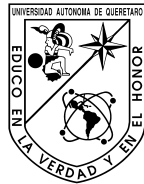


2010 Arquitectura híbrida basada en sistemas embebidos para el monitoreo remoto de largo alcance para estaciones de sensores



Universidad Autónoma de Querétaro  
Facultad de Ingeniería

Tesis  
Que como parte de los requisitos para obtener el grado de  
Doctor en Ingeniería

Arquitectura híbrida basada en sistemas embebidos para el  
monitoreo remoto de largo alcance de  
estaciones de sensores

Presenta

M.C. Manuel Toledano Ayala

Santiago de Querétaro, Qro., a 27 de agosto de 2010



## Portada Interna de Tesis

Universidad Autónoma de Querétaro  
Facultad de Ingeniería  
Doctorado en Ingeniería

ARQUITECTURA HÍBRIDA BASADA EN SISTEMAS EMBEBIDOS PARA EL MONITOREO  
REMOTO DE LARGO ALCANCE DE ESTACIONES DE SENSORES

### TESIS

Que como parte de los requisitos para obtener el grado de

Doctor en Ingeniería

#### Presenta:

M.C. Manuel Toledano Ayala

#### Dirigido por:

Dr. Gilberto Herrera Ruiz

#### SINODALES

Dr. Gilberto Herrera Ruiz  
Presidente

Dra. Rebeca del Rocío Peniche Vera  
Secretario

Dr. Edgar Alejandro Rivas Araiza  
Vocal

Dr. Luis Alfonso Franco Gasca  
Suplente

Dr. Jair García Lamont  
Suplente

Dr. Gilberto Herrera Ruiz  
Director de la Facultad

Dr. Luis Hernández Sandoval  
Director de Investigación y  
Posgrado

Centro Universitario  
Querétaro, Qro.  
Agosto de 2010  
México

## RESUMEN

Actualmente existe una tendencia tecnológica en los sistemas de comunicaciones hacia la eliminación de cables. Para países en vías de desarrollo las comunicaciones inalámbricas representan una gran oportunidad de llegar a zonas rurales o geográficamente de difícil acceso. Los costos de la infraestructura requerida se reducen comparados contra los costos del tendido de cableado. En el presente trabajo, se desarrolló un sistema de telemetría para la medición remota de largo alcance para estaciones de sensores. Se realiza la propuesta de una nueva arquitectura basada en sistemas embebidos la cual minimiza el costo de operación de la red hasta en un 66%. Después de las pruebas en campo, la calidad en el servicio de la transmisión de los datos (QoS) es alrededor del 98%. El sistema ha sido probado con una aplicación práctica que consiste en el monitoreo remoto de variables climatológicas en los altos de Chiapas, por tal motivo se a nombrado Meteo UAQ. Este sistema consta de tres partes principales: a) Unidades Terminales Remotas (RTUs), b) Unidades Terminales Base (BTU) y c) Un servidor central (CS). La adquisición de los datos la realizan las RTUs comunicándose entre ellas para formar una red de área local, inalámbrica en la banda ICM destinada por la COFETEL para aplicaciones industriales, científicas y médicas. Los datos adquiridos son gestionados por las BTUs quienes a su vez, envían esta información vía un enlace satelital a un servidor central ubicado en las instalaciones de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro. Estos datos son almacenados y presentados al usuario final a través de un portal Web, en el que se muestran gráficas de comportamiento diarias y en donde también se pueden descargar historiales en formato de archivo de texto. En este documento, el sistema se describe a detalle, se muestran pruebas de su funcionamiento y se agregan fotografías del desarrollo tecnológico.

**(Palabras clave:** Monitoreo remoto, telemetría, comunicaciones inalámbricas, estaciones meteorológicas, sistemas embebidos)

## SUMMARY

Nowadays, there exists a technological trend in communication systems to wire suppression. For the third world-countries as Mexico, wireless communications represent a great opportunity to reach rural zones that are geographically inaccessible. Costs related to the required infrastructure are reduced compared to those related to wiring. In the present document, a long-range telemetry system for the remote measurement of wireless sensor stations was developed. Also, operation costs are minimized up to 66% due to a new architecture based on embedded systems, which is also presented. After the field tests, Quality of Service (QoS) related to data transmission was found about 98%. The system has been proved with a practical application, which consists on remotely monitoring climatic variables in “Los Altos de Chiapas”, for that reason, the proposed system has been named “Meteo UAQ”. The system is constituted of three main parts: a) Remote Terminal Units (RTUs), b) Base Terminal Units and c) Central Server (CS). Data acquisition is achieved by the RTUs that have the ability to communicate with each other to form a wireless network through the band ISM designated by the COFETEL for Industrial, Scientific and Medical Applications. Acquired data are demanded by the BTUs, which send this information to a central server (by using a satellite link) located at the Facultad de Ingeniería of the Universidad Autónoma de Querétaro. This data are stored and presented to the final user through a Web portal in which daily graphics of the behavior of weather are shown and, in addition, measurement records might be downloaded in form of a text file. In this document the system is described in detail; test of the performance is included as well as photographs of the technological development.

**(Key words:** Remote monitoring, telemetry, wireless communications, weather stations, embedded systems)



## **DEDICATORIAS**

**A mi padre; que con su ejemplo me ha dado la fortaleza que necesito.**

**A mi madre; quien me ha guiado durante todos estos años.**

**A mis hermanas; por ser una motivación para salir adelante.**

**A Martha Lucía Saavedra Rivera, por su apoyo y comprensión en todo momento.**

**Manuel Toledano Ayala**

## **AGRADECIMIENTOS**

A mi asesor Gilberto Herrera Ruiz; por su pasión por la ingeniería, por su congruencia, por su apoyo desinteresado, pero sobretodo, por procurar un mejor futuro para nuestro país y su gente.

Al Dr. Genaro Soto Zarazúa y al Dr. Edgar Rivas Araiza por su apoyo.

Al Consejo Nacional de Ciencia y Tecnología por apoyar mis estudios de posgrado.

A mis profesores, asesores y compañeros por todas sus enseñanzas.

Gracias.

# ÍNDICE

	<b>Página</b>
Resumen	i
Summary	ii
Dedicatorias	iii
Agradecimientos	iv
Índice	v
Índice de tablas	vii
Índice de figuras	ix
1. INTRODUCCIÓN	1
1.1 Descripción del problema	3
1.2 Justificación	4
1.3 Objetivos e hipótesis de trabajo	6
1.3.1 Hipótesis	6
1.3.2 Objetivo general	6
1.3.3 Objetivos particulares	6
2. REVISIÓN DE LITERATURA	8
2.1 Estado del arte	8
2.1.1 Telemetría y telecontrol en sistemas de drenajes en Barcelona, España.	8
2.1.2 Sistema de Telemetría en Irrigación de distritos en Genil-Cabra, España.	10
2.1.3 Sistema de Telemetría para medir señales fisiológicas en animales en libertad, Inglaterra.	12
2.1.4 Diseño de un sistema de monitoreo del agua basado en redes de sensores inalámbricas, China.	15
2.1.5 Sistema de telemetría y telecontrol comercial	16
2.2 Fundamentación teórica	18

2.2.1	Comunicaciones de datos	18
2.2.1.1	Comunicaciones analógicas	18
2.2.1.2	Comunicaciones digitales	20
2.2.2	Configuraciones y topologías de circuitos de comunicación de datos	23
2.2.3	Modelo OSI de ISO	24
2.2.4	Tecnologías en Redes inalámbricas	27
3.	DESCRIPCIÓN DEL SISTEMA	30
3.1	Unidad Terminal Remota (RTU)	31
3.1.1	Diseño de hardware de las RTUs	31
3.1.1.1	Módulo de adquisición y procesamiento (APM)	32
3.1.1.2	Módulo de energía (SPM)	34
3.1.1.3	Módulo de comunicaciones (CM)	35
3.1.2	Software embebido en las RTUs	38
3.1.2.1	Rutina maestra	38
3.1.2.2	Subrutina del módulo de comunicaciones	43
3.2	Unidad Terminal Base (BTU)	45
3.2.1	Hardware de la Unidad Terminal Base	45
3.2.1.1	Interfaz con el módulo de RF	47
3.2.1.2	Interconexión con la red satelital	48
3.2.1.3	Módulo de alimentación de la Unidad Terminal Base	49
3.2.2	Software de la Unidad Terminal Base	50
3.3	Servidor Central (CS)	56
3.3.1	Adquisición remota de datos	57
3.3.2	Portal WEB de acceso a los datos	60
4.	PRUEBAS Y RESULTADOS	64
	CONCLUSIONES	72
	REFERENCIAS	73
	ANEXOS	76

## ÍNDICE DE TABLAS

<b>Tabla</b>	<b>Página</b>
1. División del mercado en sistemas embebidos, en software, hardware y tarjetas (Cifras en millones de dólares).	5
2. División del mercado de sistemas embebidos por zona geográfica.	5
3. Reglas de conversión de los sensores analógicos.	33
4. Reglas de conversión de los sensores digitales.	33
5. Rutina principal de la Unidad Terminal Remota.	39
6. Modo escucha de la Unidad Terminal Remota y comunicación vía inalámbrica con la Unidad Terminal Base.	40
7. Interpretación de comandos, adquisición y transmisión de datos al la Unidad Terminal Base.	41
8. Comandos que interpretan las RTUs.	43
9. Configuración de parámetros en el módulo de RF.	44
10. Definiciones pre-procesador del servidor HTTP embebido.	51
11. Código de la página Web del servidor HTTP embebido.	52
12. Requisición de datos en el servidor HTTP embebido.	54
13. Copiado en memoria de los valores recibidos por el servidor HTTP embebido.	55
14. Configuración de los parámetros de red del servidor HTTP embebido.	55
15. Resultados de las mediciones en campo con las unidades remotas diseñadas.	66

## ÍNDICE DE FIGURAS

<b>Figura</b>		<b>Página</b>
1.	Sistema de tanques de retención de agua en Barcelona.	9
2.	Distrito de Genil-Cabra.	10
3.	Sistema de telemetría de irrigación con el sistema comercial SIGA.	12
4.	Sistema de telemetría para el registro de señales fisiológicas.	13
5.	Diagrama a bloques del sistema de telemetría para señales fisiológicas.	14
6.	Diagrama a bloques del sistema de monitoreo de calidad del agua.	15
7.	Unidad terminal remota comercial.	16
8.	Conexiones de la unidad base ADCON	17
9.	Pantalla de configuración de la red inalámbrica.	17
10.	Modulación en amplitud de una señal senoidal.	19
11.	Modulación en frecuencia de una señal senoidal.	19
12.	Primera representación de ASK.	21
13.	Segunda representación de ASK.	21
14.	Tercera representación de ASK.	21
15.	Transmisión por desplazamiento en frecuencia.	22
16.	Transmisión por desplazamiento en frecuencia.	23
17.	Topologías de red en los sistemas de comunicaciones	24
18.	Modelo para la Interconexión de Sistemas Abiertos de ISO.	25
19.	Panorama general del sistema METEO-UAQ V1.3	30
20.	Módulo del acondicionamiento de señal de la RTU y recursos auxiliares.	34
21.	Parámetros destino y origen en la comunicación inalámbrica.	37
22.	Componentes principales de la Unidad Terminal Base.	46
23.	Servidor HTTP Embebido de sistema Meteo UAQ V1.3.	47
24.	Diagrama de conexiones entre el servidor HTTP embebido y el módulo de RF.	47

25. Ancho de banda requerido por la BTU para la transmisión de los datos meteorológicos.	49
26. Diagrama de conexiones del módulo de alimentación de energía.	50
27. Página Web embebida en el microcontrolador PIC18F6722.	56
28. Diagrama de flujo de la solicitud de transmisión de datos en el servidor central.	58
29. Diagrama de flujo de la rutina de adquisición.	60
30. Ubicación de las estaciones remotas de sensores, caso de aplicación: Los altos de Chiapas.	61
31. Interfaz numérica orientada al usuario final.	61
32. Interfaz gráfica orientada al usuario final.	62
33. Zona de descargas orientada al usuario final.	63
34. Pruebas preliminares de campo en La Ciudad de Querétaro.	64
35. Prueba de rango en laboratorio de las RTUs con el software X-CTU.	65
36. Protocolos más activos durante la transmisión de datos.	67
37. Portal Web del Sistema de Monitoreo Meteo UAQ V1.3	69
38. Pantalla de datos de la estación remota de Agua Prieta, Chiapas.	70
39. Gráficas 24-horas de los datos adquiridos por la estación remota de Agua Prieta, Chiapas.	70
40. Vista de la zona de descargas del sistema Meteo UAQ V1.3.	71
41. Fotografías de las estaciones meteorológicas instaladas.	71





## 1. INTRODUCCIÓN

Actualmente existe una tendencia tecnológica en los sistemas de comunicaciones hacia la eliminación de cables. Los sistemas inalámbricos representan una oportunidad de desarrollo para países en vías de desarrollo como México. Esto debido a que con un mínimo de infraestructura se puede transmitir información a distancia, con costos reducidos comparados con el costo del tendido de cableado. En particular, las tecnologías inalámbricas representan ventajas importantes en el sector productivo tales como la posibilidad de tener nuevos valores agregados como el mantenimiento remoto, configuración, diagnóstico, monitoreo o incluso algunas tareas de control. De igual manera, la reubicación de unidades terminales remotas y la extensión del sistema en general, se realizan de una manera más sencilla.

Existe un gran potencial para el desarrollo de aplicaciones con tecnologías inalámbricas, especialmente sensores inalámbricos y redes de sensores empezando por las aplicaciones militares y ambientales. Una red de sensores inalámbricos es un sistema compuesto de transceptores de Radio Frecuencia (RF), sensores, microcontroladores, y fuentes de poder (Ning, W. et al., 2005). Actualmente se cuenta con redes de sensores inalámbricos con capacidades de auto-organización, auto-configuración, auto-diagnóstico y auto-reparación.

Las tecnologías inalámbricas han tenido un rápido desarrollo en años recientes. Una de las aplicaciones más populares de las tecnologías inalámbricas son las comunicaciones interpersonales como los teléfonos celulares. Sin embargo, en aplicaciones de automatización y monitoreo ambiental existe un gran potencial sobre el uso de estas tecnologías. En algunas áreas específicas existen ambientes los cuales son comúnmente peligrosos o inaccesibles para los humanos o incluso para robots. Así, hay muchos retos para desarrollar y mantener

redes inalámbricas de sensores (WSN) en esos ambientes (Guangming, S. et al., 2008).

Las WSN permiten, entre otras cosas, aplicaciones en las que el monitoreo de las tareas es peligroso, donde no existe la infraestructura con cables o en áreas lejanas o de difícil acceso. La mayoría de las redes de sensores que involucran comunicaciones inalámbricas tienen un acondicionamiento de señal, unidades de procesamiento y señales de portadoras que son codificadas de manera digital y esto ha sido importante para la reducción del ruido comparado con los sistemas de comunicación tradicionales de amplitud modulada (AM).

También, las WSN ofrecen una robustez mejorada y flexibilidad comparada con los sistemas alámbricos y permiten desarrollar aplicaciones las cuales no podrían haber sido consideradas en años anteriores como monitorear lugares peligrosos, sin cables o de difícil acceso.

En la última década, WSN han generado un entusiasmo considerable en la comunidad de investigadores para su aplicación en una gran variedad de aplicaciones como: monitoreo ambiental, aplicaciones de rastreo militar, predicción del clima, automatización del hogar, detección de intrusos, etc. (Nadjib, A. et al., 2009, Chiara, B. et al., 2009, Muhammad, K., et al., 2010, Chong, C. et al., 2003, Kazuya, T. et al., 2009, Xu, N, 2002, Raja, J., et al., 2008, Sardini, E., 2009). Sin embargo, hay algunos puntos importantes a considerar antes de seleccionar la tecnología inalámbrica para la transferencia de datos. Estos puntos son: confiabilidad en la comunicación, rango de transmisión, tasa de transferencia de datos y consumo en potencia. Las WSN comúnmente están clasificadas en términos de rango de transmisión, de las cuales se puede encontrar como redes de corto alcance a: PAN (Red de área personal), Asociación de datos por infrarrojo (IrDA), IEEE 802.15.1 (Bluetooth) e IEEE 802.15.4 (ZigBee). Dentro de las redes de mediano alcance se encuentran las redes de área locales (LAN) y comprenden principalmente a la IEEE 802.11b comúnmente conocida como WiFi la cual es

capaz de transmitir una alta tasa de datos a expensas de un alto consumo de energía. Por parte de las redes inalámbricas de largo alcance se encuentra la comunicación satelital y los sistemas de telefonía celular como GSM/GPRS y CDMA.

El monitoreo remoto de largo alcance y las aplicaciones en automatización demandan estrategias nuevas e innovadoras además de nuevas topologías con la intención de garantizar una calidad en el servicio (QoS) mínima, reduciendo el consumo en potencia y costos de operación. Generalmente, la comunicación a gran distancia es más costosa y está sujeta a interferencias; además, en zonas de difícil acceso no se cuenta con alimentación eléctrica instalada. Esto conlleva a un uso eficiente de la energía y constituye un reto para las WSN (Akyildiz, F. et al., 2002). Especialmente, el conseguir datos para aplicaciones de monitoreo ambiental puede ser difícil cuando esta tarea involucra elementos que se encuentran esparcidos en una área grande y sus posiciones están sujetas a cambiar (Víctor, P. et al., 2009) o existe asociado, un alto consumo de energía (Supriyo C. et al., 2009). En este sentido, la tendencia gira entorno a minimizar el consumo en las WSN para que permitan al sistema ser operado con baterías. En muchas aplicaciones, esta energía es provista con la ayuda de energías provistas por el medio ambiente en donde se encuentra ubicado el dispositivo.

Como respuesta a una tendencia global en las comunicaciones inalámbricas, y con posibilidades de aplicación en el sector ambiental e industrial, nace este proyecto de investigación, en el cual se aborda la propuesta de una arquitectura híbrida basada en sistemas embebidos para el monitoreo remoto de largo alcance de estaciones de sensores.

## **1.1 Descripción del problema**

Las tecnologías inalámbricas han tenido un rápido crecimiento en los últimos años. Algunos ejemplos de ellas son las redes de área personal (WPAN) de corto alcance, comunicaciones punto a multi-punto como Bluetooth y Zigbee hasta

aquellas de rango medio como las redes de área local (WLAN), a aquellas de largo alcance como los sistemas satelitales, GSM/GPRS.

En la actualidad, la mayoría de las soluciones inalámbricas utilizan una tecnología propietaria lo cual quiere decir que se solucionan eficientemente las necesidades de sistemas particulares, pero no se adaptan a los nuevos sistemas.

A pesar de la tendencia en los sistemas de comunicaciones para la eliminación de cables, la naturaleza de las comunicaciones inalámbricas las hace susceptibles a interferencias, por esta razón, los sistemas inalámbricos necesitan de mecanismos para reducir la tolerancia a fallas y aún más, cuando las estaciones remotas se localizan en áreas geográficas que por lo general son de difícil acceso.

En este escenario, la definición de una arquitectura que permita a los usuarios monitorear estaciones remotas a gran distancia que satisfagan las necesidades de confiabilidad en los servicios es imperiosa, además, esto necesita ser conseguido a un costo razonable.

## **1.2 Justificación**

Los sistemas embebidos juegan un papel cada vez más importante en la vida diaria. Ésto se ve reflejado en el tamaño del mercado, que crece a razón de 14% al año, para pasar de \$46,000 millones de dólares en el 2004, a un estimado de \$88,000 millones en el 2009 (fuente: BBC). Este crecimiento está sustentado por el avance tecnológico, y por la aceptación de la población a nuevos y mejores servicios.

Por regla general, el software es más complejo y costoso que el hardware, y por tanto, juega un papel primordial en el desarrollo de los sistemas embebidos. El mercado de software embebido crecerá a una tasa de 16% anual, para pasar de \$1,600 millones de dólares en el 2004 a \$3,500 millones en el 2009. Por comparación, el mercado de hardware embebido crecerá a 14% anual y el de

tarjetas a 10% anual (fuente: BBC). La tabla 1 resume esta información; aquí, “*hardware*” se refiere a circuitos especializados en aplicaciones embebidas, y “*tarjetas*” a los circuitos impresos donde se montan los circuitos.

Tabla 1. División del mercado en sistemas embebidos, en software, hardware y tarjetas (Cifras en millones de dólares). Fuente: BBC.

	<i>2004</i>	<i>2009</i>	<i>% crecimiento</i>
<b>Software</b>	1,641	3,448	16%
<b>Hardware</b>	40,539	78,746	14.2%
<b>Tarjetas</b>	3,693	5,950	10%
<b>Total</b>	45,873	88,144	14%

La industria de los sistemas embebidos se puede clasificar en seis grandes áreas de aplicación: computación, comunicaciones, artículos de consumo, automotriz, de uso médico o de oficina, y de uso industrial o militar. El mercado se reparte como se muestra en la tabla 2.

