Tesis Doctoral

INTEGRACIÓN DE TECNOLOGÍAS INALÁMBRICAS EN SISTEMAS DE CONTROL Y MONITORIZACIÓN

Autor:

Juan Antonio Nazabal Urriza

upna
Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Directores:

Francisco Falcone Lanas

Ignacio Raúl Matías Maestro

“Todos somos muy ignorantes. Lo que ocurre es que no todos ignoramos las mismas cosas.”
Albert Einstein

AGRADECIMIENTOS

En primer lugar agradecer a los programas de ayudas al desarrollo de la investigación del Gobierno de Navarra, Departamento de industria (Ref. IIM010566.RI1) y al Ministerio de Economía y Competitividad a través del proyecto TEC2010-17805 y la Acción Internacional AIB2010NZ-00328 con Nueva Zelanda. Sin estas ayudas no hubiese sido posible la realización de este proyecto.

RESUMEN

En este trabajo se explora el concepto de '*Internet of Things*' y la integración de diferentes tecnologías de sensado en un mismo sistema de monitorización y control remoto. Básicamente la idea consiste en que todo esté conectado con todo a través de una red de transporte genérica aunque la red utilizada en este trabajo ha sido Internet debido a su alcance prácticamente global. Existen diferentes tecnologías de sensado a utilizar con sus diferentes ventajas e inconvenientes y en este trabajo s se propone el uso conjunto de algunas de ellas.

Por un lado se propone el uso de dispositivos inalámbricos basados en IEEE 802.15.4. Debido a que presentan un bajo consumo y poseen una baja tasa de transmisión los convierten en buenos candidatos a la hora de su utilización en redes de sensores. Además se comentan las características radioeléctricas de los mismos, así como los principales problemas radioeléctricos asociados, cobertura, calidad de enlace, etc. También se estudia su uso en sistemas de baja movilidad así como su uso colocados sobre el cuerpo humano.

Otra tecnología interesante a integrar es KNX. Este estándar domótico posee una amplia experiencia y lleva mucho tiempo en el mercado. Además existen multitud de diferentes modelos de dispositivos de diferentes fabricantes. En este trabajo se habla de cómo acceder a los datos del bus KNX mediante USB y también mediante redes IP utilizando el protocolo KNXNet/IP. Además se comentan los principales inconveniente del mismo y se propone un cambio en el protocolo para subsanarlos.

Finalmente una tecnología que proporciona grandes ventajas es la fibra óptica. Dentro del amplio abanico de posibilidades, este trabajo se centra exclusivamente en el uso de sensores basados en redes de difracción en fibra (FBG). Se comenta la dificultad de acceso a los datos de los sensores en equipos de fibra óptica debido principalmente a la falta de estandarización. Finalmente se muestra tres casos concretos de dispositivos ópticos y el mecanismo necesario para acceder a los datos de los sensores conectados a los mismos.

En último lugar se comentan varias implementaciones prácticas de diferentes sistemas de monitorización remota con la integración de dos o más de las tecnologías mencionadas anteriormente.

INDICE

.

CAPITULO 1 –MOTIVACION Y ESTADO DEL ARTE

1.1 Introducción

La gran cantidad de tipos de sensores y actuadores disponibles, tanto en lo que a tecnología de fabricación ser refiere como a la magnitud física que detecta o sobre la que actúa, hace de este un campo con amplias posibilidades. No existe el dispositivo perfecto sino que dependiendo de la tecnología utilizada en su fabricación presentará una serie de ventajas e inconvenientes que tendrán que tenerse en cuenta a la hora de utilizar unos u otros. Si a esto añadimos la posibilidad de acceso remoto a estos dispositivos, el abanico de posibilidades es prácticamente infinito. La motivación de este trabajo consiste en el acceso a los datos de sensores de diferentes tecnologías de manera conjunta y remota para posteriormente poder realizar con ellos acciones con valor añadido.
Existen multitud de interesantes propuestas que se basan de algún modo en el acceso a los datos registrados por sensores para después monitorizarlos, analizarlos o realizar acciones inteligentes sobre una serie de actuadores. A continuación se procederá a comentar brevemente algunas de las que más interés suscitan en la actualidad.

1.1.1 *E-Health*

El significado de este término engloba a todo uso de elementos electrónicos y tecnologías de la información en la asistencia sanitaria. Servicios como tele asistencia, tele consulta y gestión remota de pacientes entrarían dentro de esta categoría. Debido al inminente envejecimiento de la población cada vez existen más personas dependientes y lo que es más importante, muchas de ellas viven solas en su hogar. Por lo tanto por razones de salud y también debido al elevado coste sanitario asociado hace que esta idea sea muy interesante.

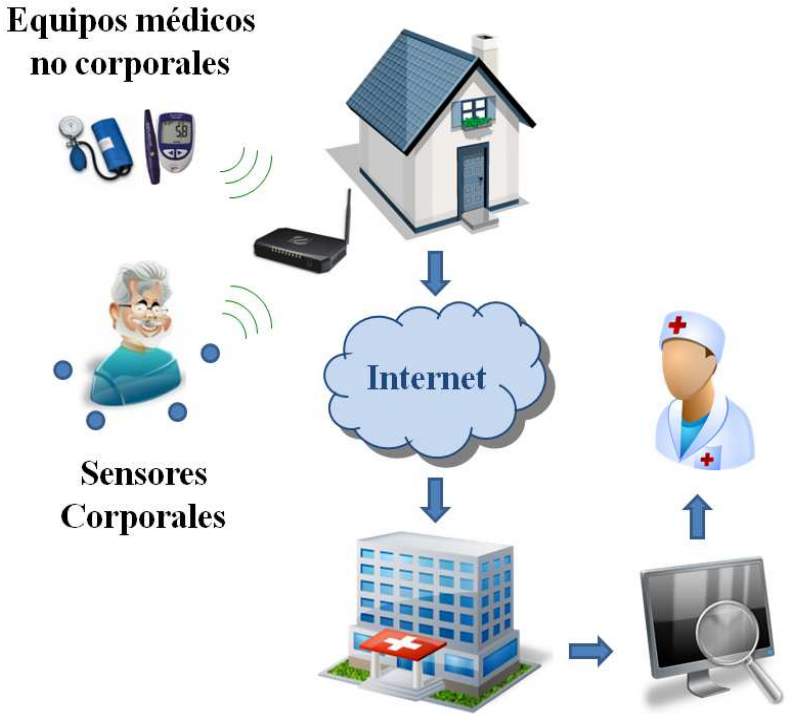


Fig. 1.1 Esquema ejemplo E-Health

Normalmente consiste en una serie de sensores que se colocan en el hogar o incluso el cuerpo de cada paciente que monitorizan su estado de salud y mandan los datos a un centro de control para el posterior análisis de los mismos. El servicio podría estar dotado de videoconferencia para poder contactar con el sujeto en caso de posibles problemas.

1.1.2 *Smart Grid*

Este término hace referencia a la red eléctrica inteligente. De manera similar que el concepto anterior, consiste en hacer un uso eficiente de la red eléctrica mediante el uso de tecnologías de la información. Gracias a su utilización aparece un amplio abanico de potenciales mejoras en aspectos como eficacia, eficiencia, flexibilidad, ajuste de carga, etc.
Debido a los elevados costes tanto del transporte de la energía eléctrica como sobre todo de su producción, la optimización de la red eléctrica es un aspecto de suma importancia. También cabe mencionar que no solamente la generación de energía eléctrica tiene un coste económico sino que además presenta un impacto ecológico haciendo de este aspecto doblemente importante.

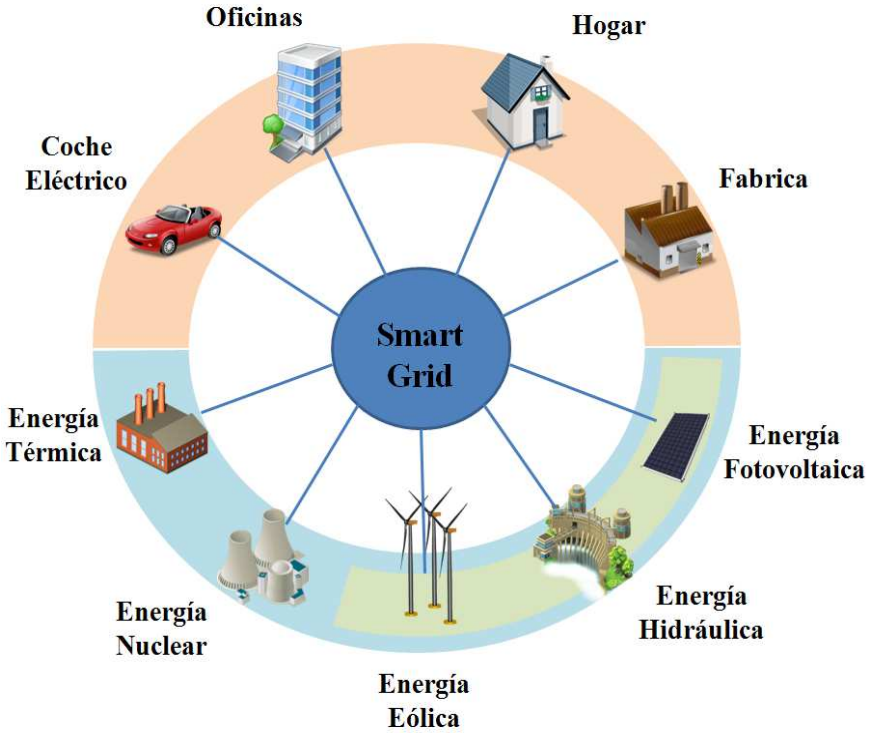


Fig. 1.2 Esquema ejemplo *Smart Grid*

1.1.3 *Smart Cities*

La complejidad de gestión de la infraestructura de una ciudad hace de esta una ardua tarea de complejidad directamente proporcional a la extensión y el número de habitantes de la misma.
Multitud de aspectos pueden ser optimizados mediante una conveniente monitorización y control, aspectos tales como el transporte, recogida de basuras, señalización vial, iluminación, etc.
El uso de tecnologías de la información representa sin duda alguna mejora en la realización de dicha tarea y la optimiza el uso de los recursos de dicha ciudad.
Una ciudad inteligente podría verse como un conjunto de subsistemas interconectados que trabajan de manera conjunta para optimizar los recursos disponibles.

Fig. 1.3 Esquema ejemplo *Smart City*

1.1.4 *Internet of Things*

Finalmente y llevando los conceptos de la ideas anteriormente mencionadas al límite de la generalidad de uso y a la distancia de comunicación entre dispositivos se tendría lo que se conoce como '*Internet of Things*'. Si se tratase de explicar esta idea, tan de moda en la actualidad, en una sola línea de texto quizás podría resumirse como la conexión de todo con todo utilizando la red global basada en '*Internet Protocol*' (IP) [1] más comúnmente conocida como simplemente Internet. Debido a que, utilizando una u otra tecnología de acceso a Internet, es posible acceder a la red desde prácticamente cualquier lugar del planeta, las posibilidades de esta idea son prácticamente ilimitadas.

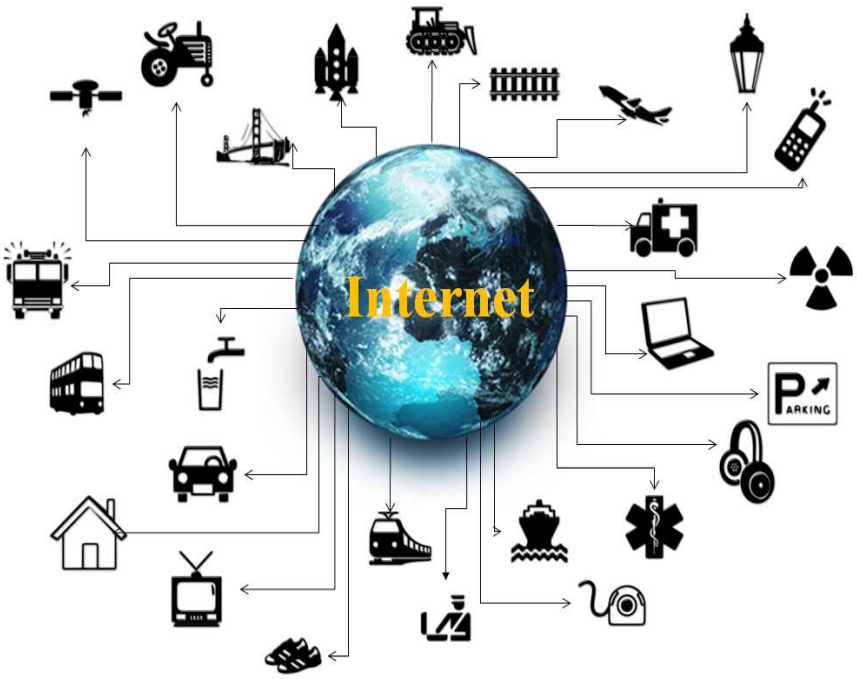


Fig. 1.4 Esquema ejemplo *Internet of Things*

1.2 Acceso

Para poder obtener remotamente los valores de sensores o realizar acciones en actuadores primero es necesario poder acceder a ellos. Teniendo en cuenta la gran multitud de diferentes tecnologías presentes en la actualidad y que, aunque muchas de ellas presentan algún tipo de estandarización, también existen muchas otras que no, esta tarea no es siempre sencilla. El diseño de un mecanismo para poder interconectar diferentes tecnologías en un solo sistema y de forma sencilla y eficaz se antoja vital.

1.3 Transporte

Una vez se dispone de los datos es necesario enviarlos al destino correspondiente. En principio, el diseño del sistema de acceso y transporte de datos debería ser independiente de las capa de red y transporte a utilizar para una máxima flexibilidad aunque en la realizad parece claro la utilización de Internet en la capa de red del sistema. Hay que tener en cuenta que el protocolo IP es no confiable y de tipo '*best effort*'. Esto significa que la red hace todo lo posible por que la información transmitida alcance su destino pero no se garantiza su llegada. En la actualidad la versión vigente del protocolo IP más comúnmente utilizada es la versión 4. Esta versión utiliza un tamaño de dirección, tanto para el origen como para el destinatario de los datos a intercambiar de 4 octetos. Esta limitación ha derivado a que en la actualidad existe el problema de que hay de escasez de direcciones IP públicas. En la siguiente versión del protocolo, la versión 6, esta contingencia está contemplada y resuelta utilizando direcciones de 8 octetos en lugar de 4, aunque hasta que no se encuentre en plena vigencia, hoy por hoy esta escasez de direcciones presenta una serie de problemas.
Para hacer frente a ella se utiliza asiduamente lo que se conoce como '*Network Address Translation*' (NAT) [2][3].

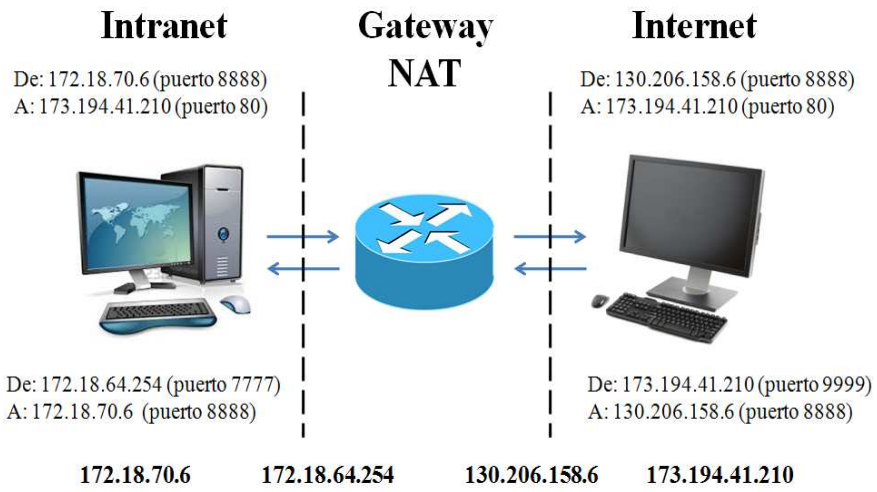


Fig. 1.5 NAT

Mediante este mecanismo de direccionamiento una red privada con direcciones IP no públicas se conecta a un equipo accesible desde el exterior que sí que tiene una dirección IP pública y que será el encargado de hacer de intermediario entre los equipos de la red privada y el exterior.
Es cuestión de crear, al nivel de la pasarela, una conversión de paquetes desde la red interna hacia la red externa. Por lo tanto, se configura cada equipo en la red privada que necesite acceso a Internet para que utilice una pasarela de NAT. Cuando un equipo de red envía una solicitud a Internet, la pasarela hace la solicitud en su lugar, recibe la respuesta y la envía al equipo que hizo la solicitud.

Debido a que la pasarela oculta completamente las direcciones internas en la red, el mecanismo de conversión de direcciones de red brinda una función segura. De hecho, para un observador externo de la red, todas las solicitudes parecen provenir de la dirección IP de pasarela.
De esta manera, en lugar de utilizar direcciones IP públicas para todos los equipos, únicamente es necesario que la tenga el equipo intérprete.
Hay que tener en cuenta además que cuando un equipo de la red privada crea una conexión al exterior a través de la pasarela normalmente el puerto utilizado no se respeta. Para mantener dicho puerto fijo hay que configurar la pasarela para enviar todos los paquetes recibidos a un puerto en particular, utilizando para ello la extensión NAT conocida como 'habilitación de puertos' o 'mapeo de puertos'.
Los proveedores de acceso a Internet (ISP) de varias tecnologías de acceso a Internet comúnmente utilizadas en la actualidad, tales como xDSL, UMTS, etc. utilizan este mecanismo y además asignan a los dispositivos direcciones dinámicas. Este tipo de direccionamiento se caracteriza por el hecho de que dicha dirección expira con el paso del tiempo y sin previo aviso, con mayor o menor frecuencia dependiendo del ISP, para dar paso a otra nueva dirección distinta de la anterior. Este hecho conlleva un problema ya que para poder acceder a un equipo conectado a Internet es necesario previamente conocer su dirección IP y esto es un problema si ésta cambia continuamente.
La solución que normalmente se utiliza para subsanar este problema es utilizar lo que se conoce como DNS dinámico (DDNS). El proveedor de este servicio asigna a una IP dinámica un nombre de red fijo y bien conocido mediante el cual se podrá acceder al equipo en todo momento. Para mantener la dirección IP asociada al nombre de red correspondiente actualizada en todo momento es necesario instalar en el equipo cuya dirección IP sea dinámica un pequeño programa que chequea periódicamente su dirección IP pública. Cuando se detecta un cambio en esta, la aplicación envía un aviso al correspondiente proveedor de DDNS y actualiza el valor de la dirección IP asociada a su nombre de red.
En cuanto al protocolo de transporte a utilizar en función de las características específicas del tipo de conexión que se desee usar la mejor opción consistiría en elegir entre '*User Datagram Protocol*' (UDP) [4] y '*Transmission Control Protocol*' (TCP) [5].
UDP es un protocolo de transporte sencillo y ligero, no orientado a conexión, sin ningún tipo de mecanismo de control de flujo. Los datos se envían sin controlar si cada paquete ha llegado, si ha llegado más de una vez o incluso si lo ha hecho de forma desordenada. Como ventaja se tiene que al no disponer de control de errores, desaparecen las esperas por paquetes perdidos y no existen retransmisiones con lo que consigue una mejor latencia.

UDP	TCP
No orientado a conexión	Orientado a conexión
Sin control de flujo	Control de flujo
Menor tamaño de cabecera	Mayor tamaño de cabecera
Envío inmediato	Envío dependiente del protocolo

Tabla 1.1 UDP vs TCP

Debido a que el tamaño de cabecera es pequeño, la proporción de datos útiles en cada datagrama y por lo tanto la eficiencia en el uso de la red es mayor. No existe procedimiento de establecimiento de conexión con lo que la latencia de inicialización de conexión es baja.
Finalmente, UDP posibilita el envío simultáneo a más de un destinatario, soportando la transmisión de tipo multidifusión.
TCP es un protocolo de transporte más complejo y pesado, orientado a conexión y con mecanismo de control de flujo asociado. El envío de información es fiable y se garantiza que los segmentos llegan y que lo hacen en orden.

Esta mayor complejidad hace necesario una cabecera con un tamaño mayor, con lo que la proporción de datos útiles enviada encada segmento es menor.
TCP presenta un buen rendimiento en un modem o una '*Local Area Network*' (LAN) pero no tanto en una conexión con pérdidas, de banda ancha o alta latencia como por ejemplo una conexión por satélite o un enlace T1.
Este protocolo presenta un mecanismo de establecimiento de conexión en el que se negocian los parámetros con las características de la conexión que se mantendrán activos mientras dure esta así como de fin de conexión. Esto hace que la latencia en la inicialización y en el fin de la conexión TCP sea mayor.
En resumen, se podría decir que en aplicaciones en el que la información a transmitir es crítica, es necesario utilizar TCP. En el caso en el que se prime la latencia y la eficiencia en el uso de la red, como bien podría ser el envío de contenido multimedia, el protocolo recomendable a utilizar sería UDP.

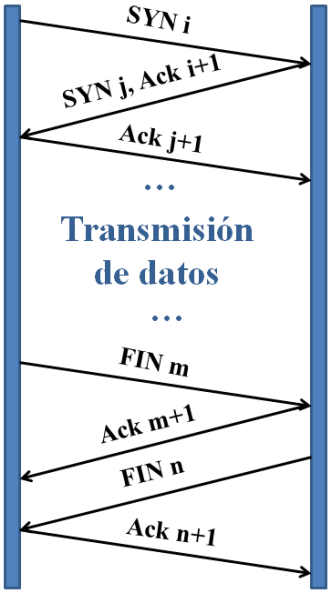


Fig. 1.6 Establecimiento de conexión TCP

1.4 IEEE 1451

El modelo IEEE 1451 [6]-[12] establece un conjunto de interfaces de comunicación de código abierto para comunicar transductores con microprocesadores y sistemas de instrumentación independientemente de la tecnología de red utilizada. Su finalidades el diseño de transductores inteligentes que puedan ser accedidos a través de diferentes redes de conexión. Consta de una familia que proporciona un conjunto de protocolos de comunicación para sistemas tanto cableados como inalámbricos.
Básicamente el estándar explica la conexión de un elemento llamado '*Transducer Interface Module*' (TIM) con un elemento de tipo '*Network Capable Application Processor*' (NCAP) conectados mediante un medio de transmisión especificado por otro miembro de la familia de estándares IEEE 1451.
Un elemento TIM es un modulo que contiene el interface, el acondicionamiento de señal, los conversores tanto analógico – digitales como digital-analógicos y generalmente el propio transductor. La complejidad de este elemento puede variar entre uno o varios transductores diferentes.
Un elemento NCAP consiste en el hardware y el software necesario para proporcionar funciones de pasarela entre los TIMs y la red de comunicación o procesador.

Fig. 1.7 IEEE 1451

La principal ventaja de su uso es que al ser un estándar implica una mayor facilidad a la hora de su implementación por diferentes entidades garantizando que si todas ellas se ciñen al mismo la interoperabilidad está garantizada. Además, es lo suficientemente completo como para cubrir la gran mayoría de transductores, es compatible con varios tipos de buses y redes y finalmente utiliza un protocolo binario eficiente.
Como principal desventaja está el hecho de su complejidad y baja adopción. En muchos casos prima la simplicidad y eficiencia en detrimento de la estandarización.

1.5 Conclusiones

El uso de la red Internet para comunicar diferentes transductores hace que sea posible acceder a ellos desde prácticamente cualquier lugar del planeta. Si a esto se añade la ingente cantidad de diferentes tipos de sensores y actuadores disponibles, el abanico de posibilidades es prácticamente infinito. Para poder realizar esta comunicación primero es necesario tener acceso a los datos, aspecto que no siempre es sencillo y después transportarlos de la manera más sencilla, general y eficaz posible. En función de la naturaleza de los datos a enviar es necesario escoger entre los diferentes protocolos de transporte disponibles y dependiendo del tipo de direccionamiento IP aparecen problemas que es necesario subsanar. En la actualidad existen soluciones disponibles, como el estándar IEEE 1451 aunque su elevada complejidad hace que presente baja aceptación y se decante por sistemas no estándar pero más simples y más eficaces.

1.6 Referencias

[1] RFC 791, Internet Protocol.
[2] RFC 2263, IP Network Address Translator (NAT) Terminology and Considerations.
[3] RFC 3022, Traditional IP Network Address Translator (Traditional NAT).
[4] RFC 768, User Datagram Protocol.
[5] RFC 793, Transmission Control Protocol.
[6] IEEE Std1451.0, 2007, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats, IEEE Computer Society.

[7] IEEE Std1451.1, 1999, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Network Capable Application Processor Information Model, IEEE Computer Society.
[8] IEEE Std1451.2, 1997, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Transducer to Microprocessor Communication Protocols & TEDS Formats, IEEE Computer Society.
[9] IEEE Std1451.3, 2003, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Digital Communication & TEDS Formats for Distributed Multidrop Systems, IEEE Computer Society.
[10] IEEE Std1451.4, 2004, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Mixed-Mode Communication Protocols & TEDS Formats, IEEE Computer Society.
[11] IEEE Std1451.5, 2007, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Wireless Communication Protocols & Transducer Electronic Data Sheet (TEDS) Formats, IEEE Computer Society.
[12] IEEE Std1451.7, 2010, IEEE Standard for a Smart Transducer Interface for Sensors and Actuators – Transducers to Radio Frequency Identification (RFID) Systems Communication Protocols and Transducer Electronic Data Sheet Formats, IEEE Computer Society.

CAPITULO 2 – INTEGRACION IEEE 802.15.4

2.1 Introducción

Las tecnologías de comunicación inalámbrica presentan la peculiaridad de que no utilizan un medio guiado de transmisión. Debido principalmente a este motivo no es necesaria la instalación de ningún tipo de cableado de comunicación a la hora de utilizar este tipo de redes. Este hecho supone una gran ventaja ya que facilita su instalación en edificios ya en construcción reduciendo la obra necesaria en gran medida. Además, son un gran candidato a utilizar cuando no es posible la realización de ningún tipo de obra, como bien pudiera ser el caso de edificios históricos o simplemente la instalación de un medio guiado de comunicación es demasiado costosa. El precio a pagar es sin embargo la utilización de una mayor cantidad de energía para realizar la transmisión de la información y la gran complejidad del canal radio, altamente dependiente del entorno. Finamente ha de tenerse en cuenta que el canal radio se comparte, tanto con otras tecnologías de comunicación inalámbrica que utilizan la misma banda frecuencial como con interferencias electromagnéticas de diversa índole.
Puede encontrarse gran cantidad de diferentes tecnologías inalámbricas de comunicación con sus diferentes ventajas e inconvenientes asociados pero este apartado se centra en exclusivamente en aquellas utilizadas en interiores de edificios.

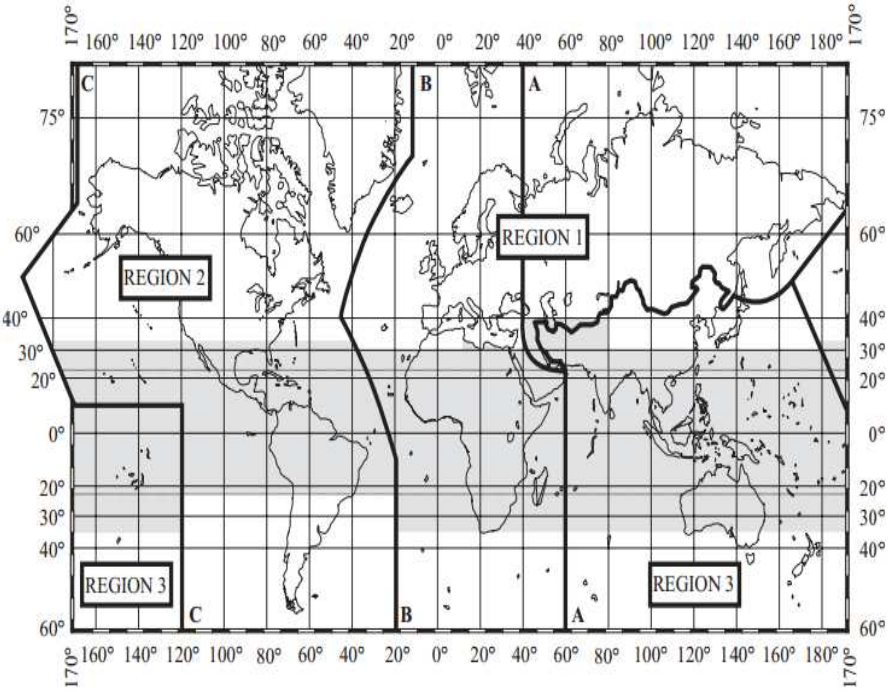


Fig. 2.1Regiones ITU-R [1]

Existe un tipo especial de bandas de frecuencia llamadas bandas '*Industrial, Scientific and Medical*' (ISM) para su uso internacional y gratuito originalmente en aplicaciones industriales, científicas y médicas. En la actualidad a crecido el uso en estas bandas de sistemas de comunicación de corto alcance y bajo consumo. Los dispositivos de comunicación que trabajen en este tipo de bandas deben tolerar las posibles interferencias producidas por equipos ISM que pueden estar presentes en ellas. El uso de estas bandas de frecuencias está regulado por el organismo

'*International Telecommunication Union*' (ITU) y la Fig. 2.1 muestra las diferentes regiones en que se divide el territorio. En la tabla 2.1 pueden verse varias de las bandas de frecuencias más utilizadas [1].
En la cualidad, la banda ISM más utilizada es posiblemente la de 2.4 GHz, debido básicamente a que puede ser usada en cualquier país y la anchura de banda disponible es bastante grande sin tener una frecuencia central extremadamente elevada. Dentro de esta banda existen multitud de diferentes tecnologías de comunicación inalámbrica que la utilizan.
En la tabla 2.2 se muestra una comparativa de las características principales de cuatro de las tecnologías más utilizadas. Las características tales como rango y consumo energético de los módulos de comunicación difieren entre distintos fabricantes con lo que realizar comparativas generales únicamente proporciona una visión global [2].

Rango Frecuencial	Ancho de Banda	Disponibilidad
13.553 MHz - 13.567 MHz (2)	14 kHz	Global
26.957 MHz - 27.283 MHz (2)	326 kHz	Global
40.660 MHz - 40.700 MHz (2)	40 kHz	Global
433.050 MHz - 434.790 MHz (1)	1.84 MHz	Región 1 (3)
902.000 MHz - 928.000 MHz (2)	26 MHz	Región 2
2.400 GHz - 2.500 GHz (2)	100 MHz	Global
5.725 GHz - 5.875 GHz (2)	150 MHz	Global
24.000 GHz - 24.250 GHz (2)	250 MHz	Global
61.000 GHz - 61.500 GHz (1)	500 MHz	Global

1) Uso sujeto a autorización especial de la administración
2) Uso sujeto a las provisiones mencionadas en RR No. 15.13
3) Excepto los países mencionados en RR No. 5.280

Tabla. 2.1 Bandas ISM

	ZigBee	Bluetooth	Wi-Fi	UWB
Estándar	802.15.4	802.15.1	802.11a/b/g	802.15.3a
Complejidad	Simple	Complejo	Muy Complejo	Complejo
Consumo	Bajo	Medio - Bajo	Alto	Alto
Nodos	> 65000	8	32	8
Extensibilidad	Árbol, Malla	*Scatternet*	ESS	Punto a punto
Tasa Binaria	250 Kbps	1 Mbps	54 Mbps	110 Mbps
Rango	10– 100 m	10 m	100 m	10 m
Ensanchamiento espectral	DSSS	FHSS	DSSS, CCK, OFDM	DS-UWB, MB-OFDM

Tabla 2.2.Comparativa características

Para su utilización en redes de sensores, aspectos como la simplicidad y sobre todo el consumo energético priman sobre otros como la velocidad de transmisión, debido principalmente a que la cantidad de datos a intercambiar es muy baja. Por lo tanto, para estos casos el uso de tecnologías

basadas en el estándar IEEE 802.15.4 [3] son las más indicadas. Además, la simplicidad de su protocolo implica una mayor facilidad el desarrollo de dispositivos que lo implementan, así como unos menores requisitos mínimos de hardware necesarios para su implementación.
Como puede verse en la Fig. 2.2, donde se muestran las características más importantes del estándar, existe la posibilidad de utilizar diferentes bandas frequenciales. Aunque las bandas de 868/915 MHz presentan un mayor alcance, su menor número de canales y sobre todo el hecho de que no sean utilizables en todos los países del mundo hace que la banda más comúnmente utilizada sea la de 2.4 GHz.

PHY (MHz)	Frequency Band (MHz)	Coverage	Data Rate (Kbps)	Number of Channels	RX Sensitivity	Modulation
868/915	868-868.6 902-928	ISM Europe ISM America	20 40	1 10	- 92 dBm - 92 dBm	BPSK BPSK
868/915 (optional)	868-868.6 902-928	ISM Europe ISM America	250 250	1 10	- 85 dBm - 85 dBm	ASK ASK
868/915 (optional)	868-868.6 902-928	ISM Europe ISM America	100 250	1 10	- 85 dBm - 85 dBm	O-QPSK O-QPSK
2450	2400-2483.5	ISM Worldwide	250	16	- 85 dBm	O-QPSK

Fig. 2.2 Características IEEE 802.15.4

Este estándar, como se puede apreciar en la Fig. 2.3, reparte la anchura espectral disponible en la banda de 3.4 GHz en forma de 16 canales de 2 MHz de anchura espectral, separados entre sí 5 MHz.

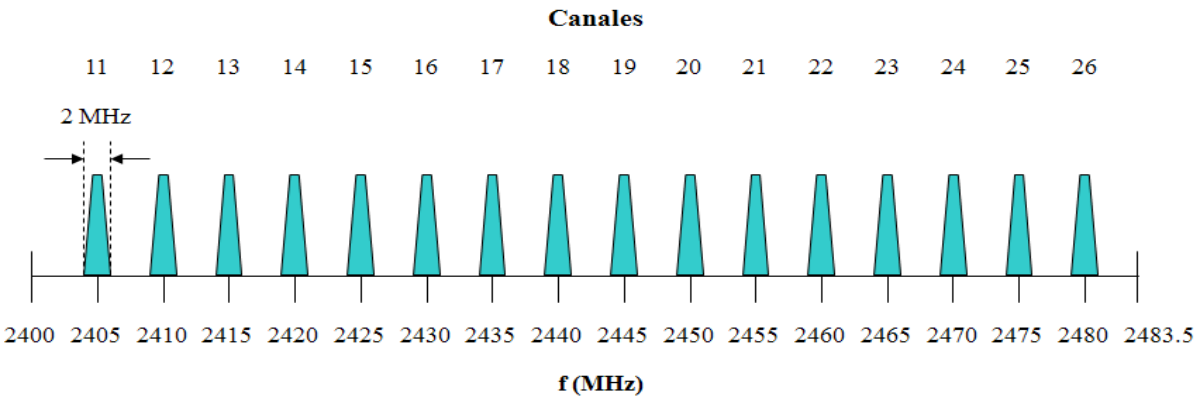


Fig. 2.3 Canales IEEE 802.15.4

2.2 Aspectos radioeléctricos

Toda tecnología de comunicación inalámbrica presenta una serie de problemas inherentes a la utilización de un medio tan hostil como es el canal radio. Al contrario que un medio de transmisión guiado, el canal radio es a menudo compartido por varios sistemas de comunicación diferentes en una misma banda frecuencial del espectro radioeléctrico, así como ruido procedente de otros dispositivos que no son de comunicación pero que generan interferencias dentro de la misma banda

frecuencial. A continuación se describen aspectos ligados tanto a fenómenos de radiopropagación como de seguridad propios de dichos sistemas inalámbricos.
Para un correcto intercambio de información entre dos dispositivos de comunicación inalámbrica es necesario que la señal enviada tenga un mínimo de potencia y calidad para que la información se transmita satisfactoriamente. Existen una serie de aspectos a tener en cuenta a la hora de determinar las distancias máximas entre dispositivos.

2.2.1 Propagación en el espacio libre

Hay que tener en cuenta que en toda fuente de radiación electromagnética, la potencia emitida se dispersa en función de la distancia a dicha fuente. Supóngase que la fuente de emisión dispone de una antena isotrópica (con un patrón de radiación esférico perfecto y una ganancia lineal unitaria).

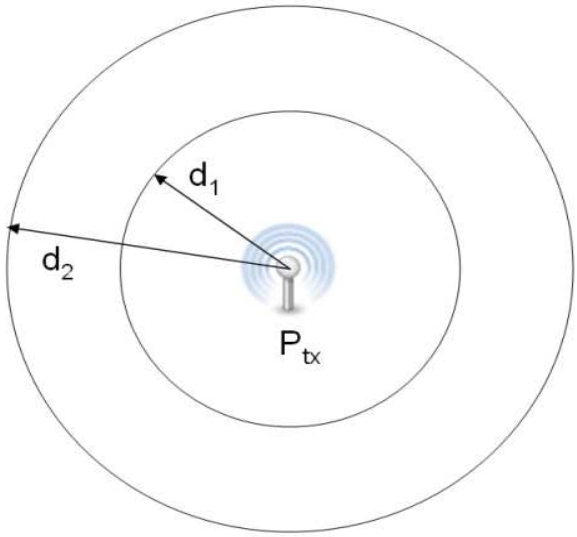


Fig. 2.4Dispersión de la potencia de la señal de transmisión

Como se puede apreciar en la Fig. 2.4, la densidad de potencia va disminuyendo de forma inversamente cuadrática conforme se va alejando de la fuente de emisión y por lo tanto la potencia útil que se dispone en recepción, dando lugar a pérdidas de propagación en el espacio libre. Si se supone que se tienen antenas isotrópicas tanto en transmisión como en recepción, las pérdidas en decibelios se calculan según la ecuación de Friis:

$$FSPL(dB) = 20\log d + 20\log f - 27.55 \quad (2.1)$$

Donde el parámetro ‘d’ es la distancia a la fuente en metros y ‘f’ es la frecuencia de la señal en MHz.
Hay que tener en cuenta que la ecuación de Friis únicamente se puede utilizar para campo lejano, es decir, para distancias mucho mayores que la longitud de onda de la señal. Para antenas de longitud menor a la mitad de la longitud de onda de la señal de emisión, se considera campo lejano a la región cuya distancia a la antena es mayor que dos veces la longitud de onda de la señal.
Para antenas de longitud mayor a la mitad de la longitud de onda, el campo lejano se determina en función de la distancia Fraunhofer calculada según la ecuación:

$$d_f = \frac{2D^2}{\lambda} \tag{2.2}$$

donde el parámetro 'D' es el diámetro de la antena y 'λ' es la longitud de onda de la señal transmitida.
Se considera campo lejano siempre y cuando la distancia a la antena es mayor a la distancia de Fraunhofer. Además, debe cumplirse que tanto el diámetro de la antena como la longitud de onda deben ser menores a la distancia de Fraunhofer.

2.2.2 Cambio de medio

Cuando una onda electromagnética atraviesa un medio y se propaga a través de otro de diferentes características, se producen una serie de fenómenos: reflexión, refracción-transmisión y absorción.
Como se puede apreciar en la Fig. 2.5, cuando una onda incide en un obstáculo, parte de la potencia es reflejada, parte es absorbida por el material y finalmente una parte es la lo atraviesa. Dependiendo de las características del obstáculo (grosor, tipo de material, etc.…) la señal refractada será de mayor o menor potencia.

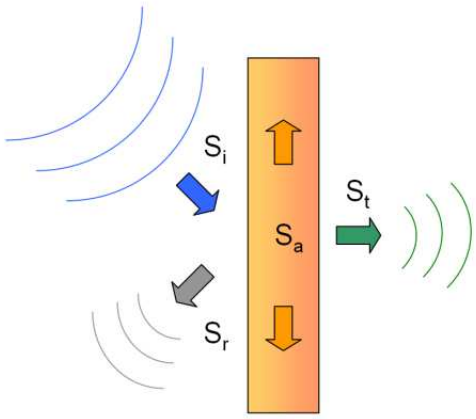


Fig. 2.5 Fenómenos de reflexión, refracción y absorción en un cambio de medio

En la tabla 2.3 se muestran los valores de atenuación típicos de diferentes materiales de construcción. Estos valores son muy importantes a la hora de realizar los cálculos de balance de potencia entre transmisor y receptor en comunicaciones inalámbricas.

Obstáculo	Atenuación (dB)
Pared de vidrio con marco de metal	6
Pared de oficina	6
Pared de bloque de hormigón	4
Puerta metálica en pared de oficina	6
Puerta metálica en pared de ladrillo	12
Ventana en pared de oficina	3
Ventana en pared de ladrillo	2
Cuerpo humano	3

Tabla 2.3 Valores de atenuación elementos construcción a 2.4 GHz

2.2.3 Difracción

Cuando una onda electromagnética incide en un obstáculo impenetrable que presenta esquinas o bordes agudos aparece el fenómeno conocido como difracción. Como se puede apreciar gráficamente en la Fig. 2.6, cuando el frente de onda llega al borde agudo del obstáculo, la onda incidente se distribuye alrededor y detrás del objeto.

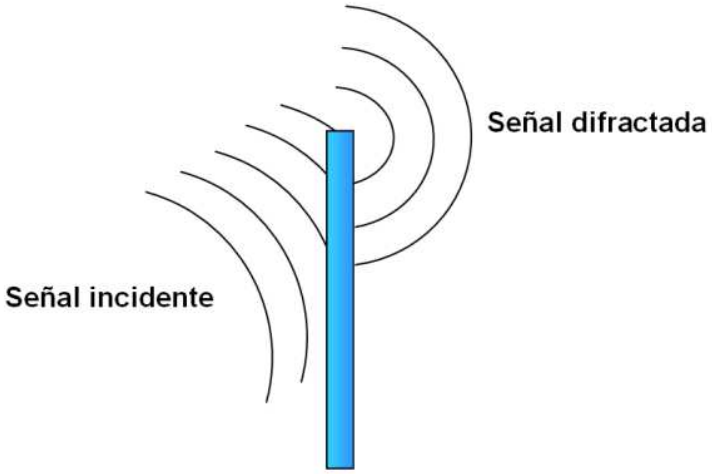


Fig. 2.6 Difracción

En función de la forma que presentan los objetos en los cuales se produce la interacción de la onda electromagnética, se calculan las pérdidas de difracción (objeto tipo filo de chuchillo, objeto redondeado y objeto agrupado). Dicho cálculo se basa en aproximaciones sobre las corrientes obtenidas mediante las integrales de Fresnel en la superficie de contacto.
Para tener en cuenta posible perdidas de potencia debidas a este fenómeno se utilizan diferentes márgenes de difracción dependiendo de la frecuencia de la señal y de las superficies iluminadas (ver Tabla 2.4).

