	Institute of Electrical and Electronics Engineers.
LAN:	Local Area Network.
MBAP:	MODBUS Application Protocol Header.
MP:	Mesh Potato.
NS2:	Nano Station 2.
OSI:	Open System Interconnection.
PC:	Computadora Personal.
PDU:	Unidad Simple de Datos.
PHP:	Personal Home Page Tools.
PLC:	Programmable Logic Controller.
SCADA:	Supervisory Control And Data Acquisition.
SMUES:	Sistema de Monitoreo Universidad de El Salvador.
SPUD:	Simple Unified Dashboard for Mesh Networks.
SSID:	Service Set IDentifier.
TCP/IP:	Transmission Control Protocol (TCP) and the Internet Protocol (IP).
THD:	Total Harmonic Distortion.
VIS:	Visualization Server.
VoIP:	Voice over Internet Protocol.
WIFI:	Wireless Fidelity.

Índice General

Capítulo 1: INTRODUCCIÓN	1
1.1. Interés de la investigación.....	1
1.2 Motivación.....	2
1.3. Objetivos del trabajo de graduación.....	2
1.4 Organización.....	3
Capítulo 2: RED DE MONITOREO UES	4
2.1. Esquema.....	5
2.2. Configuración de la red inalámbrica	6
2.3. Configuración de equipo y software	8
2.3.1. Configuración del supernodo.....	8
2.3.1.1. Configuración archivo network.....	8
2.3.1.2. Configuración archivo wireless	9
2.3.1.3. Configuración archivo batmand	10
2.3.2. Configuración del nodo repetidor.....	10
2.3.2.1. Configuración archivo network.....	11
2.3.2.2. Configuración archivo wireless	11
2.3.2.3. Configuración archivo batmand	12
2.3.3. Configuración de los nodos de medición.....	12
2.3.3.1. Configuración archivo network.....	13
2.3.3.2. Configuración archivo wireless	13
2.3.2.3. Configuración archivo batmand	14
2.3.4. Configuración LAN medidores	14
2.3.5. Configuración de red del servidor.....	17
2.3.5.1. Configuración de la tarjeta de red	17
2.3.5.2. Ruteo de la red	18
Capítulo 3: MONITOREO REMOTO DE MEDIDORES	19
3.1. Protocolo MODBUS.	19
3.1.1 Contexto del protocolo MODBUS	20
3.1.2 Descripción del protocolo.....	20
3.1.3. MODBUS TCP/IP	22
3.1.3.1 Modelo Cliente/Servidor.....	22
3.1.3.2 Descripción del protocolo MODBUS TCP/IP	22
3.1.3.2.1 Unidad de Aplicación de Datos en MODBUS TCP/IP	23
3.2. Librería Libmodbus y proceso de implementación para monitoreo.....	24

3.2.1. Instalación de Libmodbus	24
3.2.1. Modificación de la librería Libmodbus.....	25
3.3. Programa de consulta de datos	26
3.3.1. Estructura del programa en C	26
3.4. Base de datos.....	31
3.4.1. MySQL	31
3.4.2. phpMyAdmin	31
3.4.3. Instalación phpMyAdmin	31
3.5. SPUD	32
Capítulo 4: FUNCIONAMIENTO Y OPERACIÓN DE LA RED	33
4.1. Sistema de monitoreo UES (SMUES).....	36
4.2. Mediciones obtenidas en Subestación Medicina.....	37
4.2.1. Voltajes de fases.....	37
4.2.2. Corrientes de fases	37
4.2.3. Potencia Activa, Reactiva y Aparente	38
4.2.4. Factor de Potencia Trifásico	38
4.2.5. Factor de Potencia por fases.....	39
4.2.6. Frecuencia	39
4.2.7. Consumo de energía por hora	40
Capítulo 5: CONCLUSIONES Y LINEAS FUTURAS.	41
5.1. Conclusiones	41
5.2. Líneas futuras.....	42
ANEXOS	43
A.1. Instalación de OpenWrt, B.A.T.M.A.N. y servidor de visualización VIS en NS2	44
A.2. Instalación de SPUD.....	45
A.1.1. Instalación de las dependencias	45
A.2.2. Descarga del código fuente de SPUD.....	46
A.2.3. Base de datos.....	46
A.2.4. Cambiar permisos de usuarios de la carpeta app de la aplicación SPUD	47
A.2.5. Habilitar módulo rewrite de Apache	47
A.2.6. Editar el archivo de Apache2 000-default	47
A.2.7. Configurar servidor de visualización (VIS).....	48
A.2.8. Agregar tareas programadas del sistema la actualización del servidor de visualización	48
A.2.9. Iniciamos SPUD	49
A.3. Mapa MODBUS Electroindustries Shark 100S	50

A.3.1. Secciones de Mapa de Registro MODBUS	50
A.3.2. Formato de Datos	50
A.3.3. Mapa MODBUS.....	51
A.4. Mapa MODBUS Electroindustries Shark 200/200S.....	55
A.4.1. Secciones de Mapa de Registro MODBUS	55
A.4.2. Formato de Datos	55
A.4.3. Mapa MODBUS.....	56
A.5. Diccionario de datos	59
A.5.1. Lista de Tablas base de datos ‘spud’	59
A.5.2. Elementos Datos.....	61
A.5.2.1. Datos comunes para los medidores Shark 100S y Shark 200/200S	61
A.5.2.2. Datos exclusivos para los medidores Shark 100S.....	62
A.5.2.3. Datos exclusivos para los medidores Shark 200/200S	65
A.6. Medidor Shark 100S	68
A.6.1. Descripción del Hardware	68
A.6.2. Características del medidor Shark 100S.....	68
A.7. Medidor Shark 200S	70
A.7.1. Descripción del Hardware	70
A.7.2. Características del medidor Shark 200S.....	70
A.8. Datasheet Nanostation 2 [16].....	72
A.9. Datasheet Mesh Potato [33].....	74
BIBLIOGRAFÍA.....	76

Lista de Tablas

Tabla 1: Elementos de la red	6
Tabla 2: Medidores instalados campus central UES	33
Tabla 3: Mapa MODBUS medidores Shark 100S	54
Tabla 4: Mapa MODBUS medidores Shark 200/200S	58
Tabla 5: Tablas base de datos 'spud'	60
Tabla 6: Datos Comunes Shark 100S y Shark 200/200S	62
Tabla 7: Datos exclusivos Shark 100S.....	64
Tabla 8: Datos exclusivos Shark 200	67
Tabla 9: Parámetros obtenidos por el medidor Shark 100S.....	69
Tabla 10: Parámetros obtenidos por el medidor Shark 200S.....	71

Lista de figuras

Figura 1: Red de Medidores Universidad de El Salvador.....	5
Figura 2: NS2 en edificio de Escuela de Arquitectura.....	7
Figura 3: NS2 supernodo, instalado en el techo de la EIE.....	7
Figura 4: Configuración de fábrica tarjeta de red Shark 200.....	15
Figura 5: Configuración de fábrica tarjeta de red Shark 100S/200S.....	16
Figura 6: Pila de comunicación MODBUS.....	19
Figura 7: Ejemplo de arquitectura de red del protocolo MODBUS.....	20
Figura 8: Trama MODBUS.....	20
Figura 9: Transacción MODBUS sin errores y con errores.....	21
Figura 10: Modelo MODBUS TCP/IP.....	22
Figura 11: Arquitectura MODBUS TCP/IP.....	23
Figura 12: Requerimiento/Respuesta sobre MODBUS TCP/IP.....	23
Figura 13: phpMyAdmin.....	32
Figura 14: SPUD, Campus Universidad de El Salvador.....	32
Figura 15: Ubicación Medidores en campus central UES.....	34
Figura 16: Medidor Auditorium Química (subestación No. 21).....	35
Figura 17: Medidor Odontología 1 (subestación No. 24).....	35
Figura 18: Pantalla inicia SMUES.....	36
Figura 19: Pantalla consulta de datos SMUES.....	36
Figura 20: Voltajes Fases A, B y C, SE Medicina.....	37
Figura 21: Corrientes Fases A, B y C, SE Medicina.....	37
Figura 22: Potencia Activa, Reactiva y Aparente.....	38
Figura 23: Factor de Potencia trifásico, SE Medicina.....	38
Figura 24: Factor de Potencia por fases, SE Medicina.....	39
Figura 25: Frecuencia SE Medicina.....	39
Figura 26: Consumo de Energía por bloque horario, SE Medicina.....	40
Figura 27: OpenWrt en NS2.....	45
Figura 28: Pantalla de inicio SPUD.....	49
Figura 29: Medidor Electroindustries Shark 100S.....	68
Figura 30: Shark 100S y MP.....	69
Figura 31: Medidor Electroindustries Shark 200S.....	70

Capítulo 1: INTRODUCCIÓN

Entre los años 2004 y 2006 se llevaron a cabo en la Escuela de Ingeniería Eléctrica de la UES, dos importantes trabajos de graduación [1] [2]. Estos trabajos de graduación consistieron en la construcción de equipos de medición de variables eléctricas. El último de esos trabajos de graduación dotó al medidor de capacidad de comunicación a través del protocolo MODBUS TCP/IP. Dos años más tarde se desarrolló otro trabajo de graduación que también hizo uso del protocolo MODBUS [3]. De esa fecha a esta parte no se siguieron haciendo desarrollos en el área de comunicación del protocolo MODBUS.

También, vale la pena resaltar que una de las cuestiones que diferencia a este trabajo de graduación de los anteriores es que se ha trabajado con equipo manufacturado por fabricantes de reconocimiento mundial. Además, otra diferencia importante es que aquellos trabajos de graduación no pasaron de ser meros experimentos académicos. Este trabajo de graduación responde a una necesidad real requerida en la red de medidores ya instalada en la Universidad de El Salvador.

1.1. Interés de la investigación.

En el año 2012, la Universidad de El Salvador implementó una red de medición en treinta de sus subestaciones. Para lo cual se instalaron treinta medidores que monitorizan diferentes tipos de variables eléctricas. A cada medidor se le instaló un router inalámbrico tipo WiFi. Cada router forma parte de una red tipo malla, dotando al conjunto de la red de la suficiente redundancia para monitorizar los datos en un punto central de monitoreo. Veinte de los treinta medidores carecen de capacidad de almacenamiento de datos. Si bien es cierto permiten registrar el consumo acumulado de energía de las subestaciones, éstos se quedan cortos en el almacenamiento de otros parámetros como voltajes de fase y de línea, de corrientes, de distorsión armónica de voltaje y de distorsión armónica de corriente, entre otros.

Debido a esta limitación y aprovechando la infraestructura de red inalámbrica en la que operan los medidores se propuso dotarles de capacidad de almacenamiento. Esto se conseguiría mediante un mecanismo de interrogación continua. Para ello se aprovecharía que los medidores permiten la comunicación mediante el protocolo MODBUS.

La interrogación continua de los medidores no solo dotaría de capacidad de almacenamiento a los medidores que carecen de ella sino que aumentaría la capacidad de almacenamiento de los que ya la tienen. Los diez medidores con memoria tienen una capacidad de dos Megabyte (MB). Al mismo tiempo, tendríamos un mecanismo de respaldo de datos pues se contaría con los datos en la memoria del medidor y con los datos almacenados en el punto de monitoreo.

La monitorización y el almacenamiento de las variables eléctricas de treinta subestaciones de la Universidad de El Salvador permitirían realizar análisis más profundos sobre el consumo y la calidad de la energía eléctrica utilizada en el campus.

La monitorización remota de una red de medidores de energía eléctrica propuesta en este trabajo de graduación se constituiría en el núcleo de una serie de proyectos conducentes a implantar nuestro propio sistema de supervisión, control y adquisición de datos (SCADA, acrónimo en inglés de *Supervisory Control And Data Acquisition*)

Algunos programas SCADA mundialmente conocidos son: NI LabView [4], NI Lookout [5], GE Proficy HMI/SCADA [6], SIEMENS SIMATIC WinCC SCADA [7] [8]. Los principales consumidores de estos programas se encuentran en la industria de generación, de transmisión y de distribución de energía eléctrica. El uso de estos programas implica la adquisición de costosas licencias de Software. Por ejemplo, el uso de una licencia de NI Lookout en una subestación de transmisión tiene un costo de aproximadamente cinco mil dólares americanos [9]. También, es necesario la compra de tarjetas de adquisición de datos; cuyo costo oscila entre mil y dos mil dólares, dependiendo de la cantidad de puntos a monitorizar. Si se considera que el sistema de transmisión de El Salvador actualmente tiene veinticinco subestaciones, que es necesario monitorizar, el costo total se incrementa considerablemente.

Si bien es cierto que en este trabajo de graduación nunca se buscó el desarrollar un sistema SCADA, los resultados obtenidos permiten pensar en la posibilidad de poder desarrollar uno. Este trabajo de graduación será la base de futuros desarrollos conducentes a obtener un sistema SCADA construido en la Universidad de El Salvador.

1.2 Motivación.

En los últimos diez años el valor promedio de la factura mensual de electricidad de la Universidad de El Salvador se ha triplicado. Se ha pasado de máximos mensuales cercanos a los US\$ 40,000.00, en el año 2002, a máximos mensuales superiores a los US\$ 120,000.00, en el año 2012. Ese crecimiento continuo y sostenido a lo largo de estos diez años se debe a dos razones. La primera, es el incremento del costo de la energía eléctrica en los últimos diez años. La segunda razón, se debe al incremento en el consumo de energía eléctrica. Por ejemplo, la carga correspondiente a aire acondicionado se ha multiplicado por veinte. En el año 2004 la UES tenía 1,059 toneladas en equipo de aire acondicionado. El censo levantado en el año 2012 constató más de 19,000 toneladas en equipo de aire acondicionado.

1.3. Objetivos del trabajo de graduación.

1. Investigar e implementar el protocolo MODBUS, utilizando la librería de C libmodbus.
2. Configurar la comunicación de los medidores Shark 100S, Shark 200 y Shark 200S
3. Diseñar, implementar y configurar la red de comunicaciones tipo malla.
4. Crear el programa de interrogación de los medidores.
5. Guardar los datos en una base de datos MySql.
6. Instalación del software de monitoreo de red SPUD (de inglés *Simple Unified Dashboard*).
7. Realizar pruebas de cobertura de los diferentes nodos que conforman la red.

1.4 Organización

En el capítulo 1, se desarrolla el interés de la investigación, motivación y objetivos del trabajo de graduación. En el capítulo 2, se desarrolla la configuración de los equipos que forman parte de la red de medidores de energía de la Universidad de El Salvador. En el capítulo 3, se explica el protocolo MODBUS, se desarrolla el programa de interrogación y almacenamiento de las mediciones, así como el uso del SPUD para el monitoreo de la red. En el capítulo 4, se presenta el resultado de la implementación de la red Mesh, del programa de interrogación y se presentan mediciones realizadas. Finalmente en el capítulo 5 se presentan las conclusiones y líneas futuras de investigación.

Capítulo 2: RED DE MONITOREO UES

Durante los últimos cinco años, en la Escuela de Ingeniería Eléctrica de la Universidad de El Salvador se han realizado varios trabajos de graduación en el área de redes inalámbricas [10] [11] [12]. Esa experiencia en materia de redes WiFi permitió que se modificara el diseño original de la red de medidores del proyecto “Modernización de la red de distribución eléctrica de la Universidad de El Salvador”, efectuado durante los años 2011 - 2012. Originalmente se consideró que los datos de los medidores se recabarían mediante inspección visual. En otras ocasiones, y con menos frecuencia, los datos se recogerían mediante la interconexión directa de una computadora al medidor.

La inspección visual tiene el inconveniente de requerir el desplazamiento de personal. Por otra parte, la interconexión de equipo informático a medidores interconectados en la red de distribución conduce a algunos riesgos. Por ejemplo, choques eléctricos ante la exposición a alta tensión. También, se corre el riesgo de dañar el equipo por uso inadecuado de la conexión del computador al medidor.

La modificación al proyecto original consistió en dotar a la red de medidores de comunicación inalámbrica. A cada medidor se le añadió un router WiFi, con capacidad de formar una red inalámbrica tipo malla. Se aprovechó un tipo de router inalámbrico WiFi, desarrollado en principio para comunicación VoIP [13] [14]. Este router permitió la interconexión con su vecino más próximo de tal manera que cada router sea un nodo de una gran malla. Por consiguiente, ya que cada medidor está asociado a un router se tiene una única red de medidores.

Una vez establecida la infraestructura para la comunicación inalámbrica, entre los medidores, routers y servidores, se aprovechó que los medidores están dotados con la capacidad de comunicación mediante el protocolo MODBUS.

El protocolo de comunicación MODBUS es el protocolo abierto de facto en la industria de automatización. En este trabajo se le ha utilizado únicamente para adquirir datos.

2.1. Esquema.

El router utilizado para la conformación de la red tipo malla fue el router MP01 [15]. Originalmente este router fue pensado para dotar de telefonía y acceso a internet a zonas rurales. Sin embargo mediante la adecuada modificación en su configuración éste trabajo de graduación le utilizó como un medio de transporte de variables eléctricas.

En la Figura 1, se muestra el diagrama de la red malla implementa en el campus central de la Universidad de El Salvador. Ésta permite la monitorización remota de treinta subestaciones donde se instalaron medidores. Se instalaron 20 medidores Shark 100S, 9 medidores Shark 200S y 1 medidor Shark 200

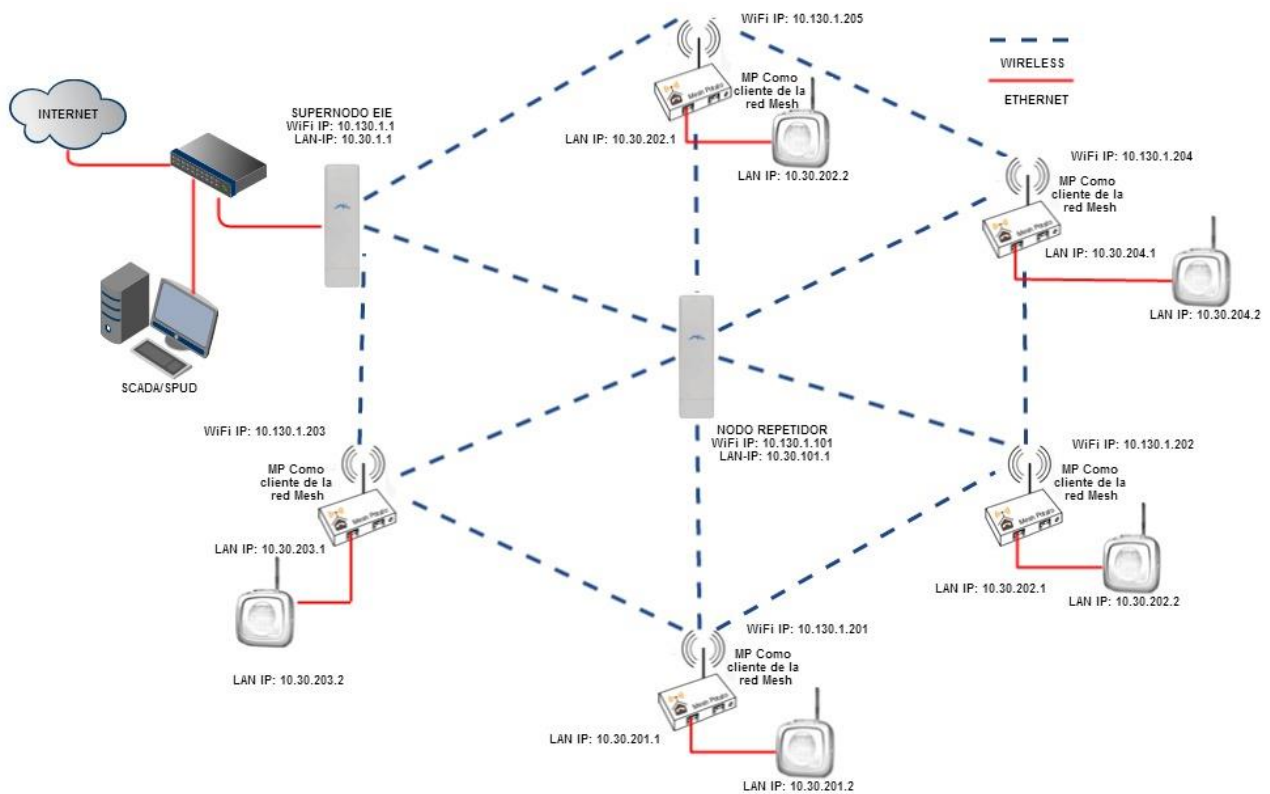


Figura 1: Red de Medidores Universidad de El Salvador

2.2. Configuración de la red inalámbrica

La red de medidores de la energía eléctrica está compuesta por nodos de medición, nodo repetidor en el edificio de la escuela de Arquitectura y un supernodo instalado en el techo de la Escuela de Ingeniería Eléctrica en la FIA.

La Tabla 1 muestra todos los elementos que conforman la red de medidores

Elemento	Descripción
 <p>Supernodo</p>	<p>El NS2 [16] mostrado es un router inalámbrico que funciona en la banda ISM de 2.4 GHz, entre sus cualidades está el poseer un sistema combinado de cuatro antenas de gran ganancia, también está diseñado para soportar condiciones extremas en ambientes al aire libre. Debido a su gran ganancia y resistencia a condiciones extremas, se utiliza en la red de medidores como el nodo principal o supernodo. Para utilizarlo como supernodo es necesario cambiar el firmware de fábrica por una versión Linux especial para routers llamada OpenWrt [17], que permite la instalación del protocolo de enrutamiento B.A.T.M.A.N [18], que se encarga de controlar el tráfico de paquetes entre nodos de la red malla.</p>
 <p>Nodos de medición</p>	<p>El equipo MP, es un router inalámbrico diseñado por Village Telco [13], el propósito principal de este router es servir como punto VoIP y de internet, tiene por lo tanto ATA VoIP y un puerto LAN, trabaja en la banda de 2.4 Ghz, el diseño del hardware es abierto así como también su firmware, está preparado para ser instalado al aire libre. Este equipo ya tiene instalado de fábrica el sistema operativo para routers OpenWrt Kamikaze [17], el paquete del protocolo de enrutamiento B.A.T.M.A.N. y el paquete para VoIP asterisk [19], En nuestra red sirve como nodo de conexión los medidores</p>
<p>Nodo repetidor</p>	<p>Al igual que el supernodo, para el nodo repetidor utilizamos un NS2 está instalado en el techo de la Escuela de Arquitectura, cumple dos propósitos: el servir de nodo repetidor y mejorar la confiabilidad de la red mallada. Se escogió este punto por ser uno de los más altos en la Facultad de Ingeniería y Arquitectura, tener línea vista con muchos puntos de medición y estar cerca de la Escuela de Ingeniería Eléctrica.</p>
 <p>Router</p>	<p>El router Linksys WRT54GS, se utiliza solo como switch en el que unimos la red Mesh con el servidor y el internet que sirve para presentar el mapa de la UES en el sistema SPUD.</p>
 <p>Servidor</p>	<p>En la Escuela de Ingeniería Eléctrica, está instalado el servidor, que cuentan con el sistema operativo Ubuntu 12.04, en el guardamos las mediciones de las subestaciones en una base de datos y realizamos el monitoreo de la red inalámbrica MESH, también se pueden configurar los medidores Shark 100S y 200/200S con el programa Communicator EXT [20] que proporciona el fabricante, esto está instalado en una máquina virtual con Windows 7.</p>

Tabla 1: Elementos de la red



Figura 2: NS2 en edificio de Escuela de Arquitectura

El NS2 instalado en una esquina de la azotea del edificio de la Escuela de Arquitectura como se muestra en la Figura 2; tiene la función de repetidor de la red Mesh. Se escogió este punto por ser uno de los más altos en la Facultad de Ingeniería y Arquitectura, y estar cerca de la Escuela de Ingeniería Eléctrica

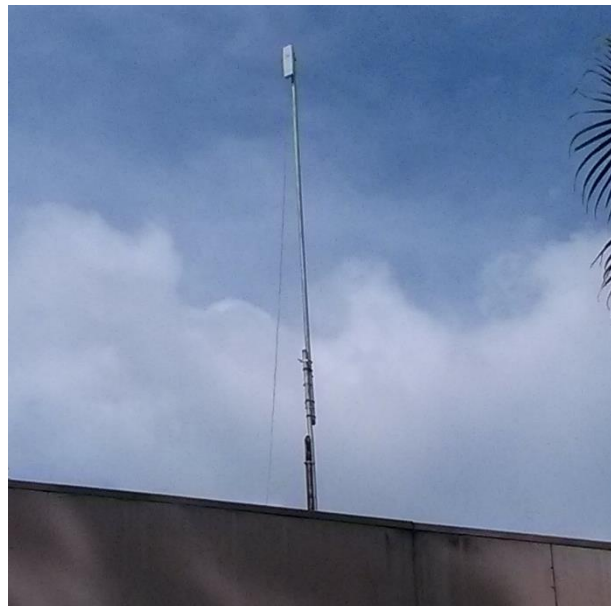


Figura 3: NS2 supernodo, instalado en el techo de la EIE

El NS2 instalado en el techo de la Escuela de Ingeniería Eléctrica se muestra en la Figura 3, es el supernodo de la red, también es el nodo que conecta la red con el servidor que interroga los medidores y guarda los datos de las mediciones de las subestaciones.

