



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

**DESARROLLO DE SOFTWARE PARA
DISPOSITIVO DE REGISTRO Y
COMUNICACIÓN ETHERNET PARA
MÓDULOS DE MEDICIÓN DE CONSUMO
ELÉCTRICO Y AGUA**

TESIS

Que para obtener el título de
Ingeniero en Computación

P R E S E N T A

Luis Alejandro De la Cruz Velázquez

DIRECTOR DE TESIS

M. I. Miguel Angel Mendoza García



Ciudad Universitaria, Cd. Mx., 2016



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Dedicatoria

A mis padres Macedonia Velázquez y Roque de la Cruz por su apoyo incondicional a lo largo de mi vida.

A mis hermanos Liliana, Gustavo, Alberto y Eduardo por su cariño y en algunos casos por soportar las incomodidades de esas noches de estudio y proyectos.

A mi amiga Alejandra Contreras por siempre creer en mí en momentos complicados.

A mi amigo Hugo Merino por acompañarme a lo largo ya de muchos años.

Agradecimientos

A las diversas luchas sociales que me han permitido gozar de la educación pública y gratuita.

A todos mis profesores que apoyaron en mi formación.

A mi amada Universidad Nacional Autónoma de México por formarme, para ser un profesional que apoye el desarrollo del país.

A Jorge Blanco por su amistad y apoyo para el desarrollo de esta tesis.

A los Ingenieros Miguel Ángel Mendoza y Enrique Gómez por: la oportunidad de participar en tan importantes proyectos, las facilidades brindadas, la confianza depositada y la orientación para el desarrollo de esta tesis.

ÍNDICE

ÍNDICE DE FIGURAS	III
ÍNDICE DE TABLAS	VII
RESUMEN	IX
INTRODUCCIÓN	1
HIPÓTESIS	5
1 ANTECEDENTES	7
1.1 MICROCONTROLADORES	7
1.1.1 <i>Historia de los microcontroladores</i>	8
1.1.2 <i>Diferencia entre microcontrolador y microprocesador</i>	10
1.1.3 <i>Tipos de arquitecturas</i>	10
1.1.4 <i>Elementos de un microcontrolador</i>	12
1.2 REDES DE COMPUTADORAS	16
1.2.1 <i>Historia de las redes de computadoras</i>	17
1.2.2 <i>Modelo OSI</i>	19
1.2.3 <i>Modelo TCP/IP</i>	21
1.2.4 <i>Comparación del modelo OSI y TCP/IP</i>	22
1.3 DISPOSITIVO MEDIDOR DE CONSUMO EA	23
1.3.1 <i>Características</i>	25
1.3.2 <i>Diagramas</i>	25
1.3.3 <i>Funcionamiento general</i>	27
1.3.4 <i>Comparación del Hardware con otros productos en el mercado</i>	30
1.4 LAS LIBRERÍAS TCP/IP DE MICROCHIP	31
2 FIRMWARE	33
2.1 INICIALIZACIÓN	34
2.1.1 <i>Variables</i>	35
2.1.2 <i>Hardware</i>	36
2.1.3 <i>Servicios</i>	36
2.2 REGISTRO DE MUESTRAS	36
2.3 PRESENTACIÓN DE INFORMACIÓN	39
2.3.1 <i>Envío de información a página web</i>	39
2.3.2 <i>Guardar información dentro de archivo de texto en memoria microSD</i>	42
2.4 RECEPCIÓN DE DATOS DE USUARIO	44
3 INTERFAZ WEB	47

3.1	PÁGINA DE INICIO	49
3.1.1	<i>Letrero de identificación del dispositivo</i>	50
3.1.2	<i>Letrero de acumulados y estado del dispositivo</i>	50
3.1.3	<i>Gráficas</i>	51
3.1.4	<i>Link de descarga de datos del día</i>	57
3.1.5	<i>Botones de navegación</i>	57
3.2	PÁGINA DE CONFIGURACIÓN	57
3.2.1	<i>Listas desplegables</i>	59
3.2.2	<i>Datos con validación</i>	60
3.3	PÁGINA DE CONFIGURACIÓN CORRECTA	62
3.4	PÁGINA DE CONFIGURACIÓN DE RED	62
3.5	PÁGINA DE REINICIO DEL DISPOSITIVO	64
3.6	ARCHIVO DE TRANSFERENCIA	65
4	RESULTADOS	67
4.1	REGISTRO DE MUESTRAS Y DESPLIEGUE DE RESULTADOS	69
4.2	VELOCIDADES DE CAPTURA	72
4.3	INTERFAZ WEB ADAPTABLE	77
4.4	TRÁFICO EN LA RED	80
4.5	INTERFAZ WEB SIN CONEXIÓN	82
4.6	RESCATE DE INFORMACIÓN DE ARCHIVO DE TRASFERENCIA	83
4.7	TRASFERENCIA DE ARCHIVO VÍA FTP	84
4.8	COMPARACIÓN DEL SISTEMA COMPLETO CON OTROS EN EL MERCADO	85
	CONCLUSIÓN	87
	TÉCNICA	87
	COMERCIAL	88
	APÉNDICES	91
	A) HOJA DE DATOS FAMILIA PIC32MX5XX/6XX/7XX	91
	B) HOJA DE DATOS DE CUANTIFICADOR DE CONSUMO ELÉCTRICO ADE7757	93
	C) CONCEPTOS BÁSICOS DE REDES	94
	GLOSARIO	95
	REFERENCIAS	97

ÍNDICE DE FIGURAS

Figura 1.1 Arquitectura Von Neumann	11
Figura 1.2 Arquitectura Harvard	12
Figura 1.3 Capas del modelo OSI	19
Figura 1.4 Capas del modelo TCP/IP	21
Figura 1.5 Módulo de desarrollo Explorer 16 y tarjeta de comunicación Ethernet. 23	
Figura 1.6 Prototipo de medidor de consumo eléctrico	23
Figura 1.7 Núcleo de Comunicación Ethernet.....	24
Figura 1.8 Tarjeta de Medición de Consumo EA y Núcleo de Comunicación Ethernet.....	24
Figura 1.9 Diagrama de Núcleo de Comunicación Ethernet	26
Figura 1.10 Diagrama para la medición de Consumo EA	26
Figura 1.11 Borne de conexión de una línea.....	27
Figura 1.12 Algunos convertidores de corriente	27
Figura 1.13 Conector para medidores de caudal	28
Figura 1.14 Medidor de caudal eléctrico de molino	28
Figura 1.15 Pila de protocolos incluidos en el MLA.....	32
Figura 2.1 Directorio de archivos de código fuente de la aplicación demostración	33
Figura 2.2 Funcionamiento general del Firmware	34
Figura 2.3 Proceso de inicialización del Medidor de Consumo EA	34
Figura 2.4 Estructura de una muestra	35
Figura 2.6 Programa MPFS2 para generar empaquetado de páginas web	40
Figura 2.5 Muestra de código HTML con enlace de datos	40
Figura 2.7 Árbol de directorios que se creará en la memoria microSD	43
Figura 3.1 Página de inicio de aplicación de demostración.....	48
Figura 3.2 Carpeta principal de la página web del Medidor de Consumo EA	48
Figura 3.3 Contenido de la carpeta protegido	48
Figura 3.4 Página de inicio del Medidor de Consumo EA	49
Figura 3.5 Gráfica multi-series, ejemplo del uso del API D3.js.....	52
Figura 3.6 Petición de usuario y contraseña para acceder a la página de configuración	58
Figura 3.7 Página de configuración del Medidor de Consumo EA	58
Figura 3.8 Página de configuración correcta	62
Figura 3.9 Página de configuración de red.....	62
Figura 3.10 Página de reinicio del dispositivo	64
Figura 3.11 Página de archivo de transferencia, vista desde Internet Explorer	65
Figura 3.12 Interacción entre Medidor de Consumo EA y cliente.....	66
Figura 3.13 Diagrama de tiempos de consulta de archivo de transferencia.....	66

Figura 4.1 Medidor de Consumo EA instalado en el edificio 5 del Instituto de Ingeniería de la UNAM	68
Figura 4.2 Medidor de Consumo EA instalado en el primer piso de edificio 12 del Instituto de Ingeniería de la UNAM.....	68
Figura 4.3 Página de inicio del Medidor de Consumo EA instalado en el edificio 12 del Ingen UNAM.....	69
Figura 4.4 Núcleo de Comunicación Ethernet conectado a tarjeta de desarrollo “Texas Instruments Tiva Launchpad TM4C123GXL”	72
Figura 4.5 Señal con frecuencia de 10[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet.....	73
Figura 4.6 Señal con frecuencia de 20[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet.....	73
Figura 4.7 Señal con frecuencia de 50[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet.....	73
Figura 4.8 Señal con frecuencia de 100[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet.....	73
Figura 4.9 Señal con frecuencia de 500[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet.....	74
Figura 4.10 Página de inicio del Medidor de Consumo EA con resultados de prueba de captura de una línea.....	74
Figura 4.11 Página de inicio del Medidor de Consumo EA con resultado de prueba de captura de línea y consola de comandos con resultado de transferencia de archivo vía FTP	75
Figura 4.12 Página de inicio del Medidor de Consumo EA con resultados de pruebas de captura de todas las líneas y flujos activos	75
Figura 4.13 Página de inicio del Medidor de Consumo EA con resultados de prueba de captura de todas las líneas y flujos activos, además de consola de comandos con resultado de transferencia de archivo vía FTP.....	76
Figura 4.14 Página de inicio de Medidor de Consumo EA visualizado desde “Mozilla Firefox” (Parte superior al fondo) y “Microsoft Edge” (Parte inferior y al frente).....	78
Figura 4.15 Página de inicio del Medidor de Consumo EA vista en navegador tamaño maximizado (rojo), minimizado (azul) y anchura mínima (verde)	78
Figura 4.16 Gráfica de la página principal del Medidor de Consumo EA que se adapta al navegador. A la izquierda el navegador minimizado y a la derecha con anchura mínima.....	79
Figura 4.17 Página Web de Medidor de Consumo EA vista desde: “Samsung Galaxy S3 mini” con SO “Android 5.1 Lollipop” (izquierda), “Microsoft Lumia 640 LTE” con SO “Windows 10 Mobile” (centro) y “ASUS MEMO Pad HD 7” con SO “Android 4.2 Jelly Bean” (derecha).....	79

Figura 4.18 Tráfico de red generado al consultar la página de inicio del Medidor de Consumo EA 80

Figura 4.19 Tráfico generado en un intervalo de 22 minutos visualizando la página del Medidor de Consumo EA..... 81

Figura 4.20 Página de inicio del Medidor de Consumo EA, sin conexión a Internet 82

Figura 4.21 Página de inicio de Cliente/Servidor que interroga a los Medidores de Consumo EA instalados en el Instituto de Ingeniería de la UNAM..... 83

Figura 4.22 Página Web del cliente/servidor que muestra la información generada, a partir de la recolectada en el archivo de transferencia del Medidor de Consumo EA del edificio 5 del Instituto de Ingeniería de la UNAM 84

Figura 4.23 Trasferencia de archivo vía FTP desde el Medidor de Consumo EA . 85

ÍNDICE DE TABLAS

Tabla 1 Niveles de voltaje lógicos	14
Tabla 2 Especificaciones del Medidor de Consumo EA	25
Tabla 3 Distribución de elementos en microcontrolador	29
Tabla 4 Costos de sistemas	31
Tabla 5 Datos utilizados para generar gráfica en página de inicio del Medidor de Consumo EA	70
Tabla 6 Resultados de prueba de captura de una fase	74
Tabla 7 Resultados de prueba de captura de una fase con transferencia de archivo vía FTP	75
Tabla 8 Resultado de prueba de captura de todas la líneas y flujos activos	76
Tabla 9 Resultado de prueba de captura con todas las líneas y flujos activos, con transferencia de archivo vía FTP	77
Tabla 10 Comparación de costo de dispositivos completos disponibles en el mercado para cubrir 3 líneas	86
Tabla 11 Comparación de funciones	86

RESUMEN

Se muestra el desarrollo del Software (Firmware y página Web) para un dispositivo que tiene la capacidad de realizar mediciones de consumo eléctrico y agua, además de poderse comunicar vía Ethernet y guardar información en memoria microSD. El dispositivo se basa en un microcontrolador PIC32MX de la empresa Microchip. Para el desarrollo se utilizó Software que facilita la misma empresa (IDE MPLAB X, compilador XC32 y Librerías TCP/IP).

Partiendo de un proyecto de demostración incluido en las librerías TCP/IP de Microchip y haciendo un desarrollo evolutivo, se realizaron pruebas para identificar los módulos que tenían que ser adecuados, así como los nuevos que se desarrollarían para cumplir con las funcionalidades del dispositivo, las cuales son: Adquirir datos de consumo eléctrico y de agua, guardarlos en memoria microSD y fungir como servidor web para el despliegue de los datos adquiridos y opciones de configuración.

El desarrollo del software a partir de las librerías TCP/IP de Microchip y el proyecto de demostración incluido, permitió desarrollar las funcionalidades requeridas en el microcontrolador, haciendo posible el monitoreo y registro del consumo eléctrico y de agua. El resultado positivo de este desarrollo da la pauta a seguir utilizando microcontroladores PIC32MX junto con las librerías TCP/IP de Microchip para este tipo de sistemas de monitoreo remoto vía Ethernet.

INTRODUCCIÓN

El uso de la tecnología es uno de los rasgos distintivos que le permitió a nuestra especie sobrevivir y mantenerse hasta nuestros días. Definida en el diccionario de la Real Academia Española (RAE) como el “Conjunto de teorías y técnicas que permiten el aprovechamiento del conocimiento científico”. Su evolución ha afectado el progreso social y económico, generalmente enfocada en el ámbito comercial y hasta hace pocos años, sin tomar seriamente las consecuencias que generan en contra del medio ambiente.

Existen gran cantidad de ramas tecnológicas, las cuales han evolucionado con distintos ritmos. Una de las que más ha crecido e influido en otras ramas, es la eléctrica. Esta ha cambiado en gran medida la forma en que interaccionamos con el planeta, la energía eléctrica se ha vuelto un pilar fundamental en nuestra civilización. Su capacidad de transformarse en luz, movimiento, calor y señales, ha logrado revolucionar la vida de los humanos.

La producción y transporte de energía eléctrica provoca costos altos, tanto económicos como ecológicos, a mediano y largo plazo. La generación de electricidad se basa mayormente en la quema de combustibles fósiles. Esta actividad libera a la atmósfera grandes cantidades de los llamados gases de efecto invernadero, de entre los cuales se destaca el CO₂. En conjunto con otro tipo de contaminaciones, como la del agua, se está produciendo un cambio climático acelerado que pone en entredicho la forma en que hemos implementado y usado la tecnología, ya que se ha dado preponderancia a aspectos económicos de corto plazo.

Estamos frente a una urgente necesidad de cambiar los paradigmas del desarrollo tecnológico y económico para enfocarnos a solventar y revertir los daños que hemos causado al planeta. Independientemente que, en la actualidad, se están empezando a adoptar medios alternativos de producción de energía eléctrica menos contaminantes. Es necesario hacer más eficiente el uso de la energía para evitar la contaminación sin sentido. En este ámbito, dentro del Instituto de Ingeniería de la UNAM, se está llevando a cabo el “Plan Estratégico de Sostenibilidad” en el cual se busca adoptar y practicar dentro de dicha institución una cultura hacia la sustentabilidad, donde uno de los puntos más importantes consiste en el “Programa de medición y ahorro de energía – Metabolismo del Instituto” en el cual se busca determinar el consumo de energía eléctrica y de agua de cada edificio del Instituto.

En base a otras ramas de la tecnología se puede lograr este objetivo, las redes de comunicaciones y las computadoras pueden realizar la recolección y análisis de

datos de consumo de forma rápida y de gran cantidad de variables, posibilitando el monitoreo en tiempo real, además de identificación de fallos y picos de consumo, buscando así una mayor eficiencia tanto en el consumo, como en el suministro de energía eléctrica.

La adopción tecnológica de las redes de comunicaciones ha hecho posible que en la actualidad la infraestructura de los edificios provea de nodos de acceso a red, haciendo factible la implementación de sistemas de monitoreo remoto.

Otros avances tecnológicos han abierto la posibilidad de desarrollar computadoras con mayores y diversas prestaciones, a costos razonablemente bajos. Si hablamos de dispositivos pequeños y de bajo consumo energético, los microcontroladores tienen ya la velocidad para el manejo de conexión Ethernet abriendo con ello una amplia gama de aplicaciones.

Teniendo en cuenta las tecnologías mencionadas, se le pidió a la coordinación de electrónica del Instituto de Ingeniería de la UNAM, el desarrollo de los medidores de consumo eléctrico y de agua. Un grupo de trabajo de la coordinación se dedicó a la investigación y desarrollo de un módulo basado en un microcontrolador que permita: la comunicación Ethernet, consultar fecha y hora de un reloj de tiempo real y la capacidad de guardar información tanto en una memoria EEPROM como en una tarjeta microSD. Además del desarrollo de una tarjeta adicional, que tenga la capacidad de medir el consumo eléctrico y de agua.

Al momento de la investigación para el desarrollo del Hardware se encontró que las empresas fabricantes de microcontroladores tales como Atmel, Texas Instruments y Microchip, han buscado ofrecer a los desarrolladores Software y módulos de desarrollo que les ayuden a realizar sus aplicaciones, logrando con ello desarrollos acelerados que deriven en productos finales o prototipos.

Tras la investigación se encontró que Microchip ofrecía tanto el Software como el Hardware con el cual se podía partir para desarrollar dispositivos con las características mencionadas.

Por parte del hardware se utilizó una tarjeta de desarrollo, con un microcontrolador de 32 bits y un módulo de comunicación Ethernet. Con el fin de probar el Software que permite desarrollar aplicaciones que utilicen la comunicación Ethernet.

Basándose en el Hardware anteriormente mencionado, el grupo de trabajo logró desarrollar el medidor de consumo eléctrico y agua, pero sin Software programado.

La presente tesis muestra el desarrollo del Software que le dará la funcionalidad al Hardware del medidor de consumo. Esto se logrará mediante la adaptación del proyecto de demostración incluida en las librerías TCP/IP de Microchip donde algunas funciones serán modificadas o añadidas. Con el objetivo de desarrollar el

Software, que permita adquirir y almacenar los datos de consumo eléctrico y agua, además de fungir como servidor web para el despliegue de datos y opciones de configuración.

HIPÓTESIS

Es posible desarrollar el Software (Firmware y Pagina Web) para que un microcontrolador con capacidades de comunicación Ethernet, realice tareas de adquisidor de datos, para una tarjeta de medición de consumo eléctrico y de agua utilizando sus puertos digitales, además de fungir como servidor web, para el despliegue de las opciones de configuración y visualización de los datos adquiridos.

1 ANTECEDENTES

Se presentan los conceptos para contextualizar y entender el contenido de la presente tesis, enfocándose en tres tópicos, los microcontroladores por ser la base del sistema, las redes de computadoras que son una capacidad esencial de funcionamiento y las características de Hardware del Medidor de Consumo Eléctrico y Agua (EA) para tener un panorama de las funcionalidades que el Software debe tener.

1.1 MICROCONTROLADORES

Son microcomputadoras en un solo circuito integrado, las cuales tienen la característica de poder ser programadas con un fin específico. El programa en su interior se vale de las capacidades de Hardware del microcontrolador, para interactuar con sensores, actuadores y otros dispositivos.

Su aparición en el mercado revolucionó la industria de la electrónica, pues sustituyó el clásico diseño digital, el cual se valía de compuertas lógicas y el uso de memorias para implementar funciones, logrando el mismo objetivo con un menor costo, tamaño y consumo energético.

Por sus características han pasado a ser piezas fundamentales en gran cantidad de industrias, que van desde la automovilística, aérea, espacial, producción de alimentos, electrodomésticos, entretenimiento, salud, etc. Esto se debe a que la mayoría de dispositivos electrónicos poseen uno o varios microcontroladores que les dan buena parte de su funcionalidad.

Los modelos más actuales tienen la capacidad de manejar interfaces de comunicación USB y Ethernet, que abre un nuevo panorama para el desarrollo de sistemas más integrales e interconectados.

1.1.1 Historia de los microcontroladores

Para el desarrollo de los primeros microcontroladores se tienen antecedentes de suma importancia, ya que de ellos dependen la gran mayoría de dispositivos electrónicos de hoy en día.

En primer lugar, tenemos al transistor desarrollado en los Laboratorios Bell en Estados Unidos en el año 1947 por John Barden, Walter Houser Brattain y William Bradford Shockley que sustituyó a los tubos de vacío, los cuales eran de gran tamaño y exigían grandes cantidades de energía para su funcionamiento, recordando que estos eran utilizados en las primeras computadoras con la finalidad de permitir o negar el paso de corriente (conmutador de estados). Los transistores de menor tamaño y consumo, posibilitaron el inicio de la miniaturización de los dispositivos eléctricos.

Posteriormente tenemos el desarrollo de los circuitos integrados, donde los primeros intentos se desarrollaron con poco éxito, hasta que en 1959 el ingeniero Jack S. Kilby, que trabajaba para Texas Instruments, desarrolló un dispositivo con base de germanio que integraba seis transistores en una misma base semiconductor. Se sucedieron más avances con la finalidad de integrar cantidades mayores de transistores en pocos milímetros cuadrados. Los circuitos integrados tienen ventajas muy marcadas a comparación de aquellos circuitos desarrollados con componentes discretos: su menor costo, menor tamaño y mayor eficiencia energética.

Teniendo esta base, podemos hablar ahora de los primeros microprocesadores que, como tal, son circuitos integrados que contienen una buena cantidad de transistores y otros elementos.

Para el año 1969 un grupo de ingenieros japoneses de la empresa Busicom que se dedicaba a la construcción de calculadoras con circuitería discreta, llegaron a los Estados Unidos y se pusieron en contacto con Intel con la finalidad de desarrollar nuevos circuitos para sus calculadoras que contuviesen una menor cantidad de elementos discretos. Intel propuso el utilizar un circuito integrado,

cuyo funcionamiento estaba determinado por el programa almacenado en su interior. La idea fue desarrollada hasta obtener un producto ya terminado, con lo cual nació el primer microprocesador. En 1971 Intel le compró la licencia a Busicom, que ignoraba el gran potencial del desarrollo. Ya en este mismo año apareció en el mercado el primer microprocesador, que Intel llamó 4004, era de 4 bits, con 45 instrucciones, direccionaba solamente 4k localidades de 4 bits y con una velocidad de reloj de 740kHz.

Ante las limitaciones que presentaba el 4004, Intel rápidamente un año después (1972) desarrolló un nuevo microprocesador llamado 8008, con las capacidades de manejar bytes completos (8 bits), con un direccionamiento de memoria de 16k bytes, 45 instrucciones y una velocidad de reloj de 800kHz.

Intel siguió con sus desarrollos y en 1973 lanza el 8080 el primer microprocesador con tecnología NMOS, con capacidades de direccionamiento de memoria de 64k bytes, 75 instrucciones y al igual que el 8008 ser de 8 bits, pero teniendo una velocidad de 2MHz.

Muchas empresas empezaron a ver en el desarrollo de microprocesadores un gran nicho de negocio, entre ellas National Semiconductors, Fairchild, Rockwell International y Motorola. Esta última lanzó el MC6800 que era muy similar en capacidades al 8080 de Intel y poseía un precio muy similar, \$179 (USD). El grupo encargado del desarrollo del Motorola MC6800 dejó la empresa y se unieron a MOS Technology y en 1975 desarrollaron el MOS6501 y MOS6502, con capacidades muy superiores al Intel 8080 y al Motorola MC6800. Y a un precio de escándalo de tan solo \$25 (USD), con lo que tanto Intel como Motorola se vieron obligados a bajar el precio de sus productos a \$70 (USD). El MOS6501 se deja de producir por problemas legales entre Motorola y MOS Technology, pero el MOS6502 se siguió produciendo y adoptando para una amplia gama de productos, entre los que figuran: Apple I, Apple II, Atari 800XL, Atari 2600, Nintendo Entertainment System (NES), etc.

La empresa Zilog en 1976 presenta su Z80, teniendo en consideración que el Intel 8080 ya tiene una gran cantidad de desarrollos, hacen al Z80 compatible, pero le añaden funcionalidades que lo hacen superior. Logrando que muchos cambiaran del Intel 8080 al Z80. Intel intenta retomar mercado con su 8085 pero ya era demasiado tarde, el Zilog Z80, el MOS6502 y el Motorola MC6800 dominaron el mercado de los microprocesadores de 8 bits.

Ante la necesidad de tener una microcomputadora en un solo circuito integrado, salen al mercado el Intel 8048 y el Motorola 6805R2 los cuales son los primeros microcontroladores. Esto igual en 1976, con lo que a partir de este año se inicia una separación muy evidente entre microcontroladores y microprocesadores.

Ambos dispositivos siguen evolucionando tecnológicamente, especializándose en tareas distintas.

1.1.2 Diferencia entre microcontrolador y microprocesador

La principal diferencia radica que un microprocesador necesita elementos adicionales para su funcionamiento, tales como memoria e interfaces para recibir y enviar datos. El microprocesador es la unidad central de una computadora, en comparación con el microcontrolador que no necesita que se le agregue ningún elemento para su funcionamiento ya que es una computadora completa en un solo circuito integrado.

Otra diferencia importante son los fines con los que se han desarrollado, por un lado, los microprocesadores se han orientado al mercado de los ordenadores personales (PC) y las estaciones de trabajo, donde es necesario un manejo de grandes cantidades de memoria, elevada capacidad de cálculo y gran velocidad de procesamiento. Por otra parte, los microcontroladores se enfocan en sistemas que realizan un pequeño número de tareas al menor costo posible. El programa que ejecutan se encuentra almacenado en su interior e interactúa con sus puertos de entrada y salida con el exterior.

Al final bastaría recalcar el costo de entre uno y otro, por un lado, un microprocesador Intel Celeron G1820 de \$42 (USD) y un microcontrolador Microchip PIC32MX795L512 de \$13 (USD) precios completamente distintos, que afectarán el costo total del sistema desarrollado.

1.1.3 Tipos de arquitecturas

Existen dos arquitecturas principales. Fueron desarrolladas enfocándose en la forma en que los distintos elementos que conforman una computadora se conectan entre sí, proporcionando distintas características.

Tomando en cuenta que existen diversas modificaciones a las arquitecturas, para aumentar el rendimiento de computadoras en tareas muy específicas.

A) Von Neumann

Es la arquitectura tradicional de una computadora, concebida en 1945 por el físico y matemático John Von Neumann. En la cual se plantea el uso de una unidad central de procesamiento (CPU) conectada a una única unidad de memoria donde se guardan las instrucciones del programa y los datos, así como una salida/entrada para los periféricos (Figura 1.1).

Algunas características a resaltar son:

- El bus que comunica a la memoria es de tamaño predeterminado, con lo que si alguna instrucción es de mayor tamaño que el bus se procederá a

hacer varias consultas a memoria, provocando mayor tiempo en la ejecución.

- No se puede cargar una nueva instrucción hasta que termine la transferencia de datos de la instrucción anterior, esto por el único bus.