Número de superficies iluminadas	Número de ventanas por superficie	Margen de difracción medio (dB)
1	1	1.3
	2	1.9
	4	2.8
	6	3.5
2	1	2.4
	2	3.4
	4	4.7
	6	5.7

Tabla 2.4. Márgenes de difracción a 2.4 GHz

2.2.4 Propagación multitrayecto

Cuando entre el emisor y el receptor de una transmisión inalámbrica existen obstáculos, la señal transmitida es reflejada debido a la presencia de dichos obstáculos y llega junto con replicas de ella y por diferentes caminos, atenuadas, desfasadas y retardadas, que a priori representan un ruido no

deseado en un sistema de comunicación. Estos diferentes caminos además son dependientes del tiempo (ver Fig. 2.7). Se puede observar como en función del instante de tiempo, se recibirán distintas componentes reflejadas por sus correspondientes diferentes caminos, que podrán variar en el siguiente instante de tiempo. Por lo tanto la señal se degrada debido a estas componentes y se tratan como perdidas por multitrayecto. Este tipo de pérdidas son muy dependientes del entorno por lo tanto son muy importantes en interiores de edificios, donde normalmente suelen existir multitud de obstáculos.

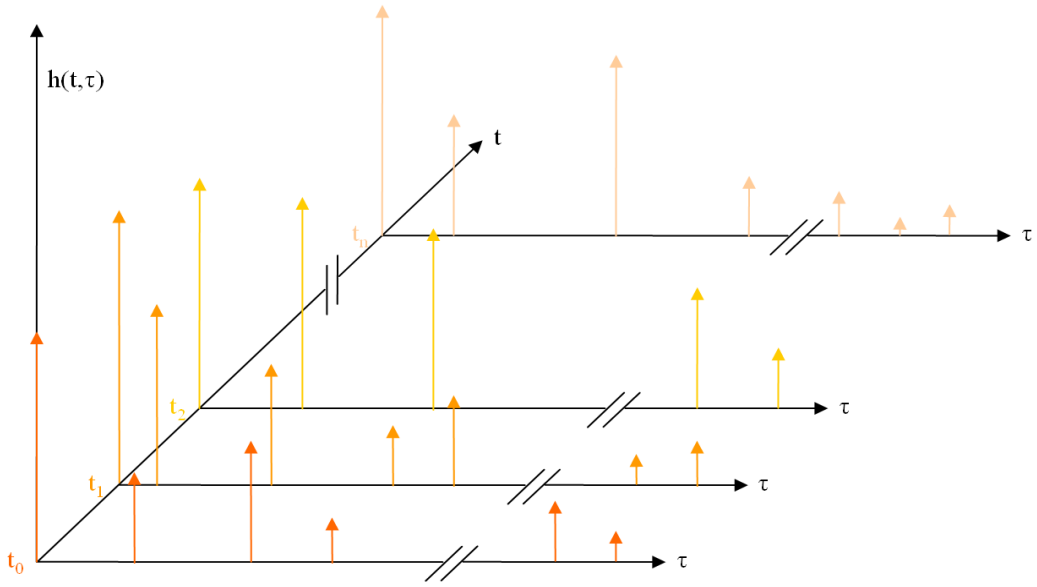


Fig. 2.7 Propagación multitrayecto

2.2.5 Atenuación por gases atmosféricos

La atenuación atmosférica por cielo claro (sin lluvia) se debe principalmente a efectos de absorción de energía de la onda de radio por efectos de resonancia en las moléculas de vapor de agua H_2O y de oxígeno O_2. En estas moléculas la distribución de los electrones en su interior presenta una asimetría eléctrica y se comportan como un dipolo eléctrico. En la Fig. 2.8 se puede ver una representación grafica de esto último para una molécula de H_2O.
Entonces, cuando una onda electromagnética incide en estas moléculas, parte de la energía interactúa con los dipolos que forman, haciéndolos vibrar y se disipa en forma de calor. La atenuación por cielo libre depende del ángulo de elevación de la antena, donde a ángulos bajos se generan mayores pérdidas y a ángulos altos menores pérdidas.

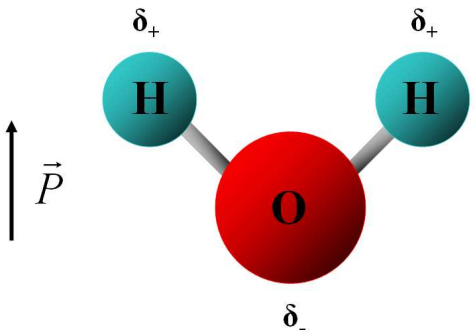


Fig. 2.8 Dipolo magnético creado por una molécula de agua

Teniendo en cuenta que las distancias máximas entre los diferentes dispositivos de una red de área personal dentro de un edificio de tamaño medio no suelen sobrepasar los cien metros, en la mayoría de los casos este efecto tiene poca influencia en los problemas de cobertura a frecuencias de trabajo en la banda de 2.4 GHz.

2.2.6 Interferencias

Toda comunicación inalámbrica debe compartir el canal radio con otras señales que se encuentren en su misma banda frecuencial de trabajo. Como se ha comentado anteriormente, la mayoría de las PANs utilizan en la actualidad la banda espectral de 2.4 GHz (2400 – 2483.5 MHz), utilizada para fines industriales, científicos y médicos, por la que es compartida por multitud de dispositivos. Las interferencias más comunes existentes en entornos de interior se describen a continuación.

2.2.6.1 Interferencia con IEEE 802.11 (Wi-Fi)

Como se aprecia la Fig. 2.9, los canales IEEE 802.11 [4] en ambos territorios se solapan con varios de los canales IEEE 802.15.4. Dependiendo de cada uno de estos territorios, se dispone de diferentes canales que se pueden utilizar sin interferencia mutua. En el caso de Europa, vemos que los canales sin solapamiento son 15, 16, 21 y 22 mientras que en EE.UU., los canales 15, 20, 25, y 26.

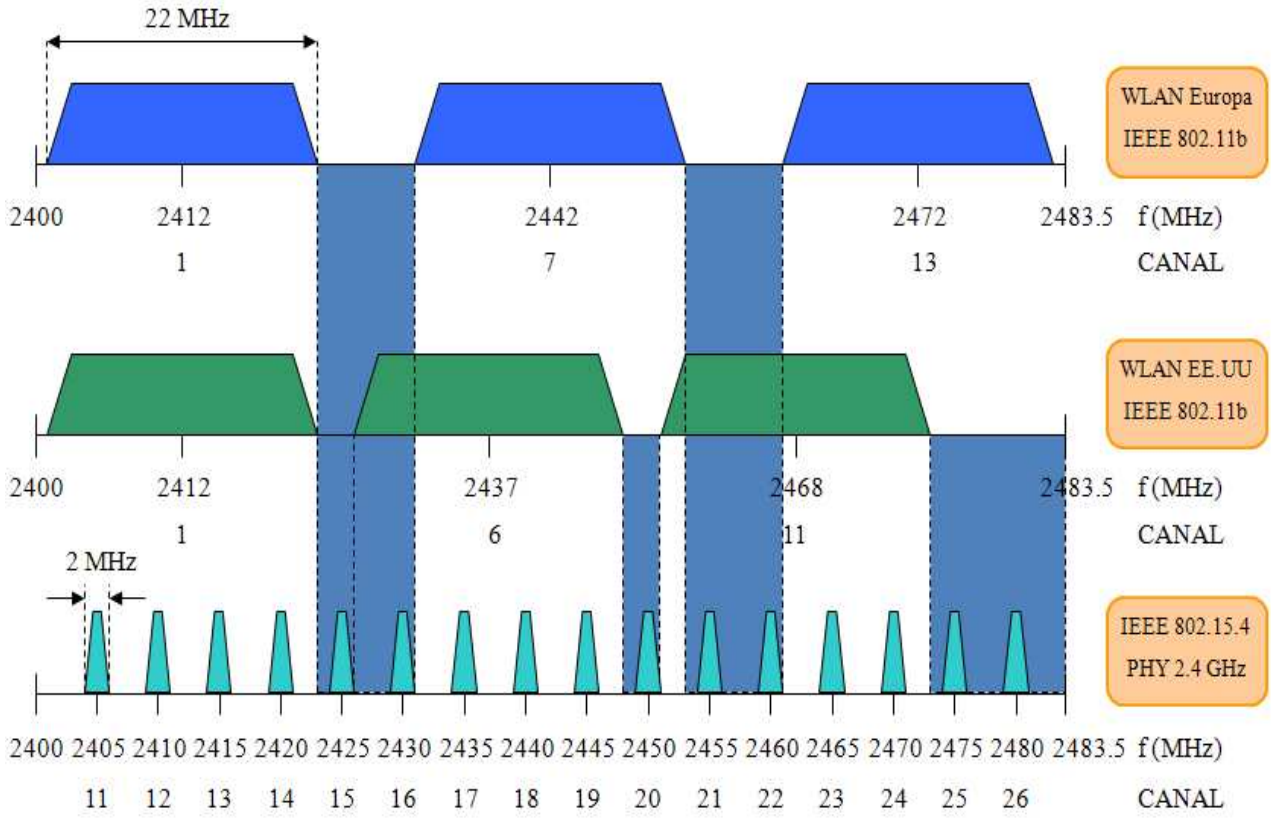


Fig. 2.9 Solapamiento de canales IEEE 802.11y IEEE 802.15.4

2.2.6.2 Interferencia con IEEE 802.15.1 (Bluetooth)

Como se ha comentado anteriormente, el estándar IEEE 802.15.1 [5] conocido comúnmente como Bluetooth realiza un salto en frecuencias Desde el punto de vista de las funcionalidades de la capa radio física, Bluetooth utiliza espectro ensanchado por salto de frecuencia y divide la banda de 2.4

GHz en 79 canales de 1 MHz (ver Fig. 2.10). De esta manera, este tipo de dispositivos salta entre estos 79 canales 1600 veces por segundo en una secuencia pseudo-aleatoria, lo que proporciona mayor inmunidad frente al ruido y posibles interferencias, sobre todo si son de banda estrecha. Precisamente debido al hecho de que se realice este salto frecuencial la interferencia con otros dispositivos que trabajen en la misma banda es menor, ya que el solapamiento espectral no será continuo.

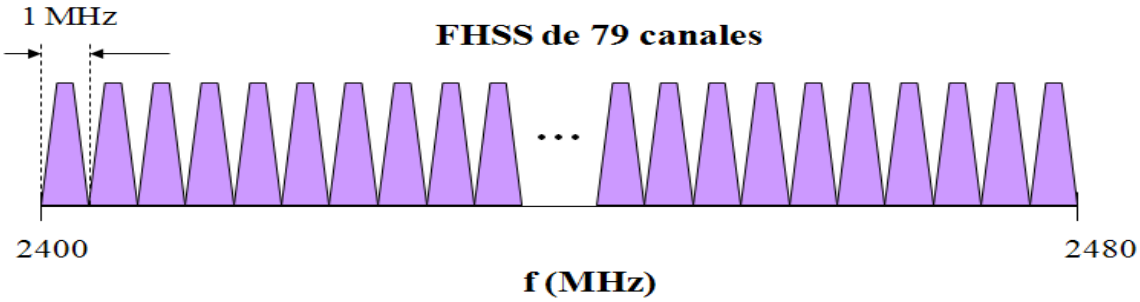


Fig. 2.10 Canalización espectral Bluetooth

2.2.6.3 Interferencia con ratones y teclados inalámbricos

Tanto los teclados como los ratones inalámbricos llevan en el mercado más de una década. Están diseñados para trabajar en enlaces de radiofrecuencia punto a punto, con un bajo rango de operación y un consumo muy reducido. Algunos de estos dispositivos trabajan en la banda de 2.4 GHz y son precisamente estos los que pueden ocasionar interferencias con otros dispositivos.

2.2.6.4 Interferencia con RFID

La identificación por radiofrecuencia o '*Radio Frequency IDentification*'(RFID) es un sistema de almacenamiento y recuperación de datos remoto cuyo propósito fundamental es transmitir la identidad de un objeto (con un número de serie asociado único) mediante ondas de radio. Esta tecnología que existe desde los años 40, se ha utilizado y se sigue utilizando para múltiples aplicaciones incluyendo casetas de peaje, control de acceso, identificación de ganado y tarjetas electrónicas de transporte pero es hasta hace unos años que se ha implantado a mayor escala. Los dispositivos RFID pueden ser pasivos o activos. El uso de dispositivos pasivos está limitado a aplicaciones de corto rango y no requieren de fuente de alimentación ya que se alimentan de una señal de radiofrecuencia generada por el lector. Por el contrario, los dispositivos activos pueden operar a distancias mayores pero necesitan alimentación externa, típicamente en forma de baterías. Existen también diferentes tipos de dispositivos en función de la frecuencia en la que trabajan, pudiendo ser esta de 135 KHz, 13.56 MHz, 433 MHz, 860-960 MHz o 2.45 GHz. Son precisamente estos últimos, activos y trabajando en la banda de 2.4 GHz los que pueden interferir en otras comunicaciones. Los parámetros de comunicación inalámbrica RFID en la banda de 2.4 GHz se describen en el estándar ISO/IEC 18000-4 [7].

2.2.6.5 Interferencia con teléfonos inalámbricos

Los teléfonos inalámbricos llevan años en el mercado y han ganado mucha popularidad. Los primeros de estos dispositivos trabajaban a 46 MHz, 49 MHz, 400 MHz o 900 MHz, pero en los últimos años muchos fabricantes han migrado a productos con frecuencias de trabajo de 2.4 GHz o 5 GHz.
No existe un protocolo estándar en este tipo de dispositivos y cada fabricante define sus propios protocolos de comunicación de radiofrecuencia.

Básicamente existen dos tipos de teléfonos inalámbricos, los analógicos y los digitales. Los primeros trabajan a una frecuencia siempre fija mientras que muchos los digitales (que están sustituyendo a los analógicos) utilizan técnicas como ensanchado por salto de frecuencia o espectro ensanchado por secuencia directa.

2.2.6.6 Interferencia con luces fluorescentes

Las luminarias fluorescentes, también llamadas tubos fluorescentes, necesitan para su funcionamiento un elemento fundamental utilizado para la cantidad de corriente, el balasto. Existen dos tipos diferentes: el balasto convencional o inductivo y el balasto electrónico. El primero trabaja a la frecuencia de la línea de potencia y las interferencias producidas se pueden eliminar por medio de filtros. Este balasto, a diferencia del balasto inductivo, se trata de de un circuito electrónico con semiconductores que genera dos bajas tensiones para encender los filamentos de los extremos, y una alta tensión de alta frecuencia aplicada entre los extremos. Los armónicos producidos en el proceso exceden el rango de los MHz y pueden llegar a causar interferencias en la banda de trabajo de 2.4 GHz.

2.2.6.7 Interferencia con hornos microondas

En la actualidad el uso de hornos microondas está muy extendido y extraño es el caso de una cocina que no disponga de uno de ellos. Los hornos microondas utilizan un dispositivo llamado magnetrón que genera cientos o miles de vatios de energía electromagnética que se utiliza para calentar comida dentro del horno. Los hornos microondas operan a 2.45 GHz en sus cercanías, justo en medio de la banda de 2.4 GHz. Estos hornos están diseñados para contener y aprovechar toda la potencia generada en la cavidad interior y disponen de aislamiento electromagnético para evitar que la energía escape. Dependiendo de la calidad de este aislamiento, se dejará pasar más o menos energía al exterior y por lo tanto en nivel de las interferencias que pueda llegar a causar depende fuertemente del mismo.

2.3 Módulos de comunicación XBee

Los modulo de comunicación inalámbrica basados en IEEE 802.15.4 que se ha utilizado en este trabajo son los modelos XBee del fabricante Digi. Básicamente existen dos modelos de XBee: el modelo normal y el Pro. Ambos son compatibles pin a pin. El modelo Pro tiene un mayor alcance en interiores (típicamente unos 90 m) en comparación con el modelo normal (unos 30 m) con un coste de un mayor consumo de corriente. Es necesario recalcar que el alcance máximo en interiores es altamente dependiente del entorno debido principalmente a la atenuación y a las componentes multitrayecto debidas a los obstáculos existentes entre dispositivos.
Se pueden comprar con diferentes configuraciones de antenas con diferentes características. Como puede apreciarse en la Fig. 2.11 de izquierda a derecha, las antenas disponibles son las de tipo '*whip*' (1.5 dBi), tipo '*chip*' (-1.5 dB) y también se puede comprar modelos para antena externa con conectores de tipo u.FL o RPSMA.

Fig. 2.11 Configuraciones de antena disponibles

A la hora de la elección de la configuración de antena, además de la ganancia de la misma hay que tener en cuenta el diagrama de radiación de la misma.
En la Fig. 2.12 pueden apreciarse los diagramas de radiación tipos proporcionados por Digi [8] para las configuraciones de antena de tipo '*whip*' y '*chip*' respectivamente.
Dependiendo de la aplicación, se deberá utilizar un modelo de antena u otro.
Si se desea conseguir la máxima ganancia de antena posible la mejor opción es usar una antena externa y utilizar un modulo con conector RPSMA o u.FL.
Puede verse como la antena de tipo 'chip' es la que peores prestaciones presenta pero es también la que proporciona una versión final del modulo más compacta.
Cuando se pretende embeber este tipo de módulos de comunicación esta es quizás la opción más usada y es la que se ha utilizado en este trabajo.

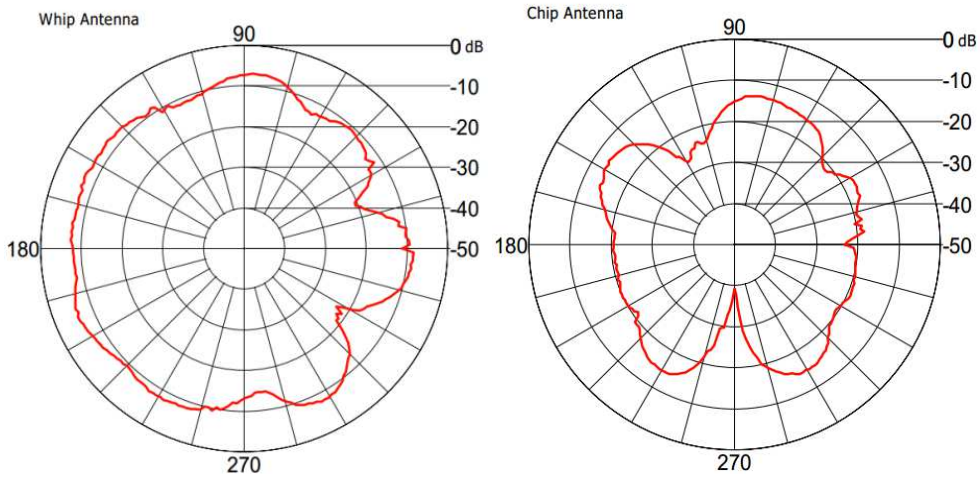


Fig. 2.12 Diagramas de radiación

El XBee, tanto el tipo normal como el Pro, puede ser de tipo 'Series 1' [9] (basados en tecnología del fabricante Freescale) o 'Series 2' [10] (basados en tecnología del fabricante Ember) y son tecnologías incompatibles entre sí.
Los 'Series 1' pueden llevar un firmware IEEE 802.15.4 o también DigiMesh. Esta última es una tecnología propietaria de Digi y comparable a ZigBee aunque no compatible con ella [11]. Los 'Series 2' solo pueden llevar firmware ZigBee. Cabe mencionar que en los 'Series 2' se puede instalar el firmware que implementa el perfil ZigBee SE ('*Smart Energy*').
Todos los módulos XBee pueden trabajar en dos modos diferentes: en modo transparente o en modo API.En modo transparente los módulos actúan como una conexión serie inalámbrica y los datos transmitidos llegan tal cual al receptor, siendo compatible con cualquier dispositivo serie. Si se utilizan los dispositivos en modo API los datos se encapsulan en tramas codificadas según un formato propietario de Digi y aparecen funcionalidades nuevas. La más interesante es sin duda la posibilidad de activar una serie de pines tanto como entradas analógicas o digitales y realizar un muestreo y envío de los datos o como salidas digitales. De esta manera en un mismo componente se tienen integra comunicación con I/O.
Hay que tener en cuenta que el rango de entrada en las entradas analogías es de un mínimo de cero voltios y un máximo de 3.3V para los 'Series 1' y de 1.2 V para los 'Series 2'. También es importante mencionar que el conversor analógico-digital presenta 10 bits de precisión.
El formato de las tramas se puede ver en la Fig. 2.13.
Es necesario recalcar que el formato de la estructura que transportan estas tramas varía entre los 'Series 1' y los 'Series 2', con lo en principio un programa que funcione para uno no funcionara para el otro.

Los XBee se programan de un modo limitado pero muy simple basado en el envío de una serie de comandos AT. Existe la posibilidad de enviar dicho comandos localmente a través de puerto serie del dispositivo o inalámbricamente, mediante comandos remotos.

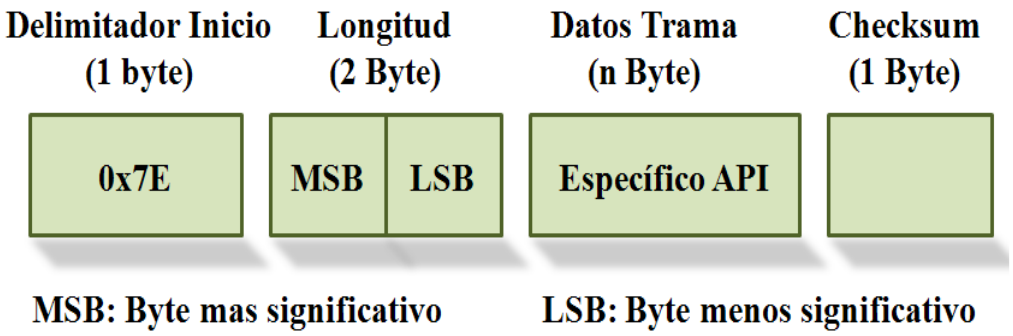


Fig. 2.13 Formato Trama modo API

Cuando una trama en modo API transporta datos el formato de los mismos puede verse en la Fig. 2.14. El primer byte indica el número total de muestras presente en la estructura de datos. Los dos bytes siguientes indican que canales están activos y si son analógicos o digitales. Los dos bytes siguientes se envían únicamente si la trama transporta datos digitales y contiene el valor de los mismos. Si la trama transporta solo valores analógicos, estos dos bytes se omiten. Finalmente, por cada valor analógico transportado se envían dos bytes: el primero con los bits más significativos y el segundo con los menos significativos del valor de 10 bits devueltos por el conversor analógico-digital.

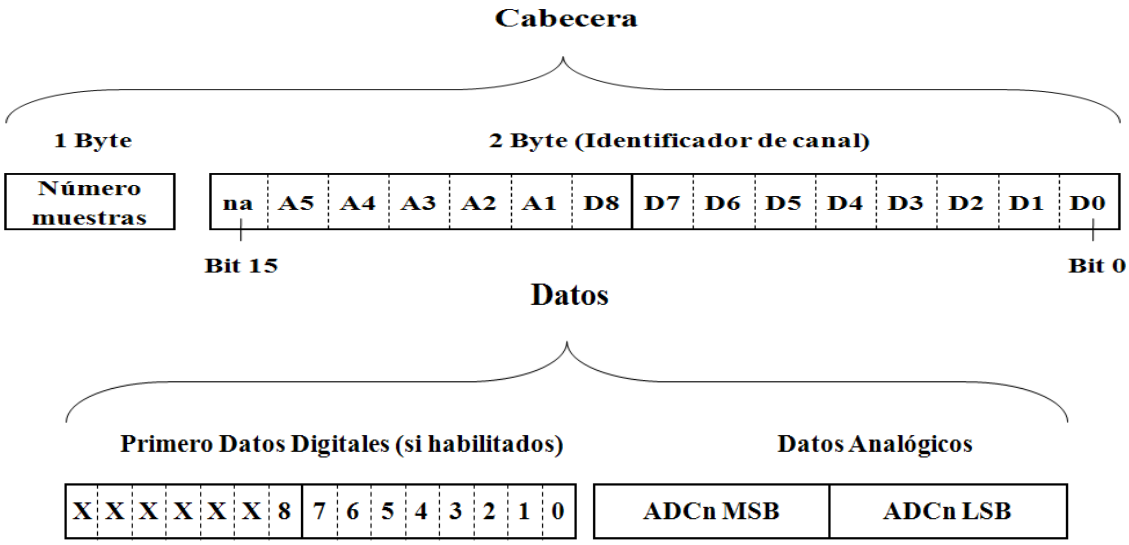


Fig. 2.14 Formato estructura I/O

A la hora de realizar la comunicación entre dos módulos es necesario que ambos tengan el mismo identificador de '*Personal Area Network*' (PAN) y que utilicen el mismo canal IEEE 802.15.4. Además ambos deben utilizar el mismo tipo de direccionamiento eligiendo entre el uso de direcciones de 16 o 64 bits.

2.4 Consumo de los módulos XBee

Un aspecto muy importante a tener en cuenta es el consumo energético de los módulos. La tabla 2.5 muestra los consumos energéticos máximos suministrados por el fabricante en función del modelo y del estado en que se encuentra.
Los módulos XBee presentan un ajuste de su potencia de transmisión aunque bastante limitado. Mediante el comando de nombre 'NI' se puede ajustar dicha potencia en cinco niveles diferentes. Para comparar los consumos de corriente suministrados por el fabricante con los reales y para ver cuáles son los consumos para los diferentes niveles de potencia, se llevó a cabo una serie de medidas de consumo de corriente.
Para ello se conectó una resistencia de un omnio entre la fuente de alimentación que suministraba energía al módulo XBee y el pin de alimentación del mismo. De esta forma, midiendo el voltaje de dicha resistencia se obtenía un valor aproximado de la corriente consumida.

	Tx	**Rx**	**Power Down**	**Sleep**
XBee Series 1	45 mA	50 mA	< 10 µA	< 50 µA
XBee Pro Series 1	250 mA	55 mA	< 10 µA	< 50 µA
XBee Pro Series 1 (International Variant)	150 mA	55 mA	< 10 µA	< 50 µA
XBee Pro Series 1 (RPSMA module)	340 mA	55 mA	< 10 µA	< 50 µA
XBee Series 2	40 mA	40 mA	< 1 µA	< 50 µA
XBee Pro Series 2 S2	295mA	45mA	< 3.5 µA	< 50 µA
XBee Pro Series 2 S2 (International Variant)	170 mA	45mA	< 3.5 µA	< 50 µA
XBee Pro Series 2 S2B	205mA	47mA	< 3.5 µA	< 50 µA
XBee Pro Series 2 S2B (International Variant)	117 mA	47mA	< 3.5 µA	< 50 µA

Tabla 2.5 Formato Trama modo API

La evolución temporal del consumo de corriente para cada uno de los niveles de potencia de transmisión de un módulo XBee Pro se llevo a cabo con un osciloscopio digital modeloDPO3014 del fabricante Tektronix.
Los datos resultantes se muestran en la Fig. 2.15.
Puede comprobarse como efectivamente los picos máximos de consumo en transmisión rondan los 250 mA como prometía el fabricante, aunque el consumo medio ronda los 200 mA. Puede verse también como el consumo en recepción ronda los 50 – 55 mA y como el consumo de corriente decrece conforme se disminuye la potencia de transmisión hasta un valor de unos 100 mA para el menor valor de la misma.

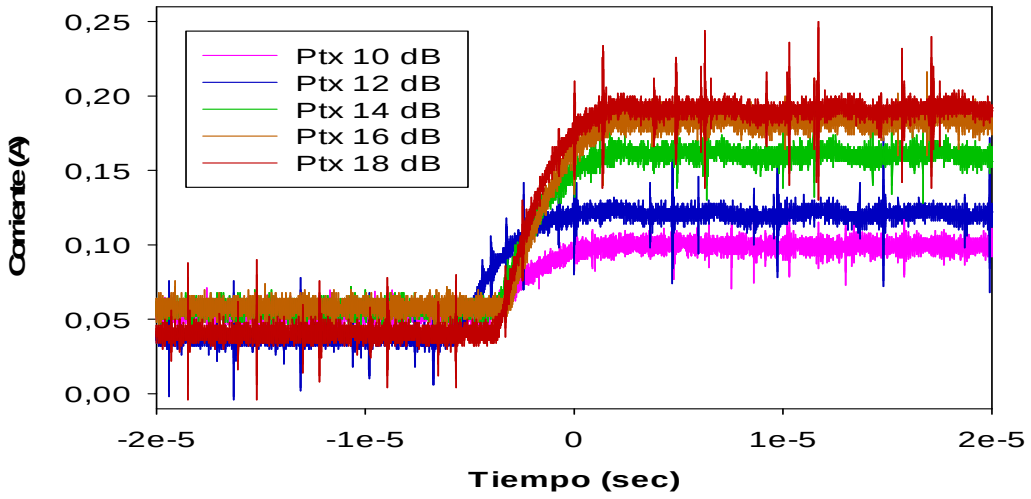


Fig. 2.15 Consumos corriente XBee Pro

Una característica muy interesante que presenta el modulo a la hora minimizar el consumo eléctrico es la posibilidad de trabajar en modo 'Sleep' en el cual la corriente consumida se reduce a unas decenas de microamperios. En la tabla 2.6 pueden verse los diferentes modos de 'Sleep' disponible, con sus respectivos consumos de corriente y tiempos de despertado.

Hay que tener en cuenta que mientras los Xbee en modos de 'Sleep' 1 y 2 se despiertan en función del nivel de voltaje del pin correspondiente, en modo 5 se despiertan cuando hay una transición de '1' a '0' y no mientras se detecte un voltaje de '0'. Cuando los XBee trabajan en modos de 'Sleep' cíclico, tanto 4 como 5, es necesario además definir cuanto tiempo de inactividad debe transcurrir desde que despiertan hasta que vuelven a dormirse.

Finalmente es necesario determinar las acciones que el módulo lleva a cabo cuando se despierta. Por un lado el dispositivo puede programarse para que nada mas despertarse realice un muestreo de todas las entradas activas y las transmita en un mismo paquete. Por otro lado también puede programarse para que nada mas despertarse le pregunte al nodo coordinador de la red si hay paquetes esperando para él. El problema de esta última opción es que el número de paquetes de petición que el coordinador puede almacenar es muy limitado y existen muchas posibilidades de que la memoria se sature y dichos paquetes se descarten.

Modo	Descripción	Tiempo de despertado	Consumo
1	Modo 'Sleep' cuando el pin de 'Sleep' esta a '1' y despertado cuando esta a '0'	13.2 msecs	< 10µA
2	Modo 'Sleep' cuando el pin de 'Sleep' esta a '1' y despertado cuando esta a '0'	2 msecs	< 50µA
4	Despertado cíclico	2 msecs	< 50µA
5	Despertado cíclico y cuando se detecta una transición de '1' a '0' en el pin de 'Sleep'	2 msecs	< 50µA

Tabla 2.6. Xbee en modo 'Sleep'

2.5 Cobertura radioeléctrica

En todo sistema de comunicación radioeléctrica es muy importante analizarla cobertura radioeléctrica en el entorno de trabajo. Como se ha comentado anteriormente, los entornos en interiores de edificios son especialmente problemáticos debido a que los obstáculos entre transmisor y receptor propician la aparición de fenómenos de atenuación, difracción, interferencia debida a componentes multitrayecto, etc. En este trabajo se realizó una serie de análisis y medidas de cobertura en el laboratorio Nº 5 del edificio de I+D de la Universidad Pública de Navarra. La estancia (ver Fig. 2.16) presenta el típico escenario de interiores, con gran cantidad de mobiliario, equipos electrónicos, muebles metálicos, etc.

Fig. 2.16 Laboratorio Nº 5 del edificio de I+D de la UPNA

Como puede verse en la imagen de la Fig. 2.17 que muestra el esquema del escenario de medición, dos de sus paredes son de hormigón y dan al exterior mientras que el resto son de aglomerado y dan al interior del edificio. Además del mobiliario existen dos columnas de hormigón, representadas por los círculos rosas.

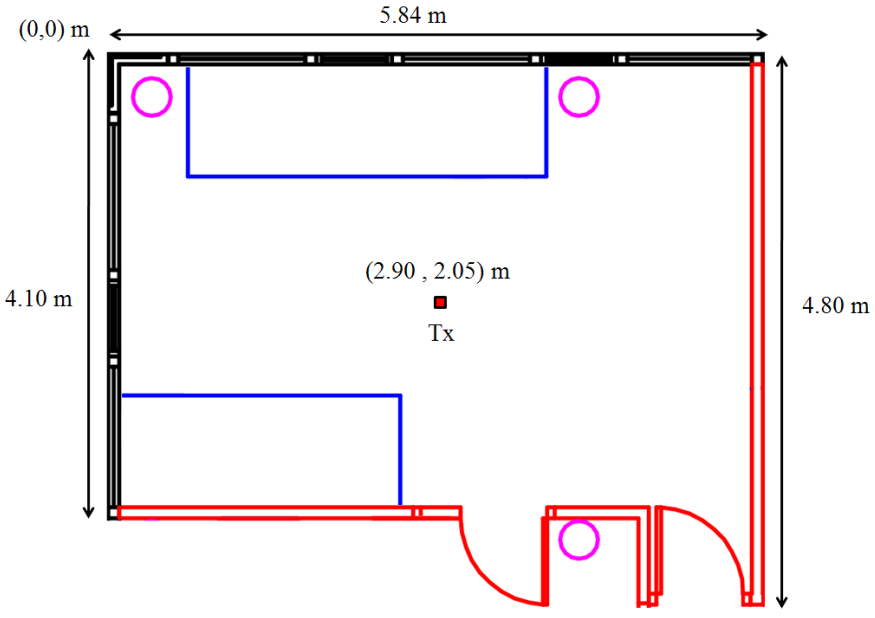


Fig. 2.17 Laboratorio Nº 5 del edificio de I+D de la UPNA

El origen de coordenadas se tomó en el extremo superior izquierdo del escenario y un módulo XBee Pro transmitiendo a máxima potencia se colocó en el centro del mismo, a una altura de 105 cm.

El primer paso del trabajo consistió en un análisis de cobertura utilizando modelos de propagación empíricos, más concretamente se utilizaron: *Linear Atenuation Path*, ITU R P. 1238, COST 231, Keenan – Motley y Multi – Wall.
Los valores obtenidos se muestran en lo Fig. 2.18.

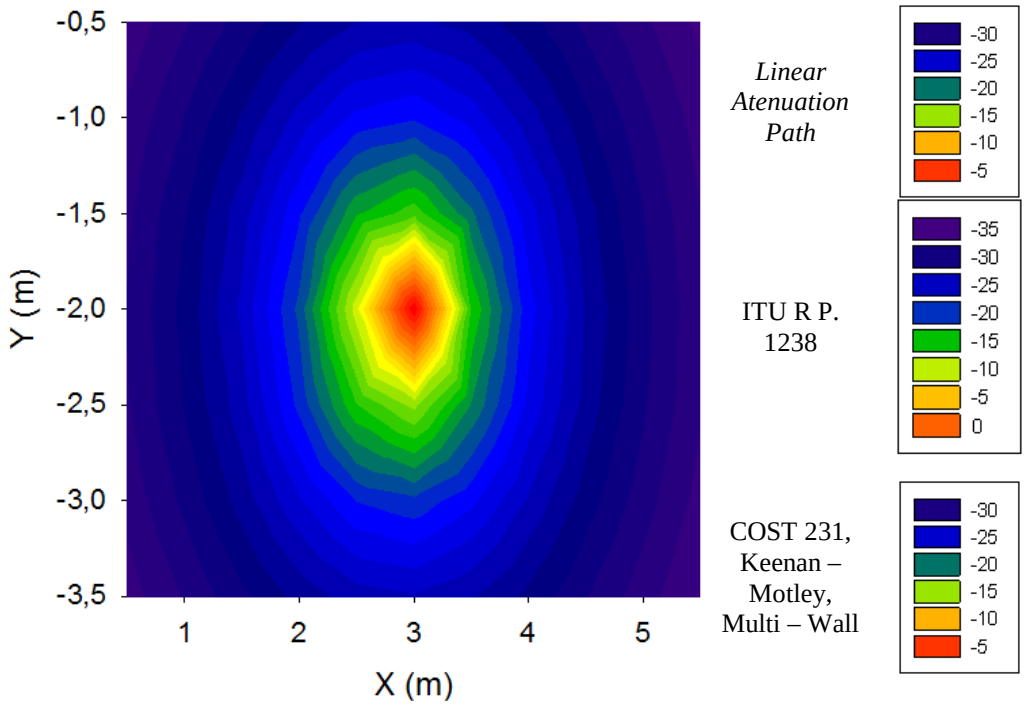


Fig. 2.18 Análisis cobertura mediante modelos empíricos

Puede apreciarse como los valores calculados presentan la misma distribución de potencia salvo un factor de escala y algunos de ellos coinciden exactamente.
A la vista de estos datos, es fácil que el uso de estos modelos es sencillo y rápido pero los datos obtenidos son muy burdos, haciéndolos útiles únicamente como análisis preliminar o en casos en los que no se requiere gran precisión.
El segundo paso del trabajo consistió en análisis de la cobertura utilizando un modelo de propagación determinista, más concretamente basado en lanzado de rayos 3D. Para ello se utilizó una aplicación basada en Matlab y desarrollada en la Universidad Pública de Navarra.
Dicha herramienta divide el espacio de análisis en una serie de tetraedros y realiza un lanzado de rayos basado en incrementos del ángulo solido asociado a los mismos.
Tiene en cuenta fenómenos electromagnéticos como reflexión, refracción y difracción de primer orden y es capaz de suministrar análisis de planos de cobertura, perfiles de retardo-potencia y planos de dispersión entre otros.
El primer paso para su uso consiste en definir el escenario. Como se ha comentado anteriormente, el volumen del mismo se divide en una serie de tetraedros para cada uno de los cuales se realizara el correspondiente análisis. Cuanto mayor sea el detalle del escenario y menor el volumen de los cuboides utilizado, mayor precisión se conseguirá en el resultado final aunque con un coste de computación mayor. Por lo tanto existe un compromiso claro entre resolución y tiempo de cómputo.
En la Fig. 2.19 puede apreciarse la modelización del escenario previa al análisis mediante trazado de rayos 3D asociado.

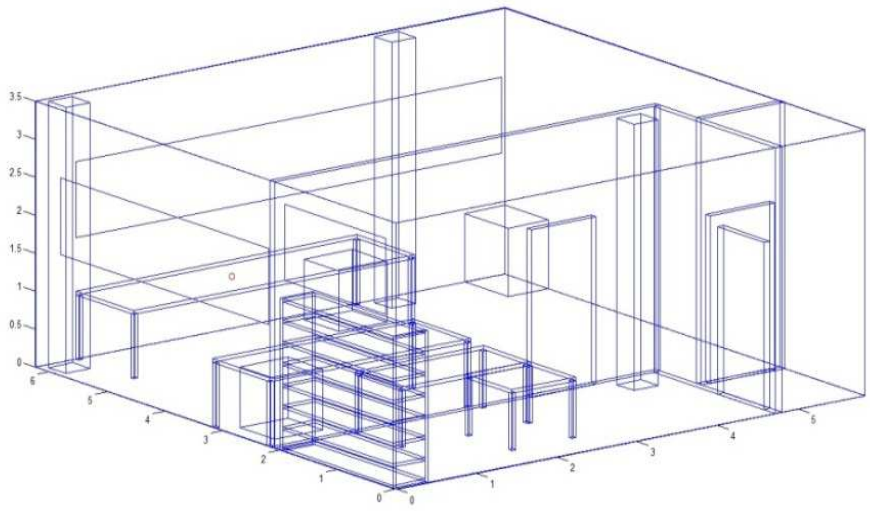


Fig. 2.19 Modelo del escenario utilizado para el trazado de rayos 3D

La Fig. 2.20 muestra el plano de cobertura calculado mediante la herramienta de trazado de rayos para una altura simular a la del dispositivo transmisor de 105 cm. Puede apreciarse como la distribución que presentan los valores calculados es significativamente más compleja.

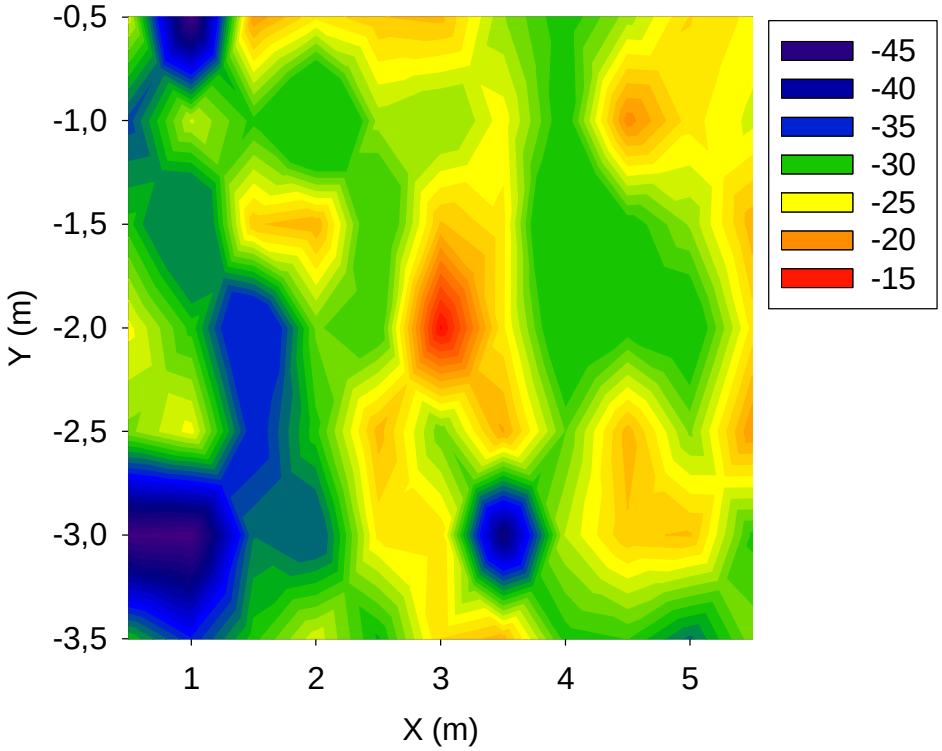


Fig. 2.20 Análisis cobertura utilizando trazado de rayos 3D

Finalmente, para determinar el grado en el que dichos modelos se ajustan a los a la realidad se procedió a la realización de una serie de medidas reales.