Tabla 2. División del mercado de sistemas embebidos por zona geográfica. Fuente: BBC.

	<i>2004</i>	<i>2009</i>	<i>% crecimiento</i>	<i>% del total en 2009</i>
<b>América</b>	22,012	36,728	10.8%	41.7%
<b>Europa</b>	7,648	14,038	12.9%	15.9%
<b>Japón</b>	7,394	13,160	12.2%	14.9%
<b>Asia</b>	8,819	24,218	22.4%	27.5%
<b>Total</b>	45,873	88,144	14%	

Como se observa, las comunicaciones representan alrededor del 25% del mercado de sistemas embebidos, y se espera un crecimiento sostenido de 12.2% anual en los próximos años. El mercado de comunicaciones tiene una característica particular que los demás no comparten. Conforme avanza la tecnología, la capacidad de los sistemas embebidos, y las necesidades de los usuarios, distintos mercados empiezan a adquirir necesidades de comunicación.

Desde los sistemas médicos, donde un doctor puede consultar diagnósticos (y hasta realizar operaciones) a distancia, hasta automóviles con conexión a Internet y servicios de localización, la capacidad de comunicarse comienza a ser un requisito de casi cualquier sistema embebido. Así, existe una oportunidad en la que se puedan diseñar sistemas en los que los dispositivos operen bajo distintos estándares, y puedan transitar de una red a otra sin intervención del usuario.

### **1.3 Objetivos e hipótesis de trabajo**

#### **1.3.1 Hipótesis**

La arquitectura del sistema de comunicaciones propuesta, aumenta la QoS en estaciones remotas en una red de largo alcance, donde el monitoreo remoto reduce los gastos operativos y aumenta la eficiencia en la administración de los datos adquiridos.

#### **1.3.2 Objetivo general**

Diseñar e implementar una arquitectura para un sistema de comunicaciones de largo alcance que permita mejorar la QoS en las redes de comunicaciones inalámbricas y reduzca los costos de operación de la red.

#### **1.3.3 Objetivos particulares**

1.- Proponer una arquitectura para el sistema de comunicaciones de largo alcance que permita reducir los errores en la transmisión dentro de una red de sensores inalámbricos.

2.- Diseñar e implementar un sistema de supervisión y control de fácil instalación, mantenimiento y flexibilidad orientado a la detección temprana de fallas.

3.- Diseñar un sistema electrónico de adquisición y comunicación que cumpla con estándares internacionales para su posible transferencia tecnológica en otros países.

4.- Desarrollar un sistema robusto y escalable minimizando grado de dependencia tecnológica para su aplicación en diferentes sectores.

## **2. REVISION DE LITERATURA**

### **2.1 Estado del arte**

Desde la aparición del hipertexto, en 1989, Internet se ha convertido en una plataforma que actualmente utilizan diferentes sectores, como el bancario, el comercial, el educativo, entre otros, dando origen a la banca electrónica, el comercio electrónico y a la educación a distancia, respectivamente. Esto hace que Internet forme parte cada vez más de nuestra vida cotidiana. Los sistemas de control no se podían quedar atrás y también han utilizado esta plataforma dando origen al control por Internet o también llamado Control IP.

Los sistemas de telemetría han tenido múltiples y diversas aplicaciones, que van desde el control de un telescopio en el observatorio de L'Estelot (Grup D'Estudis Agronomics, 2004), hasta el control de toda una planta pasteurizadora (H., Hugo, y J. Rivera, 2004), pasando por el control de servomotores [Magazine electrónico de SCM, 2004], tuberías de emergencia (Automa Inge, 2004), televigilancia (Domótica Viva, 2004), Ahorro de energía (I-Hai, 2001), invernaderos (Serodio, et al., 2001), (Ortiz, et al. 2004), maquetas de invernadero (Guzmán et al., 2004) entre otros. El principal punto en común entre todos los anteriores es que en todos ellos se ve la necesidad de implementar sistemas de monitoreo y control que sean capaces de generar gráficos en tiempo real, control, almacenamiento y generar reportes de las condiciones operativas de cada proceso en un ambiente Windows en una PC de propósito general.

#### **2.1.1 Telemetría y telecontrol en sistemas de drenajes en Barcelona, España**

Los sistemas de drenaje urbanos son generalmente redes de alcantarillas las cuales cargan aguas de desecho urbanas y agua de lluvia hacia uno o más puntos terminales donde es tratada y/o descargada hacia el medio ambiente. En muchas ciudades donde la zona urbana ha crecido y donde las tormentas de lluvia son



frecuentes, los sistemas combinados de alcantarillas son incapaces de conducir toda la lluvia y las aguas de desecho hacia las plantas de tratamiento cuando llueve intensamente. Esto ocasiona inundaciones en ciertas áreas y desbordamiento de las alcantarillas combinadas lo cual provoca que el agua no-tratada se vaya al medio ambiente.

Existe una conciencia creciente en la sociedad por la protección del medio ambiente incluyendo, entre otros temas importantes, la administración del agua en áreas urbanas, esto es, provisión de agua, distribución de agua y la recepción del agua en el medio ambiente.

El sistema de telemetría contiene indicadores de lluvia distribuidos en diferentes áreas de la ciudad y periódicamente manda información a una unidad central. El sistema de telemetría permite a los operadores monitorear la red de alcantarillas y controlar las operaciones de compuertas de tanques de retención que son utilizados cuando llueve en demasía.



Figura 1. Sistema de tanques de retención de agua en Barcelona.  
(Cembrano, G. et al., 2007).

Este sistema piloto se encuentra conectado en 12 tanques, con más de 100 sensores de nivel en las alcantarillas y está funcionando desde 1994. Existen más tanques de retención que se encuentran en construcción para los 1600 Kms de alcantarillas en la ciudad de Barcelona, conteniendo diariamente alrededor de 300,000 metros cúbicos de aguas de desechos y agua de lluvia (Cembrano, G et al., 2007).

### 2.1.2 Sistema de Telemetría en Irrigación de distritos en Genil-Cabra, España

La demanda de agua se está incrementando tanto en cantidad como en calidad; un fenómeno que está condicionado por factores sociales, políticos y del medio ambiente. Las presiones por el tener acceso al agua han generado una competencia por los recursos acuáticos disponibles en los consumidores tradicionales llámense agricultura, industria y ciudades. Estos intereses ya se encuentran limitando el desarrollo social, industrial y rural en muchos países. Además, la creciente demanda de agua no se encuentra sincronizada con el aumento de los recursos ocasionando una mayor competencia por el acceso al agua.

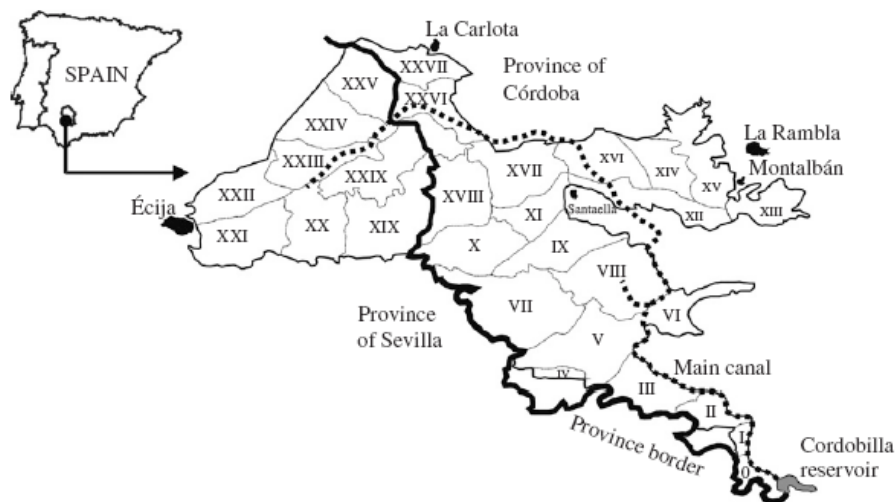


Figura 2. Distrito de Genil-Cabra. (Pulido-Calvo et al., 2007)

El distrito de Genil-Cabra actualmente tiene alrededor de 15 mil hectáreas de riego pertenecientes al río Guadalquivir en entre las provincias de Córdoba y Sevilla al sur de España.

Para determinar el consumo de agua en el distrito de irrigación, se instaló un sistema de telemetría comercial llamado SIGA. Este sistema se instaló en 28 granjas localizadas en la red de irrigación para medir volúmenes y presión de agua y ser monitoreados en tiempo real. El sistema fue instalado en dos etapas, 12 granjas del 2001 al 2002 y 16 más del 2002 al 2003 en la estación de irrigación.

El sistema de telemetría SIGA es un sistema automático que permite adquirir datos recopilados por los sensores, almacenarlos, enviarlos, procesarlos y mostrarlos de manera secuencial. Los datos en cuestión son: Hora y fecha, volúmenes de agua y presiones instantáneas.

Los componentes principales del sistema de telemetría incluyen: a) El nodo de irrigación o estación remota la cual captura señales de los medidores de agua volumétrica y la presión en los hidrantes de riego; b) La estación de concentración la cual recibe y almacena las señales de campo de los nodos de irrigación y c) El sitio de control central el cual despliega y almacena los datos adquiridos por las estaciones de concentración y sus nodos asociados.

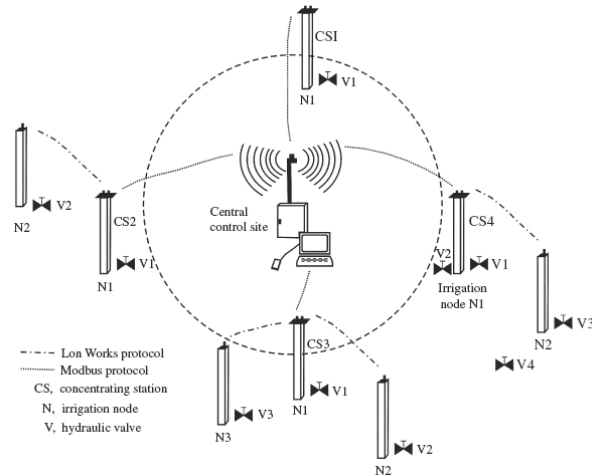


Figura 3. Sistema de telemetría de irrigación con el sistema comercial SIGA. (Pulido-Calvo et al., 2007)

Los datos son leídos cada hora para así obtener 24 lecturas diariamente. Esta frecuencia puede ser adaptada de acuerdo a las necesidades del usuario así como también la duración del riego, el volumen consumido, la hora de inicio, entre otras. Así, se logró obtener datos de la cantidad de agua demandada en cada hora para las diferentes granjas en estudio (Pulido-Calvo et al., 2007).

### 2.1.3 Sistema de Telemetría para medir señales fisiológicas en animales en libertad, Inglaterra.

En esta investigación se diseñó un sistema de telemetría para medir formas de ondas fisiológicas (electrocardiograma, electroencefalograma y señales de respiración), transmitir las, monitorearlas y registrarlas. El sistema de telemetría consiste de dos componentes primarios que se muestran en la siguiente figura: la unidad de registro de telemetría en sí misma, montada en el animal bajo estudio, y una estación base la cual se encuentra conectada a una computadora personal a través de una interfase serial.

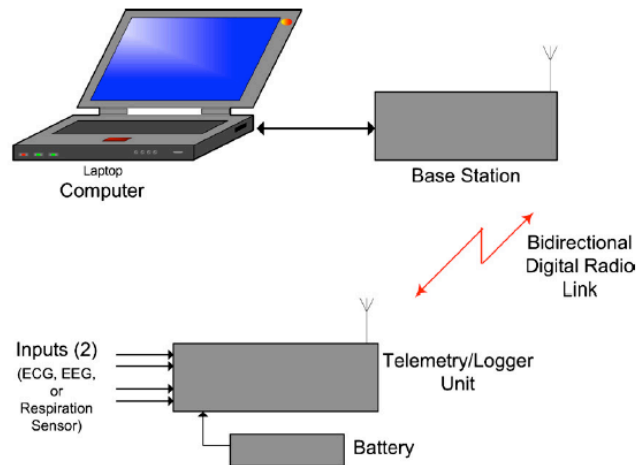


Figura 4. Sistema de telemetría para el registro de señales fisiológicas.  
(Lowe, J.C. et al., 2007)

La comunicación entre una unidad de registro telemétrica y la estación base se hace a través de un vínculo bidireccional de radiofrecuencia digital operando en la banda de aplicaciones Industriales, Científicas y Médicas en los 433 MHz. La telemetría bi-direccional también provee la habilidad de cambiar remotamente el modo de operación de la unidad de registro en modo 'solo registro' antes de que el animal pase por un área donde haya una pobre comunicación de radio.

El sistema opera en un modo de pregunta/respuesta, en el cual, un software a la medida en la computadora personal envía un comando a la estación base para que ésta le envíe datos a la unidad de registro telemétrica. La unidad de registro recibe el comando, lo procesa y luego retransmite una señal de respuesta a la unidad base y por consiguiente a la computadora personal. Esta respuesta puede ser un simple mensaje de reconocimiento, un mensaje de estado, o puede ser una trama de formas de onda digitalizadas de información fisiológica. Cada unidad de registro telemétrica tiene su propia dirección, y toda la información que pasa a través de los radios es en forma de paquetes de datos digitales diseccionados individualmente a una unidad de registro específica.

Un diagrama a bloques de la unidad de registro telemétrica se muestra en la siguiente figura. La unidad se encuentra basada en un microcontrolador de un solo chip que incorpora un núcleo de una microcomputadora, memoria de datos de programa, un convertidor analógico-digital de 12 bits, temporizadores y múltiples líneas de entrada/salida. El microcontrolador (MSP430F149 de Texas Instruments) está conectado a un radio tranceptor comercial de 433 MHz el cual provee la interfaz de radiofrecuencia. Adicionalmente, el microcontrolador se conecta a dos dispositivos de memoria HM62V815 de Hitachi lo cual permite el almacenamiento en memoria de muestras de formas de onda cuando la unidad se encuentra funcionando en modo 'solo registro'.

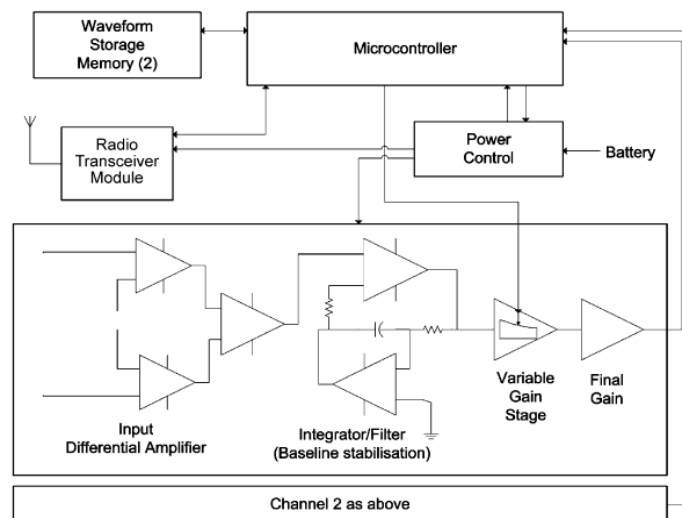


Figura 5. Diagrama a bloques del sistema de telemetría para señales fisiológicas. (Lowe, J.C. et al., 2007).

El uso de un sistema de telemetría para monitorear las señales fisiológicas de animales en laboratorio o en campo, mejora la calidad de los datos al remover la influencia del proceso de medición en sí mismo, por ejemplo el sujetar el animal o encerrarlo. La calidad de los datos también fue mejorada por el monitoreo de los datos en tiempo real. Por estas razones, el uso de sistemas de telemetría para monitorear parámetros fisiológicos en animales es altamente deseable (Lowe, J.C. et al., 2007).

### 2.1.4 Diseño de un sistema de monitoreo del agua basado en redes de sensores inalámbricas, China.

En el 2009 se diseñó un sistema de monitoreo ambiental basado en una red de sensores inalámbricos utilizando tecnología Zigbee y Gprs. Este sistema involucra tres partes principales: a) Nodos de monitoreo de datos, b) Estaciones de datos base y c) Centro de monitoreo remoto. El sistema fue diseñado para la medición de largo alcance en lagos, ríos, cuencas o pozos y permite la medición de temperatura del agua en un rango de 0 a 80 °C con una exactitud de  $\pm 0.5$  °C además de la medición de pH en un rango de 0 a 14 con una exactitud de  $\pm 0.5$  unidades pH. El diagrama general del sistema se muestra en la siguiente figura.

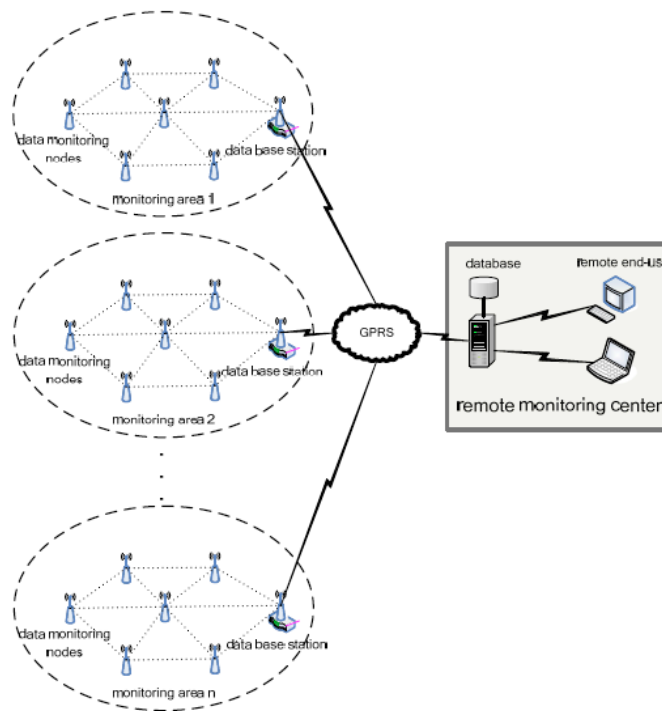


Figura 6. Diagrama a bloques del sistema de monitoreo de calidad del agua. (Peng J. et al., 2009).

Los datos adquiridos por los nodos de monitoreo son transferidos al centro de monitoreo remoto a través de una red GPRS. El centro de monitoreo remoto

analiza y procesa los parámetros de calidad del agua, enciende alarmas debido a la contaminación del agua por ejemplo.

### 2.1.5 Sistema de telemetría y telecontrol comercial

ADCON Telemetry es una empresa estadounidense que tiene a la venta un sistema de comunicación inalámbrica para mediciones a distancia de rango medio. Este sistema consta de una unidad base y diferentes unidades remotas. Las unidades remotas pueden medir hasta 12 canales analógicos y 8 entradas digitales (las cuales 4 son del tipo contador), puede controlar hasta 4 salidas. Tiene una interfase serial CMOS de 3 Volts que permite configurar la unidad remota, hacer respaldo de datos, o expansiones. Este dispositivo se basa en un microcontrolador de 8 bits que puede ser también programado en campo.

La frecuencia de operación se encuentra en el rango de 430 a 470 MHz, lo cual lo hace adaptable a la mayoría de las regulaciones de radiocomunicación en el mundo. Tiene una batería recargable de NiCd de 6.2V la cual es cargada usando ya sea un panel solar o una fuente de poder externa. Puede ser fácilmente instalada y, dependiendo del terreno, asegura una comunicación confiable de hasta 20 Kilómetros.



Figura 7. Unidad terminal remota comercial. (ADCON Telemetry, 2010).

Las unidades remotas se comunican a una unidad central que se encuentra localizada fuera del campo donde se llevan a cabo las mediciones. Esta unidad central se conecta a una computadora desde donde se puede configurar la



operación de todas las unidades remotas, dar de alta sensores, respaldar información de toda la red, configurar la frecuencia de transmisión, y otras funciones administrativas.

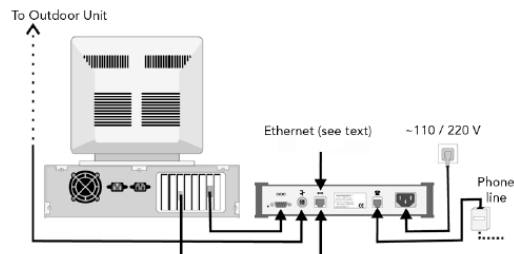


Figura 8. Conexiones de la unidad base ADCON. (ADCON Telemetry, 2010)

Una característica importante de este sistema es que el estado de los sensores conectados a las unidades terminales remotas se pueden monitorear por Internet. Esto hace que prácticamente desde cualquier parte del mundo el usuario pueda tener acceso a esta información en tiempo real.