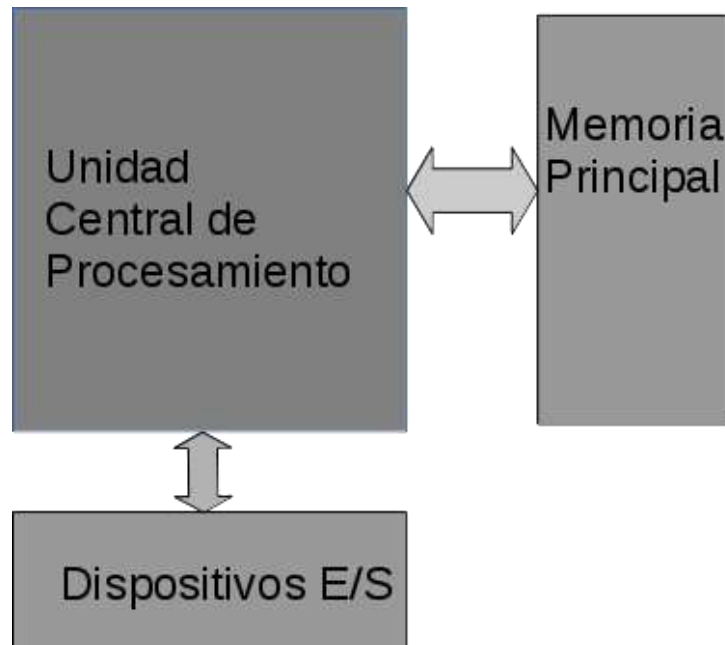


Figura 1.1 Arquitectura Von Neumann

B) Harvard

Toma el nombre de la computadora Harvard Mark I, que fue la primera computadora electromecánica desarrollada por IBM y enviada a la universidad de Harvard en 1944. Su característica principal es tener dos memorias, una dedicada al programa y otra para los datos, con buses independientes para cada una (Figura 1.2).

Algunas características que resaltan son:

- Al tener dos buses independientes se puede acceder a la memoria del programa, al mismo tiempo que se accede a la de datos, provocando un mayor rendimiento.
- Al no compartir un solo bus, en la memoria existe una independencia de la longitud de las instrucciones del programa, con lo que se puede optimizar esta para que las instrucciones solo ocupen una localidad en la memoria, logrando la reducción del tamaño del programa y una mayor velocidad en la ejecución.

En la actualidad esta arquitectura es la más utilizada, tanto en microprocesadores como en microcontroladores teniendo en cuenta que generalmente es modificada un poco, pero preservan la esencia.

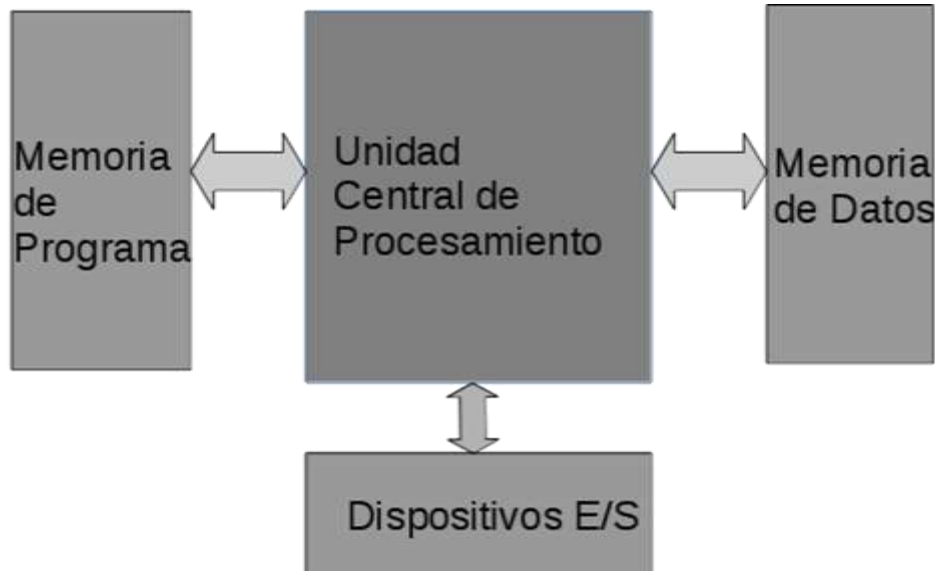


Figura 1.2 Arquitectura Harvard

1.1.4 Elementos de un microcontrolador

Un microcontrolador como se ha visto anteriormente es una computadora, dentro de un solo circuito integrado. Se enunciarán las partes internas que lo conforman ya que son elementos que al programar se deben conocer, para explotar los diversos recursos.

A) Microprocesador

De sus capacidades derivan las características de Software y Hardware del microcontrolador. Se encarga de interpretar las instrucciones del programa, direccionar la memoria de programa y ejecutar las operaciones que implican las instrucciones.

Existen tres enfoques de funcionalidades y arquitecturas para los microprocesadores, que están directamente relacionadas a las instrucciones y como estas son interpretadas.

RISC Computadora de conjunto de Instrucciones reducido, las instrucciones generalmente son ejecutadas en un solo ciclo de reloj al ser muy sencillas. La estructura de Hardware que implica es muy inferior a la de un microprocesador CISC, como consecuencia se pueden encontrar en circuitos integrados de menor tamaño. Generalmente es la arquitectura presente en la mayoría de microprocesadores de microcontroladores.

CISC Computadora de conjunto de instrucciones complejo, sus instrucciones son sofisticadas y requieren gran número de ciclos de reloj para su ejecución. Estas instrucciones permiten realizar tareas, que con instrucciones sencillas llevarían gran carga para el programador. Al principio esta arquitectura dominaba, pero se ha visto desplazada por CISC gracias a que se han encontrado mejores técnicas para la fabricación de circuitos integrados y a su mayor eficiencia.

SISC Computadora de juego de instrucciones específico, retoman la filosofía RISC de instrucciones reducidas, pero añaden algunas extras para tareas muy concretas. Se pueden encontrar en microcontroladores dedicados.

B) Memoria RAM

Es una memoria de alta velocidad, utilizada frecuentemente para almacenar temporalmente datos y variables. Tiene la característica de ser volátil, con lo que al apagar el sistema la información contenida se pierde.

A comparación con computadoras de uso general, en la RAM no se guarda una copia del programa que se está ejecutando, ya que este se encuentra almacenado en la memoria de programa.

Se le asignan localidades reservadas que son utilizadas como registros que cumplen como indicadores para tareas muy específicas.

C) Memoria ROM

Es aquella en donde se almacena el programa que se ejecutará, así como algunas variables que solo serán de lectura. Existen varias tecnologías para este tipo de memoria, pero las más utilizadas son las FLASH y las EEPROM.

Son del tipo no volátil, con lo que mantienen la información almacenada a pesar de apagar el sistema.

D) Reloj principal

Es aquel que genera los pulsos para sincronizar todas las operaciones del sistema. Los microcontroladores generalmente traen un circuito oscilador que genera una onda cuadrada o pulsos. Pero también existen aquellos que pueden recibirlo desde una fuente externa.

La frecuencia del reloj afecta directamente el tiempo en que se ejecutan las instrucciones. Se tiene que tener en cuenta que la velocidad tiene una relación directamente proporcional, en caso de aumentarla también se aumenta la temperatura y el consumo energético.

Dependiendo del tipo de microcontrolador será la velocidad máxima de operación.

E) Puertos E/S

Los puertos de Entrada o Salida son el medio por el cual el microcontrolador puede interactuar con el exterior, se encuentran divididos por canales o líneas y son la principal utilidad presente en las terminales del microcontrolador.

Dependiendo de las capacidades del microcontrolador, los canales o líneas de un puerto se podrán establecer como entrada o salida. Al igual que se les podrá asignar diversos tipos de utilidad específica.

F) Recursos adicionales

Existen muchos y variados modelos de microcontroladores. Cada fabricante toma una arquitectura base para sus productos. Y a su arquitectura le añaden diversas capacidades, variando con ello la utilidad y precio del producto.

◆ **Puertos digitales E/S**

Son la capacidad más común en los microcontroladores, consiste en que los canales de los puertos a partir de un nivel de voltaje indicarán 0 o 1. De tal forma si el canal está configurado como entrada, el voltaje debe provenir del exterior y en caso de ser salida, el microcontrolador lo proporcionará. Los niveles de voltaje para ser considerados 1 o 0 dependen de la tecnología de microcontrolador (Tabla 1).

Tabla 1 Niveles de voltaje lógicos

Nivel	TTL	CMOS	HC
Bajo 0	0[V]-0.8[V]	0[V]-1.5[V]	0[V]-1[V]
Alto 1	2[V]-5[V]	3.5[V]-5[V]	3.5[V]-5[V]

◆ **Convertidores Analógico-Digital, Digital-Analógico**

Ambos convertidores son de mucha utilidad, pero generalmente solo el Analógico-Digital se encuentra presente en la mayoría de microcontroladores y el Digital-Analógico solo en aquellos muy especiales, generalmente se agrega de manera separada.

El funcionamiento del analógico-digital consiste en que internamente el microcontrolador tiene una serie de comparadores de voltaje, cuyos niveles y el número de ellos activos se pueden configurar mediante programación. Internamente el microcontrolador interpreta cuantos comparadores se han activado dependiendo del voltaje recibido por algún canal configurado,

codificándolos en un número el cual representa digitalmente el nivel de voltaje percibido.

En el caso del digital-analógico se realiza un proceso similar, donde se tiene un circuito ya sea interno o externo del microcontrolador, el cual dependiendo de un número sumará a una salida una cantidad de voltaje, determinado por la configuración del convertidor.

◆ **Puertos de comunicación**

Existen varios tipos de interfaces, las cuales permiten al microcontrolador comunicarse con el exterior hacia otros dispositivos externos. Integrando al microcontrolador a buses de datos, regidos por algún protocolo o norma de comunicación.

Dependiendo del modelo del microcontrolador se tendrá soporte, a algunas o todas las formas de comunicación. Las más comunes son:

UART, interfaz serie asíncrona. La más común para comunicarse con una PC.

USART, interfaz serie sincronía y asíncrona. Tiene la capacidad de ser configurado como UART al igual que comunicación por solo un hilo.

I²C, interfaz serie de dos hilos, usa un hilo para el reloj y otro para la recepción/transmisión.

SPI, interfaz serie de tres hilos, de gran popularidad y mayor velocidad que I²C por tener un hilo para transmisión y otro para recepción.

CAN, interfaz de comunicación basada en una topología de bus, para entornos distribuidos. Es una buena opción para la comunicación entre múltiples microcontroladores.

USB, interfaz serial muy popular en las PC, pero su programación en microcontroladores es más complicada que todas las opciones anteriores mencionadas.

Ethernet, es la interfaz de comunicación más extendida en el mundo de la comunicación entre computadoras. Solo se encuentra disponible en microcontroladores de gama alta. Y su programación al igual que USB tiene un grado de complejidad elevado.

◆ **Temporizadores**

También conocidos como TIMER, su principal función es medir periodos de tiempo o como contadores de eventos.

En el caso del tiempo se basa en la señal de reloj que hace que el microcontrolador funcione, cada pulso detectado hace que el contador aumente y dependiendo de la configuración y capacidad, se desbordará provocando que una bandera se active, teniendo con esto la señal que un intervalo de tiempo ha pasado.

Ya como contador de eventos, bien este puede venir de una fuente de Software o Hardware. En el caso del Software se puede acceder al contador del TIMER y aumentarlo o disminuirlo con alguna condición que nos interese. Por Hardware se tiene un canal asociado que después de ser configurado, será una entrada de señal, la cual detectará cuando se pase de un estado lógico a otro. Y de igual forma el contador se puede programar para que aumente o disminuya la cuenta cada que se presente el evento de cambio de estado lógico.

◆ **Perro Guardián**

Más conocido por su nombre en inglés “WatchDog”, consiste en un contador que, al llegar a su cuenta máxima, reinicia al microcontrolador.

Es tarea del programador diseñar una rutina en donde, si el programa se ejecuta correctamente impida, que el contador llegue a su cuenta máxima. En el caso de que el programa falle o caiga en un bucle infinito el contador del perro guardián seguirá su marcha hasta provocar el reinicio. Obteniendo con ello una protección ante fallos y evitando que el sistema quede inutilizable pues con el reinicio se busca retomar el control.

◆ **Modulador de ancho de pulso**

O simplemente PWM, se basan en el uso de los TIMERS para proporcionar mediante un canal preestablecido un tren de pulsos a una frecuencia programada como señal de salida. Muy utilizado para variar la velocidad en motores.

◆ **Estado de bajo consumo o reposo**

El microcontrolador entra en un modo de operación donde su consumo es mínimo y solo espera a que ocurra un evento (generalmente un cambio de estado lógico en algún canal de entrada) para poner en funcionamiento su programa principal.

1.2 REDES DE COMPUTADORAS

Es el concepto designado a un conjunto de computadoras autónomas conectadas entre sí, con la función de compartir información. El medio por el cual se lleva tal

conexión puede ser desde un cable, fibra óptica, microondas, rayos infrarrojos, vía satelital, etc.

Han propiciado nuevos tipos de interacción, muchas actividades humanas se han visto afectadas: los negocios, las relaciones sociales, el entretenimiento y el aprendizaje.

Tan importantes son las redes de computadoras que estas proliferan en el mundo a tal grado que existe una red mundial que conecta a muchas redes, la muy afamada Internet.

1.2.1 Historia de las redes de computadoras

En la década de 1950 aparecieron las primeras computadoras, de gran tamaño y alto consumo energético, su uso se restringía a unas cuantas empresas, universidades y para fines militares. Su utilidad propició desarrollos tecnológicos con el motivo de tener mayor acceso a ellas.

A inicios de la década de 1960 aparecieron las computadoras de tiempo compartido. Pues anteriormente tener más de una computadora no era accesible. Este tipo de computadoras proporcionaba el soporte a varias terminales enlazadas a una computadora central. Y con ello albergar a varios usuarios a la vez. El gran impedimento era que entre más terminales conectados más decaía la velocidad de comunicación, a causa de que se utilizaban mecanismos de conmutación de circuitos.

Una mejor forma de comunicación se desarrolló posteriormente, llamada conmutación de paquetes. Mencionada por primera vez por Leonard Kleinrock, un graduado del MIT, donde junto con Lawrence Roberts dirigieron el programa de Ciencias de la Computación en la Agencia de Investigación de Proyectos Avanzados (ARPA por sus siglas en inglés) de Estados Unidos, donde plantearon la creación de la primera red de computadoras de comunicación de paquetes llamada ARPAnet.

Para 1972 la red ARPAnet contaba ya con 15 nodos, ese mismo año se mostró por primera vez al público en la Conferencia Internacional sobre Comunicación de Computadoras. Se completó el primer protocolo de comunicación entre terminales ARPAnet, conocido como protocolo de control de red (NCP).

Como la ARPAnet era una red cerrada, empezaron a surgir otras redes de conmutación de paquetes. Fueron tan diversas que al tiempo llegaría el momento de interconectarlas. El trabajo de interconexión fue patrocinado por la Agencia de Proyectos de Investigación Avanzados de Defensa (sucesora de ARPA y con siglas de su nombre en inglés DARPA) de Estados Unidos.

Se siguieron llevando a cabo pruebas para la transmisión y direccionamiento de información, derivando en las primeras versiones de los tres principales protocolos, que en la actualidad se siguen usando: TCP, IP y UDP.

En 1983 se lanzó oficialmente TCP/IP por parte de ARPAnet para substituir a NCP. De igual forma se empezó a utilizar el sistema de identificación DNS el cual proporciona la capacidad de traducir direcciones numéricas a nombres más comunes. En los años siguientes con la integración de los protocolos, la interacción de redes con estructuras dispares se realizó de manera más fácil.

Ya para la década de 1990 se vio la llegada de proveedores comerciales de Internet, que provocaron que redes como ARPAnet, MILTEL y NFSNET fueran cerradas y su infraestructura cedida. El acontecimiento que marcó esa década fue la aparición de la Web. La web fue desarrollada en CERN en Ginebra por Tim Barnels-Lee, que junto a un equipo de trabajo desarrollaron las primeras versiones de HTML, HTTP, la World Wide Web (WWW) un servidor web y un navegador, que son los elementos base de la Web. Y permite la visualización de contenido mediante hipertexto “para vincular y acceder a contenido de diversos tipos como una red de nodos en los que el usuario puede navegar a voluntad”.

El éxito fue tal que a finales de 1993 ya existían cerca de 200 servidores web y seguían aumentando a un paso vertiginoso. Se desarrolló una de las batallas más memorables en la industria de la computación en el área del Software, la llamada “guerra de navegadores” desarrollada por Netscape y Microsoft en 1996. Donde Microsoft aprovechó la posición dominante en el mercado de su Sistema Operativo Windows, para incluir su navegador Internet Explorer en cada copia. Haciéndose poco a poco de la gran mayoría de la cuota de mercado y llevando a la quiebra a Netscape.

El interconectar computadores, tanto personales como las utilizadas en negocios ya era algo realmente atractivo, la necesidad de consultar y compartir información. Impulso a la industria a desarrollar formas más baratas, fiables y rápidas de conexión. Ya en inicios del nuevo milenio se pudo ver el cambio del uso de modem RTPC (Red Telefónica Pública Conmutada) que con una velocidad de 56kbit/s, por los módems DSL (Línea de Abonado Digital) con una muy superior velocidad de 5Mbits/s.

La adopción de redes inalámbricas que se basan en espectro electromagnético, tales como el infrarrojo, las microondas y radio, hizo posible que los nuevos dispositivos que contienen computadoras pudieran comunicarse entre sí sin la necesidad de cables. Que con el uso de protocolos hacen prácticamente que el comportamiento de las redes inalámbricas sea casi igual de fiables que las alámbricas.

Las redes de computadoras siguen su avance tecnológico, las velocidades de transmisión siguen aumentando, gracias a nuevos tipos de cable utilizados y a la mayor proliferación de la fibra óptica, además de las nuevas formas de modulación de señales y nuevos estándares de comunicación.

Estamos frente a otro avance importante, IoT (*“Internet of Things”*) o mejor conocido como “Internet de las Cosas”, implica que pequeñas computadoras montadas sobre “cosas” estén conectados a red, proporcionando control a distancia, así como consultar datos y la posibilidad de que se gestionen de manera automática o mediante nubes. Las “cosas” pueden ser tan variadas como mesas, refrigeradores, anaqueles, elevadores, etc. Plantea un nuevo reto, pues la cantidad de “cosas” sobrepasa por mucho el número de conexiones actuales, con lo que será necesario cambiar las tecnologías actuales para dar soporte a este gran número de conexiones.

1.2.2 Modelo OSI

La Organización Internacional de Estandarización (ISO, por sus siglas en ingles) en 1970, propuso organizar las redes en siete capas, que nombraron modelo OSI (Interconexión de sistemas abiertos).

Las capas se encuentran organizadas como se observa en la figura 1.3.

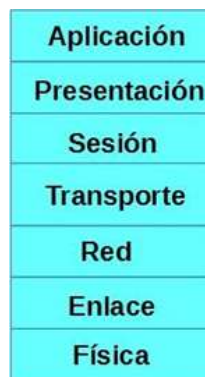


Figura 1.3 Capas del modelo OSI

El modelo se planteó mientras los protocolos que la formarían apenas se estaban desarrollando. A pesar de esto el modelo fue adoptado y tuvo gran impacto en la educación.

Cabe señalar que el modelo OSI solo indica que debe hacer cada capa, no indica los servicios y protocolos que deban contener.

A) Capa de aplicación

Provee a las aplicaciones los protocolos que utilizarán para comunicarse con las capas inferiores. Un usuario no interactúa directamente con esta capa pues

utiliza programas que son los que en verdad interactúan con la capa aplicación. Los programas le resuelven al usuario la complejidad del envío de comandos y la interpretación de datos.

B) Capa de presentación

Proporciona servicios que permiten a las aplicaciones que se comunican, interpretar el significado de los datos intercambiados. Entre los servicios se encuentran la compresión y el cifrado de los datos.

C) Capa de sesión

Permite a los usuarios de máquinas distintas establecer sesiones entre ellas. Estas sesiones incluyen los servicios de, control de dialogo, administración de “token” y sincronización.

D) Capa de transporte

Su función principal es aceptar los datos provenientes de capas superiores, dividirlos si es necesario, colocarlas en la capa de red y comprobar que los datos lleguen correctamente al otro extremo de la conexión. Asegurando un aislamiento de las capas superiores, de las diferentes tecnologías de Hardware de la conexión.

En la capa de transporte se lleva a cabo una conexión punto a punto, a diferencia de las capas inferiores donde los protocolos operan en entre cada máquina y las máquinas vecinas.

E) Capa de red

Responsable de cómo se direccionan los paquetes desde su origen a su destino. Las rutas pueden estar basadas en tablas estáticas o codificadas en red.

F) Capa de enlace

Encargada de transformar un medio de transmisión puro en una línea de comunicación, que al llegar a la capa de red aparezca libre de errores de transmisión. Esto mediante el envío de tramas de datos, donde si el servicio es confiable el receptor enviará una trama de conformación de recepción.

G) Capa física

En ella se lleva la transferencia de bits puros a través del medio físico. Se definen las interfaces mecánicas, eléctrica y de temporización de la red.

Los medios de transmisión pueden ser de tres tipos: dirigidos (cable de cobre y fibra óptica), inalámbricos (radio terrestre) y vía satélite.

1.2.3 Modelo TCP/IP

Es el modelo que se utilizó en la red ARPAnet y toma el nombre de sus dos protocolos primarios. Se planteó por primera vez en 1974 y sigue vigente.

También se le puede encontrar en la literatura como modelo de Internet. Y describe los protocolos para permitir que un equipo se comunique en una red.

Sus capas tienen la organización, tal como se muestra en la figura 1.4.

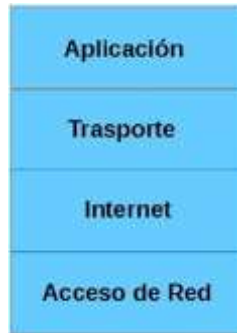


Figura 1.4 Capas del modelo TCP/IP

A) Capa aplicación

En ella residen los protocolos que interactúan con capas inferiores, son aquellos con los que los programas del usuario funcionan. Tales como la terminal virtual (TELNET), transferencia de archivos (FTP), correo electrónico (SMTP), resolución de nombres de dominio (DNS), configuración dinámica de host (DHCP) y transferencia de hipertexto (HTTP). En la actualidad existen gran número de protocolos que residen en esta capa.

Los protocolos se encuentran distribuido en los sistemas terminales, con la finalidad de que los programas que utilicen algún protocolo lo tengan disponible para intercambiar paquetes con el programa de otro sistema terminal.

B) Capa transporte

Provee la capacidad de transportar los mensajes de la capa aplicación a los puntos terminales. Existen dos protocolos TCP y UDP.

TCP proporciona a sus aplicaciones un servicio orientado a conexión, una transferencia garantizada de la información, control de flujo y control de congestión.

UDP proporciona a sus aplicaciones un servicio no orientado a conexión, no se ofrece ninguna fiabilidad, sin control de flujo, ni congestión.

C) Capa Internet

Se encarga de enviar paquetes desde cualquier parte de la red y entregarlos al destino independientemente de la ruta o red que exista a su paso. Es la responsable de direccionar, empaquetar y direccionar los datos para que sean transmitidos y lleguen a su destino.

El protocolo que domina esta capa es IP (Protocolo de Internet).

D) Capa de acceso a red

La encargada de enviar los bits dentro de la trama de un nodo al siguiente. Los protocolos de esta capa son ARP (protocolo de resolución de direcciones) y RARP (Protocolo de resolución de direcciones inversa). Ambos protocolos se basan en el identificador único que tiene cualquier host; la dirección MAC (Control de Acceso al Medio).

1.2.4 Comparación del modelo OSI y TCP/IP

El modelo OSI contiene tres conceptos básicos:

Los servicios que indican que hace cada capa.

La interfaz de una capa indica a los procesos que están sobre ella y como accederla

Cada capa puede usar cualesquiera protocolos mientras consiga que haga el trabajo.

El modelo TCP/IP no distinguía entre protocolos, servicio y modelo. Aunque las personas han tratado de adaptarlo para hacerlo más parecido a OSI.

Como consecuencia en OSI los protocolos están mejor ocultos que los de TCP/IP y pueden ser reemplazados fácilmente conforme cambia la tecnología. Y entonces cumple con uno de los objetivos primordiales de tener protocolos por capas.

Por otra parte, OSI fue concebido antes que los protocolos y al momento en que se empezaron a construir redes reales, con el modelo OSI se dieron cuenta que estas redes no coincidían con las especificaciones. Al contrario, en TCP/IP los protocolos llegaron primero y el modelo fue solo una descripción de los protocolos existentes, con lo que este modelo no acepta otras pilas de protocolos que no sean TCP/IP.

El modelo OSI ha demostrado que sus capas son realmente útiles en la exposición de redes de computadoras. Al contrario, sus protocolos no son populares. En el caso de TCP/IP el modelo como tal no existe, pero sus protocolos son muy utilizados.

1.3 DISPOSITIVO MEDIDOR DE CONSUMO EA

Este fue desarrollado en el Instituto de Ingeniería en la Coordinación de Electrónica, respondiendo a la necesidad del monitoreo remoto del consumo eléctrico y agua, como primeras variables. Se planea utilizar la experiencia adquirida para desarrollar nuevos dispositivos y así medir otro tipo de variables utilizadas en instrumentación.



Figura 1.5 Módulo de desarrollo Explorer 16 y tarjeta de comunicación Ethernet

Para su concepción, en primera instancia se optó por buscar algunos módulos de desarrollo comerciales, que cumplieran las condiciones de tener un microcontrolador y comunicación Ethernet. Encontrando alternativas tales como: “Texas Instruments Tiva LaunchPad EK-TM4C1294XL”, “Microchip PIC32 Ethernet Starter Kit” y “Microchip Explorer 16 Development Board”. Escogiendo la última, la cual tiene la versatilidad del poder cambiar el microcontrolador y añadir tarjetas de expansión (Figura 1.5). Algunos otros factores para escogerla fueron: experiencias anteriores con microcontroladores Microchip, además de que el fabricante proporciona Software para desarrollos acelerados en materia de programación.

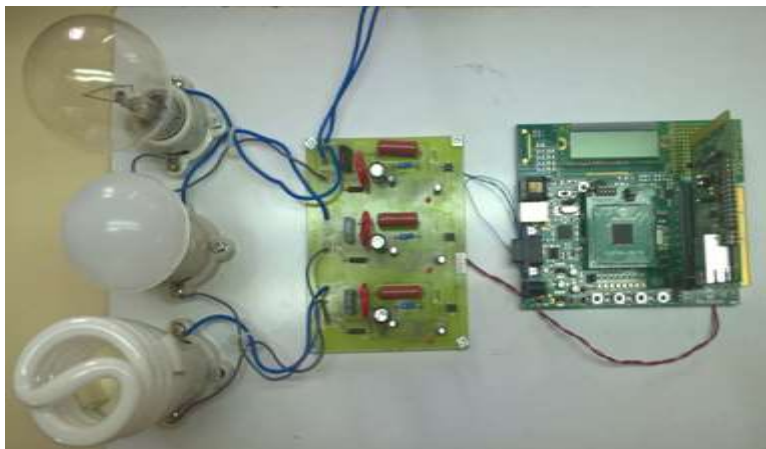


Figura 1.6 Prototipo de medidor de consumo eléctrico

La versión prototipo (Figura 1.6) estaba conformada por un medidor de consumo eléctrico que previamente se había armado, el cual indica mediante señales pulsantes que se ha consumido cierta cantidad de energía eléctrica. La señal pulsante tiene niveles de voltaje compatibles por lo que se conectó de forma directa a la tarjeta de desarrollo “Explorer 16”.