Para ello se dividió el espacio en una cuadricula con subdivisiones de tamaño 50 cm y se procedió a la medida de la potencia recibida en cada uno de los puntos mediante un analizador de espectros modelo N9912A de Agilent.
Dichos resultados se muestran la Fig. 2.21.
Ante todo hay que mencionar que el módulo XBee Pro utilizado tenía una configuración de antena de tipo '*chip*' que presenta un diagrama de radiación muy irregular. Si se compara dicho diagrama (ver Fig. 2.12) con las medidas obtenidas puede apreciarse claramente como este hecho ha influido en los resultados de las mismas.
A la vista de estos resultados puede verse como efectivamente los datos obtenidos mediante modelos empíricos no se ajustan muy bien con los datos reales mientras que el análisis mediante lanzado de rayos proporciona.

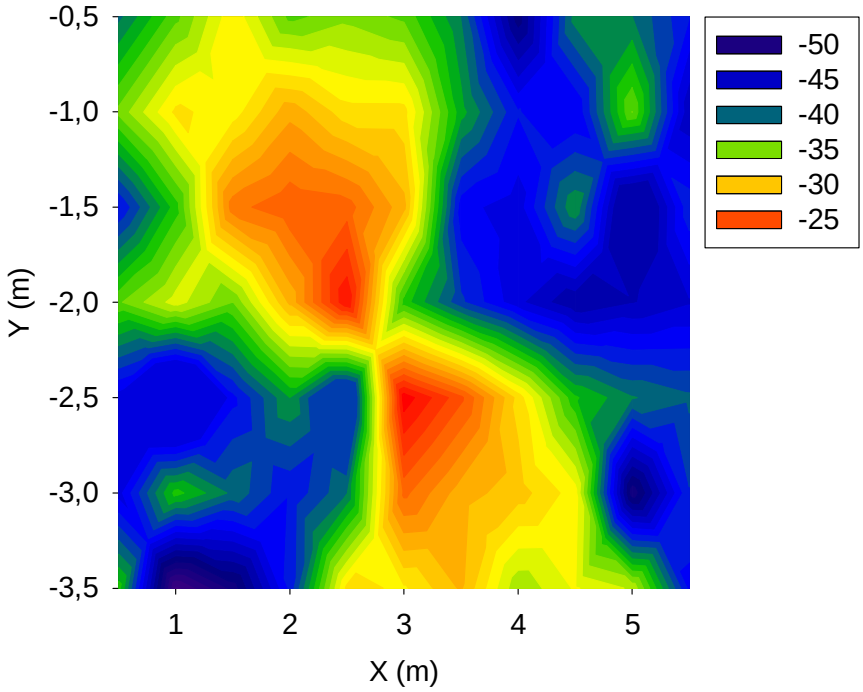


Fig. 2.21 Medidas de cobertura reales obtenidas mediante el analizador de espectros

2.6 Calidad del radioenlace sin señales interferentes

En un radioenlace, además de la cobertura radioeléctrica otro aspecto muy importante es la calidad del mismo. Es bien sabido que este parámetro es altamente afectado por la aparición de componentes multitrayecto y además que este valor decrece exponencialmente en función de la potencia recibida. En este apartado se comenta el proceso que se siguió para analizar este aspecto.
Primero se realizó para tal propósito un par de aplicaciones programadas en Java capaces conjuntamente de determinar la perdida de paquetes en un radioenlace.
La aplicación transmisora envía un número parametrizable de paquetes de datos a máxima velocidad al receptor que contiene un número de secuencia que se incrementa en una unidad con cada paquete enviado y el instante de tiempo en el que se ha enviado. Para que la aplicación receptora sepa cuándo se ha terminado de emitir, la aplicación transmisora envía un paquete de aviso especial que además lleva como información la cantidad de paquetes que se ha transmitido.
La aplicación receptora registra los paquetes recibidos y guarda los datos en un fichero de texto y comparándolos con lo que se han transmitido es capaz de calcular el '*Packet Error Rate*' (PER). Además, es capaz de obtener una estimación del '*Received Signal Strength*' (RSS) de cada paquete

recibido, valor que también se almacena en el fichero y cuyo valor medio se muestra también. Finalmente, el par de aplicaciones se puede configurar para que la transmisión de paquetes se realice con el mecanismo de envío de paquetes de confirmación activado o desactivado.

Fig. 2.22 Imagen del antiguo laboratorio de Radiocomunicación

Para ver el comportamiento de la calidad del canal radio utilizando los módulos XBee en interiores se procedió a la realización de una serie de medidas en el laboratorio Nº 5 del edificio de I+D anteriormente mencionado así como en el antiguo laboratorio de Radiocomunicación, mostrado en la Fig. 2.22.
Puede verse claramente como esta estancia es también un típico de interiores, con multitud de mobiliario, equipos electrónicos, etc.
Hay que tener en cuenta que para obtener mejores valores estadísticos es necesario que las medidas se realicen lo mas estáticamente posible, evitando cambios en el mobiliario o la presencia de personas. Además es necesario comentar que la calidad estadística de los valores obtenidos es directamente proporcional a la cantidad de paquetes transmitidos aunque con un coste de mayor tiempo de medición. En la Fig. 2.23 puede verse la representación de diversos valores de PER en función del valor RSS medio asociado para medidas de 100000 de paquetes realizadas en el laboratorio de radiocomunicación. Puede apreciarse claramente la forma exponencial que presenta la distribución comenzando a crecer de manera importante entorno a los -80 dBm de potencia recibida.

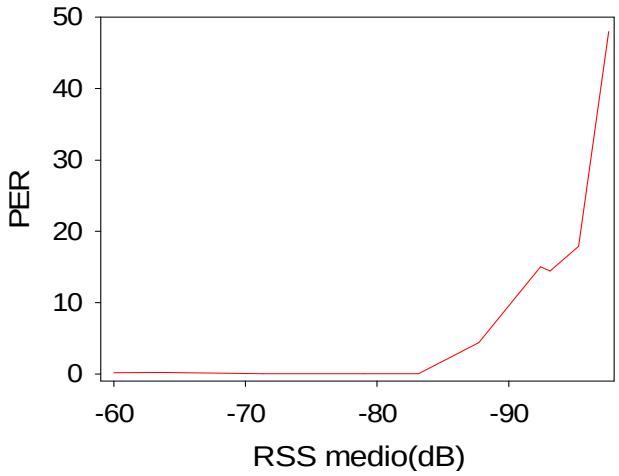


Fig. 2.23 PER vs RSS laboratorio de Radiocomunicación

La Fig. 2.24 muestra diversos valores de PER frente al valor RSS medio asociado para medidas de 50000 paquetes y en las que se analiza la calidad de enlace con y sin mecanismo de confirmación. Pude verse claramente como utilizando paquetes de conformación la calidad mejora ostensiblemente.

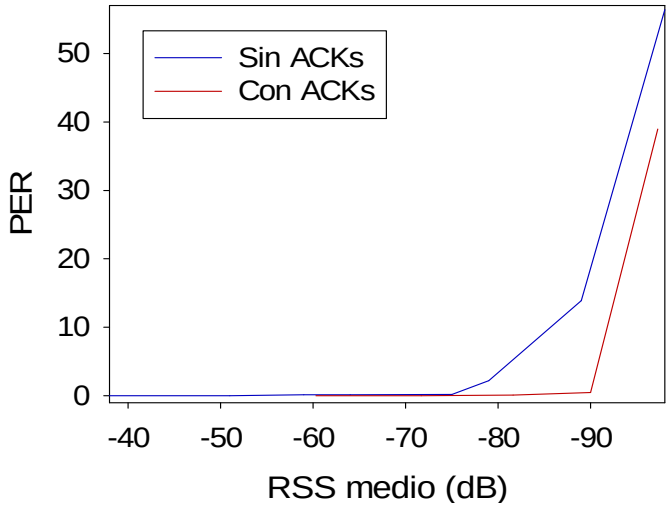


Fig. 2.24 Comparativa uso ACKs

2.7 Calidad del radioenlace con señal interferente: horno microondas

El nivel de potencia radioeléctrica recibida no es el único factor que determina la calidad del enlace. Hay que tener en cuenta que parte de esa potencia puede ser debida a fuentes interferentes existentes en la misma banda de frecuencia. En este trabajo se procedió al estudio de cómo una fuente de interferencia radioeléctrica como un horno microondas afecta a la calidad del radioenlace basado en IEEE 802.15.4.

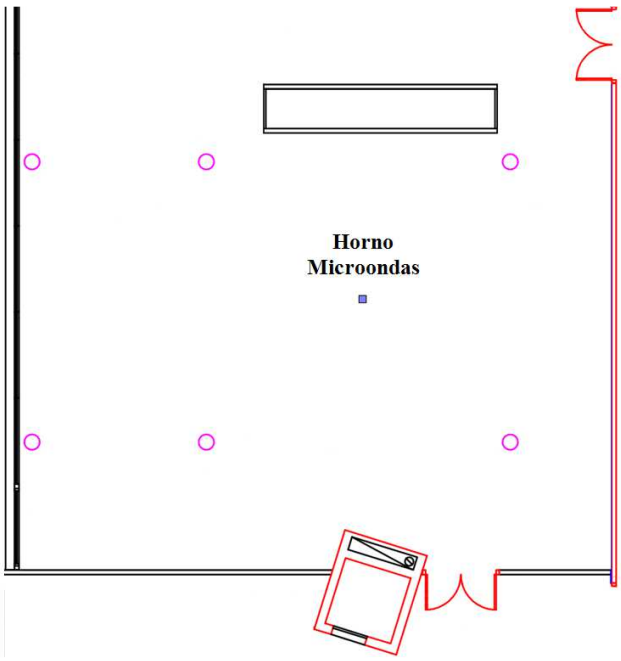


Fig. 2.25 Escenario de medición

El escenario utilizado, mostrado en la Fig. 2.25, fue la antigua planta baja del edificio de I+D de la Universidad Pública de Navarra.
La pared dibujada en rojo de la derecha es de contrachapado mientras que el resto de paredes son de cristal. La caja del ascensor es de hormigón, así como la estructura de forma rectangular cercana a la puerta de arriba. Finalmente, los círculos rosas representan columnas de hormigón. El microondas se colocó en el centro del escenario para minimizar el efecto del entorno en los resultados finales. El microondas se coloco sobre una mesa no metálica para minimizar su impacto electromagnético en las medidas finales, a una altura de 70 cm.
El modelo de microondas utilizado en este trabajo fue el modelo BMG20-4 del fabricante BlueSky. El primer paso consistió en realizar un análisis radioeléctrico de las fugas electromagnéticas del mismo.
Para ello primero se procedió a analizar el espectro de emisión de fuga. El microondas tiene tres modos de funcionamiento: en baja, media y alta potencia Teniendo en cuenta que se busca el peor caso posible, la potencia seleccionada fue la alta, cuyo espectro se muestra en la Fig. 2.26. Dicho espectro se obtuvo con un analizador de espectros modelo N9912A de Agilent y una antena omnidireccional modelo OAN-1070 del fabricante LevelOne. Para ello se colocó la antena a una distancia de 20 cm frente a la puerta del horno y a la misma altura que este.

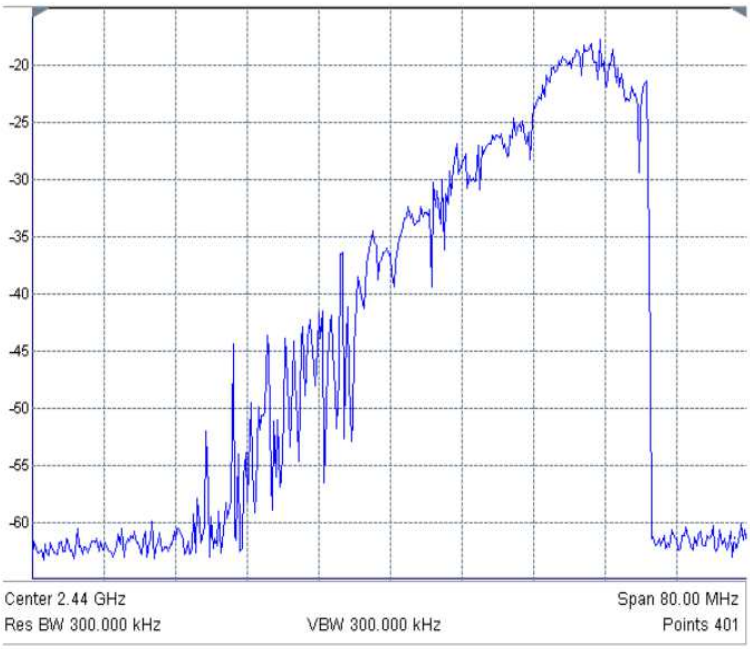


Fig. 2.26 Espectro horno microondas trabajando en alta potencia

Analizando el espectro de la señal interferente, se puede apreciar como la potencia de las componentes de la misma aumenta conforme se acerca a los 2.45 – 2.46 GHz, con unos niveles de potencia interferente de unos -18 dBm.
Colocando un marcador se observó que los mayores niveles de potencia giraban en torno a los 2.46 GHz con lo que para el resto de medidas de potencia de fuga recibida se utilizó para su medición una frecuencia central de 2.46 GHz y un '*Span*' de 5 MHz. con un tiempo de medición de treinta segundos para que la medición se estabilizase.
A continuación se procedió a la medición de la potencia de fuga recibida en diferentes puntos de tres planos rectangulares de tamaño 5 x 5 m divididos en cuadriculas con subdivisiones de tamaño 50 cm y con alturas de 20, 70 y 120 cm respectivamente. Los datos obtenidos se muestran en la Fig. 2.27 de izquierda a derecha. Como era de esperar, observando dichas imágenes puede verse como el

peor caso de fuga se encuentra en la zona inmediatamente en frente de la puerta del microondas y a la altura de este.
También hay que tener que la señal interferente debida a un horno microondas es de naturaleza dinámica y la distribución espectral de las componentes frecuenciales que sus componentes y su amplitud varían con el tiempo. Para ello se procedió a la realización de un espectrograma en modo '*Max&Hold*' y de duración 5 minutos mostrado en la Fig. 2.28.

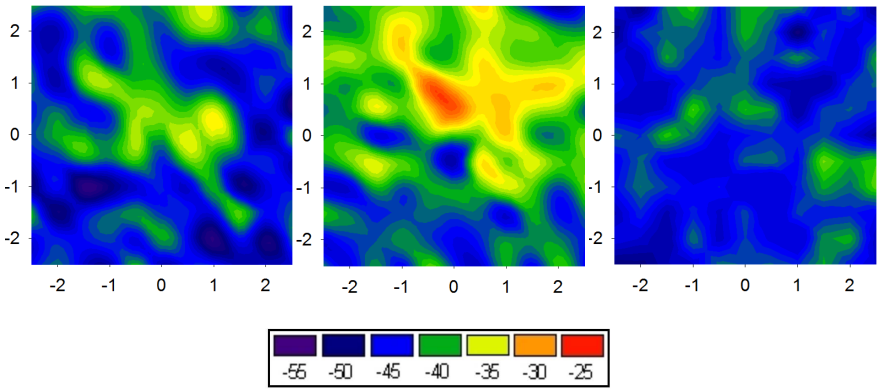


Fig. 2.27 Planos de potencia de fuga recibida

Si se observa y analiza dicha figura puede verse como efectivamente la naturaleza de la interferencia es dinámica y como las componentes de más baja frecuencia son las primeras en aparecer mientras que las componentes de más alta frecuencia y de potencias mayores no aparecen hasta los 50 segundos del encendido aproximadamente.
Como se ha comentado anteriormente la zona de mayor impacto electromagnético es aquella situada en frente y a la misma altura que el horno microondas. Por lo tanto el dispositivo IEEE 802.15.4 receptor se colocó precisamente en ese lugar, a 50 cm de la puerta del mismo.
Para calcular el mejor lugar para colocar el dispositivo transmisor se procedió a realizar una serie de medidas de potencia para diferentes distancias en línea recta en frente del microondas y a la misma altura, datos que se muestran en la Fig. 2.29.

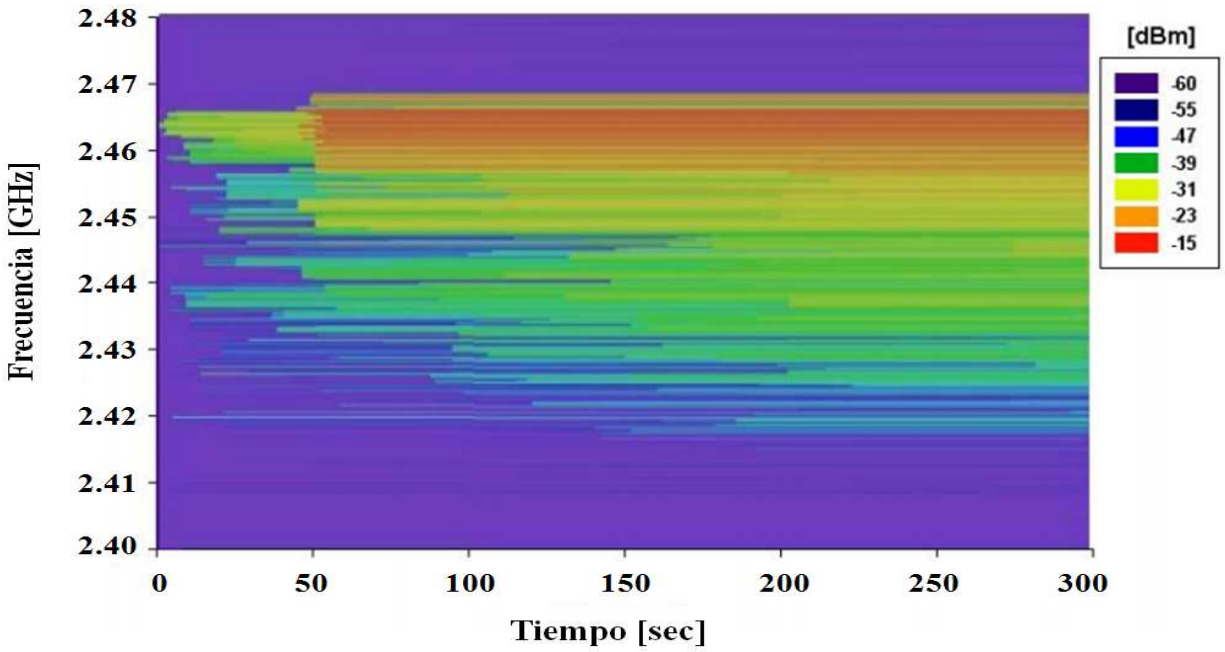


Fig. 2.28 Espectrograma "Max Hold" del horno microondas

Para la realización de las medidas de calidad se tuvo en cuenta que se iban a realizar para los cinco niveles de potencia de transmisión y siempre intentando en la medida de lo posible que la potencia recibida fuese lo más cercana al umbral de sensibilidad para analizar los peores casos posibles.
Teniendo en cuenta los niveles de potencia interferente recibida debido al microondas para las diferentes distancias en línea recta enfrente del mismo y a lo anteriormente mencionado se colocó el dispositivo transmisor a una distancia de 12 metros, también a una altura de 70 cm.
Entonces, para cada uno de los canales IEEE 802.15.4 y para las cinco diferentes potencias de transmisión, se fueron calculando el PER resultante del envío de 50000 paquetes del transmisor al receptor con el microondas funcionando a máxima potencia y con el mecanismo de envío de paquetes de confirmación desactivado.
Debido a que los módulos XBee Pro utilizados utilizan antenas de tipo de 'chip 'con un diagrama de radiación muy irregular, se trató de mantener invariables las orientaciones de las antenas tanto transmisora como receptora a lo largo de todas las mediciones.

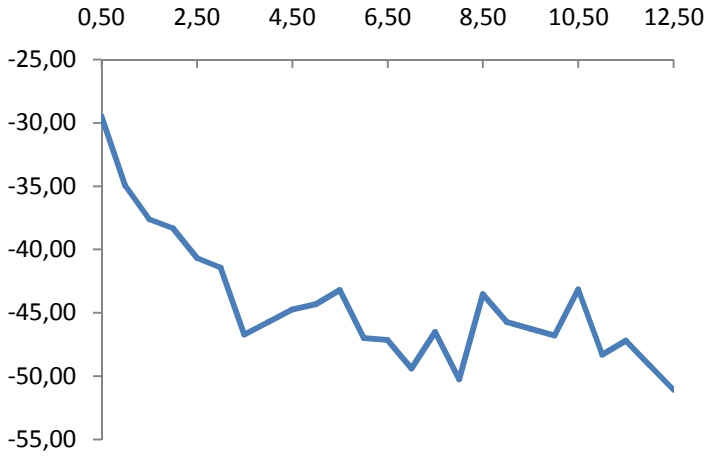


Fig 2.29 Medidas de potencia recibida a diferentes distancias enfrente del horno

Las aplicaciones Java se comunican con el hardware donde mediante el protocolo RS232 a una velocidad máxima de 57600 baudios, aspecto que limita la velocidad final de transmisión de los paquetes y por lo tanto de la duración final del test.
Los datos resultantes de dichas mediciones pueden observarse en la Fig. 2.30. Puede apreciarse como eligiendo una canal lo suficientemente alejado no aparecen problemas debido a la interferencia. Conforme se va acercando a la máxima frecuencia de interferencia es posible paliar los efectos aumentando la potencia de transmisión pero puede verse que llegado a un cierto limite la degradación del PER es inevitable.

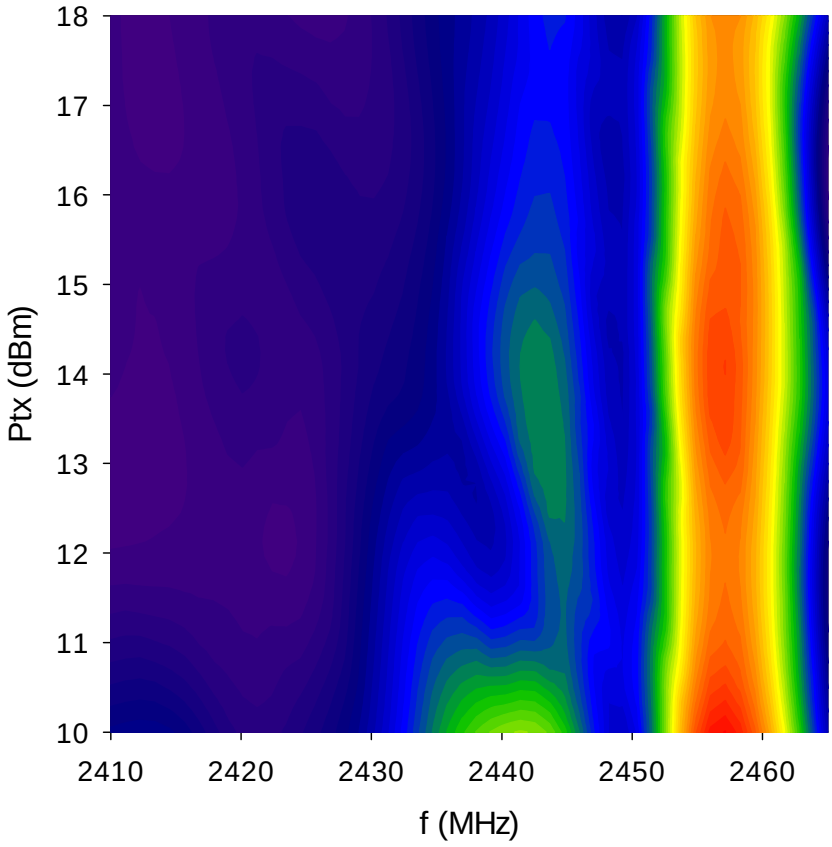


Fig. 2.30 Medidas de PER en función de la frecuencia y potencia de transmisión

2.8 Movilidad en IEEE 802.15.4

En la actualidad las implementaciones '*Vehicular Ad-Hoc Network*' (VANET) más populares stand basadas en el estándar IEEE 802.11p [12], especialmente diseñado para dispositivos de alta movilidad. Si se compara con el estándar IEEE 802.11, la duración del tiempo de símbolo para reducir los efectos del ensanchamiento Doppler así como el intervalo de guardia para reducir de la interferencia entre símbolos. Este estándar presenta dos inconvenientes. El primero es que la banda de frecuencia que utiliza requiere licencia y el segundo es el no despreciable consumo energético de los dispositivos que lo implementan. En este trabajo se procedió a el estudio la posible utilización de IEEE 802.15.4 para su uso en VANETs de baja movilidad en entornos urbanos, debido principalmente a su bajo consumo y su funcionamiento en la banda libre de 2.4 GHz.
El primer paso consistió en la realización de una simulación por ordenador.
Previamente a la realización de la simulación hay que tener en cuenta diversos aspectos de propagación radioeléctrica. El más importante de todos ellos son las pérdidas por propagación en el espacio libre (FSPL) comentado en el apartado 2.2.1.
Otro aspecto a tener en cuenta es el efecto Doppler. Consiste básicamente en el cambio relativo en la frecuencia de una onda debido a la velocidad relativa entre el emisor de la misma y el receptor. La frecuencia observada viene dada por la siguiente fórmula:

$$f = \left(\frac{v \pm v_r}{v \pm v_s}\right) f_0 \tag{2.3}$$

donde el parámetro 'f_0' es la frecuencia emitida, 'v' es la velocidad de propagación de la onda en el medio, 'v_s' es la velocidad de la fuente en relación con el medio y 'v_r' es la velocidad del receptor en relación con el medio.
Si un vehículo que transmite una onda varía su velocidad, el efecto Doppler asociado también varia, lo que resulta en una modulación aleatoria de la señal. Si el receptor no presenta movimiento alguno y el transmisor se mueve en relación a él, el desplazamiento Doppler asociado viene dado por:

$$f_d = \frac{v}{\lambda}\cos\theta \tag{2.4}$$

donde 'v' es la velocidad del transmisor, 'λ ' es la longitud de onda de la señal y 'θ' es el ángulo que forma la dirección de movimiento del transmisor en relación con el receptor.
Por lo tanto, utilizando la formula anterior el máximo desplazamiento Doppler para una velocidad máxima de 60 Km/h y una frecuencia de transmisión de 2.4 GHz es de 133 Hz. Si se compara este valor con los 2 MHz de anchura espectral de un canal IEEE 802.15.4 puede suponerse que la influencia del efecto Doppler asociado a esa velocidad es mínimo.
En entornos multitrayecto en los que varios ecos reflejados de la señal llegan al receptor con aproximadamente el mismo retardo pero con diferente ángulo de llegada y /o velocidad relativa aparece el efecto conocido como ensanchamiento Doppler. Debido a ello el espectro de la señal recibida es una versión más ruidosa y espectralmente más ancha de la señal original, resultado de una combinación de la misma con los espectros de los ecos recibidos.
Finalmente, debido al entorno en las comunicaciones inalámbricas vehiculares cambia rápidamente, las características del canal radio no son estáticas y para cuantificar esto se define el termino tiempo de coherencia. Este término representa el periodo de tiempo en el cual el comportamiento del canal puede considerarse invariante con el tiempo y se calcula como el valor inversión del desplazamiento Doppler calculado con (2).
Para una velocidad de 60 Km/h y una frecuencia de transmisión de 2.4 GHz se tiene un tiempo de coherencia de 75 ms. Este valor es mucho mayor que el tiempo de transmisión típico de un paquete IEEE 802.15.4 con lo que parece razonable que el efecto sea mínimo.
Para realizar la simulación es necesario ir más allá de utilizar simplemente el modelo de estimación de perdidas por propagación en el espacio libre. Una aproximación más realista debería considerar además que la radio propagación sufre además de al menos otra fuente de interferencia, utilizando para ello el modelo de reflexión de dos rayos (ver Fig. 2.31).

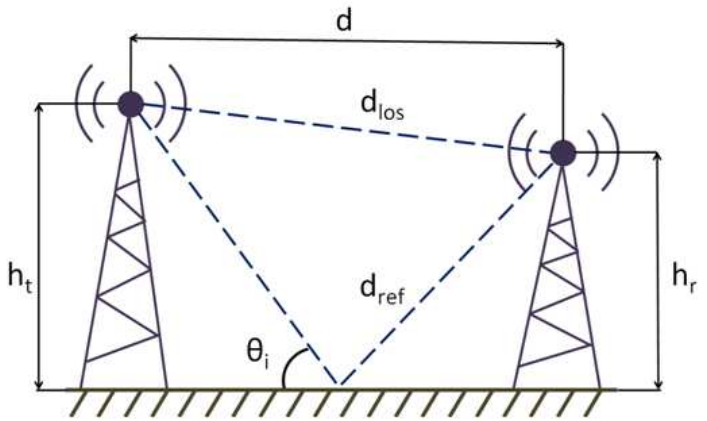


Fig. 2.31 Modelo de reflexión de dos rayos

En el destino, el rayo directo y el rayo reflejado se reciben simultáneamente, interfiriéndose mutuamente. Además, el rayo reflejado presenta una diferencia de fase con respecto al rayo directo

debido a que recorre una distancia mayor y se ve también afectado por el coeficiente de reflexión debido a la reflexión con la tierra [13][14][15].

$$L_{tr}(dB) = 20\log\left(4\pi\frac{d}{\lambda}\left|1 + \Gamma_{\perp}e^{i\varphi}\right|^{-1}\right) \quad (2.5)$$

$$\varphi = 2\pi\frac{d_{ref} - d_{los}}{\lambda} \quad (2.6)$$

$$\Gamma_{\perp} = \frac{\sin\theta_i - \sqrt{\epsilon_r - cos^2\theta_i}}{\sin\theta_i + \sqrt{\epsilon_r - cos^2\theta_i}} \quad (2.7)$$

$$\epsilon_r = \varepsilon_r - j60\sigma\lambda \quad (2.8)$$

$$d_{los} = \sqrt{d^2 + (h_t - h_r)^2} \quad (2.9)$$

$$d_{ref} = \sqrt{d^2 + (h_t + h_r)^2} \quad (2.10)$$

$$\sin\theta_i = \frac{h_t + h_r}{d_{ref}} \quad (2.11)$$

$$cos\theta_i = \frac{d}{d_{ref}} \quad (2.12)$$

Si el transmisor se encuentra en movimiento, debido a que el rayo directo y el reflejado llegan al receptor con diferentes ángulos de entrada aparece ensanchamiento Doppler. Por lo tanto y para tener en cuenta este efecto, además de la atenuación y el cambio de fase debido a la reflexión con el suelo hay que añadir al rayo reflejado un desplazamiento en frecuencia.
El material que se utilizó en la simulación para la superficie de la carretera fue asfalto. Los parámetros electromagnéticos del mismo utilizados en la simulación fueron una permitividad relativa de $\varepsilon_r = 5$, una conductividad de cero y una permeabilidad magnética de $\mu_r = 1$. Finalmente, la frecuencia de trabajo se fijó en 2.4 GHz.
La simulación se realizó utilizando la herramienta de automatización de diseño electrónico para RF y microondas '*Advanced Design System*' (ADS) de 'Agilent'. Esta herramienta posee diferentes módulos para la simulación de diferentes tecnologías RF pero en la actualidad no soporta IEEE 802.15.4. Por lo tanto para la realización de este trabajo fue necesario el desarrollo específico de dichos módulos.
Utilizando el lenguaje de programación '*ADS Ptolemy Preprocessor*' un usuario de ADS puede escribir sus propios módulos llamados '*Stars*'. De esta manera el usuario escribe el código utilizando dicho lenguaje y deja al preprocesador que genere el código estándar de inicialización para los diferentes componentes del mismo: '*Portholes*', estados, código C++ estándar asociado, etc.

Los segmentos de código C++ son muy importantes en la definición de cualquier '*Star*' y pueden aparecer en muchas directivas del preprocesador. Es importante comentar que para utilizar dicha herramienta es necesario tener instalado en el sistema un compilador C++ apropiado.
El diagrama de bloques utilizado en la simulación puede verse en la Fig.2.32.

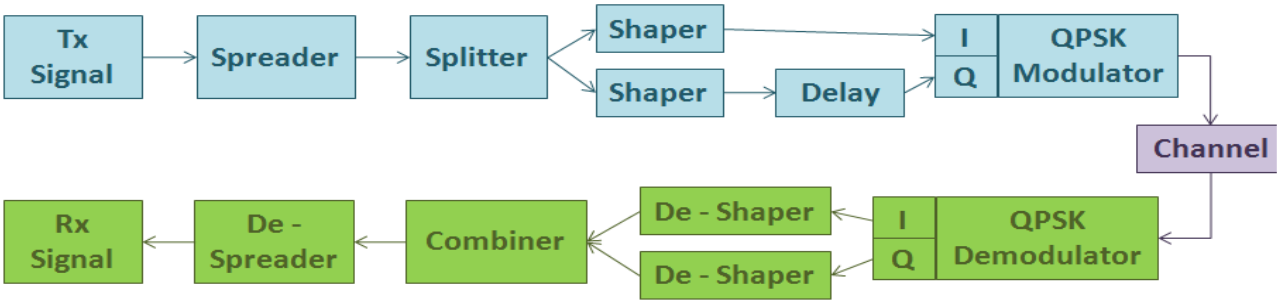


Fig. 2.32 Diagrama de bloques de la simulación en ADS

La señal a transmitir se genera mediante un bloque fuente cuya salida consiste en una secuencia aleatoria binaria con una probabilidad de que el bit sea cero parametrizable representada como un array de números enteros de valor '0' y '1'. Este flujo de datos se divide en símbolos de 4-bits de longitud y se mapean en secuencias pseudo-aleatorias de 32 chips siguiendo la tabla 24 ilustrada dentro del estándar IEEE 802.15.4. El flujo de chips así generado se divide en dos dependiendo de si el índice del bit es par o impar y a cada bit se le realiza un conformado de forma según se describe en la ecuación (2.13):

$$p(t) = \begin{cases} \sin\left(\pi \dfrac{t}{2T_c}\right) & 0 \le t \le 2T_c \\ 0 & resto \end{cases} \tag{2.12}$$

Los chips ya conformados y de índice par se envían se envían a la rama de fase del modulador QPSK y los de índice impar, se retrasan un tiempo T_c con respecto a los chips pares y se mandan a la rama en cuadratura del modulador, siendo T_c el tiempo del periodo del chip. El chip menos significativo, c0, se transmite primero y el más significativo, c31, el ultimo.
En la Fig. 2.33 se muestra un ejemplo de una secuencia de chips ya conformados:

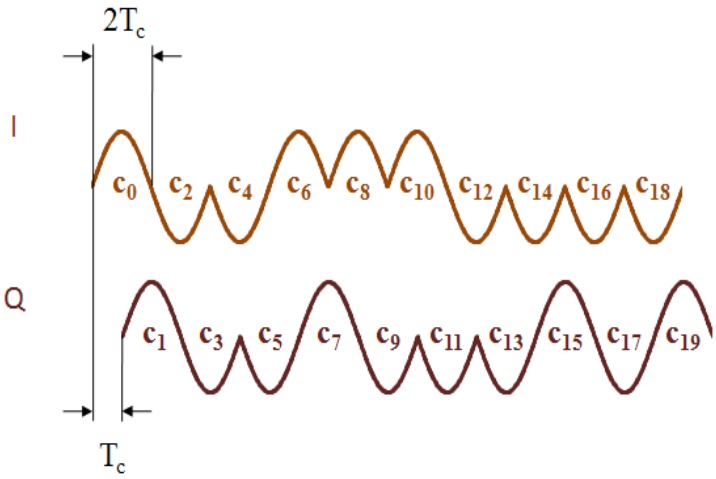


Fig. 2.33 Secuencia de chips conformados

La Fig. 2.34 muestra una representación del espectro de potencia resultante de la simulación de la señal IEEE. Como puede apreciarse, la forma que presenta el mismo contiene un lóbulo principal más ancho y una serie periódica de lóbulos laterales más estrechos y de amplitud decreciente, típica de señales inalámbricas basadas en IEEE 802.15.4.

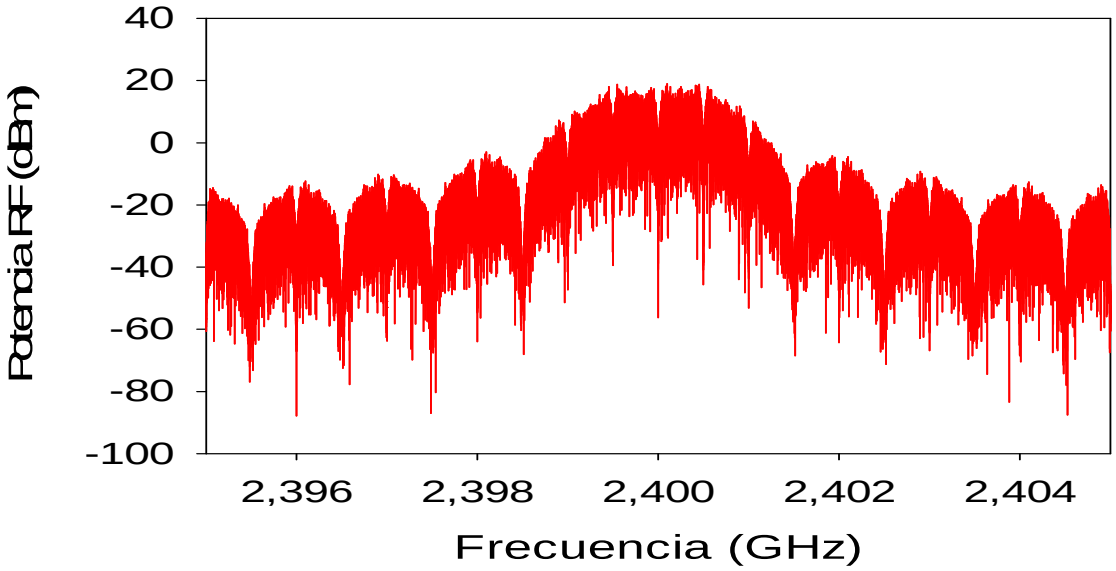


Fig. 2.34 Espectro simulado de la señal transmitida

Para determinar la importancia que la altura de las antenas tiene en la comunicación se realizó dos simulaciones diferentes: una con una altura de antenas de un metro y la otra con una altura de 2 metros.
Las figuras 2.35.a y 2.35.b muestran respectivamente los valores de RSS y de la tasa de PER en función de la distancia a la fuente para la simulación realizada para antenas de un metro de altura
Los valores obtenidos usando únicamente el modelo de reflexión de dos rayos se muestran en color azul mientras que los valores obtenidos añadiendo el efecto Doppler se muestran en color rosa. Puede apreciarse como para ambos métodos de simulación la representación del RSS es muy similar.

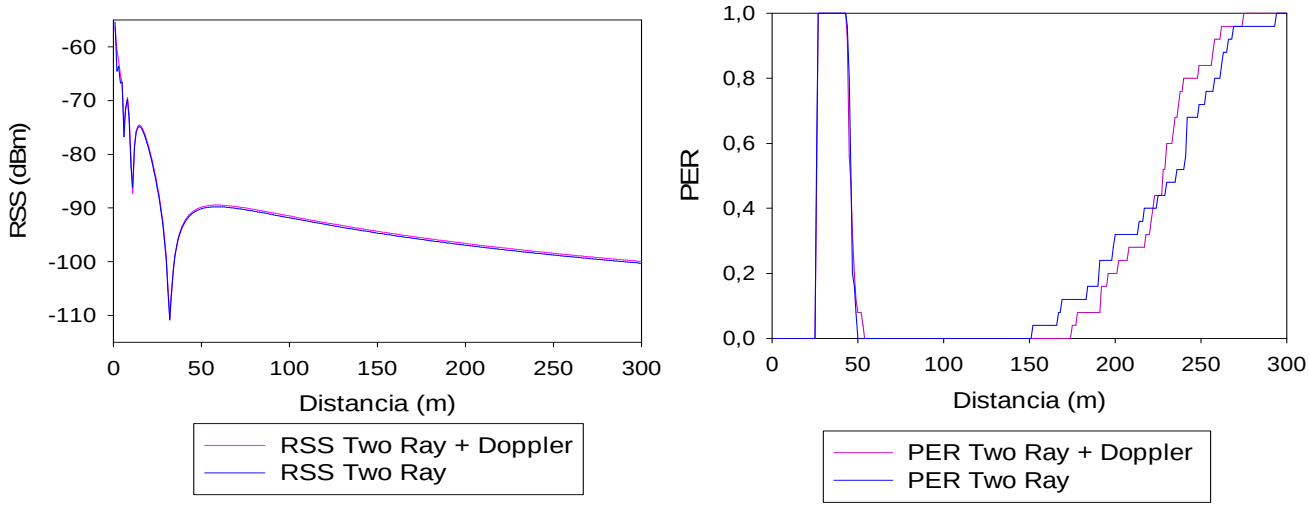


Fig. 2.35Valores de (a) RSS y (b) PER en función de la distancia para altura de antenas de 1 metro

En las figuras 2.36.a y 2.36.b se muestra la misma información que en las graficas anteriores con la única salvedad de que la longitud de antenas utilizada en la simulación es de dos metros en lugar de uno. Puede apreciarse que, como en el caso anterior y para ambos métodos de simulación, la representación de los valores de RSS son prácticamente iguales.