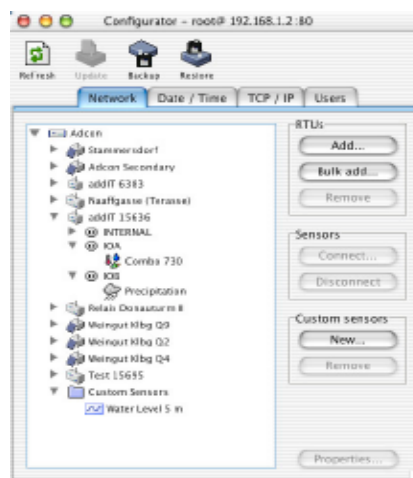


Figura 9. Pantalla de configuración de la red inalámbrica. (ADCON Telemetry, 2010).

## **2.2 Fundamentación teórica**

### **2.2.1 Comunicaciones de datos**

Debido a la naturaleza del sistema que se propone, es importante conocer fundamentos teóricos sobre las comunicaciones de datos las cuales, consideran dos tipos: comunicaciones analógicas y comunicaciones digitales. A continuación se describe cada una de ellas desde el punto de vista de las comunicaciones inalámbricas.

#### **2.2.1.1 Comunicaciones analógicas**

Las comunicaciones analógicas experimentan tres tipos principales de modulación, modulación en amplitud y modulación en frecuencia. A continuación se describen.

##### **Modulación en Amplitud**

La modulación en amplitud comúnmente llamada AM es el tipo de modulación más antigua. Consiste en modificar la amplitud de una onda portadora con una señal de información (moduladora) como se muestra en la Figura 10. En la parte superior se muestra una onda portadora (información que se quiere transmitir), en la parte de en medio, una onda portadora y en la parte inferior el resultado del proceso de modulación. Este tipo de modulación es comúnmente usado en la radio comercial.

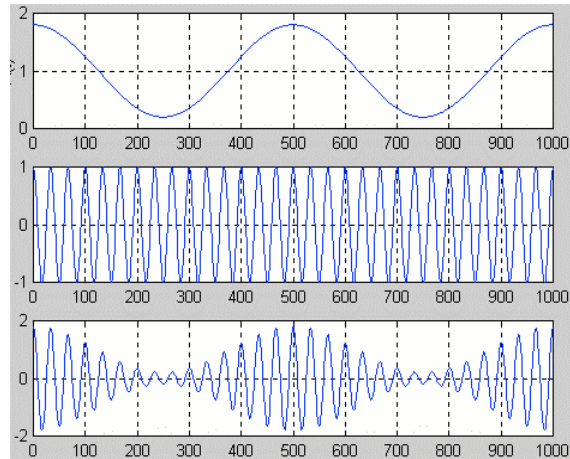


Figura 10. Modulación en amplitud de una señal senoidal.

### Modulación en frecuencia

Es el proceso de codificar información, la cual puede estar tanto en forma digital como analógica, en una onda portadora mediante la variación de su frecuencia instantánea de acuerdo con la señal de entrada (moduladora).

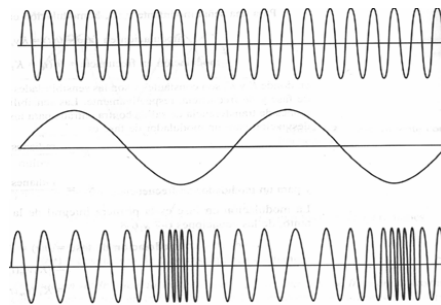


Figura 11. Modulación en frecuencia de una señal senoidal.

La modulación de frecuencia requiere un ancho de banda mayor que la modulación en amplitud para una señal modulante equivalente. En la figura 11 se muestra la modulación en frecuencia de dos ondas senoidales en la parte inferior,

en la parte superior se muestra la señal de la portadora y en la parte central la onda moduladora.

Actualmente, la comunicación por radiofrecuencia emplea técnicas de modulación digital para disminuir el ruido que pudiese estar presente en las técnicas de modulación analógicas. A continuación se presenta un resumen de las técnicas de modulación digital más comúnmente utilizadas.

### **2.2.1.2 Comunicaciones digitales**

Es un área extensa de técnicas de comunicaciones que está formada de dos partes: transmisión digital y radio digital. La transmisión digital es la transmisión de pulsos digitales entre dos o más puntos en un sistema de comunicación. Requieren de un elemento físico. Radio digital es la transmisión de portadoras analógicas, moduladas de forma digital, entre dos o más puntos en un sistema de comunicación. No requieren de un elemento físico.

Debido a la naturaleza de esta investigación, en este protocolo se hablará únicamente de las técnicas de radio digital.

#### **Transmisión por desplazamiento en amplitud Amplitude Shift Keying (ASK)**

Es una forma de modulación la cual se utiliza para representar datos digitales como variaciones en la amplitud de una onda portadora. La amplitud de la portadora varía en relación con la trama de bits de la señal modulante manteniendo la frecuencia y la fase constantes.

## Formas posibles de ASK

1.- Representar un 0 lógico como la ausencia de portadora y un 1 lógico como la presencia de portadora.

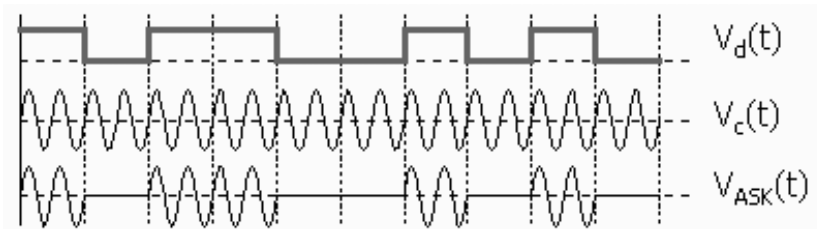


Figura 12. Primera representación de ASK.

2.- Representar un 0 lógico como un nivel de portadora menor y un 1 lógico con un nivel de portadora mayor

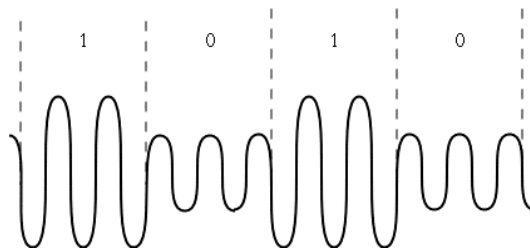


Figura 13. Segunda representación de ASK.

3.- Representar diferentes amplitudes para diferentes valores digitales

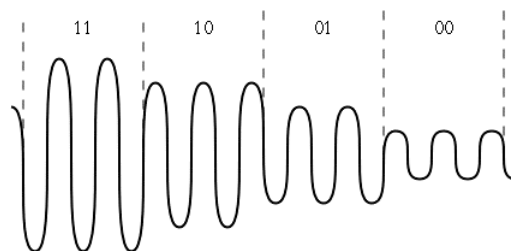


Figura 14. Tercera representación de ASK.

## Transmisión por desplazamiento en Frecuencia Frequency Shift Keying (FSK)

Es una forma de modulación la cual se utiliza para representar datos digitales como variaciones en la frecuencia de una onda portadora. La frecuencia de la portadora varía en relación con la trama de bits de la señal modulante manteniendo la amplitud constante.

Usualmente, la frecuencia es cambiada entre dos valores posibles llamados “marca” y “espacio”. Una marca se asocia a un “1 lógico” y un espacio a un “0 lógico”.

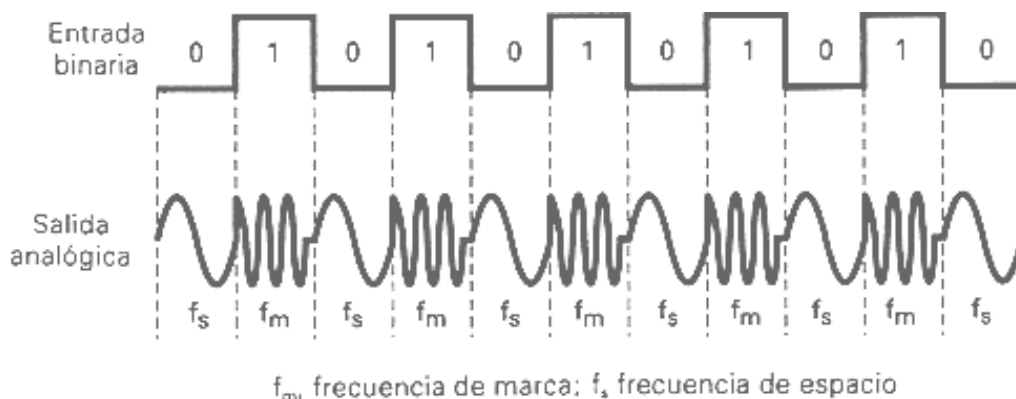


Figura 15. Transmisión por desplazamiento en frecuencia.

## Transmisión por desplazamiento en Fase Phase Shift Keying (PSK)

Es una forma de modulación la cual se utiliza para representar datos digitales como variaciones en la fase de una onda portadora donde existe un número finito de cambios de fases. La forma de representar los cambios de fases se hace comúnmente en un diagrama de constelación.

**BPSK.-** Es la forma más simple de PSK y consiste en la utilización de dos fases las cuales están separadas  $180^\circ$ . También suele ser nombrada 2-PSK.

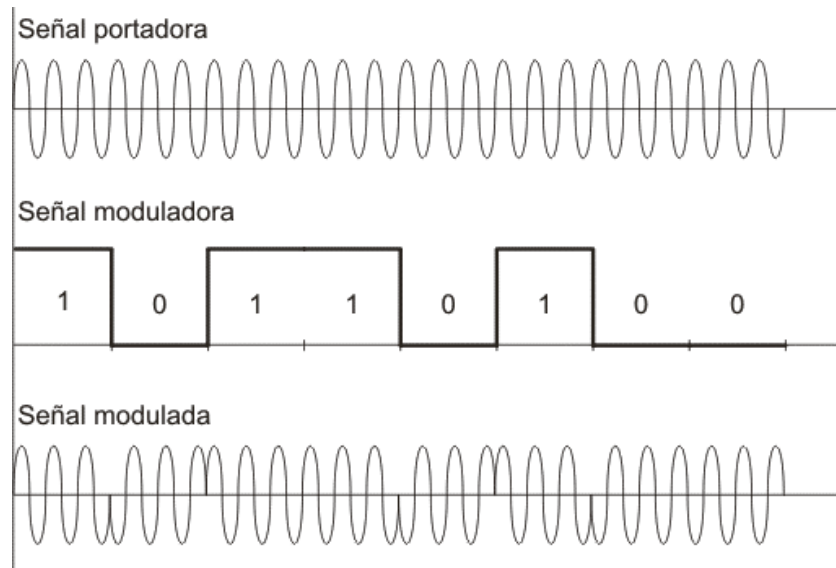


Figura 16. Transmisión por desplazamiento en frecuencia.

### 2.2.2 Configuraciones y topologías de circuitos de comunicación de datos

En un sistema de comunicaciones donde se ven involucradas dos o más unidades remotas es necesario conocer la forma en cómo se van a comunicar estas unidades. En este sentido, es conveniente definir el concepto de “Topología de red”, ésta es la disposición física en la que se conecta una red. Si una red tiene diversas topologías se le llama mixta.

Los circuitos de comunicación de datos pueden catalogarse generalmente como de punto a punto o multipunto y pueden tomar diferentes formas o diferentes topologías de red. Algunas de las topologías más comunes son las siguientes:

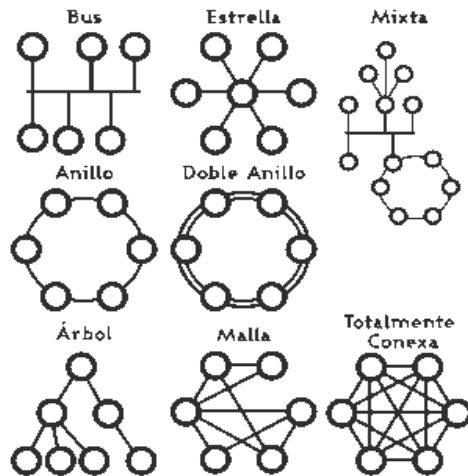


Figura 17. Topologías de red en los sistemas de comunicaciones

Es importante el concepto de topología de red ya que en ocasiones ésta característica diferencia a una red de otra. Tiene implicaciones en la red en cuanto a ahorro de energía, seguridad, prestaciones, colisiones, etc.

### 2.2.3 Modelo OSI de ISO

Al hablar de sistemas de comunicaciones no podemos dejar a un lado el modelo propuesto por la organización de estándares internacionales para la interconexión de sistemas abiertos llamado Modelo OSI.

Desde que un transmisor envía los datos hasta que un receptor los recibe en una red, se sigue una estructuración de capas o niveles propuesta por la ISO.

Los niveles OSI se dividen en 7 capas: Aplicación, Presentación; Sesión; Transporte, Red, Enlace y Físico. El propósito de cada capa es ofrecer ciertos servicios a las capas superiores, liberándolas del conocimiento detallado sobre como se realizan dichos servicios. Estos servicios se van incluyendo en unas cabeceras junto con los datos según van pasando de una capa a otra.



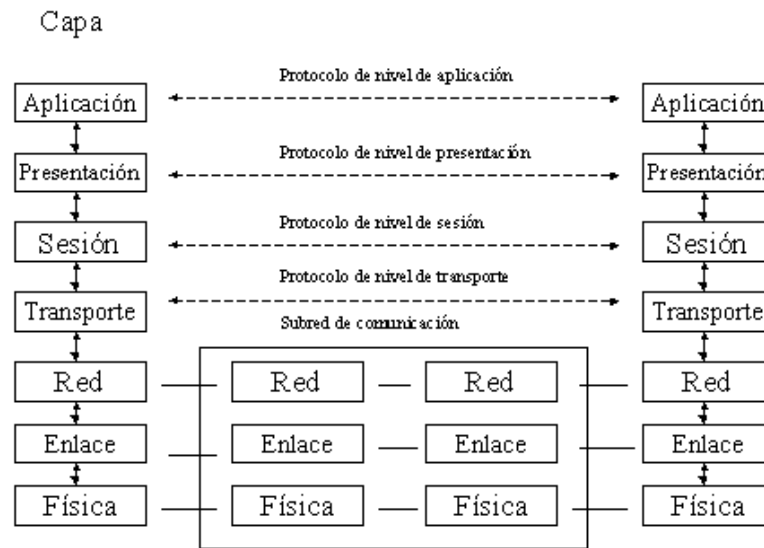


Figura 18. Modelo para la Interconexión de Sistemas Abiertos de ISO.

A continuación se describirán brevemente cada una de las capas.

### Capa Física

Se refiere a los requerimientos eléctricos como: Voltaje e Impedancia; a los requerimientos mecánicos como: tipos de conectores, forma de los conectores; requerimientos de procedimiento; transmisión síncrona, asíncrona, Full-duplex/ Half duplex / Simplex, uso de cada terminal en un conector. La unidad de transmisión de esta capa es el BIT.

### Capa Enlace

Se encarga de proporcionar un servicio conexión confiable. Tiene las funciones de: Organización de paquetes en tramas, agrega información redundante, agrega banderas, asegura la sincronización entre dos dispositivos y su unidad de transmisión es la TRAMA.

## **Capa Red**

Establece una trayectoria física y lógica entre dos nodos. Tiene las funciones de: Establece rutas de un nodo fuente a un nodo destino, direcciona nodos intermedios, reconoce prioridad en los mensajes y los envía con la prioridad establecida, ofrece servicios de interconectividad para enlazar redes por medio de ruteadores y su unidad de transmisión es el PAQUETE.

## **Capa de Transporte**

Es una capa de interfase entre las 3 capas superiores (interoperatividad) y las tres capas inferiores (interconectividad). Tiene las siguientes funciones: detecta y remueve paquetes duplicados, reenvía los paquetes, proporciona un control de alto nivel para la transferencia de datos y convierte direcciones a nombres.

## **Capa de Sesión**

Se ocupa de las funciones de gestión de red que incluyen contraseñas, monitorización e información de la red. Reestablece la comunicación si ocurre una ruptura del enlace sin perder datos. También tiene la responsabilidad de asegurar la entrega correcta de la información a la siguiente capa y controla el diálogo entre procesos para resolver las preguntas de ¿Quién transmite?, ¿cuándo?, ¿por cuánto tiempo?.

## **Capa de Presentación**

Se encarga de efectuar los mecanismos adecuados para que el formato de los datos transmitidos sea adecuado en el receptor. Es la primera capa en trabajar más sobre el contenido de la información que en cómo se establece la misma. Se

encarga de la compresión y encriptación de los datos. Y, realiza la transformación de caracteres.

### **Capa de Aplicación**

Esta capa provee los siguientes servicios al usuario, es decir, al programa de aplicación: Transferencia, administración y acceso de archivos, correo electrónico, emulación de terminales de computadoras.

### **2.2.4 Tecnologías en Redes inalámbricas**

Existen varias tecnologías utilizadas en redes inalámbricas. El empleo de cada una de ellas depende mucho de la aplicación. Cada tecnología tiene sus ventajas y desventajas. A continuación se listan las más importantes en este género.

- Infrarrojo (Infrared)
- Banda Angosta (Narrowband)
- Espectro Extendido (Spread Spectrum)

#### **Infrarrojo**

Los sistemas de comunicación por infrarrojo utilizan muy altas frecuencias, justo abajo del espectro de la luz visible para transportar datos. Como la luz, el infrarrojo no puede penetrar objetos opacos, ya sea directamente (línea de vista) o indirectamente (tecnología difundida/reflectiva). El alto desempeño del infrarrojo directo es impráctico para usuarios móviles pero su uso es prácticamente para conectar dos redes fijas. La tecnología reflectiva no requiere línea de vista pero está limitada a cuartos individuales en zonas relativamente cercanas.

## ***Banda Angosta***

Un sistema de radio de banda angosta transmite y recibe información en una radio frecuencia específica. La banda amplia mantiene la frecuencia de la señal de radio tan angostamente posible para pasar la información. El cruzamiento no deseado entre canales es evitado al coordinar cuidadosamente diferentes usuarios en diferente canal de frecuencia.

En un sistema de radio la privacidad y la no-interferencia se incrementa por el uso de frecuencias separadas de radio. El radio receptor filtra todas aquellas frecuencias que no son de su competencia. La desventaja de esta tecnología es el uso amplio de frecuencias, uno para cada usuario, lo cual es impráctico si se tienen muchos.

## **Espectro extendido**

La gran mayoría de los sistemas inalámbricos emplean la tecnología de Espectro Extendido (Spread Spectrum), una tecnología de banda amplia desarrollada por los militares estadounidenses que provee comunicaciones seguras, confiables y de misión crítica. La tecnología de Espectro Extendido está diseñada para intercambiar eficiencia en ancho de banda por confiabilidad, integridad y seguridad. Es decir, más ancho de banda es consumido con respecto al caso de la transmisión en banda angosta, pero la relación ancho de banda / potencia, produce una señal que es en efecto más fuerte y así más fácil de detectar por el receptor que conoce los parámetros de la señal de espectro extendido que está siendo difundida. Si el receptor no está sintonizado a la frecuencia correcta, una señal de espectro extendido se miraría como ruido en el fondo. Otra característica es la reducción de interferencia entre la señal procesada y otras señales no esenciales o ajenas al sistema de comunicación.

Existen dos tipos de señales de Espectro Extendido: Salto en Frecuencia (Frequency Hopping, FH) y Secuencia Directa (Direct Sequence, DS).

**Espectro extendido con salto en frecuencia (FHSS).** Utiliza una portadora que cambia de frecuencia con un patrón conocido tanto por el transmisor como por el receptor los cuales, están debidamente sincronizados comunicándose por un canal que está cambiado a cada momento en su frecuencia. El espectro extendido con salto en frecuencia es utilizado en aplicaciones por lo general punto a multipunto en las que se tiene una cantidad de receptores distribuidos.

**Espectro extendido en secuencia directa (DSSS).** DSSS genera un patrón de bits redundantes para cada bit que se transmite. Este patrón de bits es llamado código chip. Entre más grande sea este chip, es más grande la probabilidad de que los datos originales puedan ser recuperados (pero, por supuesto se requerirá más ancho de banda). Sin embargo si uno o más bits son dañados durante la transmisión, técnicas estadísticas embebidas dentro del radio transmisor podrán recuperar la señal original sin necesidad de retransmisión.