Teniendo ya la certeza que el prototipo funcionaba y guiándose por la idea de desarrollar un núcleo que permitiese la comunicación vía Ethernet, el almacenamiento mediante memoria microSD y el uso de un reloj de tiempo real, además que a este se le pudiesen añadir tarjetas adicionales. Las cuales proporcionarían las lecturas de diversos tipos de variables o transductores.

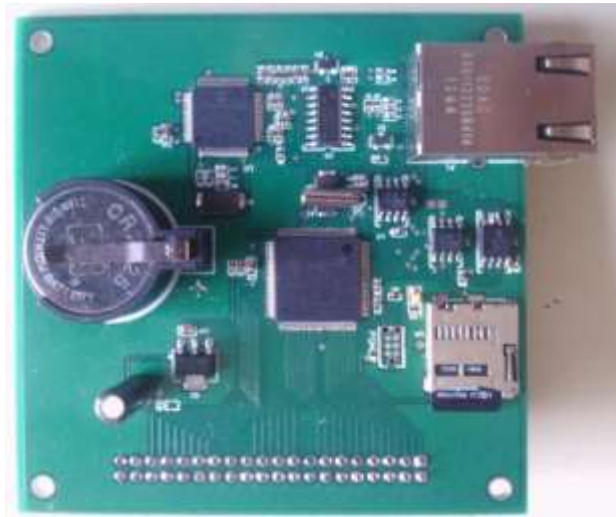


Figura 1.7 Núcleo de Comunicación Ethernet

Se desarrolló el Núcleo de Comunicación Ethernet (Figura 1.7) y la tarjeta para la medición de consumo eléctrico y agua (Figura 1.8).

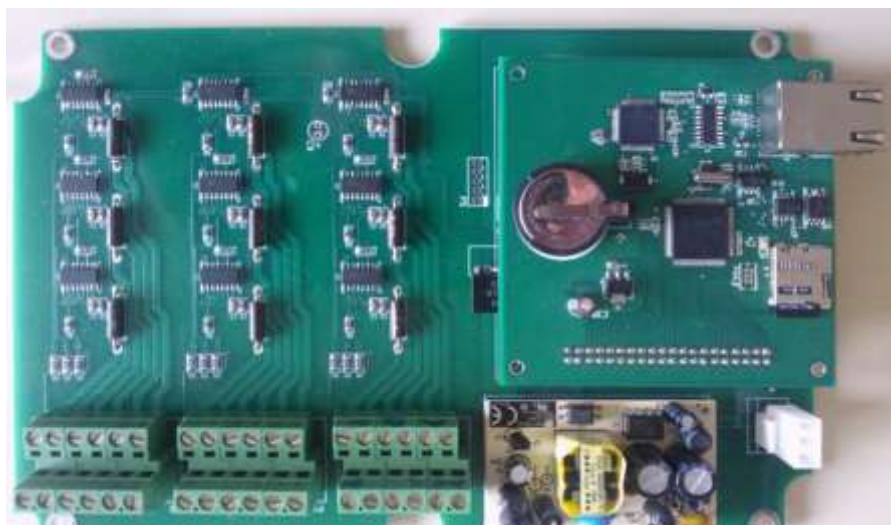


Figura 1.8 Tarjeta de Medición de Consumo EA y Núcleo de Comunicación Ethernet

1.3.1 Características

El Medidor de Consumo EA se compone del núcleo de comunicación Ethernet y la tarjeta para medición de consumo EA.

Sus especificaciones de Hardware se muestran en la tabla 2.

Tabla 2 Especificaciones del Medidor de Consumo EA

Núcleo de comunicación Ethernet	
Procesador	Microchip PIC32MX795F512L Velocidad Máxima 80Mhz Arquitectura de 32bits Memoria de Programa 512kB RAM 128kB (Véase Apéndice A)
Memoria EEPROM (Guardado de página Web y variables de programa)	Microchip 25LC1024 Capacidad de 1024 Bits
Reloj de Tiempo Real	Microchip MCP79410 Capacidad de ser alimentado por batería
LAN	Microchip ENC624J (10/100 Mbit)
Lector de Memoria microSD	Capacidad máxima de 32Gb (No incluida)
Bahía de batería	Modelo CR2016 o CR2025
Tarjeta para medición de consumo EA	
Cuantificadores de consumo Eléctrico	9 x ANALOG DEVICES ADE7757 (Véase Apéndice B)
Convertidor AC DC	90-264 C a 5 VDC

1.3.2 Diagramas

Para la programación del dispositivo es necesario conocer como están conectados los diversos elementos que lo conforman.

El Núcleo de Comunicación Ethernet tiene la peculiaridad de tener una peineta de 40 terminales las cuales están conectadas al microcontrolador y es la forma con la que se comunica con las tarjetas adicionales. En la figura 1.9 se puede observar la distribución de estos elementos.

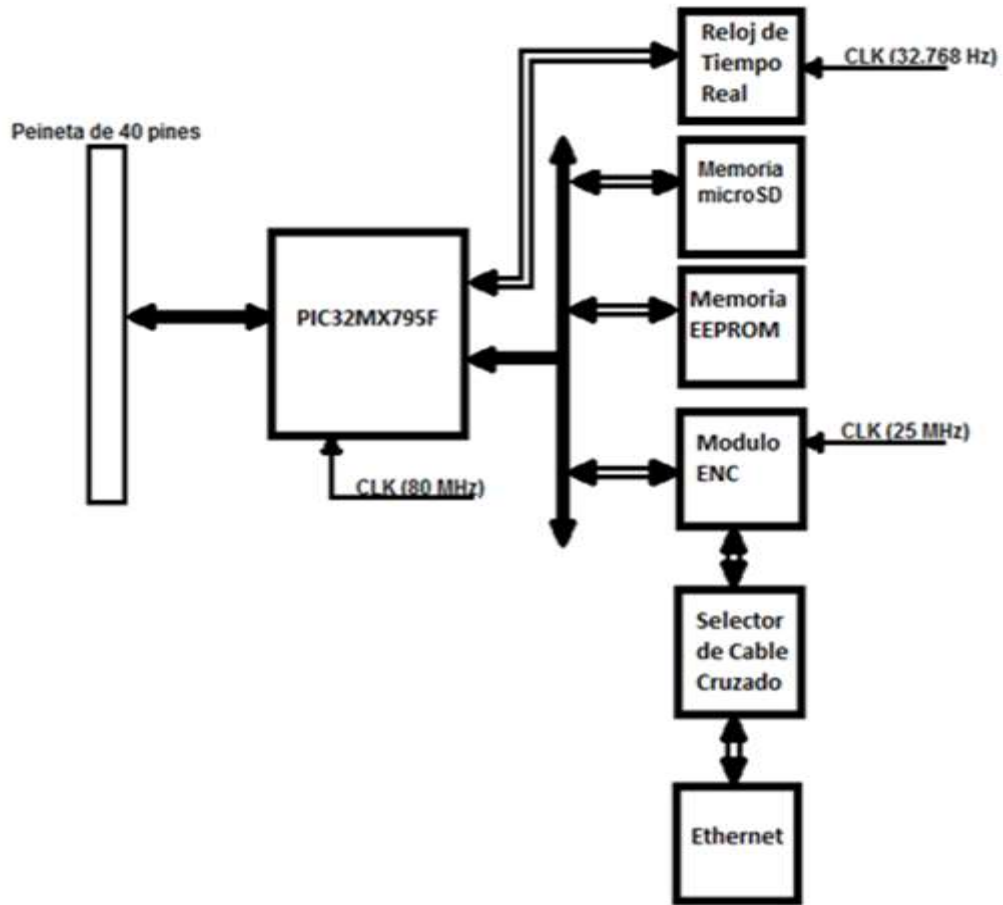


Figura 1.9 Diagrama de Núcleo de Comunicación Ethernet

La tarjeta para medición de consumo eléctrico y agua tiene la siguiente distribución de elementos (Véase la figura 1.10) donde, al igual que el núcleo, contiene su peineta de 40 terminales para comunicación.

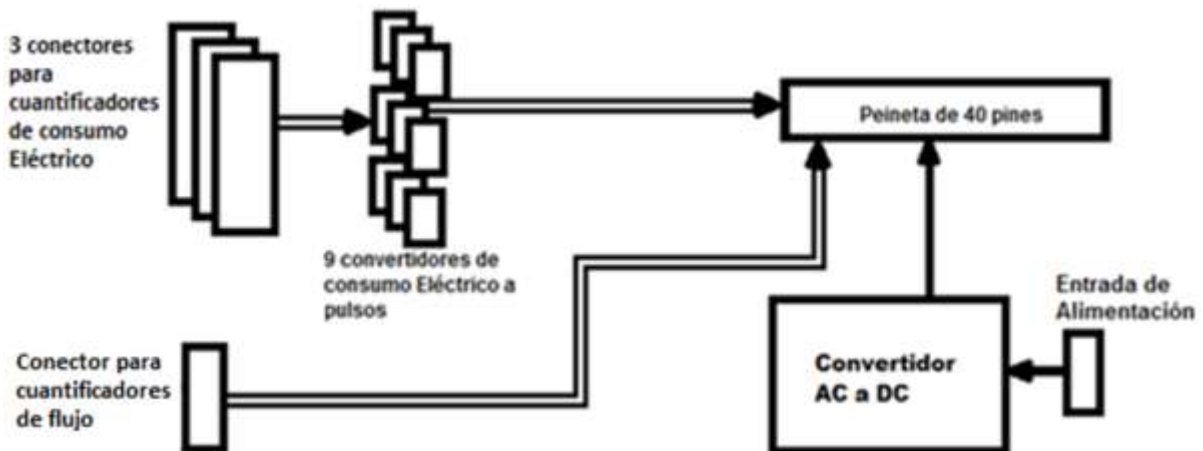


Figura 1.10 Diagrama para la medición de Consumo EA

1.3.3 Funcionamiento general

La tarjeta para medición de consumo eléctrico y agua contiene un convertidor de AC a DC que alimenta al sistema.

Se tienen tres bornes de conexión de tornillo (Figura 1.11), cada uno para una línea (En donde cada línea tiene tres fases).

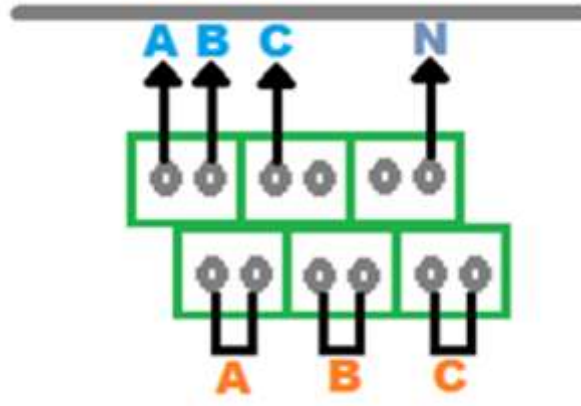


Figura 1.11 Borne de conexión de una línea

El borne de conexión se compone de:

A, B y C simples son las fases de entrada.

N es el neutro

A, B y C dobles son las conexiones hacia los transformadores de corriente.

Los transformadores de corriente que se pueden utilizar son: de dona, bipartido o flexible (Figura 1.12).



Figura 1.12 Algunos convertidores de corriente

El objetivo es pasar en medio de los convertidores el cable de corriente del elemento de estudio. Estos se conectan al borne.

También cuenta con un conector de cable a tarjeta, para medidores de caudal (Figura 1.13).

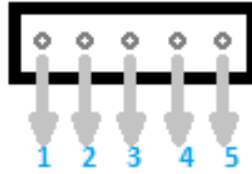


Figura 1.13 Conector para medidores de caudal

El conector se compone de:

1. Salida de alimentación positiva de 5 V
2. Tierra GND
3. Señal de entrada 1
4. Señal de entrada 2
5. Señal de entrada 3

Los medidores de caudal que se pueden conectar son del tipo eléctrico de molino (Figura 1.14).



Figura 1.14 Medidor de caudal eléctrico de molino

El circuito integrado ADE7757 contenido en la tarjeta cuantificadora se encarga de enviar un pulso cada que se acumula cierto consumo. El consumo de Agua se cuantifica de forma externa a la tarjeta de cuantificación de consumo eléctrico, esto se logra conectando un medidor de caudal del tipo electrónico molino, que cada cierto gasto másico envía un pulso. Estos pulsos se envían desde la peineta de 40 terminales hacia el núcleo. El microcontrolador recibe estas señales en canales de entrada digitales.

El núcleo tiene los elementos conectados con la siguiente disposición:

- Memoria EEPROM, Memoria microSD y Modulo ENC conectados a la interfaz SPI 4 del microcontrolador.
- Reloj de tiempo de real conectado a la interfaz I²C 2.

La disposición de terminales y canales se muestra en la tabla 3.

Tabla 3 Distribución de elementos en microcontrolador

Elemento	Terminal	Canal
SPI 4		
SDI4 (Entrada)	49	RF4
SDO4 (Salida)	50	RF5
SCK4 (Salida)	39	RF13
I2C 2		
SCL2 (Salida)	58	RA2
SDA2 (Entrada/Salida)	59	RA3
Selección de dispositivo (salidas)		
ENC	47	RD14
ENC encendido	41	RB12
EEPROM	43	RB14
MicroSD	40	RF12 (activa en bajo)
Entradas digitales		
Interrupción Reloj	51	RF3
Fase 1A	23	RB2
Fase 1B	24	RB1
Fase 1C	25	RB0
Fase 2A	20	RB5
Fase 2B	21	RB4
Fase 2C	22	RB3
Fase 3A	17	RA0
Fase 3B	18	RE8
Fase 3C	19	RE9
Flujo 1	14	RG9
Flujo 2	12	RG8
Flujo 3	11	RG7
Otras salidas		
Led	33	RB9

1.3.4 Comparación del Hardware con otros productos en el mercado

Se realiza con la finalidad de demostrar la viabilidad del Hardware desarrollado en la coordinación de electrónica del Instituto de Ingeniería de la UNAM, a nivel funcional y económico. Vale recalcar que en esta sección solo se toman en cuenta los aspectos de Hardware, ya que a todos los sistemas mencionados se les tendría que desarrollar el Software.

Se ha mencionado anteriormente que el Medidor de Consumo EA se basa en un adquisidor de datos que extrae información de las tarjetas adicionales que se le conecten. Este no es un concepto nuevo, pues existen gran cantidad de dispositivos que se basan en esta distribución y que además permiten el acceso o monitoreo remoto.

Se presenta a continuación algunos sistemas comerciales que pudiesen haber dado solución a los requerimientos del “Programa de medición y ahorro de energía – Metabolismo del Instituto”. Que por su costo o por no cumplir con algunos requisitos fueron descartados y se optó por un desarrollo propio.

El primer sistema es el conformado por “National Instruments cRIO-9063”, “National Instruments NI 9401” y tarjeta de Medición de Consumo EA. Las características que reúne este sistema son: un procesador “ARM Cortex-A9 dual-core” de 667 MHz, 512 MB de almacenamiento no volátil, 256 MB de memoria volátil, con la medición de consumo a cargo de ADE7757 con la capacidad de comunicación Ethernet y fungir como servidor web. El costo de este producto ronda los \$1644 (USD). Un costo demasiado elevado para un solo sistema teniendo en cuenta que el programa busca monitorear todos los edificios del Instituto y por esta razón fue descartado.

Ahora un sistema usando tarjetas “Arduino Mega 2560 v3”, “Arduino Energy Shield” y “Arduino Ethernet Shield 2”. Las características que reúne este sistema son: Microcontrolador ATmega2560 de 8bits, con velocidad de 16MHz, memoria no volátil de 4kB, memoria volátil de 8kB, medición de consumo energético a cargo de ADE7753, capacidad de comunicación Ethernet y fungir como servidor web, que con un costo aproximado de \$156 (USD) podrían ser la solución, pero este sistema solo es para una línea. Para tener el equivalente de la capacidad del Medidor de Consumo EA será necesario contar con 3 sistemas conformado por las ya mencionadas tarjetas, con un costo total aproximado de \$468 (USD). Este sistema también fue descartado pues los costos de Hardware del Medidor de Consumo EA son aun menores.

Se logró abatir el costo gracias a la configuración y disposición de los elementos. El núcleo de comunicación tiene un costo aproximado de \$134 (USD) y el de la

tarjeta para la medición de consumo EA de \$128 (USD) dando un total de \$262 (USD), pero este costo es solo el de los materiales, faltaría añadir costos de montaje de componentes y transporte, a pesar de esto se considera un costo mucho menor que cualquiera de las opciones analizadas anteriormente (Véase tabla 4). Y que cumple con todos los requerimientos para el “Programa de medición y ahorro de energía – Metabolismo del Instituto”. Pues se desarrolló en gran medida para darle solución.

Tabla 4 Costos de sistemas

Sistema	Costo (USD)	Costo para 3 líneas (USD)
National Instruments cRIO	1644	1644
Arduino Mega	156	468
Medidor de Consumo EA	262	262

1.4 LAS LIBRERÍAS TCP/IP DE MICROCHIP

Se encuentran incluidas en el Microchip Librerías para Aplicaciones (MLA) distribuidas de manera gratuita por Microchip donde, además, encontramos: controladores, demostraciones y utilerías.

Las librerías son de gran utilidad para desarrollos acelerados ya que proveen a los programadores de microcontroladores la posibilidad de enfocarse en la funcionalidad dejando de lado cuestiones triviales.

Otra parte importante son las demostraciones ya que no es suficiente el tener la documentación pues en ocasiones esta carece de claridad.

Las librerías TCP/IP de Microchip cumple con las características mencionadas ya que nos permite utilizar la pila de protocolos TCP/IP con facilidad e incluye una demostración del uso de la mayoría de sus funciones.

Las funcionalidades incluidas dentro de las librerías TCP/IP son (Figura 1.15):

- Soporte para los protocolos: ARP, IP, ICMP, UTP, TCP, DHCP, SNMP, HTTP, FTP, TFTP
- Soporte para Socket TCP y UDP
- Descubrimiento de dispositivos Ethernet

Las versiones más recientes del MLA solo incluye soporte de las librerías TCP/IP para Wi-Fi, para este desarrollo se utilizó la versión v2013-06-15 que es la última en incluir soporte para Ethernet.

Las librerías MLA, el compilador XC32, el IDE MPLAB X y el módulo de desarrollo “Explorer 16” proporcionan un punto de partida para el desarrollo de prototipos y sistemas finales.

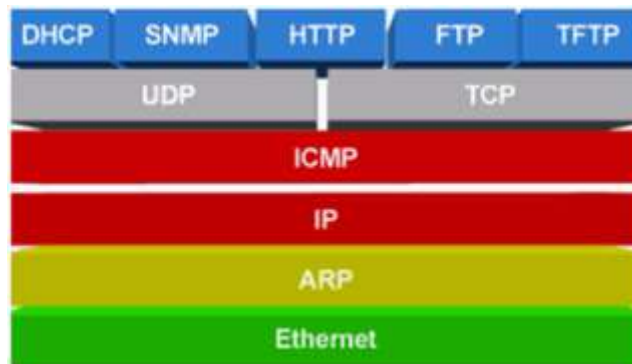


Figura 1.15 Pila de protocolos incluidos en el MLA

El desarrollo del Firmware y la página web del Medidor de Consumo EA, se basó en el proyecto de demostración incluido en las librerías TCP/IP del cual se utilizó su estructura básica y de forma evolutiva se fueron añadiendo los diversos módulos necesarios para darle la funcionalidad requerida.

El proyecto de demostración no es modificado en las secciones encargadas del manejo de los protocolos, pues estos dan la funcionalidad de comunicación Ethernet. Lo que sí es modificado para cumplir los requisitos son los servicios que están activos. Y la parte más importante son los algoritmos desarrollados que trabajan sobre estos servicios.

2 FIRMWARE

Es el programa que reside en el microcontrolador y le da toda su funcionalidad. Este se desarrolló en base al proyecto de demostración incluido en las librerías TCP/IP.

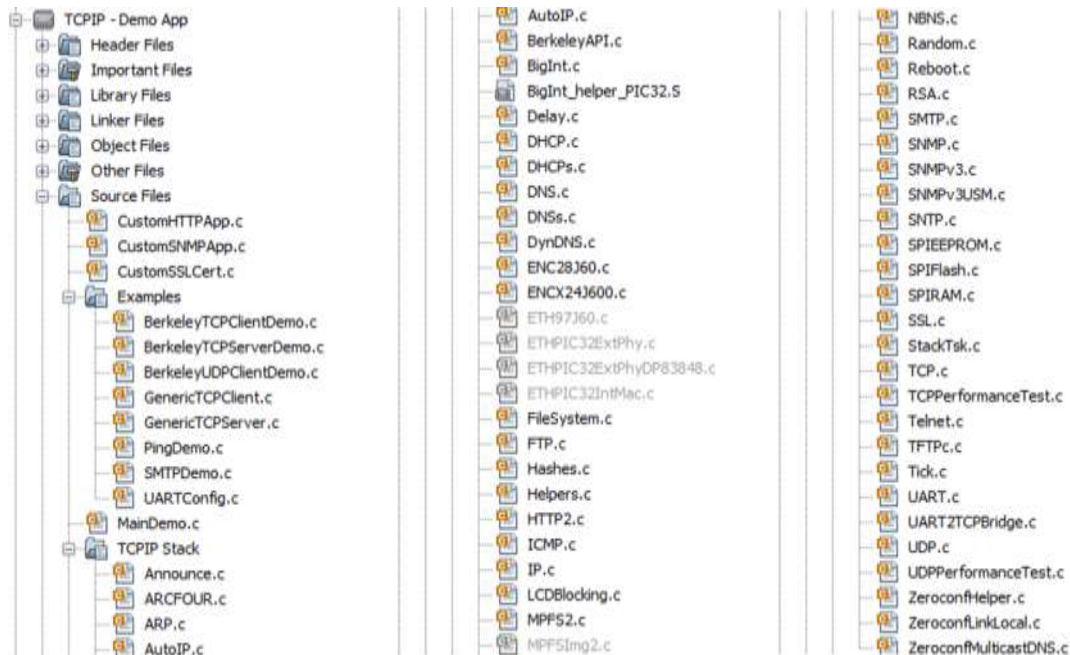


Figura 2.1 Directorio de archivos de código fuente de la aplicación demostración

El proyecto de demostración (Figura 2.1) incluye 64 archivos de código fuente o “.C” y 86 archivos de encabezado o “.h”. Estas son las dimensiones de una aplicación

básica, que si fuese desarrollada desde cero llevaría una buena cantidad de tiempo aun con un grupo de trabajo.

Ante el gran tamaño que implica la aplicación de demostración se procedió a leer la documentación y realizar pruebas para identificar sus elementos y funcionamiento, (Figura 2.2) con la finalidad de encontrar los que serían modificados o añadidos para el propósito del Medidor de Consumo EA.

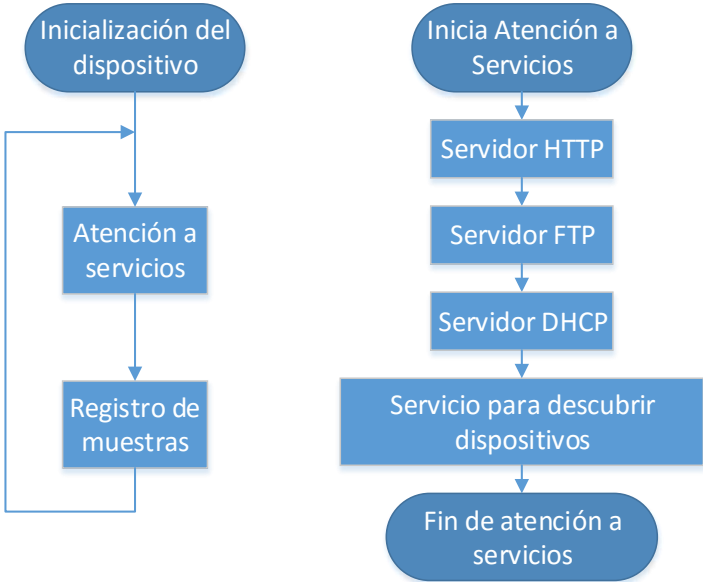


Figura 2.2 Funcionamiento general del Firmware

Además, se identificaron cuatro elementos principales: Inicialización, registro de muestras, presentación de información y recepción de datos de usuario.

2.1 INICIALIZACIÓN

Se prepara al dispositivo para su funcionamiento y se declaran la mayoría de sus recursos que serán utilizados (Figura 2.3).



Figura 2.3 Proceso de inicialización del Medidor de Consumo EA

2.1.1 Variables

El uso de variables en programación proporciona mucha de la funcionalidad y son tan esenciales que no se puede concebir ningún programa sin ellas.

Se definió la estructura muestra, la cual se observa en la figura 2.4.

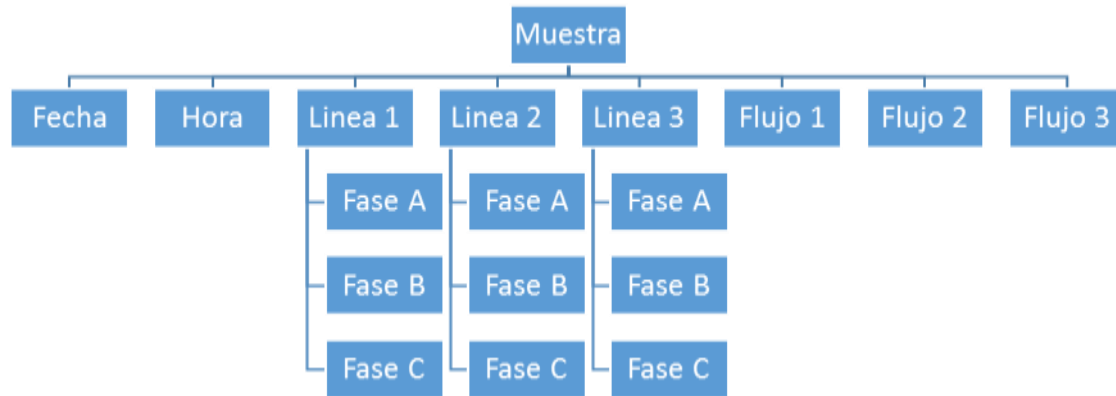


Figura 2.4 Estructura de una muestra

Se creó un arreglo con 1440 muestras, tomando en cuenta que un día (24 horas) está compuesto por 1440 minutos y que un minuto es el intervalo mínimo entre muestra y muestra. Este arreglo de muestras se almacena en la memoria RAM del microcontrolador, con lo que no es posible guardar los datos de más de un día.

Se crearon variables donde algunas son tan importantes para la ejecución del programa que no es suficiente que se almacenen en la RAM, ya que al apagar el sistema se perderían. Por tal motivo se realiza un respaldo en Memoria EEPROM para que al ser inicializado el sistema sean rescatadas y asociadas a su contraparte en RAM. Estas son para:

- Indicar qué líneas del contador se encuentran activas
- Almacenar cuentas totales
- Almacenar cuentas parciales
- Almacenar el factor de consumo eléctrico y el de agua

Otras variables creadas para el funcionamiento del programa fueron para:

- Nombre del archivo
- Estados de las entradas anteriores
- Almacenar hora y fecha
- Control de flujo del programa

Estas fueron creadas en el archivo "MainDemo.c" y "MainDemo.h".