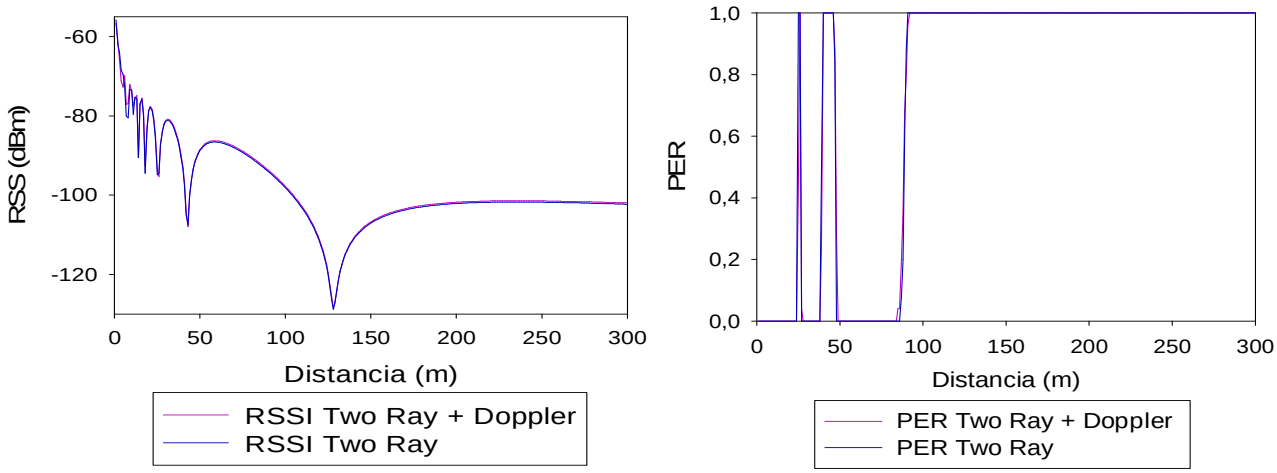


Fig. 2.36 Valores de (a) RSS y (b) PER en función de la distancia para altura de antenas de 2 metros

Comparando las figuras 2.35 y 2.36 pude verse sin embargo que existe una significativa diferencia entre las mismas y como la elección de la altura de las antenas parece un aspecto importante en los resultados finales.

Para realizar la validación de los datos obtenidos en las simulaciones fue necesaria la realización de una serie de medidas de campo. Por un lado, como se muestra en la Fig. 2.37.a, se colocó un módulo XBee Pro transmitiendo en el retrovisor de un automóvil. Por otro lado se colocó un módulo XBee Pro recibiendo en un punto fijo estático representado por un punto rojo en el escenario de medida mostrado en la Fig. 2.37 (b).

Fig. 2.37 (a) Detalle XBee Pro transmisor

Fig. 2.37 (b) Escenario

Dicha figura es una imagen vía satélite del escenario de medición situado en las cercanías del campus de la Universidad Pública de Navarra. La trayectoria de la carretera que el vehículo utilizó se muestra en color azul y el sentido seguido fue de norte a sur.
La altura de antenas utilizada fue de aproximadamente 1 metro tanto en transmisión como en recepción para poder comparar los datos de una de las simulaciones realizadas. Para el registro de la llegada de paquetes se utilizó el mismo par de aplicaciones que las usadas en el apartado 2.6.
Las figuras 2.38.b y 2.38.b muestran respectivamente los valores medidos de RSS y el número de paquetes de error frente a la distancia a la fuente para una velocidad de 30 Km/h. Si se comparan estos valores con los simulados puede verse como para casi todas las distancias los valores son muy similares. Si se compara la representación de la tasa de PER simulada con la representación del numero de paquetes de error registrada puede verse como se bastante aproximados.

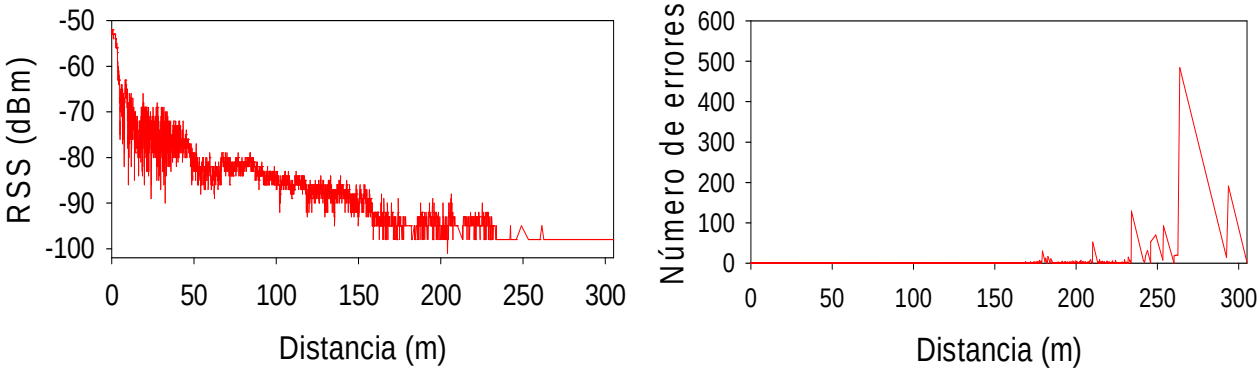


Fig. 2.38 Valores reales de (a) RSS y (b) numero de paquetes de error en función de la distancia para altura de antenas de 1 metro y 30 Km/h

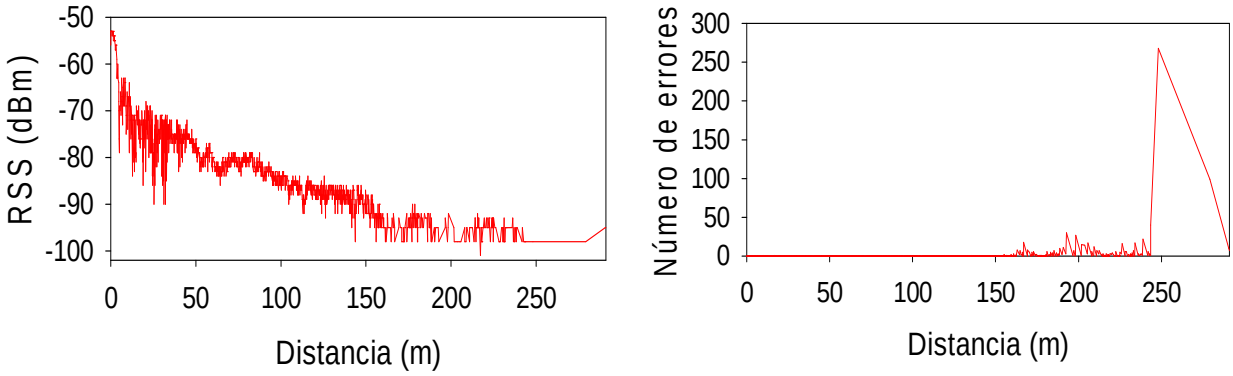


Fig. 2.39Valores reales de (a) RSS y (b) numero de paquetes de error en función de la distancia para altura de antenas de 1 metro y 60 Km/h

Puede verse además como llegando a distancias de 150 metros desde el transmisor la tasa de PER comienza a crecer desde cero así como el número de paquetes de error.
Las figuras 2.39.a y 2.39.b muestran la misma información que las anteriores pero para una velocidad de 60 Km/h. Puede verse como ambos pares de figuras son bastante similares indicando que para distancias máximas de entre 150 - 250 metros y velocidades máximas de 60 Km/h IEEE 802.15.4 presenta un buen comportamiento radio.

2.9 Sensores IEEE 802.15.4 sobre el cuerpo humano

Otra área de trabajo muy interesante que actualmente se está investigando es el efecto que el cuerpo humano tiene sobre las comunicaciones inalámbricas, tanto en su uso como canal de comunicación como en la interferencia con las señales electromagnéticas de diferentes sistemas de comunicación inalámbricos.
El cuerpo humano es un conjunto complejo formado por diferentes partes compuestas de diferentes tipos de material como huesos, líquidos y carne. Dependiendo de la naturaleza y de la morfología de dichas partes, cuando el cuerpo humano entra en contacto con un medio de transmisión inalámbrico aparece una serie de fenómenos electromagnéticos de absorción, reflexión, refracción, difracción y '*scattering*' complejos de analizar.
La caracterización electromagnética de diferentes partes del cuerpo humano ha sido ampliamente analizada para frecuencias en el rango de 10 Hz y los 20 GHz
[16][17][18].
La tabla 2.7 muestra unos pocos valores de ejemplo de los muchos disponibles para una frecuencia de trabajo de 2.45 GHz.

Tipo de tejido	Conductividad (S/m)	Permitividad relativa	Tangente de perdidas	Profundidad de penetración (m)
Sangre	2.5448	58.264	0.32046	0.016122
Fluido corporal	2.4781	68.208	0.26656	0.017846
Hueso cortical	0.80517	18.548	0.31849	0.028745
Medula ósea	0.095037	5.2969	0.13164	0.12884
Cartílago	1.7559	38.77	0.33228	0.019077
Grasa	0.10452	5.2801	0.14524	0.11702

Tabla 2.7 Tabla caracterización electromagnética partes cuerpo humano

Aunque el uso del cuerpo humano como canal no es parte de este trabajo, sí que es al menos necesario comentar la existencia del estándar IEEE 802.15.6 [19] para sistemas de comunicaciones inalámbricos dentro de o en las cercanías de un cuerpo humano. Dicho estándar propone tres diferentes tipos de capa física: una capa de banda estrecha para su uso en diferentes bandas ISM de frecuencia, otra de banda ancha para su uso en tecnologías UWB y finalmente una basada en '*Electric Field Communication*' (EFC) con una banda de trabajo centrada en 21 MHz.
En este trabajo se analizó el impacto adverso que el cuerpo humano tiene en la calidad el canal radio para comunicaciones basadas en IEEE 802.15.4. Para ello utilizaron dos modelos XBee Pro con antena 'chip' integrada y las aplicaciones Java mencionadas en el apartado 2.6 para el cálculo de la calidad del radioenlace.
Para ello un módulo XBee Pro se colocó transmitiendo a una altura de 72 cm y a una distancia de 4.5 metros del mismo se colocó otro módulo XBee Pro recibiendo la señal colocado en diferentes partes del cuerpo humano de un sujeto y a diferentes potencias de transmisión. En concreto, los puntos del cuerpo humano en donde se han colocado los dispositivos han sido el pecho y el tobillo y muñeca derechas, lugares de colocación de sensores más típicos. Las potencias de transmisión de 10 y 18 dBm respectivamente elegidas en función de la distancia entre transmisor y receptor para llevar al canal al extremo de la sensibilidad en recepción. La cantidad de paquetes enviados para cada cálculo de la calidad de enlace fue de 100000 paquetes y el mecanismo de confirmación de datos fue desactivado. Para cada punto del cuerpo humano y potencia de transmisión se realizaron cuatro medidas distintas girando 0, 90, 180 y 270 grados con respecto a la orientación entre antenas transmisora y receptora.

Después se procedió a la realización de las mismas medidas pero esta vez intercambiando el papel de los módulos, poniendo a transmitir el que antes recibía y viceversa. Los resultados se muestran en las tablas 2.8, 2.9 y 2.10.

Giro	Muñeca (altura = 84 cm)							
	Modulo fijo transmitiendo				Modulo fijo recibiendo			
	Potencia T_X 10 dBm		Potencia T_X 18 dBm		Potencia T_X 10 dBm		Potencia T_X 18 dBm	
	RSSI	PER	RSSI	PER	RSSI	PER	RSSI	PER
0º	-62,00	0,557	-48,44	0,028	-63,33	0,218	-54,98	0,032
90º	-69,39	5,481	-51,57	0,046	-65,32	0,209	-63,42	0,285
180º	-73,23	9,811	-58,65	1,661	-76,38	5,987	-61,07	0,552
270º	-65,98	2,839	-63,35	4,69	-73,85	3,099	-60,32	0,3

Tabla 2.8 Medidas de calidad de enlace para XBee en la muñeca

Giro	Pecho (altura = 1.27 cm)							
	Modulo fijo transmitiendo				Modulo fijo recibiendo			
	Potencia T_X 10 dBm		Potencia T_X 18 dBm		Potencia T_X 10 dBm		Potencia T_X 18 dBm	
	RSSI	PER	RSSI	PER	RSSI	PER	RSSI	PER
0º	-70,40	7,367	-49,70	0,011	-65,59	1,262	-52,02	0,024
90º	-57,35	0,119	-62,64	2,541	-72,44	5,928	-52,96	0,055
180º	-77,64	15,95	-60,02	0,493	-64,61	0,191	-61,51	0,569
270º	-64,84	2,246	-57,64	0,448	-62,59	0,21	-62,00	0,613

Tabla 2.9 Medidas de calidad de enlace para XBee en el pecho

Giro	Tobillo (altura = 16cm)							
	Modulo fijo transmitiendo				Modulo fijo recibiendo			
	Potencia T_X 10 dBm		Potencia T_X 18 dBm		Potencia T_X 10 dBm		Potencia T_X 18 dBm	
	RSSI	PER	RSSI	PER	RSSI	PER	RSSI	PER
0º	-76,99	15,26	-51,51	0,031	-58,71	0,108	-55,61	0,036
90º	-66,40	3,801	-57,32	0,353	-64,74	0,191	-57,81	0,151
180º	-74,31	12,37	-52,17	0,052	-71,44	1,704	-54,75	0,047
270º	-69,79	5,674	-64,49	6,755	-68,72	1,479	-56,53	0,032

Tabla 2.10 Medidas de calidad de enlace para XBee en el tobillo

Hay que comentar previamente que debido a que las antenas utilizadas fueron de tipo '*chip*', los resultados fueron influenciados por sus diagramas de radiación aunque se intentó en todo momento mantener las antenas en las mismas posiciones en todo momento. También es necesario mencionar que el modulo fijo presentaba una alineación de antena en el plano horizontal mientras que el modulo colocado en el cuerpo humano la tenía en el plano vertical.

De los datos obtenidos puede apreciarse como, como era de esperar, la potencia de transmisión es un factor determinante en la calidad final del radioenlace pero también lo es la posición del módulo inalámbrico colocado en el cuerpo humano. Puede verse como en todos los casos la mayor calidad se consigue cuando las antenas se encuentran alineadas mientras que el peor caso posible es aquel en que las antenas se encuentran alineadas pero giradas 180º y la señal tiene que atravesar el cuerpo humano completamente.

2.10 Conclusiones

Existe una gran cantidad de diferentes tecnologías de comunicación inalámbricas disponibles trabajando en diferentes bandas del espectro. Teniendo en cuenta que en este trabajo se pretende comunicar sensores y actuadores, que necesitan intercambiar poca información y en el que el consumo de energía es un aspecto importante, se ha pensado en el uso de una tecnología basada en IEEE 802.15.4. Dentro de la gran cantidad de fabricantes disponible se ha seleccionado el uso de los módulos XBee de Digi que proporcionan características simples de sensado y actuación, además de la capacidad de comunicación inalámbrica.
Se ha analizado su uso en interiores debido a su potencia uso en edificios y hogares pero debido a que estos entornos presentan un comportamiento radio complejo es necesario realizar un análisis de sus prestaciones tanto en cobertura, calidad de enlace como comportamiento frente a señales interferentes. Se ha visto como a la hora de realizar un despliegue de una red de sensores inalámbrica el uso de modelos de propagación empíricos no es suficiente y hay que utilizar algún tipo de modelo determinista.
Además, se ha analizado el comportamiento de dispositivos IEEE 802.15.4 en entornos urbanos de baja movilidad observando que presentan un comportamiento adecuado en los mismos.
Finalmente se ha estudiado el cuerpo humano como entidad interferente en comunicaciones basadas en IEEE 802.15.4 y como el emplazamiento de dispositivos inalámbricos en el cuerpo humano es determinante en la calidad final del radioenlace.

2.11 Referencias

[1] "Radio Regulations", International Telecommunication Union, 2012.

[2] Lee, J., Su, Y., Shen, C., "A Comparative Study ofWireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi", The 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON), Taipei, Taiwan, 2007.

[3] IEEE Std 802.15.4, 2006, IEEE Standard for Information Technology – Telecommunications and information exchange between systems –Local and metropolitan area networks Specific requirements; Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (WPANs), IEEE Computer Society.

[4] IEEE Std 802.11, 2007, IEEE Standard for Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks Specific requirements; Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Computer Society.

[5] IEEE Std 802.15.1, 2005, IEEE Standard for Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks Specific requirements; Part 15.1: Wireless medium access control (MAC) and physical layer (PHY) specifications for wireless personal area networks (WPANs), IEEE Computer Society.

[6] Mahanti, A., Carlsson, N., Williamson, C., Arlitt, M., "Ambient interference effects in Wi-Fi networks", Lecture Notes in Computer Science, 6091, pp. 160-173, 2010.

[7] ISO/IECStd18000-4, 2004, Information technology – Radio frequency identification for itemmanagement– Part 4: Parameters for air interface communications at 2.45 GHz.

[8] "XBee & XBee-PRO OEM RF Module Antenna Considerations", Application Note XST-AN019a, September 2005.

[9] "XBee®/XBee-PRO® RF Modules - Product Manual v1.xEx - 802.15.4 Protocol", Digi, 2009.

[10] "XBee™ Series 2 OEM RF Modules - Product Manual v1.x.1x –ZigBeeProtocol", Digi, 2007.
[11] "Wireless Mesh Networking ZigBee® vs.DigiMesh™", Digi white paper.
[12] IEEE Std 802.11p, 2010, IEEE Standard for Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks Specific requirements; Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 6: Wireless Access in Vehicular Environments , IEEE Computer Society.
[13] Sommer, C., Dressler, F., "Using the Right Two-Ray Model? A Measurement-based Evaluation of PHY Models in VANETs", Proceedings of 17th ACM International Conference on Mobile Computing and Networking (MobiCom 2011). Poster Session, Las Vegas,. NV; September 2011.
[14] Rappaport, T.S., "Wireless Communications: Principles and Practice", 2ndEdn. New Jersey, USA: Prentice Hall PTR; 2009.
[15] Xia, H., Bertoni, H. "Radio Propagation Characteristics for LOS Microcellular and Personal Communications", IEEE Trans, APS, October 1993.
[16] Gabriel, S., Lau, R.W., Gabriel, C., "The dielectric properties ofbiological tissues: II. Measurements in the frequency range 10 Hz to20 GHz," in Phys. Med. Biol., vol. 41, pp. 2251–2269.1996.
[17] Gabriel, C., "Compilation of the Dielectric Properties of Body Tissuesat RF and Microwave Frequencies," Radiofrequency Radiation Division Brooks Air Force Base, Ed., TX, 1996.
[18] "Calculation of the Dielectric Properties of Body Tissues," Institute for Applied Physics – Italian National Research Council, 2007,disponible en: http://niremf.ifac.cnr.it/tissprop/htmlclie/htmlclie.htm, últimoacceso: 04 Marzo 2013.
[19] IEEE Std 802.15.6, 2012, IEEE Standard for Information Technology – Telecommunications and information exchange between systems –Local and metropolitan area networks Specific requirements; Part 15.6: Wireless Body Area Networks, IEEE Computer Society.

CAPITULO 3 – INTEGRACION KNX

3.1 Introducción

Es un estándar abierto para todas las aplicaciones de control de la vivienda y el edificio, como por ejemplo el control de la iluminación y las persianas, así como variados sistemas de seguridad, calefacción, ventilación, aire acondicionado, monitorización, alarma, control de agua, gestión de energía, contador, así como electrodomésticos del hogar, audio/video y mucho más. Esta tecnología puede ser usada tanto en viviendas como en edificios de nueva construcción, como en los ya existentes.KNX está aprobado como Estándar Internacional (ISO/IEC 14543-3) [1], así como Estándar Europeo (CENELEC EN 50090 [2] y CEN EN 13321-1 [3]), Estándar Norteamericano (ANSI/ASHRAE 135 [4]) y Estándar en China (GB/Z 20965 [5]) por lo que el futuro de KNX está garantizado. Existe un gran número de fabricantes que manufacturan dispositivos KNX ofreciendo una gran funcionalidad dentro de los sistemas de automatización en edificios y gracias a la estandarización se garantiza que los productos de diferentes fabricantes puedan ser interconectados. Esto implica también que no haya limitación a la hora de depender de un único fabricante y si uno de ellos dejara de fabricar un producto, sería sencillo encontrar un dispositivo de similares características en otros fabricantes.
En las instalaciones tradicionales cada función requiere una línea eléctrica propia, y cada sistema de control precisa una red separada. Por el contrario, con KNX se pueden controlar, comunicar y vigilar todas las funciones de servicio y su desarrollo, con una única línea común. Con esto se puede dirigir la línea de energía sin desvíos, directamente hasta el aparato final.
La instalación en un edificio se puede realizar de un modo más sencillo desde el principio, y después se puede ampliar y modificar sin problemas. Ante cambios de uso o reorganización del espacio, KNX consigue una adaptación rápida y sin problemas, mediante una fácil ordenación (cambio de parametrización) de los componentes del bus, sin necesidad de implementar un nuevo cableado.
Este cambio de parametrización se realiza con un PC, conectado al sistema KNX, que tenga instalado el software '*EIB Tool Software*' (ETS) para proyecto y puesta en servicio, que ya se emplea en la primera puesta en marcha. En KNX se puede conectar mediante las correspondientes interfaces con los centros de control de otros sistemas de automatización de edificios o con una red digital de servicios integrados (RDSI). De este modo el uso del KNX en una vivienda unifamiliar resulta tan rentable como en hoteles, escuelas, bancos, oficinas o edificios del sector terciario.
Como puede verse en la Tabla 3.1 KNX soporta varios tipos de medios de transmisión siendo el tipo de par trenzado el más utilizado.

	Medio	**Velocidad de transmisión**
TP	Par trenzado	9600 bits/s
PL	Línea de fuerza eléctrica	1200 bits/s
RF	Radiofrecuencia Banda 868 MHz	16384 Kbit/s
IP	Red Internet basado en KNXNet/IP	Dependiente de la red

Tabla 3.1 Medios de transmisión en KNX

3.2 Acceso a dispositivos KNX a través de USB

Los dispositivos utilizados para acceder elementos KNX a través de un puerto USB se conocen como interfaces KNX/USB y el protocolo utilizado se encuentra descrito en la nota de aplicación037/02 Rev. 4[6] dentro del estándar KNX. Estos dispositivos utilizan la especificación '*Human Interface Device*' (HID) [7] para intercambiar las tramas KNX. El objetivo de dicha clase consiste básicamente en definir el uso de dispositivos usados por humanos para controlar la operación de sistemas computarizados. Aunque la asociación KNX no requiere una certificación USB para que los interfaces KNX / USB puedan trabajar con herramientas KNX, al menos deben cumplir las especificaciones USB versión 1.1.
Los datos son transferidos en tramas llamados '*Reports*' cuya carga útil está limitada a 64 octetos. La estructura de la trama se muestra en la Fig. 3.1.
El campo 'Report ID' indica el tipo el mismo y presenta un valor fijo de "01h" para indicar que es un tipo de intercambio de datos KNX. Este campo permite al controlador de la clase HID distinguir entre los diferentes tipos de datos de entrada. El campo del número de secuencia está reservado para un uso futuro y el campo de tipo de paquete indica si el envío de datos empieza y termina con el mismo paquete, si contiene datos parciales, si contiene el comienzo y datos parciales o si contiene datos parciales y el final de la comunicación.

KNX HID Report Header				**KNX HID Report Body**
ReportID	PacketInfo		Datalength	Data
	Sequence Number	Packet Type		
1 Octet	1 Octet		1 Octet	Maximum 61 Octets

Fig. 3.1 Estructura de un HID Report

El campo de datos del '*Report*' KNX HID consiste en la cabecera del protocolo de transferencia KNX USB y el cuerpo del mismo, cuya estructura se muestra en la Fig. 3.2.

KNX HID Report Body							
KNX USB Transfer Protocol Header						**KNX USB Transfer Protocol Body**	
Protocol Version	Header Length	Body Length	Protocol ID	EMI ID	Manufacturer Code	EMI Message Code	Data (cEMI/EMI1/EMI2)
1 Octet	1 Octet	2 Octet	1 Octet	1 Octet	2 Octet	1 Octet	Max. 52 Octets

Fig. 3.2 Ejemplo de estructura del cuerpo de un HID Report

El campo de versión de protocolo indica el estado de la revisión del protocolo de transferencia KNX USB utilizado en la trama. El campo de longitud de la cabecera indica el número de octetos de componen dicha cabecera. En la actualidad el único valor admitido es el '0' indicando una cabecera de 8 octetos de longitud. El campo de longitud del cuerpo indica el número de octetos que compone dicho cuerpo. Debido a que un interface conectado a un PC a través de un puerto USB puede no sola mente enviar tramas KNX sino también otros protocolos, el campo de 'Protocol ID' indica el tipo utilizado en la trama. Por ejemplo, para indicar el uso de '*tunneling*' KNX este campo toma un valor de '01h' y si el protocolo utilizado es '*Bus Access Server Feature*', '0Fh'.
Si se utiliza '*tunneling*', el campo de identificador de '*External Message Interface*' (EMI) representa el formato EMI utilizado en el cuerpo de la transferencia KNX USB. Un valor de '01h'

indica que el formato EMI utilizado es EMI1, '02h' se utiliza para EMI2 y finalmente un valor de '03h' indica que el formato usado es cEMI. En la versión '0' del protocolo del campo de identificador de protocolo siempre debe estar presente. En el caso de se esté utilizando '*tunneling*' de la capa de enlace KNX este campo debe presentar un valor de '0000h'.Cuando se utiliza '*tunneling*' KNX las tramas son "entuneladas" a través del enlace USB utilizando uno de los formatos EMI disponibles con un '*timeout*' de 1s. En dicho intervalo de tiempo el interface KNX USB debe ser capaz de recibir una trama de '*tunneling*', transmitirla a través del medio KNX y devolver una trama de confirmación al dispositivo transmisor.
Para acceder a las características del dispositivo USB se utiliza el protocolo '*Bus Access Server Feature*'. Dependiendo del campo de identificador de servicio contenido en su cabecera la trama puede ser una petición de característica (con valor '01h'), respuesta a una petición de característica ('02h'), establecimiento de una característica ('03h') o de petición de información de una característica ('04h'). Existen varias características de dispositivo disponibles y la tabla 3.2 muestra las más útiles.

Feature Identifier	Feature Name	Description	Data Length
01h	Supported EMI type	Getting the supported EMI type(s)	2 octets (B_{16})
03h	Bus connection status	Getting and informing on the bus connection status	1 bit (B_1)
05h	Active EMI type	Getting and Setting the EMI type to use.	1 octet (N_8)

Tabla. 3.2 Ejemplo de estructura del cuerpo de un HID Report

Para el acceso a dispositivos KNX a través de un interface KNX/USB en este trabajo se utilizó la librería JAVAHIDAPI [8] para intercambiar las tramas de tipo '*Report*' KNX HID y el esquema seguido se muestra en la Fig. 3.3.

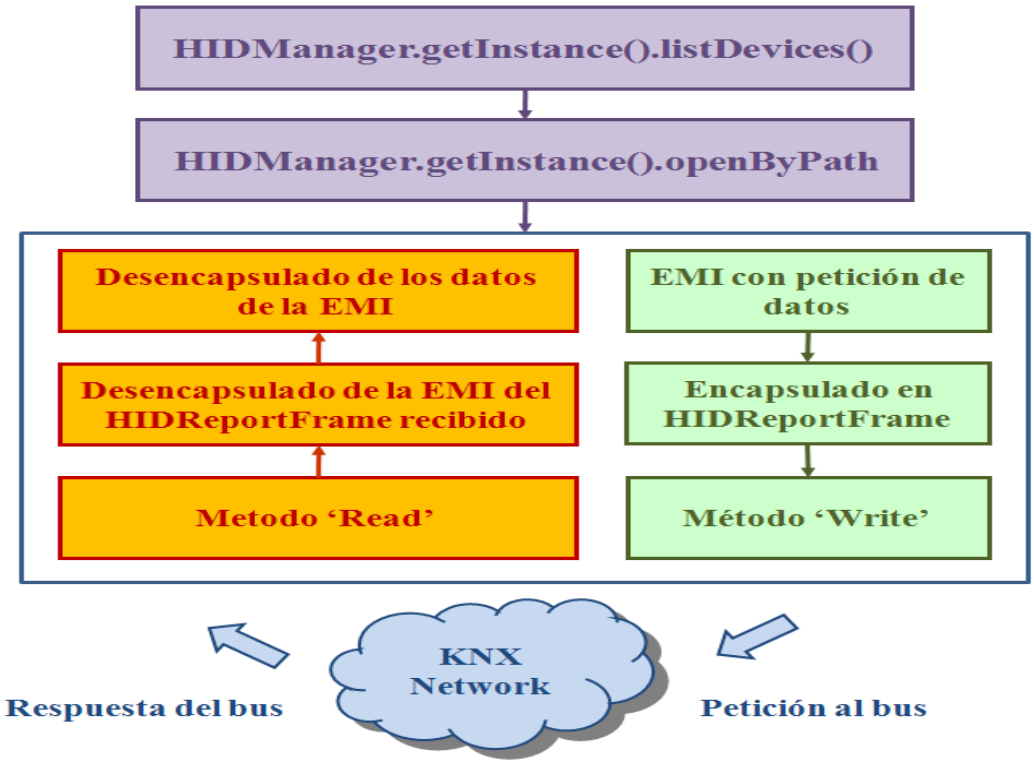


Fig. 3.3 Esquema acceso a interfaces KNX USB desde Java

Esta librería es una adaptación *'Java Native Interface'* (JNI) de la librería HIDAPI programada en C++ y permite acceder a dispositivos USB y Bluetooth basados en la clase HID en diferentes sistemas operativos tales como Windows, Linux y Mac OS X. Es importante mencionar que debido a que la librería es una adaptación de otra programada en C++, para su uso en sistemas Windows es necesario la instalación del Microsoft .Net Framework. Para acceder al listado de todos los dispositivos de tipo HID conectados al sistema la librería JKNXUSB utiliza el método 'HIDManager.getInstance().listDevices()' de la librería JAVAHIDAPI.
De esa lista se selecciona el interface KNX/USB que se desea utilizar y se abre la conexión mediante el método 'HIDManager.getInstance().openByPath' que entrega una estructura de tipo 'HIDDevice' asociada. Una vez de que el dispositivo está correctamente conectado ya es posible el envío y recepción de 'Reports' HID. Se crea un hilo de ejecución que se pone a la escucha del bus KNX mediante el método 'Read' de la estructura 'HIDDevice' correspondiente.
Cada vez que un dispositivo envía un telegrama KNX a través del bus el hilo de recepción recibe una trama de tipo 'Report' KNX HID en forma de estructura 'HIDReportFrame'. Una vez de que se dispone de los datos que componen el 'Report' es necesario extraer los datos EMI que contiene. Finalmente, en función del tipo de datos presente en la EMI es necesario decodifícala para obtener los datos finales del dispositivo KNX.
En ocasiones los elementos KNX se programan para que envíen datos únicamente bajo demanda. En ese caso, si se quiere obtener información de ellos, es necesario el envío correspondiente de una trama EMI de petición de datos encapsulada en un 'Report' KNX HID con estructura tipo 'HIDReportFrame' y enviarla mediante el método 'Write' de la estructura 'HIDDevice' asociada al dispositivo.

3.3 Acceso a dispositivos KNX a través de redes IP

Una de las características más interesantes del estándar KNX es que dentro del mismo se tiene contemplado el uso de redes IP como medio de comunicación. Esto posibilita el acceso remoto a dispositivos KNX con la consiguiente flexibilidad de monitorización y control de las instalaciones basadas en esta tecnología.

3.3.1 KNXNet/IP

La especificación KNXNet/IP [9][10][11] define la integración de la implementación del protocolo KNX sobre la capa de red IP. El protocolo KNXNet/IP se utiliza como túnel o enrutado de datos KNX a través de Internet, permitiendo un acceso remoto y un mantenimiento a través de largas distancias así como la inclusión de redes KNX en redes de alta velocidad. Para el transporte de las trama KNXNet/IP es posible utilizar TCP, UDP o incluso IP puro aunque desde un punto de vista de aplicación el uso de IP no es muy común y la especificación se ciñe estrictamente a TCP o UDP.
Como puede verse en la Fig. 3.4 KNXNet/IP presenta una arquitectura cliente/servidor. Todas las tramas KNX deben incluir una cabecera que consiste en la versión del protocolo utilizada, la información de la longitud de la carga útil que transporta y el tipo de identificador de servicio KNXnet/IP. La implementación de 'Tunneling' sobre la capa de enlace de datos KNX es obligada por el estándar entendiendo este concepto como el envío de un telegrama KNX dentro de un paquete IP y la posterior espera de la respuesta asociada o en su defecto la expiración del '*timeout*' correspondiente. La Fig. 3.4 muestra además el procedimiento de establecimiento y fin de conexión. Primero el cliente KNXNet/IP envía una trama de tipo 'CONNECT_REQUEST' a él servidor con una dirección IP conocida por el cliente. Esta trama contiene la dirección IP del cliente y será utilizada por el servidor para el envío de datos durante el resto del tiempo de conexión. Acto seguido el servidor responde enviando al cliente una trama de tipo 'CONNECT_RESPONSE' que contiene el estado de la petición y el identificador de canal de comunicación asociado a dicha conexión que será utilizado para identificar dicha conexión mientras dure esta. Tanto el cliente

como el servidor deben almacenar dicho identificador incluyéndolo en el resto de tramas a intercambiar para autentificarlas y saber quien las envía.

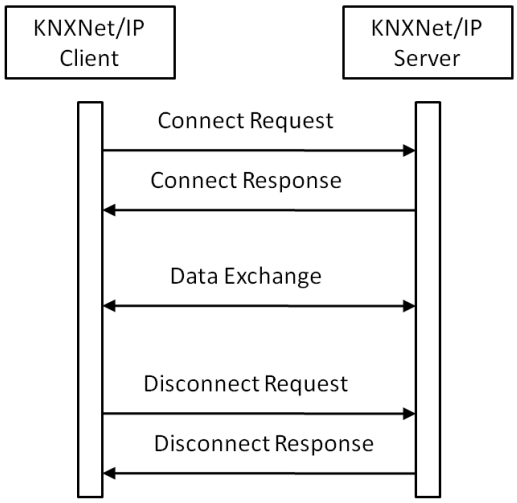


Fig. 3.4 Arquitectura cliente/servidor de KNXNet/IP

Los telegramas KNX deben enviarse siempre dentro de tramas KNXNet/IP de tipo 'TUNNELLING_REQUEST' y el tipo de formato de EMI tiene que ser necesariamente de tipo cEMI.

3.3.2 Librería Calimero

La librería 'Calimero' [12], desarrollada por la Universidad Tecnológica de Viena (Vienna University of Technology). Es una implementación en lenguaje Java de la parte cliente del protocolo KNXnet/IP que permite la comunicación con una instalación KNX mediante encapsulado IP y protocolo UDP. La librería consta de una serie de clases e interfaces, que se deben utilizar o implementar según las necesidades del software a desarrollar. En concreto, el proceso seguido para acceder a dispositivos KNX, siguiendo las directrices de Calimero, es el que se puede apreciar en la Fig. 3.5:

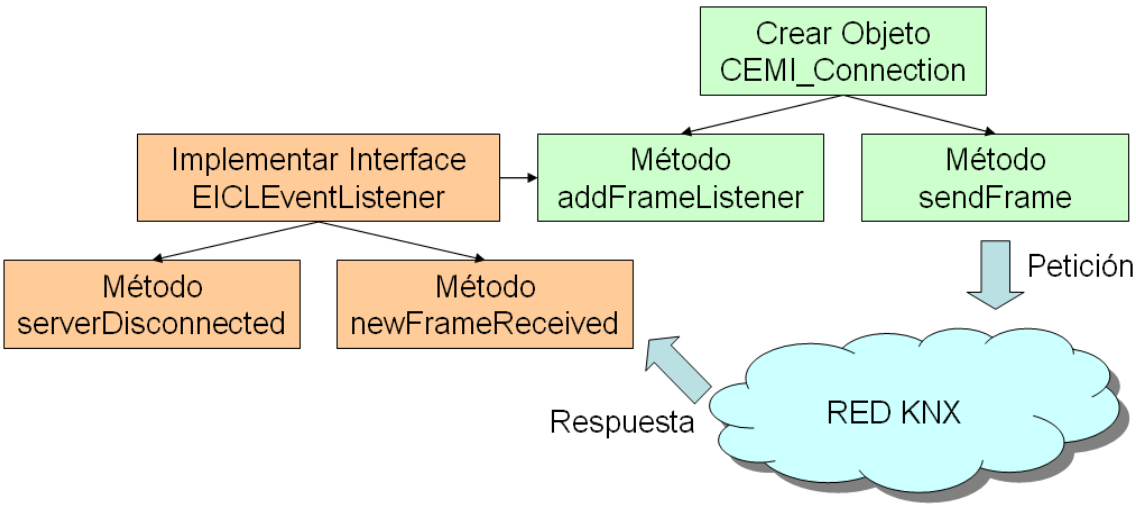


Fig. 3.5 Esquema de uso librería Calimero

Los bloques en verde representan clases y métodos ya implementados en la librería Calimero mientras que los bloques de fondo naranja representan esqueletos en los que se introduce el código deseado por el usuario y a implementar por el mismo.
El primer paso consiste en la creación de un objeto de tipo 'CEMI_Connection', que representa la conexión a nivel lógico con el interfaz KNX. Posteriormente se debe implementar con el código deseado los métodos 'serverDisconnected' y 'newFrameReceived' del interfaz 'EICLEventListener', que serán a los que se llame cuando la conexión se termine y cuando se recibe un nuevo paquete, respectivamente. A continuación se debe activar el interfaz 'EICLEventListener', utilizando el método 'addFrameListener' que lo activará y lo pondrá a la escucha. Finalmente solo queda mandar paquetes de peticiones de datos mediante el método 'sendFrame' a las direcciones de grupo que se desee para que estas manden un paquete de respuesta, activando el método 'newFrameReceived' a su llegada para su procesamiento.

3.3.3 Librería JKNXNetIP

En este trabajo, como se menciona más adelante, se utilizó la librería Calimero con resultados satisfactorio aunque hay que mencionar que implementa diferentes características que quizás en aplicaciones sencillas no sean del todo necesarias y en casos específicos no siempre puede configurarse tanto como fuese deseable. Por estos motivos se procedió al desarrollo de una librería Java propia a la que se llamó JKNXNetIP. Al igual que Calimero implementa la parte cliente del protocolo KNXNet/IP pero, como puede verse en la tabla en letra verde, únicamente los servicios imprescindibles para realizar la conexión y desconexión.

Nombre del Servicio	Código del Servicio	Implementación
SEARCH_REQUEST	0x0201	No Implementado
SEARCH_RESPONSE	0x0202	No Implementado
DESCRIPTION_REQUEST	0x0203	No Implementado
DESCRIPTION_RESPONSE	0x0204	Implementado
CONNECT_REQUEST	0x0205	Implementado
CONNECT_RESPONSE	0x0206	No Implementado
CONNECTIONSTATE_REQUEST	0x0207	No Implementado
CONNECTIONSTATE_RESPONSE	0x0208	Implementado
DISCONNECT_REQUEST	0x0209	Implementado
DISCONNECT_RESPONSE	0x020A	No Implementado

Tabla 3.3 Servicios implementados en la librería JKNXNetIP

Además, Calimero implementa multitud de tipos diferentes de datos, muchos de los cuales raramente se utilizan. La librería JKNXNetIP únicamente implementa los de uso más común, mostrados en la Fig. 3.6.

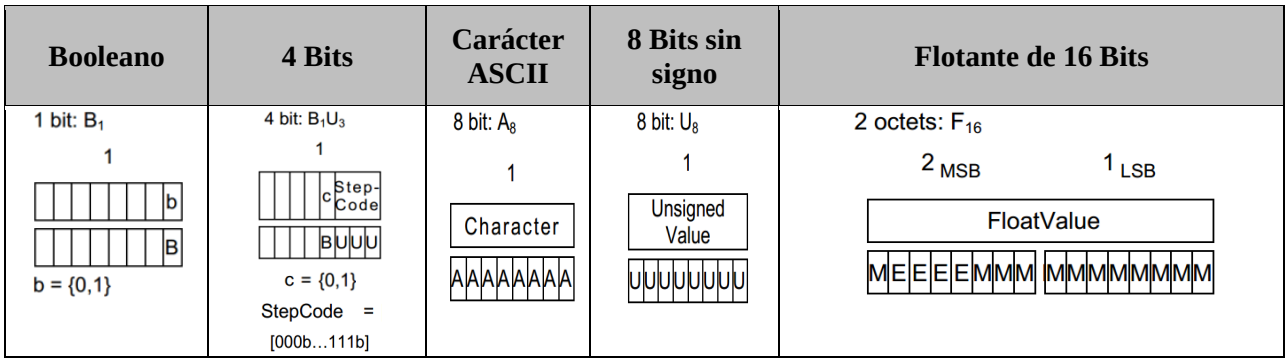


Fig. 3.6 Tipos de datos soportados por la librería JKNXNetIP

El hecho de que el código de dicha librería sea propio proporciona gran flexibilidad a la hora de realizar cambios en el mismo y dado que como hemos dicho anteriormente la implementación es mínima, el tamaño de la librería y la carga computacional asociada a la misma tiene a ser mínima también. Cabe destacar que la librería se programó para su tanto con UDP como por TCP aunque debido a que no se dispuso de un interface KNX/USB con soporte TCP no se pudo probar el funcionamiento de dicho código.

3.3.4 Problemas de conectividad en KNXNet/IP

Si se analiza el mecanismo de conexión KNXNet/IP puede apreciarse dos hechos importantes que pueden llegar a dar problemas de conectividad. El primero de ellos es que, como se ha comentado anteriormente, el cliente debe conocer previamente la dirección IP del servidor y esta debe permanecer sin cambios durante todo el tiempo de conexión. En la actualidad existen multitud de tecnologías de acceso a Internet. En muchas de ellas, como DSL, GPRS, etc. el ISP asigna a los dispositivos conectados una dirección IP dinámica. Esta dirección IP cambia sin previo aviso con el paso del tiempo y el tiempo que permanece sin cambio depende el ISP en cuestión. Como se ha comentado previamente en el apartado 1.5, la solución a este problema pasa por que el dispositivo que presenta una dirección IP dinámica debe implementar un servicio de DDNS.
El segundo de ellos es que el servidor adquiere la dirección IP del cliente una única vez, justo después de recibirla en la correspondiente trama ‘CONNECT_REQUEST’ y la utiliza el resto de la conexión para enviar los datos. Si la dirección IP del cliente cambia durante la conexión el servidor enviará, sin ningún tipo de conocimiento de ello, el resto de las tramas a la antigua dirección IP incorrecta. En este caso es necesaria la implementación de un mecanismo para que el dispositivo que presenta una dirección IP dinámica sea capaz de detectar un cambio en la misma y al mismo tiempo comunicárselo a la otra parte. Por ese motivo, tanto el cliente como el servidor deben comprobar periódicamente su dirección IP en un tercer servidor que proporcione este servicio.
Cuando el cliente o el servidor detectan un cambio en su dirección IP, en este trabajo se propone el uso de la trama mostrada en la Fig. 3.7 informando a la otra parte de la nueva dirección IP que debe utilizar en el futuro. Para futuras referencias esta trama pasará a llamarse ’ADDRESS_CHANGE_NOTIFICATION’ y el número del identificador de servicio asociado el ‘020Bh’. Los 6 primeros octetos representan la cabecera KNXNet/IP típica y el siguiente octeto el identificador del canal de comunicación asignado a la conexión.