2.3. Configuración de equipo y software

A continuación se muestran los pasos a seguir para la configuración de los equipos que forman parte de la red inalámbrica. Se muestra como se deben configurar el NS2 y el MP para que trabajen como una sola red, también se explica la configuración del enrutamiento en el servidor.

2.3.1. Configuración del supernodo

El supernodo es parte esencial de la red de medidores de energía eléctrica, su función es la de:

- Ejecutar el servidor de visualización VIS de la red.
- Servir de Gateway de la red mallada 10.30.1.x
- Servir de punto de acceso del servidor a la red de medidores

Para el supernodo utilizamos el router NS2 por las siguientes razones:

- Mayor ganancia de su antena, permitiendo tener mayor alcance. Permite conectarnos con la mayor cantidad de nodos posibles.
- Diseñado para ser instalado en ambientes exteriores soportando condiciones extremas.

Para configurar el NS2 como supernodo, el primer paso es cambiar el firmware de fábrica por el firmware OpenWrt, instalar el protocolo de enrutamiento B.A.T.M.A.N. y el servidor de visualización VIS, estos pasos están detallados en el Anexo A.1.

Una vez instalado el firmware debemos modificar los siguientes archivos de configuración, el editor de texto puede ser VI o nano que están incluidos en la distribución OpenWrt: network, Wireless y batmand.

2.3.1.1. Configuración archivo network

En este archivo se configuran los parámetros de las interfaces utilizadas por el supernodo: tipo de interface, protocolo, dirección IP, máscara de red, puerta de enlace y DNS. Se utilizan las interfaces eth0 y eth0:1 que son asociadas al puerto LAN y ath0 asociada a la conexión WLAN, el archivo de configuración se encuentra en la ubicación **/etc/config/network**, debe estar escrito como se muestra a continuación:

```

root@G1:~# vi /etc/config/network

config 'interface' 'loopback'
    option 'ifname' 'lo'
    option 'proto' 'static'
    option 'ipaddr' '127.0.0.1'
    option 'netmask' '255.0.0.0'

config 'interface' 'lan'
    option 'ifname' 'eth0'
    option 'proto' 'static'
    option 'ipaddr' '192.168.1.100'
    option 'netmask' '255.255.255.0'
    option 'gateway' '192.168.1.1'
    option 'dns' '192.168.1.1'

config alias
    option 'interface' 'lan'
    option 'proto' 'static'
    option 'ipaddr' '10.30.1.1'
    option 'netmask' '255.255.255.0'

config 'interface' 'wifi0'
    option 'ifname' 'ath0'
    option 'proto' 'static'
    option 'ipaddr' '10.130.1.1'
    option 'netmask' '255.255.255.0'

```

2.3.1.2. Configuración archivo wireless

En el archivo `wireless`, se configuran los parámetros de la conexión WiFi: canal de operación, dispositivo, encriptación, ssid, modo, bssid entre otros. El archivo se encuentra ubicado en `/etc/config/wireless`, debe estar escrito como se muestra a continuación:

```

root@G1:~# vi /etc/config/wireless

config 'wifi-device' 'wifi0'
    option 'type' 'atheros'
    option 'channel' '11'

config 'wifi-iface'
    option 'device' 'wifi0'
    option 'encryption' 'none'
    option 'ssid' 'potato'
    option 'mode' 'ahdemo'
    option 'bssid' '01:CA:FF:EE:BA:BE'
    option 'swmerge' 1
    option 'bgscan' 0
    option 'network' 'wifi0'

```

2.3.1.3. Configuración archivo batmand

En el archivo batmand, se configuran los parámetros del protocolo de enrutamiento B.A.T.M.A.N., también se le indica al router quien es el servidor de visualización (VIS) de la red. El archivo se encuentra ubicado en **/etc/config/batmand**, debe estar escrito como se muestra a continuación:

```
root@G1:~# vi /etc/config/batmand

config 'batmand' 'general'
    option 'interface' 'ath0'
    option 'hna' '10.30.1.0/24'
    option 'originator_interval' ''
    option 'preferred_gateway' ''
    option 'policy_routing_script' ''
    option 'disable_client_nat' ''
    option 'disable_aggregation' ''
    option 'gateway_class' '5000'
    option 'routing_class' ''
    option 'visualisation_srv' '10.130.1.1'
```

Luego habilitamos que el servidor DNS del supernodo inicie al momento de encenderlo, para ello colocamos un enlace simbólico del archivo dnsmasq en la ubicación **/etc/init.d/dnsmasq**

De la siguiente manera:

```
root@G1:~#/etc/rc.d# ln -s ../init.d/dnsmasq S60dnsmasq
```

Reiniciamos el router con el comando reboot, con esto queda configurado el supernodo y listo para conectarse a los demás nodos

2.3.2. Configuración del nodo repetidor

El nodo repetidor está implementado en un NS2, ubicado en el techo de la Escuela de Arquitectura. A diferencia de trabajos anteriores [21] en donde se configuran los nodos para que den acceso a internet, para nuestra aplicación solo es necesaria la configuración de acceso a la red que se configura en los siguientes archivos: network, Wireless y batmand

Lo primero que demos hacer es sustituir el firmware del NS2 por OpenWrt e instalar los paquetes del protocolo de enrutamiento B.A.T.M.A.N., los pasos a seguir se muestran en el Anexo A.1.

2.3.2.1. Configuración archivo network

El archivo de configuración se encuentra en la ubicación **/etc/config/network**, debe estar escrito como se muestra a continuación:

```
root@G1:~# vi /etc/config/network

config 'interface' 'loopback'
    option 'ifname' 'lo'
    option 'proto' 'static'
    option 'ipaddr' '127.0.0.1'
    option 'netmask' '255.0.0.0'

config 'interface' 'lan'
    option 'ifname' 'eth0'
    option 'proto' 'static'
    option 'ipaddr' '192.168.1.101'
    option 'netmask' '255.255.255.0'
    option 'gateway' '192.168.1.1'
    option 'dns' '192.168.1.1'

config alias
    option 'interface' 'lan'
    option 'proto' 'static'
    option 'ipaddr' '10.30.101.1'
    option 'netmask' '255.255.255.0'

config 'interface' 'wifi0'
    option 'ifname' 'ath0'
    option 'proto' 'static'
    option 'ipaddr' '10.130.1.101'
    option 'netmask' '255.255.255.0'
```

2.3.3.2. Configuración archivo wireless

El archivo se encuentra ubicado en **/etc/config/wireless**, debe estar escrito como se muestra a continuación:

```
root@G1:~# vi /etc/config/wireless

config 'wifi-device' 'wifi0'
    option 'type' 'atheros'
    option 'channel' '11'

config 'wifi-iface'
    option 'device' 'wifi0'
    option 'encryption' 'none'
    option 'ssid' 'potato'
    option 'mode' 'ahdemo'
    option 'bssid' '01:CA:FF:EE:BA:BE'
    option swmerge 1
    option bgscan 0
    option 'network' 'wifi0'
```

2.3.2.3. Configuración archivo batmand

El archivo se encuentra ubicado en **/etc/config/batmand**, debe estar escrito como se muestra a continuación:

```
root@G1:~# vi /etc/config/batmand

config 'batmand' 'general'
    option 'interface' 'ath0'
    option 'hna' '10.30.101.0/24'
    option 'originator_interval' ''
    option 'preferred_gateway' ''
    option 'policy_routing_script' ''
    option 'disable_client_nat' ''
    option 'disable_aggregation' ''
    option 'gateway_class' '5000'
    option 'routing_class' ''
    option 'visualisation_srv' '10.130.1.1'
```

Reiniciamos el router con el comando `reboot`, con esto queda configurado el nodo de la EIE, listo para conectarse el supernodo y los demás nodos de la red.

2.3.3. Configuración de los nodos de medición

A continuación como ejemplo, se muestra la configuración de un nodo de medición, que nos permiten conectarnos con los medidores.

El router utilizado es el MP [13], este router tiene instalado OpenWrt y el protocolo B.A.T.M.A.N. de fábrica está diseñado para proveer datos y telefonía VoIP, por esa razón incluye un puerto LAN para datos y un puerto ATA para telefonía.

Igual que con la configuración del NS2, los archivos que se modifican son los siguientes: `network`, `Wireless` y `batmand`.

2.3.3.1. Configuración archivo network

El archivo de configuración se encuentra en la ubicación `/etc/config/network`, debe estar escrito como se muestra a continuación:

```
config 'interface' 'loopback'
    option 'ifname' 'lo'
    option 'proto' 'static'
    option 'ipaddr' '127.0.0.1'
    option 'netmask' '255.0.0.0'

config 'interface' 'lan'
    option 'ifname' 'eth0'
    option 'proto' 'static'
    option 'ipaddr' '192.168.1.201'
    option 'netmask' '255.255.255.0'
    option 'gateway' '192.168.1.1'
    option 'dns' '192.168.1.1'

config alias
    option 'interface' 'lan'
    option 'proto' 'static'
    option 'ipaddr' '10.30.201.1'
    option 'netmask' '255.255.255.0'

config 'interface' 'wifi0'
    option 'ifname' 'ath0'
    option 'proto' 'static'
    option 'ipaddr' '10.130.1.201'
    option 'netmask' '255.255.255.0'
```

2.3.3.2. Configuración archivo wireless

El archivo se encuentra ubicado en `/etc/config/wireless`, debe estar escrito como se muestra a continuación:

```
root@G1:~# vi /etc/config/wireless

config 'wifi-device' 'wifi0'
    option 'type' 'atheros'
    option 'channel' '11'

config 'wifi-iface'
    option 'device' 'wifi0'
    option 'encryption' 'none'
    option 'ssid' 'potato'
    option 'mode' 'ahdemo'
    option 'bssid' '01:CA:FF:EE:BA:BE'
    option 'swmerge' 1
    option 'bgscan' 0
    option 'network' 'wifi0'
```

2.3.2.3. Configuración archivo batmand

El archivo se encuentra ubicado en `/etc/config/batmand`, debe estar escrito como se muestra a continuación:

```
root@G1:~# vi /etc/config/batmand

config 'batmand' 'general'
    option 'interface' 'ath0'
    option 'hna' '10.30.201.0/24'
    option 'originator_interval' ''
    option 'preferred_gateway' ''
    option 'policy_routing_script' ''
    option 'disable_client_nat' ''
    option 'disable_aggregation' ''
    option 'gateway_class' '5000'
    option 'routing_class' ''
    option 'visualisation_srv' '10.130.1.1'
```

2.3.4. Configuración LAN medidores

La dirección IP y máscara de red de fábrica de los medidores Shark 100S y Shark 200S es la siguiente [22] [23] [24]:

Dirección IP: 10.0.0.1
Máscara de red: 255.255.255.0

Para el medidor Shark 200 la dirección IP y máscara de red son los siguientes [25]:

Dirección IP: 10.0.0.2
Máscara de red: 255.255.255.0

De manera que lo primero que debemos hacer para configurar la tarjeta de red del medidor es configurar la tarjeta de la PC con la dirección IP 10.0.0.2 y máscara de red 255.255.255.0, cuando configuramos un medidor Shark 200 cambiamos la dirección IP de la PC a 10.0.0.3 y máscara de red 255.255.255.0, como se muestra a continuación:

```
jbonilla@jbonilla-VirtualBox:~$ sudo ifconfig eth0 10.0.0.2/24
```

Después iniciamos una sesión telnet al medidor con su dirección IP a través del puerto 9999, como se muestra a continuación:

```
jbonilla@jbonilla-VirtualBox:~$ telnet 10.0.0.1 9999
```

A continuación en la Figura 4, se muestran los valores por defecto de la tarjeta de red del medidor Shark 200:

```
1) Network/IP Settings:
   IP Address ..... 10.0.0.1
   Default Gateway ..... -- not set --
   Netmask ..... 255.255.255.0
2) Serial & Mode Settings:
   Protocol ..... Modbus/RTU,Slave(s) attached
   Serial Interface ..... 57600,8,N,1,RS232,CH1
3) Modem/Configurable Pin Settings:
   CP1 ..... Not Used
   CP2 ..... Not Used
   CP3 ..... Not Used
4) Advanced Modbus Protocol settings:
   Slave Addr/Unit Id Source .. Modbus/TCP header
   Modbus Serial Broadcasts ... Disabled (Id=0 auto-mapped to 1)
   MB/TCP Exception Codes ..... Yes (return 00AH and 00BH)
   Char, Message Timeout ..... 00050msec, 05000msec

D)efault settings, S)ave, Q)uit without save
Select Command or parameter set (1..4) to change:
```

Figura 4: Configuración de fábrica tarjeta de red Shark 200

Para configurar la dirección IP, puerta de enlace y máscara de red cambiamos los parámetros del grupo No.1 introducimos el número "1" y programamos nuestros parámetros, después introducimos la letra "S" para guardar los cambios, la configuración de las los grupos Nos.2, 3 y 4 no se debe modificar.

Para el medidor Shark 100S/200S, existen seis bloques de configuración. En la Figura 5 se muestran los valores por defecto.

```
1) Network/IP Settings:
  Network Mode.....Wired Only
  IP Address ..... 10.0.0.1
  Default Gateway ..... --- not set ---
  Netmask .....255.255.255.0
2) Serial & Mode Settings:
  Protocol ..... Modbus/RTU,Slave(s) attached
  Serial Interface ..... 57600,8,N,1,RS232,CH1
3) Modem/Configurable Pin Settings:
  CP0..! Defaults (In) CP1..! GPIO (In)  CP2..! GPIO (In)
  CP3..! GPIO (In)  CP4..! GPIO (In)  CP5..! GPIO (In)
  CP6..! GPIO (In)  CP7..! GPIO (In)  CP8..! GPIO (In)
  CP9..! GPIO (In)  CP10.! GPIO (In)
  RTS Output ..... Fixed High/Active
4) Advanced Modbus Protocol settings:
  Slave Addr/Unit Id Source .. Modbus/TCP header
  Modbus Serial Broadcasts ... Disabled (Id=0 auto-mapped to 1)
  MB/TCP Exception Codes ..... Yes (return 00AH and 00BH)
  Char, Message Timeout ..... 00050msec, 05000msec
6) WLAN Settings:
  WLAN ..... Disabled, network:LTRX_IBSS
  Topology..... AdHoc, Country: US, Channel: 11
  Security..... none
  TX Data rate..... 11 Mbps auto fallback
  Power management..... not supported in ad hoc mode
```

Figura 5: Configuración de fábrica tarjeta de red Shark 100S/200S

Para configurar la dirección IP, puerta de enlace y mascara de red cambiamos los parámetros del grupo No.1 introducimos el número "1" y programamos nuestros parámetros, después introducimos la letra "S", para guardar los cambios. En el bloque No.6 se programa la conexión WiFi, por defecto esta deshabilitada y de esa manera la dejaremos. La configuración de las los grupos Nos.2, 3, 4 y 5 no se debe modificar.

Los medidores están comunicados a través de una red LAN con los MPs, se conectan a través de la del alias o la interface eth0:1, de manera que la dirección IP del MP le sirve de puerta de enlace, por ejemplo: el medidor que está conectado al MP del primer nodo tiene la siguiente configuración:

Dirección IP: 10.30.201.2
Mascara de red: 255.255.255.255
Puerta de enlace: 10.30.201.1

Con estos parámetros hemos creado una red LAN entre el alias eth0:1 del MP y el medidor.

2.3.5. Configuración de red del servidor

En el lado del servidor es necesario realizar configuraciones de la tarjeta de red y de ruteo para tener acceso a la red inalámbrica y a las sub redes de los medidores. Con esto podemos interrogar los medidores y administrar la red inalámbrica.

2.3.5.1. Configuración de la tarjeta de red

Para la configuración de la tarjeta de red del servidor se han implementado dos direcciones IP. Esto es posible gracias a la flexibilidad que el sistema operativo nos ofrece. Se utiliza una técnica llamada IP Aliasing [26]. IP Aliasing nos permite asignar a una interface física de red tantas direcciones IP como necesitemos.

En la computadora modificamos el archivo **interfaces**, en la ruta **/etc/network/**, asignándole la dirección IP de la interface eth0, como se muestra a continuación:

```
jbonilla@jbonilla-VirtualBox:/etc/network$ vi interfaces

auto eth0
iface eth0 inet static
address 192.168.1.130
netmask 255.255.255.0
gateway 192.168.1.1
```

2.3.5.2. Ruteo de la red

Para el ruteo de la red se ha creado un batch script llamado **potato**, en la ruta **/etc/network/if-up.d**, asignándole la dirección IP de la interface virtual eth0:0 y el ruteo de las treinta redes de los medidores, de manera que siempre que inicie la computadora realice el ruteo, a continuación se muestra el archivo potato:

```
jbonilla@jbonilla-VirtualBox:/etc/network/if-up.d$ vi potato

#!/bin/sh
# Agregando interfaces y rutas para conectar la red potato

ifconfig eth0:0 10.30.1.2 netmask 255.255.255.0

#Agregando las rutas de los medidores

#Enrutando la red inalambrica
route add -net 10.130.1.0/24 gw 10.30.101.1

#Enrutando las subredes de los medidores
route add -net 10.30.1.0/24 gw 10.30.101.1
route add -net 10.30.201.0/24 gw 10.30.101.1
route add -net 10.30.202.0/24 gw 10.30.101.1
route add -net 10.30.203.0/24 gw 10.30.101.1
route add -net 10.30.204.0/24 gw 10.30.101.1
route add -net 10.30.205.0/24 gw 10.30.101.1
route add -net 10.30.206.0/24 gw 10.30.101.1
route add -net 10.30.207.0/24 gw 10.30.101.1
route add -net 10.30.208.0/24 gw 10.30.101.1
route add -net 10.30.209.0/24 gw 10.30.101.1
route add -net 10.30.210.0/24 gw 10.30.101.1
route add -net 10.30.211.0/24 gw 10.30.101.1
route add -net 10.30.212.0/24 gw 10.30.101.1
route add -net 10.30.213.0/24 gw 10.30.101.1
route add -net 10.30.214.0/24 gw 10.30.101.1
route add -net 10.30.215.0/24 gw 10.30.101.1
route add -net 10.30.216.0/24 gw 10.30.101.1
route add -net 10.30.217.0/24 gw 10.30.101.1
route add -net 10.30.218.0/24 gw 10.30.101.1
route add -net 10.30.219.0/24 gw 10.30.101.1
route add -net 10.30.220.0/24 gw 10.30.101.1
route add -net 10.30.221.0/24 gw 10.30.101.1
route add -net 10.30.222.0/24 gw 10.30.101.1
route add -net 10.30.223.0/24 gw 10.30.101.1
route add -net 10.30.224.0/24 gw 10.30.101.1
route add -net 10.30.225.0/24 gw 10.30.101.1
route add -net 10.30.226.0/24 gw 10.30.101.1
route add -net 10.30.227.0/24 gw 10.30.101.1
route add -net 10.30.228.0/24 gw 10.30.101.1
route add -net 10.30.229.0/24 gw 10.30.101.1
route add -net 10.30.230.0/24 gw 10.30.101.1
```

Capítulo 3: MONITOREO REMOTO DE MEDIDORES

En el monitoreo remoto de los medidores de las subestaciones del campus central de la Universidad de El Salvador, se utilizan varias tecnologías. Entre las que se encuentran el protocolo de comunicación MODBUS [27], la librería de C Libmodbus para la interrogación de los medidores [28], sistema operativo GNU/LINUX, base de datos MySQL [29] y php [30] .

También se monitorea el estado de los enlaces inalámbricos entre los nodos de medición, nodo repetidor y supernodo, esto se hace con el fin de evaluar y tomar decisiones en cuanto a la mejor ubicación de los nodos, este monitoreo se realiza utilizando la herramienta SPUD [31]

3.1. Protocolo MODBUS.

MODBUS es un protocolo de comunicación, ubicado en el nivel 7 del modelo OSI [27]. Éste establece comunicación Cliente/Servidor entre dispositivos conectados en diferentes tipos de buses y redes. MODBUS fue diseñado por Modicon en 1979, convirtiéndose en el protocolo de comunicación de facto desde su creación.

MODBUS es un protocolo de petición/respuesta y ofrece servicios especificados por códigos de funciones.

Es actualmente implementado usando:

- TCP/IP sobre Ethernet
- Transmisión serial asíncrona sobre una variedad de medios (cable: EIA/TIA-232-E, EIA- 422, EIA/TIA-485-A; fibra, radio, etc.)
- MODBUS PLUS, que es una implementación extendida de Modicom.

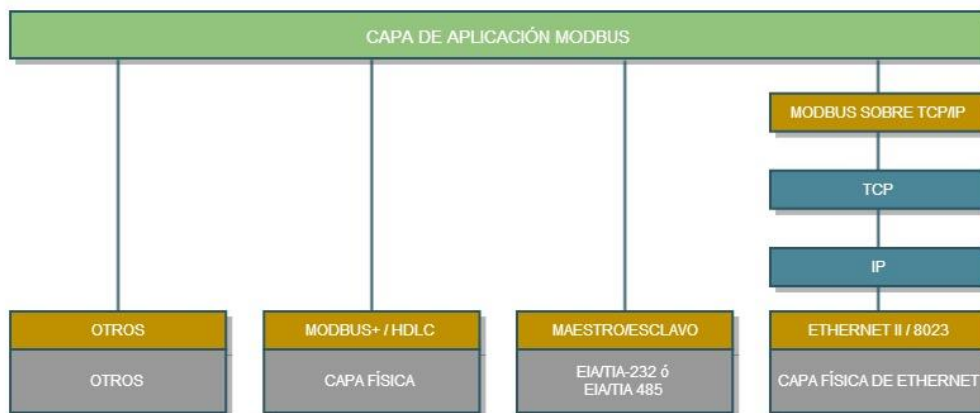


Figura 6: Pila de comunicación MODBUS

3.1.1 Contexto del protocolo MODBUS

El protocolo MODBUS nos permite una fácil comunicación con todos los tipos de redes y arquitecturas. Cada tipo de dispositivo (PLC, HMI, panel de control, controlador, I/O, entre otros), puede utilizar el protocolo MODBUS para iniciar una operación remota. La misma comunicación puede ser realizada de igual manera serie o en redes Ethernet TCP/IP. Los Gateways nos permiten la comunicación entre diferentes tipos de buses o redes utilizando el protocolo MODBUS.