### 3. DESCRIPCIÓN DEL SISTEMA

El sistema de monitoreo a gran escala, involucra una serie de estaciones remotas (estaciones meteorológicas) que comparten una red de área local inalámbrica; a través de un módulo de adquisición de datos, reciben información de los sensores climatológicos de: Temperatura del aire, Humedad Relativa, Presión Barométrica, Radiación Solar, Velocidad y Dirección del viento y Precipitación Pluvial. Estos datos son transmitidos por RF con una frecuencia de 900 MHz a una estación base o estación central, la cual recibe esta información, la valida y la presenta en una página Web almacenada en un servidor HTTP Embebido. Esta información es enviada con intervalos de 10 minutos a un servidor central (el cual se localiza en la Facultad de Ingeniería de la Universidad Autónoma de Querétaro) vía un enlace satelital en la banda Ku. Este sistema ha sido nombrado como METEO-UAQ y se muestra en la siguiente figura un diagrama general de mismo.



Figura 19. Panorama general del sistema METEO-UAQ V1.3

En el servidor central, se almacenan los datos enviados por todas las estaciones remotas, se generan datos históricos y se presentan al usuario a través de un portal Web en forma gráfica, numérica y con posibilidades de descarga de archivos.

El sistema METEO-UAQ V1.3 se ha diseñado en tres partes principales: a) Unidades remotas (RTU), Unidades Base (BTU) y Servidor Central (CS). A continuación se detallan cada una de ellas.

### **3.1 Unidad Terminal Remota (RTU)**

Una Unidad Terminal remota es un dispositivo electrónico el cual constituye una interfaz entre el mundo físico y un centro de monitoreo central. Estas terminales remotas realizan la adquisición de los datos climatológicos y los transmiten a distancia de manera inalámbrica. También, tienen la capacidad de comunicarse con otras RTUs y ejecutar acciones simples de autonomía como la retransmisión de datos.

#### **3.1.1 Diseño de hardware de las RTUs**

Las RTUs se encuentran basadas en el microcontrolador (MCU) PIC18F8723 de alto desempeño quien es el encargado de realizar tres actividades principales: adquisición, procesamiento y transferencia de los datos al módulo de comunicaciones para su transmisión dentro de la red.

Las RTUs pueden ser de dos tipos: a) Una RTU de adquisición de datos, la cual procesa canales analógicos y digitales provenientes de los sensores climatológicos. Esta misma unidad filtra las señales recibidas por los sensores y los envía a la unidad base a través de un módulo de radio frecuencia cuando se le solicitan, y b) Un repetidor usado para amplificar la señal de radiofrecuencia

cuando las distancias son mayores a 64 kilómetros o cuando no hay línea de vista entre una RTU y la unidad base. La diferencia entre una RTU del tipo a) con la del tipo b) es el módulo de adquisición y procesamiento de los datos. Esto es, las RTUs no procesan datos, simplemente retransmiten.

Los módulos de las RTUs de adquisición de datos son adquisición y procesamiento (APM), fuente de alimentación (SPM) y comunicaciones (CM). Por otro lado, las RTUs repetidoras solo tienen el APM y el SPM. A continuación se describen estos módulos

#### **3.1.1.1 Módulo de adquisición y procesamiento (APM)**

Este módulo procesa seis canales analógicos (con posibilidad de expansión hasta 16 canales independientes) los cuales son utilizados para la medición de variables climatológicas: temperatura del aire, humedad relativa, radiación solar, presión barométrica, dirección del viento y el estado que guarda la batería del sistema; además, procesa dos canales digitales provenientes de los sensores de velocidad del viento y precipitación pluvial.

Los canales analógicos están diseñados para permitir entradas con una variación de voltaje de 0 a +5 Volts (Sensor de radiación solar), de 0 a 2.5 V (Sensor de presión y voltaje de la batería) y de 0 a 1 V (Sensores de temperatura del aire, dirección del viento y humedad relativa). Estos canales analógicos son convertidos en un valor digital de 12 bits utilizando el ADC interno del microcontrolador, el cual lee el voltaje de referencia que corresponde a cada uno de los sensores para finalmente almacenados en una memoria no volátil, la memoria de EEPROM de datos.



Tabla 3. Reglas de conversión de los sensores analógicos.

Sensor	Salida	Módulo de procesamiento	Regla de conversión	Unidades
Radiación solar	0-5V	Canal 0, ADC	$\text{Radsol} = ((\text{read\_adc}() / 4095) * 5000.0) * 0.25$	$\text{W/m}^2$
Presión barométrica	0-2.5V	Canal 1, ADC	$\text{Press} = (((\text{read\_adc}() / 4095) * 5) * 240) + 700.67$	mb
Voltaje de la batería	0-2.5V	Canal 2, ADC	$\text{Bat} = ((\text{read\_adc}() * 5.) / 4095) * 3.15$	V
Dirección del viento	0-1V	Canal 3, ADC	$\text{Dir} = (\text{lectura\_adc} / 4095) * 360$	°
Temperatura del aire	0-1V	Canal 4, ADC	$\text{Temp} = (((\text{read\_adc}() * 500.0) / 4095) - 40.0)$	°C
Humedad Relativa	0-1V	Canal 5, ADC	$\text{Hum} = (((\text{read\_adc}() * 500.0) / 4095);$	%

Los canales digitales únicamente tienen dos posibles estados 0 y 5 Volts. Para el caso del sensor de precipitación pluvial, se cuenta el número de pulsos y se almacena en un contador de 32 bits, lo que quiere decir que el número máximo de cuenta será de  $2^{32-1}$  esto es: 4 294 967 295, cifra más que suficiente para almacenar pulsos de precipitación pluvial. El sensor de velocidad del viento entrega una salida de pulsos a una determinada frecuencia, la cual es directamente proporcional a la velocidad del viento. En la siguiente tabla se hace un resumen de la adquisición de datos de estos sensores:

Tabla 4. Reglas de conversión de los sensores digitales.

Sensor	Salida	Módulo de procesamiento	Regla de conversión	Unidades
Precipitación pluvial	Pulsos digitales	Interrupción externa 0	$\text{Pluv} = \text{pulse\_pluv} * .254;$	mm
Velocidad del viento	Pulsos digitales	Captura-compara (CCP1 y CCP2)	$\text{FREC} = 1 / (\text{ancho\_pulso} * 0.000002);$ $\text{Vel} = 0.7990 * \text{frec} + 0.2811$	m/s

En la siguiente figura, se muestra a manera de bloques el módulo APM conjuntamente con los sensores climatológicos, el módulo de alimentación de energía y el módulo de comunicaciones los cuales se explican en breve.

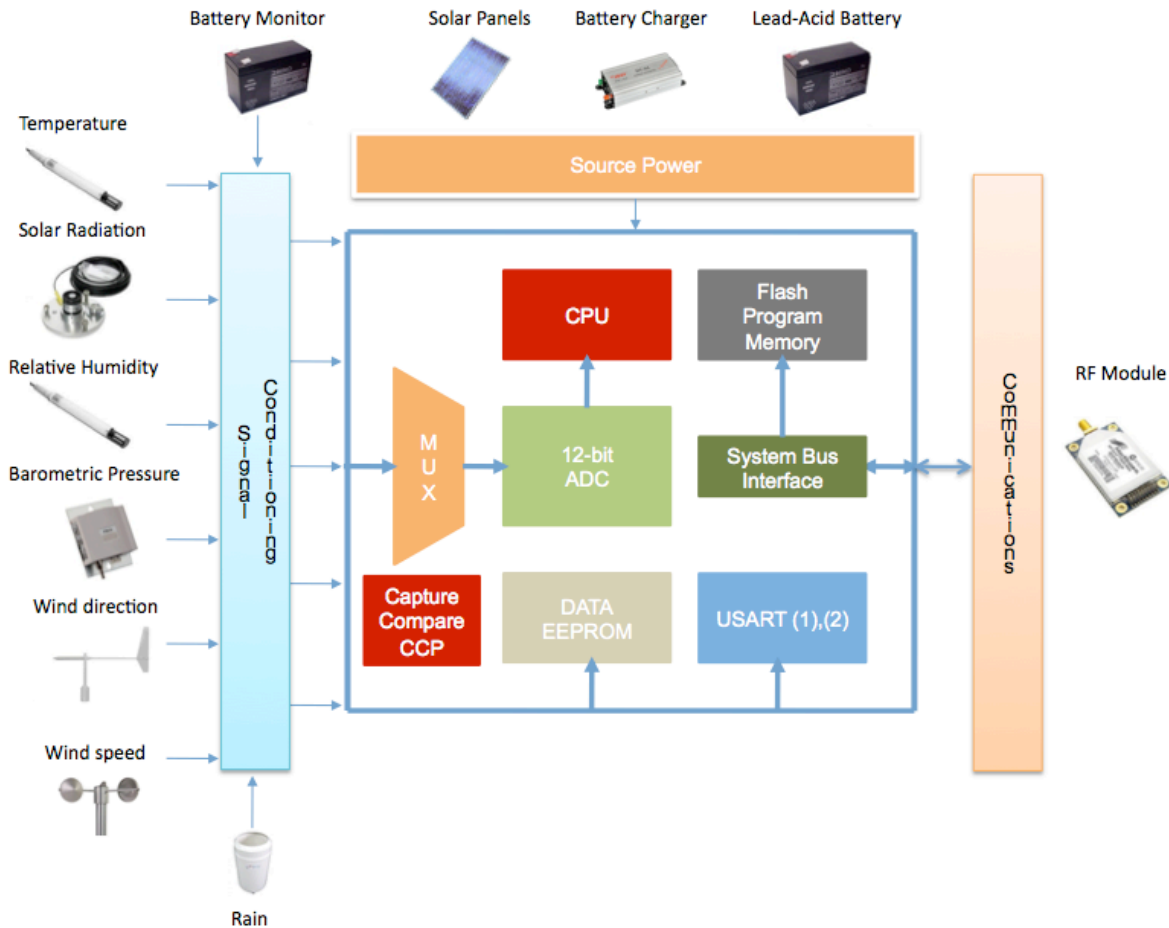


Figura 20. Módulo del acondicionamiento de señal de la RTU y recursos auxiliares.

### 3.1.1.2 Módulo de energía (SPM)

El módulo de energía está constituido de un panel solar de 20 Watts, un cargador de batería inteligente y una batería de ácido-plomo de 12 Volts a 7 Ah. El panel solar es la fuente de poder fotovoltaica para el cargador de baterías. Éste provee picos de potencia máximos de 20 Watts y 1.19 Amperes de corriente pico, suficiente para alimentar a los módulos de adquisición/procesamiento y comunicaciones.

El cargador inteligente de baterías se encuentra basado en el circuito integrado UC3906. Éste monitorea y controla tanto el voltaje de salida y la corriente del cargador a través de tres estados diferentes de carga; un estado de alta corriente, un estado controlado de sobrecarga y un estado de carga flotante o estado de reposo de precisión. El proceso de carga, en otras palabras, empieza cuando la corriente de salida del cargador se encuentra limitada en un nivel bajo hasta que la batería alcanza un voltaje específico (previniendo una corriente alta de carga si es que la batería se encuentra en corto circuito) y entonces, el cargador entra en un estado de carga en alta corriente. En este estado, el cargador proporciona un pico de corriente hasta que el voltaje de la batería alcance el 95% de su máxima capacidad. Luego, el cargador cambia al estado de sobrecarga para cargar completamente la batería, en este estado, la corriente se reduce al mínimo conforme el voltaje se acerca a su máxima capacidad y entonces, el cargador cambiará al estado flotante o estado de reposo. Ya que la carga (constituida por los módulos de adquisición/procesamiento y comunicación de datos) se encuentra conectada, si la batería cae un 10% del voltaje establecido en el estado flotante, el cargador se reiniciará a sí mismo y comenzará nuevamente el proceso de carga.

### **3.1.1.3 Módulo de comunicaciones (CM)**

Los módulos de radiofrecuencia utilizados para la comunicación de largo alcance fueron los Xtend OEM del fabricante Digi. Estos módulos permiten la interfaz con un dispositivo del tipo host a través de niveles TTL utilizando una USART o a través de una interfaz RS232.

Con estos módulos se implementó una topología de malla la cual permite a un mensaje que sea direccionado a diferentes nodos dentro de la malla hasta llegar al nodo destino. En caso de se pierda la conexión entre distintos nodos de la red (debido a pérdida de potencia en los transmisores, factores ambientales, etc.), los

datos enviados, los cuales se consideran críticos, pueden aún llegar a su destino debido a las capacidades que se han programado en estos módulos.

Las principales características de los nodos dentro de la red son las siguientes:

*Auto-reparación:* Cualquier nodo puede entrar o salir de la red en cualquier momento sin causar que toda la red inalámbrica se caiga por completo.

*Arquitectura P2P:* No se reconoce jerarquía entre módulos iguales.

*Envío de confirmaciones ACK:* Los nodos remotos responderán cuando se haya establecido una comunicación exitosa aumentando la confiabilidad en las transmisiones.

Si el módulo de RF no puede transmitir inmediatamente (ejemplo, cuando está recibiendo grandes cantidades de datos), los datos que recibe del MCU son almacenados en un buffer llamado DI.

El número de reintentos para lograr una comunicación exitosa de datos se ha configurado en 3, esto quiere decir que los paquetes de RF son enviados 4 veces. Esto se realizó configurando el parámetro NR (Network Retries).

Cuando se inicia una transmisión, sucede lo siguiente:

1. Se envía un preámbulo de 10 bits (con el número configurado de retransmisiones).
2. Se envían los datos, los cuales comienzan con el encabezado (dirección destino a nivel MAC address) y terminando con un código de redundancia cíclica (CRC) de 16 bits.
3. Se espera por un ACK por un periodo fijo de tiempo (1.8 milisegundos).
4. Si el ACK es recibido, la transmisión se ha completado correctamente.

5. De otra manera, se inicializa un temporizador el cual al desbordarse inicia una retransmisión.

Todos las RTUs necesitan transmitir los datos a las Unidades Terminales Base (BTUs) correspondientes. Para ésto, se consideran parámetros únicos relacionados con el número de serie de cada BTU: a) Dirección destino formada por parte alta y baja (DH y DL respectivamente) y b) Dirección origen parte alta y baja (SH y SL respectivamente). La BTU es quien ordena la transmisión de los datos adquiridos a las RTUs. Éstas últimas tienen configurada como dirección a transmitir la dirección origen de la BTU. Haciendo una analogía con un sistema telefónico, el número destino (DH-DL) que tienen que “marcar” las RTUs para transmitir los datos es el número único de serie (SH-SL) de la BTU. Por el contrario, si la BTU quiere comunicarse con cada una de las RTUs, tendrá que “marcar” el número de serie específico (SH-SL) de cada una de las RTUs. En la siguiente figura se ilustra esta relación de direcciones.

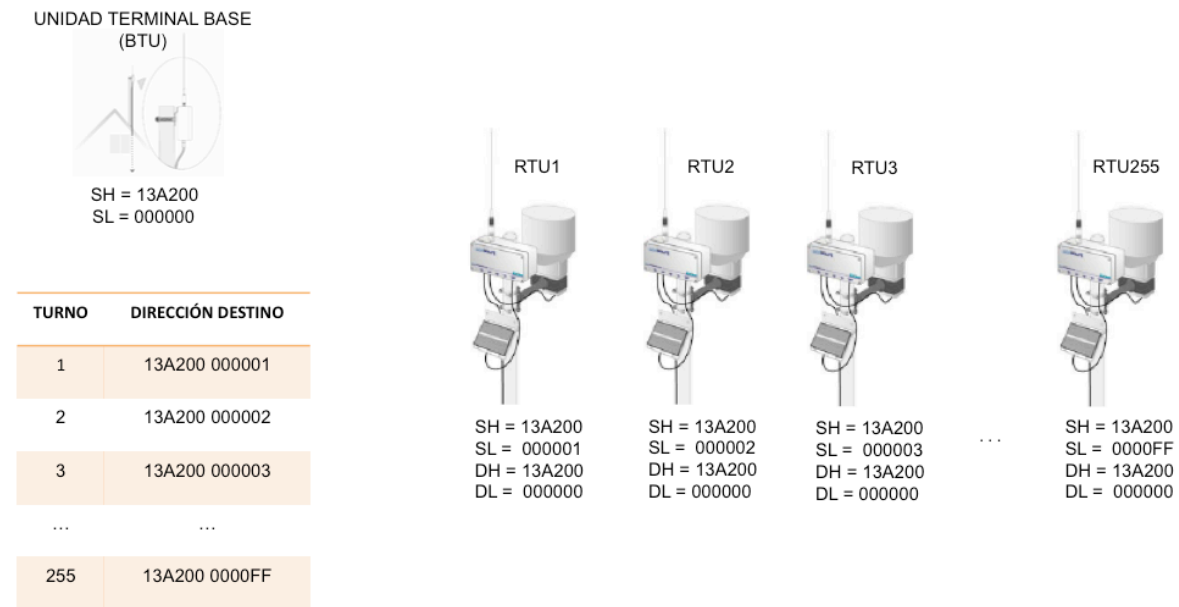


Figura 21. Parámetros destino y origen en la comunicación inalámbrica.

### **3.1.2 Software embebido en las RTUs**

Un sistema embebido se encuentra basado típicamente en un microcontrolador o un procesador de señales digitales (DSP) el cual tiene que cumplir una o varias funciones dedicadas (normalmente en tiempo real). El software embebido es aquel contrapuesto al software de alto nivel y que a diferencia de éste último, no es visible por el usuario. El término “embebido” se refiere a que normalmente el sistema se encuentra “incrustado” en la aplicación que gobierna.

El software que se ejecuta en el microcontrolador PIC18F8723 se programó en lenguaje C utilizando el compilador de Custom Computer Services (CCS). A continuación se abordan las rutinas más importantes.

#### **3.1.2.1 Rutina maestra**

La rutina maestra de las RTUs está predominantemente controlada por interrupciones, esto quiere decir que las tareas que el sistema realiza son inicializadas por diferentes tipos de eventos. Entre las interrupciones utilizadas se encuentran; la interrupción del módulo CCP (relacionada con la medición de la frecuencia del sensor de velocidad de viento), la interrupción externa (la cual se usa en el cálculo de la cantidad de lluvia que ha caído) y la del módulo USART (la cual se activa cuando se solicita la adquisición/transmisión de los datos climatológicos).

El modelo de comunicación utilizado es del tipo Maestro-Escavo; esto es, el esclavo se encuentra siempre en modo de escucha y sólo transmite cuando se le solicita. El maestro es el servidor central quien genera una petición a la BTU, la cual se encuentra en modo escucha, ésta a su vez identifica el destino de la petición y solicita los datos remotos de las RTUs quienes también se encuentran en modo escucha. Este modelo reduce significativamente el tráfico dentro de la red.

En las siguientes tablas se incluyen fragmentos extraídos de la rutina principal de la RTU los cuales se explican para una mejor comprensión del lector. En la Tabla 5 se incluyen las directivas pre-procesador las cuales definen el microcontrolador a utilizar (línea 1), la resolución del ADC (línea 2), la palabra de configuración (línea 4), la frecuencia del oscilador (línea 5), también configuran la USART para la comunicación con el servidor HTTP embebido (línea 6) y con el módulo de radiofrecuencia (línea 7), incluyen librerías adicionales (líneas 3, 8 y 9) y finalmente incluyen el controlador para el reloj de tiempo real utilizado (línea 10).

Tabla 5. Rutina principal de la Unidad Terminal Remota.

Línea	Código
1	<code>#include &lt;18F8723.h&gt;</code>
2	<code>#device adc = 12</code>
3	<code>#include &lt;string.h&gt;</code>
4	<code>#fuses XT,WDT8192,NOPROTECT,BROWNOUT</code>
5	<code>#use delay(clock=4000000)</code>
6	<code>#use rs232(baud=9600, xmit=PIN_C6, rcv=PIN_C7, stream=PORT1) //Para RF</code>
7	
8	<code>#include &lt;stdlib.h&gt;</code>
9	<code>#include &lt;math.h&gt;</code>
10	<code>#include &lt;ds1307.c&gt;</code>

Las solicitudes de información las envía la BTU utilizando la red inalámbrica y llegan a las RTUs través del módulo de comunicaciones. Estas peticiones de envío de datos se transmiten a través de RS232 y se basan en comandos cortos consistentes de un carácter ASCII seguido de un “*enter*”. Estos datos se convierten a valores TTL por un circuito integrado MAX232 y se reciben por la terminal RX del módulo USART del microcontrolador. Cuando la RTU recibe cualquier carácter por el puerto USART se genera una interrupción. En la tabla 6 se incluye el código que da lectura a los caracteres recibidos.

Tabla 6. Modo escucha de la Unidad Terminal Remota y comunicación vía inalámbrica con la Unidad Terminal Base.