1	06h	Header Size
2	11h	Protocol Version
3	02h	Service Type Identifier 020Bh
4	0Bh	
5	00h	Payload Length 7 Octets
6	07h	
7	01h	Communication Channel Identifier, e. g. 01h

Fig. 3.7 Trama ADDRESS_CHANGE_NOTIFICATION

Cuando se recibe una trama de tipo 'ADDRESS_CHANGE_NOTIFICATION' la parte receptora debe actualizar la dirección IP del emisario de la misma y devolverle una trama de confirmación. Para futuras referencias esta trama pasará a llamarse 'ADDRESS_CHANGE_NOTIFICATION_ACK' y el número del identificador de servicio asociado el '020Ch'. La estructura de dicha trama puede verse en la Fig. 3.8 y el último octeto representa el código de estado del proceso de actualización de la dirección IP. Si no se ha encontrado ningún error en el mismo, el valor de este campo es '00h'.

1	06h	Header Size
2	11h	Protocol Version
3	02h	Service Type Identifier 020Ch
4	0Ch	
5	00h	Payload Length 8 Octets
6	08h	
7	01h	Communication Channel Identifier, e. g. 01h
8	00h	Status Code (NO_ERROR)

Fig. 3.8 Trama ADDRESS_CHANGE_NOTIFICATION_ACK

Cuando se recibe una de estas tramas, dependiendo de si el protocolo utilizado es TCP o UDP es necesario tomar diferentes acciones. Para establecer una conexión TCP cada dispositivo debe intercambiar una serie de paquetes de señalización y si una de las direcciones IP cambia, también lo hace el socket TCP asociado y la sesión TCP se pierde. En este caso se propone que el dispositivo cuya dirección IP cambie debe mandar a la otra parte una trama de tipo 'DISCONNET_REQUEST' y cerrar el enlace KNXNet/IP.
El protocolo UDP no está orientado a conexión y si la dirección IP de una de las partes cambia, el viejo socket UDP asociado en la otra parte puede ser reemplazado por uno nuevo con la dirección IP actualizada sin ningún tipo de problema. Además y debido a la naturaleza De envío de datos del protocolo UDP, se propone que en todo momento si el servidor KNXNet/IP recibe una trama de petición de datos del cliente, este obtenga la dirección IP correspondiente de la cabecera del paquete UDP asociado y mande la respuesta a dicha dirección. De esta forma, si existen problemas de conexión debidos a una dirección IP dinámica se garantiza que los paquetes de respuesta de datos se envían siempre al destino correcto.

3.3.5 Problemas de seguridad en KNXNet/IP

Para proteger la información intercambiada por el cliente y el servidor KNXNet/IP es necesario el encriptado de los datos. De esta manera un posible atacante puede acceder a los datos encriptados pero no puede obtener la información original de los mismos o inyectar código malicioso en los paquetes IP para una activación no autorizada de actuadores KNX [13][14][15].

En este trabajo se propone el uso de un algoritmo de clave simétrica para encriptar todos los datos de una trama KNXNet/IP salvo la cabecera. Leyendo esta última puede accederse a la versión del protocolo usado en la trama y determinar si se está utilizando el viejo o la nueva versión propuesta y necesita desencriptado.
Los algoritmos de encriptado simétrico utilizan la misma clave tanto para la encriptación como el desencriptado. La clave representa un secreto compartido por dos o más entidades que puede usarse para la protección de datos privados transmitidos a través de un medio inseguro. Este tipo de algoritmos pueden ser de tipo bloque o de tipo flujo. Los algoritmos de tipo bloque encriptan un conjunto de bits como una unidad individual, rellenando el número de datos de la información original hasta que este alcance un tamaño múltiplo al tamaño de bloque y los algoritmos de tipo flujo realizan la encriptación bit a bit.
Existen multitud de algoritmos basados en clave simétrica de tipo bloque tales como MARS, RC6, Serpent,Twofish, Rijndael (más conocido en la actualidad como AES), etc. Existe multitud de literatura acerca de estos algoritmos, con diferentes comparativas y análisis [16][17] y es complicado resumir toda esta información en unas cuantas líneas pero básicamente podría decirse que el más seguro pero también el más lento es el Serpent, Rijnadel es el más rápido y Twofish representa una soluciona caballo entre ambos.
El número de algoritmos de clave simétrica de tipo flujo es también elevado pudiendo elegir entre algunos como HC-128, Rabbit, Salsa20, SOSEMANUK, Grain , MICKEY, Trivium, etc.
También en este caso hay gran cantidad de publicaciones que analizan el rendimiento de estos algoritmos [18][19][20] y las expectativas de si un algoritmo es eficiente dependen de la aplicación específica de uso. Es complicado esperar que una única implementación satisfaga todas las necesidades.
En este trabajo se ha propuesto el uso del algoritmo 'Salsa20' porque en general presenta unas mejores características que AES, posee un gran rendimiento software [21], una velocidad media/alta y un área media en implementaciones hardware de tipo '*Field Programmable GateArrays'* (FPGA) [22] y es una alternativa interesante a AES en microprocesadores AVR de 8-bits [23].

3.3.6 Problemas de autentificación en KNXNet/IP

Encriptando la carga útil de las tramas KNXNet/IP se evita el acceso no autorizado a la información que transportan pero no previene que clientes desconocidos y no autorizados se conecten al servidor y pidan información, incluso si no pueden desencriptarla. Por lo tanto parece claro que se necesita algún tipo de mecanismo de autentificación.
Para proveer al protocolo KNXNet/IP de este mecanismo se propone utilizar criptografía basada en clave pública. Este tipo de algoritmos dos claves diferentes, una de las cuales es secreta y la otra pública y dicho par de claves esta enlazados matemáticamente. La clave pública se utiliza para encriptar el mensaje mientras que la clave privada se utiliza para desencriptarlo. La clave pública puede ser enviada a través de un medio no seguro mientras que la clave privada asociada debe permanecer en secreto en todo momento y no ser revelada a nadie no autorizado a acceder a la información.
Estos métodos criptográficos utilizan algoritmos matemáticos asimétricos basados en problemas matemáticos en los cuales es computacionalmente sencillo generar las claves pública y privada, desencriptar el mensaje cifrado conociendo la clave privada asociada y encriptar el mensaje utilizando la clave pública pero es extremadamente complicado derivar la clave privada o el mensaje original basándose únicamente en el conocimiento de la clave pública. Existen multitud de algoritmos disponibles de este tipo entre los cuales se incluyen RSA, ECC, Diffie–Hellman, NTRU, etc. También en este case existe gran variedad de literatura que analizan las características de dichos algoritmos [24][25] y muestran comparativas de los mismos [26]. En este trabajo se ha decidió utilizar NTRU porque presenta un mejor rendimiento que el resto de sus competidores [27][28] y incluso que algún algoritmo de clave simétrica [29].

3.3.7 Propuesta de solución a los problemas de seguridad y autentificación en KNXNet/IP

Una vez decididos los algoritmos a utilizar el siguiente paso es describir el mecanismo propuesto en este trabajo para solucionar los problemas anteriormente comentados. Para realizar la conexión primero el cliente debe crear una par de claves NTRU y enviar al servidor una trama modificada de tipo 'CONNECT_REQUEST', cuya estructura se muestra en la Fig. 3.9. El primer octeto después de la cabecera representa la versión del protocolo utilizada, el siguiente indica el tipo de conexión y el último representa que se está utilizando la capa KNX. Finalmente, el resto de los octetos representan la clave pública NTRU del cliente.

1	06h	Header Size
2	11h	Protocol Version
3	02h	Service Type Identifier 0205h
4	05h	
5	…	Payload Length
6	…	
7	01h	Host Protocol Code, e.g. 01h, for UDP over IPv4
8	04h	Connection Type Code, e.g. 04h, TUNNEL_CONNECTION
9	02h	KNX Layer, e.g. TUNNEL_LINKLAYER
10	…	Client's NTRU Public Key
	…	
N	…	

Fig. 3.9 Trama CONNECT_REQUEST modificada propuesta

Una vez que el servidor recibe una trama de tipo 'CONNECT_REQUEST', este debe obtener la clave pública NTRU del cliente, generar una secuencia aleatoria y encriptarla condicha clave. Acto seguido debe enviar una trama de petición de autentificación con la estructura mostrada en la Fig. 3.10. Para futuras referencias esta trama pasará a llamarse 'AUTHETICATION_REQUEST' y el número del identificador de servicio asociado el '020Dh'

El primer octeto después de la cabecera representa el identificador de canal de comunicación temporal asignado a la conexión y el siguiente indica si ha ocurrido algún error en el proceso. Si no ha habido error, el valor de este campo es '0h'.Finalmente el resto de los octetos representan la secuencia aleatoria encriptada con la clave pública NTRU del cliente.

1	06h	Header Size
2	11h	Protocol Version
3	02h	Service Type Identifier 020Dh
4	0Dh	
5	…	Payload Length
6	…	
7	01h	Temporal Communication Channel Identifier, e. g. 01h
8	00h	Status Code (NO_ERROR)
9	…	Random Sequence encrypted with Client's Public Key
	…	
N	…	

Fig. 3.10 Trama AUTHETICATION_REQUEST propuesta

Cuando el cliente recibe una trama de tipo 'AUTHETICATION_REQUEST', este tiene que obtener la secuencia aleatoria encriptada, desencriptada utilizando su clave privada NTRU y volverla a encriptar pero esta vez utilizando la clave Salsa20 que conocen tanto el cliente como el servidor. Acto seguido debe enviar al servidor una trama de respuesta de autentificación cuya estructura se muestra en la Fig. 3.11. Para futuras referencias esta trama pasará a llamarse 'AUTHETICATION_RESPONSE' y el número del identificador de servicio asociado el '020Eh'. El primero octeto después de la cabecera representa el identificador de canal de comunicación temporal, el siguiente indica si ha habido algún tipo de error y el resto de octetos representan la secuencia aleatoria encriptada con la clave Salsa20. Cuando el servidor recibe una trama de tipo 'AUTHETICATION_RESPONSE' este debe obtener la secuencia aleatoria encriptada, desencriptarla utilizando su clave Salsa20 que tiene que ser igual a la del cliente y compararla con la secuencia aleatoria que él había generado. Si las secuencias son idénticas eso quiere decir que tanto el cliente como el servidor comparten clave Salsa20 y el proceso de autentificación está completo.

1	06h	Header Size
2	11h	Protocol Version
3 4	02h 06h	Service Type Identifier 0206h
5 6	00h 0Ah	Payload Length 10 Octets
7	01h	Temporal Communication Channel Identifier, e. g. 01h
8	00h	Status Code (AUTHORIZED)
9	01h	Host Protocol Code, e.g. 01h, for UDP over IPv4
10	0x04	Connection Type Code, e.g. 04h, TUNNEL_CONNECTION

Fig. 3.11 Trama AUTHETICATION_RESPONSE propuesta

El primer octeto después de la cabecera representa el identificador de canal de comunicación a utilizar durante el resto de la conexión, el siguiente indica si la conexión se ha establecido o no y finalmente el último octeto indica el tipo de conexión.

1	06h	Header Size
2	11h	Protocol Version
3 4	02h 0Eh	Service Type Identifier 020Eh
5 6	… …	Payload Length
7	01h	Communication Channel Identifier, e. g. 01h
8	00h	Status Code (NO_ERROR)
9 N	… … …	Random Sequence encrypted with Salsa20 Key

Fig. 3.12 Trama CONNECT_RESPONSE modificada propuesta

Finalmente, después de que el servidor recibe una trama de tipo 'AUTHETICATION_RESPONSE', este envía al cliente una trama modificada de tipo 'CONNECT_RESPONSE' indicando al cliente si la conexión está establecida y autorizada o no. La estructura propuesta de dicha trama se muestra en la Fig. 3.12.

Para probar la solución propuesta se llevo a cabo su implementación real en una pequeña instalación KNX montada para tal efecto, mostrada esquemáticamente en la Fig. 3.13.

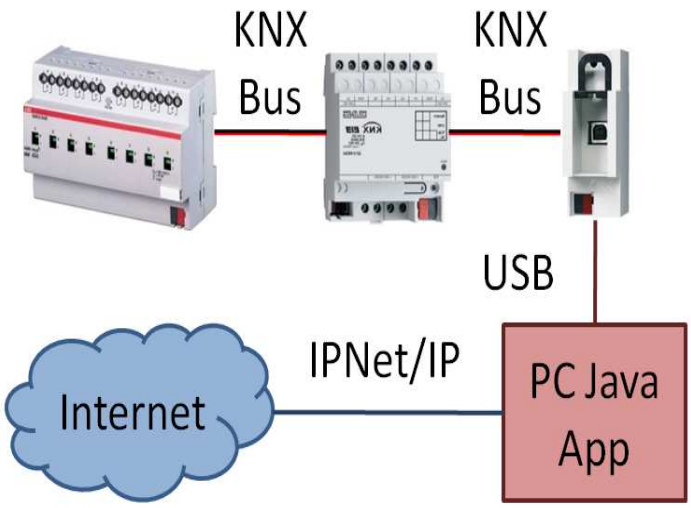


Fig. 3.13 Instalación prueba protocolo propuesto

Dicha instalación consistió en un módulo de cuatro entradas analógicas modelo 2214 REG A del fabricante Jung y un módulo de ocho canales actuadores modelo SA/S 8.10.1 de Siemens. Para acceder al bus KNX la instalación incluía un interface KNX / USB modelo 2130 USB REG de Jung. Dicho interface se conectó a un puerto USB de un viejo PC con procesador Pentium 4 y 1 GByte de memoria RAM con Windows XP SP3 instalado.
Además, se procedió a la implementación de la parte servidor del protocolo KNXNet/IP propuesto programada en Java y ejecutándose en dicho PC. Dicha aplicación obtenía los daos del bus KNX siguiendo la nota de aplicación 037/02 del estándar KNX de acuerdo a como se explica en el apartado 3.2. Además utiliza la librería Java BouncyCastle Crypto [30] para el uso del algoritmo Salsa02 y también la librería java que implementa el algoritmo NTRU disponible en la página web 'The NTRU Project' [31].
Para dotar al servidor de un nombre de red estático y bien conocido se utilizó el servicio gratuito de DDNS suministrado por el sitio 'no-ip' [32]. Para su utilización, primero hay que configurar el servicio y el nombre de red asignado al equipo y después es necesaria la instalación de una pequeña aplicación llamada '*Dynamic Update Client*' (DUC) que es la encargada de comprobar el cambio de dirección IP. Como se ha comentado previamente en el apartado 1.5, en el caso de uso de direcciones IP dinámicas, tanto el cliente como el servidor necesitan un mecanismo para conocer cuando su dirección IP ha cambiado. Como se dijo anteriormente el método más común es comprobar periódicamente su dirección IP en un servidor externo destinado a tal fin.
Existen varios sitio en Internet que proveen gratuitamente de este servicio y el mecanismo es básicamente el mismo. Consiste en realizar una petición HTTP de tipo GET a dicho servidor par que este último responda con la dirección actual del cliente en formato de cadena de caracteres. En este trabajo se utilizó el servicio proporcionado por el sitio web 'dyndns' [33] con buenos resultados.
También es posible implementar este servicio mediante un sencillo script programado en PHP y ejecutándose en un servidor web. En este trabajo también se contempló esta opción y se colocó un script PHP programado para tal efecto en un servidor web gratuito proporcionado por el sitio 'FreeHostingEU' [34] con buenos resultados. Finalmente, para probar la instalación y la aplicación Java implementando la parte servidor del protocolo KNXNet/IP propuesto también se realizó la implementación Java de la parte cliente correspondiente.

Para comprobar la correcta estructura y el contenido de las tramas KNXNet/IP intercambiadas entre cliente y servidor en el establecimiento de la conexión se llevó a cabo una captura de paquetes mediante la herramienta de análisis de protocolos de red Wireshark [35]. Las figuras desde 3.14 a 3.17 muestran las capturas de dichas tramas. Debido a la gran cantidad de datos que contienen las tramas de tipo 'CONNECT_REQUEST' y 'AUTHENTICATION_REQUEST' las figuras asociadas muestran únicamente una porción de los datos originales pero suficientes para observar su estructura y contenido general.

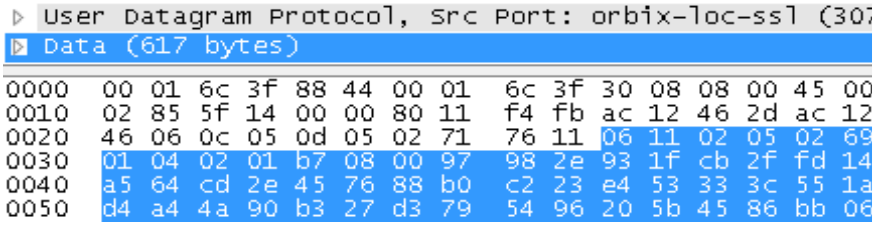


Fig. 3.14 Captura de trama "CONNECT_REQUEST" modificada

Como puede verse en la Fig. 3.14, la trama de tipo 'CONNECT_REQUEST' modificada no lleva ningún tipo de encriptación pero no contiene información crítica, únicamente la clave publica NTRU del cliente codificada como una clase Java de tipo 'EncryptionPublicKey' [33]. Esta estructura presenta una longitud de 608 octetos pero esto no representa ningún problema para su transporte ya que un paquete UDP puede transportar hasta un máximo de 65536 octetos de datos.

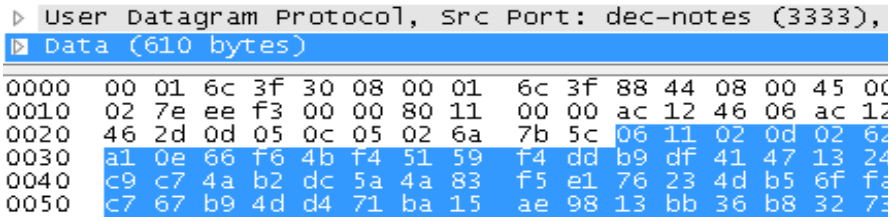


Fig. 3.15 Captura de trama "AUTHENTICATION_REQUEST" propuesta

La Fig. 3.15 muestra una captura de la trama de tipo 'AUTHENTICATION_REQUEST' y puede verse como todos los datos salvo la cabecera se encuentran encriptados mediante la clave pública NTRU del cliente y accesibles únicamente por el mismo. De esta manera nadie salvo el cliente puede tener acceso al identificador temporal del canal de comunicación asociado a dicha conexión.

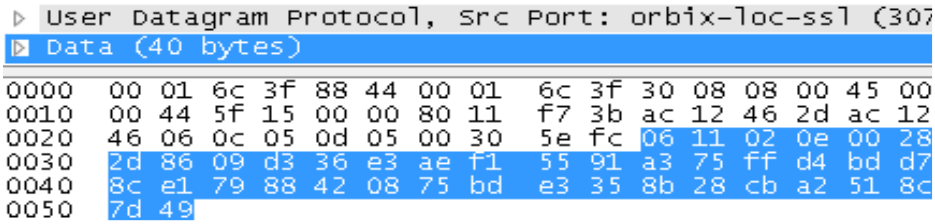


Fig. 3.16 Captura de trama "AUTHENTICATION_RESPONSE" propuesta

La Fig. 3.16 muestra una captura de la trama de tipo 'AUTHENTICATION_RESPONSE' y como todos los datos salvo la cabecera están encriptados utilizando la clave Salsa20.

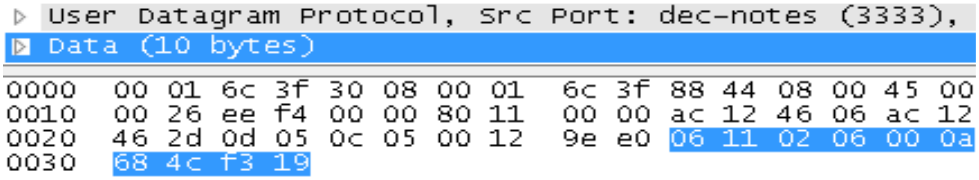


Fig. 3.17 Captura de trama “CONNECT_RESPONSE” propuesta

Esta clave es de únicamente 32 bytes de longitud y el tamaño de los datos encriptados es más pequeño si lo comparamos con NTRU. Finalmente la Fig. 3.17 muestra una captura de una trama de tipo ‘CONNECT_RESPONSE’ modificada y encriptada también con la clave Salsa20 y que será utilizada el resto de la conexión.

3.4 Conclusiones

Puede verse como la tecnología KNX presenta una gran flexibilidad y robustez y es una interesante candidata a la hora de integrarla con tecnologías inalámbricas. En este trabajo se muestra las posibilidades de acceso al bus KNX mediante USB y también utilizando como medio de transporte Internet. Ésta última opción presenta características muy interesantes pero también una serie de inconvenientes que se analizan y se tratan de subsanar mediante una propuesta de cambio en su protocolo añadiendo mecanismos para mejorar la conectividad, seguridad y autentificación de la información.

3.5 Referencias

[1] ISO/IECStd14543-3, 2000, Information technology -- Home electronic system (HES) architecture -- Part 3: Communication layers and initiation.
[2] Cenelec EN50090, Home and Building Electronic System (HBES), 2005.
[3] CEN EN Std13321-1, 2012, Open data communication in building automation,controls and building management — Home andbuilding electronic system Part 1: Product and system requirements.
[4] ANSI/ASHRAE Std 135, 2010, BACnet- A Data Communication Protocol for Building Automation and Control Networks.
[5] GB/Z Std20965, 2007, Control network HBES technology specification—Home and building control system.
[6] “KNX on USB Protocol Specification & KNX USB Interface Device Requirements”, Application Note 037/02 Rev. 4, 2003.
[7] Device Class Definition for Human Interface Devices, V 1.11,disponible en: http://www.usb.org/developers/devclass_docs/HID1_11.pdf,últimoacceso: 5 Marzo 2013.
[8] Librería Javahidapi, disponible en: http://code.google.com/p/javahidapi/, ultimo acceso: 28 December 2012.
[9] KNX Association, KNX Standard, KNXNet/IP: Overview, Chapter 3/8/1, 2009.
[10] KNX Association, KNX Standard, KNXNet/IP: Core, Chapter 3/8/2, 2009.
[11] KNX Association, KNX Standard, KNXNet/IP: Tunneling, Chapter 3/8/4, 2009.
[12] Librería Calimero, disponible en:http://sourceforge.net/p/calimero/wiki/Home/, último acceso: 28 December 2012.
[13] Cavalieri, S., “Implementing encryption and authentication in KNX using Diffie-Hellman and AES algorithms”, Industrial Electronics, IECON '09. 35th Annual Conference of IEEE, Porto, Portugal, 2009.
[14] Lechner, D., Granzer, W., Kastner, W., “Security for KNXnet/IP”, KNX Scientific Conference, Sint-Katelijne-Waver, Belgium, 2008.

[15] Cavalieri, S., Cutuli, G., “Introducing Security and Authentication in KNX”, KNX Scientific Conference, Sint-Katelijne-Waver, Belgium, 2008.
[16] Nechvatal, J., Barker, E., Bassham, L., Burr, W., Dworkin, M., Foti, J., Roback, E., “Report on the development of the Advanced Encryption Standard (AES)”, National Institute of Standards and Technology, Vol. 106, No. 3, p. 511.
[17] Schneier, B., Kelsey, J., Whiting, D., Wagner, D., Hall, C., Ferguson, N., Kohno, T., Stay, M., “The Twofish Team's Final Comments on AES Selection”, May 15, 2000, disponible en: http://csrc.nist.gov/archive/aes/round2/comments/20000515-bschneier.pdf, ultimo acceso: 25 January 2013.
[18] The eSTREAM Portfolio in 2012, January 2012, disponible en: http://www.ecrypt.eu.org/documents/D.SYM.10-v1.pdf, últimoacceso: 25 January 2013.
[19] Mukherjee, P., “An Overview of eSTREAM Ciphers”, disponible en: http://cs.au.dk/~pratyay/eSTREAM.pdf, ultimo acceso: 25 January 2013.
[20] Good, T., Benaissa, M., “Hardware Results for Selected Stream Cipher Candidates”, The SASC 2007 Workshop Record, Ruhr University Bochum, Germany, 2007.
[21] Bernstein, D.J., “Salsa20/8 and Salsa20/12”, disponible en: http://cr.yp.to/snuffle/812.pdf, ultimo acceso: 25 January 2013.
[22] Rogawski, M., “Hardware evaluation of eSTREAM Candidates: Grain, Lex, Mickey128, Salsa20 and Trivium”, The SASC 2007 Workshop Record, Ruhr University Bochum, Germany, 2007.
[23] Meiser, G., Eisenbarth, T., Lemke-Rust, K., Paar, C., Görtz, H., “Efficient implementation of eSTREAM ciphers on 8-bit AVR microcontrollers”, IEEE Third International Symposium on Industrial Embedded Systems - SIES 2008, Montpellier / La Grande Motte, France, 2008.
[24] Narasimham, C., Pradhan, J., “Performance Analysis of Public key Cryptographic Systems RSA and NTRU”, IJCSNS International Journal of Computer Science and Network Security, 7 (8) (2007).
[25] Narasimham, C., Pradhan, J., “Evaluation of Performance Characteristics of Cryptosystem Using Text Files”, Journal of Theoretical and Applied Information Technology, 4 (1) (2008).
[26] Karu, P.,Loikkanen, J., “Practical comparison of fast public-key cryptosystems”, Seminar on Network Security, Telecommunications Software and Multimedia Laboratory, Kelsinki University of Technology, Kelsinki, 2000.
[27] Hermans, J., Vercauteren, F., Preneel, B., “Speed records for NTRU”, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 5985 LNCS, pp. 73-88.
[28] D'Souza, R., “The NTRU Cryptosystem: Implementation and Comparative Analysis”, disponible en: http://teal.gmu.edu/courses/ECE543/project/reports_2001/dsouza.pdf , último acceso: 25 Enero 2013.
[29] Mote, Y., Nehete, P., Gaikwad, S., “Superior Security Data Encryption Algorithm (NTRU)”, International Journal of Engineering Sciences, 6 (2012).
[30] Librería BouncyCastleCrypto Library, disponible en: http://www.bouncycastle.org/, último acceso: 25 January 2013.
[31] Librería NTRU, disponible en: http://tbuktu.github.com/ntru/, último acceso: 25 January 2013.
[32] “no-ip”, proveedor de servicio de DDNS, disponible en: http://www.noip.com/, último acceso: 25 January 2013.
[33] “dyndns” proveedor de servicio de DDNS, disponible en: http://checkip.dyndns.org/, último acceso: 25 January 2013.
[34] “FreeHostingEU” servicio de “Web Hosting” gratuito, disponible en: http://www.freehostingeu.com/, último acceso: 25 January 2013.

[35] “Wireshark”, Herramienta de análisis de protocolos de red, disponible en: http://www.wireshark.org/, último acceso: 25 January 2013.

CAPITULO 4 – INTEGRACION FBG

4.1 Introducción

Los sensores basados en redes de difracción en fibra óptica (FBG), han comenzado a utilizarse en la monitorización de estructuras civiles debido a que el abaratamiento en los costes de los sensores y los dispositivos asociados les ha dado la capacidad de competir con sistemas tradicionales de sensores con cableado convencional.
En este proyecto se ha utilizado sensores de fibra óptica ya que esta tecnología presenta una serie de ventajas sobre medios de transmisión guiados más comunes como el par trenzado o el coaxial.

- Es inmune totalmente a las interferencias electromagnéticas.
- Es segura. Al permanecer el haz de luz confinado en el núcleo, no es posible acceder a los datos trasmitidos por métodos no destructivos.
- Es segura, ya que se puede instalar en lugares donde pueda haber sustancias peligrosas o inflamables, ya que no transmite electricidad.
- Es ligera. El peso de un carrete no es ni la décima parte de uno de cable coaxial.
- Libre de Corrosión. Son pocos los agentes que atacan al cristal de silicio.
- Baja Atenuación. La fibra óptica alcanza atenuaciones del orden de 0.15 dB/Km.

Una red de difracción Bragg[1-5] consiste en un segmento de fibra óptica en el cual se ha introducido una variación periódica del índice de refracción a lo largo de su núcleo. Su funcionamiento está basado en la reflexión Fresnel, que básicamente consiste en que cuando una onda electromagnética que se desplaza por un medio caracterizado por un índice de refracción n_1 incide sobre la interfase con otro medio que posee un índice de refracción n_2, una parte de la onda se refleja y otra porción se transmite al otro medio. Se fabrican mediante una exposición parcial de la fibra a luz ultravioleta, siguiendo un patrón establecido de intensidad. La incidencia de luz ultravioleta provoca en el núcleo de la fibra la ruptura de los enlaces atómicos del cristal de óxido de silicio dopado con germanio que lo compone, cambiando el índice de refracción. Mediante este proceso se consigue generar una modulación del índice efectivo de refracción de los modos que se transmite por la fibra óptica, a lo largo de una longitud establecida.

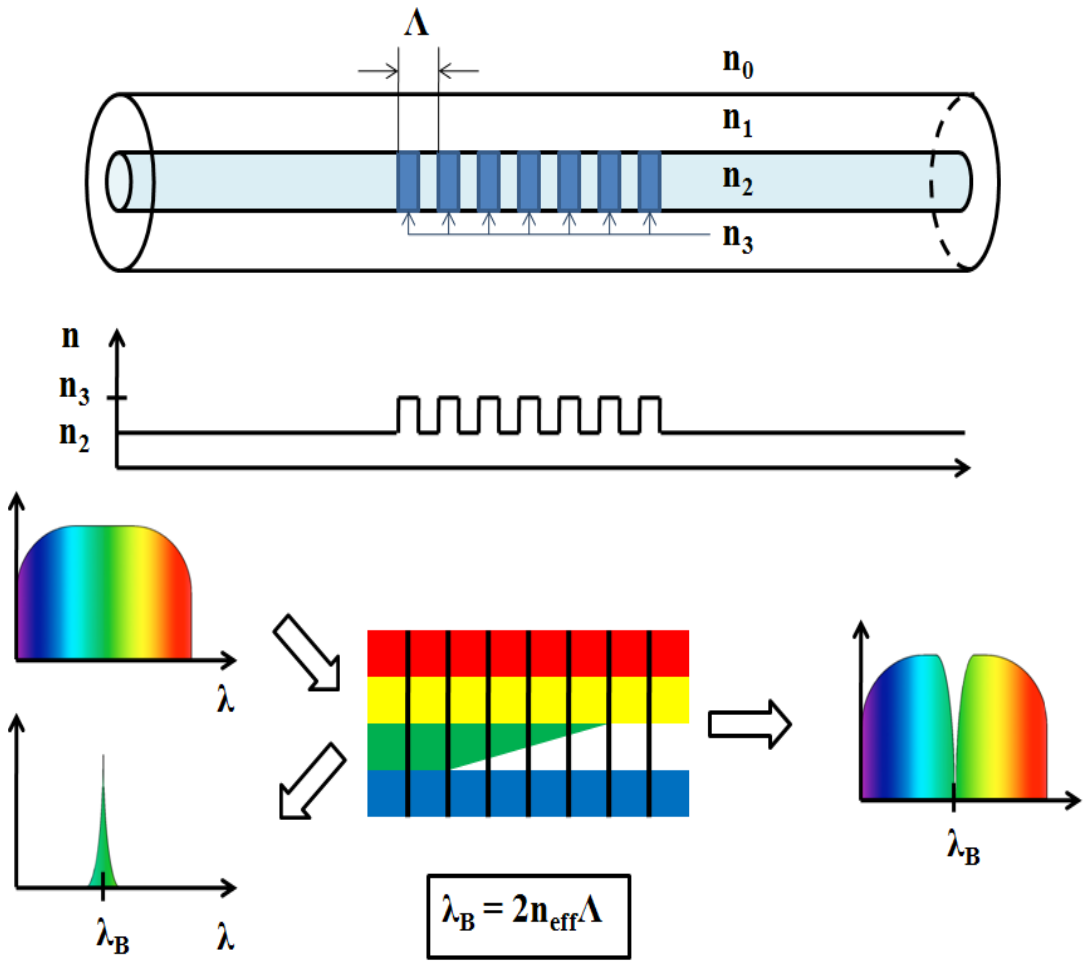


Fig. 4.1 Esquema de funcionamiento de sensor FBG

Esta variación en el índice de refracción consigue que la red de difracción refleje longitudes de onda particulares y transmita todas las demás, lo que genera su aplicabilidad dual: como un filtro óptico en línea para bloquear ciertas longitudes de onda, o como un reflector de longitudes de onda específicas.

Esta reflexión es realmente un pico espectral muy estrecho, llamado longitud de onda de Bragg (λ_B), perteneciente a la radiación incidente. Esta respuesta depende de dos factores: el período de la perturbación (Λ, red de índices de refracción) y el índice de refracción efectivo de la fibra (n_{eff}). Todo cambio en alguno o ambos parámetros, modifica el espectro reflejado. Estos cambios dependen a su vez de los llamados esfuerzos laterales (temperatura, presión). De ahí que puedan ser usados como filtro selectivo para emplearlos efectivamente como sensores puntuales de deformación o esfuerzo, temperatura y presión. En la Fig. 4.1 se puede ver el funcionamiento de una red Bragg de forma gráfica.

Una característica muy importante de los sensores FBG es que ofrece la posibilidad de multiplexar varios sensores en una misma fibra, como se puede ver en la Fig. 4.2 Mediante esta opción no es necesario tirar un cable para cada sensor sino que con únicamente una fibra sería suficiente, con el consiguiente ahorro de material y costes de instalación.

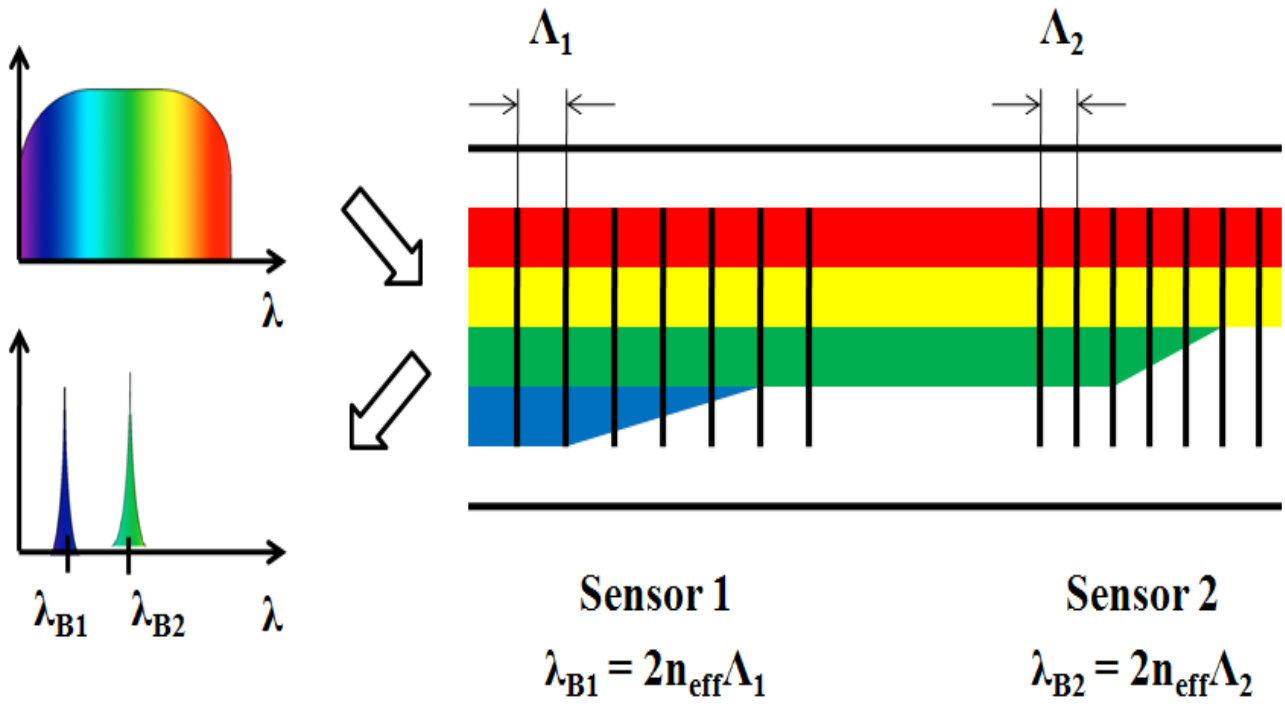


Fig. 4.2 Multiplexación de sensores FBG

4.2 Acceso a los datos en dispositivos ópticos

En la actualidad existe una falta de estandarización en los diferentes equipos de medición basados en fibra óptica y muchas veces las soluciones finales se basan en sistemas o equipos diseñados a medida. De esta manera acceder a los datos a través de dichos dispositivos no es una tarea cambiante y a menudo no muy sencilla. De esta manera es muy difícil hablar de un acceso de tipo generalizado a datos de sensores de fibra óptica y hay que hablar de soluciones particulares en equipos particulares. En este trabajo se ha analizado el acceso a tres equipos diferentes los cuales estaban basados en aplicaciones programadas en 'LabVIEW'.

4.2.1 Interrogador de fibra óptica FS 5200

Este dispositivo es un interrogador óptico del fabricante FiberSensing modelo FS5200. Posee cuatro canales ópticos con una frecuencia de muestreo de una muestra por segundo y una resolución de 1 pm. Aunque el número de sensores máximos que se pueden conectar a un mismo canal dependerá del tipo de los mismos se recomienda que nunca sea mayor a 20. Además presenta un puerto Ethernet con conector RJ45 mediante el cual es posible comunicarse con el mismo. La aplicación de control del interrogador está programada en LabVIEW y utiliza una serie de variables compartidas para acceder a los diferentes datos. Este tipo de variable es similar a el tipo 'local' o 'global' pero proporciona funcionalidades adicionales pensadas para intercambiar datos en tiempo real entre diferentes VI que se encuentren ejecutándose en la misma máquina o en maquinas diferentes dentro de una misma red. Existen diferentes tipos de variables compartidas y las utilizadas por el software propietario del interrogador son del tipo '*network-published*', pensadas para el envío de información a través de la red. Para ello se hace uso del protocolo '*NI-Publish Subscribe Protocol*' (NI-PSP), un protocolo propietario que funciona sobre TCP o UDP basado en eventos. Este tipo de variables puede ser accedido sin LabVIEW utilizando por ejemplo 'LabWindows/CVI' y

'Measurement Studio'. Además, se puede leer o escribir sobre variables compartidas desde una programa en lenguaje C mediante el uso la API 'Windows COM OPC Automation'.'
En concreto el software propietario dispone de tres variables compartidas. Una de ellas consiste en un 'array' con todos los nombres de los sensores conectados al interrogados, otra en un 'array' con las correspondientes unidades de los sensores anteriormente mencionados y finalmente tendremos una última variable compartida con los valores de dichos sensores también en forma de 'array'. La manera de acceder a estas variables es bastante sencilla, como se puede ver en el ejemplo de la Fig. 4.3:

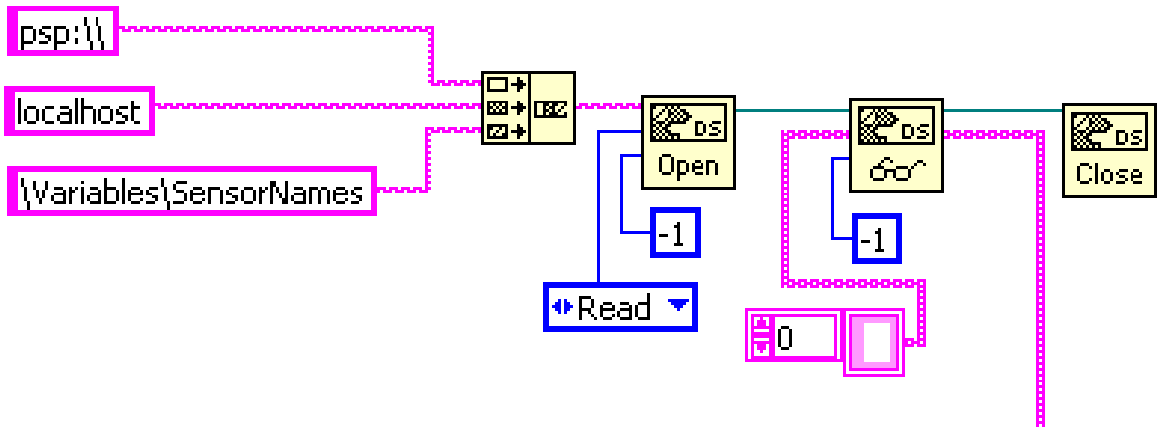


Fig. 4.3 Ejemplo de acceso a variable compartida

Los bloques que necesitaremos en este VI se encuentran en la librería 'DataSocket'. Para poder acceder a una variable compartida se debe utilizar el bloque 'Open' de dicha librería. La URL que se deberá introducir sigue el siguiente formato:

psp://computer/library/shared_variable

donde 'psp' indica el tipo de protocolo a utilizar para variables compartidas, 'computer' indica el nombre de red de la máquina en cuestión y el resto corresponde a la librería y nombre de la variable compartida a acceder.