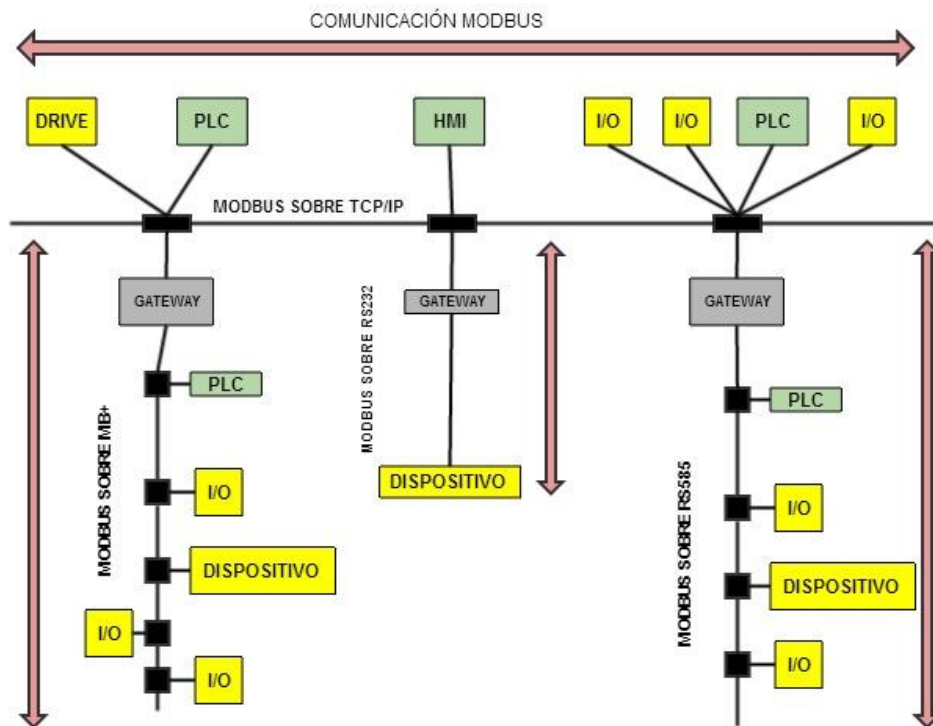


Figura 7: Ejemplo de arquitectura de red del protocolo MODBUS

3.1.2 Descripción del protocolo

El protocolo MODBUS define una Unidad Simple de Datos (PDU) subyacente en las capas de comunicación. El mapeo del protocolo MODBUS en buses o redes específicas pueden introducir algunos campos adicionales en la Unidad de Aplicación de Datos MODBUS (ADU).

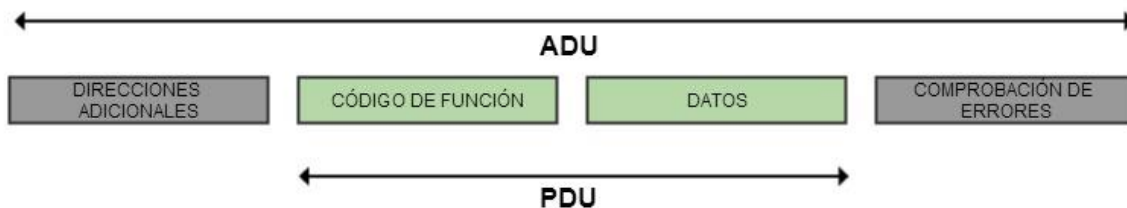


Figura 8: Trama MODBUS

La Unidad de Aplicación de Datos MODBUS es construida por el cliente que inicia una transacción MODBUS. Esta función indica al servidor que tipo de acción realizar. El protocolo de aplicación MODBUS establece el formato del requerimiento iniciado por el cliente.

El campo de código de función de una unidad de datos MODBUS se codifica en un byte. Los códigos válidos están en el intervalo de 1 a 255 decimal (el rango de 128 a 255 está reservado y utilizado para la respuesta de excepciones). Cuando se envía un mensaje desde un cliente a un dispositivo servidor el código de la función MODBUS le dice al servidor qué tipo de acción debe realizar. El código de función "0" no es válido.

Algunas subfunciones son añadidas a los códigos de función MODBUS para definir múltiples acciones.

El campo de datos enviada por el servidor a un cliente contiene información adicional que el servidor usa para tomar la acción definida por el código de función MODBUS. Esta puede incluir elementos como valores discretos y direcciones de registros.

El campo de datos puede no existir (o tiene longitud cero) en ciertos casos de requerimientos, en este caso el servidor no requiere información adicional. El código de función solamente especifica la acción.

Si no existe error relacionado con el requerimiento en la función MODBUS de una ADU recibida el campo de datos de la respuesta del servidor al cliente contiene los datos solicitados. Si ocurre algún error relacionado con la función MODBUS solicitada, el campo contiene un código de excepción que la aplicación del servidor puede utilizar para determinar la siguiente acción a tomar.

Cuando el servidor responde al cliente, se utiliza el campo de código de la función para indicar respuesta normal (libre de errores) o de algún tipo de error (llamado excepción). Para una respuesta normal, el servidor simplemente hace eco a la solicitud del código de función original.

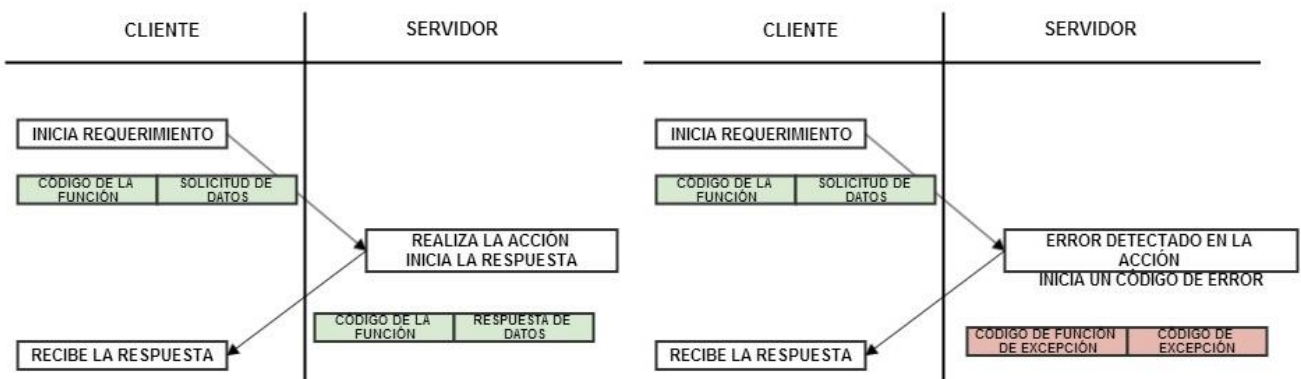


Figura 9: Transacción MODBUS sin errores y con errores

Para una respuesta de excepción, el servidor regresa un código que es equivalente a la función original del PDU solicitado con su bit más significativo con el valor lógico 1

3.1.3. MODBUS TCP/IP

A continuación se describirá el protocolo MODBUS sobre TCP/IP, esta variante del protocolo se incluyó para hacerlo compatible con redes ETHERNET [32].

3.1.3.1 Modelo Cliente/Servidor

El protocolo MODBUS provee una comunicación Cliente/Servidor entre dispositivos conectados a una red ETHERNET TCP/IP.

Este modelo Cliente/Servidor está basado en cuatro tipos de mensajes:

1. Solicitud MODBUS,
2. Confirmación MODBUS,
3. Indicación MODBUS,
4. Respuesta MODBUS



Figura 10: Modelo MODBUS TCP/IP

Una solicitud MODBUS es el mensaje enviado en la red por el Cliente para iniciar una transacción.

Una indicación MODBUS es el mensaje solicitado recibido en el lado del Servidor, una respuesta MODBUS es el mensaje de respuesta enviado por el Servidor, una confirmación MODBUS es el mensaje de respuesta recibido en el lado del cliente.

Los servicios de mensajería MODBUS (modelo cliente/servidor) se utilizan para intercambio de información en tiempo real:

- entre dos aplicaciones del dispositivo,
- entre la aplicación del dispositivo y otro dispositivo,
- entre HMI / SCADA aplicaciones y dispositivos,
- entre un PC y un programa de dispositivo que proporciona servicios en línea.

3.1.3.2 Descripción del protocolo MODBUS TCP/IP

Un sistema de comunicación a través de MODBUS TCP/IP puede incluir diferentes tipos de dispositivos:

- Un Cliente/Servidor MODBUS TCP/IP conectado a una red TCP/IP
- Los dispositivos de interconexión como puente, router o gateway de interconexión entre la red TCP/IP y una sub-red de dispositivos MODBUS serial que nos permita la conexión Cliente/Servidor.

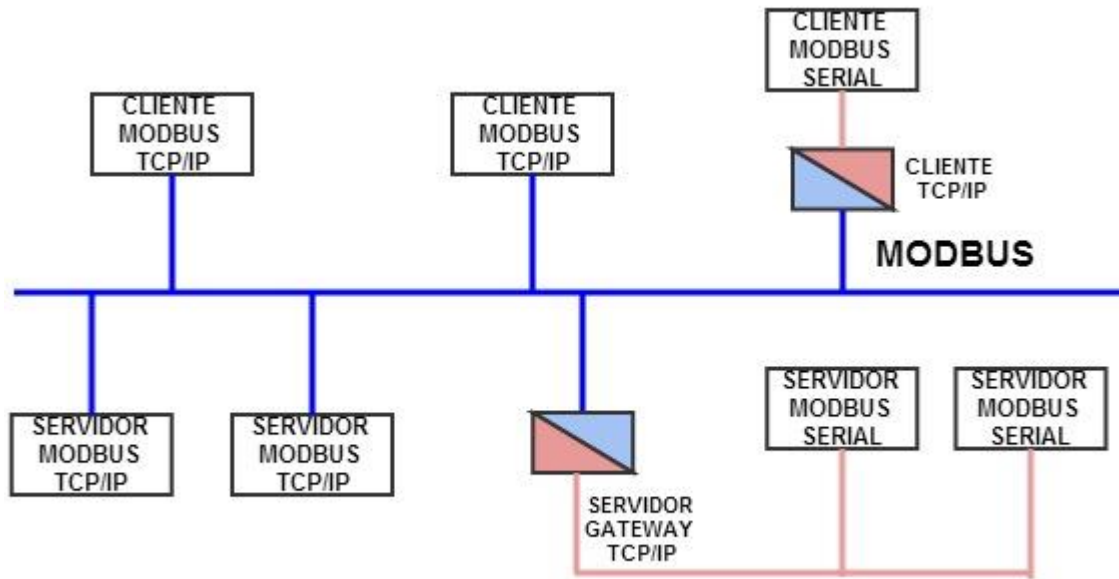


Figura 11: Arquitectura MODBUS TCP/IP

3.1.3.2.1 Unidad de Aplicación de Datos en MODBUS TCP/IP

En la Figura 12 se describe el encapsulado de una petición o respuesta MODBUS cuando se realiza por medio de MODBUS TCP/IP

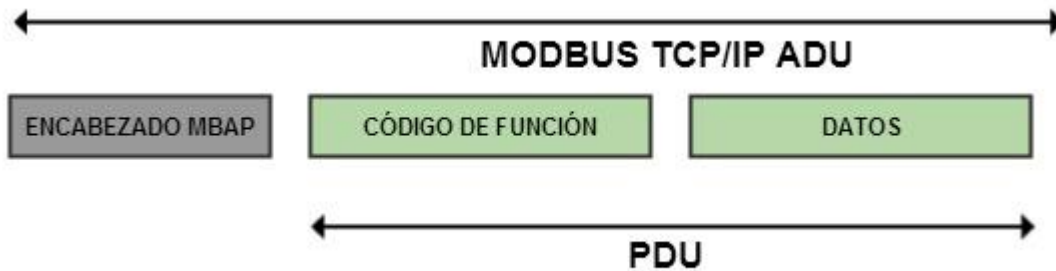


Figura 12: Requerimiento/Respuesta sobre MODBUS TCP/IP

Un encabezado dedicado es usado en el protocolo TCP/IP para identificar la Unidad de Aplicación de Datos MODBUS. Ese encabezado se conoce como MBAP (MODBUS Application Protocol header).

Este encabezado proporciona algunas diferencias en comparación con la Unidad de Aplicación de Datos MODBUS RTU usada en la comunicación serial:

- El campo MODBUS 'slave address' usado en MODBUS serial es remplazado por un unico byte 'Unit Identifier' con la cabecera MBAP. El 'Unit Identifier' se utiliza para comunicarse a través de dispositivos como puentes, enrutadores y puertas de enlace

que utilizan una sola dirección IP para soportar múltiples unidades MODBUS independientes.

- Todas las solicitudes y las respuestas MODBUS están diseñados de tal manera que el destinatario puede verificar que un mensaje ha finalizado. Para los códigos de función, donde el PDU MODBUS tiene una longitud fija, el código de función solo es suficiente. Para los códigos de función que llevan una cantidad variable de datos en la solicitud o respuesta, el campo de datos incluye un conteo de bytes.
- Cuando se lleva a MODBUS sobre TCP, información de la longitud adicional es llevada en el encabezado MBAP para que el receptor reconozca límites de mensaje, incluso si el mensaje se ha dividido en varios paquetes para su transmisión. La existencia de normas de longitud explícita e implícita, y el uso de un código de comprobación de errores CRC-32 (en Ethernet) resulta en una probabilidad infinitesimal de la no detección de errores de un mensaje de petición o respuesta.

3.2. Librería Libmodbus y proceso de implementación para monitoreo.

Libmodbus es una librería de código abierto para el envío y recepción de datos utilizando el protocolo de comunicación MODBUS [28]. Esta librería está escrita en C y soporta las variantes RTU (serial) y TCP (Ethernet). Licenciada bajo LGPL 2.1+, eso implica que se puede conocer el código fuente, estudiarlo y modificarlo para adaptarlo a nuestra necesidad como se ha hecho en este trabajo. La librería es una implementación de alto nivel del protocolo MODBUS tanto serial, TCP/IPv4 y TCP/PI IPv4 e IPv6. Tiene funciones para establecer comunicación MODBUS TCP/IP y manejo de datos que utilizamos en el programa de adquisición de datos que desarrollamos en lenguaje C para comunicarnos e interrogar los medidores Shark 100S, Shark 200 y Shark 200S.

3.2.1. Instalación de Libmodbus

Primero se instalan las herramientas necesarias para la compilación de la librería, en Ubuntu se instala el paquete build-essential desde la línea de comandos como se muestra a continuación

```
jbonilla@jbonilla-VirtualBox:~$ sudo apt-get install build-essential
```

La librería se descarga desde la página <http://libmodbus.org/download/>, en este trabajo se utiliza la versión 3.0.3.

Después se descomprime el archivo libmodbus-3.0.3.tar.gz y se ingresa al directorio donde se descomprimieron los archivos

```
jbonilla@jbonilla-VirtualBox:~$ tar xvzf libmodbus-3.0.3.tar.gz
jbonilla@jbonilla-VirtualBox:~$ cd libmodbus-3.0.3
```

Se instala la librería manualmente de la siguiente manera

```
jbonilla@jbonilla-VirtualBox:~/libmodbus-3.0.3$ ./configure
jbonilla@jbonilla-VirtualBox:~/libmodbus-3.0.3$ make
jbonilla@jbonilla-VirtualBox:~/libmodbus-3.0.3$ sudo make install
```

Con el script *configure* se comprueba que estén instalados todas las librerías para para compilar libmodbus, sino hay errores crea el archivo *make*, con *make* se compila la librería libmodbus y con *sudo make install* se instala.

3.2.1. Modificación de la librería Libmodbus

Una de las muchas ventajas del software libre es la disposición del código fuente para su estudio y modificación. Esa ventaja se ha utilizado con la librería libmodbus, se han hecho unas pequeñas modificaciones para incluir una función para la conversión de datos en formato int_32 (entero de 32bits) a flotante. La función se ha declarado en el archivo de cabecera “modbus.h” y la función se escribió en el módulo “modbus-data.c”. El nombre de esta nueva función es “modbus_get_int_from_int32(const uint16_t *src)”. Esta recibe como argumento un número de cuatro bytes y nos regresa su valor en un número enteros. A continuación se muestra la declaración de la función y la función:

Declaración de la función en el archivo “modbus.h”

```
//Función Definida para obtener los datos enteros de int32 a int
int modbus_get_int_from_int32(const uint16_t *src);
//fin de modificacion, modificado por Juan Jose Bonilla
```

Definición de la función en el módulo “modbus-data-c”

```
//Funcion para convertir de int32 a int, modificado por Juan Jose Bonilla

int modbus_get_int_from_int32(const uint16_t *src)
{
    int valor = 0;
    int32_t i;

    i = (((int32_t)src[1]) << 16) + src[0];
    memcpy(&valor, &i, sizeof(float));

    return valor;
}
```

Después de modificar el código fuente de la librería libmodbus se instala manualmente como se mostró anteriormente.

Con esta función podemos convertir tipos de datos sin signo 32 bits y sin signo 16 bits. Para los datos en formato float IEEE 745 la librería libmodbus nos provee la función “float modbus_get_float(const uint16_t *src)” que recibe como parámetro cuatro bytes en formato IEEE 745 y regresa un número flotante.

Con la conversión de estos tres tipos de datos es suficiente para obtener los valores de las variables eléctricas que proporcionan los medidores.

3.3. Programa de consulta de datos

El programa que se ha desarrollado para la consulta remota de los medidores está programado en lenguaje C. Este utiliza las librerías Libmodbus versión 3.0.3.

La aplicación se ha programado en sistema operativo GNU/Linux usando la distribución Ubuntu 12.04.

3.3.1. Estructura del programa en C

La aplicación está formada por cinco archivos:

- main.c
- consulta100.h
- consulta100.c
- consulta200.h
- consulta200.c

En la función main.c es la función principal del programa, que se encarga de llamar las funciones que consultan y guardan los datos en los medidores.

En consulta100.h y consulta100.c se define la función “consulta_medidor_100 (char *m, char *n)” que interroga y guarda los datos de los medidores Shark 100S, esta función acepta dos parámetros:

- Dirección IP del medidor
- Nombre de la tabla de la subestación en la base de datos SPUD de MySQL

Como se muestra a continuación:

```
consulta_medidor_100 (“10.30.202.2”, “Economía1”);
```

En donde “10.30.202.2” es la dirección IP del medidor de la subestación Economía 1 y “Economía1” es el nombre de la tabla correspondiente en la base de datos SPUD.

Para los medidores Shark 200 y 200S también se ha creado la función “consulta_medidor_200 (char *m, char *n)”, definida en consulta200.h y consulta200.c, esta función acepta dos parámetros:

- Dirección IP del medidor
- Nombre de la tabla de la subestación en la base de datos SPUD de MySQL

Como se muestra a continuación:

```
consulta_medidor_200 ("10.30.201.2", "Derecho");
```

En donde "10.30.201.2" es la dirección IP del medidor de la subestación de Derecho y "Derecho" es el nombre de la tabla correspondiente en la base de datos SPUD.

Las funciones `consulta_medidor_100` (`char *m`, `char *n`) y `consulta_medidor_200` (`char *m`, `char *n`), se encargan de interrogar los medidores, obtener los datos MODBUS de la memoria de los equipos, convertir los datos en formato entero y guardarlos en la base de datos correspondiente.

Para iniciar en las funciones creadas para interrogar los medidores agregamos los archivos de cabeceras correspondientes a las librerías `libmodbus` y `mysql`. Así como también la declaración de la función que puede ser `consulta100.h` ó `consulta200.h` dependiendo el tipo de medidor que estamos interrogando. A continuación se muestran los archivos de cabeceras incluidos para la función `consulta100.h` y también la función en sí:

```
#include <stdio.h>
#include <unistd.h>
#include <string.h>
#include <stdlib.h>
#include <time.h>
#include <sys/time.h>
#include <errno.h>
#include <modbus/modbus.h> //Libreria Libmodbus
#include <my_global.h> //Libreria mysql
#include <mysql.h> //Libreria mysql
#include <my_sys.h> //Libreria mysql
#include "consulta100.h" //declaración de la función consulta100.c

int consulta_medidor_100 (char *m, char *n) /*Definicion de la funcion*/
```

Después creamos la conexión a la base de datos `mysql` que guardara las mediciones correspondientes de la subestación, se definen los parámetros del servidor, nombre de la base de datos, usuario y contraseña en los siguientes punteros: `*server`, `*database`, `*user`, `*password`, estos punteros son del tipo de dato `char`, a continuación se muestra la configuración de nuestra base de datos:

```

/*Creando conexión con la base de datos MySQL */

MYSQL *conn;
MYSQL_RES *res;
MYSQL_ROW row;
char *server = "localhost";
char *user = "root";
char *password = "root"; /* set me first */
char *database = "spud";
conn = mysql_init(NULL);
/* Conectando a la base de datos */
if (!mysql_real_connect(conn, server,
    user, password, database, 0, NULL, 0)) {
    fprintf(stderr, "%s\n", mysql_error(conn));
    exit(1);
}
/* enviando consula sql */
if (mysql_query(conn, "show tables")) {
    fprintf(stderr, "%s\n", mysql_error(conn));
    exit(1);
}
res = mysql_use_result(conn);
/* output table name */
printf("MySQL Tables in spud database:\n");
while ((row = mysql_fetch_row(res)) != NULL)
    printf("%s \n", row[0]);

```

Se crea la conexión con el medidor por medio del protocolo MODBUS sobre TCP/IP. Para ello se crea un nuevo contexto MODBUS TCP. Se utiliza la función `modbus_new_tcp(m, 502)`. En esta se definen dos parámetros. El primero es la dirección IP del medidor, se utiliza la variable "m" que contiene la dirección IP del medidor, y el otro parámetro es el puerto TCP utilizado para la comunicación entre nuestra PC y el medidor, este por defecto es el puerto "502". Otro punto importante al iniciar la comunicación con el medidor es definirle un número de esclavo de dispositivo. Todos los medidores utilizan el "1" y se define utilizando la función `modbus_set_slave(ctx, numero de esclavo)`.

Si no se establece la comunicación con el servidor MODBUS el puntero `ctx` puede tomar dos valores: `NULL` si no está instalada la librería `Libmodbus` y `-1` si no ha podido establecer comunicación con el servidor por parámetros erróneos o estar indisponible. La consulta a los medidores se realiza de la siguiente manera:

```

/*Creamos la consulta al medidor con la direccion IP definida en el valor de m*/

ctx = modbus_new_tcp(m,502);
modbus_set_slave(ctx,1);
if (ctx == NULL) {
    fprintf(stderr, "Unable to allocate libmodbus context\n");
    return -1;
}
if (modbus_connect(ctx) == -1) {
    fprintf(stderr, "Connection failed: %s\n", modbus_strerror(errno));
    modbus_free(ctx);
    return -1;
}

```

Una vez creado el contexto TCP/IP, es posible consultar los registros de los medidores almacenados en las ubicaciones de memoria del medidor. El bloque de memoria tiene la información del medidor como nombre del equipo, número de serie, modelo y también las mediciones. El medidor clasifica las mediciones por bloques. El mapa MODBUS es bastante extenso e incluye funciones de escritura y control de registros que no se utilizaron. Por eso solo consultamos los siguientes bloques: bloque primario de lectura, bloque primario de energía y bloque THD.

Para realizar esto utilizamos la función `modbus_read_registers(modbus_t *ctx, int addr, int nb, uint16_t *dest)`, que admite los parámetros: `*ctx` que es la definición de los parámetros de conexión MODBUS TCP/IP, `addr` que define en que registro se inicia la consulta, `nb` indica cuantos registros se consultan y `*dest` es un arreglo a donde se guardan cada registro como una palabra de 16 bits.

Se implementó de la siguiente manera en el programa de consulta, los datos obtenidos los guardamos en los arreglos `tab_reg` para los registros del bloque primario, `tab_ereg` para los registros de energía y `tab_thd` los registros de thd de voltaje y corriente.