Línea	Código
1	#int_RDA
2	void rcv_data1()
3	{
4	buf[i]=fgetc(PORT1);
5	if(buf[i]==0x0d)
6	{
7	str_flag=1;
8	buf[i+1]=0;
9	strcpy(str,buf);
10	i=0;
11	}
12	Else
13	i++;
14	}



En la línea 1 se declara el tipo de interrupción a utilizar, en este caso, se utiliza la interrupción serial. La línea que sigue a la declaración de la interrupción nombra a la rutina que da servicio a la interrupción. En la línea 4, se lee el caracter recibido y se almacena en un buffer para después compararse (línea 5) con el caracter 0x0D (*enter*); si esto sucede, se activa una bandera que indica que se ha recibido un “*enter*” (línea 17), se agrega un caracter nulo al final de la cadena recibida (línea 18) lo cual es requisito para poderse comparar o copiar a otra cadena y finalmente se copia a la cadena “*str*” (línea 19). Por el contrario, si el caracter recibido no es “*enter*”, únicamente se incrementa un índice y termina la interrupción. La bandera de cadena recibida “*str\_flag*” permite a la rutina principal salir del modo escucha e interpretar la petición de datos. En la siguiente tabla se ilustra un fragmento de código en el que se realiza la adquisición de los datos dependiendo del comando recibido.

Tabla 7. Interpretación de comandos, adquisición y transmisión de datos al la Unidad Terminal Base.

Línea	Código
1	get_string();
2	//-----TEMPERATURA-----//
3	if(strcmp(Ch0,str)==0)
4	{
5	set_adc_channel(0);
6	delay_us(20);
7	Temp = (((read_adc())*500.0)/resolucion)-40.0; //Grados centígrados
8	fprintf(PORT1,"T%.2f", Temp);
9	fputc(0x0D,PORT1);
10	}

En la línea 1 de la tabla anterior, se encuentra la función *“get\_string”*. Ésta mantiene a la RTU en un modo de escucha, esto es, el contador del programa se encuentra en una iteración dentro de esta función preguntando si la bandera de cadena recibida es igual a 1 y no es sino hasta que esta condición se satisface que el contador de programa deja la función *“get\_string”* para dar paso a la adquisición de los datos y a la transmisión de los mismos.

En la línea 3 se compara la cadena recibida con un valor esperado (para temperatura: *“T,0x0D,0”*). Si esto se cumple, entonces se configura el canal cero del convertidor analógico digital (línea 5) para su futura lectura.

En la línea 6 se espera un tiempo de 20  $\mu$ s recomendado por el fabricante. La conversión de un número digital a grados centígrados se realiza en la línea 7 y en este caso, se almacena en la variable *Temp*. La transmisión de esta variable se realiza en la línea 8, y en línea 9 se transmite un *“enter”* que es el caracter que indica a la BTU que los datos se han enviado.

Este procedimiento se repite para todas las variables climatológicas y para datos que se solicitan pero que propiamente no vienen de sensores climatológicos por ejemplo el punto de rocío (calculado a partir de una relación de temperatura y humedad), el voltaje de la batería, o también para comandos remotos como el reinicializar el CPU (ver Tabla 8).

Tabla 8. Comandos que interpretan las RTUs.

Identificador	Descripción	Canal
T,0x0d,0	Temperatura	Analógico 0
H,0x0d,0	Humedad relativa	Analógico 1
P,0x0d,0	Presión barométrica	Analógico 2
S,0x0d,0	Velocidad del viento	Módulo CCP
R,0x0d,0	Radiación solar	Analógico 4
L,0x0d,0	Precipitación pluvial	Int. Externa
D,0x0d,0	Dirección del viento	Analógico 5
Z,0x0d,0	Reinicializa remotamente el microcontrolador	N.A.
C,0x0d,0	Punto de rocío	N.A.
V,0x0d,0	Voltaje de la batería	Analógico 10
A,0x0d,0	Fija el reloj de tiempo real	Módulo I <sup>2</sup> C

### 3.1.2.2 Subrutina del módulo de comunicaciones

El módulo de comunicaciones tiene dos funciones principales: la configuración del módulo de RF, la decodificación de los datos recibidos y la transmisión de los datos adquiridos.

Para realizar la codificación de los datos transmitidos, el microcontrolador es programado para entrar al modo “Configuración” a través del envío serial de los caracteres “+++”, enseguida se configuran los parámetros de red a través de comandos AT.

Tabla 9. Configuración de parámetros en el módulo de RF.

<b>Comando</b>	<b>Valor</b>	<b>Descripción</b>	<b>Rango</b>
AT ID	3332	<i>Identificador.</i> Configura el identificador del fabricante. Sólo los módulos con el mismo ID pueden comunicarse.	0x10-0x7FF
AT HP	0	<i>Canal de transmisión.</i> Configura el canal cero de espectro extendido sobre el cual se comunicará el módulo de RF. Canales separados minimizan la interferencia entre múltiples redes de módulos operando en la misma vecindad.	0-9
AT DL	0x0000010	<i>Dirección destino baja.</i> Configura la parte baja de la dirección a la cual la RTU debe transmitir (0x0000010 corresponde a la BTU)	0-0xFFFFFFFF
AT DH	0x0013A200	<i>Dirección destino alta.</i> Configura la parte alta de la dirección a la cual la RTU debe transmitir (0x0013A200 por omisión)	0-0xFFFFFFFF
AT NR	2	<i>Reintentos de red.</i> Establece el número de intentos de entrega de paquetes a 3. Si el primer envío no recibe un acuse de recibido, entonces se reintentará enviar los paquetes de datos dos veces más.	0-0xFF
AT NH	7	<i>Salto de red.</i> Máximo número de saltos dentro de la red.	0-0xFF
AT PL	4	<i>Potencia de transmisión.</i> Selecciona la máxima potencia de transmisión (1 Watt)	1-4
AT WR	N.A.	<i>Guardar.</i> Guarda los parámetros configurados anteriormente.	N.A.
AT CN	N.A.	<i>Salir.</i> Sale del modo configuración.	N.A.

Una vez configurados los parámetros del módulo de RF el microcontrolador configura el módulo en el estado de transmisión/recepción, modo en el cual permanece en condiciones normales de operación.

### **3.2 Unidad Terminal Base (BTU)**

Esta unidad terminal remota dice cuándo y quien puede transmitir dentro de la red inalámbrica. Se encuentra basada en el microcontrolador de alto desempeño PIC18F6722 y consta de un módulo de alimentación de energía, un módulo de comunicaciones en RF y un módulo servidor http embebido. A diferencia de las RTUs, sólo puede haber una BTU por cada red inalámbrica de estaciones de sensores. A continuación se describe tanto el hardware como el software de esta BTU.

#### **3.2.1 Hardware de la Unidad Terminal Base**

La función principal de la BTU consiste en interpretar las solicitudes de adquisición de datos que se originan en el servidor central y efectuar las peticiones correspondientes con cada una de las RTUs de la red. Así, la BTU por un lado recibe datos satelitales y por otro envía peticiones a una frecuencia de 900 MHz en la banda ICM. En la siguiente figura, se muestran los bloques funcionales de la Unidad Terminal Base.

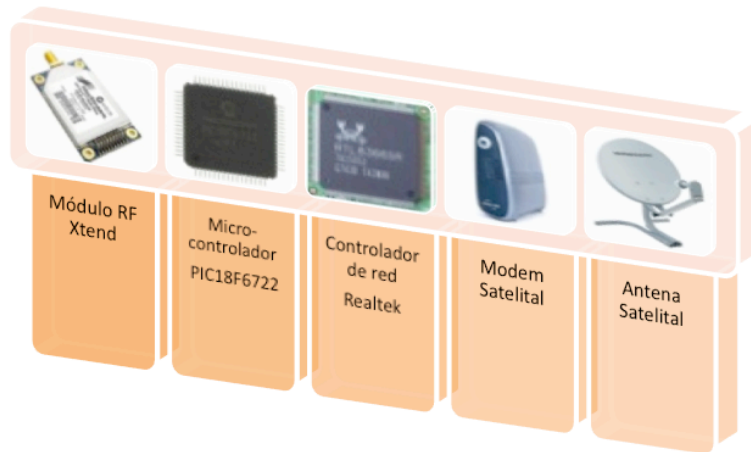


Figura 22. Componentes principales de la Unidad Terminal Base.

El módulo satelital consta de un módem Skystar 360E el cual se encarga de recibir señales con frecuencias del rango de 11.7 a 12.7 GHz, decodificarlas y enviarlas a través de Ethernet a 10 Mbps. Utilizando un cable UTP categoría 5, estas señales son conducidas a un conector RJ-45 y llegan a un controlador de red RTL8019S el cual las envía al microcontrolador PIC18F6722 a través del puerto SPI. El microcontrolador interpreta estas peticiones y envía por RS232 comandos al módulo de RF Xtend para su transmisión en el aire. Estos comandos son los que interpretan las RTUs se muestran en la Tabla X. Una vez que las RTUs adquieren los datos, los transmiten a la BTU y los datos siguen el mismo camino de regreso.

De los componentes principales de la Figura 22, tanto el microcontrolador como el controlador de red se encuentran embebidos en una misma tarjeta impresa (Figura 23) que por la función que realiza ha sido nombrada “Servidor HTTP Embebido”. El servidor HTTP embebido permite presentar en una página Web los datos recopilados por los diferentes módulos de adquisición y procesamiento para su consulta remota por parte del servidor central (CS). La principal ventaja de incluir un módulo servidor HTTP embebido radica en que evita la necesidad de tener una computadora en campo encendida las 24 horas debido a que los protocolos de red necesarios se encuentran programados dentro del microcontrolador.



Figura 23. Servidor HTTP Embebido de sistema Meteo UAQ V1.3

### 3.2.1.1 Interfaz con el módulo de RF

El servidor HTTP embebido cuenta con un puerto RS232 el cual permite comunicar al microcontrolador PIC18F6722 con el módulo de RF. Las conexiones en hardware para conectar ambos dispositivos incluyen un circuito integrado MAX232 para adecuar las señales de -12 y +12 Volts a 0 y 5 Volts a la entrada de la terminal RX en el módulo de RF. También, cuando un el módulo de RF necesita transferir información al microcontrolador (por la terminal TX) lo hace utilizando lógica TTL positiva, éstos datos llegan al circuito integrado MAX232 y entran al servidor HTTP embebido. Adicionalmente, se agrega un adaptador módem-null que permite conectar RX a TX y viceversa. El diagrama de conexiones se ilustra en la siguiente figura.

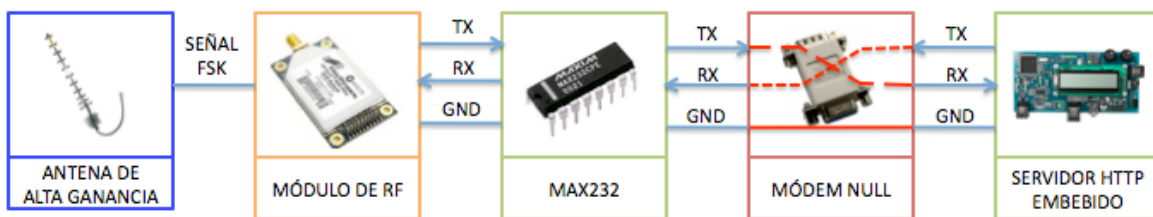


Figura 24. Diagrama de conexiones entre el servidor HTTP embebido y el módulo de RF.

### **3.2.1.2 Interconexión con la red satelital**

Cabe señalar, que para un sistema de comunicaciones de largo alcance (comunicación entre poblaciones o ciudades) las transmisiones de RF de las RTUs no son una solución viable. Debido a la curvatura de la tierra, la distancia máxima de transmisión está limitada. En el sistema desarrollado, se pretende monitorear las estaciones de sensores a grandes distancias y en este sentido, se ha escogido la transmisión de datos a través de la red satelital en la banda Ku. Para lograr este objetivo, la BTU juega un papel fundamental ya que hace las veces de puerta de enlace, esto es, permite enlazar una red local de sensores basada en radiofrecuencia, con una red de área extensa (WAN) interconectada por sistemas satelitales.

#### *Principio de funcionamiento*

Desde el servidor central se gestiona la información a través de una petición HTTP en una página WEB. Enseguida la información pasará por una puerta de enlace (Gateway) quien es la encargada de interconectar la red LAN (donde se encuentra el servidor central) con la red pública. Posteriormente, una estación satelital terminal de apertura muy pequeña (VSAT) será la encargada de enviar esta información a un satélite de la banda Ku. Esta banda se encuentra en un rango de frecuencias de 12 a 14 GHz y tiene las ventajas de usar transmisores de menor potencia con el inconveniente de poca capacidad de transmisión de datos, a diferencia de los enlaces satelitales en banda C cuyo rango de frecuencias se encuentra de 4 a 6 GHz y pueden transmitir grandes cantidades de datos. Sin embargo, para la aplicación que se requiere, la tasa de transferencia de datos es baja. Para comprobar esto, se realizaron pruebas preliminares para cuantificar las velocidades máximas necesarias para la transmisión de datos comúnmente llamada “velocidad de subida”. Las pruebas preliminares se realizaron haciendo peticiones a través del protocolo HTTP a la unidad base del sistema de comunicaciones y se muestran en la siguiente figura.



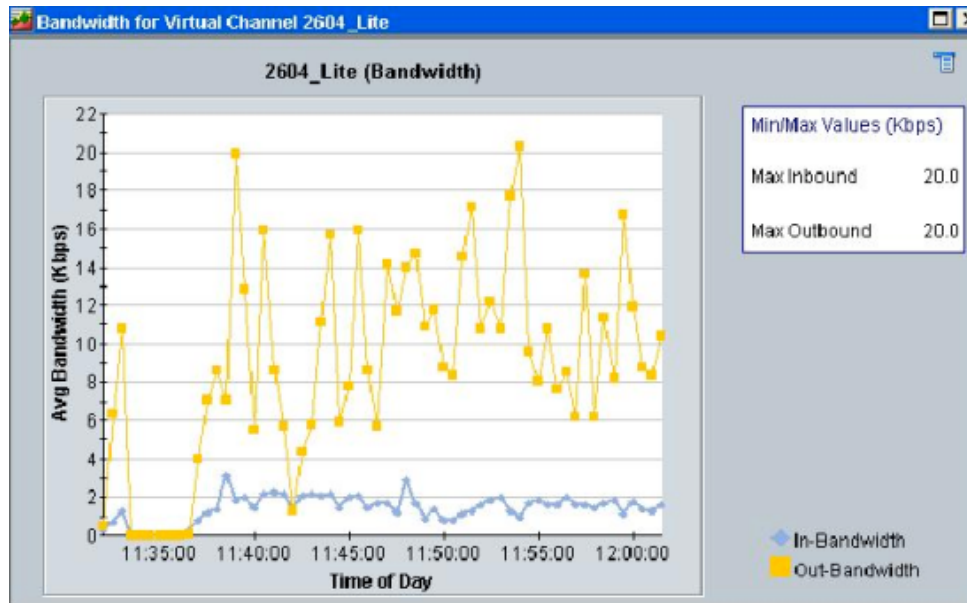


Figura 25. Ancho de banda requerido por la BTU para la transmisión de los datos meteorológicos.

En la figura anterior se muestra que la velocidad máxima de transferencia de datos se encuentra alrededor de 3 Kbps, la cual es muy baja, puesto que la información que se está enviando es básicamente los datos de las variables que se están monitoreando. Este análisis permite seleccionar la velocidad del enlace, que en este caso fue de 19.2 Kbps, más que suficiente para los requerimientos del sistema de monitoreo.

### 3.2.1.3 Módulo de alimentación de la Unidad Terminal Base

El módulo de alimentación de energía es sumamente importante para los sistemas de telemetría de largo alcance para zonas de difícil acceso ya que generalmente no se cuenta con un suministro eléctrico que permita conectar los equipos de medición ni mucho menos un enlace satelital. A continuación se describe el módulo de alimentación de energía utilizado para la Unidad Terminal Base.

El módulo de alimentación de energía consiste de un arreglo de dos paneles solares conectados eléctricamente entre sí y montados sobre un marco de aluminio que permite dar la inclinación necesaria al arreglo. Estos paneles generan se conectan a un controlador el cual tiene dos funciones principales: una es cargar de tres baterías de descarga profunda hasta su nivel óptimo desconectando la alimentación de los paneles solares cuando éstas se encuentran en su carga máxima y la otra, proporcionar 12 Volts al inversor de corriente directa a corriente alterna el cual alimenta al módem satelital. El diagrama de conexiones se muestra en la siguiente figura.

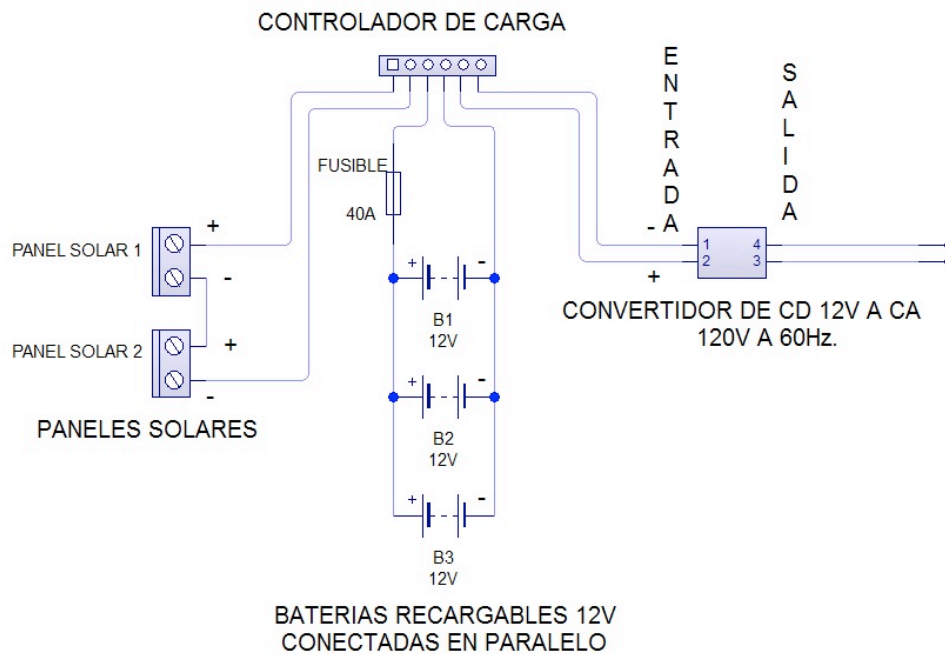


Figura 26. Diagrama de conexiones del módulo de alimentación de energía.

### 3.2.2 Software de la Unidad Terminal Base

Al igual que las RTUs, el software de las BTUs se encuentra programado en lenguaje C y se utiliza el compilador de Custom Computer Services (CCS). El software embebido lleva a cabo la adquisición de datos, la comunicación con el

módulo de radiofrecuencia y la comunicación con el módulo del servidor HTTP embebido.

### Software del servidor HTTP embebido

En esta sección se incluirá únicamente parte del código que ha sido programado con la intención de explicar las partes principales del mismo. El código en extenso se podrá encontrar en los anexos.

La primera parte del código inicia con las definiciones pre-procesador en las cuales se definen los protocolos de red que serán utilizados y las librerías específicas del sistema UAQ Meteo V1.3 (líneas 4 a 8). Enseguida en la línea número 10 se define el tiempo de espera en milisegundos al término del cual la BTU solicitará un nuevo dato a las RTUs. El archivo de cabecera que se incluye en la línea 11 permite modificar los parámetros de red como dirección IP, máscara de subred y puerta de enlace, así como el de la línea 12 incluye contiene las definiciones específicas para "uaq.c".

Tabla 10. Definiciones pre-procesador del servidor HTTP embebido.

Línea	Código
1	////////////////////////////////////
2	// HTTP SERVER TO REMOTE MONITOR METEOROLOGIC STATIONS
3	////////////////////////////////////
4	#define STACK_USE_ICMP 1
5	#define STACK_USE_ARP 1
6	#define STACK_USE_TCP 1
7	#define STACK_USE_HTTP 1
8	#define STACK_USE_UAQ 1
9	/// DEFINITIONS FOR UAQ TELEMTRY
10	#define WAITING_TIME 1000
11	#include "ccstcpip.h"
12	#include "uaq.h"

Como se ha mencionado anteriormente, el módulo del servidor HTTP embebido incluye una página Web, la cual reside en la memoria flash del microcontrolador PIC18F6722. La programación de esta página Web se hace en lenguaje HTML programado dentro del compilador de lenguaje C. El código es el siguiente.

Tabla 11. Código de la página Web del servidor HTTP embebido.