4.2.2 Placa óptica FS 1500

Este dispositivo es una placa óptica basada en una tarjeta de captura de datos PXI DAQ modelo PXI6040 del fabricante National Instruments. Posee un único canal óptico e integra un diseño propietario basado en '*add/drop WDM*' que permite la medición de un máximo de cuatro sensores simultáneamente siempre y cuando trabajen en las bandas adecuadas de 1541.5, 1547.9, 1554.3 o 1560.8 nm respectivamente. La tasa de adquisición es parametrizable hasta un máximo de 2 KS/s con una resolución de 0.1pm para mediciones directas y de $0.5x10^{-3}$ para mediciones dinámicas en FFT.
El bus '*eXtensions for Instrumentation*' PXI es un bus industrial de comunicaciones estándar para instrumentación y control. Las siglas significan una extensión del bus PCI pensada para aplicaciones de instrumentación.
Para poder acceder a la tarjeta PXI desde el este módulo primero se debe instalar el software de servicios de medida y software para la adquisición de datos del fabricante National Instruments llamado Ni-DAQ. Como se puede apreciar en la Fig. 4.4, consiste en unos drivers que realizan la

función de pasarela entre la parte hardware (en este caso de la tarjeta PXI) con alguno de los lenguajes de programación soportados (en este caso LabVIEW).
Existen dos versiones, una es la NI-DAQmx, versión más moderna y potente pero que no soporta dispositivos de cierta antigüedad y NI-DAQ Tradicional, más antigua pero que soporta dispositivos no soportados por NI-DAQmx. La tarjeta PXI instalada en el interrogador es una PXI-6040E, no soportada por NI-DAQmx y por lo tanto se necesita recurrir a NI-DAQ Tradicional.

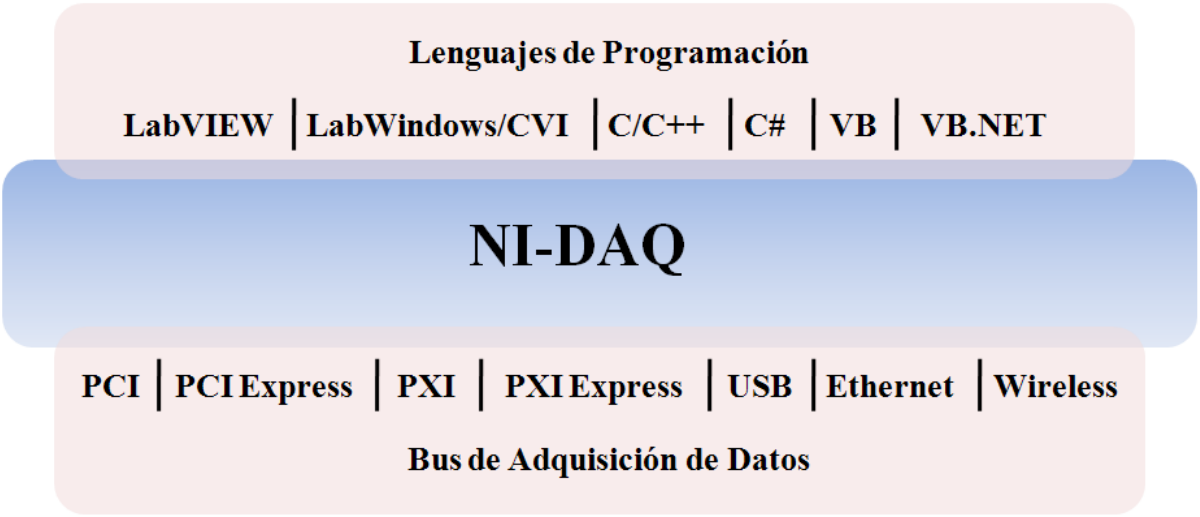


Fig. 4.4 Esquema NI-DAQ

Acto seguido y utilizando la aplicación 'Measurements&Automation' instalada junto con la herramienta NI-DAQ se selecciona el tipo de tarjeta y se configuran sus parámetros. Además es necesario incluir una serie de librerías en la carpeta de nombre 'vi.lib' dentro de la ruta de instalación de LabVIEW. Realizando todo esto ya es posible acceder a la tarjeta PXI a través de LabVIEW.
El último paso consiste en configurar dicha tarjeta para su uso como capturadora óptica. Para ello el fabricante FiberSensing pone a disposición de una librería LabVIEW que contiene un VI de ejemplo de uso y un fichero de calibración del dispositivo.
El primer paso consiste definir las diferentes ganancias para cada banda. Dichos valores deben ser introducidos en forma de 'array' como se muestra en la Fig. 4.5.a ya sea como control o como constante. Como se muestra en la Fig. 4.5.b, estos datos se introducen directamente en el modulo correspondiente suministrado por el fabricante llamado 'FS1500_SetGain' que configura la tarjeta en función de los parámetros de entrada.

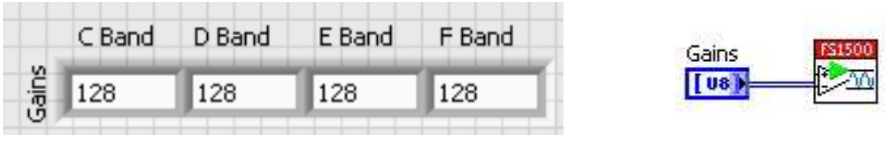


Fig. 4.5 Configuración de las ganancias y los voltajes de la tarjeta

Las figuras 4.6 y 4.7 muestran una captura de pantalla del diagrama de bloques del VI ejemplo suministrado por el fabricante. En la Fig. 4.6 puede verse como el primer paso consiste en la lectura del fichero de configuración. Acto seguido se selecciona y configura el dispositivo NI-DAQ a

utilizar, el canal de dicho dispositivo y el número de muestras por segundo. Después se inicia la adquisición y se entra en un bucle que va leyendo los datos.

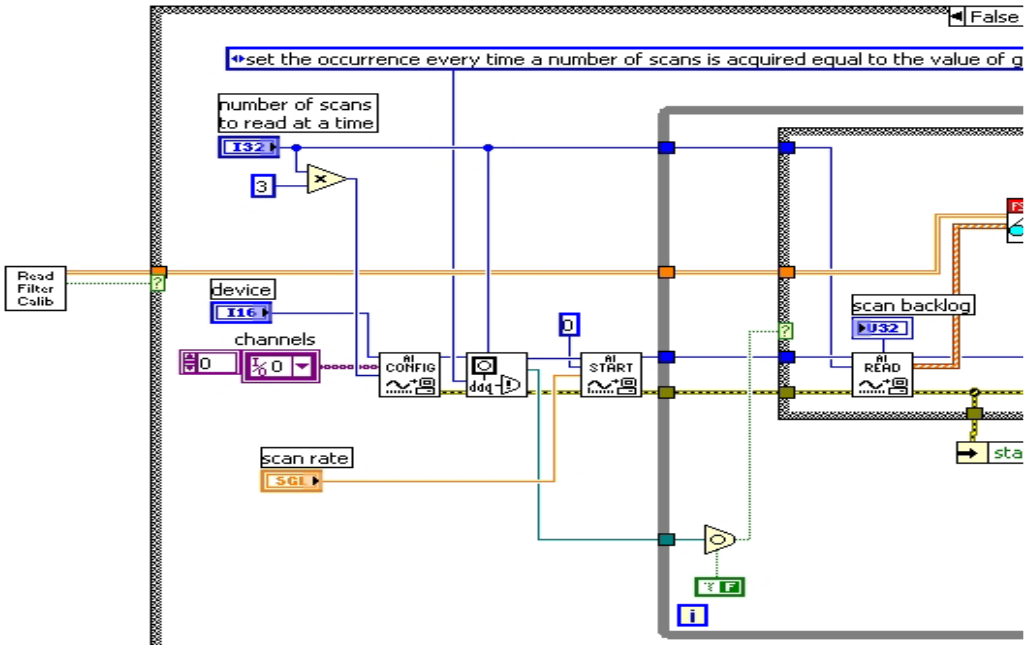


Fig. 4.6 Diagrama VI ejemplo FS 1500

En la Fig. 4.7 puede verse como los datos leídos se introducen en el modulo 'Read_Basic' suministrado por el fabricante y que devuelve la señal óptica adquirida en forma de voltajes y como longitudes de onda.

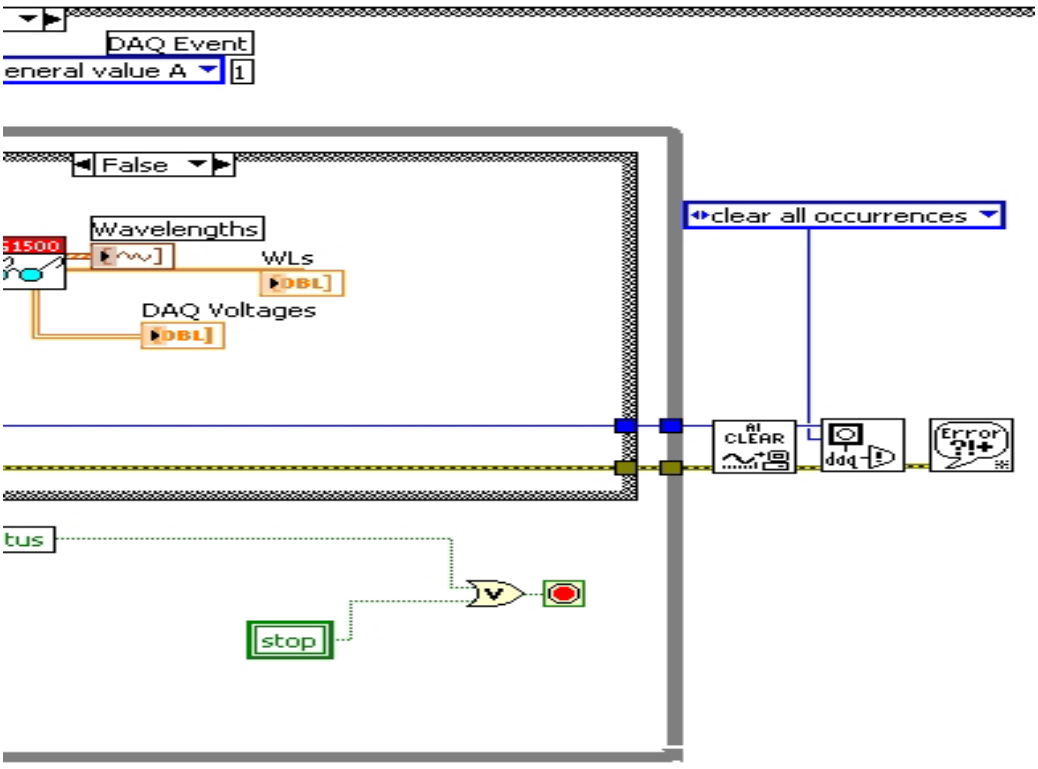


Fig. 4.7 Diagrama VI ejemplo FS 1500

4.2.3 Interrogador de fibra óptica SM 125

Este dispositivo es un interrogador óptico del fabricante MicronOptics modelo SM 125. Presenta cuatro canales ópticos, una frecuencia de muestreo de 5 Hz y un rango de medida de longitud de onda de 1510-1590 nm con una resolución de 1 pm. Además presenta un puerto Ethernet con conector RJ45 mediante el cual es posible comunicarse con el mismo.
En este caso el lenguaje de programación de la aplicación de control es también LabVIEW y el fabricante suministra el código de una aplicación de ejemplo.
Como puede verse en la Fig. 4.8, la forma de comunicación con el interrogador consiste en abrir una conexión TCP con el puerto 50000 y la dirección IP del interrogador, en concreto 10.0.0.122. Cabe destacar que al ser esta una dirección de tipo privado el computador al que se conecte debe pertenecer a la misma red para que se puedan comunicar mutuamente con lo que generalmente será necesario configurar la dirección IP del mismo manualmente. Una vez que la conexión está abierta es posible acceder a los datos de los sensores conectados al mismo. La Fig. 4.8 muestra el diagrama del VI que realiza la lectura de los sensores.

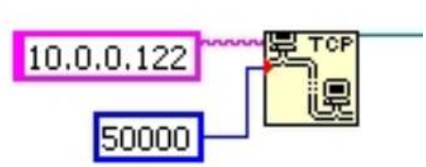


Fig. 4.8 Configuración de las ganancias y los voltajes de la tarjeta

Para realizar la lectura en este trabajo se procedió a la implementación del siguiente VI cuyo diagrama se muestra en las figuras 4.9 y 4.10.
Una vez de que la conexión TCP está abierta, hay que enviar una cadena de texto de longitud 10 octetos con contenido exacto ‘get_data’ y esperar a que el interrogador envía los datos.
Como puede verse en el diagrama, el fabricante proporciona un bloque llamado ‘ReadHeader’ que se encarga de procesar la información recibida. Si observamos dicho bloque más detenidamente podemos ver que dicha información tiene una cabecera de 40 octetos que procesa en dos tandas.
En la primera tanda procesa los primeros 20 octetos divididos en 5 campos de formato numérico entero de cuatro octetos cada uno. Para el cálculo de valor de cada campo se suma el valor como numero entero de los dos octetos menos significativos y se le suma el valor como entero de los dos octetos más significativos multiplicados por 65536. Dichos campos contienen la información del tamaño de la cabecera, la longitud de onda mínima, el incremento en longitud de onda, el número de puntos de datos que contiene y el DUT.
Cabe destacar además que al valor de los campos de longitud de onda mínima y al del incremento de longitud de onda el valor final es dividido por 10000.
En la segunda tanda el VI procesa los 20 octetos siguientes divididos también en cinco campos diferentes de 4 octetos cada uno. Dichos campos contienen entre otros datos, la información del tamaño de la cabecera, la versión utilizada y el número de canales.
El resto de la información quitando las cabeceras contiene los diferentes los valores de potencia de los diferentes puntos que componen el espectro óptico de la señal. Por lo tanto, se inicia con el valor de potencia del primer punto situado en la longitud de onda mínima obtenida en la cabecera anterior y se sigue con cada uno de los valores de los puntos siguientes incrementando en longitud de onda con el valor obtenido de la cabecera anterior. Además puede verse como el valor de potencia de cada punto es representado por un valor en coma flotante de dos octetos de longitud.

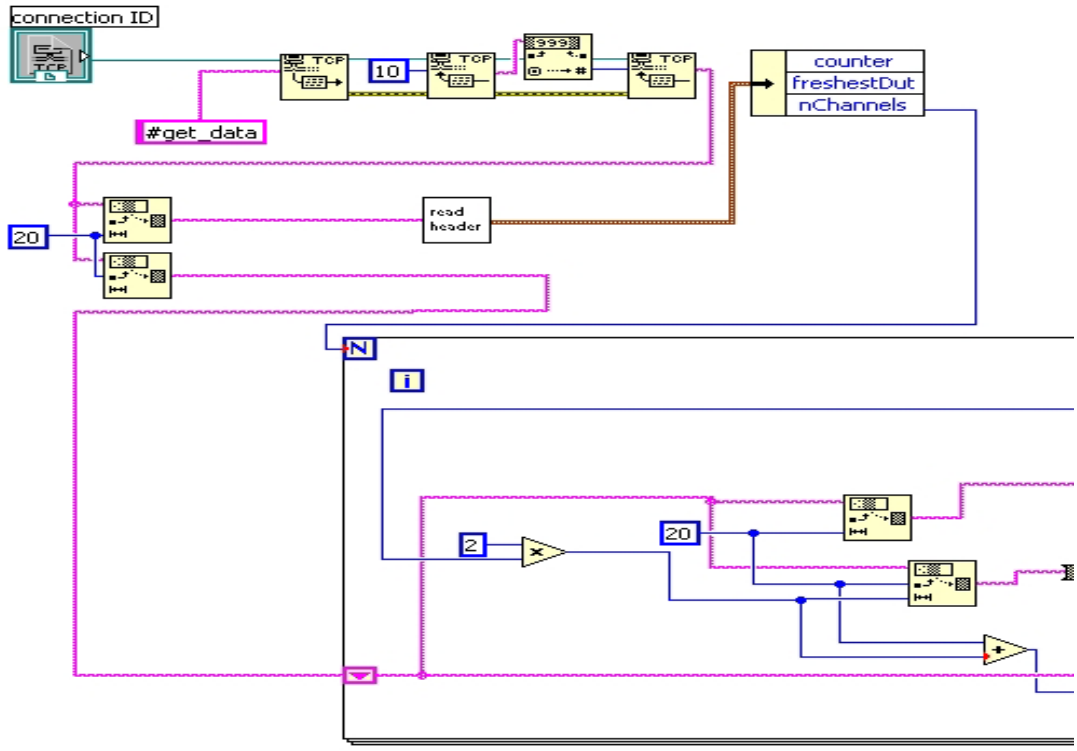


Fig. 4.9 Configuración de las ganancias y los voltajes de la tarjeta

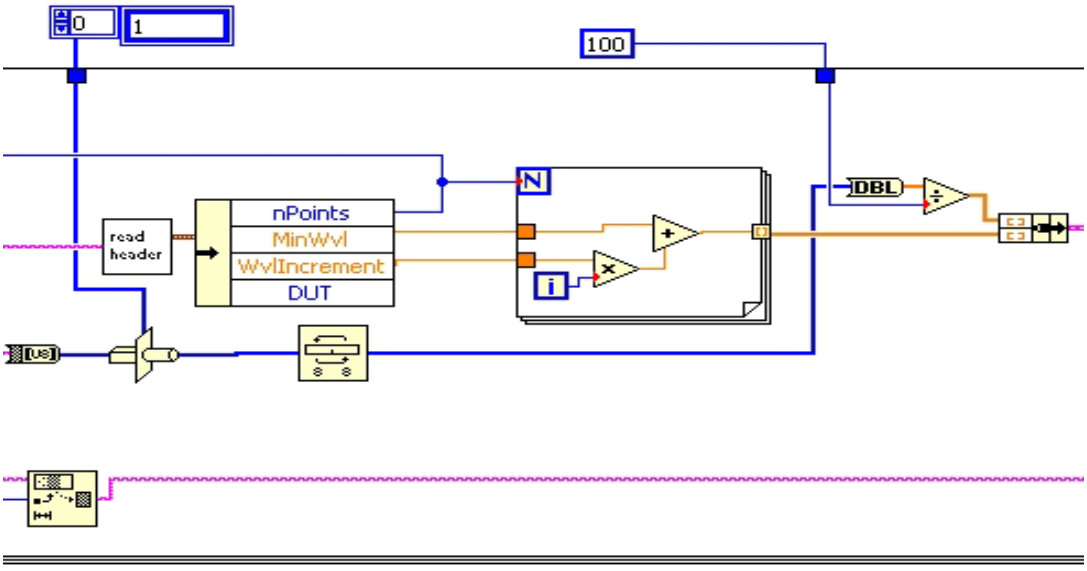


Fig. 4.10 Configuración de las ganancias y los voltajes de la tarjeta

En este trabajo se procedió a la implementación de una aplicación para este dispositivo que permite su uso como interrogador, como '*Optical Spectrum Analyzer*' OSA y además posibilitaba el registro y almacenado automatizado de los datos de los sensores conectados al mismo. Además permite compensar valores de sensores de deformación con otro de temperatura.

Para el funcionamiento como OSA únicamente es necesario mostrar en pantalla los datos leídos con el método anteriormente mencionado. En la Fig. 4.11 puede verse una captura de la aplicación funcionando como OSA y con dos sensores conectados. Mediante la lista desplegable de canales disponibles situada en la parte inferior izquierda de la pantalla puede seleccionarse la visualización individual de un canal en concreto o de todos a la vez. Finalmente es necesario comentar que la

aplicación posibilita la realización de un 'Zoom' seleccionando mediante una ventana configurable la zona de pantalla de interés.
Para el funcionamiento como interrogador el proceso es más complicado. El primer paso para obtener los datos de los sensores es realizar una detección de picos en los datos obtenidos del espectro óptico. Para evitar falsos positivos el VI utilizado para tal efecto, cuando detecta un posible pico, comprueba el valor de un numero parametrizable de puntos espectrales consecutivos para asegurarse de que efectivamente son menores y el posible pico no es una fluctuación. Es necesario comentar que para la correcta detección de los sensores conectados al interrogador es vital un configurar cada canal óptico con un valor adecuado de este parámetro.

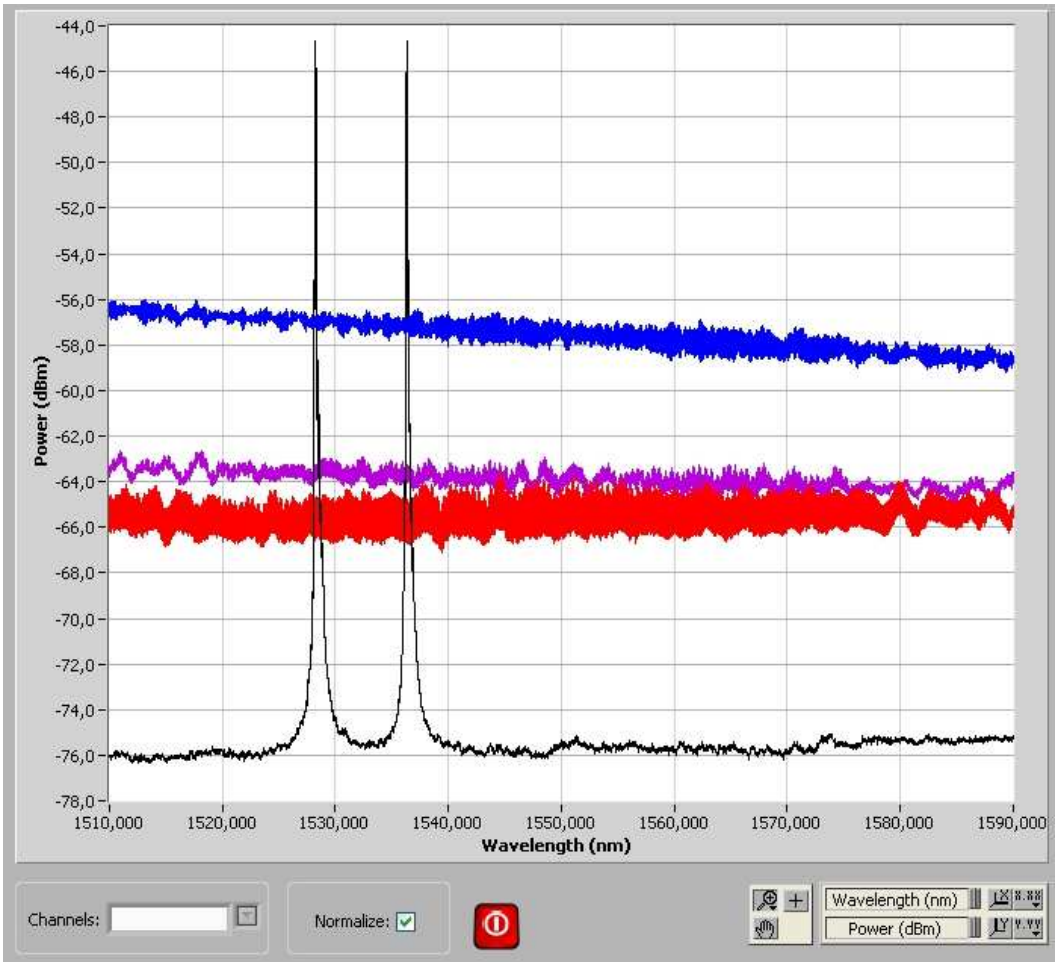
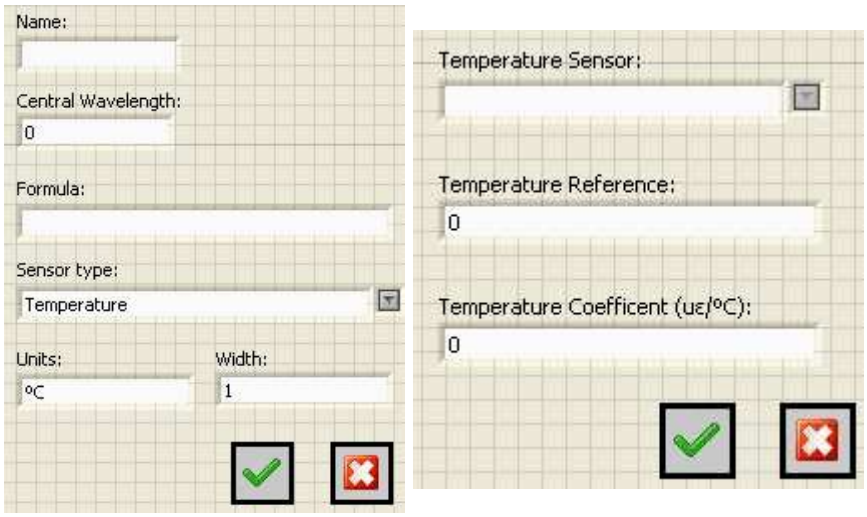


Fig. 4.11 Funcionamiento de la aplicación en modo OSA

Una vez los sensores son detectados es necesario configurarlos. Seleccionando el sensor correspondiente aparece el dialogo cuya captura se muestra en la Fig. 4.12.a, donde se edita el nombre del sensor, su longitud de onda central, el polinomio utilizado para transformar la longitud de onda medida a valor real de medición, el tipo de sensor y las unidades. Si el tipo de sensor seleccionado es de tipo deformación, entonces aparece un dialogo para seleccionar y configurar el sensor de temperatura asociado para la realización de la compensación (ver Fig. 4.12.b).



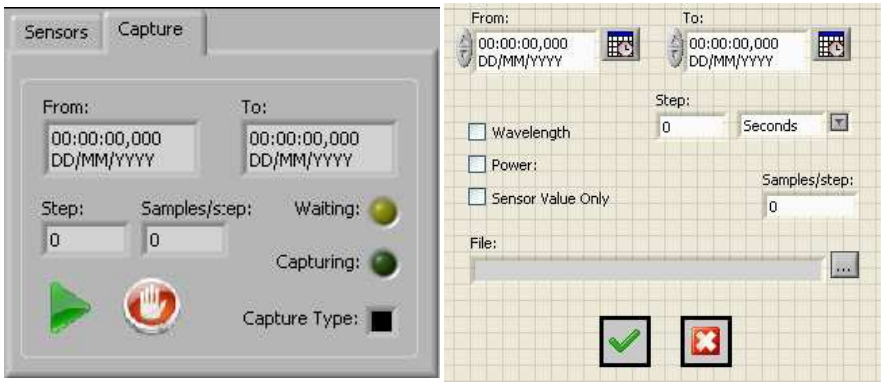
(a) (b)

Fig. 4.12 Capturas de diálogos para configuración de los sensores

Como se puede ver en la Fig. 4.13 los sensores detectados se muestran en forma de tabla. Cada fila corresponde a un sensor conectado y detectado por la aplicación y los valores de cada columna corresponden a su nombre, longitud de onda, valor de potencia óptica y valor final en unidades reales detectados en cada momento.

Fig. 4.13 Captura muestra información medida de los sensores

En cuanto a la automatización del almacenado de los datos, el proceso se monitoriza y configura en el panel de la pestaña de captura del interface principal de la aplicación. Como se puede ver en la Fig. 4.14.a, en dicho panel se puede configurar y ver el estado de la captura en curso. En el mismo aparecen datos como el instante de tiempo de inicio de la captura, el instante final, el campo de paso que indica el tiempo entre punto de almacenado y el numero de muestras tomadas por punto. El indicador luminoso amarillo se activa cuando existe una automatización en curso pero todavía no se ha iniciado al almacenado de datos y el indicador verde se activa cuando existe una automatización en curso y se está almacenando datos. El botón con forma de mano detiene cualquier automatización en curso y el botón verde en forma de triangulo configura y crea una nueva automatización. En el caso de pulsarse, aparece el dialogo de configuración mostrado en la Fig. 4.14.b. Como puede observarse en la misma puede elegirse la información relativa a los sensores que se desea almacenar así como el nombre y la ruta del fichero de texto que contendrá dichos valores.



(a) (b)

Fig. 4.14 Capturas de diálogos para configuración de los sensores

4.3 Conclusiones

Puede verse que el uso de la fibra óptica como medio de transmisión y como tecnología de sensado presenta una serie de ventajas importantes frente a otras tecnologías. Quizás uno de los mayores problemas de su uso, además de su elevado precio, es la falta de estandarización. Por lo tanto intentar encontrar una manera generalizada de acceso a los datos para diferentes equipos de diferentes fabricantes es una tarea compleja y normalmente la solución final es válida únicamente para un equipo en concreto. Como puede verse muchos equipos de este tipo utilizan LabVIEW como lenguaje de programación y aunque no es uno de los más eficientes es quizás uno de los más fáciles de programar para personas ajenas al mundo de la programación clásica. En este trabajo se muestran métodos de acceso a los datos de varios dispositivos ópticos reales concretos basados en variables compartidas de LabVIEW y envío de información a través de IP.

4.4 Referencias

[1] Hongo, A., Kojima, S., Komatsuzaki, S., "Applications of fiber Bragg grating sensors and high-speed interrogation techniques", Structural Control and Health Monitoring, Vol. 12, no. 3, 269-282, 2005.

[2] Kersey, D., Davis, M.A., Patrick, H.J., LeBlane, M., Koo, K.P., Askins, C.G., Putnam, M.A., Friebele, E.J., "Fiber grating sensors," J. Lightwave. Technology., vol. 15, no. 8, pp. 1442–1463, Aug. 1997

[3] Liu, S., Yu, Y., Zhang, J., Chen, X., "Real-Time monitoring sensor system for Fiber Bragg Grating array", IEEE Photonics Technology letters, Vol. 19, no.19, 1493-1495, Oct. 2007.

[4] Lopez-Higuera, J.M., "Handbook of Optical Fiber Sensing Technology", John Wiley & Sons, 2002.

[5] Hill, K.O., Meltz, G., "Fiber Bragg Grating Technology Fundamentals and Overview", Journal of Lightwave Technology, Vol. 15, no. 8, 1263-1276, 1997.

CAPITULO 5 – APLICACIÓN PRÁCTICA

5.1 Introducción

Durante este trabajo se participó en diferentes proyectos de investigación y desarrollo y en una colaboración con la universidad de Massey en Nueva Zelanda con el resultado de varias implementaciones reales de sistemas basados en integración y/o monitorización de tecnologías de comunicación inalámbricas.

5.2 Proyecto Tracasa

En este trabajo se realizó el desarrollo y la instalación de un sistema constructivo de monitorización de fachadas aplicado al sector de la arquitectura. En concreto, se pretende monitorizar el comportamiento del cerramiento realizado a la torre de comunicaciones del edificio de Tracasa, compuesto por unos novedosos paneles de *composite*. Se busca conocer el comportamiento dichos paneles en tiempo real y bajo condiciones de servicio, así como prever la ocurrencia de alteraciones severas por causa de agentes externos (variaciones de temperatura, cargas de viento, vibraciones, entre otros).Dado que como se ha mencionado anteriormente la torre se utilizará para comunicaciones, resulta especialmente indicada la utilización de sensores de fibra óptica, ya que estos sensores no afectan a las señales recibidas por las antenas, ni las emisiones de las antenas alteran las medidas de los sensores.
El sistema desarrollado consta de la integración de una red de sensores de fibra óptica para monitorizar la temperatura y deformación de los paneles del cerramiento y una red paralela de sensores domóticos basados en KNX.
La instalación de los sensores ópticos se realizó sobre la parte tanto exterior como interior de varios de los paneles que componen el cerramiento. Los sensores domóticos se situaron dentro de una estación meteorológica situada en la azotea del edificio mediante la cual se pretendía tanto comprobar el correcto funcionamiento de los sensores de temperatura ópticos como analizar el comportamiento del material según diferentes condiciones de viento y/o lluvia. Debido a que no existía conexión a Internet por cable en el interior de la torre fue necesario dotarla de una conexión 3G. Debido a que los dispositivos de conectividad a Internet 3G USB no proporcionan una conexión estable ininterrumpida, fue necesaria la instalación de un router 3G.

5.2.1 Red óptica

Los sensores utilizados en la red óptica para monitorizar el comportamiento de los paneles del cerramiento fueron20 de temperatura y 10 de deformación conectados a un interrogador de fibra óptica modelo FS 5200 de Fibersensing (ver apartado 4.2.1). Hay que tener en cuenta que los valores registrados por los sensores de deformación basados en FBG son dependientes de la temperatura de trabajo con lo que si se necesita más precisión es necesaria la instalación conjunta de un sensor de temperatura para en función de la misma corregir el valor final de deformación obtenido. Los sensores de deformación utilizados fueron los modelos FS 6200 de Fibersensing y los modelos OS 3100 y OS3200 del MicronOptics mientras que los sensores de temperatura utilizados fueron los modelos FS 6300 de Fibersensing. Ese necesario comentar que los sensores de FiberSensing fabricados con poliamida son fáciles y rápidos de instalar y se colocaron en el interior de la torre mientras que los sensores de MicronOptics son más resistentes y debido a las condiciones meteorológicas se colocaron en el exterior de la misma.
Por otro lado el sistema tiene integrado un acelerómetro de fibra óptica para monitorizar las vibraciones de la torre, más concretamente un modelo GS6500 de Gavea Sensors. Debido a que en

este trabajo se pretenden registrar vibraciones con componentes frecuenciales de hasta 400 Hz, de acuerdo con el teorema de muestreo es necesario una la adquisición de al menos 800 muestras por segundo. Ya que la frecuencia de muestreo proporcionada por el FS 5200 es insuficiente, el acelerómetro óptico se conectó a una tarjeta FS 1500 (ver apartado 4.2.2).
La distribución de los sensores de fibra se realizó en cuatro ramas cada una de las cuales se conectó a un canal óptico diferente del interrogador. Hay que tener en cuenta que el número máximo de sensores conectado a un canal óptico está limitado por las pérdidas de de inserción en los conectores, perdidas por el doblado de la fibra, etc.
Además, a la hora de elegir sensores que se van a conectar a un mismo canal óptico es necesario que no se interfieran espectralmente. Hay que tener en cuenta que el margen espectral necesario para los sensores de temperatura, según las especificaciones y para un rango de temperatura de 100ºC, es de ±1.5nm (100ºC x 10pm / ºC + 0.5nm). Para el caso de los sensores de deformación y para un rango de funcionamiento de 500 με, el margen espectral es de ±1.1 nm. (500 με x 1.2pm / με + 500 pm). La Fig. 5.1 muestra la distribución y emplazamiento de varios de estos sensores.

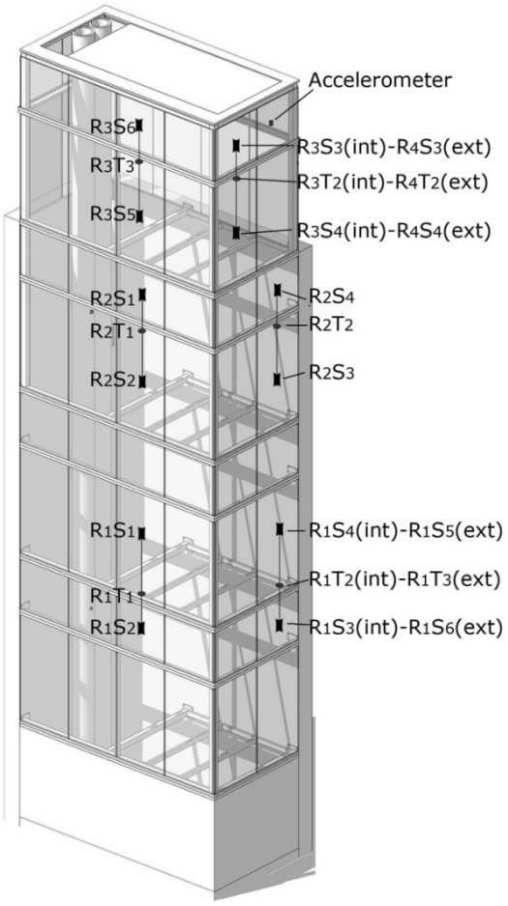


Fig. 5.1 Distribución de sensores de fibra óptica

5.2.2 Red KNX

La red KNX instalada constaba de una única estación meteorología que se colocó en la azotea de la torre y que daba información acerca de las condiciones meteorológicas que podían afectar al cerramiento y de un interface KNXNet/IP. La estación proporcionaba medidas en tiempo real de la velocidad del viento, de la temperatura en el exterior, de diferentes medidas de luminosidad y la presencia de lluvia. El modelo utilizado fue el 2224 HW del fabricante Jung.

El interface proporcionaba la posibilidad de interactuar sobre dispositivos KNX a través de una conexión basada en IP (ver apartado 3.3). El modelo utilizado fue el N148/21 del fabricante Siemens.
El rango de temperatura del sensor de temperatura era de -22ºC a 50 ºC. El sensor de viento medía la velocidad del mismo en un rango de entre 0 y 40 m/s con una precisión de ±2 m/s aunque no proporcionaba su dirección. De manera similar, la estación detectaba la presencia de lluvia pero no la intensidad de la misma. Finalmente, los tres sensores de luminosidad separados 90º entre sí tenían un rango de funcionamiento de entre 1 y 110 KLx mientras que el rango del sensor de luminosidad crepuscular era de entre 0 y 60 KLx.

5.2.3 Descripción del sistema

Desde el punto de vista de hardware, la composición de los elementos que lo componían y su interconexión puede verse en la Fig. 5.2.

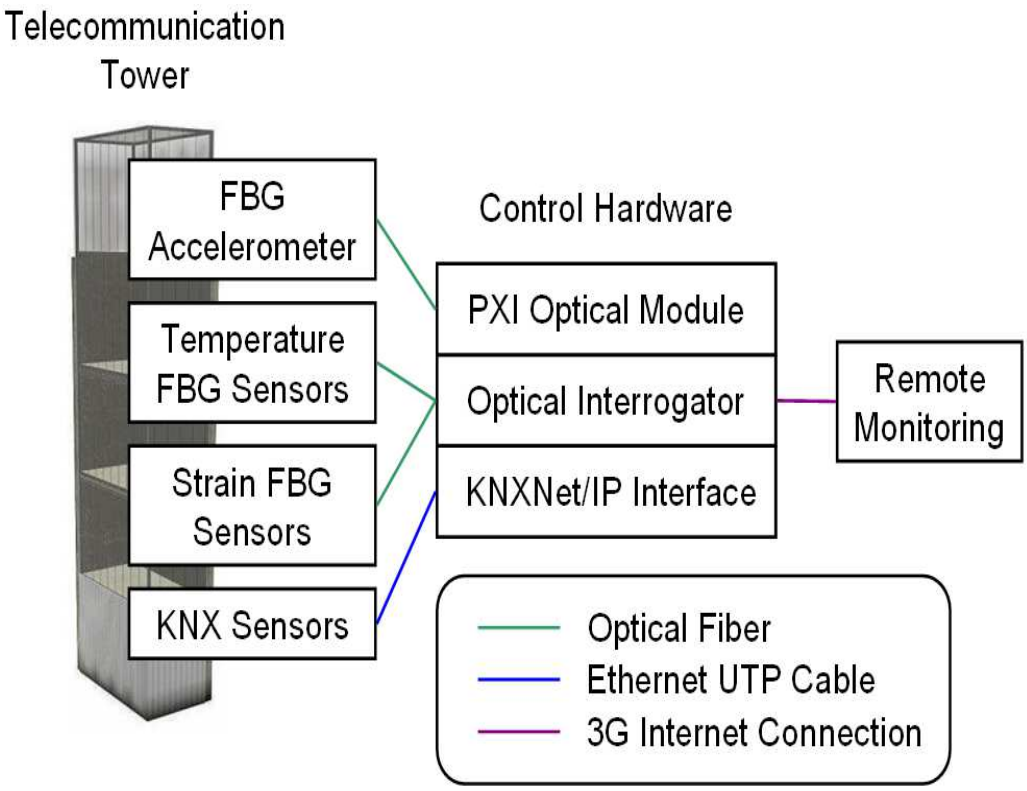


Fig. 5.2 Esquema hardware del sistema

En dicha figura puede verse como el acelerómetro óptico estaba conectado por fibra óptica a la tarjeta óptica FS 1500, los sensores de temperatura y deformación estaban conectados por fibra óptica al interrogador de fibra óptica y la estación meteorología KNX estaba conectada al bus mediante cable TP así como el interface KNX. Finalmente la conectividad a Internet era proporcionada por un *router* 3G.
Desde el punto de vista de software, el sistema constaba de cinco módulos independientes, intercomunicados utilizando TCP o UDP en función de la naturaleza de los datos a intercambiar. En la Fig. 5.3 puede verse un esquema software del sistema.