A continuación se muestra como se implementó en código C:

```
/*Consultamos los registros del medidor*/

/*Primary Reading Block, Bloque Primario de Lecturas*/
rc = modbus_read_registers(ctx, 999, 30, tab_reg);
if (rc == -1) {
    fprintf(stderr, "%s\n", modbus_strerror(errno));
    return -1;
}

/*Primary Energy Block, Bloque Primario de Energia*/
rc = modbus_read_registers(ctx, 1099, 18, tab_ereg);
if (rc == -1) {
    fprintf(stderr, "%s\n", modbus_strerror(errno));
    return -1;
}

/*THD Block, Bloque THD*/
rc = modbus_read_registers(ctx, 3999, 42, tab_thd);
if (rc == -1) {
    fprintf(stderr, "%s\n", modbus_strerror(errno));
    return -1;
}
```

Con los registros del medidor procedemos a convertirlos de datos de 32 bits con signo para los registros del bloque primario con la función `modbus_get_flota(toFloat)`, 32 bits sin signo para los datos del bloque primario de energía y de 16 bits sin signo para los datos del bloque THD con la función `modbus_get_int_from_int32(toInt)`.

Finalmente las mediciones de guardan en variables del tipo flotante para el bloque de medición primario y bloque THD, y en variables de tipo entero para el bloque de medición de energía.

3.4. Base de datos

3.4.1. MySQL

La base de datos está diseñada en MySQL 5.1, se utiliza la base de datos 'spud' de la aplicación WEB de monitoreo de enlaces inalámbricos SPUD, se le añaden las tablas de cada una de las subestaciones.

Para instalar las librerías de librerías Mysql, desde la línea de comando ejecutamos los siguientes comandos, se instalan los paquetes mysql-client y libmysqlclient-dev, como se muestra a continuación

```
jbonilla@jbonilla-VirtualBox:~$ sudo apt-get install libmysqlclient-dev mysql-client
```

Las tablas de datos de los medidores Shark 100S, Shark 200 y 200S son diferentes como se observa en el anexo A.5.

3.4.2. phpMyAdmin

Para gestionar la base de datos, se instaló una aplicación WEB llamada phpMyAdmin, con esta herramienta es posible administrar nuestras bases de datos MySQL desde el navegador, para lo cual debemos ingresar a la dirección <http://localhost/phpmyadmin> si estamos en el servidor MySQL, para gestionar la base de datos desde otra computadora sustituimos localhost por la dirección IP del servidor MySQL.

3.4.3. Instalación phpMyAdmin

Instalamos el paquete phpmyadmin de la siguiente manera:

```
jbonilla@jbonilla-VirtualBox:~$ sudo apt-get install phpmyadmin
```

Después lo configuramos con los datos de nuestro servidor MySQL, finalmente hay que hacer un enlace simbólico desde /usr/share/phpmyadmin hacia /var/www/phpmyadmin con el siguiente comando:

```
jbonilla@jbonilla-VirtualBox:~$ sudo ln -s /usr/share/phpmyadmin /var/www/
```

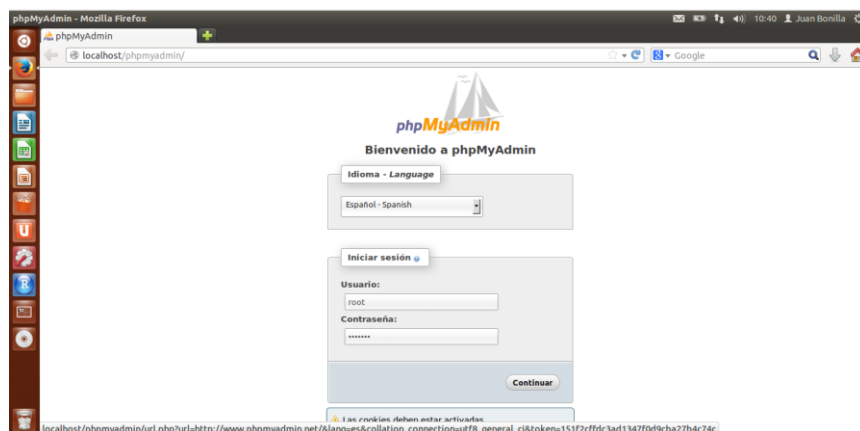


Figura 13: phpMyAdmin

3.5. SPUD

Como parte de la monitorización remota de los medidores de energía eléctrica es necesario tener control del medio de comunicación y del estado de sus enlaces, como ya especifico estamos utilizando una red MESH WiFi que trabaja bajo el estándar IEEE 802.11b/g. El protocolo BATMAND nos permite desde la línea de comando observar los enlaces de la red MESH y el estado de sus conexiones, sin embargo no es fácil visualizar estas variables desde una consola, por esta razón se utiliza una aplicación WEB desarrollado por Village Telco llamado SPUD [31].

La aplicación es desarrollada en PHP utilizando el Framework CakePHP y el API 1.3 de Google Maps.

SPUD nos permite: Manejo de clientes, en nuestro caso los nodos de medición, nodo repetidor y supernodo. Monitorización de enlaces entre los nodos de la red Mesh, de manera más fácil por medio de la interface WEB y los mapas de Google. Personalización de los nodos de enlace, entre otros.

En la Figura 14, se observa SPUD con el monitoreo de los enlaces de los nodos de la red MESH utilizada. La instalación y configuración de SPUD esta descrita en el anexo A.2.

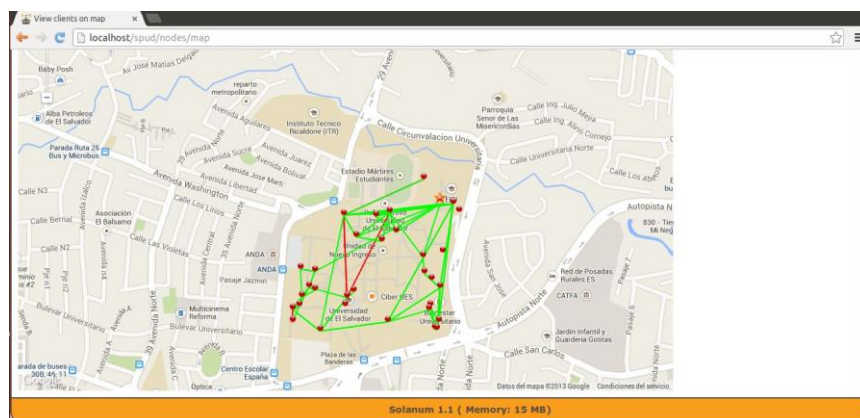


Figura 14: SPUD, Campus Universidad de El Salvador

Capítulo 4: FUNCIONAMIENTO Y OPERACIÓN DE LA RED

La Tabla 2 muestra las subestaciones integradas con su dirección IP y el tipo de medidor.

ID	Subestación	Dirección IP	Tipo de medidor	Tipo de subestación
1	Derecho	10.30.201.2	Shark 200S	3 fases
2	Economía 1	10.30.202.2	Shark 100S	1 fase
3	Economía 2	10.30.203.2	Shark 100S	1 fase
4	Economía 3	10.30.204.2	Shark 100S	1 fase
5	Economía 4	10.30.205.2	Shark 100S	1 fase
6	Economía 5	10.30.206.2	Shark 200S	3 fases
7	Economía 6	10.30.207.2	Shark 100S	1 fase
8	Humanidades 1	10.30.208.2	Shark 100S	3 fases
9	Humanidades 2	10.30.209.2	Shark 100S	1 fase
10	Humanidades 3	10.30.210.2	Shark 100S	1 fase
11	Comedor UES	10.30.211.2	Shark 100S	1 fase
12	Cafetines	10.30.212.2	Shark 100S	1 fase
13	Periodismo	10.30.213.2	Shark 100S	3 fases
14	Primario FIA	10.30.214.2	Shark 200	3 fases
15	Auditorium Mármol	10.30.215.2	Shark 100S	1 fase
16	Mecánica - Complejo	10.30.216.2	Shark 100S	1 fase
17	Agronomía	10.30.217.2	Shark 100S	3 fases
18	Agronomía Decanato	10.30.218.2	Shark 100S	1 fase
19	Agronomía Galera	10.30.219.2	Shark 100S	1 fase
20	Agronomía Química	10.30.220.2	Shark 100S	3 fases
21	Química	10.30.221.2	Shark 200S	3 fases
22	Química Imprenta	10.30.222.2	Shark 100S	1 fase
23	Odontología Imprenta	10.30.223.2	Shark 100S	1 fase
24	Odontología 1	10.30.224.2	Shark 200S	3 fases
25	Odontología 2	10.30.225.2	Shark 200S	3 fases
26	Odontología 3	10.30.226.2	Shark 200S	3 fases
27	Medicina	10.30.227.2	Shark 200S	3 fases
28	Psicología	10.20.228.2	Shark 100S	3 fases
29	Rectoría	10.30.229.2	Shark 200S	3 fases
30	Artes	10.30.230.2	Shark 200S	3 fases

Tabla 2: Medidores instalados campus central UES

En la Figura 15, se muestra la ubicación de los medidores en el campus central de la UES. La visualización se realiza mediante la herramienta SPUD. SPUD permite el monitoreo de los enlaces inalámbricos de la red entre los nodos de medición, repetidor y supernodo. Los colores de los enlaces definen su estado, enlace color verde significa enlace en buen estado, color rojo enlace débil y color negro enlace muy débil.

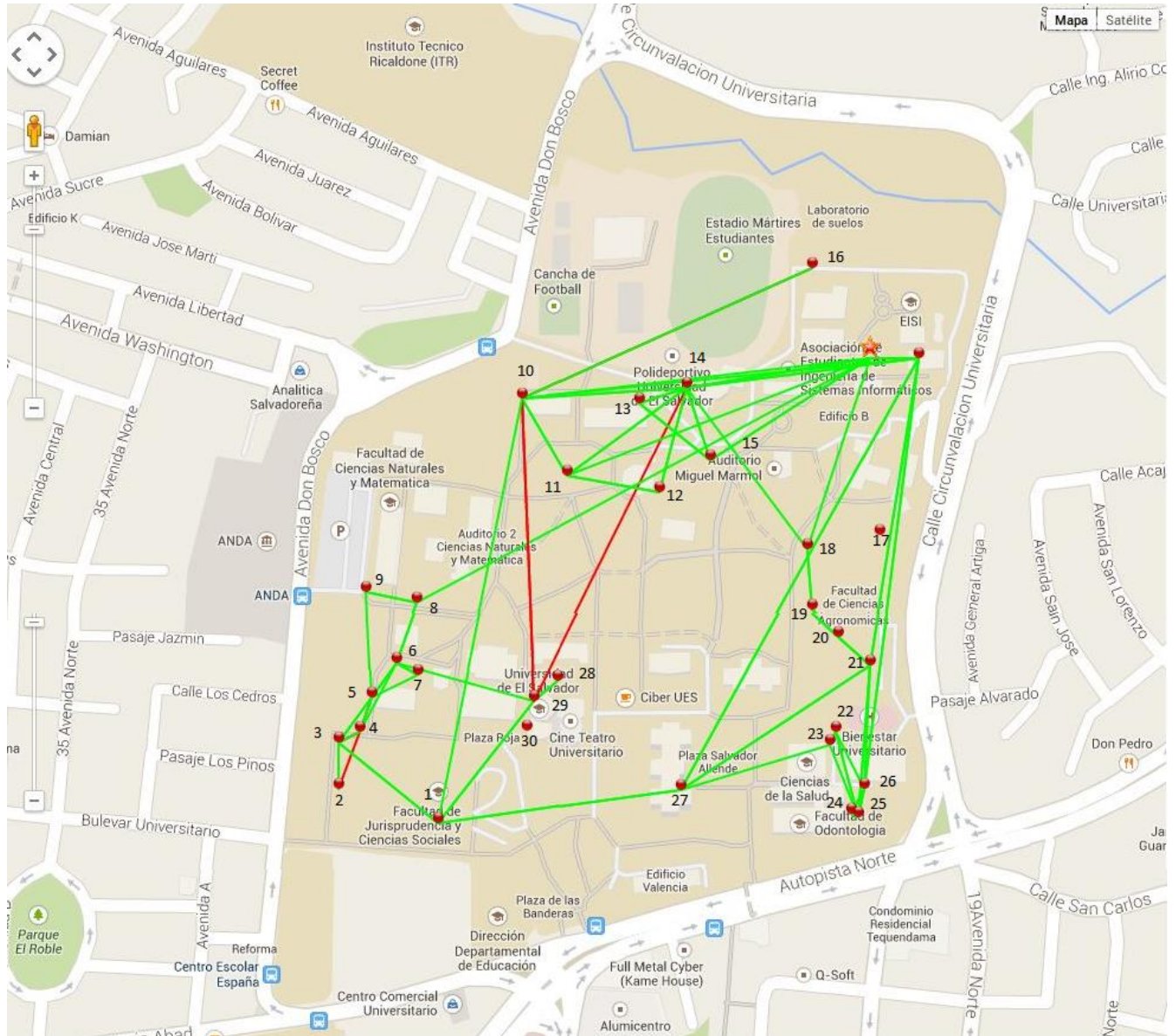


Figura 15: Ubicación Medidores en campus central UES

Como se observa en la figura No.15, se logra la comunicación entre todos los nodos de la red, también se observa que los nodos 1, 14, 27 y el repetidor en la Escuela de Arquitectura fortalecen la red inalámbrica al tener conexión con muchos nodos. En cuanto a los nodos 28, 29 y 30 es necesario colocar un nodo repetidor en el centro de la UES para crear otras rutas de salida hacia la red.



Figura 16: Medidor Auditorium Química (subestación No. 21)



Figura 17: Medidor Odontología 1 (subestación No. 24)

4.1. Sistema de monitoreo UES (SMUES)

Para la consulta de las mediciones guardadas en la base de datos MySQL se desarrolló una aplicación WEB con HTML, javascript y PHP llamada SMUES, la pantalla principal de la aplicación se muestra en la Figura 18.

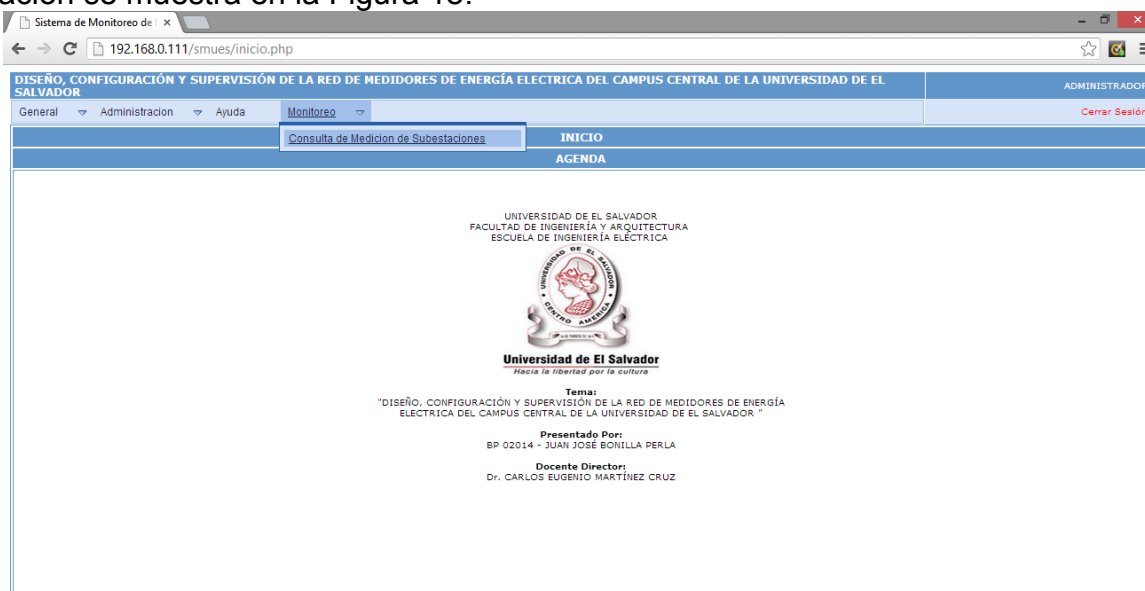


Figura 18: Pantalla inicia SMUES

Esta aplicación permite al usuario luego de introducir el usuario y la clave correcta, escoger la subestación, periodo de tiempo e intervalo de tiempo para tener las mediciones en la pantalla o como un archivo xls, la pantalla de consulta se presenta en la figura No. 19.

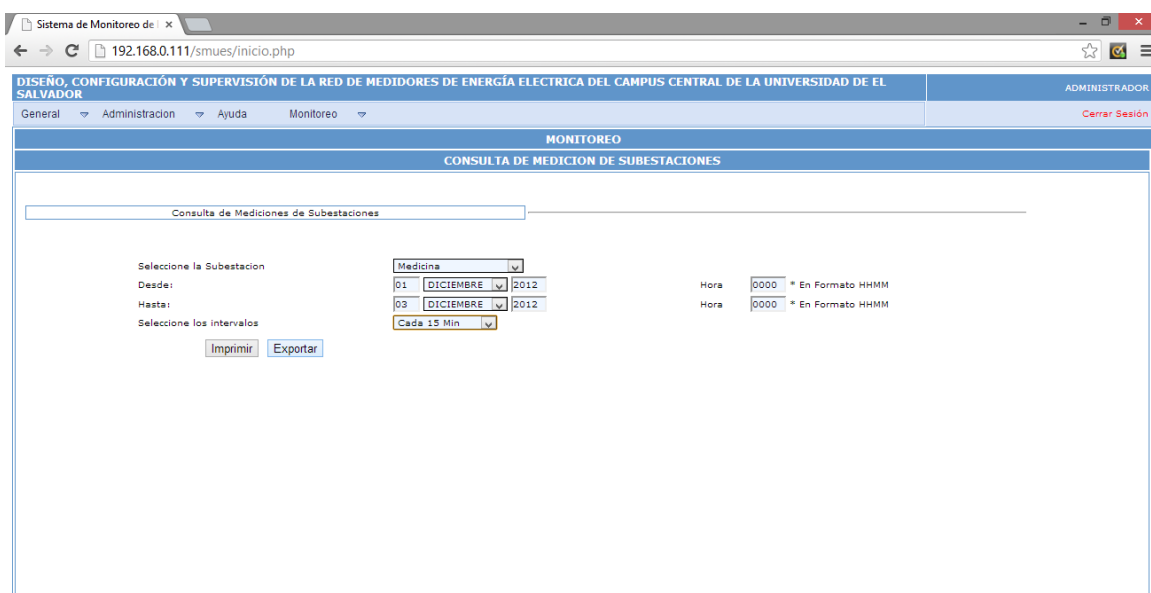


Figura 19: Pantalla consulta de datos SMUES

4.2. Mediciones obtenidas en Subestación Medicina

En la siguientes gráficas se muestran las mediciones obtenidas del medidor de la subestación de la Facultad de Medicina de la UES para el día 10 de diciembre del año 2012, las mediciones se extrajeron de la base de datos cada 15 minutos:

4.2.1. Voltajes de fases

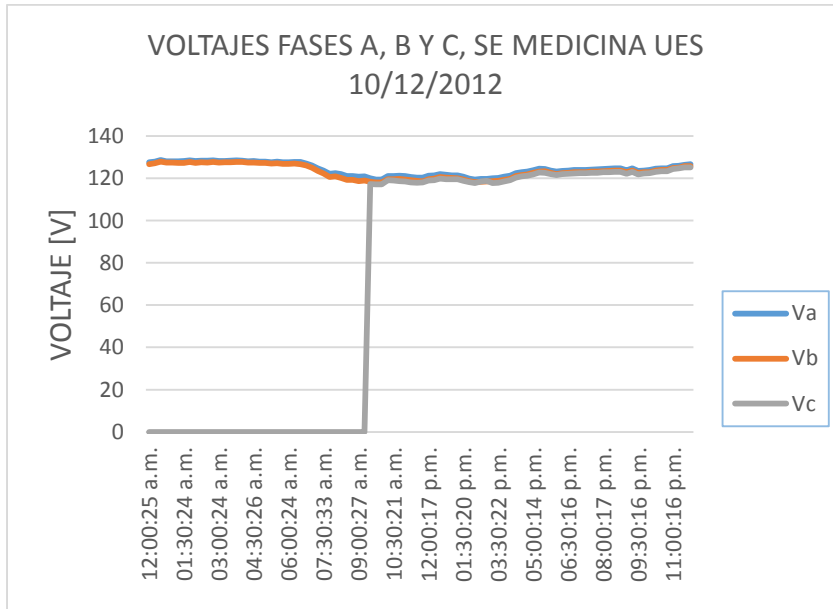


Figura 20: Voltajes Fases A, B y C, SE Medicina

4.2.2. Corrientes de fases

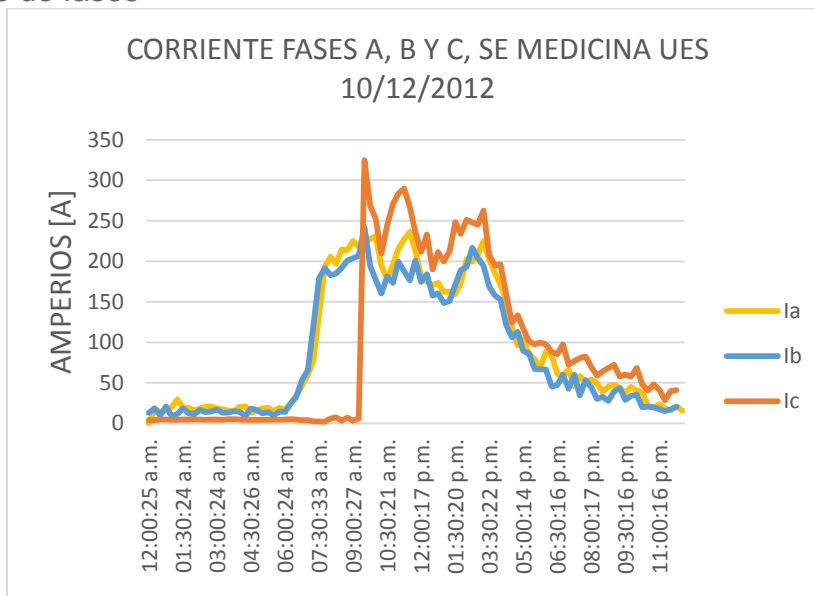


Figura 21: Corrientes Fases A, B y C, SE Medicina

4.2.3. Potencia Activa, Reactiva y Aparente

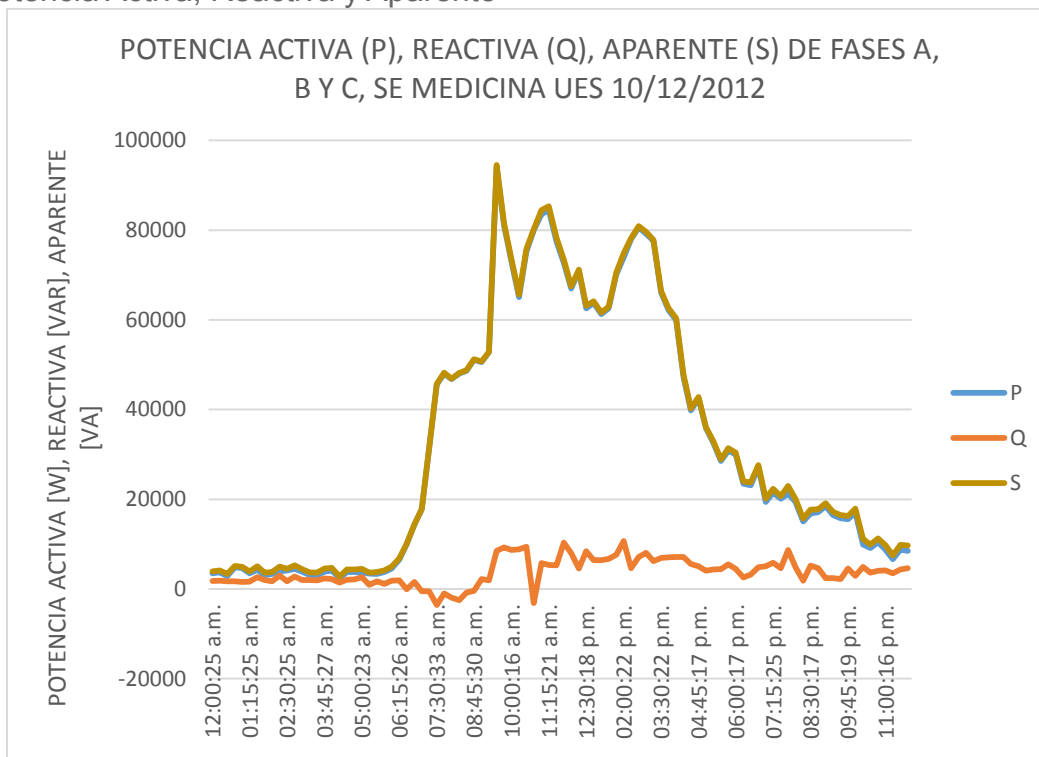


Figura 22: Potencia Activa, Reactiva y Aparente

4.2.4. Factor de Potencia Trifásico

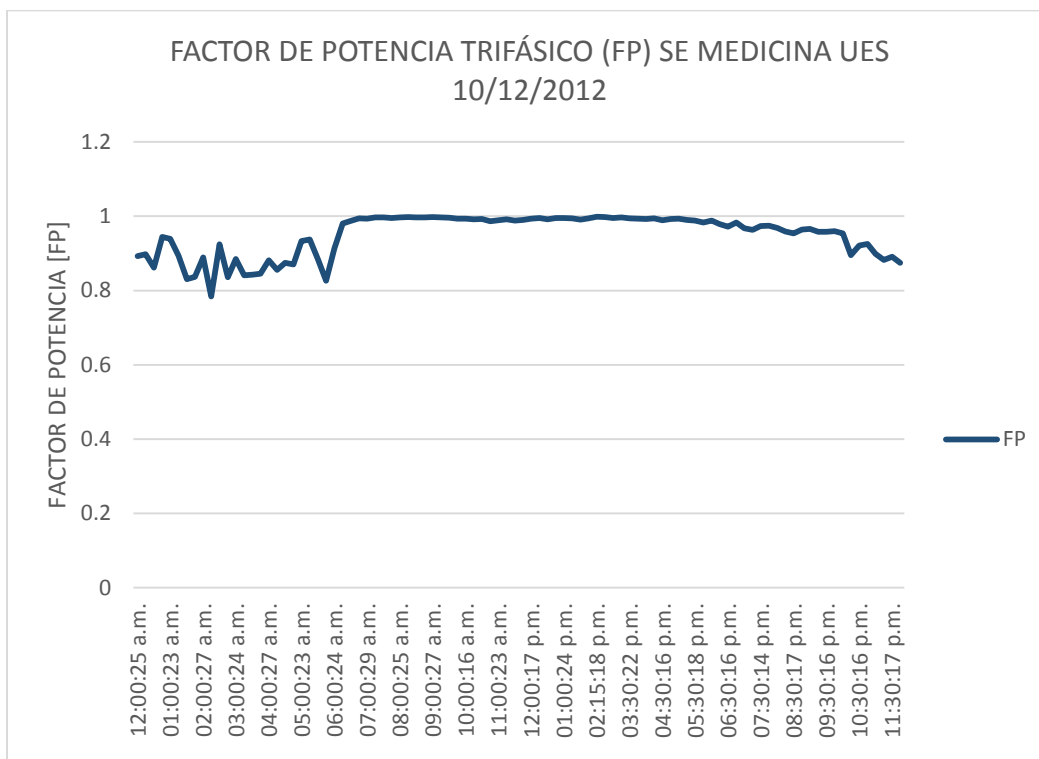


Figura 23: Factor de Potencia trifásico, SE Medicina

4.2.5. Factor de Potencia por fases

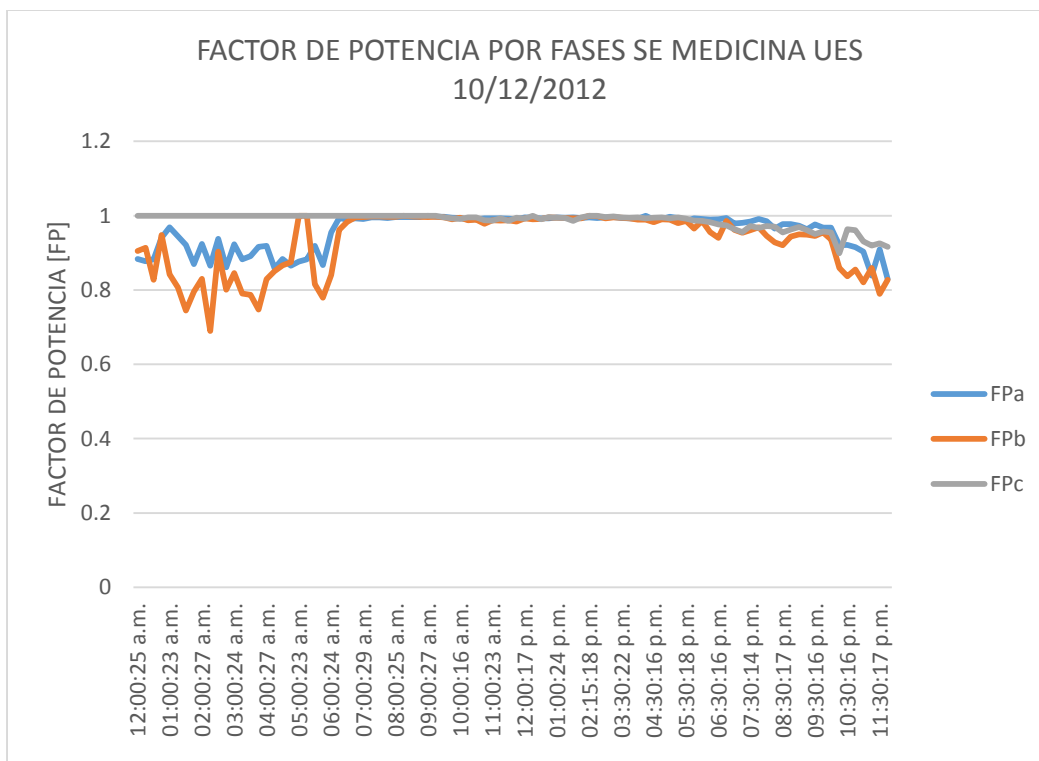


Figura 24: Factor de Potencia por fases, SE Medicina

4.2.6. Frecuencia

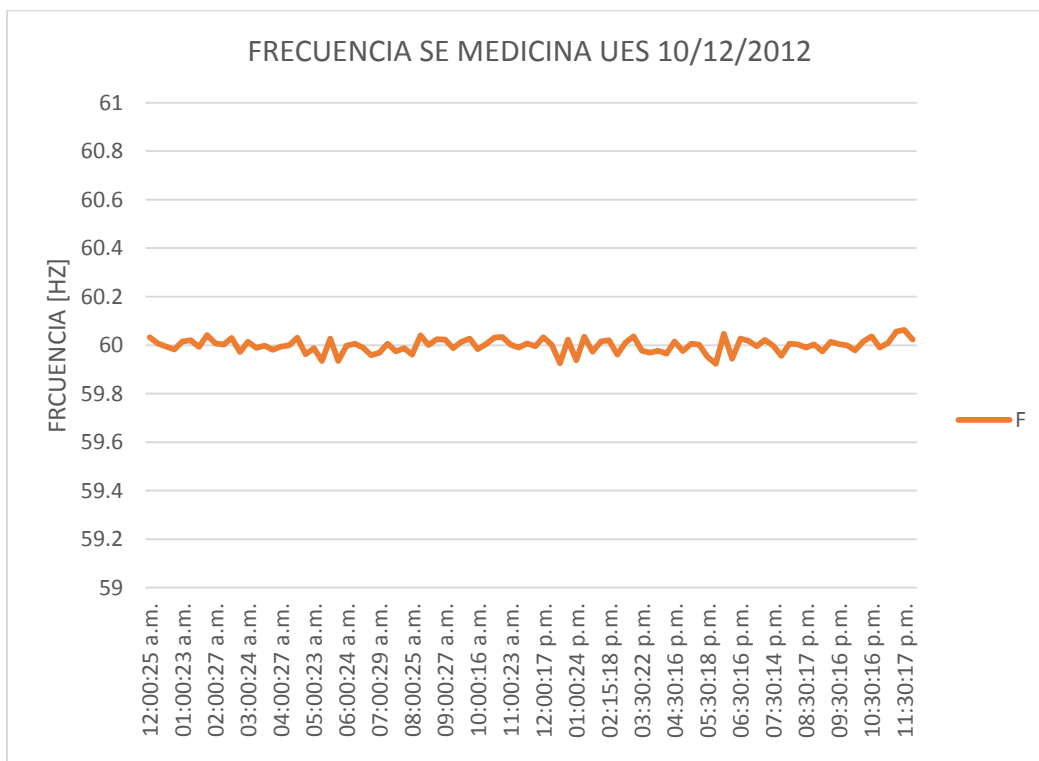


Figura 25: Frecuencia SE Medicina

4.2.7. Consumo de energía por hora

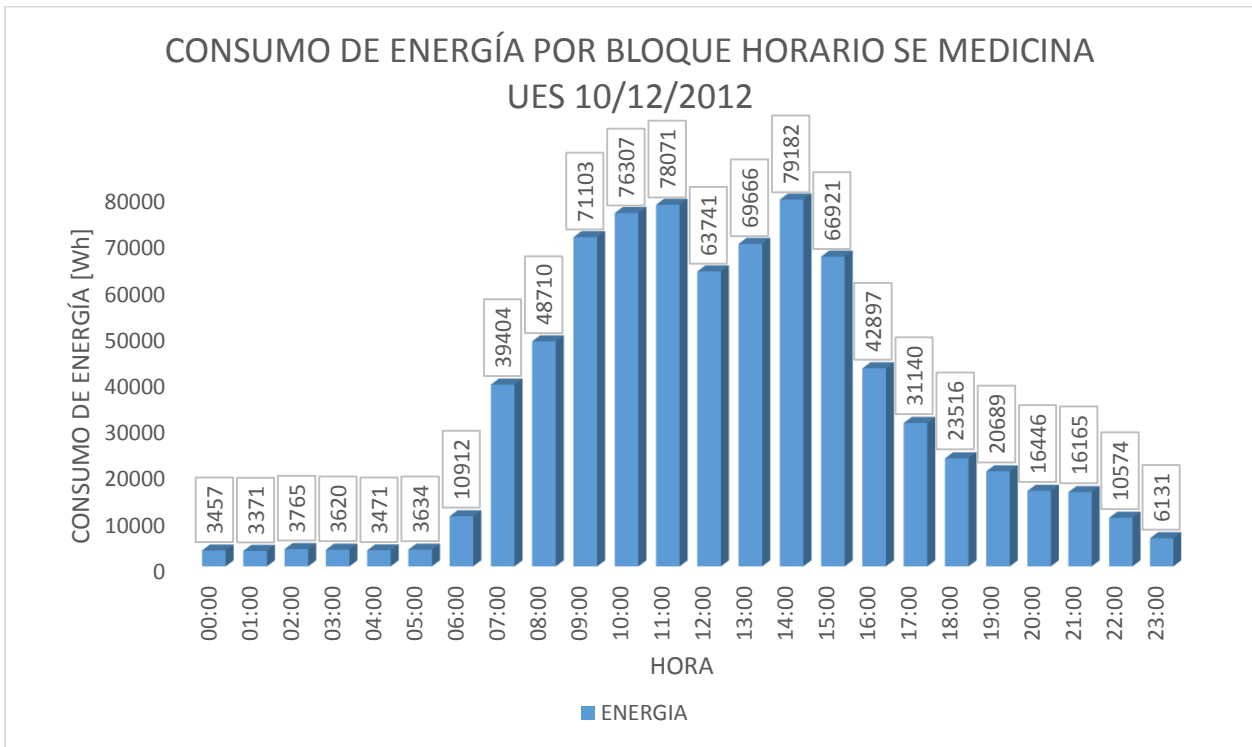


Figura 26: Consumo de Energía por bloque horario, SE Medicina

Capítulo 5: CONCLUSIONES Y LINEAS FUTURAS.

5.1. Conclusiones

- La red de medidores de energía eléctrica del campus central de la Universidad de El Salvador es un proyecto que parte de la necesidad de conocer el consumo de energía en la UES. La red permite realizar un mejor análisis y conocer las características de consumo de cada subestación. Con esto se podrán tomar decisiones consecuentes a mejorar la eficiencia en el consumo de energía en la Universidad.
- La red inalámbrica tipo Mesh implantada en la Universidad de El Salvador, no es lo suficientemente robusta debido las limitantes de los enlaces WiFi.
- Las aplicaciones desarrolladas permitieron la interrogación de los medidores y la configuración de los equipos en treinta subestaciones de manera remota. Con ello se evitan riesgos para el personal técnico y estudiantes al no tener que estar cerca de áreas energizadas para obtener estos datos o configurar los medidores.
- Con las mediciones guardadas en la base de datos, se supera la limitación de los medidores que no tienen memoria para guardar las mediciones.
- La aplicación desarrollada para la consulta y almacenamiento de las mediciones, así como la aplicación de consulta de las mediciones en la base de datos (SMUES), se realizó con tecnologías open source. Estas incluyen GNU/Linux Ubuntu 12.04, gcc, libmodbus, php y MySQL. Esto permitió ahorrar una significativa cantidad de dinero en licencias. También se obtiene total control del software para estudiarlo y modificarlo como se realizó en el presente trabajo.
- El protocolo MODBUS es ampliamente usado en la industria de automatización, siendo el protocolo de comunicación de facto de la industria. MODBUS es un protocolo abierto y esta implementado en muchos medidores, como es el caso de los medidores adquiridos por la UES.

5.2. Líneas futuras.

Se pueden hacer mejoras, localizando nuevos puntos de repetición en puntos claves del campus central de la Universidad de El Salvador para fortalecer la red. También se puede mejorar el software de monitoreo del estado de la red investigando o desarrollando alternativas a SPUD.

En cuanto al software de interrogación y almacenamiento de datos, se puede investigar en la presentación de los datos en tiempo real a través de aplicaciones WEB o nativas. La base de datos se puede mejorar depurando los datos a guardar y aplicando buenas practicas SQL.

En un futuro se pueden incorporar más medidores desarrollados en la Escuela de Ingeniera Eléctrica que utilicen el protocolo MODBUS.

ANEXOS

A.1. Instalación de OpenWrt, B.A.T.M.A.N. y servidor de visualización VIS en NS2

Para unir el router NS2 a la red de los MP's, es necesario cambiar el firmware de fábrica por el firmware OpenWrt [21], y además instalar el software B.A.T.M.A.N. y el servidor de visualización VIS, que nos sirven para enrutar la red y en el caso del supernodor también como servidor de visualización para lo demás nodos de la red.

Los siguientes tres archivos son necesarios para hacer el cambio de firmware:

- open-mesh-flash
- openwrt-atheros-ubnt2-squashfs.bin
- openwrt-atheros-Linux-lzma

Estos se descargan de las siguientes direcciones:

<http://download.villagetelco.org/ns2/open-mesh-flash>

<http://download.villagetelco.org/ns2/openwrt-atheros-vmlinux.lzma>

<http://download.villagetelco.org/ns2/openwrt-atheros-ubnt2-squashfs.bin>

El procedimiento para preparar el NS2 es el siguiente:

- Mantener el botón de reset presionado, mientras se energiza el NS2 y esperar hasta que las luces led hagan barrido, esto indicara que el NS2 está listo para recibir los archivos con el nuevo firmware.
- Luego en la PC corriendo desde la aplicación Terminal de Ubuntu y ubicado en la carpeta que contiene los archivos, digitar en la línea de comandos la siguiente instrucción:

```
jbonilla@jbonilla-VirtualBox:~$ sudo ./open-mesh-flash eth0 openwrt-atheros-ubnt2-squashfs.bin  
openwrt-atheros-vmlinux.lzma
```

Luego esperar a que el NS2 deje de hacer el barrido de luces y este estará listo.

Hay que tener en cuenta que la nueva dirección IP del NS2 es ahora 192.168.10.20, por lo que en la PC es necesario configurar la tarjeta LAN con la siguiente instrucción:

```
jbonilla@jbonilla-VirtualBox:~$ sudo ifconfig eth0 192.168.10.10/24
```

Luego hacer telnet al NS2 y habilitar contraseña del NS2, con esto telnet será deshabilitado y ssh será habilitado así:

```
jbonilla@jbonilla-VirtualBox:~$ telnet 192.168.10.20  
Openwrt@OpenWrt: passwd
```

Para acceder al NS2 con la IP que tenga predeterminada (para el caso ahora es 192.168.1.100) y nueva contraseña; se hace vía ssh así:

A.2.2. Descarga del código fuente de SPUD

La aplicación la descargamos en el directorio `/var/www/`, para eso en la consola ingresamos

```
jbonilla@jbonilla-VirtualBox:~$ cd /var/www/
```

Después con el programa `subversion` descargamos la última versión de SPUD de la siguiente manera:

```
jbonilla@jbonilla-VirtualBox:~$ svn co  
http://dev.villagetelco.org/svn/villagetelco/spud/trunk spud
```

A.2.3. Base de datos

- a) Creamos una nueva base de datos para la aplicación, usamos el siguiente comando en la consola `mysql`:

```
jbonilla@jbonilla-VirtualBox:~$ mysqladmin -u root -p create spud
```

Si le hemos configurado la base de datos con clave, se pedirá la clave al momento de crearla

- b) Editamos la configuración de la base de datos SPUD, para esto modificamos el archivo de configuración `database.php` con nuestro editor favorito en nuestro caso utilizamos `VI`

```
jbonilla@jbonilla-VirtualBox:~$ vi /var/www/spud/app/config/database.php
```

Luego en el archivo `database.php` configuramos los parámetros del tipo de base de datos, el host, usuario, nombre de la base de datos y password

```
var $default = array (  
    'driver' => 'mysql',  
    'persistent' => false,  
    'host' => 'localhost',  
    'login' => 'root',  
    'password' => '',  
    'database' => 'spud',  
    'prefix' => '',  
);
```

Donde 'driver' es el tipo de base de datos para la aplicación el driver es 'mysql', 'host' es el nombre o la dirección IP del servidor donde se encuentra instalada la base de datos nosotros al tenerla instalada en nuestra PC la llamamos 'localhost', 'login' es el nombre del usuario de la base de datos usaremos 'root' para nuestro caso, 'password' es la contraseña del usuario de la base de datos, si esta no está definida dejamos en blanco la opción, 'database' es el nombre de la base de datos en para la aplicación se llama 'spud'.

c) Creamos el esquema de la base de datos de la siguiente manera desde la consola:

```
jbonilla@jbonilla-VirtualBox:~$ cd /var/www/spud/app/install/
jbonilla@jbonilla-VirtualBox:/var/www/spud/app/install$ mysql -u root spud -p
<spud_db_schema.sql
Enter password:
jbonilla@jbonilla-VirtualBox:/var/www/spud/app/install$
```

A.2.4. Cambiar permisos de usuarios de la carpeta app de la aplicación SPUD

Desde la consola se aplican los siguientes comandos:

```
jbonilla@jbonilla-VirtualBox:~$ cd /var/www/spud/app/
jbonilla@jbonilla-VirtualBox:/var/www/spud/app$ chown -Rf www-data tmp/
```

A.2.5. Habilitar módulo rewrite de Apache

```
jbonilla@jbonilla-VirtualBox:~$ cd /etc/apache2/
jbonilla@jbonilla-VirtualBox:/etc/apache2$ a2enmod rewrite
```

A.2.6. Editar el archivo de Apache2 000-default

Podemos usar vi como se muestra a continuación o cualquier otro editor como gedit

```
jbonilla@jbonilla-VirtualBox:~$ vi /etc/apache2/sites-enabled/000-default
```

```
Options Indexes FollowSymLinks MultiViews
AllowOverride All
```

Reiniciamos Apache

```
jbonilla@jbonilla-VirtualBox:~$ sudo /etc/init.d/apache2 restart
[sudo] password for jbonilla:
 * Restarting web server apache2                                apache2:
Could not reliably determine the server's fully qualified domain name, using 127.0.1.1 for
ServerName
... waiting apache2: Could not reliably determine the server's fully qualified domain name,
using 127.0.1.1 for ServerName
[ OK ]
jbonilla@jbonilla-VirtualBox:~$
```

A.2.7. Configurar servidor de visualización (VIS)

Configuramos la dirección IP de nuestro servidor de visualización

Donde 'host' es la dirección de nuestro servidor de visualización y 'port' es el puerto TCP/IP por el cual se establece la comunicación, los valores por default en el archivo config.php apuntan al VIS de Bo Kaap en Sudáfrica, nosotros lo sustituimos por nuestros datos

```
jbonilla@jbonilla-VirtualBox:~$ vi /var/www/spud/app/config/config.php
$config ['VIS']= array (
    'host'           => '10.130.1.1'
    'port'           => '2005'
    'timeout'        => '30'
    'vis_version'    => 'legacy'
    'mode'           => 'batman'
);
```

A.2.8. Agregar tareas programadas del sistema la actualización del servidor de visualización

a) Abrimos el editor del cronjob con el siguiente comando

```
jbonilla@jbonilla-VirtualBox:~$ crontab -e
```

b) Agregamos la siguiente línea

```
*/* * * * * root /usr/bin/wget -O - -q -t 1 http://localhost/spud/nodes/update >/dev/null
2>&1
```

A.2.9. Iniciamos SPUD

En el navegador Web iniciamos SPUD accediendo a la siguiente dirección:

<http://localhost/spud>

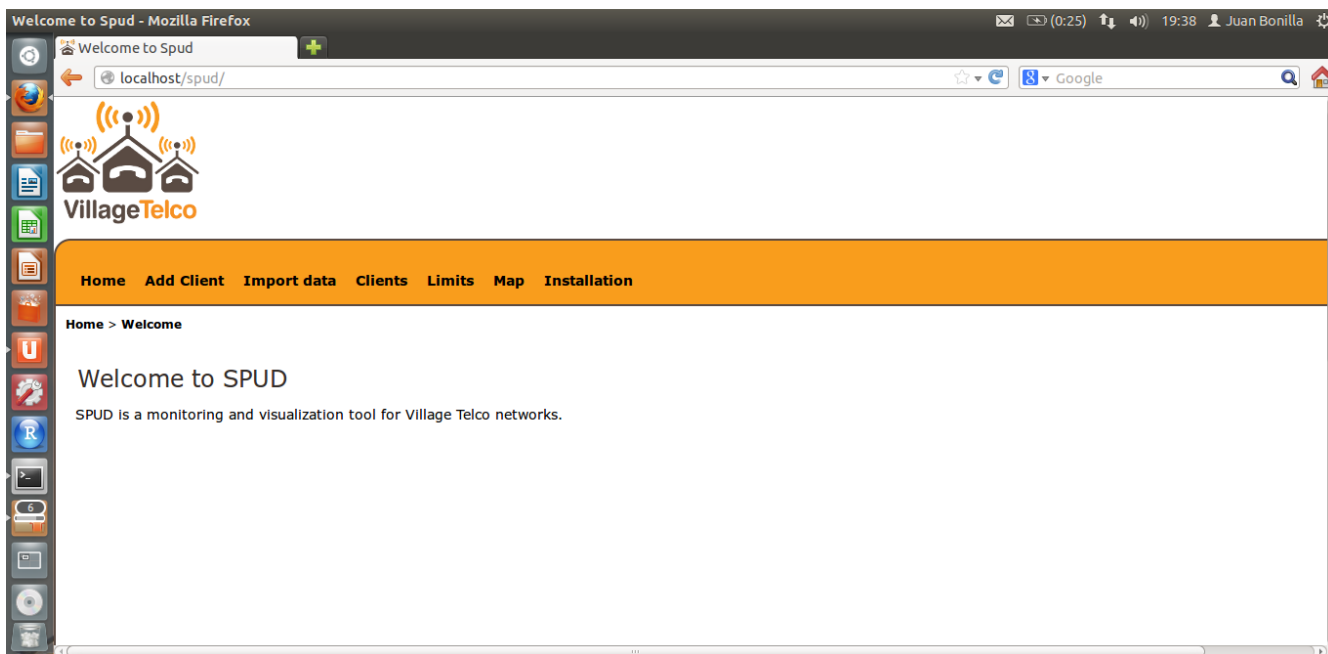


Figura 28: Pantalla de inicio SPUD

A.3. Mapa MODBUS Electroindustries Shark 100S

El mapa de MODBUS para el medidor Shark 100-S proporciona detalles e información acerca de las posibles lecturas del medidor y su programación. El medidor Shark® 100-S puede ser programado con los botones en la caratula del medidor, o mediante el uso de software.