Línea	Código
13	const char HTML_INDEX_PAGE[]="
14	<HTML>
15	<BODY BGCOLOR=#FFFFFF TEXT=#000000>
16	<title>UNIVERSIDAD AUTONOMA DE QUERETARO - TELEMETRY WRMS METEO: V1.0</title>
17	<CENTER><H1>UNIVERSIDAD AUTONOMA DE QUERETARO</H1></CENTER>
18	<CENTER><H3>Wireless Remote Meteorologic Station: METEO V1.0</H2></CENTER>
19	<h3>METEOROLOGIC SENSORS</h3>
20	<h4>Direccion del viento 1: <!-- -->%7 V 
21	Velocidad del viento 1: <!-- -->%6 m/s 
22	Temperatura 1: <!-- -->%1 °C 
23	Humedad relativa 1: <!-- -->%2 %% 
24	Presion barometrica 1: <!-- -->%3 mB 
25	Precipitacion 1: <!-- -->%5 mm 
26	Radiación solar 1: <!-- -->%4 w/m2 
27	Punto de rocío 1: <!-- -->%8 °C 
28	Voltaje de la batería 1: <!-- -->%9 V  
29	Direccion del viento 2: <!-- -->%G V 
30	Velocidad del viento 2: <!-- -->%F m/s 
31	Temperatura 2: <!-- -->%A °C 
32	Humedad relativa 2: <!-- -->%B %% 
33	Presion barometrica 2: <!-- -->%C mB 
34	Precipitacion 2: <!-- -->%E mm 
35	Radiación solar 2: <!-- -->%D w/m2 
36	Punto de rocío 2: <!-- -->%H °C 
37	Voltaje de la batería 2: <!-- -->%I V </h4>
38	</BODY>

39	<SCRIPT TYPE="text/javascript">
40	<!--
41	document.write("Last reading: " + new Date());
42	//-->
43	</SCRIPT>
44	</HTML>
45	";

En la línea 13 se define una constante del tipo *char* el cual contiene el código en un lenguaje diferente al C del compilador. Éste lenguaje es especificado en la línea 14, HTML. En la línea 15 se define fondo blanco y color del texto negro. De las líneas 16 a 19 se encontrarán definidos los títulos y encabezados de la página Web. De las líneas 20 a 37 se encontrará la etiqueta que hace referencia a la lectura de cada uno de los sensores seguida por el signo porcentual y un número que indican al compilador el origen de los datos (los cuales se explicarán más adelante). La página Web también incluye un pequeño *script* de Java el cual muestra la fecha y hora la última hora de la adquisición, este código se muestra de las líneas 39 a 43.

En la Tabla 12 se muestra parte del código cuya función es solicitar que se transmitan los datos vía radiofrecuencia. En la línea 46 se hace referencia a una función contenida en "http.c" la cual permite relacionar el identificador '%id' con un proceso en específico, en este caso, un valor analógico o digital. Las variables se encuentran definidas de las líneas 47 a 50. El llamado a la función de la línea 51 permite habilitar las interrupciones para que cuando se reciba un dato en el puerto serial se interrumpa al microprocesador e inmediatamente se copie este dato a un buffer mapeado en la memoria RAM. En la línea 52 empieza la estructura que identifica compara el 'id' solicitado en la página Web para comenzar la solicitud de los datos. Esto es, si en el código de la página Web se encuentra la línea: *Temperatura 1: <!-- -->%1 °C<br>*, entonces el identificador será el que sigue al signo "%". Luego, en el código de la tabla 3 se encontrará de las líneas 54 a 61 que el servidor HTTP embebido envía vía serial el comando "T" seguido de un

“enter” (0x0d). Enseguida espera 1 segundo y copia los datos que se han recibido en el buffer A, a la cadena “new\_str”.

Tabla 12. Requisición de datos en el servidor HTTP embebido.

Línea	Código
46	int8 http_format_char(int32 file, char id, char *str, int8 max_ret) {
47	char new_str[10];
48	int8 len=0;
49	float i;
50	*str=0;
51	acquire_data(); //MT ENABLE INTERRUPTIONS
52	switch(id) {
53	case '1': //TEMPERATURE
54	str_flag='T';
55	printf("T");
56	delay_ms(10);
57	putc(0x0d);
58	len=10;
59	delay_ms(WAITING_TIME);
60	strcpy(new_str,bufA);
61	rs232_flag=0;
62	break;
63	...

Este procedimiento se realiza para la adquisición de cada uno de los sensores climatológicos, la relación es la siguiente: Temperatura “T”, Humedad Relativa “H”, Presión barométrica “P”, Radiación Solar “R”, Velocidad del viento “S”, Dirección del viento “D”, Precipitación pluvial “L”, Punto de rocío “C” y voltaje de la batería “V”. En la Tabla 13, se muestra el código para copiar estos datos a memoria y mostrarlos por consiguiente en la página Web.

Tabla 13. Copiado en memoria de los valores recibidos por el servidor HTTP embebido.

Línea	Código
64	if (len) {
65	if (len>max_ret) {len=max_ret;}
66	memcpy(str,new_str,len);
67	}
68	return(len);
69	}

En la Figura 27 se muestra la página embebida en el microcontrolador PIC18F6722. Esta página puede ser consultada de manera local a través de un equipo que tenga una dirección IP la cual se encuentre dentro del mismo segmento de red. La configuración de la IP del servidor HTTP embebido se programa en el archivo de cabecera “ccstcpip.h”. En la tabla 14 se muestra la función que contiene la dirección IP, la máscara de subred y la puerta de enlace configurada al iniciar la rutina principal.

Tabla 14. Configuración de los parámetros de red del servidor HTTP embebido.

Línea	Código
1	void IPAddrInit(void) {
2	//IP address of this unit
3	MY_IP_BYTE1=192;
4	MY_IP_BYTE2=168;
5	MY_IP_BYTE3=1;
6	MY_IP_BYTE4=10;
7	//network gateway
8	MY_GATE_BYTE1=192;
9	MY_GATE_BYTE2=168;
10	MY_GATE_BYTE3=1;
11	MY_GATE_BYTE4=1;
12	//subnet mask
13	MY_MASK_BYTE1=255;

14	MY_MASK_BYTE2=255;
15	MY_MASK_BYTE3=255;
16	MY_MASK_BYTE4=0;
17	}

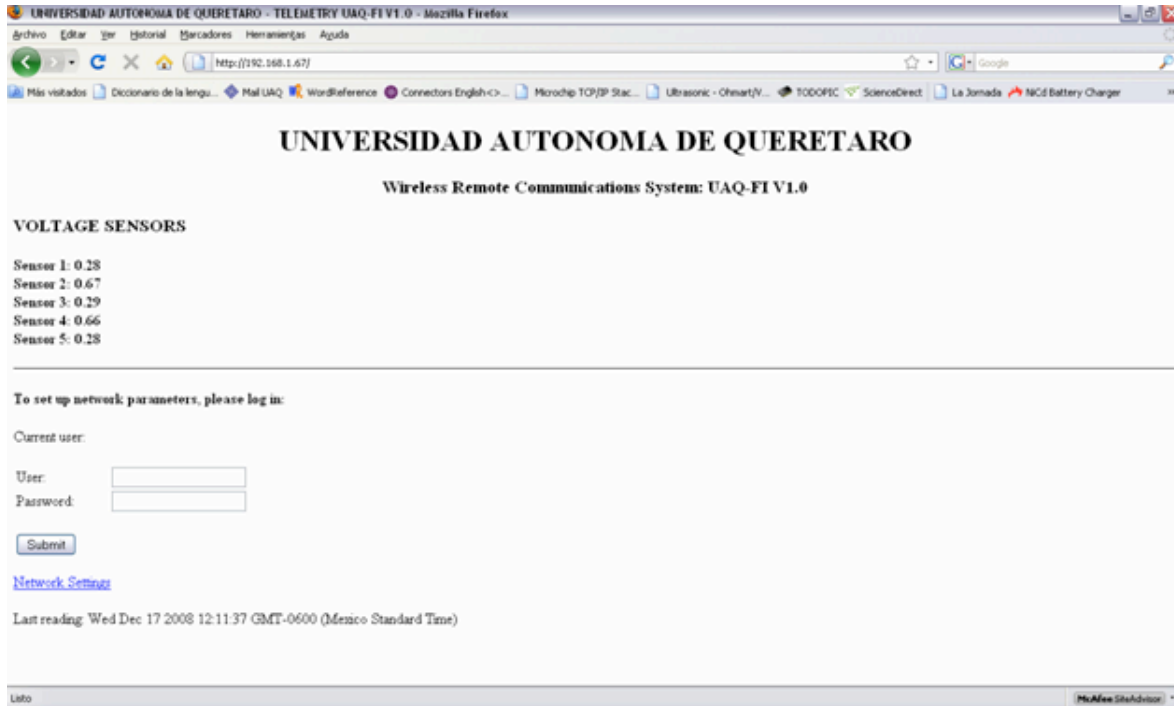


Figura 27. Página Web embebida en el microcontrolador PIC18F6722.

### 3.3 Servidor Central (CS)

El servidor central es el encargado administrar las peticiones de consulta/descarga de datos por parte de los usuarios finales, mantener los datos actualizados y de generar los historiales correspondientes. Cuando un usuario solicita acceso al sistema de telemetría a través de un explorador de Internet (FireFox, Internet Explorer, Chrome, Safari, entre otros), se genera una petición al CS, este se encarga de mostrar la información almacenada y de actualizarla cada 10 minutos.



### 3.3.1 Adquisición remota de datos

La adquisición remota de los datos involucra dos partes principales: el cliente (representado por el servidor HTTP Embebido de la BTU) y el servidor que contiene el software de administración y control de la red, éste último gestiona los datos meteorológicos de las estaciones remotas y alimenta de información a la base de datos para su futuro análisis y desplegado en tiempo real en el portal Web.

El usuario final tiene dos opciones de obtener los datos almacenados; una es vía Web a través del portal; allí encontrará los datos actualizados cada diez minutos, gráficas del comportamiento de las variables climatológicas en las ultimas 24 horas, registro máximo y mínimo del día, promedio de las mediciones del día y en el caso del viento, cuál fue la dirección en grados de la máxima velocidad registrada y dos, descargar los datos vía FTP directamente del servidor utilizando un cliente FTP.

En el servidor central se encuentra un programa en lenguaje php que solicita la información a las Unidades Terminales Base cada diez minutos. En este sentido, una rutina con un ciclo principal infinito monitorea cada segundo el reloj del servidor para controlar la temporización entre solicitudes de datos. Cuando se cumple la condición del transcurso de diez minutos se inicia una nueva solicitud. La condicionante está expresada de la siguiente manera:

$$(\text{min \% delay} == 0) \ \& \ (\text{seg} == 0)$$

Siendo "*min*" el valor obtenido del reloj del sistema operativo, la variable "*delay*" (de tipo entero sin signo) es previamente configurada por el usuario para determinar las pausas entre adquisiciones y "*seg*" es la variable donde se copia el valor de los segundos del reloj del sistema. Mientras no se cumpla la condición, el programa se mantiene actualizando las variables del tiempo "*min*" y "*seg*". En la

Figura 28 se muestra el diagrama de flujo del ciclo principal del software de adquisición del servidor central.

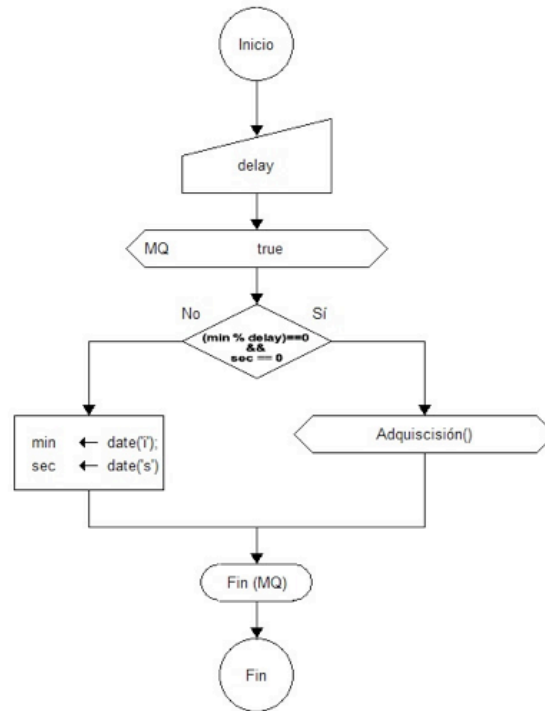


Figura 28. Diagrama de flujo de la solicitud de transmisión de datos en el servidor central.

Cuando se cumple la condición del diagrama de flujo anterior, el programa ejecuta la rutina de adquisición. Ésta consulta, en primera instancia el número de estaciones remotas almacenadas en la tabla de información específica de las BTU's, de esta manera, el ciclo comienza desde 1 hasta el valor devuelto por la consulta. Enseguida se realizan una serie de verificaciones previas a la conexión con la BTU:

1. Se efectúa la consulta del estado de una bandera para saber si la estación debe ser monitoreada o no (pueden deshabilitarse las estaciones en caso de que estén en mantenimiento, etc.). En caso de que la consulta regrese un valor falso, la leyenda "NEBLE" se agregará en la base de datos y se

imprimirá en todos los campos de la tabla de datos de la estación en cuestión.

2. Si la consulta anterior regresa un valor verdadero, se comprueba el estado de la conexión de Internet, esto se hace por medio de un “ping” o en otras palabras, el envío de paquetes de solicitud de eco y de respuesta de eco (ambos definidos en el protocolo de red ICMP). El “ping” se realiza al servidor de “Google” con dirección IP: 66.102.7.99. Si la petición devuelve un valor mayor de cero, se intentará establecer conexión con la BTU, de lo contrario se insertará en la base de datos la fecha y hora la leyenda "NNET" en todos los campos de la tabla de datos de la estación a monitorear. Esto significa que la conexión a Internet de donde se encuentra servidor se ha perdido.
3. Se realiza la petición de envío de información a la BTU por medio del método “get” luego, la respuesta debe contener los valores que fueron adquiridos por las RTUs. Con ayuda de este comando, se hace un rastreo de palabras claves o símbolos (dependiendo del tipo de estación a monitorear). La rutina de búsqueda permite consultar el tipo de estación de que se trata, puede ser del tipo UAQ (propietaria diseñada en la facultad de ingeniería) o una estación comercial (se realizaron pruebas con la marca Campbell Scientific). Una vez identificado el tipo de estación, se realiza un barrido del contenido de la página y se busca una palabra, o par de palabras claves como por ejemplo: “Humedad Relativa:”, se lee un espacio en blanco y se adquiere el número siguiente. En la siguiente figura se ilustra un diagrama de flujo al respecto.

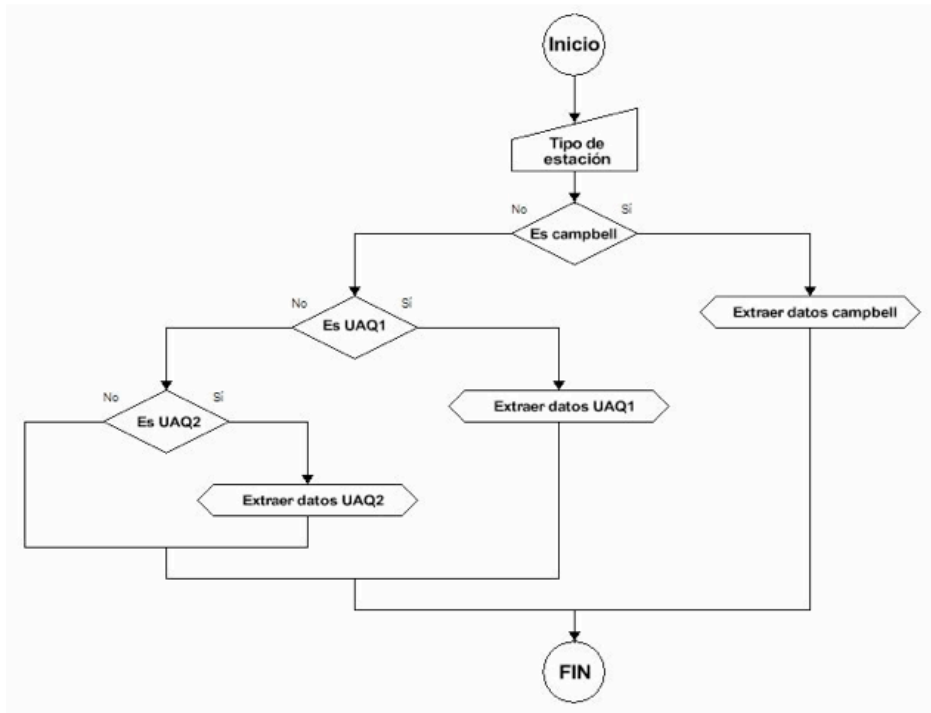


Figura 29. Diagrama de flujo de la rutina de adquisición.

### 3.3.2 Portal WEB de acceso a los datos

En el servidor central ubicado en la Facultad de Ingeniería de la UAQ, se encuentra almacenado un portal Web en donde se muestra en un mapa la ubicación de cada una de las Unidades Remotas (Figura 30).

Al dar clic en el mapa, se redirecciona al usuario a una página Web que presenta de manera numérica y gráfica los datos adquiridos por las RTUs. En la Figura 31 se muestra un ejemplo de los valores numéricos adquiridos por los sensores climatológicos y también el estado que guarda la batería del sistema.

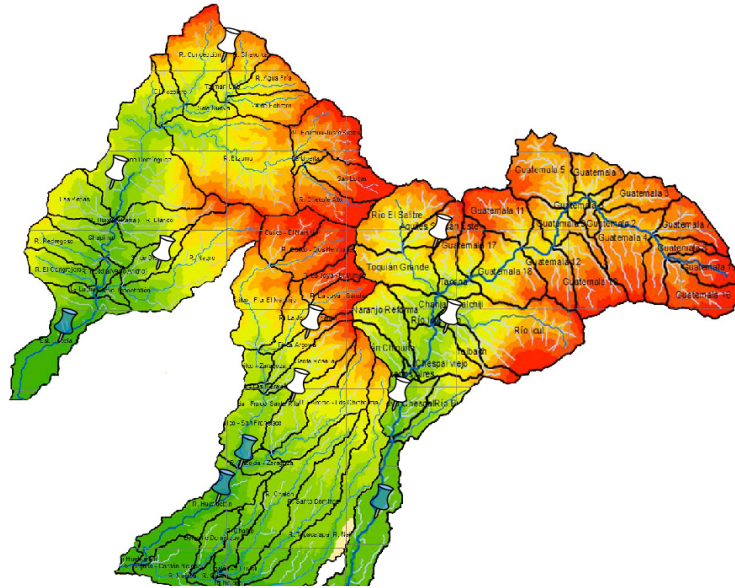


Figura 30. Ubicación de las estaciones remotas de sensores, caso de aplicación: Los altos de Chiapas.

**El último reporte fue obtenido el día de hoy:  
Jueves 24 de Septiembre del 2009 a las 03:36 hrs GMT-6**

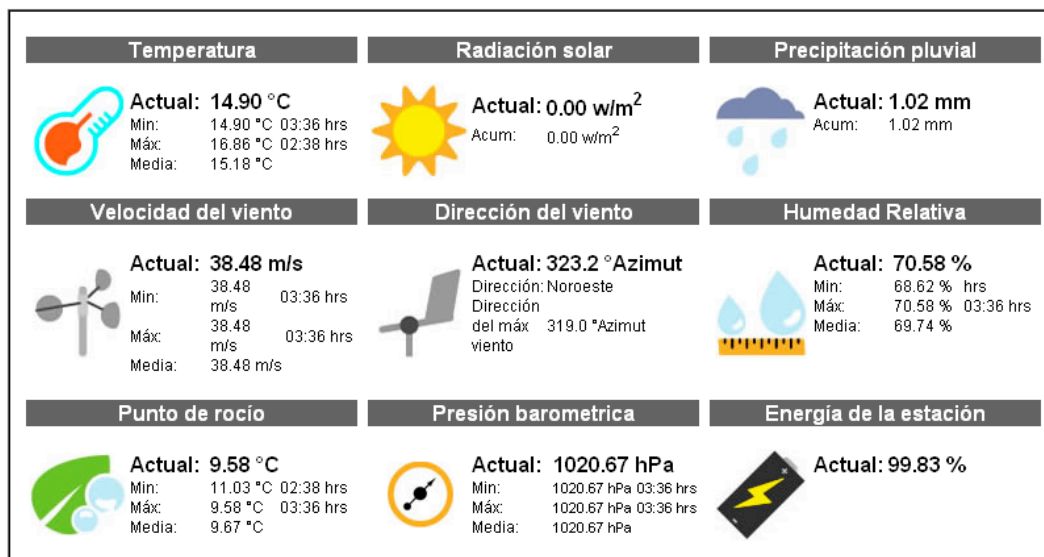


Figura 31. Interfaz numérica orientada al usuario final.

De igual manera, se programó una interfaz gráfica la cual muestra el historial de los datos de las últimas 24 horas en una gráfica del tipo circular. Ésto es, un dato

recién adquirido se grafica en el extremo derecho de la gráfica y, conforme avanza el día, esta medición se va recorriendo hacia la izquierda, de tal forma que transcurridas 24 horas, el dato adquirido ya no se muestra en la gráfica. En la Figura 32 se ilustra esta interfaz.