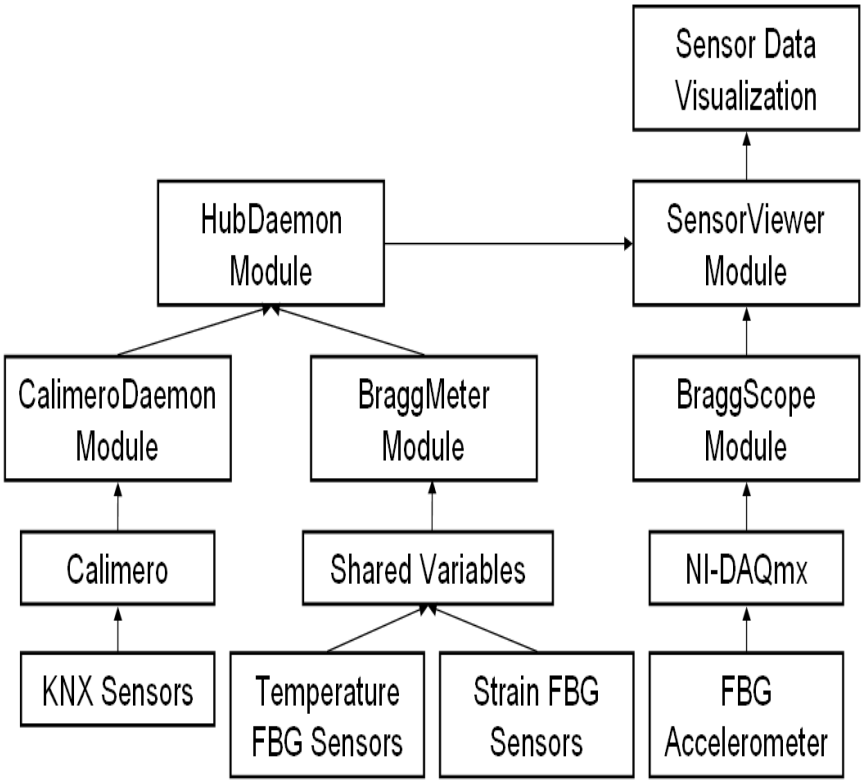


Fig. 5.3 Esquema software del sistema

En dicha figura puede verse como el módulo CalimeroDaemon era el encargado de obtener los datos de los sensores KNX a través del interface KNXNet/IP utilizando la librería Calimero. Puede verse también como el módulo BraggMeter se encargaba de obtener los datos de los sensores de temperatura y deformación conectados al interrogador de fibra óptica accediendo a las variables compartidas de la aplicación propietaria programada en LabVIEW del mismo. En cuanto al acelerador de fibra óptica, puede apreciarse como a través del software NI-DAQmx el módulo BraggScope adquiría los datos de dicho sensor. Aunque no puede apreciarse en la figura cabe mencionar también que este módulo almacenaba dichos datos en una base de datos MySQL.
Para poder almacenar valores en la base de datos mediante el VI es necesario tener instalado en LabVIEW la librería LabSQL [1], que da la funcionalidad de interactuar con la base de datos. Además será necesario instalar los drivers de Windows de ODBC (si no están ya instalados) y configurar un nuevo drivers ODBC. Para ello se debe ir a '*panel de control->ODBC Data Sources->System DSN->add->mysql OBDC 3.51 driver*'. Después se debe configurar el DSN creado y en la pestaña de 'login' rellenar los parámetros de la conexión.
El bloque principal que se utilizara será el indicado en la Fig. 5.4:

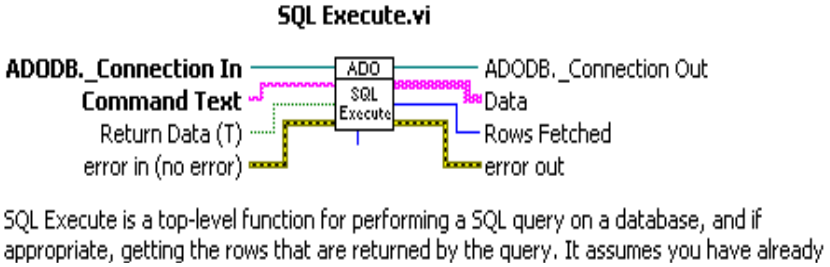


Fig. 5.4 Bloque para ejecutar sentencia SQL

Para la utilización de he dicho bloque se debe introducir una cadena de texto con la sentencia SQL que se quiera ejecutar y dará como salida los datos procesados.
La señal que se obtenía del acelerómetro cada segundo consta de 8192 bytes, valor que interesaba reducir para mejorar la eficiencia de la red y sobre todo teniendo en cuenta que para la transmisión de datos en tiempo real conviene minimizar en la manera de lo posible el tamaño de los mismos para minimizar el retardo
Entonces, para reducir la cantidad de datos a enviar se comprimía en origen y se descomprimía en destino. De esta forma se pagaba la reducción de la cantidad de datos intercambiados con tiempo de cómputo de compresión/descomprensión en ambos extremos.
Para realizar la compresión en el VI, será necesario instalar en LabVIEW las rutinas para la compresión de datos [2]. Como se puede ver en la Fig. 5.5, el bloque a utilizar es muy sencillo. Únicamente hay que introducirle los datos a comprimir y el nivel de compresión. Hay que tener en cuenta que junto con el ejecutable que obtenemos al compilar el módulo se debe colocar la librería 'zlib.dll' para que funcione correctamente.

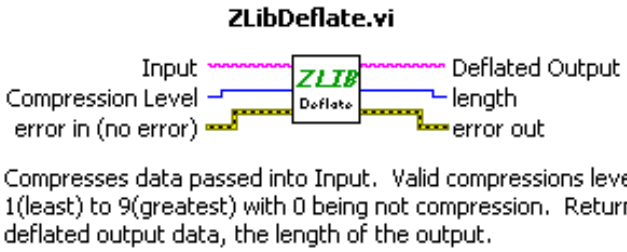


Fig. 5.5 Bloque para comprimir datos utilizando Zlib

Mediante esta compresión se pasa de enviar 8192 bytes de datos a unos 200 o 300 bytes, dependiendo del instante de captura
El módulo HubDaemon tenía una función de concentrador de los datos de los sensores ópticos y KNX y además era el encargado de almacenarlos en una base de datos MySQL. Finalmente el módulo SensorViewer era el encargado de pedir remotamente los datos de todos los sensores a los módulos HubDaemon y BraggScope para su posterior visualización.
Debido a que se realizo la programación de los módulos para que se intercomunicasen utilizando el protocolo IP este hecho dotaba al sistema de gran flexibilidad ya que dichos módulos podían funcionar en maquinas diferentes trabajando en diferentes emplazamientos siempre y cuando dispusiesen de acceso a Internet.
Las diferentes funcionalidades del SensorViewer están separadas en cinco pestañas: exterior, tiempo real, acelerómetro, base de datos, y configuración.
En la Fig. 5.6 se puede ver una captura de pantalla de la pestaña "exterior". La pantalla de visualización está dividida en dos zonas con distintas vistas en 3D de la torre aunque debido al tamaño de la misma únicamente se muestra una porción de la misma. En ellas se pueden ver tanto los valores y unidades de todos los sensores como el punto físico en el que se encuentran instalados. Los valores de los sensores se actualizan cada segundo y su color de fondo es azul para los sensores ópticos externos, magenta para los internos y amarillos para los sensores de la red KNX. La configuración gráfica de los sensores se realiza mediante un fichero de configuración, en la que cada una de las líneas de texto representa los parámetros de cada uno de los sensores.

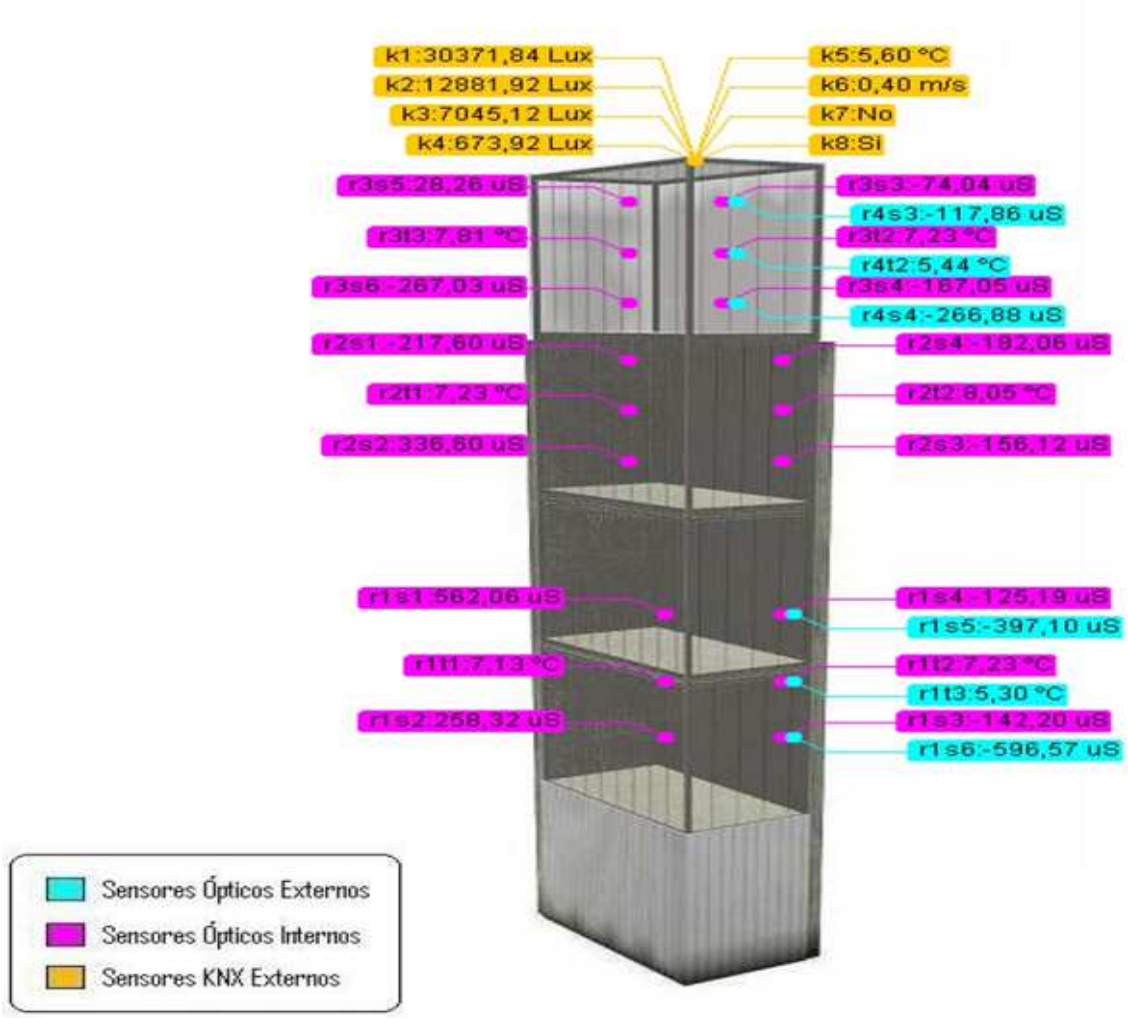


Fig. 5.6 Esquema software del sistema

Para la representación grafica de los datos de los sensores se ha utilizado en algunos casos la librería JFreeChart [3], escrita 100% en java y de uso gratuito. Posee una gran variedad de gráficos de distintos tipos y además existen gran variedad de ejemplos de cómo utilizarlos. Soporta varios formatos de salida, incluyendo componente Swing, fichero de imagen (incluyendo PNG y JPEG), y formatos de archivos de vectores gráficos (incluyendo PDS, EPS y SVG). Finalmente JFreeChart es de código abierto y de libre uso. Se distribuye bajo los términos de 'GNU Lesser General PublicLicense' (LGPL) y nos permite utilizarla en software propietario.

La Fig. 5.7 muestra una captura de pantalla de la pestaña 'base de datos'.

En esta pestaña se procede a la visualización de los datos de los sensores almacenados en la base de datos. Debido al tamaño de la misma, dicha captura únicamente muestra una porción de la pantalla original. En la parte superior izquierda, en el recuadro "Seleccionar intervalo tiempo" se debe introducir mediante los botones "desde" y "hasta" el comienzo y el fin del intervalo de tiempo en el que se quiera observar los valores de los sensores.

Una vez seleccionados ambos, será necesario pulsar el icono con forma de "v" amarilla para realizar la consulta a la base de datos. A continuación, en el recuadro contiguo de "seleccionar datos" se puede seleccionar mediante listas desplegables tanto el tipo de sensor como el tipo de tabla a acceder. Mediante esta última se puede acceder a la tabla de datos "normal" (la que tiene almacenados los datos adquiridos mediante una captura programada) y la tabla de datos "vibración" (la que tiene almacenados los datos adquiridos cuando el acelerómetro detecta una vibración por

encima del umbral). Una vez seleccionados tanto el tipo de sensor como la tabla, los sensores disponibles que se ajusten a esta selección aparecerán en el recuadro de la derecha “Sensores disponibles” en forma de botones. El reborde del sensor de cada botón será de color rojo cuando el sensor se encuentre seleccionado y negro en caso contrario, pasando de un estado al otro con la pulsación del mismo. En la parte central de la pestaña se tiene una grafica en la que se pueden ver los valores de los sensores seleccionados, cada uno de ellos con un color diferente y representado en la leyenda de la grafica. Adicionalmente, en el recuadro “estadísticas” se dispone de una lista desplegable mediante la cual se selecciona el sensor y otra con el tipo de estadística a calcular, pudiéndose elegir entre el máximo valor registrado dentro del intervalo, el mínimo, la media y la varianza. Finalmente mediante el icono en forma de disquete del recuadro “guardar datos” se accede al dialogo mediante el cual se puede guardar los datos de un intervalo de tiempo concreto en un archivo de texto.

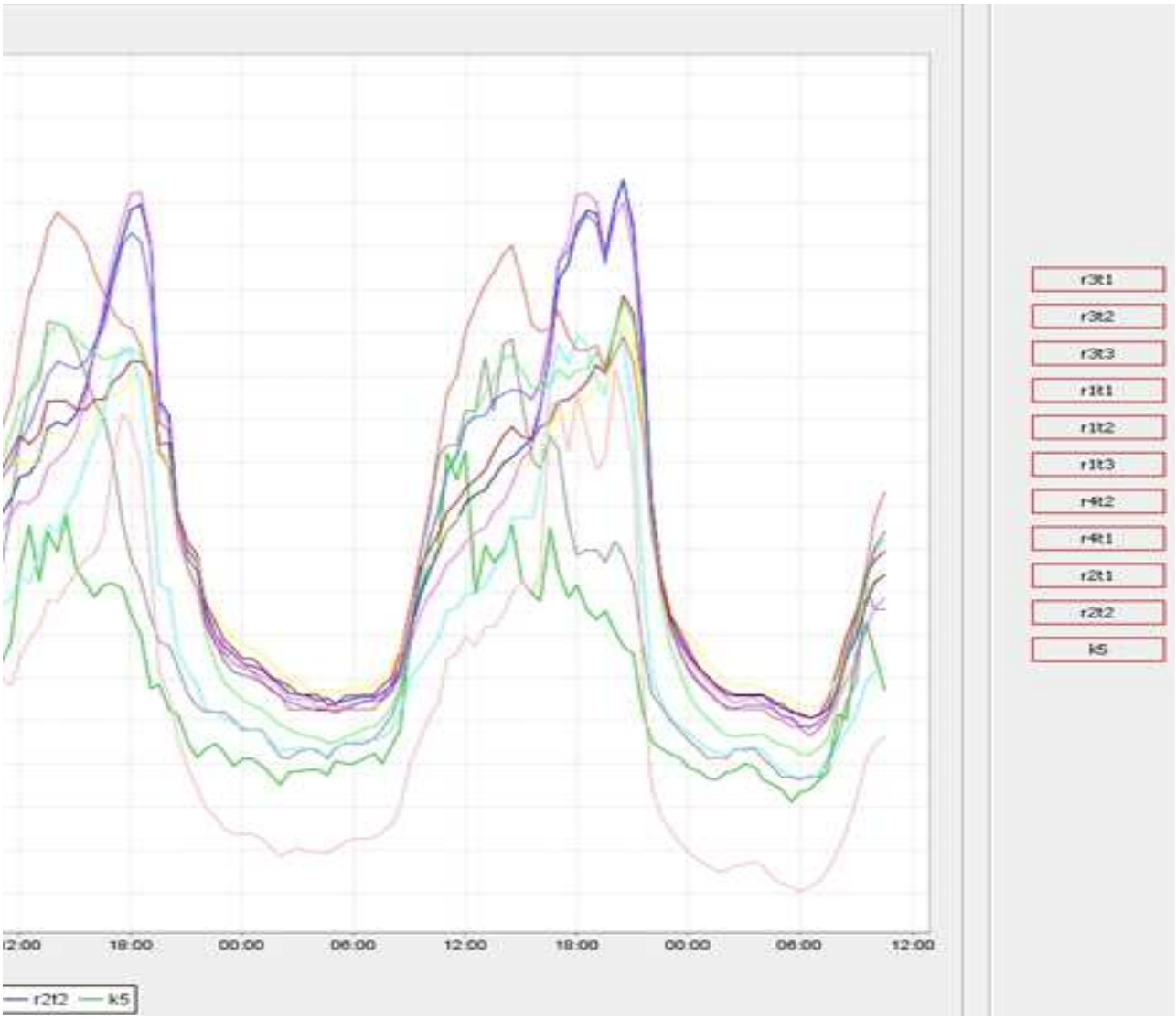


Fig. 5.7 Esquema software del sistema

5.3 Ekihouse

En este trabajo se formó parte del equipo que la Universidad del País Vasco llevó en su participación en la competición Solar Decathlon Europe 2012 de casas autosuficientes impulsadas

con energía solar. Más concretamente se procedió al desarrollo e instalación de una versión modificada del sistema comentado en el apartado 5.2 y una aplicación programada en Android para acceder a dispositivos KNX a través de Wi-Fi e Internet. El la Fig. 5.8b puede verse una fotografía de la vivienda llamada 'Ekihouse' que tomó parte en el concurso. En la parte superior de la Fig. 5.8 puede verse el armario con los diferentes módulos KNX de la instalación. Debajo del mismo se aprecia el interrogador de fibra óptica FS 5200 utilizado en el proyecto debajo del cual se encuentra el *router* inalámbrico que proporciona conectividad Wi-fi al sistema.

Fig. 5.8 (a) Vivienda Ekihouse (b) Armario y equipos de la instalación

5.3.1 Sistema integrado de monitorización

El sistema instalado consistió en la integración de sensores de fibra óptica, KNX y IEEE 802.15.4 para la monitorización remota en tiempo real y el posterior almacenado de los valores registrados por los mismos. La Fig. 5.9 muestra un esquema hardware del sistema.

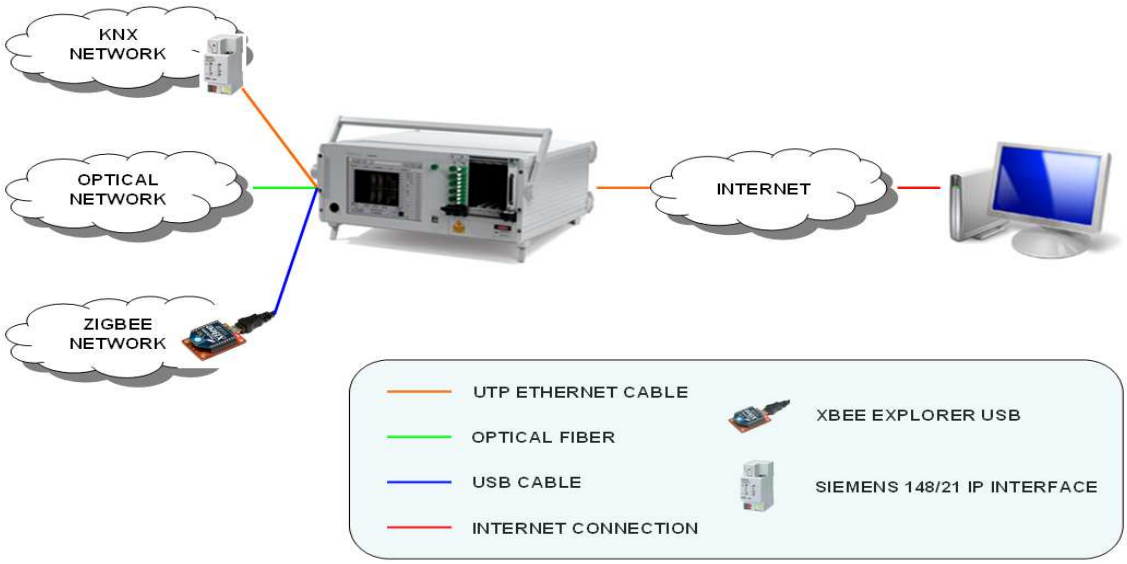


Fig. 5.9 Esquema hardware del sistema

La red KNX estaba formada por un sensor de temperatura modelo WS 10 T y otro de luminosidad modelo WS 10 D conectados a un modulo de cuatro entradas analógicas modelo 2214 REG A, todos ellos del fabricante Jung. Además, dicha red poseía un modulo de ocho actuadores modelo SA/S 8.10.1'y un interface KNXNet/IP modelo N148/21, ambos del fabricante Siemens. Los sensores ópticos conectados al interrogador fueron dos sensores de temperatura FS 6300 de Fibersensing.
Los dispositivos IEEE 802.15.4 utilizados fueron XBee Pro 'Series 1'. Para poder conectar el XBee receptor de los datos que envían los diferentes sensores XBee al interrogador se utilizó un XBee Explorer. Dicho dispositivo realiza la función de puente entre el puerto RS232 del XBee y un puerto USB.
En la Fig. 5.10 puede verse el esquema software del sistema.

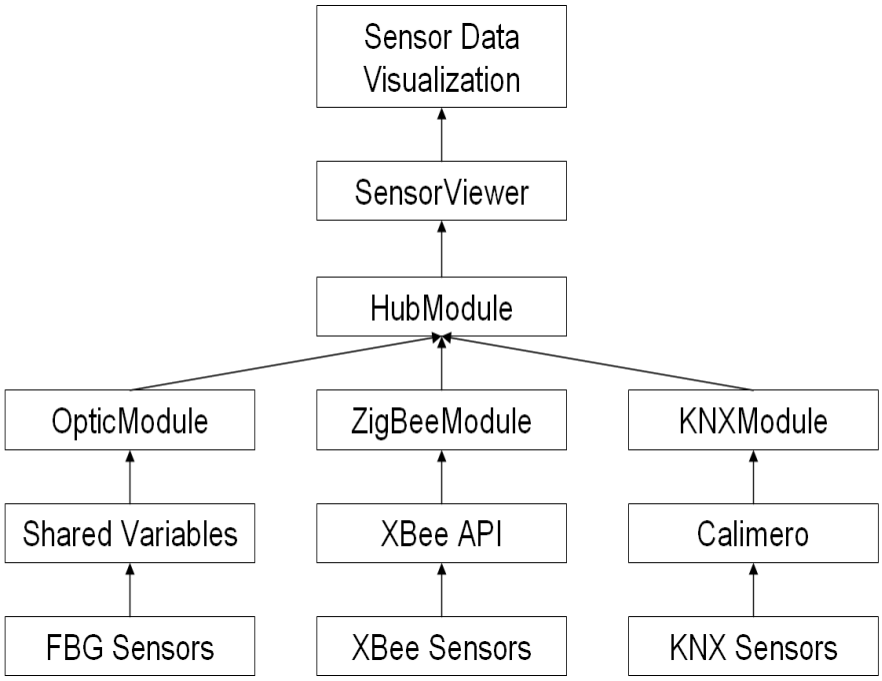


Fig. 5.10 Esquema software del sistema

Puede verse como el módulo OpticModule era el encargado de acceder a los datos de los sensores de fibra óptica conectados al interrogador a través de las variables compartidas de la aplicación propietaria que corría en el mismo.
El módulo ZigBeeModule obtenía los datos de los sensores XBee a través del puerto USB asociado al XBee Explorer donde se encontraba conectado el XBee receptor. Para acceder a los datos dicho módulo utilizaba la librería xbee-api [4]. El proceso seguido se muestra en la Fig. 5.11.
Primero se crea una estructura de tipo XBee y mediante el método 'Open' se realiza la conexión serie con el dispositivo. Después se entra en un bucle en el que se leen los datos recibidos de los sensores XBee en forma de estructura 'XBeeResponse' mediante el método 'GetResponse()'. Una vez realizado esto se procede a convertir la estructura de tipo 'XBeeResponse' a tipo 'RxResponseIoSample'. Finalmente, ejecutando el método 'getSamples()[0].getAnalog()' de dicha estructura se accede al valor final de los datos del sensor. El módulo KNXModule era el encargado de obtener los datos de los sensores KNX a través del interface KNXNet/IP utilizando la librería Calimero. Puede verse también como el módulo HubModule era el encargado de recibir los valores de todos los sensores a través de los módulos correspondientes para ponerlos a disposición del

módulo SensorViewer para la visualización remota de los mismos. Aunque no se aprecia en la figura, el módulo HubModule también se encargaba del almacenado de los valores de los sensores.

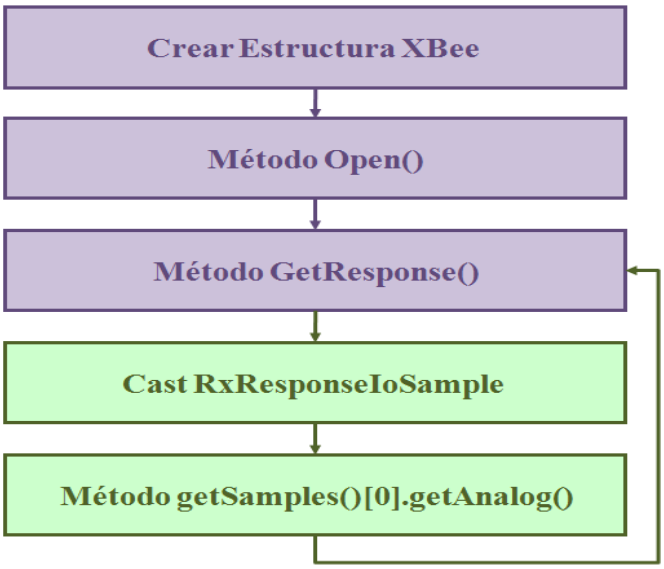


Fig. 5.11 Esquema software del sistema

5.3.2Aplicación Android para el control de dispositivos KNX

El sistema Android es básicamente un kernel Linux sobre el cual se han implementado diferentes capas con una serie de librerías y *Frameworks* para su uso principalmente en dispositivos embebidos como *SmartPhones* y tablet PCs. La aplicación de control desarrollada utiliza la librería JKNXNetIP comentada en el apartado 3.3.3. Como se ha comentado anteriormente, esta librería es una implementación mínima de la parte cliente del protocolo KNXNet/IP y su funcionamiento es muy similar a la de la librería Calimero. La Fig. 5.12 muestra el esquema de uso de la misma.

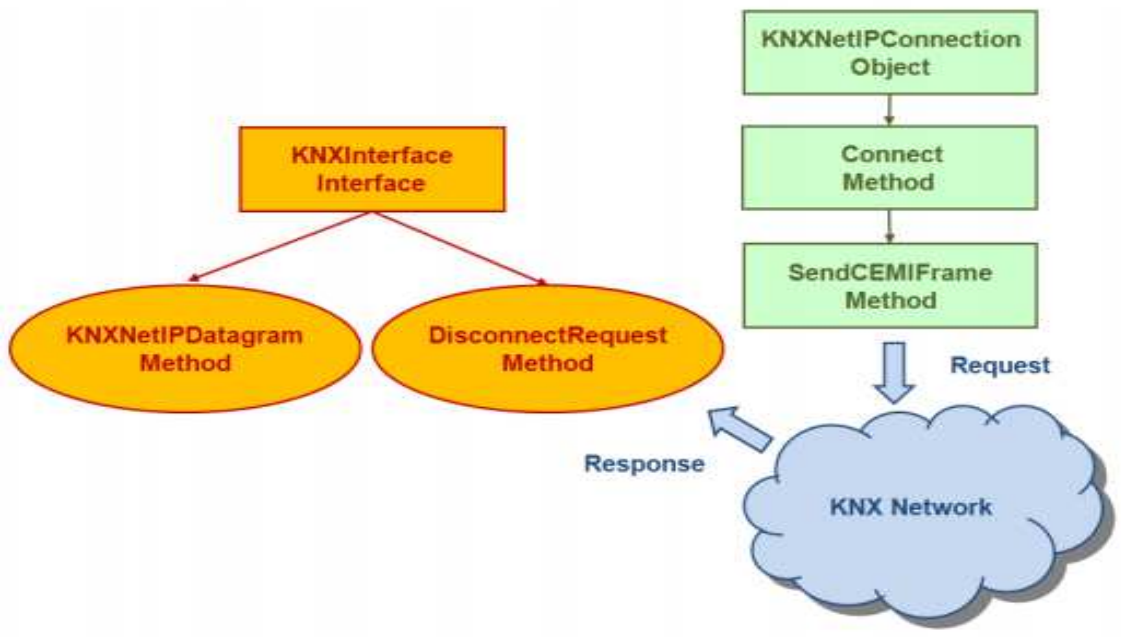


Fig. 5.12 Esquema de uso librería JKNXNetIP

El primer paso en el uso de la aplicación consiste en definir la conexión IP a utilizar. Como puede verse en la Fig. 5.13.a, cada conexión debe de tener un nombre que la identifique y la dirección IP asociada a la misma. Si dicha conexión utiliza NAT, es necesario indicarlo chequeando la casilla correspondiente. Una vez la conexión está configurada, hay que añadir cada una de las habitaciones que compone el edificio y en cada una de ellas introducir y configurar los elementos KNX que contienen. En la Fig. 5.13.b puede verse la pantalla de configuración de dichos elementos. Primero es necesario dar al elemento un nombre, además de la dirección de grupo dentro de la red KNX a la que pertenece. Finalmente, mediante la primera lista desplegable es necesario seleccionar que tipo de elemento es y mediante la segunda lista desplegable, seleccionar si el elemento es un sensor o un actuador. Se probó la aplicación accediendo mediante Wi-fi a los valores del sensor de temperatura y el de luminosidad KNX satisfactoriamente, así como apagando/encendiendo una lámpara que se colocó en un actuador del módulo 'SA/S 8.10.1' para tal uso.

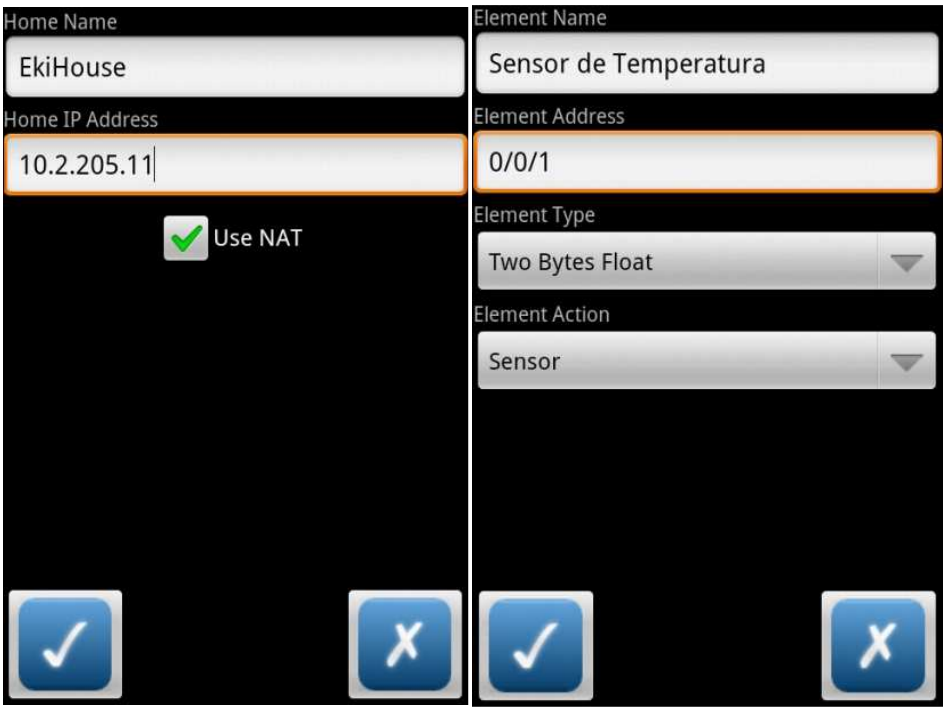


Fig. 5.13 Capturas de pantalla aplicación Android de control KNX

5.4. Monitorización y almacenado de datos XBee sobre IPv6

En este trabajo y en colaboración con la Universidad Massey de Nueva Zelanda se procedió al diseño de un sistema de monitorización remota y posterior almacenado de datos de sensores inalámbricos basados en módulos XBee 'Series 2' a través de datagramas UDP sobre paquetes IP versión 6. En dicha Universidad se había desarrollado una aplicación ejecutada sobre un *router* inalámbrico modelo WRT54G del fabricante Linksys a el que se conectaba el coordinador XBee de la red de sensores inalámbricos. Los datos eran leídos del mismo, la información del valor de los sensores desencapsulada y posteriormente codificada en forma de datagrama UDP sobre IP versión 6. En este proceso se procedió a desarrollar una aplicación Java capaz de desencapsular los datos presentes en dichos datagramas y almacenarlos en una base de datos basada en MySQL. Finalmente se procedió a la implementación de un script programado en PHP para la visualización web de los datos almacenados en dicha base de datos. Los datos enviados por los sensores XBee se recibían en el coordinador en una trama XBee de tipo 'ZigBee IO Data SampleRxIndicator' [5] y la estructura de los datos presentes en dicha trama se mantenía en el datagrama UDP final. Por lo tanto, a la hora

de obtener los datos reales de uno de estos datagramas era necesario obtener y desencapsular dicha estructura (ver Fig. 5.14).

Parámetro	Offset	Ejemplo	Descripción
Opciones de Recepción	14	0x01	0x01: Paquete confirmado 0x02: Paquete de tipo Broadcast
Número de Muestras	15	0x01	Número de muestras presente en el paquete (siempre 1)
Mascara Canal Digital	16	0x00	Mascara de bits indicando que entradas digitales están activas
	17	0x1C	
Mascara Canal Analógico	18	0x02	Mascara de bits indicando que entradas analógicas están activas
Muestras Digitales (si hay)	19	0x00	Datos entradas digitales
	20	0x14	
Muestras Analógicas	21	0x02	Cada muestra analógica consta de dos bytes que forman el valor de 10 bits devuelto por el ADC
	22	0x25	

Fig. 5.14Estructura de los datos presentes en una trama de tipo ‘ZigBee IO Data SampleRxIndicator’’

Hay que tener en cuenta que en el caso de que el sensor conectado al módulo XBee correspondiente sea de tipo analógico, el valor contenido en la estructura no es el valor final medido sino que es el valor de 10 bits devuelto por el conversor analógico-digital. Para obtener el valor final habrá que multiplicar dicho valor por un factor de conversión adecuado.

Una vez de que se tenían los datos de los sensores estos se almacenaban en una base de datos MySQL. Para ello se crearon dos tablas diferentes.

La primera, llamada ‘DeviceInfo’, contenía el PAN y el identificador de 64 bits del módulo XBee al que estaba conectado cada sensor. Además contenía también el canal al que estaba conectado, así como el nombre, unidades y factor de conversión correspondiente.

La segunda tabla, llamada ‘DeviceData’, contenía los datos de los sensores. Cada entrada correspondía a un nuevo valor y estaba compuesta del identificador de 64 bits correspondiente, el canal utilizado, el instante de llegada y el valor medido.

El tiempo de llegada se representaba como el número de milisegundos transcurridos desde el 1 de Enero de 1970 00:00:00.000 GMT (calendario gregoriano). Al almacenar el instante de tiempo de llegada de los datos en forma de número, las peticiones MySQL realizadas a la base de datos en función del mismo son más rápidas.

Para la visualización de los datos almacenados el script PHP implementado utilizaba la librería programada en Javascript para ‘Flot’ [6] y por lo tanto podía ser utilizado en cualquier navegador web con Javascript activo. Para ello se seleccionaba el sensor a visualizar de entre todos los disponibles mediante una lista desplegable, se introducía el intervalo de tiempo de interés y se pulsaba un botón para realizar la consulta de datos correspondiente a la base de datos MySQL y sacarlos por pantalla. Como puede verse en la captura de pantalla de la Fig. 5.15, los datos reales finales se representan en función del tiempo y del intervalo seleccionado del mismo. Cabe mencionar además que el script permitía la realización de un ‘Zoom’ seleccionando mediante una ventana configurable la zona de pantalla de interés.

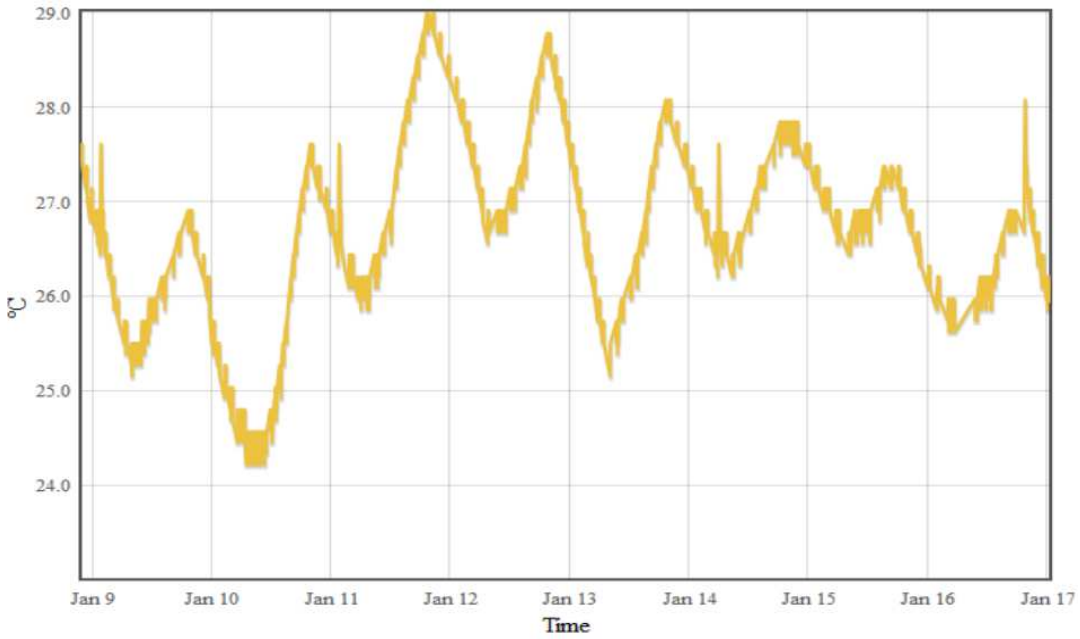


Fig. 5.15 Captura de pantalla ejecución script de visualización de base de datos

Finalmente es necesario comentar la importancia de la optimización de los datos finales a almacenar. Hay que tener en cuenta que cuantos más se almacenen no solo se necesitará más espacio sino que las consultas a la base de datos MySQL serán más lentas.
Para ello será necesario implementar un algoritmo que sea capaz de, en función de los datos reales y en tiempo real, almacenar únicamente los datos más relevantes de la señal original de manera que tanto la señal original como la almacenada presenten un grado de similitud suficiente y parametrizable.
Existen multitud de diferentes métodos, basados tanto en promedios como en estadísticas, para determinar el grado de similitud entre dos señales. Debido a las necesidades de tiempo real, los métodos a utilizar deberán cumplir que su cálculo sea incremental y además lo más rápido posible.
A continuación se procede a la descripción del algoritmo utilizado.
El primer valor de la señal original se asigna como valor de 'crunch' y también se almacena como el primer valor de señal procesada. Además, se inicializa el mecanismo de cálculo de similitud.
A continuación, cada vez que llegue un nuevo valor de la señal original, se calcula el grado de similitud con la señal de valor constante definida por el valor de 'crunch'. Entonces se compara ese valor de similitud con un umbral parametrizable y predefinido. Si el grado de similitud se considera suficiente, no se realiza ninguna acción. Si en caso contrario dicho valor excede el umbral, el dato se almacena, el valor de 'crunch' se actualiza con el mismo y se inicializa el mecanismo de cálculo de similitud.
Como mecanismo de cálculo del grado de similitud, primero se pensó en el uso de la diferencia entre el valor medio de la señal original. El costo computacional del cálculo del mismo es mínimo y los resultados obtenidos, que mostraremos más adelante, fueron satisfactorios.
También se pensó el uso del coeficiente de correlación y cuyo cálculo se realiza según la ecuación (5.1).

$$r_{xy} = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{\sqrt{n \sum x_i^2 - (\sum x_i)^2}\sqrt{n \sum y_i^2 - (\sum y_i)^2}} \tag{5.1}$$

El problema es que el cálculo de dicho factor no está definido en el caso que alguna de las dos señales presente una varianza de valor cero, como es nuestro caso.

También se pensó en el uso se pensó el uso del ratio de correlación, cuyo cálculo se realiza según la ecuación (5.2).

$$\eta^2 = \frac{\sum_x n_x (\overline{y}_x - \overline{y})^2}{\sum_{x,i}(y_{xi} - \overline{y})^2} \tag{5.2}$$

donde n_x indica el número de elementos del grupo x, $\overline{y}_x$ representa la media del grupo x, y_{xi} el elemento i-esimo del grupo x e $\overline{y}$ representa la media de las medias de los diferentes grupos.
El problema que se encontró en su uso fue que presenta un cambio demasiado brusco como para poderlo utilizar en este algoritmo.
Finalmente se pensó en el uso de un test estadístico tipo F-test basado en Anova, cuyo F-ratio para nuestro caso en concreto se calcula utilizando la ecuación 5.3:

$$F = \frac{MS_B}{MS_W} \tag{5.3}$$

$$MS_B = \sum_x n_x (\overline{y}_x - \overline{y})^2 \tag{5.4}$$

$$S_W = \sum_{x,i} (y_{xi} - \overline{y})^2 \tag{5.5}$$

$$f_W = n_x - 1 \tag{5.6}$$

$$MS_W = \frac{S_W}{f_W} \tag{5.7}$$

Hay que tener en cuenta que conforme se analiza mayor número de elementos, el cálculo se ralentiza, además que la capacidad de memoria necesaria es mayor.
Si se pretende realizar el cálculo en tiempo real y con un consumo de recursos aceptable es necesario buscar algún cambio en la ecuación. En nuestro caso, se procedió a la realización del cálculo de Sw de una manera aproximada e incremental, sumando en cada iteración la suma de cuadrados con respecto a la media total en dicha iteración.
Para probar el algoritmo se procedió a la adquisición de la señal de sendos sensores de temperatura y humedad que se colocaron en un despacho, enviando datos durante varios días cada 30 segundos. La señal del sensor de temperatura presenta un valor medio de 25.43 ºC y una varianza de 3.04 mientras que la señal del sensor de humedad presenta un valor medio de 43.75 % RH y una varianza de 6.65.
A continuación en la Fig 5.16 se muestra el número de puntos de la señal procesada en función del factor de correlación de la misma para la señal del sensor de temperatura. Puede verse que aunque los resultados son bastante similares, el comportamiento del método de grado de similitud basado en Anova es algo más estable que el basado en la diferencia con respecto a la media.