2.1.2 Hardware

Se configuró el Hardware del microcontrolador (Véase las secciones 1.3.1 y 1.3.2), compuesto por:

- Reloj principal a 80 MHz
- Puerto de comunicación SPI 4, para interactuar con Memoria microSD, memoria EEPROM y módulo ENC
- Puerto de comunicación I²C 2, para interactuar con Reloj de Tiempo Real (RTCC)
- Puertos digitales para leer pulsos y habilitar elementos.

Esto desde el archivo “MainDemo.c” mediante la función “InitializeBoard()”.

2.1.3 Servicios

Se habilitaron los siguientes servicios de comunicación Ethernet desde el archivo de encabezado “TCPIP ENC624.h”:

- ICMP en modo cliente y servidor que proporciona la capacidad de que el dispositivo envíe y reciba transmisiones de Pings.
- Servidor HTTP2 que permite enviar páginas web.
- Auto IP para obtener la dirección de protocolo de manera automática.
- DHCP cliente para obtener parámetros de red de manera automática.
- DHCP servidor para proporcionar parámetros de red a otros dispositivos de manera automática.
- Servidor FTP el cual proporciona la capacidad de transferir archivos.
- ANNOUNCE para descubrir dispositivos Microchip en red.

2.2 REGISTRO DE MUESTRAS

Se lleva a cabo conforme el siguiente pseudocódigo el cual registra los pulsos tanto de las entradas digitales provenientes de los contadores como del reloj de tiempo real. Vale señalar que la alarma está configurada para activarse cada minuto.

Procesos IO

- ¿Si alarma activada?
 - Desactivar la alarma
 - Obtener fecha y hora del reloj
 - ¿si fecha actual es diferente a fecha de muestras?
 - Inicializa muestras con fecha actual
 - Verificar intervalo
 - ¿Si intervalo igual a un minuto?

- Guardar en memoria
- ¿si se buscó índice?
 - Índice igual a índice buscado
- ¿Si intervalo igual a cinco minutos?
 - ¿Si las unidades de los minutos de la hora son igual a cinco o cero?
 - Guardar en memoria
 - ¿si se buscó índice?
 - Índice igual a índice buscado
- ¿Si intervalo igual a diez minutos?
 - ¿si las unidades de los minutos de la hora son igual a cero?
 - Guardar en memoria
 - ¿si se buscó índice?
 - Índice igual a índice buscado
- ¿Si intervalo igual a 15 minutos?
 - ¿si los minutos de la hora es igual a cero ó 15 ó 30 ó 45?
 - Guardar en memoria
 - ¿si se buscó índice?
 - Índice igual a índice buscado
- ¿Si intervalo igual a 30 minutos?
 - ¿si los minutos de la hora es igual a cero ó 30?
 - Guardar en memoria
 - ¿si se buscó índice?
 - Índice igual a índice buscado
- En cualquier otro caso
 - No guardar en memoria
- ¿Si guardar en memoria?
 - Guardar en muestra de transferencia fecha, hora y diferencia entre contador total y contador pivote.
 - Guardar la muestra de transferencia en el arreglo de muestras en su respectivo índice
 - Guardar contadores totales en contadores pivote
 - Guardar contadores pivote en memoria EEPROM
 - ¿Si memoria microSD insertada?
 - Guardar en archivo
 - ¿Si índice igual a número de muestras?
 - Índice igual a cero
 - Si no
 - Incrementar índice

- ¿Si Fase1A cambió de estado?
 - Aumentar contador total1
 - Guardar contador total1 en memoria EEPROM
- ¿Si Fase1B cambió de estado?
 - Aumentar contador total2
 - Guardar contador total2 en memoria EEPROM
- ¿Si Fase1C cambió de estado?
 - Aumentar contador total3
 - Guardar contador total3 en memoria EEPROM
- ¿Si Fase2A cambió de estado?
 - Aumentar contador total4
 - Guardar contador total4 en memoria EEPROM
- ¿Si Fase2B cambió de estado?
 - Aumentar contador total5
 - Guardar contador total5 en memoria EEPROM
- ¿Si Fase2C cambió de estado?
 - Aumentar contador total6
 - Guardar contador total6 en memoria EEPROM
- ¿Si Fase3A cambió de estado?
 - Aumentar contador total7
 - Guardar contador total7 en memoria EEPROM
- ¿Si Fase3B cambió de estado?
 - Aumentar contador total8
 - Guardar contador total8 en memoria EEPROM
- ¿Si Fase3C cambió de estado?
 - Aumentar contador total9
 - Guardar contador total9 en memoria EEPROM
- ¿Si Flujo1 cambió de estado?
 - Aumentar contador total10
 - Guardar contador total10 en memoria EEPROM
- ¿Si Flujo2 cambió de estado?
 - Aumentar contador total11
 - Guardar contador total11 en memoria EEPROM
- ¿Si Flujo3 cambió de estado?
 - Aumentar contador total12
 - Guardar contador total12 en memoria EEPROM

Esto se agregó en el archivo “mainDemo.c”.

Algunas consideraciones que se cumplen con este pseudocódigo son:

- No se presentan todos los intervalos de muestreo pues solo aumentan el tamaño del pseudocódigo y donde su funcionamiento es muy similar a los que si son mostrados.
- Mientras exista un cambio de estado en los puertos digitales se guardará el contador respectivo en la memoria EEPROM con la finalidad de que las cuentas totales y la cuenta pivote no se pierdan en caso de falla eléctrica o apagar el sistema. Ya que la pérdida de estos datos acumulados implicaría que cada que el sistema se inicie empezaría desde cero.
- El arreglo de muestras guarda temporalmente todas las muestras que se hayan realizado (posteriormente se utilizarán para ser graficadas). La forma en que se indica el lugar que le corresponde a cada muestra es mediante un índice, la búsqueda del índice se realiza mediante “búsqueda binaria” donde al tener el arreglo con las horas en orden se tiene:

Búsqueda de índice

- Límite inferior igual a cero, límite superior igual a número de muestras
- Mientras límite inferior menor o igual a límite superior
 - Centro igual a ((límite superior menos límite inferior) entre dos) más límite inferior
 - ¿Si la hora de la muestra con índice igual al centro es igual a la hora buscada?
 - Regresa centro
 - Si no entonces
 - ¿Si hora buscada menor a la hora de la muestra con índice igual al centro?
 - Límite superior igual a centro menos uno
 - Si no
 - Límite inferior igual a centro más uno

Con esto se logra no recorrer todo el arreglo en búsqueda del índice que le corresponde, sino que vamos dividiendo el margen de búsqueda, teniendo un mejor rendimiento que buscar en la totalidad.

2.3 PRESENTACIÓN DE INFORMACIÓN

Es la forma en que la información recolectada por el microcontrolador se mostrará al usuario.

2.3.1 Envió de información a página web

Se basa en el uso del servicio de HTTP2 que se encontró implementada en la aplicación de demostración.

La infraestructura ya desarrollada por Microchip permite un fácil enlace de datos con páginas web.

Basta con colocar dentro de la página web que irá sobre el microcontrolador una etiqueta entre caracteres “~” (Figura 2.5) y proceder al empaquetado de la página con el programa MPFS2 (Figura 2.6) incluido en el MLA. De forma automática se actualizará el archivo “HTTPPrint.h” donde se añadirá el identificador a la nueva variable enlazada.

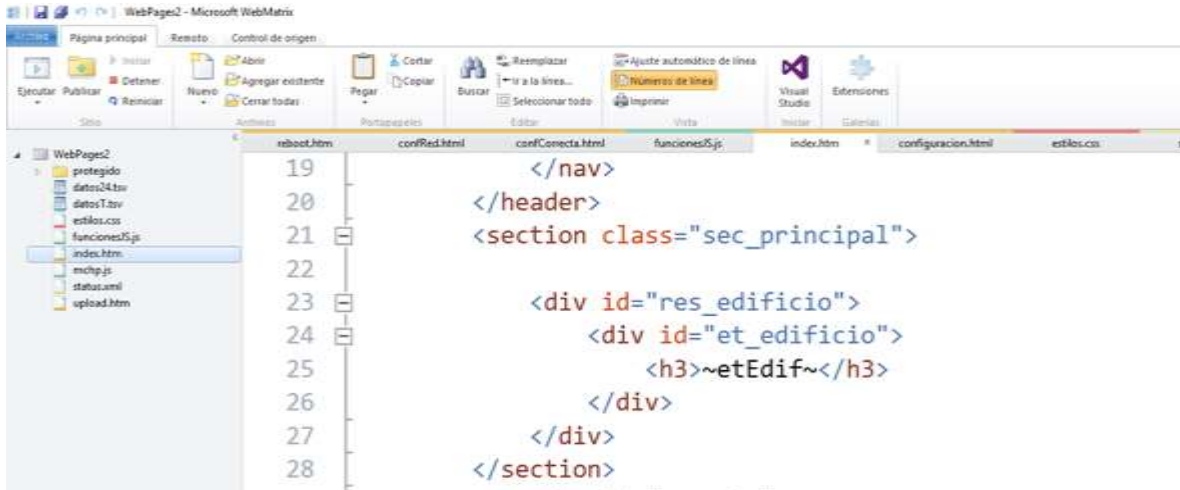


Figura 2.5 Muestra de código HTML con enlace de datos

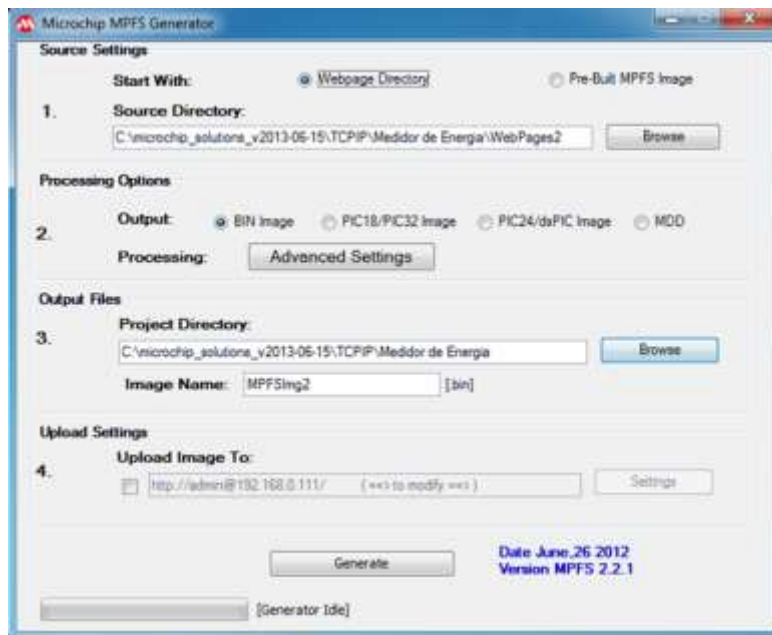


Figura 2.6 Programa MPFS2 para generar empaquetado de páginas web

Cuando realiza una petición de alguna variable enlazada se compara cual de todas las que están dadas de alta ha sido y al ser encontrada se lleva a cabo una tarea determinada por el programador.

A continuación se muestra el pseudocódigo utilizado para mostrar los datos almacenados de las muestras del día. Recordando que por cuestiones de rendimiento se utiliza el arreglo de muestras guardadas en memoria RAM que solo tiene la capacidad de guardar la información de 24 horas. Y que en la página web para su despliegue se tiene una etiqueta con la siguiente forma “~datos24~”.

Datos24

- ¿Si elementos por imprimir igual a cero?
 - Elementos por imprimir igual a índice
- Si no entoces
 - Obtener espacios disponibles para imprimir
 - Mientras espacios disponibles para imprimir mayor a 200 y elementos por imprimir mayor a cero
 - Imprimir fecha de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - Imprimir hora de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - ¿si linea1 activa?
 - Imprimir contador de faseA1, faseB1 y faseC1 de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - ¿si linea2 activa?
 - Imprimir contador de faseA2, faseB2 y faseC2 de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - ¿si linea3 activa?
 - Imprimir contador de faseA3, faseB3 y faseC3 de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - ¿si flujo1 activa?
 - Imprimir contador de flujo1 de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - ¿si flujo2 activa?
 - Imprimir contador de flujo2 de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - ¿si flujo3 activa?

- Imprimir contador de flujo3 de muestra cuyo índice es igual a (índice menos elementos por imprimir)
 - Obtener espacios disponibles para imprimir
 - Elementos por imprimir menos uno

Existe una variable muy importante a la hora de imprimir via pagina web, “curHTTP.callbackPos” que en el pseudocódigo encontramos como “Elementos por imprimir”. Su función consiste en indicar que todavia existen elementos por imprimir, además de servir de índice de información enviada ya que al ser el bufer de salida limitado, la informacion debe ser segmentada y con esta variable, se sabe en qué parte se quedó.

El proceso de envio de información toma un poco de tiempo, con lo que se vuelve necesario para el correcto funcionamiento del programa no quedar esperando a que el bufer se vacie. Se tiene que seguir atendiendo otros servicios y procesos, el flujo del programa regresa a esta sección y verifica mediante la variable “curHTTP.callbackPos” si hay mas datos por imprimir, este proceso se repite hasta que “curHTTP.callbackPos” sea igual a cero.

Al igual que en el pseudocódigo “datos24” se crearon funciones mas pequeñas para imprimir otro tipo de datos, que por su menor tamaño y parecido no se considera esencial mostrar su pseudocódigo.

Entre ellas se encuentra:

- Dirección de IP asignada
- Dirección MAC asignada
- Dirección de puerta de enlace asignada
- Datos de transferencia
- Estado del dispositivo
- Nombre de variables
- Consumo eléctrico
- Consumo de Agua
- Intervalo seleccionado
- Etiqueta de edificio
- Entradas seleccionadas

2.3.2 Guardar información dentro de archivo de texto en memoria microSD

Se lleva a cabo cada que la alarma se activa. Se crea, si es necesario (en caso de no existir), el árbol de directorios mostrado en la figura 2.7.

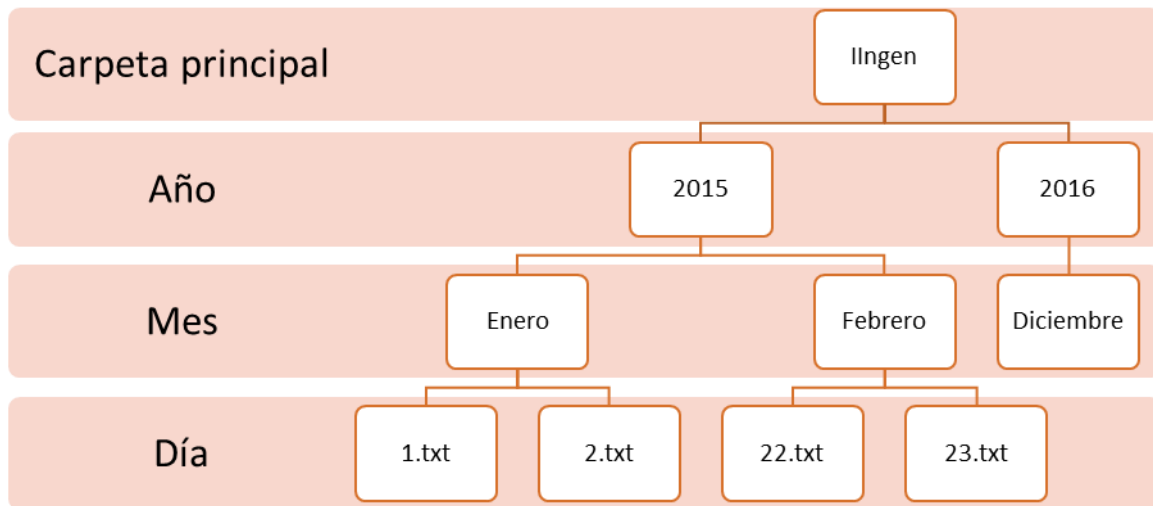


Figura 2.7 Árbol de directorios que se creará en la memoria microSD

Donde por cada día se crea un archivo nombrado con el número de día al que corresponde.

Cabe resaltar que el manejo de memoria microSD no estaba soportado en la aplicación de demostración, con lo que se hizo un trabajo de integración.

El pseudocódigo siguiente plasma la forma en que se almacenan las muestras:

Guarda en memoria microSD

- ¿Se tiene memoria microSD insertada?
 - Obtener fecha
 - ¿se tiene árbol de archivos que corresponde a la fecha?
 - ¿Se tiene archivo del día actual?
 - Crear archivo con nombre del número de día
 - Abrir archivo del día actual
 - ¿Si archivo del día actual se pudo abrir?
 - Escribir en archivo fecha y hora
 - ¿Si línea 1 activa?
 - Escribir en archivo sus contadores de fase multiplicado por factor de consumo eléctrico.
 - Si no
 - Escribir tres “?” en archivo
 - ¿Si línea 2 activa?
 - Escribir en archivo sus contadores de fase multiplicado por factor de consumo eléctrico.
 - Si no

- Escribir tres “?” en archivo
- ¿Si línea 3 activa?
 - Escribir en archivo sus contadores de fase multiplicado por factor de consumo eléctrico.
- Si no
 - Escribir tres “?” en archivo
- ¿Si Flujo 1 activo?
 - Escribir en archivo su contador de multiplicado por factor de flujo.
- Si no
 - Escribir en archivo “?”
- ¿Si Flujo 2 activo?
 - Escribir en archivo su contador de multiplicado por factor de flujo.
- Si no
 - Escribir en archivo “?”
- ¿Si Flujo 3 activo?
 - Escribir en archivo su contador de multiplicado por factor de flujo.
- Si no
 - Escribir en archivo “?”
- Escribir un salto de línea en archivo
- Cerrar archivo

2.4 RECEPCIÓN DE DATOS DE USUARIO

El programa desarrollado necesita algunos parámetros de configuración que, dependiendo del lugar en donde será instalado el Medidor de Consumo EA, serán modificados a conveniencia por el usuario.

La información se recibe mediante algún formulario dentro de la página web, esta se puede enviar mediante método “POST” o “GET” al microcontrolador. En la mayoría de formularios utilizados se envía la información mediante método “POST” ya que nos permite la transferencia de información sin que aparezca en la URL, que es lo que ocurre si se usa “GET”.

Los métodos “GET” y “POST” se encuentran en los servicios proporcionado por HTTP.

Cuando se envía la información, los datos llegan al buffer TCP. Estos datos incluyen qué página web los envía, qué identificador tiene la variable enviada y el dato de la variable.

En una primera instancia el programa identifica si se ha enviado información y espera a recibir la totalidad de estos. Procede a identificar qué página la envió y ejecuta la función asociada a la página. En esta función se discrimina el identificador de la o las variables que se encuentran en la página. Se rescatan los datos y estos están ya a disposición del programa para ser asociados a la variable interna, sin antes ser procesada pues es recibida en código ASCII. Para finalizar se redirige al usuario una página web de respuesta para indicar que el proceso se ha llevado a cabo.

Los datos que se reciben y se asocian a variables internas son:

- Activación/Desactivación de líneas
- Activación/Desactivación de flujos
- Fecha y Hora
- Intervalo de muestreo
- Reiniciar contadores
- Etiqueta de localización del dispositivo
- Configuraciones de red

Todo esto se realiza en el archivo "CustomHTTPApp.c" y en la función "HTTPExecutePost()".

3 INTERFAZ WEB

Es la principal forma de interacción con el Medidor de Consumo EA, desde ella se pueden consultar los datos generados tanto en forma de texto o gráficas, además de permitir la configuración de algunas variables.

Para el despliegue de páginas Web se utiliza el servicio HTTP incluido en el Firmware del microcontrolador. Las páginas se desarrollaron utilizando el lenguaje de marcado de hipertexto HTML5, junto con hojas de estilos en cascada CSS y el lenguaje de programación JavaScript.

Como una característica que no es esencial, pero proporciona una forma de interpretar la información de manera más sencilla, se incluyó la capacidad de generar gráficas en la página principal. Se utilizó una serie de librerías (API) que funcionan con JavaScript llamada D3.js que facilita la creación de gráficas.

Cabe señalar que la aplicación de demostración que se ha utilizado como base, incluía una página web (Figura 3.1) en la que se muestra el uso del enlace de datos y de funciones para realizar consultas de información dinámica. Estas funciones utilizan AJAX que se basa en JavaScript y es utilizado para mantener comunicación asíncrona en segundo plano con el servidor y proporcionar una retroalimentación dinámica a las páginas web.

La página web que se diseñó contiene 13 archivos, donde podemos encontrar páginas HTML, así como documentos para enlace de datos, archivos de estilo CSS y archivos JS que contienen funciones, se pueden observar los directorios en las figuras 3.2 y 3.3.

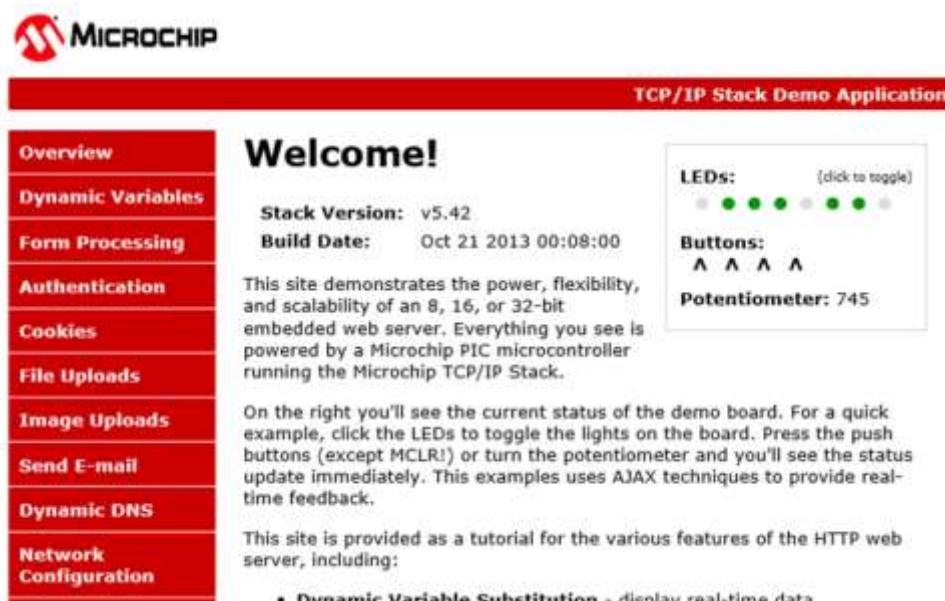


Figura 3.1 Página de inicio de aplicación de demostración

protegido	05/11/2015 04:00 ...	Carpeta de archivos	
datos24.tsv	28/04/2015 01:22 ...	Archivo TSV	1 KB
datosT.tsv	11/06/2015 12:25 ...	Archivo TSV	1 KB
estilos	11/09/2015 01:29 ...	Documento de ho...	5 KB
funcionesJS	10/09/2015 05:20 ...	Archivo JS	21 KB
index	10/09/2015 04:53 ...	Archivo HTM	3 KB
mchp	23/06/2015 05:00 ...	Archivo JS	4 KB
status	23/06/2015 01:28 ...	Documento XML	1 KB
upload	12/05/2015 02:11 ...	Archivo HTM	1 KB

Figura 3.2 Carpeta principal de la página web del Medidor de Consumo EA

confCorrecta	02/09/2015 06:32 ...	Archivo HTML	3 KB
configuracion	11/09/2015 12:39 ...	Archivo HTML	11 KB
confRed	03/09/2015 04:24 ...	Archivo HTML	5 KB
reboot.cgi	18/09/2014 08:05 ...	Archivo CGI	1 KB
reboot	03/09/2015 04:24 ...	Archivo HTM	3 KB

Figura 3.3 Contenido de la carpeta protegido

3.1 PÁGINA DE INICIO

Es aquella con la que el Medidor de Consumo EA responde al colocar en el navegador la dirección IP asignada a este (Figura 3.4).



Figura 3.4 Página de inicio del Medidor de Consumo EA

En ella se pueden apreciar los siguientes elementos:

1. Letrero de identificación del dispositivo.
2. Botones para ir a la Configuración (engrane) y refrescar la página (casa).
3. Letrero de Acumulados y Estado del Dispositivo.
4. Gráfica de consumo eléctrico.
5. Gráfica de flujo.
6. “*Link*” de descarga de archivo de registro del día actual.
7. Letrero de información.

De los elementos que lo conforman, encontramos aquellos que son solo texto e imágenes y los que tienen alguna función asociada. A continuación mostramos aquellos que tienen una función asociada.

3.1.1 Letrero de identificación del dispositivo

Este está enlazado con una variable dentro del microcontrolador, la cual está respaldada en memoria EEPROM pues es necesario conservar el nombre aun después de apagar el medidor.

Se utilizó el enlace básico, se coloca el nombre de la variable entre “~”. Dentro del código HTML la encontramos de la siguiente forma “~etEdif~”. Y al ser consultada la página automáticamente se substituye la etiqueta por el valor de la variable enlazada en este caso “Edificio 12”.

3.1.2 Letrero de acumulados y estado del dispositivo

Tiene la peculiaridad de que los datos son actualizados cada 30 segundos. Se eligió este intervalo, ya que hacerlo de forma más rápida implica que se está generando tráfico en la red y si se diera el caso de que se esté usando una red de datos de uso medido, provocaría un rápido consumo de estos. Adicionalmente no se considera esencial realizar actualizaciones más rápidas ya que el medidor actualiza esta variable cada minuto, con lo que no es relevante pedir información actualizada cuando el medidor no la tiene.

Esta variable utiliza la infraestructura de la aplicación de demostración donde apoyándose en un archivo intermedio, el cual contiene solo la variable que se actualiza rápidamente (para no actualizar todo el documento completo). Con funciones AJAX se hace la petición al Medidor de Consumo de las variables que se encuentren en el archivo intermedio, cada cierto tiempo. Con funciones JavaScript se rescata la información del archivo intermedio y se muestra la página.

En este caso el enlace de datos se realiza mediante el archivo intermedio, con lo que es en este donde se coloca la etiqueta de la variable entre “~”. El archivo

intermedio es del tipo: Lenguaje de Marcas Extensible (XML). Utilizado para comunicar varias aplicaciones y expresar información de manera más estructurada. Se ha utilizado para facilitar a las funciones JavaScript la identificación de la información requerida.

El contenido del archivo intermedio usado es el siguiente:

```
<response>
<edo>~edo~</edo>
<\response>
```

Donde Medidor de Consumo EA substituye “~edo~” por una cadena de texto que contiene la siguiente información:

- Fecha actual
- Hora actual
- El nombre de las líneas y su acumulado total. En caso de no estar activa se muestra el nombre de la línea seguido de NA.
- El total del acumulado de todas las líneas.
- Si existe al menos una línea activa, se muestra el factor de consumo eléctrico asignado. En caso contrario no se muestra esta información.
- El nombre de los flujos y su acumulado total. En caso de no estar activo se muestra el nombre del flujo seguido de NA.
- Si existe al menos un flujo activo, se muestra el factor de flujo asignado. En caso contrario no se muestra esta información.