A.3.1. Secciones de Mapa de Registro MODBUS

El Mapa de registro MODBUS del medidor Shark® 100-S [23] incluye las siguientes secciones:

Sección de Datos Fijos, registros del 1 al 47, los detalles de información fija del medidor.

Sección datos del Medidor, Registros del 1000 al 5003, detalles de las lecturas del medidor, incluyendo lecturas Primaria, Bloque de Energía, Demanda de Bloque, Bloque Angulo de Fase, Bloque de Estado, Bloque THD, Mínimos y Máximos en Regular y Bloques de Estampado de Tiempo, Bloques Opción de Tarjeta y Acumuladores.

Sección Comandos, Registros del 20000 al 26011, detalles del Medidor, Bloque de Restablecimiento, Bloque de programación, Otro bloque de comandos y Cifrado en bloque.

Sección de Ajustes programables, Registros del 30000 al 30067, todos los detalles de los ajustes se pueden programar para configurar su medidor. Sección Lecturas Secundaria.

Sección Lecturas Secundaria, Registros del 40001 al 40100, detalles del medidor, Lecturas secundarias.

A.3.2. Formato de Datos

ASCII: Caracteres ASCII empaquetados 2 por registrarse en alto, bajo orden y sin ningún carácter de terminación.

Ejemplo: "Shark 100" sería 4 registros que contienen 0x5378, 0x6172, 0x6B31, 0x3030.

SINT16/UINT16: 16-bits con signo / sin signo entero.

SINT32/UINT32: 32-bits con signo / sin signo entero, que abarca 2 registros. El registro más bajo su dirección es el medio de alto orden.

FLOAT: 32-bit IEEE número punto flotante que abarca 2 registros. El registro más bajo su dirección es la media de orden. (Es decir, contiene el Exponente) Superior (es decir, contiene el exponente).

A.3.3. Mapa MODBUS

El mapa MODBUS es bastante extenso e incluye funciones de escritura y control de registros que no utilizamos, por eso se presentan solamente la parte del mapa que se ha utilizado en el trabajo, que son: bloque primario de lectura, bloque primario de energia y bloque THD (distorsión armónica).

Modbus Address		Description ¹	Format	Range ⁸	Units or Resolution	Comments	# Reg
Hex	Decimal						
Fixed Data Section							
Identification Block							read-only
0000 - 0007	1 - 8	Meter Name	ASCII	16 char	none		8
0008 - 000F	9 - 16	Meter Serial Number	ASCII	16 char	none		8
0010 - 0010	17 - 17	Meter Type	UINT16	bit-mapped	-----t-----vw	t = transducer model (1=yes, 0=no), vw = V-switch(1 to 4)	1
0011 - 0012	18 - 19	Firmware Version	ASCII	4 char	none		2
0013 - 0013	20 - 20	Map Version	UINT16	0 to 65535	none		1
0014 - 0014	21 - 21	Meter Configuration	UINT16	bit-mapped	-----f-----	fffff = calibration frequency (50 or 60)	1
0015 - 0015	22 - 22	ASIC Version	UINT16	0-65535	none		1
0016 - 0026	23 - 39	Reserved					17
0027 - 002E	40 - 47	Reserved					8
							Block Size:
Meter Data Section²							
Primary Readings Block, 6 cycles (IEEE Floating Point)							read-only
0383 - 0384	900 - 901	Watts, 3-Ph total	FLOAT	-9999 M to +9999 M	watts		2
0385 - 0386	902 - 903	VARs, 3-Ph total	FLOAT	-9999 M to +9999 M	VARs		2
0387 - 0388	904 - 905	VA, 3-Ph total	FLOAT	-9999 M to +9999 M	VA		2
							Block Size:
6							
Primary Readings Block, 60 cycles (IEEE Floating Point)							read-only
03E7 - 03E8	1000 - 1001	Volts A-N	FLOAT	0 to 9999 M	volts		2
03E9 - 03EA	1002 - 1003	Volts B-N	FLOAT	0 to 9999 M	volts		2
03EB - 03EC	1004 - 1005	Volts C-N	FLOAT	0 to 9999 M	volts		2
03ED - 03EE	1006 - 1007	Volts A-B	FLOAT	0 to 9999 M	volts		2
03EF - 03F0	1008 - 1009	Volts B-C	FLOAT	0 to 9999 M	volts		2
03F1 - 03F2	1010 - 1011	Volts C-A	FLOAT	0 to 9999 M	volts		2
03F3 - 03F4	1012 - 1013	Amps A	FLOAT	0 to 9999 M	amps		2
03F5 - 03F6	1014 - 1015	Amps B	FLOAT	0 to 9999 M	amps		2
03F7 - 03F8	1016 - 1017	Amps C	FLOAT	0 to 9999 M	amps		2
03F9 - 03FA	1018 - 1019	Watts, 3-Ph total	FLOAT	-9999 M to +9999 M	watts		2
03FB - 03FC	1020 - 1021	VARs, 3-Ph total	FLOAT	-9999 M to +9999 M	VARs		2
03FD - 03FE	1022 - 1023	VA, 3-Ph total	FLOAT	-9999 M to +9999 M	VA		2
03FF - 0400	1024 - 1025	Power Factor, 3-Ph total	FLOAT	-1.00 to +1.00	none		2
0401 - 0402	1026 - 1027	Frequency	FLOAT	0 to 65.00	Hz		2
0403 - 0404	1028 - 1029	Neutral Current	FLOAT	0 to 9999 M	amps		2
							Block Size:
30							
Primary Energy Block							read-only

Modbus Address		Description ¹	Format	Range ⁵	Units or Resolution	Comments	# Reg
Hex	Decimal						
044B - 044C	1100 - 1101	W-hours, Received	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format	* Wh received & delivered always have opposite signs	2
044D - 044E	1102 - 1103	W-hours, Delivered	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format	* Wh received is positive for "view as load", delivered is positive for "view as generator"	2
044F - 0450	1104 - 1105	W-hours, Net	SINT32	-99999999 to 99999999	Wh per energy format		2
0451 - 0452	1106 - 1107	W-hours, Total	SINT32	0 to 99999999	Wh per energy format	* 5 to 8 digits	2
0453 - 0454	1108 - 1109	VAR-hours, Positive	SINT32	0 to 99999999	VARh per energy format	* decimal point implied, per energy format	2
0455 - 0456	1110 - 1111	VAR-hours, Negative	SINT32	0 to -99999999	VARh per energy format	* resolution of digit before decimal point = units, kilo, or mega, per energy format	2
0457 - 0458	1112 - 1113	VAR-hours, Net	SINT32	-99999999 to 99999999	VARh per energy format		2
0459 - 045A	1114 - 1115	VAR-hours, Total	SINT32	0 to 99999999	VARh per energy format		2
045B - 045C	1116 - 1117	VA-hours, Total	SINT32	0 to 99999999	VAh per energy format	* see note 10	2
Block Size							18
Primary Demand Block (IEEE Floating Point)							read-only
07CF - 07D0	2000 - 2001	Amps A, Average	FLOAT	0 to 9999 M	amps		2
07D1 - 07D2	2002 - 2003	Amps B, Average	FLOAT	0 to 9999 M	amps		2
07D3 - 07D4	2004 - 2005	Amps C, Average	FLOAT	0 to 9999 M	amps		2
07D5 - 07D6	2006 - 2007	Positive Watts, 3-Ph, Average	FLOAT	-9999 M to +9999 M	watts		2
07D7 - 07D8	2008 - 2009	Positive VARs, 3-Ph, Average	FLOAT	-9999 M to +9999 M	VARs		2
07D9 - 07DA	2010 - 2011	Negative Watts, 3-Ph, Average	FLOAT	-9999 M to +9999 M	watts		2
07DB - 07DC	2012 - 2013	Negative VARs, 3-Ph, Average	FLOAT	-9999 M to +9999 M	VARs		2
07DD - 07DE	2014 - 2015	VAs, 3-Ph, Average	FLOAT	-9999 M to +9999 M	VAs		2
07DF - 07E0	2016 - 2017	Positive PF, 3-Ph, Average	FLOAT	-1.00 to +1.00	none		2
07E1 - 07E2	2018 - 2019	Negative PF, 3-PF, Average	FLOAT	-1.00 to +1.00	none		2
Block Size							20
Primary Minimum Block (IEEE Floating Point)							read-only
0BB7 - 0BB8	3000 - 3001	Volts A-N, Minimum	FLOAT	0 to 9999 M	volts		2
0BB9 - 0BBA	3002 - 3003	Volts B-N, Minimum	FLOAT	0 to 9999 M	volts		2
0BBB - 0BBC	3004 - 3005	Volts C-N, Minimum	FLOAT	0 to 9999 M	volts		2
0BBD - 0BBE	3006 - 3007	Volts A-B, Minimum	FLOAT	0 to 9999 M	volts		2
0BBF - 0BC0	3008 - 3009	Volts B-C, Minimum	FLOAT	0 to 9999 M	volts		2
0BC1 - 0BC2	3010 - 3011	Volts C-A, Minimum	FLOAT	0 to 9999 M	volts		2
0BC3 - 0BC4	3012 - 3013	Amps A, Minimum Avg Demand	FLOAT	0 to 9999 M	amps		2
0BC5 - 0BC6	3014 - 3015	Amps B, Minimum Avg Demand	FLOAT	0 to 9999 M	amps		2
0BC7 - 0BC8	3016 - 3017	Amps C, Minimum Avg Demand	FLOAT	0 to 9999 M	amps		2
0BC9 - 0BCA	3018 - 3019	Positive Watts, 3-Ph, Minimum Avg Demand	FLOAT	0 to +9999 M	watts		2
0BCB - 0BCC	3020 - 3021	Positive VARs, 3-Ph, Minimum Avg Demand	FLOAT	0 to +9999 M	VARs		2
0BCD - 0BCE	3022 - 3023	Negative Watts, 3-Ph, Minimum Avg Demand	FLOAT	0 to +9999 M	watts		2
0BCF - 0BD0	3024 - 3025	Negative VARs, 3-Ph, Minimum Avg Demand	FLOAT	0 to +9999 M	VARs		2
0BD1 - 0BD2	3026 - 3027	VAs, 3-Ph, Minimum Avg Demand	FLOAT	-9999 M to +9999 M	VAs		2
0BD3 - 0BD4	3028 - 3029	Positive Power Factor, 3-Ph, Minimum Avg Demand	FLOAT	-1.00 to +1.00	none		2

Modbus Address		Description ¹	Format	Range ⁸	Units or Resolution	Comments	# Reg
Hex	Decimal						
0BD5 - 0BD6	3030 - 3031	Negative Power Factor, 3-Ph, Minimum Avg Demand	FLOAT	-1.00 to +1.00	none		2
0BD7 - 0BD8	3032 - 3033	Frequency, Minimum	FLOAT	0 to 65.00	Hz		2
						Block Size	34
Primary Maximum Block (IEEE Floating Point)							read-only
0C1B - 0C1C	3100 - 3101	Volts A-N, Maximum	FLOAT	0 to 9999 M	volts		2
0C1D - 0C1E	3102 - 3103	Volts B-N, Maximum	FLOAT	0 to 9999 M	volts		2
0C1F - 0C20	3104 - 3105	Volts C-N, Maximum	FLOAT	0 to 9999 M	volts		2
0C21 - 0C22	3106 - 3107	Volts A-B, Maximum	FLOAT	0 to 9999 M	volts		2
0C23 - 0C24	3108 - 3109	Volts B-C, Maximum	FLOAT	0 to 9999 M	volts		2
0C25 - 0C26	3110 - 3111	Volts C-A, Maximum	FLOAT	0 to 9999 M	volts		2
0C27 - 0C28	3112 - 3113	Amps A, Maximum Avg Demand	FLOAT	0 to 9999 M	amps		2
0C29 - 0C2A	3114 - 3115	Amps B, Maximum Avg Demand	FLOAT	0 to 9999 M	amps		2
0C2B - 0C2C	3116 - 3117	Amps C, Maximum Avg Demand	FLOAT	0 to 9999 M	amps		2
0C2D - 0C2E	3118 - 3119	Positive Watts, 3-Ph, Maximum Avg Demand	FLOAT	0 to +9999 M	watts		2
0C2F - 0C30	3120 - 3121	Positive VARs, 3-Ph, Maximum Avg Demand	FLOAT	0 to +9999 M	VARs		2
0C31 - 0C32	3122 - 3123	Negative Watts, 3-Ph, Maximum Avg Demand	FLOAT	0 to +9999 M	watts		2
0C33 - 0C34	3124 - 3125	Negative VARs, 3-Ph, Maximum Avg Demand	FLOAT	0 to +9999 M	VARs		2
0C35 - 0C36	3126 - 3127	VAs, 3-Ph, Maximum Avg Demand	FLOAT	-9999 M to +9999 M	VAs		2
0C37 - 0C38	3128 - 3129	Positive Power Factor, 3-Ph, Maximum Avg Demand	FLOAT	-1.00 to +1.00	none		2
0C39 - 0C3A	3130 - 3131	Negative Power Factor, 3-Ph, Maximum Avg Demand	FLOAT	-1.00 to +1.00	none		2
0C3B - 0C3C	3132 - 3133	Frequency, Maximum	FLOAT	0 to 65.00	Hz		2
						Block Size	34
THD Block ¹³							read-only
0F9F - 0F9F	4000 - 4000	Volts A-N, %THD	UINT16	0 to 9999, or 65535	0.1%		1
0FA0 - 0FA0	4001 - 4001	Volts B-N, %THD	UINT16	0 to 9999, or 65535	0.1%		1
0FA1 - 0FA1	4002 - 4002	Volts C-N, %THD	UINT16	0 to 9999, or 65535	0.1%		1
0FA2 - 0FA2	4003 - 4003	Amps A, %THD	UINT16	0 to 9999, or 65535	0.1%		1
0FA3 - 0FA3	4004 - 4004	Amps B, %THD	UINT16	0 to 9999, or 65535	0.1%		1
0FA4 - 0FA4	4005 - 4005	Amps C, %THD	UINT16	0 to 9999, or 65535	0.1%		1
0FA5 - 0FA5	4006 - 4006	Phase A Current 0th harmonic magnitude	UINT16	0 to 65535	none		1
0FA6 - 0FA6	4007 - 4007	Phase A Current 1st harmonic magnitude	UINT16	0 to 65535	none		1
0FA7 - 0FA7	4008 - 4008	Phase A Current 2nd harmonic magnitude	UINT16	0 to 65535	none		1
0FA8 - 0FA8	4009 - 4009	Phase A Current 3rd harmonic magnitude	UINT16	0 to 65535	none		1
0FA9 - 0FA9	4010 - 4010	Phase A Current 4th harmonic magnitude	UINT16	0 to 65535	none		1
0FAA - 0FAA	4011 - 4011	Phase A Current 5th harmonic magnitude	UINT16	0 to 65535	none		1
0FAB - 0FAB	4012 - 4012	Phase A Current 6th harmonic magnitude	UINT16	0 to 65535	none		1
0FAC - 0FAC	4013 - 4013	Phase A Current 7th harmonic magnitude	UINT16	0 to 65535	none		1
0FAD - 0FAD	4014 - 4014	Phase A Voltage 0th harmonic magnitude	UINT16	0 to 65535	none		1
0FAE - 0FAE	4015 - 4015	Phase A Voltage 1st harmonic magnitude	UINT16	0 to 65535	none		1

Modbus Address		Description ¹	Format	Range ⁵	Units or Resolution	Comments	# Reg
Hex	Decimal						
0FAF - 0FAF	4016 - 4016	Phase A Voltage 2nd harmonic magnitude	UINT16	0 to 65535	none		1
0FB0 - 0FB0	4017 - 4017	Phase A Voltage 3rd harmonic magnitude	UINT16	0 to 65535	none		1
0FB1 - 0FB8	4018 - 4025	Phase B Current harmonic magnitude			same as Phase A Current 0th to 7th harmonic magnitudes		8
0FB9 - 0FBC	4026 - 4029	Phase B Voltage harmonic magnitude			same as Phase A Voltage 0th to 3rd harmonic magnitudes		4
0FBD - 0FC4	4030 - 4037	Phase C Current harmonic magnitude			same as Phase A Current 0th to 7th harmonic magnitudes		8
0FC5 - 0FC8	4038 - 4041	Phase C Voltage harmonic magnitude			same as Phase A Voltage 0th to 3rd harmonic magnitudes		4
						Block Size	42
Phase Angle Block ⁴							
						read-only	
1003 - 1003	4100 - 4100	Phase A Current	SINT16	-1800 to +1800	0.1 degree		1
1004 - 1004	4101 - 4101	Phase B Current	SINT16	-1800 to +1800	0.1 degree		1
1005 - 1005	4102 - 4102	Phase C Current	SINT16	-1800 to +1800	0.1 degree		1
1006 - 1006	4103 - 4103	Angle, Volts A-B	SINT16	-1800 to +1800	0.1 degree		1
1007 - 1007	4104 - 4104	Angle, Volts B-C	SINT16	-1800 to +1800	0.1 degree		1
1008 - 1008	4105 - 4105	Angle, Volts C-A	SINT16	-1800 to +1800	0.1 degree		1
						Block Size	6
Status Block							
						read-only	
1387 - 1387	5000 - 5000	Meter Status	UINT16	bit-mapped	--exmpch ssssssss	exmpch = EEPROM block OK flags (e=energy, x=max, n=min, p=programmable settings, c=calibration, h=header), sssssss = state (1=Run, 2=Limp, 10=Prog Set Update via buttons, 11=Prog Set Update via IrDA, 12=Prog Set Update via COM2)	1
1388 - 1388	5001 - 5001	Limits Status ⁷	UINT16	bit-mapped	87654321 87654321	high byte is setpt 1, 0=in, 1=out low byte is setpt 2, 0=in, 1=out	1
1389 - 138A	5002 - 5003	Time Since Reset	UINT32	0 to 4294967294	4 msec	wraps around after max coun	2
						Block Size	4
Commands Section⁴							
Resets Block ⁷							
						write-only	
4E1F - 4E1F	20000 - 20000	Reset Max/Min Blocks	UINT16	password ⁸			1
4E20 - 4E20	20001 - 20001	Reset Energy Accumulators	UINT16	password ⁸			1
						Block Size	2
Meter Programming Block							
						read/conditional write	
55EF - 55EF	22000 - 22000	Initiate Programmable Settings Update	UINT16	password ⁸		meter enters PS update mod	1
55F0 - 55F0	22001 - 22001	Terminate Programmable Settings Update	UINT16	any value		meter leaves PS update mode via reset	1

Tabla 3: Mapa MODBUS medidores Shark 100S

A.4. Mapa MODBUS Electroindustries Shark 200/200S

El mapa de MODBUS para el medidor Shark 200 proporciona detalles e información acerca de las posibles lecturas del medidor y su programación. El medidor Shark® 200 puede ser programado con los botones en la caratula del medidor, o mediante el uso de software.

A.4.1. Secciones de Mapa de Registro MODBUS

El Mapa de registro MODBUS del medidor Shark® 200 [25] incluye las siguientes secciones:

Sección de Datos Fijos, registros del 1 al 47, los detalles de información fija del medidor

Sección datos del Medidor, Registros del 1000 al 12031, detalles de las lecturas del medidor, incluyendo lecturas Primaria, Bloque de Energía, Demanda de Bloque, Bloque Angulo de Fase, Bloque de Estado, Bloque THD, Mínimos y Máximos en Regular y Bloques de Estampado de Tiempo, Bloques Opción de Tarjeta y Acumuladores. Modo de funcionamiento Lecturas.

Sección Comandos, Registros del 20000 al 26011, detalles del Medidor, Bloque de Restablecimiento, Bloque de programación, Otro bloque de comandos y Cifrado en bloque.

Sección de Ajustes programables, Registros del 30000 al 33575, todos los detalles de los ajustes se pueden programar para configurar su medidor. Sección Lecturas Secundaria.

Sección Lecturas Secundaria, Registros del 40001 al 40100, detalles del medidor, Lecturas secundarias.

Sección Recuperación de Registros, Registros del 49997 al 51095, los detalles de recuperación de registros.

A.4.2. Formato de Datos

ASCII: Caracteres ASCII empaquetados 2 por registrarse en alto, bajo orden y sin ningún carácter de terminación.

Ejemplo: "Shark 100" sería 4 registros que contienen 0x5378, 0x6172, 0x6B31, 0x3030.

SINT16/UINT16: 16-bits con signo / sin signo entero.

SINT32/UINT32: 32-bits con signo / sin signo entero, que abarca 2 registros. El registro más bajo su dirección es el medio de alto orden.

FLOAT: 32-bit IEEE número punto flotante que abarca 2 registros. El registro más bajo su dirección es la media de orden. (Es decir, contiene el Exponente) Superior (es decir, contiene el exponente).

A.4.3. Mapa MODBUS

El mapa MODBUS es bastante extenso e incluye funciones de escritura y control de registros que no utilizamos, por eso se presentan solamente la parte del mapa que se ha utilizado en el trabajo, que son: bloque primario de lectura y bloque primario de energía .