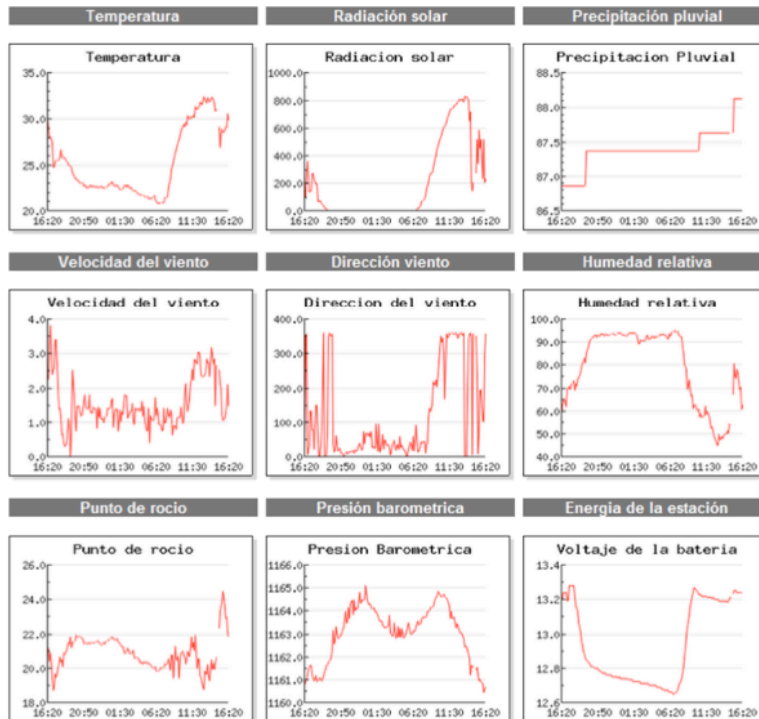


Figura 32. Interfaz gráfica orientada al usuario final.

Si el usuario desea tener una vista ampliada de las mediciones de una variable en específico, solo tiene que dar clic sobre la gráfica correspondiente y automáticamente se presentará una ventana emergente con la gráfica ampliada.

Adicionalmente a la presentación de los datos de manera numérica y gráfica, desde el portal Web es posible descargar los datos adquiridos por las RTUs desde una zona de descargas la cual se muestra en la siguiente figura. Ahí, el usuario debe seguir cuatro sencillos pasos para descargar información del servidor central.

1. Elegir los tipos de variables que se incluirán en el archivo txt.
2. Seleccionar el intervalo de fechas de interés.

3. Escoger (si es que existe) una hora de adquisición específica. (Si no la hay, el archivo se generará de las 00:00 a las 23:50.
4. Dar clic en “descargar datos”.

**1.- Escoja las variables a descargar:**

<input type="checkbox"/> Precipitación pluvial	<input type="checkbox"/> Velocidad del viento
<input type="checkbox"/> Temperatura	<input type="checkbox"/> Dirección del viento
<input type="checkbox"/> Radiación solar	<input type="checkbox"/> Humedad relativa
<input type="checkbox"/> Punto de rocío	<input type="checkbox"/> Presión barométrica
<input type="checkbox"/> Energía de la estación	<input type="checkbox"/> Todas

**2.- Escoja desde que fecha hasta que fecha desea los datos a descargar:**

<b>Desde:</b>			<b>Hasta:</b>		
Día	Mes	Año	Día	Mes	Año
01	Enero	2009	01	Enero	2009

**3.- Escoja desde que hora hasta que hora desea los datos a descargar:**

<b>Desde:</b>		<b>Hasta:</b>	
Hora	Minutos	Hora	Minutos
00	00	00	00

**4.- Por último oprima el botón de “Descargar datos”:**

Figura 33. Zona de descargas orientada al usuario final.

#### 4. PRUEBAS Y RESULTADOS

Por la complejidad del sistema, se han realizado pruebas en distintas etapas de desarrollo. Siguiendo la metodología de pruebas en el sistema de comunicaciones, las primeras pruebas consistieron en una comunicación punto a punto entre una BTU y una RTU en línea de vista. Se probó el sistema con un simple loop-back. Ésto es, se envía un dato de 8 bits desde la BTU hasta la RTU y ésta última regresa exactamente el mismo dato. Estas pruebas preliminares confirmaron una comunicación exitosa a una distancia de 8.77 Kilómetros en línea de vista desde las instalaciones de la Facultad de Ingeniería de la UAQ hasta el Cerro del Cimatario ubicado en la ciudad de Querétaro como se ilustra en la siguiente figura.

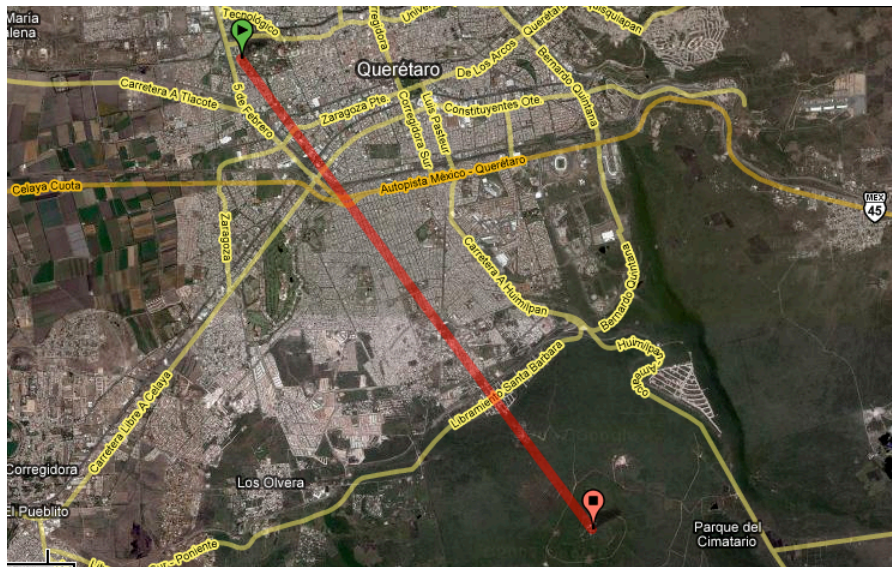


Figura 34. Pruebas preliminares de campo en La Ciudad de Querétaro.

Una vez realizadas las pruebas de comunicación entre un transmisor y un receptor en ambas direcciones, el siguiente paso fue realizar pruebas de comunicación entre más de dos RTUs. Así, se implementó una topología de red del tipo malla con direccionamiento dinámico. Esto quiere decir lo siguiente:



1. La topología de malla interconecta a todas las RTUs del sistema de comunicaciones.
2. Se implementó una arquitectura igual a igual (P2P), esto es, no existe jerarquía entre las RTUs de la red,
3. Se descubren nuevas rutas para llegar de un dispositivo a otro cuando se incorpora un nuevo dispositivo.
4. La unidad base puede transmitir en modo unicast o broadcast.

La prueba de laboratorio se realizó con 4 RTUs conformando la malla. El paquete de datos consistió en una secuencia de caracteres alfanuméricos en un lazo del tipo loop-back mencionado anteriormente. Los resultados obtenidos arrojaron una confiabilidad del 99.5% lo cual, en otras palabra significa que de 1,000 paquetes transmitidos tendremos 5 paquetes recibidos incorrectamente. Este análisis se realizó con un software de análisis llamado X-CTU el cual se muestra en la siguiente figura.

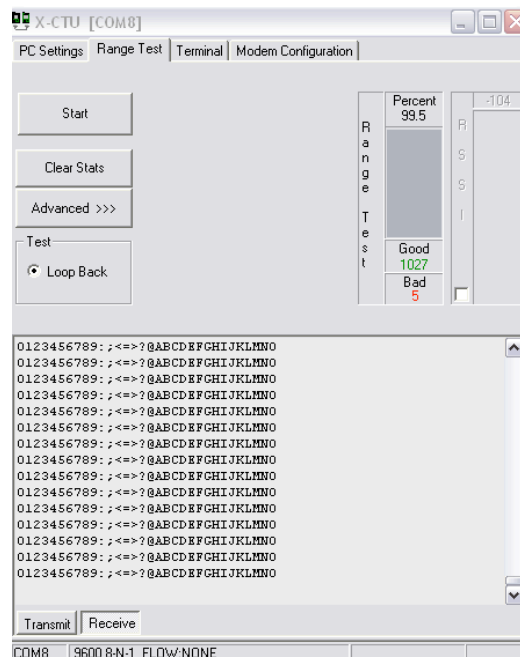


Figura 35. Prueba de rango en laboratorio de las RTUs con el software X-CTU.

Con los datos de confiabilidad obtenidos, se procedió más adelante a realizar pruebas de comunicación en campo en 11 puntos diferentes. Las pruebas realizadas consistieron básicamente en un estudio de línea de vista (LOS), en el cual se colocó una RTU y una BTU de prueba. La BTU inicia la transmisión y la RTU responde exactamente la misma información la cual le fue enviada. En este sentido, se calcula la relación paquetes (enviados)/(recibidos correctamente) y se obtiene un porcentaje. Para puntos de prueba en donde se tuvo línea de vista, el sistema de comunicaciones tuvo error de apenas 0.02%. Por ejemplo de Nueva Victoria a Agua prieta se transmitieron 1249 paquetes de datos, de los cuales 3 llegaron incorrectamente. Esto es, apenas el 0.02% de error, por lo cual la regla de decisión permite elegir el punto como apropiado para establecer una estación terminal remota. Los resultados de este estudio se muestran en la siguiente tabla.

Tabla 15. Resultados de las mediciones en campo con las unidades remotas diseñadas.

PUNTO DE PRUEBA	NOMBRE DEL PUNTO DE PRUEBA	COORDENADAS GPS	PUNTO DE REFERENCIA	LÍNEA DE VISTA	RSSI %	PAQUETES BUENOS	PAQUETES MALOS	FRECUENCIA (MHZ)
1	Las Cruces	577062 E 1695731 N	2	SÍ	99.8	1251	2	9000 CH(0)
2	Las Cruces	577062 E 1695731 N	1	SÍ	100	2000	0	9000 CH(0)
3	Nueva Victoria	563612 E 1690648 N	1	SÍ	99.8	1251	2	9000 CH(0)
4	Agua Prieta	568104 E 1682104 N	1	NO	0	0	1250	9000 CH(0)
5	Niquivil	582519 E 1684780N	1	NO	0	0	1250	9000 CH(0)
6	Nueva Victoria	563612 E 1690648 N	1	SÍ	100	2000	0	9000 CH(0)
7	Agua Prieta	568104 E 1682104 N	6	SÍ	99.8	1249	3	9000 CH(0)
8	Niquivil	582519 E 1684780N	6	NO	0	0	1250	9000 CH(0)
9	El Chaparron	584262 E 1672452 N	6	SÍ	100	2000	0	9000 CH(0)
10	Niquivil	582519 E 1684780N	9	NO	0	0	1250	9000 CH(0)
11	Union Roja	584125 E 1663839 N	9	NO	0	0	1250	9001 CH(0)

En la tabla anterior, se hace referencia en la quinta columna al parámetro RSSI, éste se refiere por sus siglas en inglés a la Índice de la Intensidad de Señal Recibida, parámetro que nos indica la fuerza con la que una señal recibida en términos de porcentaje.

Hasta este momento se han mostrado únicamente resultados de las pruebas de la transmisión de las RTUs en modo loop-back. Sin embargo, es importante evaluar el número de errores que se generan en el proceso de transferencia de datos desde una petición generada a través de un explorador de Internet hasta la correcta transmisión de los mismos por parte del sistema instalado consistente de la red de

radiofrecuencia más la red satelital. En este sentido hay un protocolo de comunicación llamado ICMP, protocolo de mensajes de control de Internet (del inglés Internet Control Message Protocol), éste es el sub-protocolo de control y notificación de errores del Protocolo de Internet (IP). Como tal, se usa para enviar mensajes de error, indicando por ejemplo que un servicio determinado no está disponible o que un router o host no puede ser localizado. Así, se monitoreo el enlace satelital en el punto “Agua prieta” considerado en Chiapas y los resultados se muestran en la siguiente gráfica.



Figura 36. Protocolos más activos durante la transmisión de datos. Fuente: Globalsat.

La figura anterior fue proporcionada por el proveedor de las terminales satelitales VSATs, Globalsat. En esta figura se muestra que el protocolo HTTP ocupa prácticamente un 97% del ancho de banda que ha sido utilizado y el protocolo ICMP apenas un 3% lo cual indica una alta confiabilidad en el enlace satelital y en el manejo de las transmisiones en el servidor HTTP embebido programado en el sistema de comunicaciones.

### *Afectaciones del clima a los sistemas satelitales*

Es importante mencionar que los sistemas satelitales tienen afectaciones por fenómenos naturales como la lluvia. La atenuación debido a la lluvia es la interferencia ocasionada por gotas de lluvia en señales electromagnéticas que viajan a través de la atmósfera. Cuando este fenómeno ocurre, la transmisión se debilita por los fenómenos de absorción y dispersión los cuales se resumen a continuación.

- **Absorción:** Ocurre cuando una onda electromagnética choca con una gota de lluvia y parte de la energía es absorbida por el agua y transformada en energía calorífica.
- **Dispersión:** Ocurre cuando las gotas de lluvia en la atmósfera ocasionan que las ondas electromagnéticas se desvíen de su trayecto inicial.

Los satélites de comunicaciones se ubican comúnmente en la órbita geoestacionaria ubicada a 35,786 kilómetros de la tierra (éste es el caso del proveedor de servicios satelitales Globalsat, contratado para el proyecto. Debido al recorrido que tienen las señales satelitales a través de la atmósfera, una célula de lluvia en cualquier parte del trayecto de la señal ocasionará alguna reducción en la potencia de la transmisión.

Sin embargo, por lo general la potencia de la señal se puede ver afectada de 2 a 3 minutos durante una precipitación pluvial promedio, y hasta 15 minutos para precipitaciones extremosas (caso extremadamente raro). Con base a estos datos proporcionados por el proveedor de servicios satelitales y considerando que el periodo de muestreo considerado es de 10 minutos y que el sistema solicita retransmisiones de datos cuando se detecta algún error en la comunicación, se decidió utilizar la banda Ku para la transmisión de los datos.

Como resultado, se ha logrado diseñar un sistema basado en una arquitectura modular que permite monitorear a gran distancia estaciones de sensores. Como caso de aplicación, se consideró la medición del clima en estaciones meteorológicas remotas. A continuación se muestran imágenes del sistema desarrollado.



Figura 37. Portal Web del Sistema de Monitoreo Meteo UAQ V1.3.



Figura 38. Pantalla de datos de la estación remota de Agua Prieta, Chiapas.

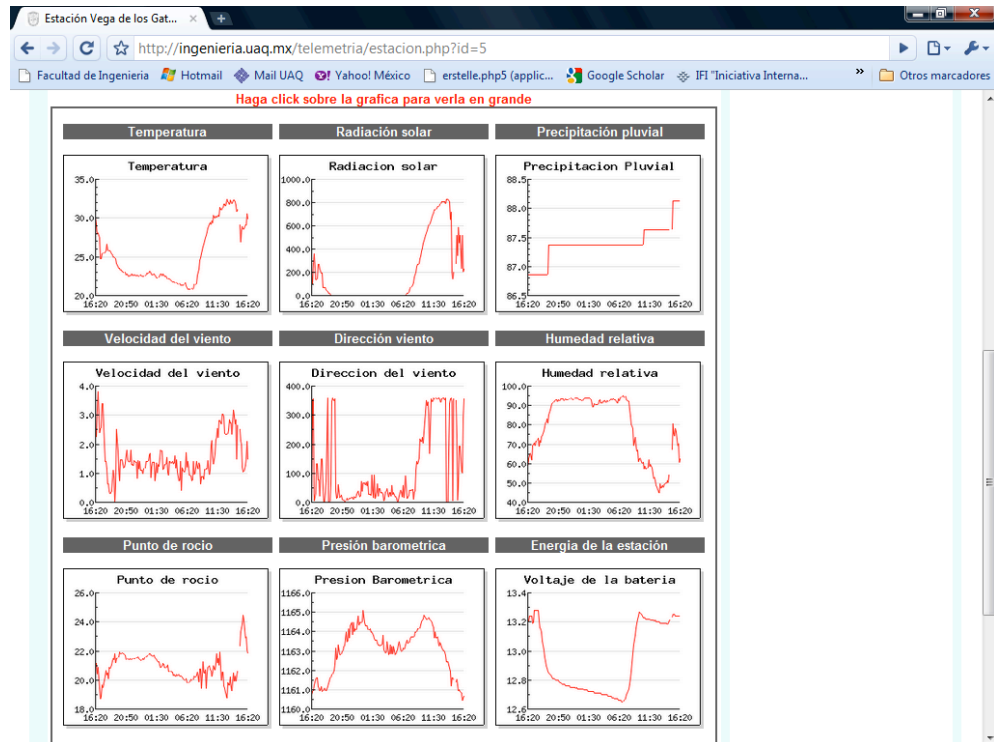


Figura 39. Gráficas 24-horas de los datos adquiridos por la estación remota de Agua Prieta, Chiapas.

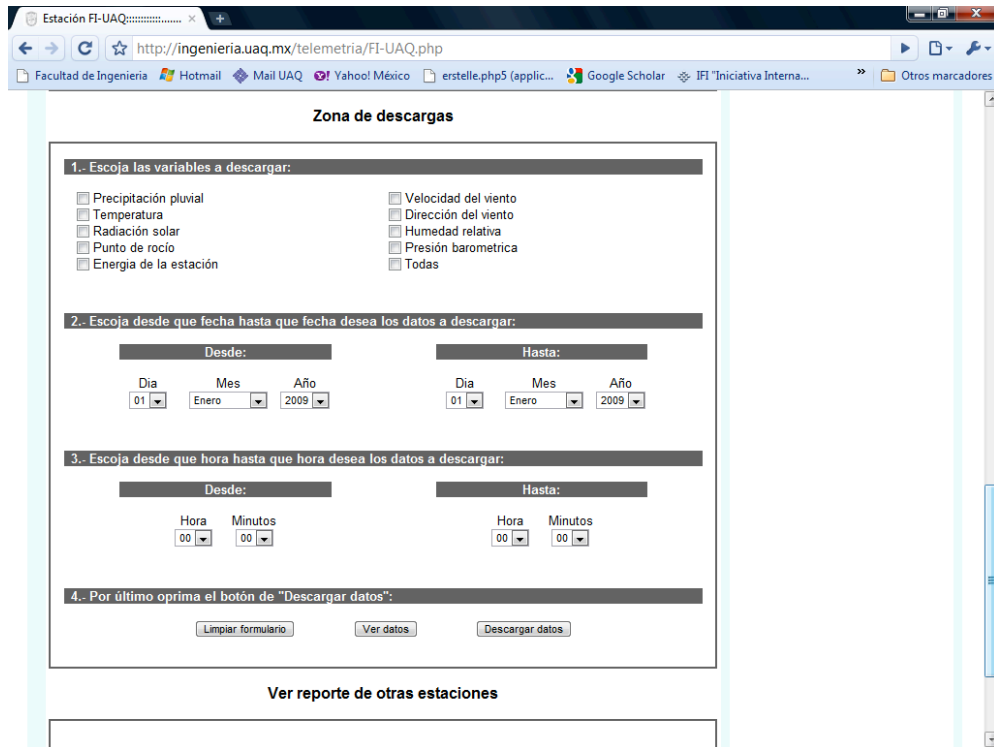


Figura 40. Vista de la zona de descargas del sistema Meteo UAQ V1.3.



a)



b)



c)

Figuras 41 a,b,c. Fotografías de las estaciones meteorológicas instaladas.

## CONCLUSIONES

Se ha desarrollado un sistema de monitoreo remoto de variables climatológicas de largo alcance. Las pruebas realizadas demuestran que el sistema tiene una alta calidad en el servicio (QoS) en las transmisiones satelitales (del 97%) en cuanto a posibles errores en la transmisión de los datos. En el estudio de línea de vista, los puntos analizados tuvieron una QoS del 99.8% en los puntos propuestos.

La red de estaciones de sensores instalado en Chiapas, el cual transmite en la banda ICM a 900 MHz reduce el costo de operación de la red en un 66% ya que evita el uso de un enlace satelital en cada una de las estaciones remotas.

Aunque el sistema de monitoreo está enfocado a la medición de variables climatológicas, se encuentra diseñado de manera modular, de tal forma que al cambiar los sensores de la red, se puede orientar hacia múltiples aplicaciones industriales, científicas y médicas y en general aplicaciones en donde el monitoreo remoto sea peligroso, inaccesible o económicamente inviable para el usuario.

La medición a distancia de las variables climatológicas en los altos de Chiapas, favorece la administración del sistema, reduce costos de operación (debido a la accidentada orografía de la zona) y constituye la base para que se desarrollen diversos estudios con fines climatológicos o para la prevención de pérdidas humanas en desastres naturales.