Y con la función JavaScript, “*getXMLValue()*” se busca la etiqueta “edo”, se toma su valor y se muestra en la página.

3.1.3 Gráficas

Son una forma de mostrar la información y que el usuario la interprete fácilmente. Para su concepción se usó la API JavaScript D3.js la cual sirve perfectamente para estos propósitos.

En primera instancia se realizó una investigación para encontrar una forma sencilla y ligera de realizar gráficas, se probó la alternativa de dibujar directamente sobre el “Canvas” de HTML5 utilizando JavaScript pero se encontró con una gran cantidad de complicaciones, las funciones que se tenían que desarrollar tomarían gran cantidad de tiempo de desarrollo, con lo que se decidió buscar alguna API para el trazado de las gráficas, encontrando al D3.js que es muy especializado en este tema, no utiliza complementos adicionales (“plugins”), provee gran cantidad de ejemplos de uso y lo más importante, es Software libre con licencia BSD 3. La cual permite el uso del código fuente en Software no libre.

Se estudiaron varias gráficas de ejemplo, donde la que más aportó para entender el uso de la API fue la multi-series (Figura 3.5). Y de donde se tomó la estructura básica, añadiendo funcionalidades que se necesitaban para un correcto despliegue.

mbostock's block #3884955 · October 13, 2012

Multi-Series Line Chart

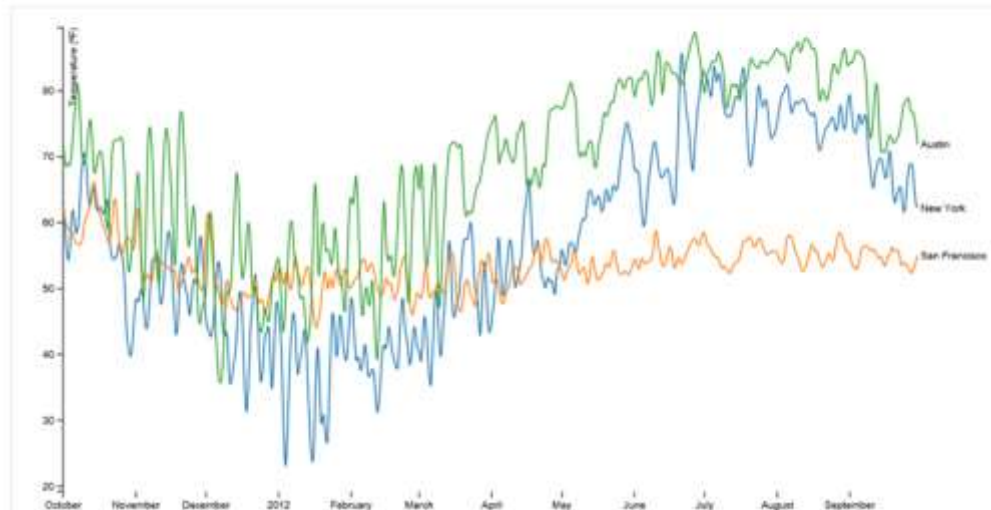


Figura 3.5 Gráfica multi-series, ejemplo del uso del API D3.js

Las gráficas tienen la característica de obtener los datos que se muestran desde el archivo llamado “datos24.tsv”. La extensión del archivo “.tsv” hace referencia a que es un archivo que utiliza tabuladores como carácter de separación de filas.

El archivo “datos24.tsv” utiliza un enlace de datos básico y el Medidor de Consumo EA actualiza los datos según el intervalo de muestreo. Y la información que contiene se detalló en la sección 2.3.1 y de forma sencilla, contiene las muestras que se llevaron a cabo a lo largo del día, en forma de lista, teniendo la estructura de fecha, hora y cuentas parciales tanto de fases como de líneas según la configuración.

Se muestra a continuación el algoritmo que se desarrolló para mostrar las dos gráficas, teniendo funcionalidades de mostrar solo las gráficas si se tienen datos que mostrar (si mínimo una línea o un flujo está activo) para no mostrar graficas vacías, así como la capacidad de ocultar o mostrar solo las líneas o flujos que se desee, mediante la selección del cuadro de activación.

Carga al iniciar navegación

- Se verifica que exista conexión a internet
- ¿Si se tiene conexión a internet?

- Se grafica de nuevo con nuevos datos
- Se repite cada minuto
 - Se grafica de nuevo con nuevos datos

Se grafica de nuevo

- ¿Si se usan nuevos datos?
 - Cargar datos a la gráfica
- Si no
 - Dibuja

Carga datos a la gráfica

- Se crean arreglos de datos de líneas y de flujos
- Extraer encabezados del archivo datos24.tsv
 - ¿si se encuentra la llave Fase_1A?
 - Se grafica consumo eléctrico
 - ¿si se encuentra la llave Fase_2A?
 - Se grafica consumo eléctrico
 - ¿si se encuentra la llave Fase_3A?
 - Se grafica consumo eléctrico
 - ¿si se encuentra la llave Flujo_1?
 - Se grafica flujo
 - ¿si se encuentra la llave Flujo_2?
 - Se grafica flujo
 - ¿si se encuentra la llave Flujo_3?
 - Se grafica flujo
- ¿Si se grafica el consumo eléctrico?
 - Se obtiene el identificador de donde se colocará la gráfica de consumo eléctrico dentro de la pagina
 - Se añade en página web la etiqueta “<h3>Consumo Eléctrico</h3>”
 - Se crea elemento svg con identificador #grafica1
- ¿Si se grafica el flujo?
 - Se obtiene el identificador de donde se colocará la gráfica de flujo dentro de la página
 - Se añade en página web la etiqueta “<h3>Flujo</h3>”
 - Se crea elemento svg con identificador #grafica2
- Revisar todas las llaves
 - ¿Si se encuentra la llave Línea_1A?
 - Se añade al arreglo de líneas a Línea 1
 - ¿Si no encuentra visible el cuadro de activación 0?

- Se añade en la sección de #grafica1 el letrero “Línea 1” un cuadro de activación con identificador “chLin1” y se establece como visible al cuadro de activación 0.
 - Se añade al cuadro de activación con identificador “chLin1” la función de Graficar de nuevo sin datos nuevos
- ¿Si se encuentra la llave Línea_2A?
 - Se añade al arreglo de líneas a Línea 2
 - ¿Si no encuentra visible el cuadro de activación 1?
 - Se añade en la sección de #grafica1 el letrero “Línea 2” un cuadro de activación con identificador “chLin2” y se establece como visible al cuadro de activación 1.
 - Se añade al cuadro de activación con identificador “chLin2” la función de Graficar de nuevo sin datos nuevos
- ¿Si se encuentra la llave Línea_3A?
 - Se añade al arreglo de líneas a Línea 3
 - ¿Si no encuentra visible el cuadro de activación 2?
 - Se añade en la sección de #grafica1 el letrero “Línea 3” un cuadro de activación con identificador “chLin2” y se establece como visible al cuadro de activación 2.
 - Se añade al cuadro de activación con identificador “chLin3” la función de Graficar de nuevo sin datos nuevos
- ¿Si se encuentra la llave Flujo_1?
 - Se añade al arreglo de Flujos a Flujo 1
 - ¿Si no encuentra visible el cuadro de activación 3?
 - Se añade en la sección de #grafica2 el letrero “Flujo 1” un cuadro de activación con identificador “chFlujo1” y se establece como visible al cuadro de activación 3.
 - Se añade al cuadro de activación con identificador “chFlujo1” la función de Graficar de nuevo sin datos nuevos
- ¿Si se encuentra la llave Flujo_2?
 - Se añade al arreglo de Flujos a Flujo 2
 - ¿Si no encuentra visible el cuadro de activación 4?
 - Se añade en la sección de #grafica2 el letrero “Flujo 2” un cuadro de activación con identificador “chFlujo2” y se establece como visible al cuadro de activación 4.

- Se añade al cuadro de activación con identificador “chFlujo2” la función de Graficar de nuevo sin datos nuevos
 - ¿Si se encuentra la llave Flujo_3?
 - Se añade al arreglo de Flujos a Flujo 3
 - ¿Si no encuentra visible el cuadro de activación 5?
 - Se añade en la sección de #grafica2 el letrero “Flujo 3” un cuadro de activación con identificador “chFlujo3” y se establece como visible al cuadro de activación 5.
 - Se añade al cuadro de activación con identificador “chFlujo3” la función de Graficar de nuevo sin datos nuevos
 - Dibuja

Dibuja

- Elimina elemento svg de la página web
- Elimina elemento svg de la página web
- Se crea elemento svg con identificador #grafica1
- Se crea elemento svg con identificador #grafica2
- Se obtiene la longitud del arreglo de líneas
- Se crea un arreglo auxiliar
- ¿Si la longitud del arreglo de líneas es mayor a cero?
 - Se revisa todo el arreglo de líneas
 - Si el arreglo de líneas contiene a Línea 1 y el cuadro de activación 0 está activo
 - Se añade a un arreglo auxiliar Línea 1
 - Si el arreglo de líneas contiene a Línea 2 y el cuadro de activación 1 está activo
 - Se añade a un arreglo auxiliar Línea 2
 - Si el arreglo de líneas contiene a Línea 3 y el cuadro de activación 2 está activo
 - Se añade a un arreglo auxiliar Línea 3
 - Se añade al dominio 1 la cadena auxiliar
- Si no
 - Se muestra en la consola “No se tienen líneas activas ☹”
- Se obtiene la longitud del arreglo de flujos
- Se limpia la cadena auxiliar
- ¿Si la longitud del arreglo de flujos es mayor a cero?
 - Se revisa el arreglo de flujos

- Si el arreglo de flujos contiene a Flujo 1 y el cuadro de activación 3 está activo
 - Se añade al arreglo auxiliar Flujo 1
 - Si el arreglo de flujos contiene a Flujo 2 y el cuadro de activación 4 está activo
 - Se añade al arreglo auxiliar Flujo 2
 - Si el arreglo de flujos contiene a Flujo 3 y el cuadro de activación 5 está activo
 - Se añade al arreglo auxiliar Flujo 3
 - Se añade al dominio 2 la cadena auxiliar
 - Si no
 - Se muestra en consola “No se contienen flujos activos ☹”
- Se crean los puntos para las líneas de la gráfica 1
 - Se obtiene del dominio 1 el mapeo de datos
 - El nombre de la línea
 - Los valores de la línea
 - Si el nombre es línea 1
 - Se acumula el valor de la fase 1A más fase 1B más fase 1C
 - Se crea el punto con hora y acumulado
 - Si el nombre es línea 2
 - Se acumula el valor de la fase 2A más fase 2B más fase 2C
 - Se crea el punto con hora y acumulado
 - Si el nombre es línea 3
 - Se acumula el valor de la fase 3A más fase 3B más fase 3C
 - Se crea el punto con hora y acumulado
- Se crean los puntos para las líneas de la gráfica 2
 - Se obtiene del dominio 2 el mapeo de datos
 - El nombre del flujo
 - Los valores de flujo
 - Si el nombre es flujo 1
 - Se acumula el valor de la fase 1A más fase 1B más fase 1C
 - Se crea el punto con hora y acumulado
 - Si el nombre es flujo 2
 - Se acumula el valor de la fase 2A más fase 2B más fase 2C
 - Se crea el punto con hora y acumulado

- Si el nombre es flujo 3
 - Se acumula el valor de la fase 3A más fase 3B más fase 3C
 - Se crea el punto con hora y acumulado
- Se crea el dominio X que va desde 00:00:00 a 23:59:00
- Se crea el dominio Y1 que va desde 0 a el valor máximo contenido en los datos de líneas
- Se crea el dominio Y2 que va desde 0 a el valor máximo contenido en los datos de flujos
- Añadir a svg1 los ejes XY cuyos dominios están dados por X y Y1
- Añadir a svg2 los ejes XY cuyos dominios están dados por X y Y2
- Añadir a svg1 etiquetas para identificar líneas de consumo eléctrico
- Añadir a svg2 etiquetas para identificar líneas de flujo
- Se añade el dominio 1 que contiene los puntos para generar líneas al svg1
- Se añade el dominio 2 que contiene los puntos para generar líneas al svg2
- Se pintan las líneas dentro del svg1
- Se pintan las líneas dentro del svg2
- Al finalizar el pintado de líneas se añade una etiqueta que corresponde al nombre de cada una.

3.1.4 Link de descarga de datos del día

Donde se obtiene el archivo “datos24.tsv” para descargar por el usuario. Se utilizó la clásica etiqueta de links para navegación, junto con “download” para indicarle al navegador que no queremos visualizar el archivo, sino que lo queremos descargar. El código para lograrlo es el siguiente:

```
<a href="datos24.tsv" download="datos.tsv">
```

3.1.5 Botones de navegación

Se trata de la imagen de casa y engrane, los cuales tienen la función de redirigir al usuario a otras partes de la página. La casa nos regresa al inicio y el engrane a configuración.

Se utilizaron imágenes con “link” para que hagan su función.

3.2 PÁGINA DE CONFIGURACIÓN

Esta tiene la característica de estar dentro de la carpeta protegido. La cual requiere que para acceder al contenido sea necesaria una autenticación (Figura 3.6) dada por usuario y contraseña para que un usuario no autorizado no pueda modificar los datos.



Figura 3.6 Petición de usuario y contraseña para acceder a la página de configuración

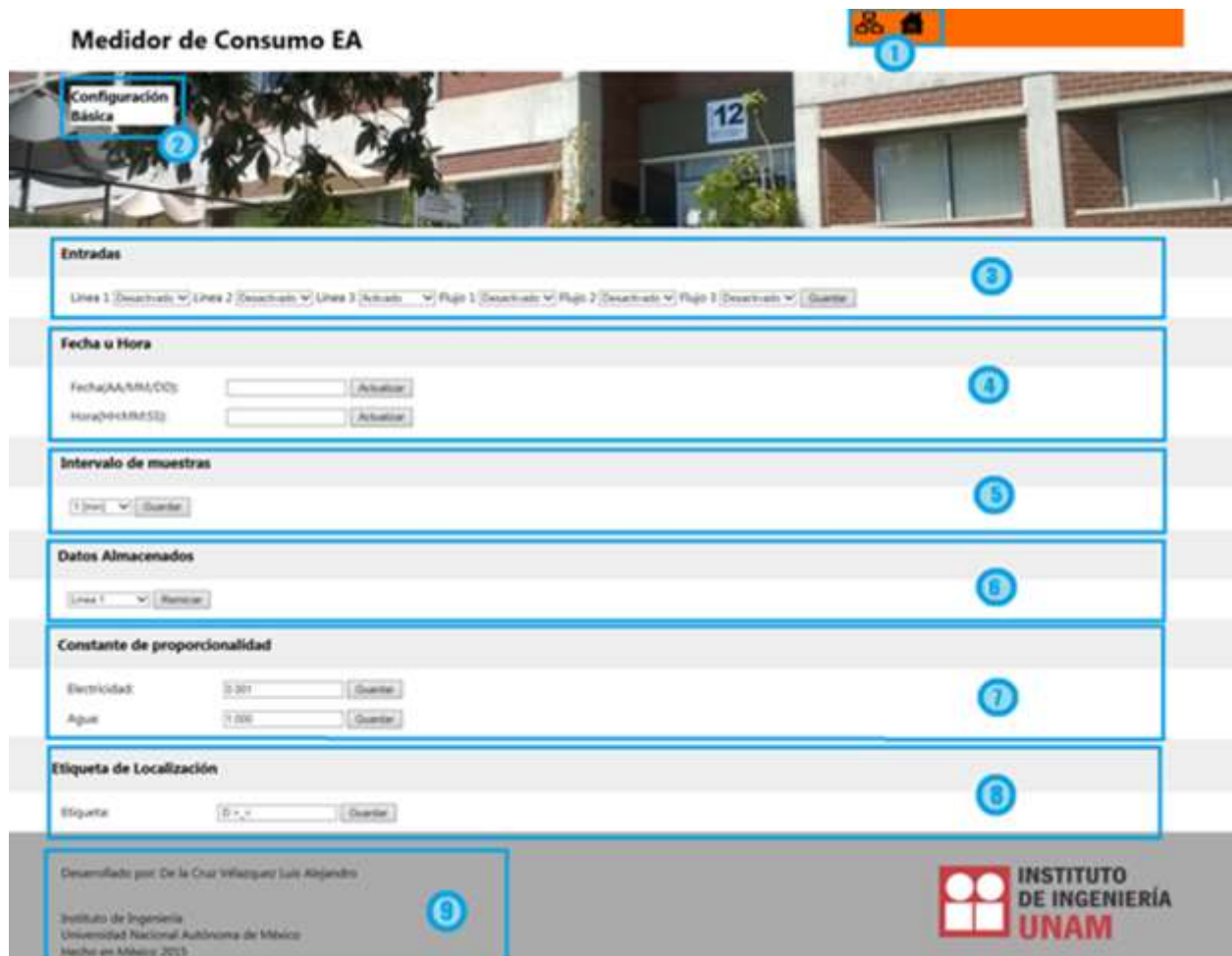


Figura 3.7 Página de configuración del Medidor de Consumo EA

A la página de configuración (Figura 3.7) se accede mediante el icono de engrane desde la página principal o colocando la dirección de la página en el navegador.

En ella se presentan los siguientes elementos:

1. Botones de navegación para ir a configuraron de red (computadoras conectadas) e inicio (casa)
2. Letrero de identificación de la página
3. Entradas activas
4. Fecha y hora
5. Intervalo de muestra
6. Reiniciar datos almacenados
7. Constantes de proporcionalidad de Electricidad y agua
8. Etiqueta de localización
9. Etiqueta con información

En la sección 2.4 ya se ha tratado cómo es que se recibe la información y se asocia a variables internas. Ahora se explicarán algunas validaciones y el comportamiento de los formularios.

3.2.1 Listas desplegadas

Agrupar a la configuración de entradas, intervalo de muestras y datos almacenados. Se explica el funcionamiento de las entradas, pero los otros dos formularios funcionan de forma muy similar o más simple (en el caso de datos almacenados donde no es necesario mostrar su estado).

Se trata de un formulario que agrupa seis campos de selección los cuales tienen la función de activar o desactivar las entradas del dispositivo. Cada campo de selección tiene un identificador propio el cual envía al Medidor de consumo EA el valor de 0 para desactivado y 1 para activado mediante método "POST".

A continuación, se muestra el código de un campo de selección. Existe un enlace de datos que permite obtener información desde el medidor de consumo para poder observar el estado actual de la entrada (si se encuentra activada o desactivada al momento de entrar en la página de configuración).

```
<select name="Linea1">  
<option value="1" ~enSelec(0,TRUE)~>Activado </option>  
<option value="0" ~enSelec(0,FALSE)~>Desactivado</option>  
</select>
```

Se tiene un enlace de datos básico, cada etiqueta representa elementos diferentes en el caso de que la línea 1 estuviese activada con anterioridad la etiqueta "`~enSelec(0,TRUE)~`" será sustituida por "selected" y "`~enSelec(0,FALSE)~`" por una cadena vacía. Con lo que el navegador interpretará que la opción activada es la que se tiene que mostrar en el formulario como seleccionada.

En el caso de intervalo de muestras y datos almacenados se envía un intervalo de números mayor pues tienen más opciones.

3.2.2 Datos con validación

En esta clasificación entran los formularios fecha u hora, constante de proporcionalidad y etiqueta de localización.

Consiste en que antes de ser enviada la información mediante funciones JavaScript la información sea validada, en caso de cumplir los criterios esta es enviada al Medidor de Consumo EA, en caso contrario se envía un mensaje al usuario para que corrija sus errores. Esto para evitar que dentro del microcontrolador se tuviesen que implementar estas validaciones, además que es mucho más sencillo llevarlas a cabo mediante JavaScript y de esta forma el microcontrolador se dedica a sus tareas principales.

El pseudocódigo que describe la validación se muestra a continuación. Se debe tener en cuenta que los formularios mencionados llaman a esta función, esta discrimina quien la llama y utiliza expresiones regulares para validar el formato.

Valida tipo

- ¿Si es fecha?
 - Se obtiene la fecha del formulario
 - Se valida formato de fecha
 - ¿Si el formato es válido?
 - Se comprueba que la fecha exista
 - ¿Si la fecha existe?
 - Se envía la información al Medidor de Consumo EA
 - Si no
 - Se muestra una alerta “La fecha introducida no existe.”
 - Se envía una cadena vacía al Medidor de Consumo EA
 - Si no
 - Se muestra una alerta “El formato de la fecha es incorrecto”
 - Se envía una cadena vacía al Medidor de Consumo EA
- ¿Si es hora?
 - Se obtiene la hora desde el formulario
 - Se valida formato de hora
 - ¿Si el formato es válido?
 - Se envía la información al Medidor de Consumo EA
 - Si no
 - Se muestra una alerta “El formato de la fecha es incorrecto”
 - Se envía una cadena vacía al Medidor de Consumo EA

- ¿Si es constante de consumo eléctrico?
 - Se obtiene la constante de consumo eléctrico desde el formulario
 - Se valida formato de constante de consumo eléctrico
 - ¿Si el formato es válido?
 - Se envía la información al Medidor de Consumo EA
 - Si no
 - Se muestra una alerta “El formato de constante es incorrecto”
- ¿Si es constante de flujo?
 - Se obtiene la constante de flujo desde el formulario
 - Se valida formato de constante de flujo
 - ¿Si el formato es válido?
 - Se envía la información al Medidor de Consumo EA
 - Si no
 - Se muestra una alerta “El formato de constante es incorrecto”
- ¿Si es etiqueta de localización?
 - Se obtiene la etiqueta de localización desde el formulario
 - Se valida formato de etiqueta de localización
 - ¿Si el formato es válido?
 - Se envía la información al Medidor de Consumo EA
 - Si no
 - Se muestra una alerta “El texto es demasiado grande”

Valido formato

- ¿Si es fecha?
 - Se valida con la expresión regular $\d{2}\d{2}\d{2}$
- ¿Si es hora?
 - Se valida con la expresión regular $(0[0-9]|1\d|2[0-3]):([0-5]\d):([0-5]\d)$
- ¿Si es constante de flujo o constante de consumo eléctrico?
 - Se valida con la expresión regular $\d+(\.\d{1,3})?$
- ¿Si es etiqueta de localización?
 - Se valida con la expresión regular $(\u|s){1,18}$
- ¿Si se validó la expresión regular y no es una cadena vacía?
 - El formato es válido
- Si no
 - El formato no es válido

Solo bastaría mencionar que las constantes de proporcionalidad y etiqueta de localización muestran el valor que tienen asignado previamente. Mediante el uso del enlace básico de datos dentro del formulario. Como se muestra a continuación:

<input id="cAgua" name="cAgua" type="text" value="~cAgua~">

3.3 PÁGINA DE CONFIGURACIÓN CORRECTA

Esta se muestra (Figura 3.8) después de realizar alguna modificación correcta dentro de la página de configuración. Es una simple página HTML.



Figura 3.8 Página de configuración correcta

3.4 PÁGINA DE CONFIGURACIÓN DE RED

En ella se encuentran las opciones configurables en relación a conectividad de red del dispositivo (Figura 3.9). A ella se accede desde la página de configuración,



Figura 3.9 Página de configuración de red

presionando el ícono de computadoras conectadas en los botones de navegación o colocando la dirección en algún navegador.

Esta tiene implementada seguridad para que no se pueda modificar la configuración por cualquier usuario. Si anteriormente se ha autenticado el usuario en la página de configuración no se requiere nuevamente.

En ella (Figura 3.9) se presentan los siguientes elementos:

1. Botones de navegación para regresar a la configuración básica (engrane) o ir al inicio (casa).
2. Etiqueta de identificación de la página.
3. Opciones de configuración de red.
4. Letrero de información.

Las opciones de configuración incluyen:

- Dirección MAC
- Nombre de "Host"
- Activar/Desactivar servicio de DHCP
- Dirección IP
- Gateway
- Máscara de sub red
- DNS primario
- DNS alternativo

En la sección 1.2.3 se menciona la función de los parámetros que se pueden configurar.

El formulario tiene la peculiaridad de que si se tiene activa el recuadro de activación de DHCP no se podrá modificar la Dirección IP, Gateway, Máscara de sub red, DNS primario y DNS alternativo. Esto es porque al tener activo este servicio no es necesario modificar estos parámetros pues se configuran de manera automática. En caso contrario estas estarán disponibles para modificar según las necesidades del usuario.

El recuadro de activación tiene asociada una variable para mostrar el estado que se tenía anteriormente, además de una función que utiliza JavaScript la cual, al cambiar el estado del recuadro de activación esta busca y cambia la propiedad de los elementos mencionados anteriormente y los deshabilita o habilita según sea el caso.

Todos los demás elementos del formulario igualmente tienen una variable enlazada, con una variable interna del microcontrolador para ver su último valor.

Tal y como se ha mostrado en secciones pasadas se usa el enlace básico dentro del formulario tal y como se muestra a continuación:

```
<input type="text" name="ip" value="~config_ip~" />
```

3.5 PÁGINA DE REINICIO DEL DISPOSITIVO

Al realizarse alguna modificación dentro de la página de configuraciones de red y presionar guardar. Se redirigirá a esta página (Figura 3.10) mientras el dispositivo se reinicia. En la página se muestra un “link” que contiene el nombre de host del dispositivo. Si el enrutador o computadora tienen el servicio de DNS activado al presionar el “link” se mostrará la página del dispositivo en su nueva ubicación. En caso contrario se hace necesario utilizar el programa de Microchip “TCPIP Discoverer” para encontrar su nueva dirección IP.

Esta página solo tiene una variable que se encuentra enlazada con el microcontrolador, se trata del “link” que tiene el nombre de host. En este caso se usa el enlace simple. Todo lo demás se trata de una simple página HTML.



Figura 3.10 Página de reinicio del dispositivo

3.6 ARCHIVO DE TRANSFERENCIA

Se creó con la finalidad de que sea consultado por un cliente, el cual recaba la información que el dispositivo genera. La información se actualiza según el intervalo seleccionado de muestreo, la información presentada corresponde al último corte (Figura 3.11).



Figura 3.11 Página de archivo de transferencia, vista desde Internet Explorer

La información se presenta en un renglón, con los siguientes elementos:

- Dirección IP
- Fecha
- Hora
- Acumulado de Fase 1A
- Acumulado de Fase 1B
- Acumulado de Fase 1C
- Acumulado de Fase 2A
- Acumulado de Fase 2B
- Acumulado de Fase 2C
- Acumulado de Fase 3A
- Acumulado de Fase 3B
- Acumulado de Fase 3C
- Acumulado de Flujo 1
- Acumulado de Flujo 2
- Acumulado de Flujo 3

Cada uno separado por un tabulador del siguiente dato, a excepción del último que tiene un salto de línea.

En caso de que no se tenga activa la línea (compuesta por tres fases) o flujo se mostrará un signo de “?” en lugar de la lectura.

La forma de acceder a esta información es colocar la dirección del dispositivo en el navegador añadiendo “/datos.tsv”.

Petición de archivo de transferencia (Figura 3.12).