Modbus Address				Description (Note 1)	Format	Range (Note 6)	Units or Resolution	Comments	# Reg
Hex	Decimal - see Section B-6								
Fixed Data Section									
Identification Block									read-only
0000	-	0007	1 - 8	Meter Name	ASCII	16 char	none		8
0008	-	000F	9 - 16	Meter Serial Number	ASCII	16 char	none		8
0010	-	0010	17 - 17	Meter Type	UINT16	bit-mapped	-----t vvvvvvvv	t = transducer model (1=yes, 0=no), vvv = V-switch: V1 = standard 200, V2 = V1 plus logging, V3 = V2 plus THD, V4 = V3 plus relays, V5 = V4 plus waveform capture up to 64 samples/cycle and 3 Meg, V6 = V4 plus waveform capture up to 512 samples/cycle and 4 Meg	1
0011	-	0012	18 - 19	Firmware Version	ASCII	4 char	none		2
0013	-	0013	20 - 20	Map Version	UINT16	0 to 65535	none		1
0014	-	0014	21 - 21	Meter Configuration	UINT16	bit-mapped	-----ccc ---fffff	ccc = CT denominator (1 or 5), fffff = calibration frequency (50 or 60)	1
0015	-	0015	22 - 22	ASIC Version	UINT16	0-65535	none		1
0016	-	0017	23 - 24	Boot Firmware Version	ASCII	4 char	none		2
0018	-	0018	25 - 25	Option Slot 1 Usage	UINT16	bit-mapped	same as register 10000 (0x270F)		1
0019	-	0019	26 - 26	Option Slot 2 Usage	UINT16	bit-mapped	same as register 11000 (0x2AF7)		1
001A	-	001D	27 - 30	Meter Type Name	ASCII	8 char	none		4
001E	-	0026	31 - 39	Reserved				Reserved	9
0027	-	002E	40 - 47	Reserved				Reserved	8
									Block Size:
									47
Meter Data Section (Note 2)									
Primary Readings Block									read-only
03E7	-	03E8	1000 - 1001	Volts A-N	FLOAT	0 to 9999 M	volts		2
03E9	-	03EA	1002 - 1003	Volts B-N	FLOAT	0 to 9999 M	volts		2
03EB	-	03EC	1004 - 1005	Volts C-N	FLOAT	0 to 9999 M	volts		2
03ED	-	03EE	1006 - 1007	Volts A-B	FLOAT	0 to 9999 M	volts		2
03EF	-	03F0	1008 - 1009	Volts B-C	FLOAT	0 to 9999 M	volts		2
03F1	-	03F2	1010 - 1011	Volts C-A	FLOAT	0 to 9999 M	volts		2
03F3	-	03F4	1012 - 1013	Amps A	FLOAT	0 to 9999 M	amps		2
03F5	-	03F6	1014 - 1015	Amps B	FLOAT	0 to 9999 M	amps		2
03F7	-	03F8	1016 - 1017	Amps C	FLOAT	0 to 9999 M	amps		2
03F9	-	03FA	1018 - 1019	Watts, 3-Ph total	FLOAT	-9999 M to +9999 M	watts		2
03FB	-	03FC	1020 - 1021	VARs, 3-Ph total	FLOAT	-9999 M to +9999 M	VARs		2
03FD	-	03FE	1022 - 1023	VAs, 3-Ph total	FLOAT	-9999 M to +9999 M	VAs		2
03FF	-	0400	1024 - 1025	Power Factor, 3-Ph total	FLOAT	-1.00 to +1.00	none		2
0401	-	0402	1026 - 1027	Frequency	FLOAT	0 to 65.00	Hz		2

0403	-	0404	1028	-	1029	Neutral Current	FLOAT	0 to 9999 M	amps		2
0405	-	0405	1030	-	1031	Watts, Phase A	FLOAT	-9999 M to +9999 M	watts		2
0407	-	0408	1032	-	1033	Watts, Phase B	FLOAT	-9999 M to +9999 M	watts		2
0409	-	040A	1034	-	1035	Watts, Phase C	FLOAT	-9999 M to +9999 M	watts		2
040B	-	040C	1036	-	1037	VARs, Phase A	FLOAT	-9999 M to +9999 M	VARs		2
040D	-	040E	1038	-	1039	VARs, Phase B	FLOAT	-9999 M to +9999 M	VARs		2
040F	-	0410	1040	-	1041	VARs, Phase C	FLOAT	-9999 M to +9999 M	VARs		2
0411	-	0412	1042	-	1043	VAs, Phase A	FLOAT	-9999 M to +9999 M	VAs		2
0413	-	0414	1044	-	1045	VAs, Phase B	FLOAT	-9999 M to +9999 M	VAs		2
0415	-	0416	1046	-	1047	VAs, Phase C	FLOAT	-9999 M to +9999 M	VAs		2
0417	-	0418	1048	-	1049	Power Factor, Phase A	FLOAT	-1.00 to +1.00	none		2
0419	-	041A	1050	-	1051	Power Factor, Phase B	FLOAT	-1.00 to +1.00	none		2
041B	-	041C	1052	-	1053	Power Factor, Phase C	FLOAT	-1.00 to +1.00	none		2
041D	-	0425	1054	-	1052	Reserved				Reserved	9
										Block Size:	63
Primary Energy Block										read-only	
050B	-	050C	1500	-	1501	W-hours, Received	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format	* Wh received & delivered always have opposite signs	2
050D	-	050E	1502	-	1503	W-hours, Delivered	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format	* Wh received is positive for "view as load", delivered is positive for "view as generator"	2
050F	-	050E0	1504	-	1505	W-hours, Net	SINT32	-99999999 to 99999999	Wh per energy format	* 5 to 8 digits	2
05E1	-	05E2	1506	-	1507	W-hours, Total	SINT32	0 to 99999999	Wh per energy format		2
05E3	-	05E4	1508	-	1509	VAR-hours, Positive	SINT32	0 to 99999999	VARh per energy format	* decimal point implied, per energy format	2
05E5	-	05E6	1510	-	1511	VAR-hours, Negative	SINT32	0 to -99999999	VARh per energy format	* resolution of digit before decimal point = units, kilo, or mega, per energy format	2
05E7	-	05E8	1512	-	1513	VAR-hours, Net	SINT32	-99999999 to 99999999	VARh per energy format		2
05E9	-	05EA	1514	-	1515	VAR-hours, Total	SINT32	0 to 99999999	VARh per energy format	* see note 10	2
05EB	-	05EC	1516	-	1517	VA-hours, Total	SINT32	0 to 99999999	VAh per energy format		2
05ED	-	05EE	1518	-	1519	W-hours, Received, Phase A	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format		2
05EF	-	05F0	1520	-	1521	W-hours, Received, Phase B	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format		2
05F1	-	05F2	1522	-	1523	W-hours, Received, Phase C	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format		2
05F3	-	05F4	1524	-	1525	W-hours, Delivered, Phase A	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format		2
05F5	-	05F6	1526	-	1527	W-hours, Delivered, Phase B	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format		2
05F7	-	05F8	1528	-	1529	W-hours, Delivered, Phase C	SINT32	0 to 99999999 or 0 to -99999999	Wh per energy format		2
05F9	-	05FA	1530	-	1531	W-hours, Net, Phase A	SINT32	-99999999 to 99999999	Wh per energy format		2
05FB	-	05FC	1532	-	1533	W-hours, Net, Phase B	SINT32	-99999999 to 99999999	Wh per energy format		2
05FD	-	05FE	1534	-	1535	W-hours, Net, Phase C	SINT32	-99999999 to 99999999	Wh per energy format		2
05FF	-	0600	1536	-	1537	W-hours, Total, Phase A	SINT32	0 to 99999999	Wh per energy format		2
0601	-	0602	1538	-	1539	W-hours, Total, Phase B	SINT32	0 to 99999999	Wh per energy format		2
0603	-	0604	1540	-	1541	W-hours, Total, Phase C	SINT32	0 to 99999999	Wh per energy format		2
0605	-	0605	1542	-	1543	VAR-hours, Positive, Phase A	SINT32	0 to 99999999	VARh per energy format		2
0607	-	0608	1544	-	1545	VAR-hours, Positive, Phase B	SINT32	0 to 99999999	VARh per energy format		2
0609	-	060A	1546	-	1547	VAR-hours, Positive, Phase C	SINT32	0 to 99999999	VARh per energy format		2
060B	-	060C	1548	-	1549	VAR-hours, Negative, Phase A	SINT32	0 to -99999999	VARh per energy format		2
060D	-	060E	1550	-	1551	VAR-hours, Negative, Phase B	SINT32	0 to -99999999	VARh per energy format		2

060F	-	0610	1552	-	1553	VAR-hours, Negative, Phase C	SINT32	0 to -99999999	VARh per energy format	2
0611	-	0612	1554	-	1555	VAR-hours, Net, Phase A	SINT32	-99999999 to 99999999	VARh per energy format	2
0613	-	0614	1556	-	1557	VAR-hours, Net, Phase B	SINT32	-99999999 to 99999999	VARh per energy format	2
0615	-	0616	1558	-	1559	VAR-hours, Net, Phase C	SINT32	-99999999 to 99999999	VARh per energy format	2
0617	-	0618	1560	-	1561	VAR-hours, Total, Phase A	SINT32	0 to 99999999	VARh per energy format	2
0619	-	061A	1562	-	1563	VAR-hours, Total, Phase B	SINT32	0 to 99999999	VARh per energy format	2
061B	-	061C	1564	-	1565	VAR-hours, Total, Phase C	SINT32	0 to 99999999	VARh per energy format	2
061D	-	061E	1566	-	1567	VA-hours, Phase A	SINT32	0 to 99999999	VAh per energy format	2
061F	-	0620	1568	-	1569	VA-hours, Phase B	SINT32	0 to 99999999	VAh per energy format	2
0621	-	0622	1570	-	1571	VA-hours, Phase C	SINT32	0 to 99999999	VAh per energy format	2
									Block Size:	72
Primary Demand Block										read-only
07CF	-	07D0	2000	-	2001	Amps A, Average	FLOAT	0 to 9999 M	amps	2
07D1	-	07D2	2002	-	2003	Amps B, Average	FLOAT	0 to 9999 M	amps	2
07D3	-	07D4	2004	-	2005	Amps C, Average	FLOAT	0 to 9999 M	amps	2
07D5	-	07D6	2006	-	2007	Positive Watts, 3-Ph, Average	FLOAT	-9999 M to +9999 M	watts	2
07D7	-	07D8	2008	-	2009	Positive VARs, 3-Ph, Average	FLOAT	-9999 M to +9999 M	VARs	2
07D9	-	07DA	2010	-	2011	Negative Watts, 3-Ph, Average	FLOAT	-9999 M to +9999 M	watts	2
07DB	-	07DC	2012	-	2013	Negative VARs, 3-Ph, Average	FLOAT	-9999 M to +9999 M	VARs	2
07DD	-	07DE	2014	-	2015	VAs, 3-Ph, Average	FLOAT	-9999 M to +9999 M	VAs	2
07DF	-	07E0	2016	-	2017	Positive PF, 3-Ph, Average	FLOAT	-1.00 to +1.00	none	2
07E1	-	07E2	2018	-	2019	Negative PF, 3-PF, Average	FLOAT	-1.00 to +1.00	none	2
07E3	-	07E4	2020	-	2021	Neutral Current, Average	FLOAT	0 to 9999 M	amps	2
07E5	-	07E6	2022	-	2023	Positive Watts, Phase A, Average	FLOAT	-9999 M to +9999 M	watts	2
07E7	-	07E8	2024	-	2025	Positive Watts, Phase B, Average	FLOAT	-9999 M to +9999 M	watts	2
07E9	-	07EA	2026	-	2027	Positive Watts, Phase C, Average	FLOAT	-9999 M to +9999 M	watts	2
07EB	-	07EC	2028	-	2029	Positive VARs, Phase A, Average	FLOAT	-9999 M to +9999 M	VARs	2
07ED	-	07EE	2030	-	2031	Positive VARs, Phase B, Average	FLOAT	-9999 M to +9999 M	VARs	2
07EF	-	07F0	2032	-	2033	Positive VARs, Phase C, Average	FLOAT	-9999 M to +9999 M	VARs	2
07F1	-	07F2	2034	-	2035	Negative Watts, Phase A, Average	FLOAT	-9999 M to +9999 M	watts	2
07F3	-	07F4	2036	-	2037	Negative Watts, Phase B, Average	FLOAT	-9999 M to +9999 M	watts	2
07F5	-	07F6	2038	-	2039	Negative Watts, Phase C, Average	FLOAT	-9999 M to +9999 M	watts	2
07F7	-	07F8	2040	-	2041	Negative VARs, Phase A, Average	FLOAT	-9999 M to +9999 M	VARs	2
07F9	-	07FA	2042	-	2043	Negative VARs, Phase B, Average	FLOAT	-9999 M to +9999 M	VARs	2
07FB	-	07FC	2044	-	2045	Negative VARs, Phase C, Average	FLOAT	-9999 M to +9999 M	VARs	2
07FD	-	07FE	2046	-	2047	VAs, Phase A, Average	FLOAT	-9999 M to +9999 M	VAs	2
07FF	-	0800	2048	-	2049	VAs, Phase B, Average	FLOAT	-9999 M to +9999 M	VAs	2
0801	-	0802	2050	-	2051	VAs, Phase C, Average	FLOAT	-9999 M to +9999 M	VAs	2
0803	-	0804	2052	-	2053	Positive PF, Phase A, Average	FLOAT	-1.00 to +1.00	none	2
0805	-	0806	2054	-	2055	Positive PF, Phase B, Average	FLOAT	-1.00 to +1.00	none	2
0807	-	0808	2056	-	2057	Positive PF, Phase C, Average	FLOAT	-1.00 to +1.00	none	2
0809	-	080A	2058	-	2059	Negative PF, Phase A, Average	FLOAT	-1.00 to +1.00	none	2
080B	-	080C	2060	-	2061	Negative PF, Phase B, Average	FLOAT	-1.00 to +1.00	none	2
080D	-	080E	2062	-	2063	Negative PF, Phase C, Average	FLOAT	-1.00 to +1.00	none	2
									Block Size:	64

Tabla 4: Mapa MODBUS medidores Shark 200/200S

A.5. Diccionario de datos

A.5.1. Lista de Tablas base de datos 'spud'

Nombre	Descripción
Agronomia	Almacena los datos de las mediciones de la subestación de Agronomía
AgronomiaDecanato	Almacena los datos de las mediciones de la subestación Agronomía Decanato
AgronomiaGalera	Almacena los datos de las mediciones de la subestación Agronomía Galera
AgronomiaQuimica	Almacena los datos de las mediciones de la subestación Agronomía Química
Artes	Almacena los datos de las mediciones de la subestación Artes
AuditoriumMarmol	Almacena los datos de las mediciones de la subestación Auditorium Mármol
Cafetines	Almacena los datos de las mediciones de la subestación Auditorium Cafetines
ComedorUES	Almacena los datos de las mediciones de la subestación Comedor UES
Derecho	Almacena los datos de las mediciones de la subestación Derecho
Economia1	Almacena los datos de las mediciones de la subestación Economía 1
Economia2	Almacena los datos de las mediciones de la subestación Economía 2
Economia3	Almacena los datos de las mediciones de la subestación Economía 3
Economia4	Almacena los datos de las mediciones de la subestación Economía 4
Economia5	Almacena los datos de las mediciones de la subestación Economía 5
Economia6	Almacena los datos de las mediciones de la subestación Economía 6
HORAS	Almacena los valores de las horas para realizar la consulta Web
Humanidades1	Almacena los datos de las mediciones de la subestación Humanidades 1
Humanidades2	Almacena los datos de las mediciones de la subestación Humanidades 2
Humanidades3	Almacena los datos de las mediciones de la subestación Humanidades 3
intervalosreportes	Almacena los valores de los intervalos de consulta Web, minutos, horas
limits	Almacena los valores de los límites de la calidad de la señal en SPUD
links	Almacena los enlaces entre los nodos
MecanicaComplejo	Almacena los datos de las mediciones de la subestación Mecánica Complejo Deportivo
Medicina	Almacena los datos de las mediciones de la subestación Medicina

Nombre	Descripción
Menus	Almacena los datos para el menú del programa de consulta Web
missing_nodes	Usado por SPUD para almacenar los nodos encontrados pero no agregados
modulos	Usado por la aplicación Web para manejar los módulos de control de la aplicación
nodes	Almacena la información de cada nodo de la red mesh
Odontologia1	Almacena los datos de las mediciones de la subestación Odontología 1
Odontologia2	Almacena los datos de las mediciones de la subestación Odontología 2
Odontologia3	Almacena los datos de las mediciones de la subestación Odontología 3
OdontologiaImprenta	Almacena los datos de las mediciones de la subestación Odontología Imprenta
Perfiles	Perfiles de los usuarios del programa de consulta Web
Periodismo	Almacena los datos de las mediciones de la subestación Periodismo
permisosperfiles	Permisos de los perfiles de la aplicación Web
permisosusuarios	Permisos de los usuarios de la aplicación Web
PrimarioFia	Almacena los datos de las mediciones de la subestación Primaria de la FIA
Psicologia	Almacena los datos de las mediciones de la subestación Psicología
Quimica	Almacena los datos de las mediciones de la subestación Química
QuimicalImprenta	Almacena los datos de las mediciones de la subestación Química Imprenta
Rectoria	Almacena los datos de las mediciones de la subestación Rectoría
subestaciones	Almacena el nombre de las subestaciones, usado en la aplicación Web
Usuarios	Almacena los usuarios de la aplicación Web

Tabla 5: Tablas base de datos 'spud'

A.5.2. Elementos Datos

En las siguientes tablas se muestran los datos almacenados en la base de datos 'spud' para los medidores Shark 100S y Shark 200/200S.

A.5.2.1. Datos comunes para los medidores Shark 100S y Shark 200/200S

Código	Nombre	Tipo de Dato	Descripción
Fecha_hora	Fecha y hora	timestamp	Almacena la fecha y hora de la medición en formato AAAA-MM-DD HH:MM:SS
Va	Voltaje línea fase A	float	Almacena el valor del voltaje de la fase A
Vb	Voltaje línea fase B	float	Almacena el valor del voltaje de la fase B
Vc	Voltaje línea fase C	float	Almacena el valor del voltaje de la fase C
Vab	Voltaje línea fases A-B	float	Almacena el valor del voltaje de línea entre las fases A y B
Vbc	Voltaje línea fases B-C	float	Almacena el valor del voltaje de línea entre las fases B y C
Vca	Voltaje línea fases C-A	float	Almacena el valor del voltaje línea entre las fases C y A
Ia	Corriente fase A	float	Almacena el valor de la corriente de la fase A
Ib	Corriente fase B	float	Almacena el valor de la corriente de la fase B
Ic	Corriente fase C	float	Almacena el valor de la corriente de la fase C
P	Potencia Activa	float	Almacena el valor de la Potencia Activa total de las tres fases
Q	Potencia Reactiva	float	Almacena el valor de la Potencia Reactiva total de las tres fases
S	Potencia Aparente	float	Almacena el valor de la Potencia Aparente total de las tres fases
FP	Factor de Potencia	float	Almacena el valor del Factor de Potencia total de las tres fases
F	Frecuencia	float	Almacena el valor de la Frecuencia
Ineutral	Corriente en el Neutro	float	Almacena el valor de la corriente en el Neutro
WhRec	Watts-horas recibidos	Int	Almacena el valor de los Watt-horas recibidos, es un valor acumulado
WhDes	Watts-horas despachadas	Int	Almacena el valor de los Watt-horas despachados, es un valor acumulado
WhNet	Watts-horas netos	Int	Almacena el valor de los Watt-horas netos, es un valor acumulado
WhTot	Watt-horas totales	Int	Almacena el valor de los Watt-horas totales, es un valor acumulado

Código	Nombre	Tipo de Dato	Descripción
VARhPos	VAR-horas positivos	Int	Almacena el valor de los Voltios amperios reactivos-horas positivos, es un valor acumulado
VARhNeg	VAR-horas negativos	Int	Almacena el valor de los Voltios amperios reactivos-horas negativos, es un valor acumulado
VARhNet	VAR-horas netos	Int	Almacena el valor de los Voltios amperios reactivos-horas negativos, es un valor acumulado
VARhTot	VAR-horas totales	Int	Almacena el valor de los Voltios amperios reactivos-horas totales, es un valor acumulado
VAhTot	VA-horas totales	Int	Almacena el valor de los Voltios amperios-horas totales, es un valor acumulado

Tabla 6: Datos Comunes Shark 100S y Shark 200/200S

A.5.2.2. Datos exclusivos para los medidores Shark 100S

Código	Nombre	Tipo de Dato	Descripción
THDVa	THD Voltaje fase A	float	Almacena el valor del total de la distorsión armónica del voltaje de la fase A
THDVb	THD Voltaje fase B	float	Almacena el valor del total de la distorsión armónica del voltaje de la fase B
THDVc	THD Voltaje fase C	float	Almacena el valor del total de la distorsión armónica del voltaje de la fase C
THDIa	THD corriente fase A	float	Almacena el valor del total de la distorsión armónica de corriente de la fase A
THDIb	THD corriente fase B	float	Almacena el valor del total de la distorsión armónica de corriente de la fase B
THDIc	THD corriente fase C	float	Almacena el valor del total de la distorsión armónica de corriente de la fase C
Ia0	Armónico fundamental de corriente fase A	Int	Almacena el valor del armónico fundamental de corriente de la fase A
Ia1	Primer armónico de corriente fase A	Int	Almacena el valor del primer armónico de la corriente de la fase A
Ia2	Segundo armónico de corriente fase A	Int	Almacena el valor del segundo armónico de la corriente de la fase A
Ia3	Tercer armónico de corriente fase A	Int	Almacena el valor del tercer armónico de la corriente de la fase A
Ia4	Cuarto armónico de corriente	Int	Almacena el valor del cuarto armónico de la corriente de la fase A

Código	Nombre	Tipo de Dato	Descripción
	fase A		
la5	Quinto armónico de corriente fase A	Int	Almacena el valor del quinto armónico de la corriente de la fase A
la6	Sexto armónico de corriente fase A	Int	Almacena el valor del sexto armónico de la corriente de la fase A
la7	Séptimo armónico de corriente fase A	Int	Almacena el valor del séptimo armónico de la corriente de la fase A
Va0	Armónico fundamental de voltaje fase A	Int	Almacena el valor del armónico fundamental de voltaje de la fase A
Va1	Primer armónico de voltaje fase A	Int	Almacena el valor del primer armónico del voltaje de la fase A
Va2	Segundo armónico de voltaje fase A	Int	Almacena el valor del segundo armónico del voltaje de la fase A
lb0	Armónico fundamental de corriente fase B	Int	Almacena el valor del armónico fundamental de corriente de la fase B
lb1	Primer armónico de corriente fase B	Int	Almacena el valor del primer armónico de la corriente de la fase B
lb2	Segundo armónico de corriente fase B	Int	Almacena el valor del segundo armónico de la corriente de la fase B
lb3	Tercer armónico de corriente fase B	Int	Almacena el valor del tercer armónico de la corriente de la fase B
lb4	Cuarto armónico de corriente fase B	Int	Almacena el valor del cuarto armónico de la corriente de la fase B
lb5	Quinto armónico de corriente fase B	Int	Almacena el valor del quinto armónico de la corriente de la fase B
lb6	Sexto armónico de corriente fase B	Int	Almacena el valor del sexto armónico de la corriente de la fase B
lb7	Séptimo armónico de corriente fase B	Int	Almacena el valor del séptimo armónico de la corriente de la fase B
Vb0	Armónico fundamental de voltaje fase B	Int	Almacena el valor del armónico fundamental de voltaje de la fase B
Vb1	Primer armónico de voltaje	Int	Almacena el valor del primer armónico del voltaje de la fase B

Código	Nombre	Tipo de Dato	Descripción
	fase B		
Vb2	Segundo armónico de voltaje fase B	Int	Almacena el valor del segundo armónico del voltaje de la fase B
Ic0	Armónico fundamental de corriente fase C	Int	Almacena el valor del armónico fundamental de corriente de la fase C
Ic1	Primer armónico de corriente fase C	Int	Almacena el valor del primer armónico de la corriente de la fase C
Ic2	Segundo armónico de corriente fase C	Int	Almacena el valor del segundo armónico de la corriente de la fase C
Ic3	Tercer armónico de corriente fase C	Int	Almacena el valor del tercer armónico de la corriente de la fase C
Ic4	Cuarto armónico de corriente fase C	Int	Almacena el valor del cuarto armónico de la corriente de la fase C
Ic5	Quinto armónico de corriente fase C	Int	Almacena el valor del quinto armónico de la corriente de la fase C
Ic6	Sexto armónico de corriente fase C	Int	Almacena el valor del sexto armónico de la corriente de la fase C
Ic7	Séptimo armónico de corriente fase C	Int	Almacena el valor del séptimo armónico de la corriente de la fase C
Vc0	Armónico fundamental de voltaje fase C	Int	Almacena el valor del armónico fundamental de voltaje de la fase C
Vc1	Primer armónico de voltaje fase C	Int	Almacena el valor del primer armónico del voltaje de la fase C
Vc2	Segundo armónico de voltaje fase C	Int	Almacena el valor del segundo armónico del voltaje de la fase C