## BIBLIOGRAFÍA

- Akyildiz, F., Su, W., Y. S. y Cayirci, E., (2002). **Wireless sensor networks: A survey.** *Computer Networks: The International Journal of Computer and Telecommunications Networking* 2002, pp 393–422.
- Calvo, I. et al., (2006). **A methodology based on distributed object-oriented Technologies for providing remote access to industrial plants.** *Control Engineering Practice*. 8:975-990.
- Cembrano, G. et al., (2004). **Optimal control of urban drainage systems. A case study. A novel telemetric logging system for recording physiological signals in unrestrained animals.** *Control Engineering Practice*. 12:1-9.
- Chiara, B. et al., (2009). **An overview on wireless sensor networks technology and evolution.** *Sensors*, 9:6869-6896.
- Chong, C. y Kumar, S., (2003). **Sensor networks: Evolution, opportunities, and challenges.** *Proc. IEEE* 2003, 91:1247-1256.
- Guangming, S. et al., (2008). **A mobile sensor network system for monitoring of unfriendly environments.** *Sensors*, 8: 7259-7274.
- Guzmán, J., M. Berengel y F. Rodríguez, (2004). **Laboratorio remoto para el control de una maqueta de invernadero.** Universidad de Almería. España.
- Hua, S. et al., (2007). **Remote maintenance of control system performance over the Internet.** *Control Engineering Practice*, 5:533-544.
- I-Hai, P. et al., (2002). **Internet based monitoring and controls for HVAC applications.** *IEEE Industry applications*. 1:1-14.

- Iraqi, A. et al., (2003). **Design of a wireless communications module for telemetry in civil infrastructure monitoring.** *IEEE Transactions on Instrumentation and Measurement.* 52:1-10.
- Jae-Hoon, S., Woen-Sik H. y Young-Man, L., (2006). **Development of environmental control network system in greenhouse.** 22:1-12.
- Kazuya, T. et al., (2009). **Deployment design of wireless sensor network for simple multi-point surveillance of a moving target.** *Sensors,* 9:3563-3585.
- Kiyoshi H. et al., (2009). **Field servers and sensor service grid as real-time monitoring infrastructure for ubiquitous sensor networks.** *Sensors,* 9:2363-2370.
- Lal, D., et al., (2003). **Measurement and characterization of link quality metrics in energy constrained wireless sensor networks.** In *Proceedings of GLOBECOM*, San Francisco, CA, USA, 1–5 Octubre, 2003; pp. 446–452.
- Lowe, J.C. et al., (2007). **A novel telemetric logging system for recording physiological signals in unrestrained animals.** *Computers and Electronics in Agriculture.* 57:74-79.
- Muhammad, K. y Khaled, A., (2010). **Cryptanalysis and security improvements of ‘Two-factor user authentication in wireless sensor networks’.** *Sensors,* 10:2450-2459.
- Nadjib, A. et al., (2009). **A tabu search WSN deployment method for monitoring geographically irregular distributed events.** *Sensors,* 9:1625-1643.
- Ordoñez, H., y Rivera J., (2004). **Control por internet de una planta pasteurizadora.** *Artículos digitales de Nacional Instruments. Chih., México.*

- Ortiz, C., Zambrano S. y San Martín, C., (2004). **Automatización de un Invernadero.** *Artículos digitales de Nacional Instruments. México.*
- Peng J. et al., (2009). **Design of a water environment monitoring system based on wireless sensor networks.** *Sensors, 9:6411-6434.*
- Pulido-Calvo, I. et al., (2007). **Linear regressions and neural approaches to water demand forecasting in irrigation districts with telemetry systems.** *Biosystems Engineering. 2:283-293.*
- Sacile, R. (2007). **Remote real time monitoring and control of contamination in storage tank systems of petrol products.** *Journal of Cleaner Production, 13:1295-1301.*
- Supriyo C. y Havinga, P., (2009). **Improving temporal coverage of an energy-efficient data extraction algorithm for environmental monitoring using wireless sensor networks.** *Sensors, 9:4941-4954.*
- Víctor P., Jiménez G., y García, A. (2009). **Field measurements and guidelines for the application of wireless sensor networks to the environment and security.** *Sensors. 9:10309-10325.*
- Sardini, E. y Serpelloni, M., (2009). **Passive and Self-Powered Autonomous Sensors for Remote Measurements.** *Sensors, 9: 943-960.*
- Xu, N. (2002). **A survey of sensor network applications.** *IEEE Commun. Mag., 40: 102–114.*

# **ANEXOS**

## Anexo 1. Código fuente del servidor HTTP embebido

```
////////////////////////////////////
// HTTP SERVER TO REMOTE MONITOR METEOROLOGIC STATIONS
// A webserver that accepts GET and POST commands.
////////////////////////////////////

#define STACK_USE_ICMP 1
#define STACK_USE_ARP 1
// #define STACK_USE_DHCP 1
#define STACK_USE_TCP 1
#define STACK_USE_HTTP 1
#define STACK_USE_UAQ 1

/// DEFINITIONS OF UAQ TELEMETRY
#define WAITING_TIME 1000
#include "ccstcpip.h"
#include "uaq.h"

const char HTML_INDEX_PAGE[]="
<HTML>
<BODY BGCOLOR=#FFFFFF TEXT=#000000>
<title>UNIVERSIDAD AUTONOMA DE QUERETARO - TELEMETRY WRMS METEO: V1.0</title>
<CENTER><H1>UNIVERSIDAD AUTONOMA DE QUERETARO</H1></CENTER>
<CENTER><H3>Wireless Remote Meteorologic Station: METEO V1.0</H2></CENTER>
<h3>METEOROLOGIC SENSORS</h3>
<h4>Direccion del viento 1: <!-- -->%7 V<br>
Velocidad del viento 1: <!-- -->%6 m/s<br>
Temperatura 1: <!-- -->%1 ∞C<br>
Humedad relativa 1: <!-- -->%2 %%<br>
Presion barometrica 1: <!-- -->%3 mB<br>
Precipitacion 1: <!-- -->%5 mm<br>
Radiacion solar 1: <!-- -->%4 w/m2<br>
Punto de rocio 1: <!-- -->%8 ∞C<br>
Voltaje de la bateria 1: <!-- -->%9 V<br><br>

Direccion del viento 2: <!-- -->%G V<br>
Velocidad del viento 2: <!-- -->%F m/s<br>
Temperatura 2: <!-- -->%A ∞C<br>
Humedad relativa 2: <!-- -->%B %%<br>
Presion barometrica 2: <!-- -->%C mB<br>
Precipitacion 2: <!-- -->%E mm<br>
Radiacion solar 2: <!-- -->%D w/m2<br>
Punto de rocio 2: <!-- -->%H ∞C<br>
Voltaje de la bateria 2: <!-- -->%I V<br></h4>

</BODY>
<SCRIPT TYPE="text/javascript">
<!--
document.write("Last reading: " + new Date());
//-->
</SCRIPT>
```

```

</HTML>
";

const char HTML_SETTINGS_PAGE[]="
<HTML><BODY BGCOLOR=#FFFFFF TEXT=#000000>
<CENTER><H1>UNIVERSIDAD AUTONOMA DE QUERETARO</H1></CENTER>
<CENTER><H3>Wireless Remote Meteorologic Station: METEO V1.0</H2></CENTER>
<BR>
<h3>NETWORK SETTINGS</h3>
<BR>
<form method=GET>
<table>
  <tr ALIGN=center>
    <td>IP Adress:</td><td><input type=text size=3 maxlength=3 name=ip0 value=%w>.
    <input type=text size=3 maxlength=3 name=ip1 value=%x>.
    <input type=text size=3 maxlength=3 name=ip2 value=%y>.
    <input type=text size=3 maxlength=3 name=ip3 value=%z></td>
  </tr>
  <tr>
    <td>Subnet Mask:</td><td><input type=text size=3 maxlength=3 name=sn0>.
    <input type=text size=3 maxlength=3 name=sn1>.
    <input type=text size=3 maxlength=3 name=sn2>.
    <input type=text size=3 maxlength=3 name=sn3></td>
  </tr>
  <tr>
    <td>Gateway:</td><td><input type=text size=3 maxlength=3 name=gt0>.
    <input type=text size=3 maxlength=3 name=gt11>.
    <input type=text size=3 maxlength=3 name=gt22>.
    <input type=text size=3 maxlength=3 name=gt3></td>
  </tr>
  <tr><td><br><input type=submit value=\"Submit\"></td></tr>
</table>
</form>
<hr>

<P><A HREF=\"/\">Back to readings</A>
</BODY>
</HTML>
";

```

```

int32 http_get_page(char *file_str) {
  int32 file_loc=0;
  static char index[]="/";
  static char about[]="/settings";

  //printf("\n\nRequest %s ",file_str); MT
  if (strcmp(file_str,index)==0)
    file_loc=label_address(HTML_INDEX_PAGE);
  else if (strcmp(file_str,about)==0)
    file_loc=label_address(HTML_SETTINGS_PAGE);
  return(file_loc);
}

```

```

int8 http_format_char(int32 file, char id, char *str, int8 max_ret) {
    char new_str[10];
    int8 len=0;
    float i;
    int z; //To initialize buffers A to R
    *str=0;
    acquire_data(); //MT ENABLE INTERRUPTIONS

    switch(id) {
        case '1': //Dirección del viento
            str_flag='T';
            printf("T");
            delay_ms(10);
            putc(0x0d);
            len=10;
            delay_ms(WAITING_TIME);
            strcpy(new_str,bufA);
            rs232_flag=0;
            break;

        case '2': //Velocidad del viento
            str_flag='H';
            printf("H");
            delay_ms(10);
            putc(0x0d);
            len=10;
            delay_ms(WAITING_TIME);
            strcpy(new_str,bufB);
            rs232_flag=0;
            break;

        case '3': //Temperatura
            str_flag='P';
            printf("P");
            delay_ms(10);
            putc(0x0d);
            len=10;
            delay_ms(WAITING_TIME);
            strcpy(new_str,bufC);
            rs232_flag=0;
            break;

        case '4': //Humedad relativa
            str_flag='R';
            printf("R");
            delay_ms(10);
            putc(0x0d);
            len=10;
            delay_ms(WAITING_TIME);
            strcpy(new_str,bufD);
            rs232_flag=0;
            break;

        case '5': //Presión
            str_flag='L';
            printf("L");
            delay_ms(10);

```

```

putc(0x0d);
len=10;
delay_ms(WAITING_TIME);
    strcpy(new_str,bufE);
    rs232_flag=0;
break;

case '6':          //Lluvia
    str_flag='S';
    printf("S");
    delay_ms(10);
    putc(0x0d);
    len=10;
    delay_ms(WAITING_TIME);
        strcpy(new_str,bufF);
        rs232_flag=0;
    break;

case '7':

    str_flag='D';          //Dirección del viento
    printf("D");
    delay_ms(10);
    putc(0x0d);

    len=10;
    delay_ms(WAITING_TIME);
        strcpy(new_str,bufG);

        rs232_flag=0;
    break;

case '8':
    str_flag='C';          //Punto de Rocío
    printf("C");
    delay_ms(10);
    putc(0x0d);
    len=10;
    delay_ms(WAITING_TIME);
        strcpy(new_str,bufH);
        rs232_flag=0;
    break;

case '9':
    str_flag='V';          //Voltaje de la batería
    printf("V");
    delay_ms(10);
    putc(0x0d);
    len=10;
    delay_ms(WAITING_TIME);
        strcpy(new_str,bufI);
        rs232_flag=0;
    break;

```



////////// REMOTE METEOROLOGIC STATION

```
case 'A':          //Temperatura
str_flag='t';
printf("t");
delay_ms(10);
putc(0x0d);
len=10;
delay_ms(WAITING_TIME);
strcpy(new_str,bufJ);
rs232_flag=0;
break;
```

```
case 'B':          //Humedad
str_flag='h';
printf("h");
delay_ms(10);
putc(0x0d);
len=10;
delay_ms(WAITING_TIME);
strcpy(new_str,bufK);
rs232_flag=0;
break;
```

```
case 'C':          //Presión
str_flag='p';
printf("p");
delay_ms(10);
putc(0x0d);
len=10;
delay_ms(WAITING_TIME);
strcpy(new_str,bufL);
rs232_flag=0;
strcpy(new_str,"pERR");
break;
```

```
case 'D':          //Radiación solar
str_flag='r';
printf("r");
delay_ms(10);
putc(0x0d);
len=10;
delay_ms(WAITING_TIME);
strcpy(new_str,bufM);
rs232_flag=0;
break;
```

```
case 'E':          //Lluvia
str_flag='l';
printf("l");
delay_ms(10);
putc(0x0d);
```

```
len=10;
delay_ms(WAITING_TIME);
strcpy(new_str,bufN);
rs232_flag=0;
break;
```

```

case 'F':          //Velocidad del viento
    str_flag='s';
    printf("s");
    delay_ms(10);
    putc(0x0d);

    len=10;
    delay_ms(WAITING_TIME);
    strcpy(new_str,bufO);
    rs232_flag=0;
    break;

case 'G':
    str_flag='d';          //Dirección del viento
    printf("d");
    delay_ms(10);
    putc(0x0d);
    len=10;
    delay_ms(WAITING_TIME);
    strcpy(new_str,bufP);
    rs232_flag=0;
    break;

case 'H':
    str_flag='c';          //Punto de Rocío
    printf("c");
    delay_ms(10);
    putc(0x0d);
    len=10;
    delay_ms(WAITING_TIME);
    strcpy(new_str,bufQ);
    rs232_flag=0;
    break;

case 'I':
    str_flag='v';          //Voltaje de la batería
    printf("v");
    delay_ms(10);
    putc(0x0d);
    len=10;
    delay_ms(WAITING_TIME);

    strcpy(new_str,bufR);
    rs232_flag=0;
    break;
}

if (len) {
    if (len>max_ret) {len=max_ret;}
    memcpy(str,new_str,len);
}
return(len);
}

```

```

void http_exec_cgi(int32 file, char *key, char *val) {
    static char led1_key[]="led1";
    static char led2_key[]="led2";
    static char lcd_key[]="lcd";
    static char ip0_key[]="ip0";
    int8 v;
    printf("\r\nCGI FILE=%LD KEY=%S VAL=%S", file, key, val);

    if (strcmp(key,led1_key)==0) {
        v=atoi(val);
        if (v) {output_low(USER_LED1);}
        else {output_high(USER_LED1);}
    }

    if (strcmp(key,led2_key)==0) {
        v=atoi(val);
        if (v) {output_low(USER_LED2);}
        else {output_high(USER_LED2);}
    }

    if (strcmp(key,lcd_key)==0) {
        printf(lcd_putc,"%f%s",val);
    }

    if (strcmp(key,ip0_key)==0) {
        printf(lcd_putc,"%f%s",val);
    }
}

void main(void)
{
    MACAddrInit();
    IPAddrInit();
    init_user_io();
    lcd_init();
    printf(lcd_putc,"UAQ METEO V1.0");
    printf("IP Address = %u.%u.%u.%u\n\r",MY_IP_BYTE1,MY_IP_BYTE2,MY_IP_BYTE3,MY_IP_BYTE4);
    printf("Subnet Mask=
%u.%u.%u.%u\n\r",MY_MASK_BYTE1,MY_MASK_BYTE2,MY_MASK_BYTE3,MY_MASK_BYTE4);
    printf("Gateway =
%u.%u.%u.%u\n",MY_GATE_BYTE1,MY_GATE_BYTE2,MY_GATE_BYTE3,MY_GATE_BYTE4);*/

    StackInit();

    while(TRUE) {

        LCDTask();
        StackTask();
    }
}

```

## Anexo 2. Código fuente de la adquisición de datos de las estaciones terminales remotas.

```
/******  
*  
*          RS-232 Communications with dsPIC30F3014  
*          (Meteorologic Station)  
*  
*****  
* FileName:    UAQ.C  
* Dependencies:  uaq_meteo.h  
* Processor:    PIC18, PIC24F, PIC24H, dsPIC30F, dsPIC33F  
* Compiler:     CCS PCH  
* Company:     Universidad Autonoma de Queretaro  
* Author:      Manuel Toledano Ayala  
* Software License Agreement  
*  
* This software is owned by Universidad Autonoma de Querétaro  
*****/  
  
#include "uaq_defs.h"  
#int_rda  
  
rcv_data(){  
rs232_flag=0;  
if (str_flag=='T')  
{  
    bufA[i]=getchar(); //Read TEMPERATURE chars received trough RS232  
    if(bufA[i]==0x0d){  
        bufA[i+1]=0;  
        i=0;  
        rs232_flag=1;  
    }  
    else  
    {  
        i++;  
        if(i>=buf_length)  
            i=0;  
    }  
}  
if (str_flag=='H')  
{  
    bufB[i]=getchar(); //Read HUMIDITY chars received trough RS232  
    if(bufB[i]==0x0d){  
        bufB[i+1]=0;  
        i=0;  
        rs232_flag=1;  
    }  
    else  
    {  
        i++;  
        if(i>=buf_length)  
            i=0;  
    }  
}  
  
if (str_flag=='P')  
{  
    bufC[i]=getchar(); //Read PREASURE chars received trough RS232  
    if(bufC[i]==0x0d){
```

```

        bufC[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}

if (str_flag=='R')
{
    bufD[i]=getchar(); //Read SOLAR RADIATION chars received trough RS232
    if(bufD[i]==0x0d){
        bufD[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}

if (str_flag=='L')
{
    bufE[i]=getchar(); //Read RAIN chars received trough RS232
    if(bufE[i]==0x0d){
        bufE[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}

if (str_flag=='S')
{
    bufF[i]=getchar(); //Read WIND SPEED chars received trough RS232
    if(bufF[i]==0x0d){
        bufF[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}

if (str_flag=='D')
{
    bufG[i]=getchar(); //Read WIND DIRECTION chars received trough RS232
    if(bufG[i]==0x0d){
        bufG[i+1]=0;

```

```

        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}
if (str_flag=='C')
{
    bufH[i]=getchar(); //Read PUNTO DE ROCÃO chars received trough RS232
    if(bufH[i]==0x0d){
        bufH[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}
if (str_flag=='V')
{
    bufI[i]=getchar(); //Read VOLTAGE OF BATTERY chars received trough RS232
    if(bufI[i]==0x0d){
        bufI[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}

////////// REMOTE WIRELESS METEOROLOGIC STATION
if (str_flag=='t')
{
    bufJ[i]=getchar(); //Read TEMPERATURE chars received trough RS232
    if(bufJ[i]==0x0d){
        bufJ[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}
if (str_flag=='h')
{
    bufK[i]=getchar(); //Read HUMIDITY chars received trough RS232
    if(bufK[i]==0x0d){
        bufK[i+1]=0;

```

```

        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}
if (str_flag=='p')
{
    bufL[i]=getchar(); //Read PREASURE chars received trough RS232
    if(bufL[i]==0x0d){
        bufL[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}
if (str_flag=='r')
{
    bufM[i]=getchar(); //Read SOLAR RADIATION chars received trough RS232
    if(bufM[i]==0x0d){
        bufM[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}
if (str_flag=='l')
{
    bufN[i]=getchar(); //Read RAIN chars received trough RS232
    if(bufN[i]==0x0d){
        bufN[i+1]=0;
        i=0;
        rs232_flag=1;
    }
    else
    {
        i++;
        if(i>=buf_length)
            i=0;
    }
}
if (str_flag=='s')
{
    bufO[i]=getchar(); //Read WIND SPEED chars received trough RS232
    if(bufO[i]==0x0d){
        bufO[i+1]=0;
        i=0;
        rs232_flag=1;
    }
}

```

```

else
{
i++;
if(i>=buf_length)
i=0;
}
}
if (str_flag=='d')
{
bufP[i]=getchar(); //Read WIND DIRECTION chars received trough RS232

if(bufP[i]==0x0d){
bufP[i+1]=0;
i=0;
rs232_flag=1;
}
else
{
i++;
if(i>=buf_length)
i=0;
}
}
if (str_flag=='c')
{
bufQ[i]=getchar(); //Read PUNTO DE ROCÃO chars received trough RS232
if(bufQ[i]==0x0d){
bufQ[i+1]=0;
i=0;
rs232_flag=1;
}
else
{
i++;
if(i>=buf_length)
i=0;
}
}
if (str_flag=='v')
{
bufR[i]=getchar(); //Read VOLTAGE OF BATTERY chars received trough RS232
if(bufR[i]==0x0d){
bufR[i+1]=0;
i=0;
rs232_flag=1;
}
else
{
i++;
if(i>=buf_length)
i=0;
}
}
}
void acquire_data()
{
enable_interrupts(INT_RDA);
enable_interrupts(GLOBAL);
}

```