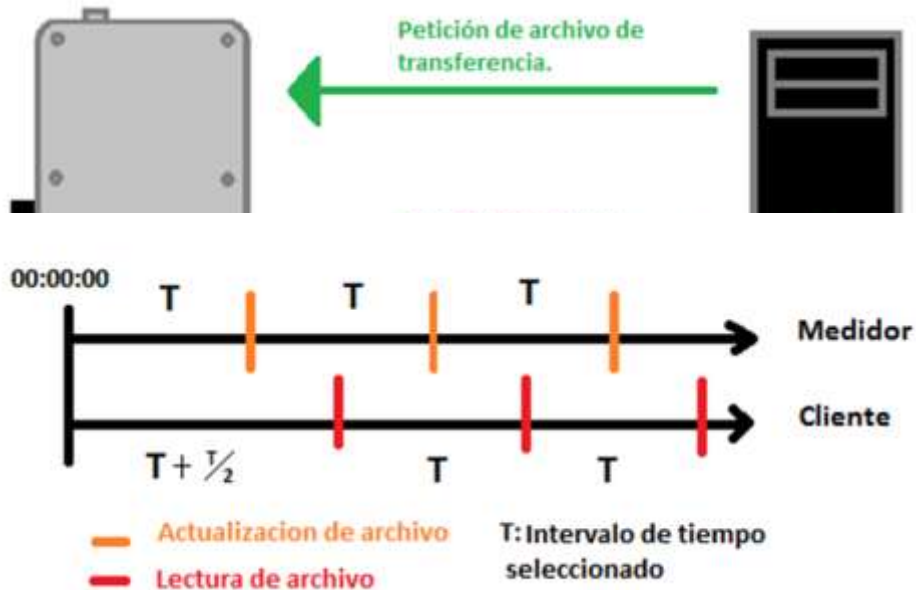


Figura 3.13 Diagrama de tiempos de consulta de archivo de transferencia

En el caso de la interrogación o consulta del cliente, se tendrá que desarrollar en este un programa que de igual forma mediante intervalos realice la petición del archivo de transferencia, se debe tener una sincronía como se muestra en la figura 3.13, pues la información del último corte solo está presente el tiempo que tarda en realizar uno nuevo.

El tener este archivo posibilita que un cliente pueda pedir información a varios medidores de consumo EA. Dentro del cliente se puede tener un programa que analice las lecturas y realice estadísticas más complejas, como consumo total de varios medidores, promedios, etc.

4 RESULTADOS

Como anteriormente se ha mencionado el desarrollo del Software del Medidor de Consumo EA se realizó de forma evolutiva, primero en pruebas dentro del laboratorio y cuando se alcanzó un grado de desarrollo se procedió a la instalación de un medidor en el tablero de distribución del área correspondiente a la coordinación de electrónica, ubicada en el primer piso del edificio 12 del Instituto de Ingeniería de la UNAM (Figura 4.1).

Cumplida la mayoría del desarrollo se colocó un segundo Medidor de Consumo EA en la acometida principal del edificio cinco de la misma institución (Figura 4.2).

El Software que se desarrolló le permite al Medidor de Consumo EA una serie de funcionalidades y características que se mencionarán a continuación y se finaliza con una breve comparación con otros productos en el mercado que tienen una funcionalidad parecida.



Figura 4.1 Medidor de Consumo EA instalado en el edificio 5 del Instituto de Ingeniería de la UNAM



Figura 4.2 Medidor de Consumo EA instalado en el primer piso de edificio 12 del Instituto de Ingeniería de la UNAM

4.1 REGISTRO DE MUESTRAS Y DESPLIEGUE DE RESULTADOS

Se presenta la información recabada por el Medidor de Consumo EA (Figura 4.3), en un intervalo de 00:00:00 H a 11:50:00 H, del día 17 de noviembre de 2015, el cual se encuentra instalado en el edificio 12.

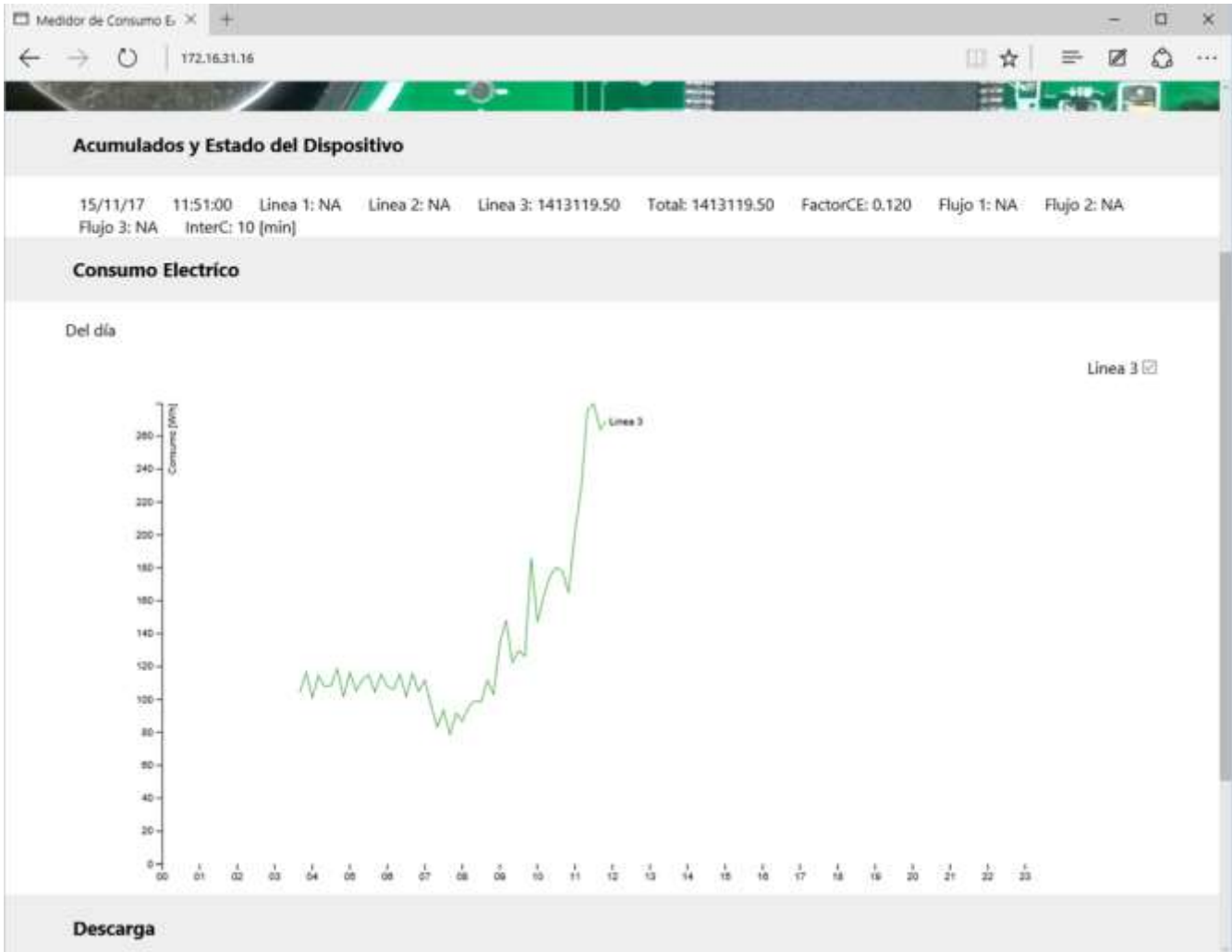


Figura 4.3 Página de inicio del Medidor de Consumo EA instalado en el edificio 12 del IIngen UNAM

Se pueden observar varias peculiaridades:

- El medidor solo tiene activa la línea 3.
- Los datos graficados no inician desde 00:00 H ya que ocurrió un fallo de energía, el cual terminó entre las 03:00 H y las 04:00 H. Con lo que el sistema empezó a graficar de nuevo, después del fallo.
- Se tiene un factor de consumo eléctrico de 0.12, con lo que cualquiera de los acumulados de consumo eléctrico se multiplica por este factor.
- Se tiene un intervalo de muestreo de 10 minutos.

- Podemos deducir algunos escenarios, que en la madrugada (antes de las 06:30:00) el consumo es muy similar con lo que los elementos que consumen energía se mantuvieron igual desde la noche, a la llegada de algunas personas al edificio estas apagan algunos elementos pudieran ser lámparas, ya que la fuente de luz natural ya se encuentra presente. Después de las 09:00:00 H se observa un aumento de consumo con lo que podemos identificar que se está llevando mayor actividad dentro del edificio.

Observaremos a detalle los datos generados descargando el archivo correspondiente a este intervalo de tiempo (Tabla 5).

Tabla 5 Datos utilizados para generar gráfica en página de inicio del Medidor de Consumo EA

Fecha	Hora	Fase_3A	Fase_3B	Fase_3C
17/11/2015	03:40:00	35.88	37.68	30.6
17/11/2015	03:50:00	36	37.8	42.6
17/11/2015	04:00:00	36.12	37.56	27.36
17/11/2015	04:10:00	35.76	36.48	42.36
17/11/2015	04:20:00	35.64	37.8	34.44
17/11/2015	04:30:00	35.76	37.8	34.92
17/11/2015	04:40:00	36	37.92	44.64
17/11/2015	04:50:00	36	38.04	27.36
17/11/2015	05:00:00	36.12	38.04	42.24
17/11/2015	05:10:00	36.12	38.04	31.2
17/11/2015	05:20:00	36.24	37.68	37.92
17/11/2015	05:30:00	38.4	37.68	39.12
17/11/2015	05:40:00	36.96	37.68	29.88
17/11/2015	05:50:00	36.6	37.8	40.92
17/11/2015	06:00:00	36.24	37.2	34.44
17/11/2015	06:10:00	36.24	36.24	33.72
17/11/2015	06:20:00	35.88	37.8	41.64
17/11/2015	06:30:00	36	38.28	27.36
17/11/2015	06:40:00	36	37.92	41.88
17/11/2015	06:50:00	35.76	38.04	31.32
17/11/2015	07:00:00	35.76	37.8	38.16
17/11/2015	07:10:00	19.68	37.92	38.76
17/11/2015	07:20:00	14.4	38.04	30.84
17/11/2015	07:30:00	13.68	37.92	42
17/11/2015	07:40:00	13.56	37.8	27.36

17/11/2015	07:50:00	13.2	36.48	42
17/11/2015	08:00:00	13.44	39.84	33.6
17/11/2015	08:10:00	13.56	45.36	36.36
17/11/2015	08:20:00	13.68	45.12	40.44
17/11/2015	08:30:00	13.44	47.4	37.68
17/11/2015	08:40:00	13.68	47.4	50.52
17/11/2015	08:50:00	14.28	54	34.92
17/11/2015	09:00:00	14.04	69.72	50.88
17/11/2015	09:10:00	22.56	76.68	48.84
17/11/2015	09:20:00	23.04	56.4	43.08
17/11/2015	09:30:00	23.76	56.28	49.56
17/11/2015	09:40:00	30.24	54.24	41.76
17/11/2015	09:50:00	30.84	104.04	51.12
17/11/2015	10:00:00	28.92	81.24	36.84
17/11/2015	10:10:00	32.64	81.84	47.88
17/11/2015	10:20:00	36.96	94.08	43.8
17/11/2015	10:30:00	40.08	96.36	43.92
17/11/2015	10:40:00	39.48	87.96	50.28
17/11/2015	10:50:00	37.32	87.96	39.6
17/11/2015	11:00:00	39.48	109.92	52.56
17/11/2015	11:10:00	43.92	142.2	42
17/11/2015	11:20:00	75.12	157.8	42.84
17/11/2015	11:30:00	74.28	155.76	49.68
17/11/2015	11:40:00	71.64	152.28	40.44
17/11/2015	11:50:00	65.64	149.64	53.88

Se puede observar en la tabla generada que:

- El primer corte después del fallo de energía es a las 03:40:00 H. Cabe señalar que en el archivo que se guarda en la memoria SD la información aparece completa, desde 00:00:00 H hasta 11:50:00 H, con la excepción de la muestra en donde ocurrió el fallo de energía. Recordar que en la gráfica no se muestran los datos completos desde el archivo guardado en la memoria microSD, tanto por el manejo de archivos dentro del microcontrolador, los tiempos de carga del archivo y la reducida capacidad de memoria disponible.
- Observamos los registros de la tercera línea, la cual está compuesta por tres fases.
- El intervalo entre corte y corte es de 10 minutos.

- Cotejando la gráfica y los datos encontramos que el consumo más alto en el intervalo de prueba se lleva a cabo en el corte de las 11:30:00 H sumando en la línea 279.72 W/h.

4.2 VELOCIDADES DE CAPTURA

Se muestran distintos escenarios para identificar la velocidad máxima de captura de los pulsos que pudieran ser enviados por los cuantificadores de consumo.

Se procedió a conectar el núcleo de comunicación Ethernet a una tarjeta de desarrollo “Texas Instruments Tiva LaunchPad TM4C123GXL” (Figura 4.4), la cual tiene un programa que envía 500 pulsos para verificar que el núcleo los captura.

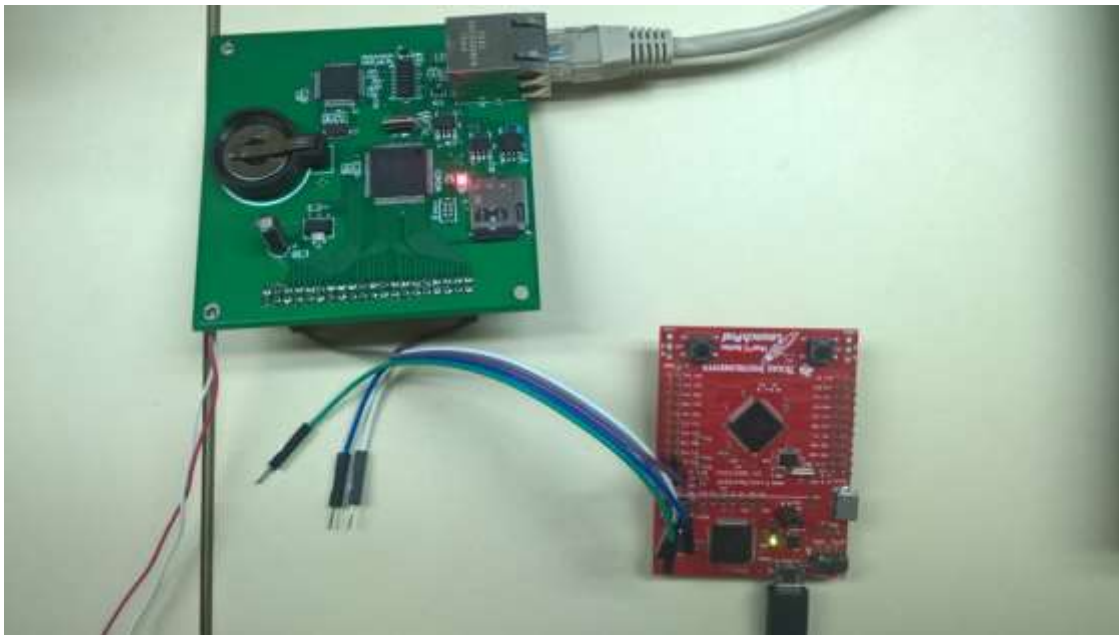


Figura 4.4 Núcleo de Comunicación Ethernet conectado a tarjeta de desarrollo “Texas Instruments Tiva Launchpad TM4C123GXL”

En un principio se verificó la velocidad de los pulsos enviados por la tarjeta Tiva mediante osciloscopio PicoScope identificando los siguientes valores:

Todas con un ciclo de trabajo de 50%.

Señal con frecuencia de 10[Hz] pues el periodo de la señal, como se muestra en la figura 4.5 es de 100[ms].

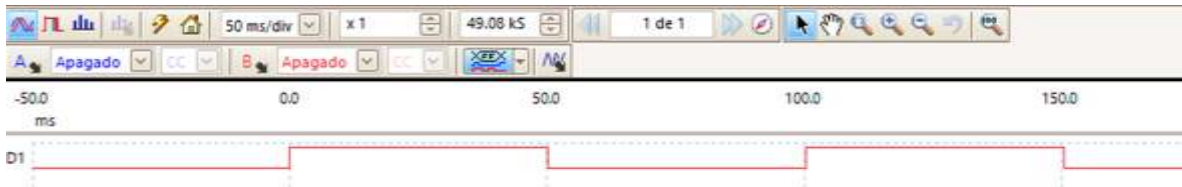


Figura 4.5 Señal con frecuencia de 10[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet

Señal con frecuencia de 20[Hz] pues el periodo de la señal, como se muestra en la figura 4.6 es de 50[ms].

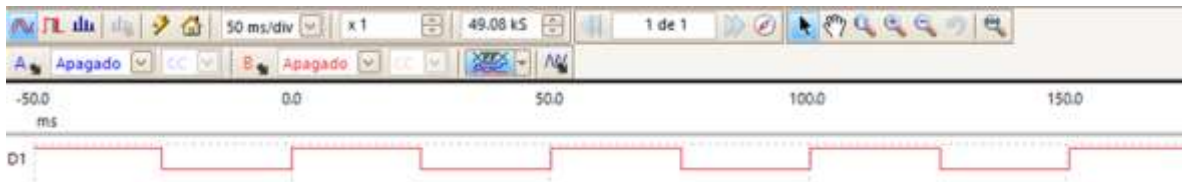


Figura 4.6 Señal con frecuencia de 20[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet

Señal con frecuencia de 50[Hz] pues el periodo de la señal, como se muestra en la figura 4.7 es de 20[ms].



Figura 4.7 Señal con frecuencia de 50[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet

Señal con frecuencia de 100[Hz] pues el periodo de la señal, como se muestra en la figura 4.8 es de 10[ms].

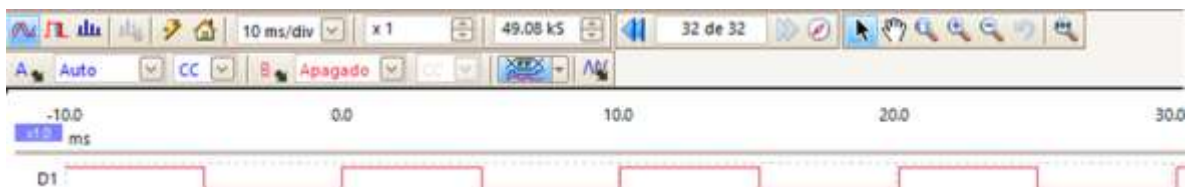


Figura 4.8 Señal con frecuencia de 100[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet

Señal con frecuencia de 500[Hz] pues el periodo de la señal, como se muestra en la figura 4.9 es de 2[ms].

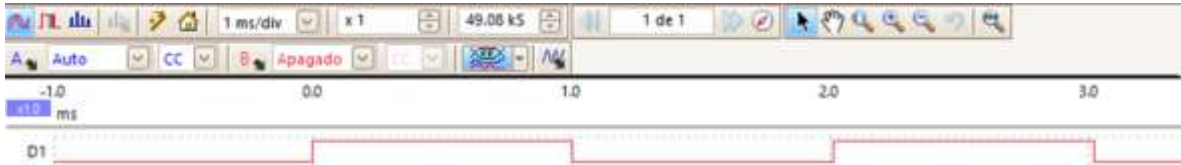


Figura 4.9 Señal con frecuencia de 500[Hz] utilizada para probar velocidad de captura del núcleo de comunicación Ethernet

En una primera fase de la prueba solo se conectó una línea del núcleo a una de la tarjeta de desarrollo Tiva y se ejecutó el programa para comprobar si recibía los 500 pulsos y se cambió la velocidad con respecto a las frecuencias identificadas anteriormente. El resultado se consulta vía página web (Figura 4.10).



Figura 4.10 Página de inicio del Medidor de Consumo EA con resultados de prueba de captura de una línea

Se muestra los resultados obtenidos en la tabla 6.

Tabla 6 Resultados de prueba de captura de una fase

Frecuencia [Hz]	Pulsos recibidos
10	500
100	500
500	250

En la segunda fase de la prueba se mantiene conectada solo una línea y se envían los mismos 500 pulsos, pero ahora mientras se realiza la captura, se descarga de un archivo vía FTP (Figura 4.11). El archivo corresponde a uno generado por el mismo Medidor de Consumo EA con un tamaño de 80128 bytes.

Los resultados obtenidos se muestran en la tabla 7.



Figura 4.11 Página de inicio del Medidor de Consumo EA con resultado de prueba de captura de línea y consola de comandos con resultado de transferencia de archivo vía FTP

Tabla 7 Resultados de prueba de captura de una fase con transferencia de archivo vía FTP

Frecuencia [Hz]	Pulsos recibidos
10	499
100	500
500	210

La tercera fase de la prueba consiste en concretar las 12 entradas del núcleo de comunicación a igual número de salidas de la tarjeta Tiva, la cual envía 500 pulsos por cada salida y cambiar la frecuencia para identificar el número de pulsos capturados (Figura 4.12).



Figura 4.12 Página de inicio del Medidor de Consumo EA con resultados de pruebas de captura de todas las líneas y flujos activos

Se muestran los resultados en la tabla 8.

Tabla 8 Resultado de prueba de captura de todas la líneas y flujos activos

Frecuencia [Hz]	Pulsos recibidos en línea 1	Pulsos recibidos en línea 2	Pulsos recibidos en línea 3	Pulsos recibidos en flujo 1	Pulsos recibidos en flujo 2	Pulsos recibidos en flujo 3
10	1500	1500	1500	500	500	500
20	1467	1472	515	11	9	1
50	1475	21	7	1	3	2

Recuerde que una línea está compuesta por tres fases con lo que si no tuviese pérdidas la lectura seria de 1500.

La cuarta y última fase se envían los 500 pulsos a cada una de las 12 entradas del núcleo de comunicación mientras se realiza la descarga del mismo archivo de la segunda fase (Figura 4.13).



Figura 4.13 Página de inicio del Medidor de Consumo EA con resultados de prueba de captura de todas las líneas y flujos activos, además de consola de comandos con resultado de transferencia de archivo vía FTP

El resultado obtenido se observa en la tabla 9.

Tabla 9 Resultado de prueba de captura con todas las líneas y flujos activos, con transferencia de archivo vía FTP

Frecuencia [Hz]	Pulsos recibidos en línea 1	Pulsos recibidos en línea 2	Pulsos recibidos en línea 3	Pulsos recibidos en flujo 1	Pulsos recibidos en flujo 2	Pulsos recibidos en flujo 3
10	1494	1494	1494	496	496	495

Con estos resultados se identifica que en las primeras dos fases de la prueba la velocidad máxima permitida es de 100[Hz] y no se registran pérdidas significativas. Esto se debe a que solo se está consultando una entrada y el programa del microcontrolador puede atender sin ningún problema, aun si está realizando alguna otra tarea.

Las últimas dos fases presentan el caso extremo, se puede identificar que la velocidad máxima es cercana a los 10[Hz], esta es reducida ya que se consultan todas las entradas y dentro del programa del microcontrolador esto conlleva tiempo, el cual es factor para que se puedan dar casos de pérdidas de pulsos. Si se aumenta la frecuencia de los pulsos ya se presenta una drástica pérdida, pues solo alcanza a verificar la primera entrada, ya cuando es momento de atender a las demás, los pulsos ya han ocurrido.

Al momento de que el microcontrolador atiende otras tareas a esta velocidad (10[Hz]) ya se presenta una pérdida de datos pequeña pero tolerable. En las fases se presenta una pérdida de 2 pulsos de cada 500 o bien menos del 1% y en los flujos, 5 pulsos de cada 500 o bien 1%.

La frecuencia máxima de los contadores incluidos en la tarjeta para la medición de consumo eléctrico y agua es de 11 [Hz]. Esta frecuencia solo es alcanzada cuando se presenta un consumo alto y depende directamente de los convertidores de corriente utilizados para hacer la medición. Con lo que aun en la velocidad máxima se tienen muy poca pérdida de pulsos.

4.3 INTERFAZ WEB ADAPTABLE

Al utilizar HTML5 junto con CSS (hojas de estilos) se logró el desarrollo de una interfaz web, compatible con los diversos navegadores, que se adapta al tamaño del navegador y se presenta de forma correcta en dispositivos móviles.

Se muestra a continuación la interfaz en diversos navegadores (Figura 4.14).

En la parte superior y al fondo de la captura de pantalla se muestra el navegador “Mozilla Firefox 42” y en la parte inferior y al frente a el navegador “Microsoft Edge”.

Como se puede apreciar la página principal del Medidor de Consumo EA se muestra correctamente en ambos navegadores en una computadora personal (PC) que en este caso tiene como sistema operativo “Windows 10”.

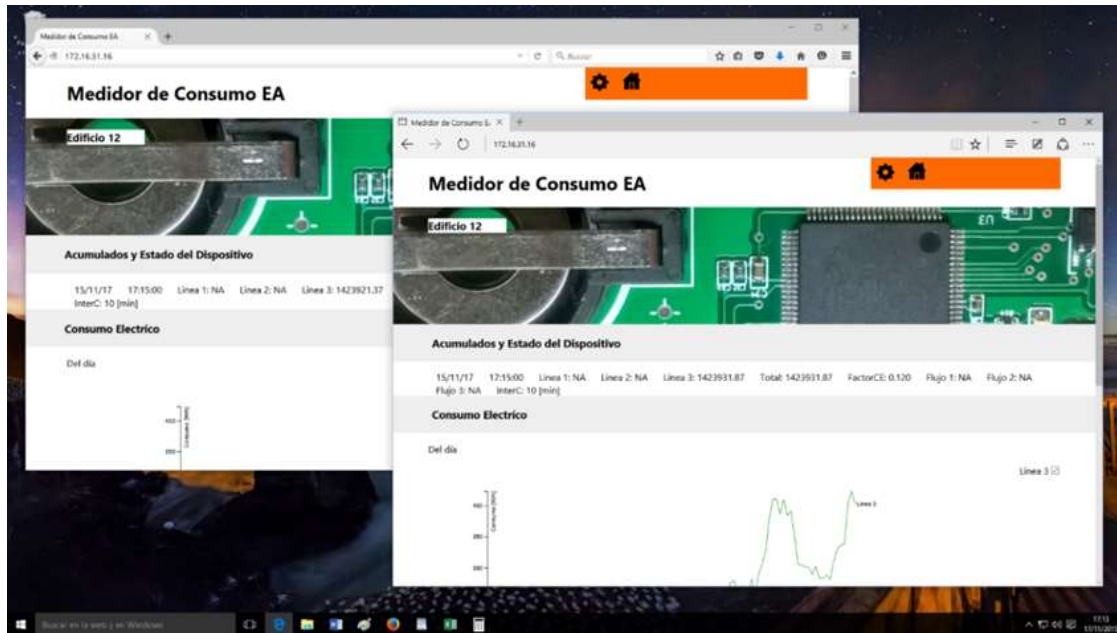


Figura 4.14 Página de inicio de Medidor de Consumo EA visualizado desde “Mozilla Firefox” (Parte superior al fondo) y “Microsoft Edge” (Parte inferior y al frente)

Ahora la adaptabilidad a diversos tamaños de ventana (Figura 4.15).