Tabla 7: Datos exclusivos Shark 100S

A.5.2.3. Datos exclusivos para los medidores Shark 200/200S

Código	Nombre	Tipo de Dato	Descripción
Pa	Potencia Activa fase A	float	Almacena el valor de la Potencia Activa de la fase A
Pb	Potencia Activa fase B	float	Almacena el valor de la Potencia Activa de la fase B
Pc	Potencia Activa fase C	float	Almacena el valor de la Potencia Activa de la fase C
Qa	Potencia Reactiva fase A	float	Almacena el valor de la Potencia Reactiva de la fase A
Qb	Potencia Reactiva fase B	float	Almacena el valor de la Potencia Reactiva de la fase B
Qc	Potencia Reactiva fase C	float	Almacena el valor de la Potencia Reactiva de la fase C
Sa	Potencia Aparente fase A	float	Almacena el valor de la Potencia Aparente de la fase A
Sb	Potencia Aparente fase B	float	Almacena el valor de la Potencia Aparente de la fase B
Sc	Potencia Aparente fase C	float	Almacena el valor de la Potencia Aparente de la fase C
FPa	Factor de Potencia fase A	float	Almacena el valor del Factor de Potencia de la fase A
FPb	Factor de Potencia fase B	float	Almacena el valor del Factor de Potencia de la fase B
FPc	Factor de Potencia fase C	float	Almacena el valor del Factor de Potencia de la fase C
WhReca	Watts-horas recibidos fase A	Int	Almacena el valor de los Watt-horas recibidos de la fase A, es un valor acumulado
WhRecb	Watts-horas recibidos fase B	Int	Almacena el valor de los Watt-horas recibidos de la fase B, es un valor acumulado
WhRecc	Watts-horas recibidos fase C	Int	Almacena el valor de los Watt-horas recibidos de la fase C, es un valor acumulado
WhDesa	Watts-horas despachadas fase A	Int	Almacena el valor de los Watt-horas despachados de la fase A, es un valor acumulado
WhDesb	Watts-horas despachadas fase B	Int	Almacena el valor de los Watt-horas despachados de la fase B, es un valor acumulado
WhDesc	Watts-horas despachadas fase C	Int	Almacena el valor de los Watt-horas despachados de la fases C, es un valor acumulado
WhNeta	Watts-horas netos fase A	Int	Almacena el valor de los Watt-horas netos de la fase A, es un valor acumulado
WhNetb	Watts-horas netos fase B	Int	Almacena el valor de los Watt-horas netos de la fase B, es un valor acumulado

Código	Nombre	Tipo de Dato	Descripción
WhNetc	Watts-horas netos fase C	Int	Almacena el valor de los Watt-horas netos de la fase C, es un valor acumulado
WhTota	Watts-horas totales fase A	Int	Almacena el valor de los Watt-horas totales de la fase A, es un valor acumulado
WhTotb	Watts-horas totales fase B	Int	Almacena el valor de los Watt-horas totales de la fase B, es un valor acumulado
WhTotc	Watts-horas totales fase C	Int	Almacena el valor de los Watt-horas totales de la fase C, es un valor acumulado
VArhPosa	VAR-horas positivos fase A	Int	Almacena el valor de los Voltios amperios reactivos-horas positivos de la fase A, es un valor acumulado
VArhPosb	VAR-horas positivos fase B	Int	Almacena el valor de los Voltios amperios reactivos-horas positivos de la fase B, es un valor acumulado
VArhPosc	VAR-horas positivos fase C	Int	Almacena el valor de los Voltios amperios reactivos-horas positivos de la fase C, es un valor acumulado
VArhNega	VAR-horas negativos fase A	Int	Almacena el valor de los Voltios amperios reactivos-horas negativos de la fase A, es un valor acumulado
VArhNegb	VAR-horas negativos fase B	Int	Almacena el valor de los Voltios amperios reactivos-horas negativos de la fase B, es un valor acumulado
VArhNegc	VAR-horas negativos fase C	Int	Almacena el valor de los Voltios amperios reactivos-horas negativos de la fase C, es un valor acumulado
VArhNeta	VAR-horas netos fase A	Int	Almacena el valor de los Voltios amperios reactivos-horas netos de la fase A, es un valor acumulado
VArhNetb	VAR-horas netos fase B	Int	Almacena el valor de los Voltios amperios reactivos-horas netos de la fase B, es un valor acumulado
VArhNetc	VAR-horas netos fase C	Int	Almacena el valor de los Voltios amperios reactivos-horas netos de la fase C, es un valor acumulado
VArhTota	VAR-horas totales fase A	Int	Almacena el valor de los Voltios amperios reactivos-horas totales de la fase A, es un valor acumulado
VArhTotb	VAR-horas totales fase B	Int	Almacena el valor de los Voltios amperios reactivos-horas totales de la fase B, es un valor acumulado
VArhTotc	VAR-horas totales fase C	Int	Almacena el valor de los Voltios amperios reactivos-horas totales de la fase C, es un valor acumulado

Código	Nombre	Tipo de Dato	Descripción
VAhTota	VA-horas totales fase A	Int	Almacena el valor de los Voltios amperios-horas totales de la fase A, es un valor acumulado
VAhTotb	VA-horas totales fase B	Int	Almacena el valor de los Voltios amperios-horas totales de la fase B, es un valor acumulado
VAhTotc	VA-horas totales fase C	Int	Almacena el valor de los Voltios amperios-horas totales de la fase C, es un valor acumulado

Tabla 8: Datos exclusivos Shark 200

A.6. Medidor Shark 100S

Medidor de Electricidad Multifunción de Alto desempeño y Bajo Costo, la descripción del medidor a continuación está referida al medidor Shark 100S [23]

A.6.1. Descripción del Hardware

El Shark 100S es un medidor de energía multifunción diseñado para ser utilizado en subestaciones eléctricas, tableros y como medidor de energía para el equipos de OEM's. La unidad proporciona la medición de múltiples funciones de todos los parámetros eléctricos.

La unidad está diseñada con capacidades avanzadas de medición, permitiendo que alcance Exactitud del alto desempeño. El Shark 100S está especificado como medidor de energía clase 0.2% para uso de facturación así como un medidor altamente exacto para la indicación en tablero.

El medidor Shark 100S proporciona un HOST de capacidades adicionales, incluyendo su estándar de RS-485, Ethernet RJ-450 puerto IrDA, protocolos MODBUS RTU, MODBUS ASCII, y DNP 3.0



Figura 29: Medidor Electroindustries Shark 100S

A.6.2. Características del medidor Shark 100S

- Clase 0,2% medidor de facturación certificable y medición de demanda
- Cumple con las clases ANSI C12.20 (0,2%) e IEC 687 (0,2%)
- Medición multifunción incluyendo voltaje, corriente, potencia, frecuencia, energía, etc.
- Medición de calidad de energía (% THD y Límites de alarma)
- Tecnología V-Switch™ - actualizable en campo sin retirar medidor
- Barra analógica de % de carga.
- Fácil de usar, programación desde la carátula del medidor (funciones básicas)
- Puerto IrDA para lectura remota a través de PDA

- Comunicación Serial RS-485 ó RJ-45, Comunicación MODBUS.
- Ethernet Inalámbrico Wi-Fi, y Ethernet por cable

La tabla siguiente lista los valores medidos disponibles en Tiempo Real, Promedio, Máximos y Mínimos.

Valores Medidos por el Shark 100S

Valores Medidor	Tiempo Real	Promedio	Max	Min
Voltaje L-N	X		X	X
Voltaje L-L	X		X	X
Corriente por Fase	X	X	X	X
Corriente en el Neutro	X			
Watts	X	X	X	X
VAR	X	X	X	X
VA	X	X	X	X
+Watt-hr	X	X	X	X
-Watt-hr	X			
Watt-hr Net	X			
+VAR-hr	X			
-VAR-hr	X			
VAR-hr Net	X			
VA-hr	X			
Frecuencia	X		X	X
%THD	X		X	X
Ángulos de Voltajes	X			
Ángulos de Corrientes	X			
Barra % de Carga	X			

Tabla 9: Parámetros obtenidos por el medidor Shark 100S

El medidor Shark mide armónicos hasta el 7º orden de corriente y el 3º orden de voltaje



Figura 30: Shark 100S y MP

A.7. Medidor Shark 200S

Medidor de Electricidad Multifunción de Alto desempeño y Bajo Costo, la descripción del medidor a continuación está referida al medidor Shark 200S [24], la diferencia con el Shark 100S es la capacidad de almacenar mediciones por tener una memoria interna de 2 MB.

A.7.1. Descripción del Hardware

El Shark 200S es un medidor de energía multifunción diseñado para ser utilizado en subestaciones eléctricas, tableros y como medidor de energía para el equipos de OEM's. La unidad proporciona la medición de múltiples funciones de todos los parámetros eléctricos.

La unidad está diseñada con capacidades avanzadas de medición, permitiendo que alcance Exactitud del alto desempeño. El Shark 200S está especificado como medidor de energía clase 0.2% para uso de facturación así como un medidor altamente exacto para la indicación en tablero.

El sub-medidor Shark 200S tiene hasta 2 MB de memoria para el registro y grabación de datos. Esta ofrece 3 localidades de memoria para tendencias históricas, 1 localidad para Límites y Alarmas, y 1 localidad para Eventos del Sistema.



Figura 31: Medidor Electroindustries Shark 200S

A.7.2. Características del medidor Shark 200S

- Clase 0,2% medidor de facturación certificable y medición de demanda (solo Clase 10)
- Cumple con las clases ANSI C12.20 (0,2%) e IEC 62053-22 (0,2%)
- Medición multifunción incluyendo voltaje, corriente, potencia, frecuencia, energía, etc.
- Pantalla tipo LED de 3 líneas, de alto brillo
- Memoria Flash de 2MB, para el registro y grabación de datos.

- Reloj de Tiempo Real, para el estampado de tiempo de los registros.
- Barra analógica de % de carga, para percepción de un medidor análogo..
- Protocolos de comunicación, Modbus RTU, Modbus TCP (sobre Ethernet)
- Comunicación Serial RS485
- Fácil de programar desde su teclado frontal
- Ethernet Inalámbrico WiFi, y Ethernet por cable
- Interface directa con la mayoría de Edificios Administradores de Energía
- Puerto IrDA para Lectura Remota a través de Laptop PC.
- DNP 3.0.

La tabla siguiente lista los valores medidos disponibles en Tiempo Real, Promedio, Máximos y Mínimos.

Valores Medidos por el Shark 200S

Valores Medidor	Tiempo Real	Promedio	Max	Min
Voltaje L-N	X		X	X
Voltaje L-L	X		X	X
Corriente por Fase	X	X	X	X
Corriente en el Neutro	X			
Watts	X	X	X	X
VAR	X	X	X	X
VA	X	X	X	X
+Watt-hr	X	X	X	X
-Watt-hr	X			
Watt-hr Net	X			
+VAR-hr	X			
-VAR-hr	X			
VAR-hr Net	X			
VA-hr	X			
Frecuencia	X		X	X
%THD	X		X	X
Ángulos de Voltajes	X			
Ángulos de Corrientes	X			
Barra % de Carga	X			

Tabla 10: Parámetros obtenidos por el medidor Shark 200S

A.8. Datasheet Nanostation 2 [16]

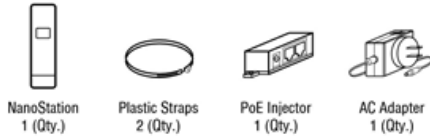


SYSTEM INFORMATION							
Processor Specs	Atheros AR2315 SOC, MIPS 4KC, 180MHz						
Memory Information	16MB SDRAM, 4MB Flash						
Networking Interface	1 X 10/100 BASE-TX (Cat. 5, RJ-45) Ethernet Interface						
REGULATORY / COMPLIANCE INFORMATION							
Wireless Approvals	FCC Part 15.247, IC RS210, CE						
RoHS Compliance	YES						
RADIO OPERATING FREQUENCY 2412-2462 MHz							
TX SPECIFICATIONS			RX SPECIFICATIONS				
	DataRate	TX Power	Tolerance		DataRate	Sensitivity	Tolerance
802.11b	1Mbps	26 dBm	+/-1dB	802.11b	1Mbps	-97 dBm	+/-1dB
	2Mbps	26 dBm	+/-1dB		2Mbps	-96 dBm	+/-1dB
	5.5Mbps	26 dBm	+/-1dB		5.5Mbps	-95 dBm	+/-1dB
	11Mbps	26 dBm	+/-1dB		11Mbps	-92 dBm	+/-1dB
g OFDM	6Mbps	26 dBm	+/-1dB	g OFDM	6Mbps	-94 dBm	+/-1dB
	9Mbps	26 dBm	+/-1dB		9Mbps	-93 dBm	+/-1dB
	12Mbps	26 dBm	+/-1dB		12Mbps	-91 dBm	+/-1dB
	18Mbps	26 dBm	+/-1dB		18Mbps	-90 dBm	+/-1dB
RANGE PERFORMANCE							
Outdoor (BaseStation Antenna Dependent):			Over 15km				
INTEGRATED ADAPTIVE ANTENNA POLARITY + EXTERNAL ANTENNA SUPPORT (4 OPTIONS TOTAL)							
Gain	10dBi (2400-2483.5MHz)		External Connector	RP-SMA			
Polarization	Multi-Polarized		3dB Beamwidth Elevation	30 degrees			
Polarization Selection	Software Controlled		3dB Beamwidth Azimuth	60 degrees			
Azimuth		Elevation					
PHYSICAL / ELECTRICAL / ENVIRONMENTAL							
Enclosure Size	26.4 cm x 8 cm x 3cm						
Weight	0.4kg						
Enclosure Characteristics	Outdoor UV Stabilized Plastic						
Mounting Kit	Pole Mounting Kit included						
Max Power Consumption	4 Watts						
Power Supply	12V, 1A (12 Watts). Supply and injector included						
Power Method	Passive Power over Ethernet (pairs 4,5+; 7,8 return)						
Operating Temperature	-20C to +70C						
Operating Humidity	5 to 95% Condensing						
Shock and Vibration	ETSI300-019-1.4						
SOFTWARE							
visit www.ubnt.com/airos							



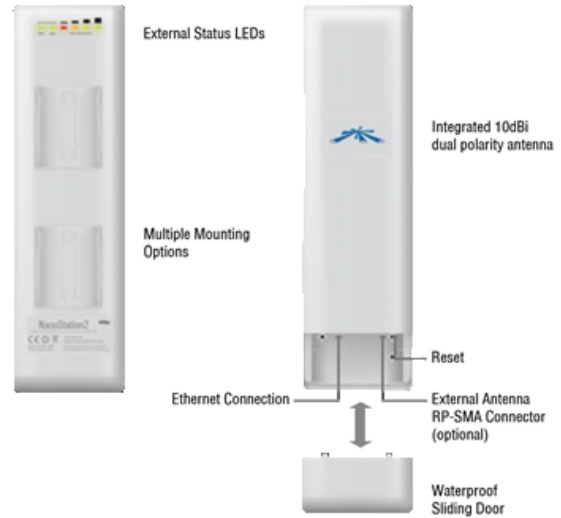
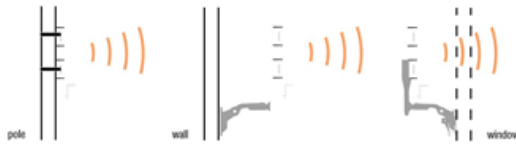


Package Contents



Mounting Options

- 1. Pole Mount (Standard)
- 2. Wall Mount (optional)
- 3. Window Mount (optional)



CPU	Atheros 180MHz MIPS
RAM	16MB RAM
Flash	4MB FLASH
Wireless	2.4GHz, 802.11b/g
Channel width	5/10/20MHz
Antenna Gain	10dBi x2
Polarity	Adaptive Vertical/Horizontal
Ext. Ant. Option	Yes, RP-SMA Connector
Range	15km+ (100km using ext ant.)
Throughput	25Mbps+ TCP/IP
Mounting	Pole Mount (straps included)
Accessories	Ubiquiti Window/Wall Mount (sold seperately)
Size	26.4cm x 8cm x 3cm
Weight	0.4 kg
Power Supply	12V, 1A POE (included)
Approvals	FCC 15.247, IC, CE



Air OS is an intuitive, versatile, highly developed Ubiquiti firmware technology that is included with NanoStation.

It is exceptionally intuitive and was designed to require no training to operate. Behind the user interface is a powerful firmware architecture which enables hi-performance outdoor multipoint networking.



NanoStation utilizes Adaptive Antenna Polarity technology, which can statically or dynamically software switch antenna polarities to optimize your connections.



A.9. Datasheet Mesh Potato [33]

Feature	Benefit
Integrated Wifi and ATA in a single box	Low power consumption, ease of setup, fewer cables, fewer points of failure, and low cost.
Mesh, Client or AP mode Wifi	Can be a component in Mesh Wifi networks or connect to existing Wifi networks
One-IP configuration	Set one IP and the device is ready to make an receive phone calls. Quickly build voice and IP networks and roll out a Wireless PBX in minutes.
Web GUI or Phone UI	Configure via a web interface or telephone IVR menu. With Phone UI no laptop is required for on-site installation.
UV-resistant, weather-proof enclosure	Long life outdoors, no need to purchase a separate enclosure for out door use.
Built in mounting points	Simple and low cost outdoor installation.
Rugged design	Withstands abuse that would destroy other products such as power surges, brownouts, reverse DC, over-voltage DC, and static electricity.
Wide range DC power supply	Runs on any DC voltage from 10V to 40V, or any AC voltage from 110 to 250VAC via wall-plug type power supply. A solar panel can be directly connected – saving money on a solar regulator.
Power efficient	Consumes just 2.5W (DC). Can run on a 10W solar panel.
Power over Ethernet (PoE) and Power over Telephone Line (PoTL)	Just one cable run for both power and telephone. PoTL injector included.
Open Hardware Design	No vendor lock-in. Open to improvement by anyone.
Open Source firmware	Linux, OpenWRT, B.A.T.M.A.N., and Asterisk. Stable, reliable community developed software. Open to improvement, adaptation, and innovation.
Asterisk	Extremely configurable, add IVR menus, connect IP Phones and billing systems

Overview

- Atheros AR2317 system on a Chip (SoC)
- Silicon labs FXS port chipset
- MIPS 4k processor 180 MHz
- One 10/100Mbit LAN port
- 8 MByte Serial Flash EEPROM
- 16 MByte RAM

Wireless LAN

- IEEE 802.11b/g
- Frequency Band : 2.4 to 2.462GHz
- Antenna Type : Internal Omnidirectional PCB Antenna
- Transmit EIRP power: 1-24 Mbit 20dBm or 36-54 Mbit 17dBm

Interfaces/Ports

- LAN Port : 1 x RJ-45
- FXS Port : 1 x RJ-11

Firmware

- Linux kernel 2.26.3
- OpenWRT Kamikaze (customised version)
- B.A.T.M.A.N. mesh routing daemon Version 0.3
- Asterisk 1.4.11

Environmental

- Operating Humidity: 5 to 95% Condensing
- Operating Temperature: -20° to +70°

Electrical

- Power Type : wide range, protected DC port
- Power Options: AC adaptor (supplied) PoE or PoTL (Power over Telephone Line)
- DC Input Voltage : 9-40 VDC
- DC Input Power : 2.5W DC nominal
- Protected Ports: DC, RJ11 phone , RJ45 Ethernet
- Protection: overvoltage, reverse DC, nearby lightning, static electricity

Physical

- Casing: UV-protected, weatherproof outdoor enclosure
- LED Status Indicators : Power – LAN – etc
- Case Colour : Beige
- Width : 106mm
- Depth : 55mm
- Height : 228mm
- Weight : 0.3 Kg

BIBLIOGRAFÍA

- [1] Fernando Alberto y Cortez Franco , Daniel Antonio y López Hernández , Douglas Alberto Arévalo Navas, "Diseño e implementación de un medidor trifásico multifunción utilizando el IC ADE7754," Universidad de El Salvador, 2005.
- [2] David Arturo y de la Cruz Chinchilla, Ronald Edwin Campos Pérez, "Sistema de lectura de medidores de energía eléctrica basado en Protocolos de internet," Universidad de El Salvador, 2006.
- [3] Gustavo y Villatoro Ríos, Wilman Roberto Gómez Pérez, "Diseño y construcción de un módulo electrónico programable vía ethernet desde la web para monitorear controlar y administrar energía en aplicaciones comerciales," Universidad de El Salvador, 2008.
- [4] National Instrument. Labview. [Online]. <http://www.ni.com/labview/esa/>
- [5] National Instrument. Lookout. [Online]. <http://sine.ni.com/nips/cds/view/p/lang/es/nid/12511>
- [6] General Electric. Proficy HMI/SCADA. [Online]. <http://www.ge-ip.com/Proficy-HMI-SCADA>
- [7] SIEMENS. SIMATIC WinCC SCADA. [Online]. <http://www.automation.siemens.com/mcms/human-machine-interface/en/visualization-software/scada/pages/default.aspx>
- [8] Ricardo Salvador Chacón Morales, "Simulación SCADA (Control, supervisión y adquisición de datos) de una planta generadora de energía eléctrica a base de energía geotérmica.," Universidad de El Salvador, 2012.
- [9] National Instrument Lookout. Precio licencia Lookout. [Online]. <http://sine.ni.com/nips/cds/view/p/lang/es/nid/12515>
- [10] Roberto Carlos y Ramos Osorio, Carlos Ernesto García Rivas, "Construcción de dos sistemas solares fotovoltaicos para alimentar sistemas de telecomunicación rurales," Universidad de El Salvador, 2012.
- [11] Jonathan Alberto Zaldaña, "Medidor inalámbrico de consumo de energía eléctrica de bajo costo," Universidad de El Salvador, 2011.
- [12] Mario Ernesto López Lara, "Aplicación de comunicación móvil mediante teléfonos celulares a través de Wifi," Universidad de El Salvador, 2011.
- [13] Village Telco. [Online]. <http://villagetelco.org/>
- [14] Rony Stalyn Sánchez Morales, "Servicio de respuesta de voz iterativa para el Hospital Nacional Rosales," Universidad de El Salvador, 2012.
- [15] Villagetelco. Mesh Potato, Small Enterprise/Campus Network User Guide. [Online]. http://download.villagetelco.org/archive/secn/SECN-1/SECN_UserGuideV1d9a.pdf
- [16] Ubiquiti Networks, Inc. NanoStation 2. [Online]. <http://www.ubnt.com/nanostation>
- [17] OpenWrt Wireless Freedom. OpenWrt. [Online]. <https://openwrt.org/>
- [18] Open-Mesh. B.A.T.M.A.N. [Online]. <http://www.open-mesh.org/projects/open-mesh/wiki>
- [19] Open source framework for building communications applications. Asterisk. [Online]. www.asterisk.org
- [20] Electroindustries. Communicator EXT. [Online]. <http://www.electroind.com/em-communicator.html>
- [21] Luis Alonso y Tobías Vides , Román Abad Colocho Susaña, "Telefonía Inalámbrica y red de acceso a Internet para los Municipios de Salcoatitán, Juayúa y Ataco," Universidad de El

Salvador, 2011.

- [22] Electroindustries Shark 100. Manual de Instalación y Operación. [Online]. <http://www.electroind.com/pdf/sp/SP-100T-Man.pdf>
- [23] Electroindustries Shark 100S. Manual De Instalación y Operación. [Online]. <http://www.electroind.com/pdf/sp/SP-100S-man.pdf>
- [24] Electroindustries Shark 200S. Manual de Instalación y operación. [Online]. http://www.electroind.com/pdf/sp/01_29_13/ES149721_Shark_200S_man_sp.pdf
- [25] Electroindustries Shark 200. Manual de Instalación y Operación. [Online]. http://www.electroind.com/pdf/sp/ES149701_SP_Shark200_manual.pdf
- [26] IP Aliasing. IP Aliasing en un minuto. [Online]. <http://jqm.io/files/ipaliasing.pdf>
- [27] MODBUS, Especificación. MODBUS Application Protocol Specification V1.1b3. [Online]. http://modbus.org/docs/Modbus_Application_Protocol_V1_1b3.pdf
- [28] Libmodbus, A Modbus library for Linux, Mac OS X, FreeBSD, QNX and Win32. Libmodbus - a fast and portable Modbus library. [Online]. <http://libmodbus.org/>
- [29] MySQL. MySQL: The world's most popular open source database. [Online]. <http://www.mysql.com/>
- [30] PHP. PHP: Hypertext Preprocessor. [Online]. <http://php.net/>
- [31] SPUD. SPUD – Simple Unified Dashboard for mesh networks. [Online]. <http://villagetelco.org/2011/06/spud-simple-unified-dashboard-for-mesh-networks/>
- [32] TCP/IP, MODBUS. MODBUS Messaging on TCP/IP Implementation Guide V1.0b. [Online]. http://modbus.org/docs/Modbus_Messaging_Implementation_Guide_V1_0b.pdf
- [33] Village Telco. Mesh Potato. [Online]. <http://villagetelco.org/mesh-potato/>
- [34] Ubiquiti Networks, Inc. Bullet 2. [Online]. <http://www.ubnt.com/bullet>