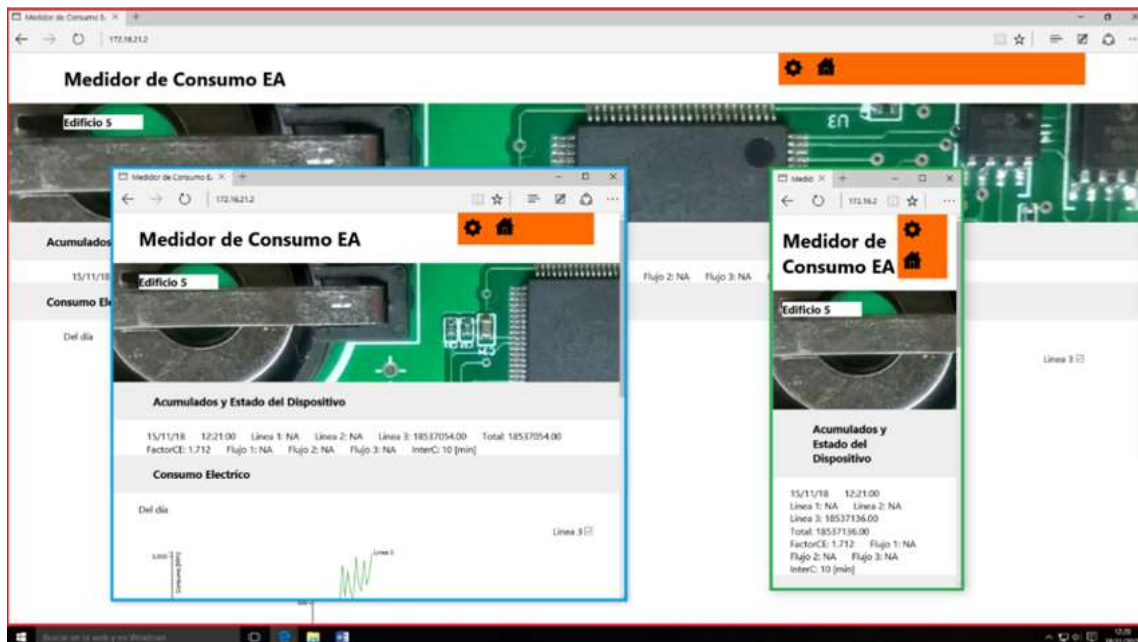


Figura 4.15 Página de inicio del Medidor de Consumo EA vista en navegador tamaño maximizado (rojo), minimizado (azul) y anchura mínima (verde)

Donde se observa el navegador maximizado, minimizado y en anchura mínima. Algunos elementos se reorganizan para su mejor visualización como lo son los botones de navegación y la información de acumulados y estado del dispositivo.

Al igual que los elementos anteriores las gráficas también se ajustan al tamaño del navegador (Figura 4.16).

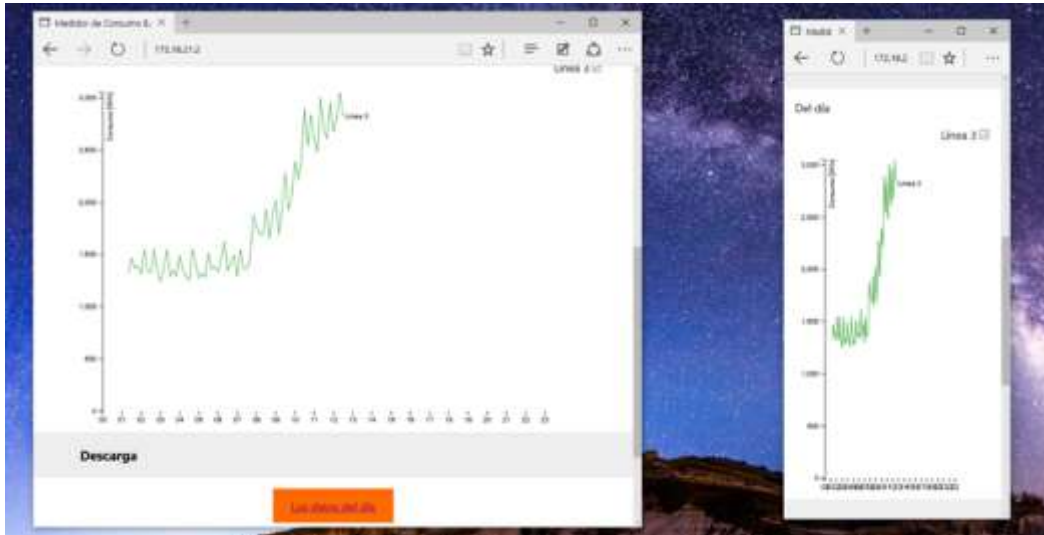


Figura 4.16 Gráfica de la página principal del Medidor de Consumo EA que se adapta al navegador. A la izquierda el navegador minimizado y a la derecha con anchura mínima

Ahora en dispositivos móviles (Figura 4.17).

Donde se destaca la visualización de las gráficas, que no utilizan complementos. Con lo que sin ningún problema se visualizan en cualquier tipo de dispositivo móvil.

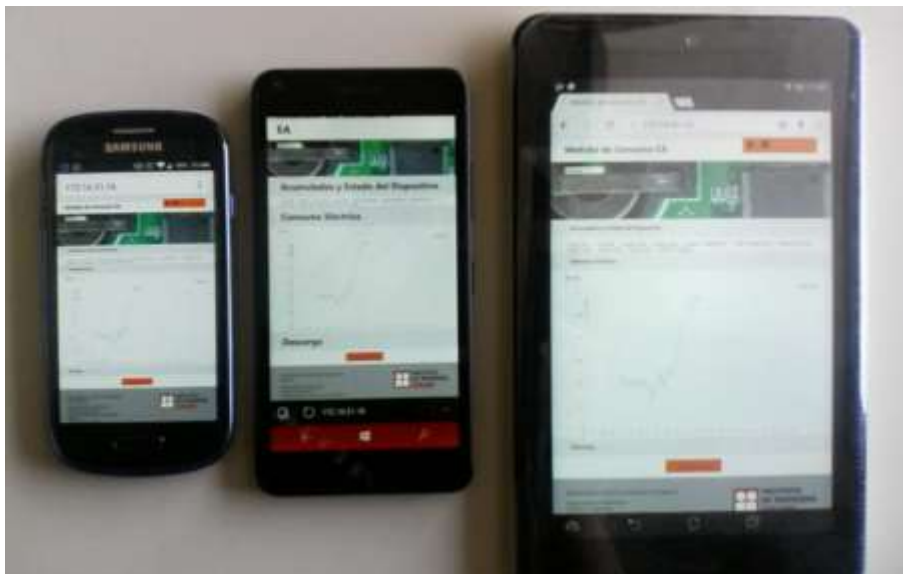


Figura 4.17 Página Web de Medidor de Consumo EA vista desde: "Samsung Galaxy S3 mini" con SO "Android 5.1 Lollipop" (izquierda), "Microsoft Lumia 640 LTE" con SO "Windows 10 Mobile" (centro) y "ASUS MEMO Pad HD 7" con SO "Android 4.2 Jelly Bean" (derecha)

4.4 TRÁFICO EN LA RED

En el desarrollo de la página web se tomó en cuenta la generación mínima de tráfico en la red, ya que el ancho de banda es un recurso limitado que debe ser usado de manera eficiente.

Se muestra a continuación una captura de pantalla del monitor de tráfico de red (Figura 4.18) que se genera al consultar un Medidor de Consumo EA.

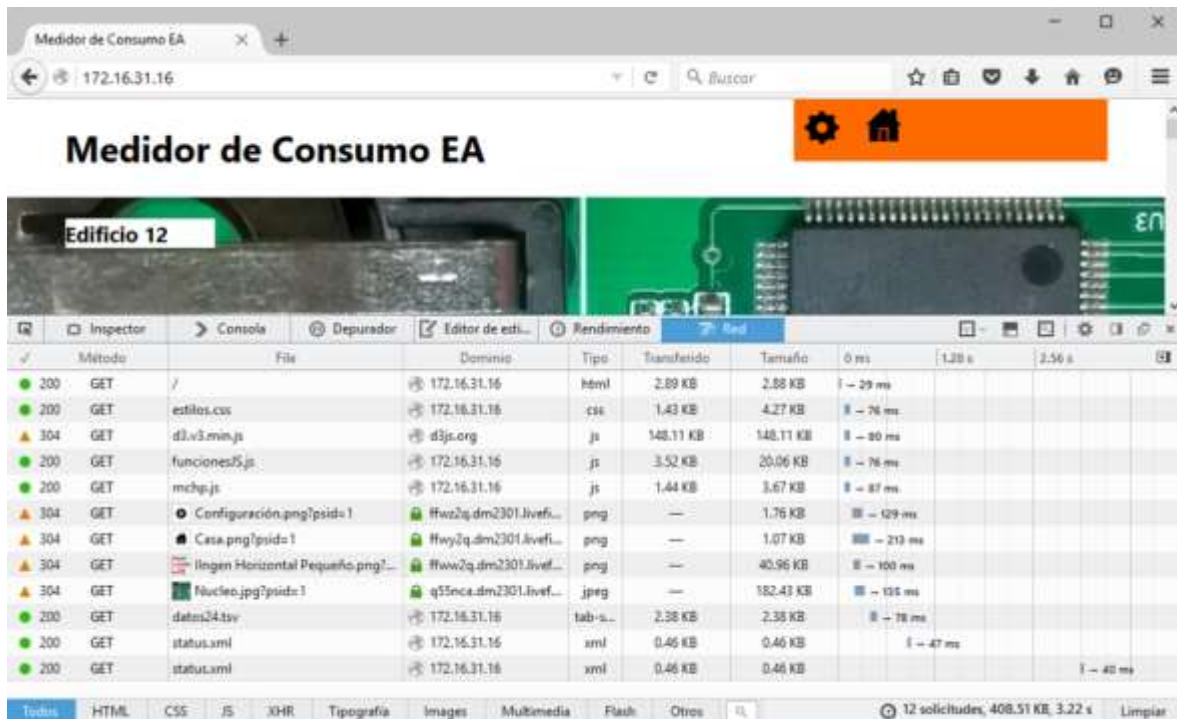


Figura 4.18 Tráfico de red generado al consultar la página de inicio del Medidor de Consumo EA

En la figura 4.18 se observa qué archivos son los solicitados para la visualización de la página de inicio, son en total 11. En la primera columna con un ícono de color verde son los archivos obtenidos desde el Medidor de Consumo EA y con un triángulo amarillo aquellos que son obtenidos desde internet.

El archivo “datos24.tsv”, como se ha mencionado anteriormente, se utiliza para generar las gráficas y el archivo “status.xml” se tienen los datos de acumulados y estado del dispositivo. Este último es el que más se requiere pues se ha programado que se actualice cada 30[s] y el de “datos24.tsv” cada minuto.

En la captura anterior, en la parte inferior derecha observamos el consumo el tráfico generado de 408.51 kB, donde el único archivo actualizado es el de “status.xml” el cual se muestra dos veces en la lista. Este es el tráfico generado al consultar o actualizar la página.

Al visualizar la página por cierto tiempo, en este caso veremos la figura 4.19.

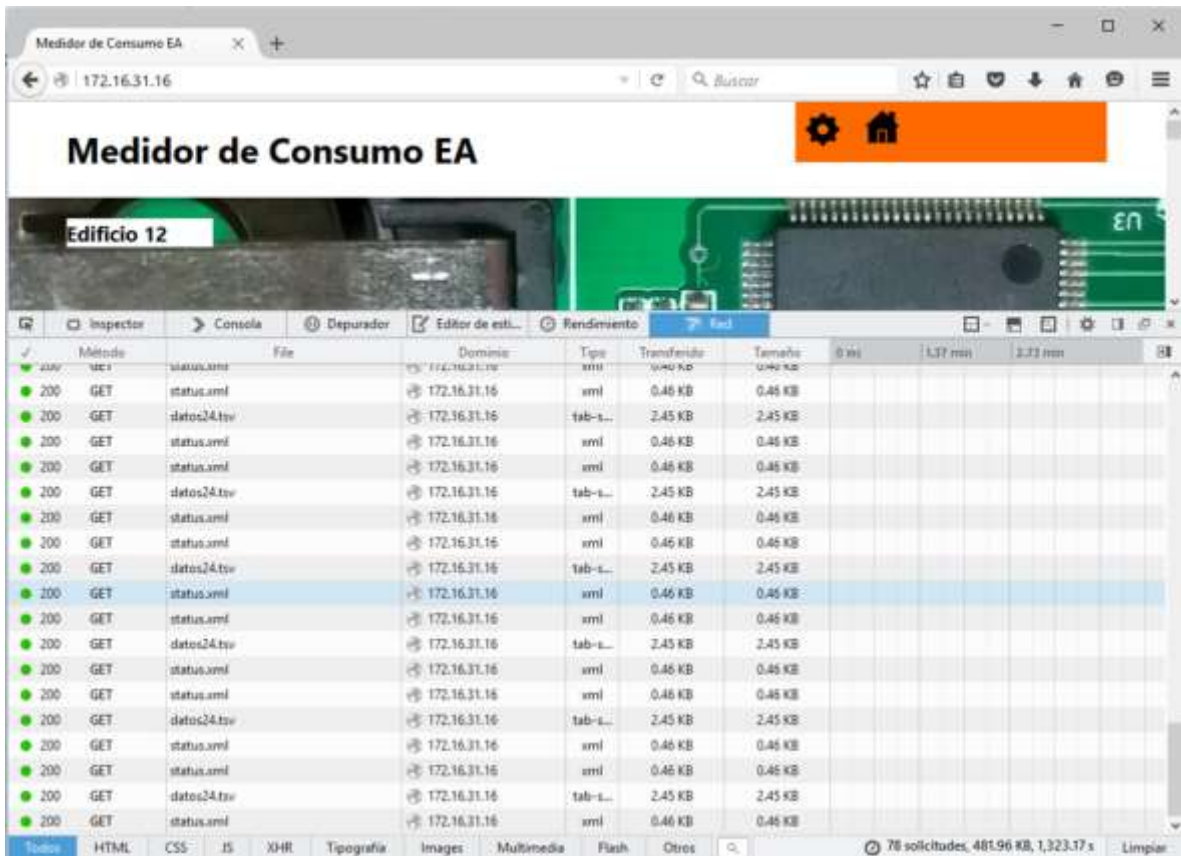


Figura 4.19 Tráfico generado en un intervalo de 22 minutos visualizando la página del Medidor de Consumo EA

Se puede observar que como anteriormente se ha mencionado solo los archivos “datos24.tsv” y “status.xml” son actualizados, también en la parte inferior derecha se observa el tráfico generado en 22 minutos, el cual es de 482kB. Si a este valor le restamos el tráfico cuando se inicia la visualización (409kB) tenemos que en 22 minutos se generó un tráfico de 73kB. A partir de esto podemos obtener que en un minuto se genera 3.4kB. Se debe tener la consideración que en esta prueba solo se tiene una línea activa, el tráfico aproximado que se generaría al tener las otras dos líneas activas y todos los flujos sería, el resultado de multiplicar el valor obtenido por minuto por cuatro (recordar que una línea contiene tres fases, que es equivalente a los tres flujos), lo que nos da por resultado 13.6kB. Este es un tráfico bajo, que si consultáramos vía internet desde una red de datos de uso medido (celular) con un Mega Byte (MB) nos permitiría visualizar la página del Medidor de Consumo EA (con todas las líneas y flujos activos) durante 45 minutos.

Para obtener el resultado de 45 minutos se realizó lo siguiente:

Teniendo en cuenta que:

$$1MB = 1024kB$$

Para consultar la página por primera vez son necesarios 409kB tenemos:

$$1024kB - 409kB = 615kB$$

Los 615kB son los disponibles para la actualización de datos. Si el medidor tiene todos los flujos y líneas activas, por minuto se genera un consumo de 13.6kB de esta forma

$$\frac{615kB}{13.6 \frac{kB}{\text{minuto}}} \approx 45 \text{ minutos}$$

4.5 INTERFAZ WEB SIN CONEXIÓN

El tamaño de la página web que se almacena dentro de la EEPROM no debe ser mayor a 32kB. Esta limitante no permite almacenar las imágenes y el API D3.js, estas son almacenadas en un servidor conectado a internet.

El Medidor de Consumo EA no hace uso de las imágenes y del API D3.js solo las redirige. El navegador web identifica estos recursos y los solicita a su respectivo servidor en el cual se encuentran alojados. Si no se cuenta con conexión estas no se muestran (Figura 4.20).



Figura 4.20 Página de inicio del Medidor de Consumo EA, sin conexión a Internet

4.6 RESCATE DE INFORMACIÓN DE ARCHIVO DE TRASFERENCIA

Mencionado en la sección 3.6, se muestra a continuación como un cliente recolecta la información de los dos Medidores de consumo EA instalados en el Instituto de Ingeniería de la UNAM y a la vez funge como servidor web para publicar los datos recolectados (Figura 4.21). Este fue desarrollado por otro grupo de trabajo perteneciente al mismo instituto, que ejemplifica el uso y utilidad de este archivo.

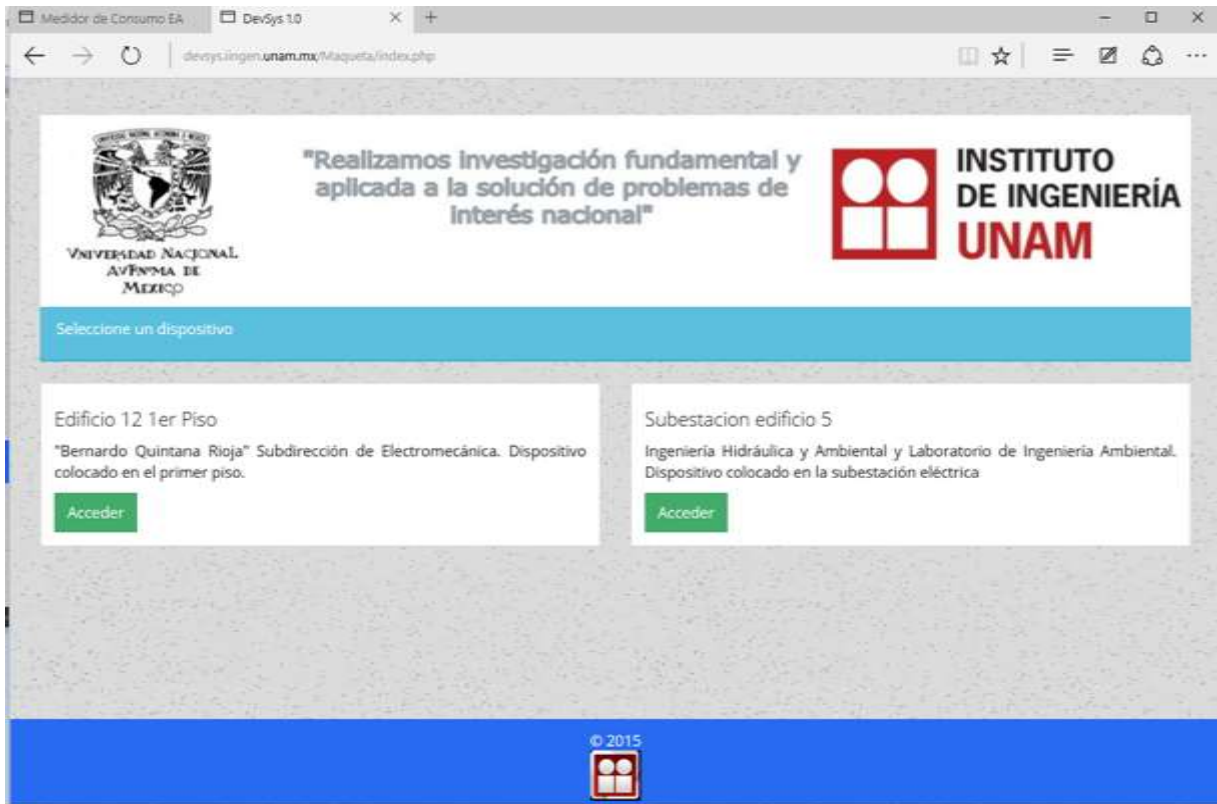


Figura 4.21 Página de inicio de Cliente/Servidor que interroga a los Medidores de Consumo EA instalados en el Instituto de Ingeniería de la UNAM

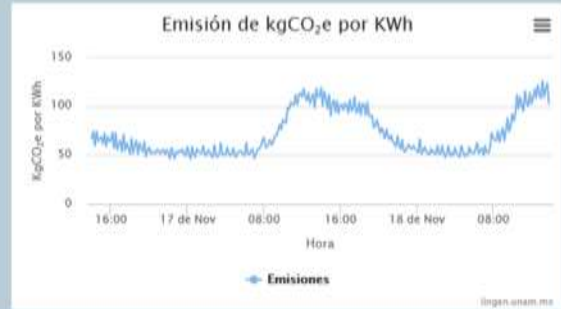
Los datos obtenidos desde los medidores de consumo son registrados y son tratados para generar más información como lo son las emisiones de CO₂ asociadas al consumo eléctrico y el consumo total (Figura 4.22).

Emisiones de dióxido de carbono

Pese al desarrollo de las energías renovables, la generación de electricidad sigue siendo mayoritariamente mediante la quema de combustibles fósiles (carbón, gas o petróleo), que emiten CO₂ y otros gases de efecto invernadero.

En este grupo de gráficas podrá consultar la emisión de kilogramos CO₂ equivalente emitidos a la atmósfera con base en el consumo eléctrico medido por la estación de monitoreo.

[Ver más...](#)



Consumo de energía eléctrica por fase

En este grupo de gráficas se muestra la evolución del consumo eléctrico por cada fase registrada por la estación de monitoreo.



Consumo total

En este grupo de gráficas se muestra la evolución del consumo eléctrico total, calculado con base en el consumo por cada fase registrada por la estación de monitoreo.

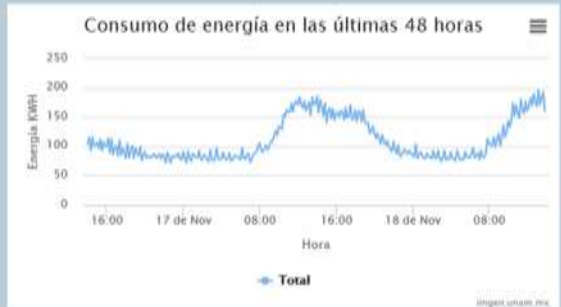


Figura 4.22 Página Web del cliente/servidor que muestra la información generada, a partir de la recolectada en el archivo de transferencia del Medidor de Consumo EA del edificio 5 del Instituto de Ingeniería de la UNAM

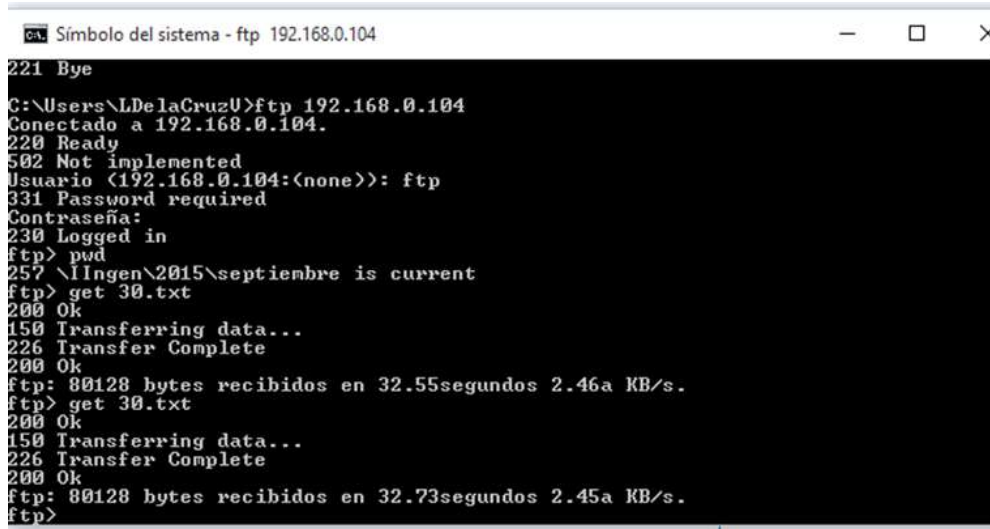
4.7 TRASFERENCIA DE ARCHIVO VÍA FTP

FTP es un servicio que se encuentra activo en el Medidor de Consumo EA para descargar los archivos contenidos en la memoria microSD. A continuación se muestra cómo se ha descargado un archivo utilizando esta capacidad.

Esta transferencia se realiza desde el símbolo del sistema de Windows. Se ejecuta el programa FTP con la dirección del dispositivo, se coloca el usuario y contraseña, se navega por el directorio de la memoria hasta llegar al nivel de interés, para

posteriormente ejecutar la instrucción “GET” más el nombre del archivo para que empiece la transferencia (Figura 4.23).

Cabe mencionar que el medidor debe ser accesible por red para utilizar esta capacidad.



```
Símbolo del sistema - ftp 192.168.0.104
221 Bye
C:\Users\LDeLaCruzU>ftp 192.168.0.104
Conectado a 192.168.0.104.
220 Ready
502 Not implemented
Usuario (192.168.0.104:(none>): ftp
331 Password required
Contraseña:
230 Logged in
ftp> pwd
257 \Ingen\2015\septiembre is current
ftp> get 30.txt
200 Ok
150 Transferring data...
226 Transfer Complete
200 Ok
ftp: 80128 bytes recibidos en 32.55segundos 2.46a KB/s.
ftp> get 30.txt
200 Ok
150 Transferring data...
226 Transfer Complete
200 Ok
ftp: 80128 bytes recibidos en 32.73segundos 2.45a KB/s.
ftp>
```

Figura 4.23 Trasterencia de archivo vía FTP desde el Medidor de Consumo EA

Se puede observar las instrucciones utilizadas para descargar el archivo con nombre 30.txt, con un tamaño de 8018 bytes, cuya tasa de trasterencia fue de 2.46kB/s (Figura 4.23).

4.8 COMPARACIÓN DEL SISTEMA COMPLETO CON OTROS EN EL MERCADO

Se realiza con el fin de tener un panorama general de los productos disponibles en el mercado, que si bien no dan solución a las necesidades del “Programa de medición y ahorro de energía – Metabolismo del Instituto” son sistemas similares y pueden ser tomados como base para asignar un costo total del sistema incluyendo el del desarrollo del Software.

En primera instancia tenemos el dispositivo desarrollado por la empresa española Circuitor cuyo modelo corresponde al “CIRWATT B 405-VT5B-A0B10” el cual tiene las capacidades de: ser un contador trifásico de energía activa y reactiva, con rango de entrada de hasta 400[V] y 5[A], tiene alimentación por batería, comunicación Ethernet y RS232. Con un costo de \$1073 (USD). Es un dispositivo que tiene la ventaja de medir tanto la energía activa como la reactiva, a comparación del Medidor de Consumo EA que solo mide la activa. Sus desventajas

radican en que: solo es para una línea y además no tiene la capacidad de medir flujo.

Ahora un dispositivo desarrollado por otra empresa española, OpenDomo cuyo modelo es “ODEnergy TN5E Trifásico”, que tiene las características de: ser un contador trifásico de energía activa y reactiva, con rango de entrada de hasta 265[V] y la corriente depende del tipo de transformador que se le conecte, comunicación Ethernet y servidor web. Con un costo de \$496 (USD). Al igual que el mencionado anteriormente tiene la ventaja de medir tanto corriente activa como la reactiva, una característica importante en el monitoreo eléctrico que carece el Medidor de Consumo EA. Este producto incluye la posibilidad de consulta de la información en una red local y que envíe su información a un servidor en internet para un monitoreo más detallado (creación de históricos). Es un dispositivo interesante, pero su capacidad de solo una línea y no medir el flujo es lo que más lo diferencia del Medidor de Consumo EA.

Se muestra en la tabla 10, los costos si se necesitara cubrir la capacidad del Medidor de Consumo EA con estos dispositivos.

Tabla 10 Comparación de costo de dispositivos completos disponibles en el mercado para cubrir 3 líneas

Sistema	Costo (USD)	Costo para 3 líneas (USD)
Circuitor CIRWATT	1073	3219
OpenDomo ODEnergy	496	1488

En la tabla 11 se muestra la comparación de funcionalidades de los 3 dispositivos.

Tabla 11 Comparación de funciones

Sistema	Servidor Web	Guardado en Memoria no volátil	Consulta de información desde un servidor externo	Actualización de fecha y hora automática	Descarga de archivos de muestras
Circuitor CIRWATT	Si	Si	Si	No	Si
OpenDomo ODEnergy	Si	Si	Si	No	Si
Medidor de Consumo EA	Si	Si	Si	Si	Si

CONCLUSIÓN

Se ha logrado cumplir con la hipótesis planteada, el software (Firmware y página Web) le ha dado la funcionalidad requerida al Hardware desarrollado en el Instituto de Ingeniería de la UNAM cumpliendo con las necesidades del “Programa de medición y ahorro de energía – Metabolismo del Instituto”. Estamos frente a un producto final que representa un primer desarrollo con esta filosofía, (adquisidores conectados a Ethernet) llevada a cabo en la coordinación de Electrónica. Este desarrollo funcionará como base en siguientes adquisidores, con variados tipos de transductores y variables utilizadas en la instrumentación.

Para el desarrollo se presentaron una serie de dificultades: la poca claridad de la documentación de las librerías TCP/IP y del proyecto de demostración proporcionados por Microchip, identificar cómo funcionaba el proyecto de demostración, demora al momento de la llegada de las tarjetas del Medidor de Consumo EA, algunos errores de lógica al programar, fallos al momento de guardar y consultar información desde la memoria EEPROM, integración de funciones no presentes en las librerías TCP/IP concretamente las del reloj de tiempo real RTCC y almacenamiento en memoria microSD, estas y algunas otras provocaron que el tiempo de desarrollo se extendiera por seis meses, tiempo en el cual una a una fueron superadas.

Con el desarrollo de este proyecto he podido aplicar y consolidar gran cantidad de los conocimientos adquiridos a lo largo de la carrera, los cuales se ven traducidos en algo tangible y funcional, que es y será utilizado en favor del medio ambiente.

En las siguientes secciones se presentan los aspectos técnicos y comerciales que derivaron del desarrollo del Medidor de Consumo EA.

TÉCNICA

Al momento de la búsqueda de herramientas para el desarrollo del Software se encontró con las librerías TCP/IP de Microchip. El poder comprender su estructura y funcionamiento permitió cumplir la hipótesis planteada en esta tesis. Estas librerías TCP/IP son ampliamente recomendables para futuros desarrollos pues a pesar de que su documentación no es muy clara provee a los programadores una estructura base que puede ser adaptable a las necesidades de cada proyecto.

En relación a la página web se pudo realizar una página dinámica que explota las capacidades del enlace de datos, además de la utilización del API D3.js para el despliegue de gráficos vistosos, que junto con HTML5, CSS y JavaScript le dieron

una funcionalidad adaptable y multiplataforma que le dará al usuario una experiencia de uso agradable y con un bajo consumo de datos, pues se tienen los elementos necesarios para la visualización y configuración de forma minimalista.

El dispositivo Medidor de Consumo EA se basa en la lectura de pulsos que son enviados por sus contadores. La frecuencia de los pulsos depende de: los convertidores de corriente utilizados para la medición y el consumo registrado. El programa se basa en la lectura de estos pulsos, pero se pueden presentar situaciones en las que pulsos no sean contados, pues el microcontrolador no se dedica exclusivamente a atender la cuenta de los pulsos sino, además, atiende las peticiones vía Ethernet y almacenar los registros en la memoria microSD y EEPROM. Se identificó mediante las pruebas un rendimiento con un error del 1%, con lo que podemos asegurar que es un sistema que, a pesar de ser mejorable, cumple perfectamente con su cometido.

La posible mejora del sistema sería aquella en donde el microcontrolador no lleve toda la carga y se busque tener elementos que realicen las cuentas de consumo de forma independiente. Para que el microcontrolador incluido en el núcleo de comunicación los interroge solo cuando sea necesario, no como ahora que se cuentan pulsos, que a pesar de que pocas veces se pierden, se puede dar el caso.

Al momento de escribir esta tesis se tienen ya dos dispositivos Medidores de Consumo EA instalados en el Instituto de Ingeniería de la UNAM, funcionando de forma correcta. Se han presentado por la coordinación de Electrónica a posibles interesados que han elogiado el trabajo desarrollado. Lo que ha abierto la posibilidad de desarrollar nuevos modelos.

COMERCIAL

Si bien en la introducción de esta tesis, se mencionó que la tecnología se ha enfocado mayormente al ámbito comercial, también se menciona que se tiene que cambiar este enfoque y buscar la forma en que esta apoye la conservación del planeta. En este caso el desarrollo del Medidor de Consumo EA está completamente justificado, pues con estos se busca obtener datos para una mayor eficiencia de consumo eléctrico y de agua, logrando con ello ahorro de recursos naturales y económicos.

En la sección 1.3.4 se puede observar que el Hardware desarrollado en el Instituto de Ingeniería representa en primer lugar un ahorro, ya que es considerablemente más barato que aquel que es ya comercializado. En segunda instancia se ha adquirido la experiencia para el desarrollo de dispositivos nuevos basado en

comunicación Ethernet. En tercer lugar, estos desarrollos ¡son propios!, con lo que la modificación y adaptabilidad no dependen de terceros.

En las comparaciones mencionadas en la sección 1.3.4 podemos observar también que estamos frente a un segmento de productos dedicados a la adquisición de datos con pocos competidores en el mercado. Donde si bien la comparación con el cRIO es completamente desproporcionada pues este tiene capacidades muy superiores, podemos identificar que no siempre es necesaria tanta potencia de Hardware y es cuando el núcleo de comunicación encaja en aquellos proyectos donde están sobrados ese tipo de sistemas.

Si hablamos ahora de módulos de desarrollo comerciales que contienen microcontroladores solo tendríamos la opción de añadir a estos las tarjetas para expandir sus funciones de adquirentes de datos pero esto tiene una desventaja, el valor agregado al producto reside en solo juntar tarjetas ya diseñadas, en comparación con el desarrollo y diseño total de un sistema que tiene una última y muy importante ventaja: ser patentable.

En el desarrollo del Medidor de Consumo EA se ha identificado este segmento de mercado: adquirentes de datos conectados a Ethernet basado en microcontroladores, con un amplio potencial de comercialización, donde la combinación del núcleo de comunicación, con sus respectivas tarjetas adquirentes, son económicamente viables y donde estos desarrollos son solo una base para futuros productos, con gran potencial.

En la sección 4.8 observamos una comparación con sistemas completos ya comercializados, que si bien son similares no cumplen con los requerimientos solicitados para el “Programa de medición y ahorro de energía – Metabolismo del Instituto”, les faltaría el monitoreo del consumo de agua. Son dispositivos con un costo relativamente alto que dan la posibilidad de asignarle un valor al Medidor de Consumo EA, teniendo como premisa el ser más barato que aquellos que son comerciales.

Si tomamos como referencia el costo del dispositivo “ODEnergy” mencionado en la sección 4.8 que para cubrir la capacidad del Medidor de Consumo EA sería de \$1488 (USD), si tomamos como base el precio del material para armar el Medidor de Consumo EA que fue de \$262(USD) y colocamos un precio final del Medidor de Consumo EA en algo así como \$1300(USD). Se tiene un margen de \$1038(USD) los cuales tienen que ser asignados para cubrir los costos de transporte de los materiales, ensamble, desarrollo de Software y de Hardware Sin duda alguna es un margen bastante atractivo para llevar a cabo la asignación.

El Medidor de Consumo EA actualmente se ocupará para dar solución a los requerimientos del “Programa de medición y ahorro de energía – Metabolismo del Instituto” y su desarrollo fue financiado para no comprar equipos comerciales.

Se debe considerar la posibilidad de ser comercializado.

Bastaría recordar que este dispositivo ayuda al mejor aprovechamiento de los recursos naturales y la inversión que se lleva a cabo es menor a los beneficios a mediano y largo plazo.

APÉNDICES

A) HOJA DE DATOS FAMILIA PIC32MX5XX/6XX/7XX



PIC32MX5XX/6XX/7XX

32-bit Microcontrollers (up to 512 KB Flash and 128 KB SRAM) with Graphics Interface, USB, CAN, and Ethernet

Operating Conditions

- 2.3V to 3.6V, -40°C to +105°C, DC to 80 MHz

Core: 80 MHz/105 DMIPS MIPS32® M4K®

- MIPS16e® mode for up to 40% smaller code size
- Code-efficient (C and Assembly) architecture
- Single-cycle (MAC) 32x16 and two-cycle 32x32 multiply

Clock Management

- 0.9% internal oscillator
- Programmable PLLs and oscillator clock sources
- Fail-Safe Clock Monitor (FSCM)
- Independent Watchdog Timer
- Fast wake-up and start-up

Power Management

- Low-power management modes (Sleep and Idle)
- Integrated Power-on Reset, Brown-out Reset
- 0.5 mA/MHz dynamic current (typical)
- 41 µA IPD current (typical)

Graphics Features

- External graphics interface with up to 34 Parallel Master Port (PMP) pins:
 - Interface to external graphics controller
 - Capable of driving LCD directly with DMA and internal or external memory

Analog Features

- ADC Module:
 - 10-bit 1 Msps rate with one Sample and Hold (S&H)
 - 16 analog inputs
 - Can operate during Sleep mode
- Flexible and independent ADC trigger sources
- Comparators:
 - Two dual-input Comparator modules
 - Programmable references with 32 voltage points

Packages

Type	GFN		TQFP		TFBGA	VTLA
Pin Count	64	64	100	100	121	124
I/O Pins (up to)	51	51	83	83	83	83
Contact/Lead Pitch	0.50	0.50	0.40	0.50	0.80	0.50
Dimensions	9x9x0.9	10x10x1	12x12x1	14x14x1	10x10x1.1	9x9x0.9

Note: All dimensions are in millimeters (mm) unless specified.

Timers/Output Compare/Input Capture

- Five General Purpose Timers:
 - Five 16-bit and up to two 32-bit Timers/Counters
- Five Output Compare (OC) modules
- Five Input Capture (IC) modules
- Real-Time Clock and Calendar (RTCC) module

Communication Interfaces

- USB 2.0-compliant Full-Speed OTG controller
- 10/100 Mbps Ethernet MAC with MII and RMII interface
- CAN module:
 - 2.0B Active with DeviceNet™ addressing support
- Six UART modules (20 Mbps):
 - Supports LIN 1.2 protocols and IrDA® support
- Up to four 4-wire SPI modules (25 Mbps)
- Up to five I²C modules (up to 1 Mbaud) with SMBus support
- Parallel Master Port (PMP)

Direct Memory Access (DMA)

- Up to eight channels of hardware DMA with automatic data size detection
- 32-bit Programmable Cyclic Redundancy Check (CRC)
- Six additional channels dedicated to USB, Ethernet and CAN modules

Input/Output

- 15 mA or 10 mA source/sink for standard VOH/VOL and up to 22 mA for non-standard VOH1
- 5V-tolerant pins
- Selectable open drain and pull-ups
- External interrupts

Qualification and Class B Support

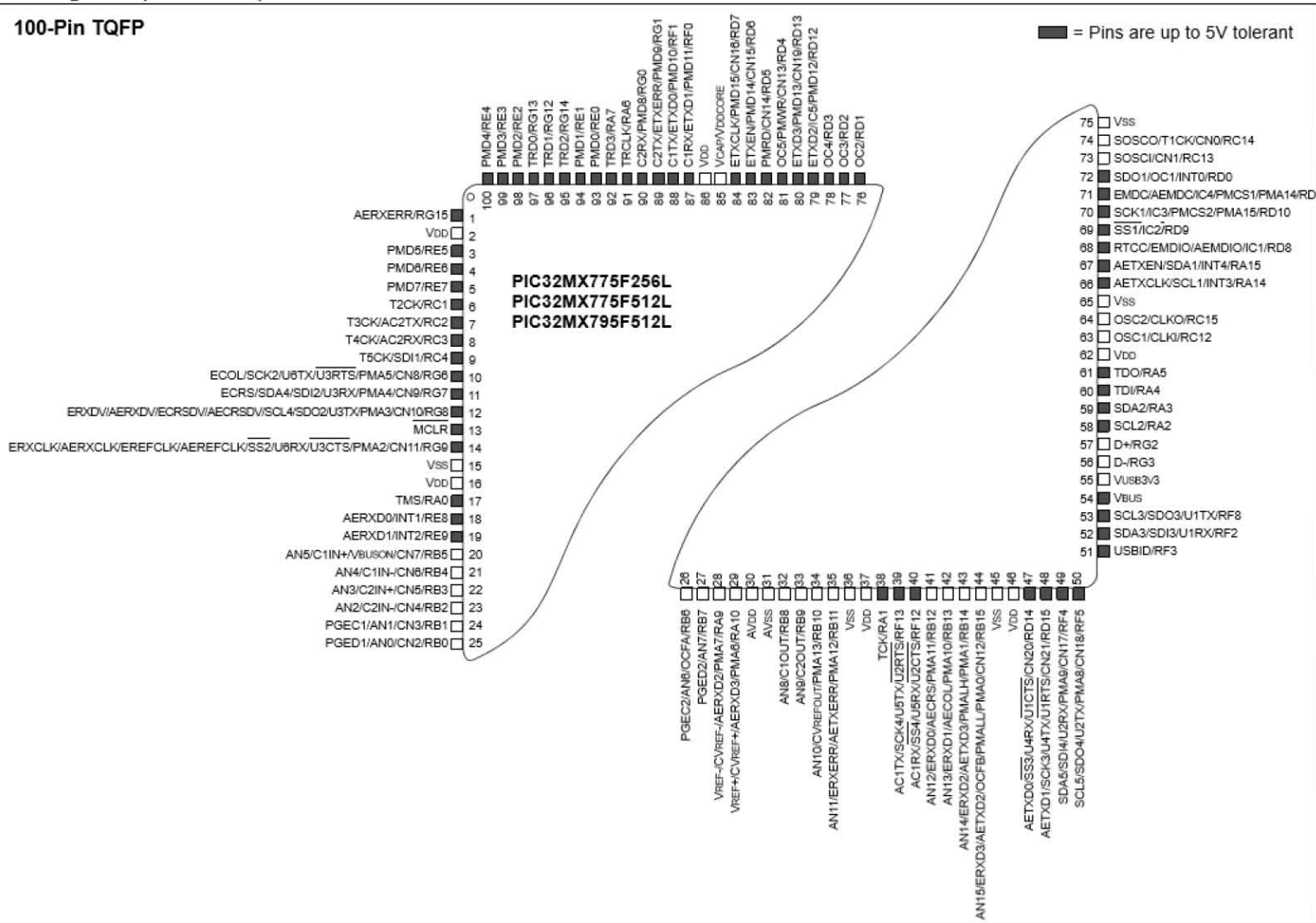
- AEC-Q100 REVG (Grade 2 -40°C to +105°C) planned
- Class B Safety Library, IEC 60730

Debugger Development Support

- In-circuit and in-application programming
- 4-wire MIPS® Enhanced JTAG interface
- Unlimited program and six complex data breakpoints
- IEEE 1149.2-compatible (JTAG) boundary scan

Pin Diagrams (Continued)

100-Pin TQFP



B) HOJA DE DATOS DE CUANTIFICADOR DE CONSUMO ELÉCTRICO ADE7757



Energy Metering IC with Integrated Oscillator

ADE7757*

FEATURES

- On-Chip Oscillator as Clock Source
- High Accuracy, Supposes 50 Hz/60 Hz IEC 521/IEC 61036
- Less than 0.1% Error over a Dynamic Range of 500 to 1
- The ADE7757 Supplies Average Real Power on the Frequency Outputs F1 and F2
- The High Frequency Output CF Is Intended for Calibration and Supplies Instantaneous Real Power
- The Logic Output REVP Can Be Used to Indicate a Potential Miswiring or Negative Power
- Direct Drive for Electromechanical Counters and 2-Phase Stepper Motors (F1 and F2)
- Proprietary ADCs and DSP Provide High Accuracy over Large Variations in Environmental Conditions and Time
- On-Chip Power Supply Monitoring
- On-Chip Creep Protection (No Load Threshold)
- On-Chip Reference 2.5 V (20 ppm/°C Typical) with External Overdrive Capability
- Single 5 V Supply, Low Power (20 mW Typical)
- Low Cost CMOS Process
- AC Input Only

GENERAL DESCRIPTION

The ADE7757 is a high accuracy electrical energy measurement IC. It is a pin reduction version of the ADE7755 with an enhancement of a precise oscillator circuit that serves as a clock source to the chip. The ADE7757 eliminates the cost of an external crystal or resonator, thus reducing the overall cost of a meter

built with this IC. The chip directly interfaces with the shunt resistor and operates only with ac input.

The ADE7757 specifications surpass the accuracy requirements as quoted in the IEC 61036 standard. The AN-679 Application Note can be used as a basis for a description of an IEC 61036 low cost watt-hour meter reference design.

The only analog circuitry used in the ADE7757 is in the Σ - Δ ADCs and reference circuit. All other signal processing (e.g., multiplication and filtering) is carried out in the digital domain. This approach provides superior stability and accuracy over time and extreme environmental conditions.

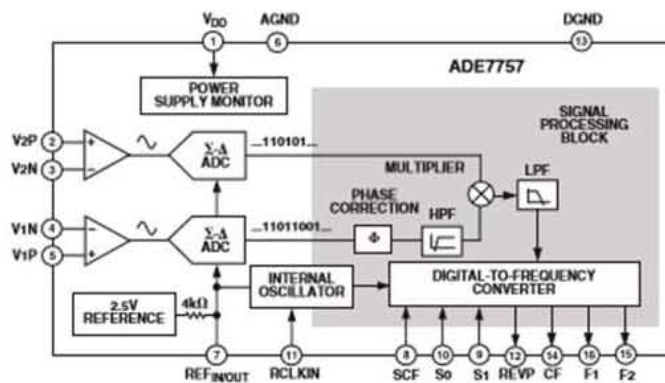
The ADE7757 supplies average real power information on the low frequency outputs F1 and F2. These outputs may be used to directly drive an electromechanical counter or interface with an MCU. The high frequency CF logic output, ideal for calibration purposes, provides instantaneous real power information.

The ADE7757 includes a power supply monitoring circuit on the V_{DD} supply pin. The ADE7757 will remain inactive until the supply voltage on V_{DD} reaches approximately 4 V. If the supply falls below 4 V, the ADE7757 will also remain inactive and the F1, F2, and CF outputs will be in their nonactive modes.

Internal phase matching circuitry ensures that the voltage and current channels are phase matched while the HPF in the current channel eliminates dc offsets. An internal no-load threshold ensures that the ADE7757 does not exhibit creep when no load is present.

The ADE7757 is available in a 16-lead SOIC narrow-body package.

FUNCTIONAL BLOCK DIAGRAM



*U.S. Patents 5,745,323; 5,760,617; 5,862,069; 5,872,469.

REV. A

Information furnished by Analog Devices is believed to be accurate and reliable. However, no responsibility is assumed by Analog Devices for its use, nor for any infringements of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of Analog Devices. Trademarks and registered trademarks are the property of their respective owners.

One Technology Way, P.O. Box 9106, Norwood, MA 02062-9106, U.S.A.
Tel: 781/329-4700 www.analog.com
Fax: 781/326-8703 © 2003 Analog Devices, Inc. All rights reserved.

C) CONCEPTOS BÁSICOS DE REDES

Se presentan una serie de conceptos básicos utilizados en redes de computadoras.

“Nodo”: cualquier punto de interconexión en una red.

“Host”: Se considera así a todas las computadoras o dispositivos terminales que estén conectados a una red.

Cliente: una aplicación o dispositivo que es capaz de enviar peticiones y que recibe información de otra aplicación o dispositivo comúnmente llamado servidor, mediante alguna conexión.

Servidor: una aplicación o dispositivo que envía información en forma de respuesta a peticiones de una aplicación o dispositivo comúnmente llamado cliente, mediante alguna conexión.

Norma: Regla o conjunto de reglas que hay que seguir para llevar a cabo una acción.

Protocolo: Conjunto de normas que regulan la forma que de realizarse el intercambio de información entre dispositivos.

Estándar: Producto de Software o Hardware que cumple con determinadas reglas fijadas por acuerdo internacional, nacional o industrial.

Paquete: Es un segmento de datos que será enviado a otro dispositivo a través de red, este tiene un encabezado donde se incluye información variada dependiendo del protocolo por el que se rija. Generalmente dirección destino.

Servicio orientado a conexión: básicamente el usuario del servicio establece una conexión, la utiliza y luego la abandona. En la mayoría de casos los bits llegan en el mismo orden que son enviados. Al iniciar la conexión se lleva a cabo una negociación para determinar el máximo tamaño de transferencia, así como la calidad del servicio.

Servicio no orientado a conexión: los datos incluyen la dirección de destino y si se envían varios estos son independientes con lo que puede ocurrir que datos lleguen retrasados o adelantado.

GLOSARIO

A

algoritmo

Conjunto de reglas bien definidas, ordenadas y finitas que permiten realizar una actividad. · 53

API

Conjunto de subrutinas, funciones y procedimientos que se encuentran disponibles para ser utilizado por otro software con la finalidad de facilitar la programación. · 47, 51, 52, 86, 87, 93

E

EEPROM

Memoria no volátil programable y borrable eléctricamente. · 2, 13, 26, 30, 38, 39, 40, 41, 42, 50, 86, 94

F

fase

Cada uno de los circuitos de una corriente eléctrica alterna. · 47, 48, 57, 58, 77, 78, 80

Firmware

Contenido programable en Hardware. · IX, 5, 34, 36, 47

H

Hardware

Elementos físicos que componen un sistema informático. · 2, 3, 7, 12, 13, 16, 20, 25, 31, 32, 39, 93, 94, 95, 101

HTML

Lenguaje de marcado de hipertexto. · 18, 48, 50, 64, 67

I

IDE

Entorno de desarrollo integrado, proporciona los servicios para que un programador desarrolle Software. · IX, 34

J

JavaScript

Lenguaje de programación interpretado orientado a objetos. · 47, 48, 50, 51, 52, 62, 66, 93

L

línea

Medio físico por el cual se realiza la transmisión de la energía eléctrica. · 28, 32, 47, 48, 51, 53, 56, 57, 61, 68, 72, 74, 75, 77, 79, 80, 85, 91

M

medición

Comparación de un objeto con un patrón, para identificar cuántas veces está contenido el patrón en el objeto. · 1, 5, 25, 26, 27, 28, 31, 32, 81, 90, 93, 95

medidores de caudal

Trasductor encargado de medir el flujo de algún líquido. · 29

memoria no volátil

Aquella en donde si el sistema se desenergiza la información permanece. · 32

memoria volátil

Aquella en la que si se desenergiza el sistema la información se pierde. · 32

N

NMOS

Tipo de semiconductor que se carga negativamente. · 9

S

Software

Conjunto de programas, lenguajes de programación y datos que controlan que un sistema informático funcione y realice determinadas tareas. · IX, 2, 3, 7, 12, 16, 18, 24, 31, 52, 70, 90, 93, 95, 101

W

Windows

Sistema operativo desarrollado por Microsoft. · 19, 82, 89

REFERENCIAS

- Aguayo S., P. (10 de Noviembre de 2004). *Introducción al Microcontrolador*. Recuperado el 25 de Noviembre de 2015, de MCI Electronics:
<http://www.olimex.cl/tutorial/tutorial1.pdf>
- Alfaro Puyou, L. (Septiembre de 2002). *Microcontrolador Z8: Historia*. Recuperado el 25 de Noviembre de 2015, de Universidad Autónoma de San Luis Potosí, Facultad de Ciencias:
<http://galia.fc.uaslp.mx/~cantocar/z8plus/documentos/historia01.pdf>
- Analog Devices, Inc. (2003). *Hoja de datos ADE7757*. Recuperado el 25 de Noviembre de 2015, de Analog Devices:
<http://www.analog.com/media/en/technical-documentation/data-sheets/ADE7757.pdf>
- Barajas, E. (2002). *Breve historia de las redes locales*. Recuperado el 25 de Noviembre de 2015, de Posgrados ITH:
<http://www.angelfire.com/linux/emilio/introduccion.pdf>
- Boylestad, R. L., & Nashelsky, L. (2003). *Electrónica, teoría de circuitos y dispositivos electrónicos*. México: PEARSON EDUCACIÓN.
- Carpeño Ruiz, A., Corredor López, F. J., Pescador del Oso, F., Hernández Bermejo, J., & Nieto Valhondo, J. (5 de Abril de 2010). *Introducción a los Microcontroladores*. Recuperado el 25 de Noviembre de 2015, de OpenCourseWare, Universidad Politécnica de Madrid:
http://ocw.upm.es/tecnologia-electronica/microprocesadores/contenidos/Archivos_Semana_1/introduccion_a_los_microcontroladores.pdf
- Circuitor. (2014). *Ficha Técnica CIRWATT B 405-VT5B-A0B10*. Obtenido de Circuitor.es: http://circuitor.es/docs/FT_Q1_CIRWATT-B410T_SP.pdf
- Henao R., S. I. (17 de Febrero de 2012). *Microcontroladores Vs Microprocesadores*. Recuperado el 25 de Noviembre de 2015, de TIC TAC:
<http://tic-tac.teleco.uvigo.es/perfiles/blogs/microcontroladores-vs-microprocesadores>
- Kurose, J. F., & Ross, K. W. (2010). *Redes de Computadoras: Un enfoque descendente, 5ta Edición*. Madrid, España: PEARSON EDUCACIÓN.
- Microchip Technology Inc. (2009). *32-Bit Language Tools Libraries*. U.S.A.

- Microchip Technology Inc. (2012). *Microchip TCP/IP Stack Help*. U.S.A.
- Microchip Technology Inc. (2013). *Hoja de datos familia PIC32MX5XX/6XX/7XX*. Recuperado el 25 de Noviembre de 2015, de Microchip:
<http://ww1.microchip.com/downloads/en/DeviceDoc/61156H.pdf>
- Microchip Technology Inc. (2013). *MPLAB® XC32 C/C++ Compiler User's Guide*. U.S.A.
- Nobel Media. (2014). *The history of the Integrated Circuit*. Recuperado el 25 de Noviembre de 2015, de The Official Web Site of the Nobel Prize:
http://www.nobelprize.org/educational/physics/integrated_circuit/history/index.html
- Open Domo. (2013). *Guía de instalación del ODEnergy trifásico*. Obtenido de Open Domo Wiki: http://www.opendomo.com/wiki/images/d/d5/ODEnergyT-manual_ES.pdf
- Tanenbaum, A. S. (2003). *Redes de Computadoras, 4ta Edición*. México: PEARSON EDUCACIÓN.