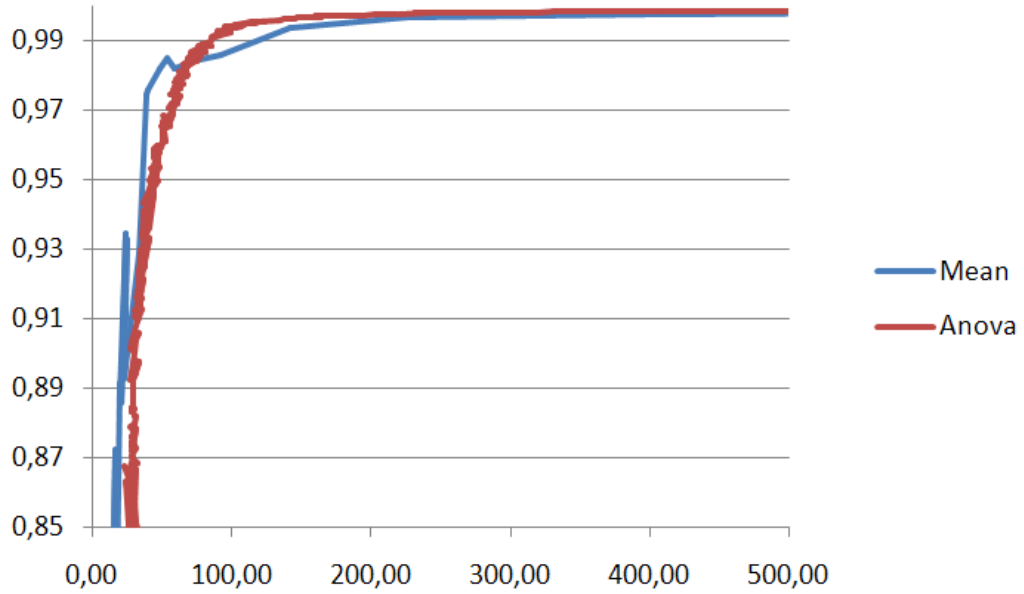


Fig. 5.16 Representación Numero de puntos vs factor de correlación sensor de temperatura

En la Fig 5.17 se muestra una representación similar a la anterior pero esta vez para la señal del sensor de humedad.
Puede verse como aquí la diferencia entre la estabilidad del método Anova y el de la media es mayor en el caso anterior debido principalmente en que la varianza de la señal del sensor de humedad también es mayor que en el caso de la señal del sensor de temperatura.
Por lo tanto, y en conclusión, podría decirse que aunque los resultados obtenidos con ambos métodos son muy similares, utilizando el método Anova se consiguen unos resultados más estables pagando como precio un mayor costo de procesamiento.

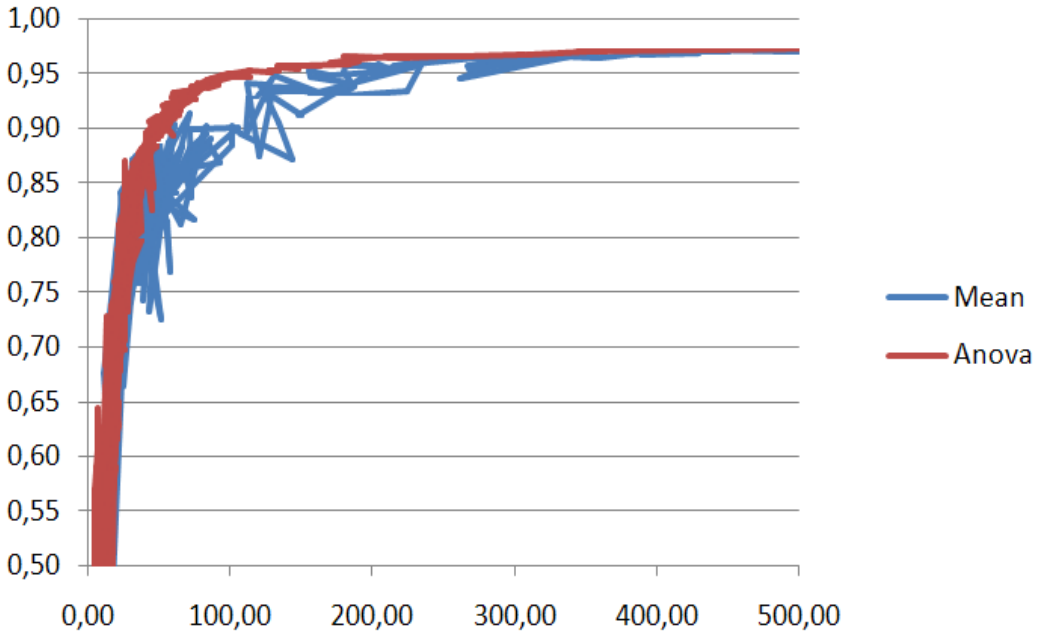


Fig. 5.17 Representación Numero de puntos vs factor de correlación sensor de humedad

5.5. Proyecto Nasistic

En la actualidad se está trabajando dentro del proyecto Nasistic en la implementación de un sistema de sensores basados en IEEE 802.15.4 para la monitorización del comportamiento de personas ancianas en el hogar para determinar posibles problemas de salud.
Para ello se pretende colocar una serie de sensores en objetos de usos cotidiano y en función de una serie de reglas parametrizables y el valor registrado por los mismos determinar si existe algún tipo de problema y si es el caso tomar la acción pertinente. El proyecto no está finalizado y está sujeto a constantes cambios pero en la Fig. 5.18 se pretende mostrar un boceto del esquema seguido a la hora de implementar dicho sistema.
En dicha figura puede verse como existe un elemento llamado Hub que se encarga de recibir los datos de todos los sensores XBee disponibles que a su vez sin enviados a un servidor web desarrollado por la empresa D2D siguiendo su protocolo de comunicación especifico JSON [8] basado en el formato de intercambio de datos basado en el subconjunto del lenguaje de programación Javascript. Dicho servidor web facilita un servicio de monitorización mediante el cual personal cualificado puede determinar, en función de los datos recibidos, un comportamiento anómalo del paciente.

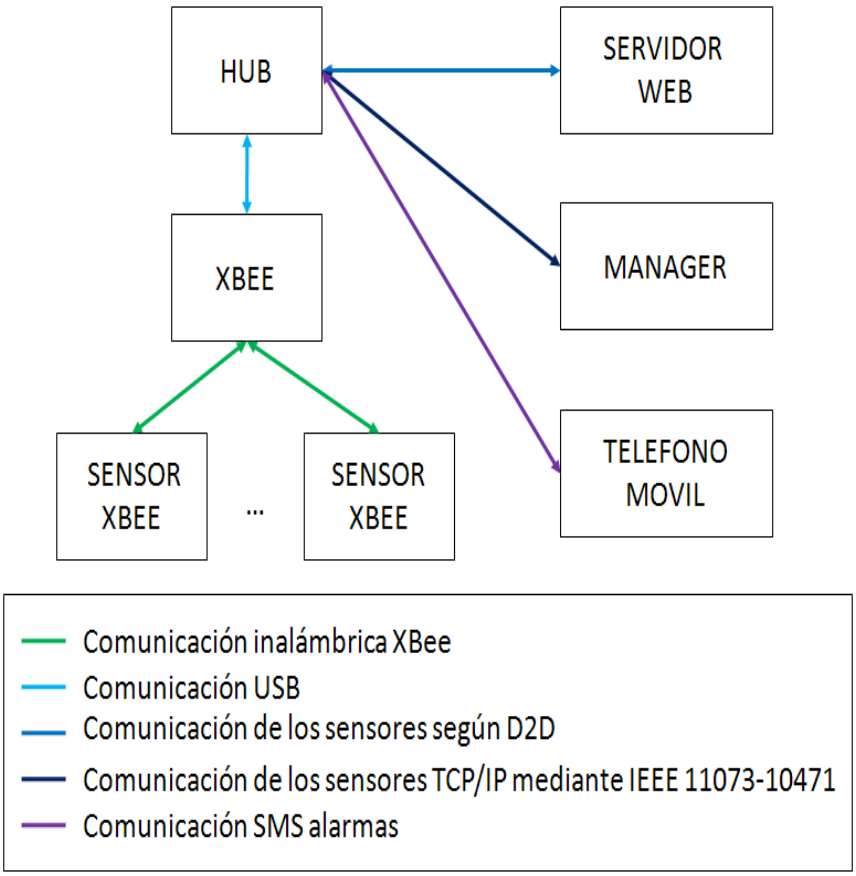


Fig. 5.18 Esquema del sistema de monitorización del comportamiento

Por otro lado, en función de los datos registrados por los sensores y las alarmas definidas en el sistema se prevé el envió de un SMS avisando a diferentes entidades dependiendo de la gravedad de las mismas.
Finalmente está contemplado dotar al dispositivo Hub de una implementación basada en la parte 10471 del estándar IEEE 11073 [9] para la comunicación de dispositivos personales de salud. El

proceso de comunicación, que puede verse en la Fig. 5.19, se basa en el envió de los datos basados en 11073-10471 mediante 'entunelamiento' TCP/IP.

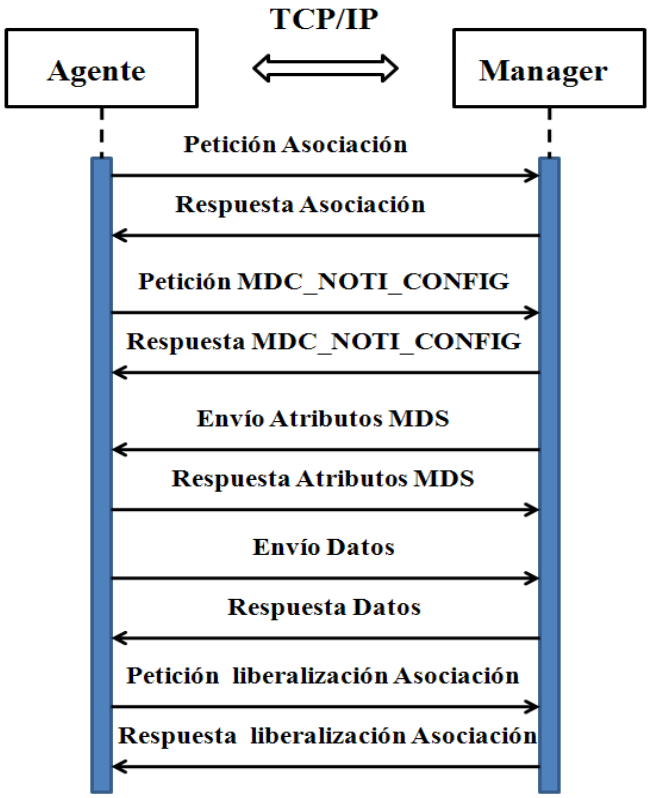


Fig. 5.19 Esquema comunicación 11073-10471 mediante 'entunelamiento' TCP/IP

Cabe mencionar que para conseguir un menor consumo de las baterías de los sensores inalámbricos se ha tratado de evitar el uso de soluciones comerciales alimentadas mayormente a 12 VDC y encontrar sensores que funcionen con una tensión de alimentación de 3.3 VDC. En la tabla 5.2 pueden verse los diferentes tipos de sensores utilizados.

Modelo	Tipo	Unidades	Rango
OKD-HZ21FA	Caudal de agua	Voltios	[0.5 , 3.3]
HAMLIN 57145	Cierre	Voltios	[0, 3.3]
SEN-09375	Peso	Kg	[0, 10]
TGS2610	Gas LP	ppm	[100,1000000]
TGS6810	Metano y Gas LP	ppm	[0,10000]
HIH-4000-001	Humedad	% RH	[0, 100]
Propio	Inundación	Voltios	[0, 3.3]
PIR	Detección movimiento	Voltios	[0.4,3.3]
TGS5042	Monóxido de carbono	ppm	[0, 10000]
LM335A	Temperatura	ºC	[0, 57]

Tabla 5.2 Lista de sensores

Para identificar el tipo de sensor dentro del sistema utilizaremos el comando NI. Básicamente consiste en un registro de 20 caracteres alfanuméricos al que se puede acceder. Utilizaremos los cuatro primeros caracteres para almacenar el tipo de sensor al que ese XBee está conectado (por

ejemplo 0014 para el sensor de tipo temperatura MDC_AI_TYPE_SENSOR_TEMP [9]). El resto de los caracteres representan el multiplicador necesario para pasar del valor de la entrada analógica del sensor a las unidades finales, representado como los valores hexadecimales de los 8 bytes que componen el valor del multiplicador en formato de coma flotante.
El Hub está implementado en la placa computadora (SBC) de bajo coste Raspberry Pi basada en tecnología ARM. Este dispositivo posee, entre otros, dos puertos USB, un puerto Ethernet, una salida HDMI y una serie de puertos de propósito general GPIO. Existen diferentes sistemas operativos disponibles y en este trabajo se ha utilizado la versión Linux optimizada basada en Debian llamada Raspbian. Debido a que la aplicación desarrollada para el Hub está programada en Java es necesario la instalación de la maquina virtual Java en el mismo. Existen varias alternativas disponibles aunque en este proyecto y dependiendo de las necesidades hardware de la aplicación final se baraja el uso de OpenJDK 7 y el Java SE Embedded 7 de 'Oracle'. Es necesario comentar que aunque la primera opción es más lenta que la segunda, al contrario que esta es gratuita.
Finalmente y para el correcto uso de los módulos XBee como dispositivos de tipo puerto serie es necesario instalar la librería correspondiente 'librxtx-java'.

5.6. Propuesta y simulación modelo Smart Grid

La idea de edificios/barrios/ciudades autosostenibles energéticamente está cobrando fuerza en la actualidad. En este trabajo se propone un modelo de cooperación energética entre edificios tipo 'Smart Grid'. Por un lado tendremos que cada edificio tiene sus fuentes de generación de energía propias y una serie de aparatos eléctricos que consumen electricidad dotados cada uno de ellos de un elemento que llamaremos 'Appliance controller' (AC). Cada edificio cuenta con un nodo de control que llamaremos 'Building Controller' (BC), que contará además con un banco de baterías para almacenar la posible energía generada en exceso.

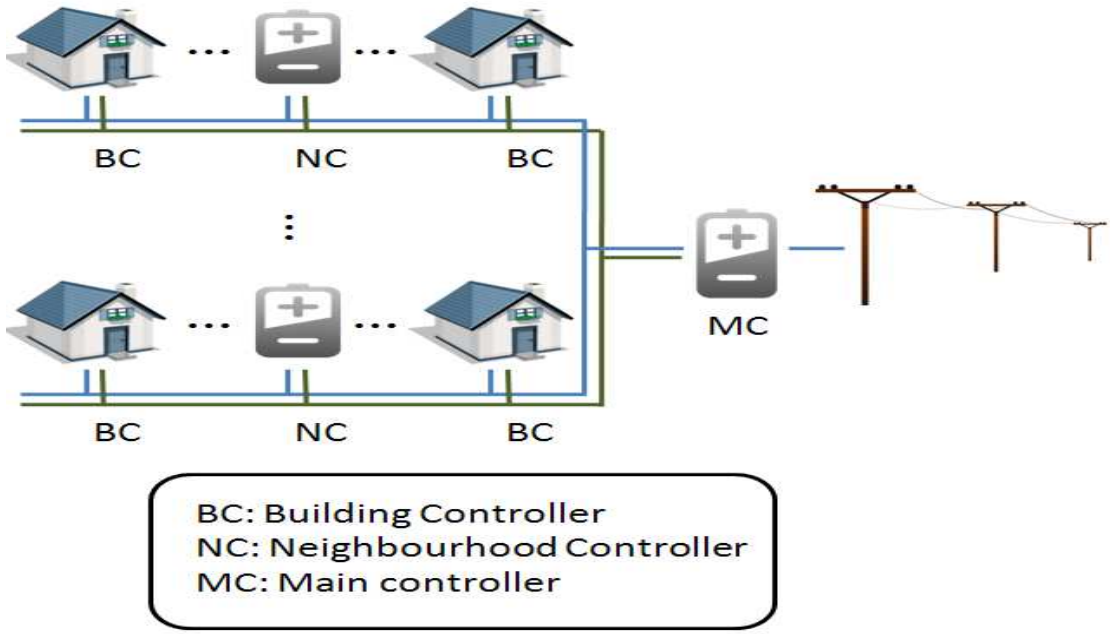


Fig. 5.20 Esquema del modelo de 'Smart Grid' propuesto

Como puede verse en la Fig. 20 los edificios se agrupan en barrios (controlados por un nodo que llamaremos 'Neighbourhood Controller') y los barrios se agrupan en una red principal (controlados por un nodo que llamaremos 'Main Controller'). El modelo propone el uso de dos tipos de redes separadas: una red de potencia eléctrica (representada en color azul) y una red de comunicación paralela (representada en color verde). El medio de comunicación y la tecnología utilizada en la

misma es transparente para el modelo, pudiéndose utilizar por ejemplo para distancias cortas tecnología de tipo '*Power Line Communications*' (PLC) y así compartir el medio o para largas distancia utilizar fibra óptica, GSM, etc… Cuando un aparato eléctrico en un edificio necesita consumir energía, su AC envía una solicitud de conexión al BC correspondiente y este, en función de características tales como la prioridad de la conexión, nivel de capacidad de su banco de baterías, etc… decide si aceptar la conexión, ponerla en espera o directamente rechazarla. Cuando a un nodo llega más energía de la que puede almacenar, este exceso de energía es enviado para su almacenado a un nodo jerárquicamente superior (un BC a un NC, un NC al MC y el MC a la red eléctrica común). Cuando un nodo necesita energía debido a que la capacidad de su banco de baterías es insuficiente para mantener el consumo de los aparatos eléctricos conectados al mismo, este defecto de energía se toma de un nodo jerárquicamente superior.

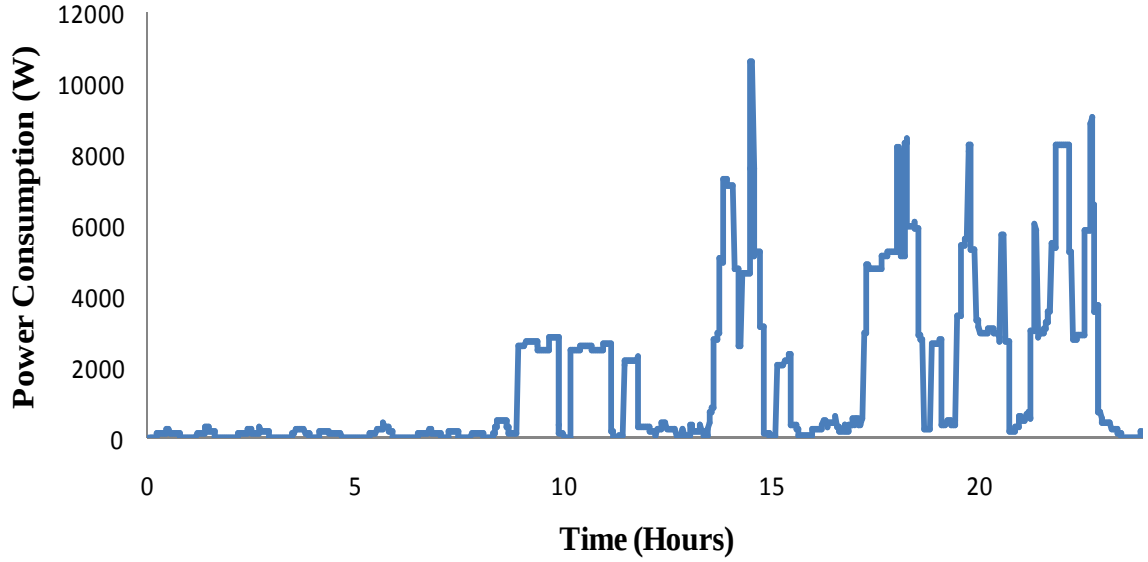


Fig. 5.21 Simulación del consumo energético

Para probar el funcionamiento del modelo propuesto se ha procedido a la implementación de un simulador que se encuentra en fase de desarrollo programado en Java. El primer paso para la simulación es generar la simulación del consumo eléctrico. Para ello primero se asigna aleatoriamente la presencia de diferentes aparatos eléctricos en cada vivienda utilizando datos estadísticos de propiedad [10][11]. Después, para cada vivienda, se simula la ocupación activa diaria de la misma basándose en el número de personas que la habitan y el tipo de día de la semana (si es fin de semana o no). El modelo [12] utilizado para ello se basa en datos estadísticos tomados de 'UK 2000Time Use Survey' (TUS) [13].

Finalmente, el último paso consiste en simular la actividad de dichos aparatos eléctricos. Para un instante de tiempo determinado y dependiendo del día de la semana, la ocupación activa calculada anteriormente y el perfil de actividad diaria [11][14] se calcula la probabilidad de que un aparato eléctrico consumo o no energía. El perfil de actividad diaria esta también basado en datos estadísticos tomados de 'UK 2000Time Use Survey' (TUS) [13]. La Fig. 5.21 muestra el resultado de la simulación del consumo energético debido a todos los aparatos eléctricos conectados en una vivienda de dos ocupantes.

El simulador contempla el uso de energía solar como fuente de generación eléctrica. Simular la energía que produce un panel solar es complicado debido a que principalmente depende de las condiciones del tiempo: si está lloviendo, si está nuboso, etc… En este trabajo se ha supuesto en todo caso condiciones de cielo totalmente despejado. El cálculo de la potencia generada se basa en

la irradiación solar que recibe el panel en cada instante. Dicha irradiación puede calcularse utilizando la siguiente ecuación:

$$S_i = I_{SC} * \sin\theta \tag{5.1}$$

donde I_{sc} representa la constante solar () y θ el ángulo de altitud solar [15]. El cálculo del ángulo de altitud solar depende a su vez de factores como la latitud, longitud y tiempo local de la localización geográfica a analizar, siendo necesario ajustar la hora local en función de la zona horaria y del cambio de hora debido al ahorro energético. La Fig. 5.22 muestra la irradiación solar en Pamplona para el día más largo y el más corto del año respectivamente.

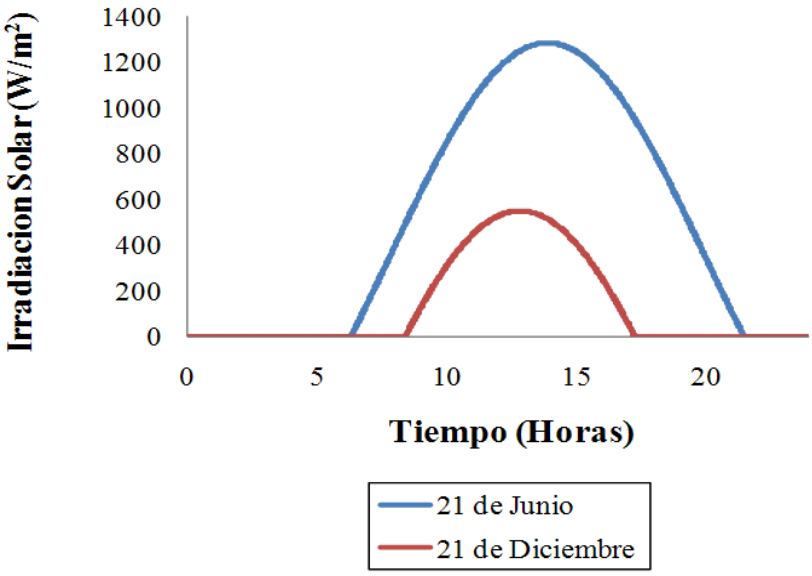


Fig. 5.22 Simulación de la irradiación solar para los días de solsticio

Por otro lado, el simulador contempla también la generación de energía eólica como fuente de generación eléctrica.

La velocidad del viento es un fenómeno estocástico que depende en gran medida de factores como la localización geográfica, la altura, etc… Es bien sabido que las distribuciones de probabilidad de los valores de la velocidad del viento siguen una distribución Weibull [16][17] independientemente de la localización geográfica [18]. Por lo tanto, para la realización de la simulación, el simulador tomo como parámetros los factores de forma y escala de dicha distribución que se ajustan a la distribución de velocidad de viento en la localización geográfica a simular. Para cada punto a simular se genera un valor aleatorio entre 0 y 1 y se calcula la velocidad de viento asociada utilizando la distribución acumulativa inversa de la misma. Con este valor de velocidad de viento y la curva de generación de potencia del molino eólico se calcula la potencia eléctrica generada asociada [19].

La Fig. 5.23 muestra el resultado de la simulación de la velocidad del viento y la correspondiente potencia eólica generada por un molino eólico con potencia nominal de 7500 W, una velocidad de arranque de viento de 4.5 m/s, un alcance de potencia nominal a 9 m/s y una velocidad de corte de viento de 45 m/s. La velocidad de viento ha sido modelada utilizando una distribución con un factor de escala de 7.93 m/s y un factor de forma de 1.97.

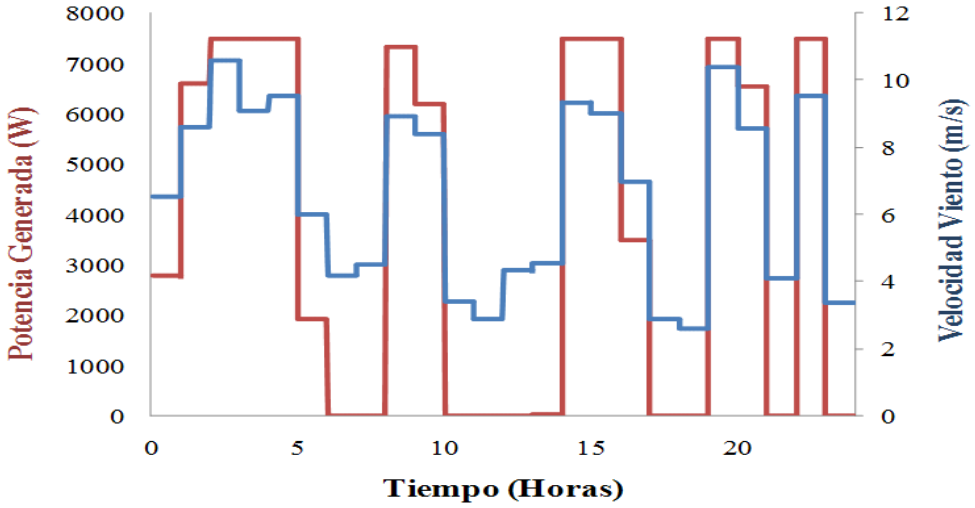


Fig. 5.23 Simulación de la velocidad del viento y potencia generada asociada

5.7. Conclusiones

A la hora de conectar diferentes sensores ópticos basados en FBG en un mismo canal de un interrogador de fibra óptica es muy importante seleccionar la longitud de onda de trabajo de los mismos para que no se interfieran espectralmente. Además, si se utilizan sensores FBG de deformación es necesario colocar otro sensor de temperatura lo más cercano posible para que en función del valor registrado por este se pueda compensar el efecto que la temperatura tiene sobre dichos sensores. A la hora de utilizar un acelerómetro es necesario ajustar convenientemente la tasa de muestreo de adquisición de los datos en función de la frecuencia máxima de la vibración que se pretenda registrar, utilizando para ello el teorema del muestreo.

Cuando se utilizan módulos XBee 'Series 2' hay que tener en cuenta que el máximo voltaje que se puede suministrar a las entradas analógicas del mismo es de 1.2 VDC. Este hecho implica una limitación a la hora de elegir los sensores a conectar aunque generalmente se puede solucionar utilizando divisores de tensión.

Cuando la cantidad de datos a almacenar en una base de datos sea considerable es conveniente buscar un mecanismo alternativo para almacenar la menor cantidad posible de los mismos para optimizar los tiempos de consulta y la cantidad final necesario de almacenaje.

5.7. Referencias

[1] Librería LabSQL, disponible en: http://jeffreytravis.com/lost/labsql.html, último acceso: 12Marzo 2013.

[2] Rutinas 'LabVIEW' para la compresión de datos, disponible en: http://zone.ni.com/devzone/cda/epd/p/id/3662, último acceso: 12Marzo 2013.

[3] Librería 'JFreeChart', disponible en: http://www.jfree.org/jfreechart/, último acceso: 12Marzo 2013.

[4] Librería 'xbee-api', disponible en: http://code.google.com/p/xbee-api/, último acceso: 12Marzo 2013.

[5] "XBee®/XBee-PRO® ZB RF Modules", disponible en: ftp://ftp1.digi.com/support/documentation/90000976_G.pdf, último acceso: 12Marzo 2013.

[6] Librería 'Flot', disponible en: http://code.google.com/p/flot/, último acceso: 12Marzo 2013.

[7] Lomax, R. G., Hahs-Vaughn, D. L., "Statistical Concepts: A Second Course", Routledge, New York,USA, 2007.

[8] "ECMAScript Language Specification", Standard ECMA-262 3rd Edition, December 1999.

[9] IEEE Std11073-10471, 2008, IEEE Engineering in Medicine and Biology Society – Health informatics—Personal health device communication; Part 10471: Device specialization—Independent living activity hub.

[10] Richardson, I., Thomson, M., Infield, D., Clifford, C., "Domestic electricity use: A high-resolution energy demand model", Energy and Buildings, 42 (10), pp 1878-1887, October 2010.

[11] Richardson, I., Thomson, M., "Domestic Electricity Demand Model Simulation Example", Loughborough University Institutional Repository, 2010, http://hdl.handle.net/2134/5786

[12] Richardson, I., Thomson, M., Infield, D., "A high-resolution domestic building occupancy model for energy demand simulations", Energy and Buildings, 40 (8), pp 1560-1566, 2008.

[13] Ipsos-RSL and Office for National Statistics, United Kingdom Time Use Survey, 2000 (Computer File), 3rd ed., UK Data Archive (distributor), Colchester, Essex, September 2003, SN: 4504.

[14] Iqbal, M., "An Introduction to Solar Radiation", Canada: Academic Press, 1983.

[15] Li, G., "Solar Altitude and Azimuth Angle Calculation", Available online at: http://lgy.myweb.uga.edu/pdf/solar.pdf, last access: 22/08/2013.

[16] Edwards, P., Hurst, R., "Level-crossing statistics of the horizontal wind speed in the planetary surface boundary layer", Chaos, 11 (3), pp. 611-618, Sept. 2001.

[17] Archer, C., Jacobson, M., "Evaluation of global wind power", Geophysical Research, Vol. 110, 2005, D12110.

[18] Archer, C., Jacobson, M., "Spatial and temporal distributions of U.S. winds and wind power at 80 m derived from measurements", Geophysical Research, Vol. 108, NO. D9, 2003.

[19] Akdağ, S. A., Güler, Ö. , "Comparison of Wind Turbine Power Curve Models", in International Renewable Energy Congress, Sousse, Tunisia, November 5-7, 2010.

CAPITULO 6 – CONCLUSIONES Y TRABAJO FUTURO

En este trabajo se analiza la integración de tres tecnologías de sensado con diferentes medios de transmisión en sistemas de monitorización y control remoto. Además se presentan varias implementaciones prácticas llevadas a cabo con la integración de dos o más tecnologías en un mismo sistema. Puede verse los diferentes usos de cada uno de ellos y como el abanico de posibilidades es enorme.

En este trabajo se ha estudiado el uso de dispositivos IEEE 802.15.4 en condiciones de baja movilidad. Una nueva línea de trabajos futuros podría ser la realización de un estudio similar pero en condiciones de alta movilidad. Por otro lado, en este trabajo se ha procedido a la realización de una VANET simple implementando la parte de la comunicación infraestructura viaria – vehicular. Otra posible línea de trabajos futuros podría consistir en añadir al sistema la implementación de la comunicación intervehicular, con los problemas radioeléctricos y de red asociados.

En cuanto a la utilización de tecnologías inalámbricas para su uso en interiores, en este trabajo se han utilizado módulos XBee basados en IEEE 802.15.4. Para futuros trabajos quizás podrían reemplazarse por módulos de diferentes fabricantes, así como diferentes sistemas de comunicación basados en IEEE 802.15.4 como ZigBee, 6LowPAN, etc. También sería interesante la utilización de otras alternativas en la banda de2.4 GHZ como dispositivos Bluetooth de bajo consumo '*Bluetooth low energy*' (BLE).

Hay que tener en cuenta también que no solamente el consumo es importante sino que también lo es el alcance máximo. Debido a que la potencia de transmisión necesaria aumenta conforme también lo hace la frecuencia de trabajo otra alternativa interesante sería explorar el uso de dispositivos de comunicación inalámbrica en bandas de frecuencia más bajas.

Uno de los problemas de estas bandas es que no son de uso global y están limitadas únicamente a diferentes países. Además, existen pocos o nulos sistemas de comunicación estándar que funcionen en ellas y los dispositivos que se encuentran están basados en soluciones propietarias.

En el reciente estándar IEEE 802.15.4d se incluye el uso de nuevas capas físicas basadas en basadas en FSK, OFDM y O-QPSK y de nuevas bandas de uso. En la tabla 6.1 pueden verse las bandas contempladas y los países en los que es posible utilizarlas.

En cuanto a la integración con KNX, un posible trabajo futuro consistiría en trabajar con el circuito integrado 'TP UART 2'. Básicamente proporciona un interface entre el bus KNX basado en cable TP y un puerto serie UART. De esta manera es posible acceder de manera bastante sencilla a los datos KNX. Otra opción a analizar podría ser el uso de un interface KNXNet/IP con soporte TCP y probar la librería JKNXNetIP. También sería interesante estudiar alternativas como el uso de dispositivos basados en LonWorks o quizás en BACnet.

Finalmente, como conclusión final, hace falta definir una norma para estandarizar el acceso a sensores y actuadores de manera remota. En vista del poco éxito obtenido por el estándar IEEE 1451, debido principalmente a su complejidad, parece claro que el nuevo estándar debe ser ante todo sencillo de implementar.

Frecuencia (MHz)	Banda
169.400–169.475 (Europe)	169 MHz
450–470 (US FCC Part 22/90)	450 MHz
470–510 (China)	470 MHz
779–787 (China)	780 MHz
863–870 (Europe)	863 MHz
896–901 (US FCC Part 90)	896 MHz
901–902 (US FCC Part 24)	901 MHz
902–928 (US)	915 MHz
917–923.5 (Korea)	917 MHz
920–928 (Japan)	920 MHz
928–960 (US, non-contiguous)	928 MHz
950–958 (Japan)	950 MHz
1427–1518 (US and Canada, non-contiguous)	1427 MHz
2400–2483.5	2450 MHz

Tabla 6.1 Bandas de frecuencia en IEEE 802.15.4d

GLOSARIO

ADS	Advanced Design System
DDNS	Dinamyc Domain Name System
EFC	Electric Field Communication
EMI	External Message Interface
ETS	EIB Tool Software
FBG	Fiber Bragg grating
FPGA	Field Programmable Gate Arrays
FSPL	Free-Space Path Loss
HID	Human Interface Device
HTTP	HyperText Markup Language
ISM	Industrial, Scientific and Medical Radio Bands
IP	Internet Protocol
ISP	Internet Service Provider
ITU	International Telecommunication Union
JNI	Java Native Interface
LAN	Local Area Network
NAT	Network Address Translation
NCAP	Network Capable Application Processor
OSA	Optical Spectrum Analyzer
PAN	Personal Area Network
PER	Packet Error Rate
PXI	eXtensions for Instrumentation
RFID	Radio Frequency IDentification
RSS	Received Signal Strength
TCP	Transmission Control Protocol
TIM	Transducer Interface Module
UDP	User Datagram Protocol
VANET	Vehicular Ad-Hoc Network
UWB	Ultra Wide Band

ANEXO

Congresos Nacionales

[1] Unzu, R., Nazabal, J.A., Vargas, G., Hernandez, R., Fernandez-Valdivielso, C., Urriza, N., Galarza, M.,Beloki, G., Yoldi, A., Matias, I., Branchi, P., Borquez, L., Lopez-Amo, M., ”Red de sensores de fibra óptica para la monitorización de nuevos sistemas de cerramientos aplicados en el sector de la construcción", Actas del OPTOEL´09 pp.311-316 ISBN: 978-84-692-3931-5, Málaga, 2009.

[2] Nazabal, J.A., Vargas, G., Hernández, R., Fernández-Valdivielso, C., Falcone, F., Lopez-Amo, M., "Integración de Sensores de Fibra Óptica en una Red KNX y su monitorización Remota", KNX International Forum 2010, Madrid, España, 2010.

[3] Nazabal, J.A., Armendariz, A., Torres, V., Becerra, J., Fernández, C., Falcone, F., "Análisis de Topología de Sensores en Prestaciones de Sistemas Zigbee Interiores", XXV Simposium Nacional de la Unión Científica Internacional de Radio , Bilbao , España ,2010.

[4] Nazabal, J.A., Torres, V., Becerra, J., Falcone, F., Fernández-Valdivielso, C., Matías, I.R., "Integración de redes de sensores en edificios inteligentes", I encuentro de Investigadores en Infraestructuras Inteligentes, Guadalajara, España, 2011.

[5] López, P., Nazábal, J.A., Azpilicueta, L., Martínez, G., García, R., Soret, J., Torres, J., Martos, J., Fernández, C., Falcone, F., "Análisis del Comportamiento de Redes de Sensores Inalámbricas en entornos Interiores Heterogéneos", XXVI Simposium Nacional de la Unión Científica Internacional de Radio , Leganés , España , 2011.

[6] López, P., Nazábal, J.A., Azpilicueta, L., Fernández-Valdivielso, C., Falcone, F., "Impacto de la potencia de fuga de hornos microondas en redes 802.15.4", XXVII Simposium Nacional de la Unión Científica Internacional de Radio, Elche, España, 2012.

Congresos Internacionales

[1] Nazabal, J.A., Torres, V., Becerra, J., Falcone, F., Fernández, C., Matías, I.R., " Fiber Optic Sensor in a KNX Network and Remote Monitoring", KNX Scientific Conference, Pamplona, Spain, 2010.

[2] Nazabal, J.A., Torres, V., Becerra, J., Falcone, F., Fernández, C., Matías, I.R., "Integration of Wireless Systems in Indoor Intelligent Home Systems", KNX Scientific Conference, Pamplona, Spain, 2010.

[3] Unzu, R., Nazabal, J.A., Vargas, G., Hernández, R., Fernández-Valdivielso, C., Urriza, N., Galarza, M., Lopez-Amo, M., "Fiberoptic sensor network for monitoring new building cladding systems ", 4th European Workshop on Optical Fibre Sensors (EWOFS 2010), Porto, Portugal, 2010.

[4] Becerra, J., Nazabal, J.A., Torres, V., Esparza, F., Navarro-Cía, M., Beruete, M., Fernández, C., Falcone, F., "Wireless Channel Modeling for Campus Sensor Networks", 14th International Symposium on Antenna Technology and Applied Electromagnetics and the American Electromagnetics Conference, Antem/Amerem, Ottawa, Canada, 2010.

[5] Torres, V., Nazabal, J.A., Fernández, C., Falcone, F., "Topologycal Analysis of Performance in Indoor Zigbee Networks", 5th European Conference on Antennas and Propagation (EuCAP 2011), Rome, Italy, 2011.

[6] López, P., Nazábal, J.A., Torres, V., Fernández, C., Falcone, F., "Analysis of Topological Impact in Wireless Indoor Sensor Networks", IEEE AP-S International Symposium on Antennas and Propagation 2011 and USNC/URSI National Radio Science Meeting 2011, Spokane, USA, 2011.

[7] López, P., Nazábal, J.A., Torres, V., Fernández, C., Falcone, F., "Analysis of Topological Impact in Wireless Indoor Sensor Networks", IEEE AP-S International Symposium on Antennas and Propagation 2011 and USNC/URSI National Radio Science Meeting 2011, Spokane, USA, 2011.

[8] Nazabal, J.A., Fernández-Valdivielso, C., Falcone, F., Matías, I.R., "Integration of Hybrid Sensing Networks in Indoor Intelligent Homes", 5th International Conference on Sensing Technology (ICST 2011), Palmerston North, New Zealand, 2011.

[9] Torres, V., Nazabal, J.A., Navarro, M., Beruete, M., Ramos, V., Fernandez, C., Falcone, F., "Analysis of Electromagnetic Dossimetry of Indoor ZigBee Networks", Progress In Electromagnetics Research Symposium 2011, PIERS 2011 in Marrakesh, Marrakesh, Morocco, 2011.

[10] Nazabal, J.A., Branchi, P., Gómez, J., Falcone, F., Fernández-Valdivielso, C., Matías, I.R., "AND@DOM: Android Application for Interacting with KNX networks via KNXNet/IP", KNX Scientific Conference, Las Palmas G.C., Spain, 2012.

[11] Nazabal, J.A., Fernández-Valdivielso, C., Falcone, F., Matías, I.R., “Home Automation based Sensor System for Monitoring Elderly People Safety", 6th International Conference on Sensing Technology (ICST 2012), Kolkata, India, 2012.

[12] Rivares, C., Azpilicueta, L., Lopez, P., Nazabal, J.A., Falcone, F., "Influence of Human Body and Indoor Scenarios in On-Body Wireless Communication Systems", APS/URSI 2013, Florida, USA, 2013.

[13] Lopez, P., Nazabal, J.A., Azpilicueta, L., Fernandez, C., Falcone, F., "Impact and Characterization of the Microwave Oven Power Leakage on 802.15.4 Networks", APS/URSI 2013, Florida, USA, 2013.

Publicaciones Nacionales

[1] Unzu, R., Nazabal, J.A., Vargas, G., Hernández, R., Fernández-Valdivielso, C., Urriza, N., Galarza, M., Beloki, G., Yoldi, A., Matías, I., Branchi, P., Borquez, L., López-Amo,

M., “Red de sensores de fibra óptica para la monitorización de nuevos sistemas de cerramientos aplicados en el sector de la construcción”, Óptica pura y aplicada, Vol. 42, pp 153,161, 2009.

Publicaciones Internacionales

[1] Nazabal, J.A., Gómez, J., Falcone, F., Fernández-Valdivielso, C., Branchi, P.E., Matías, I.R., “Android application for accessing KNX devices via IP connection”, International Journal of Smart Home 6 (4), pp. 39-46, 2012.

[2] Iturri, P.L., Nazábal, J.A., Azpilicueta, L., Rodriguez, P., Beruete, M., Fernández-Valdivielso, C., Falcone, F., “Impact of high power interference sources in planning and deployment of Wireless Sensor Networks and devices in the 2.4 GHz frequency band in heterogeneous environments”, Sensors (Switzerland) 12 (11) , pp. 15689-15708, 2012.

[3] Nazabal, J.A., Iturri, P.L., Azpilicueta, L., Falcone, F., Fernández-Valdivielso, C., “Performance analysis of IEEE 802.15.4 compliant wireless devices for heterogeneous indoor home automation environments”, International Journal of Antennas and Propagation 2012, art. No. 176383, 2012.

[4] Unzu, R., Nazabal, J.A., Vargas, G., Hernández, R.J., Fernández-Valdivielso, C., Urriza, N., Galarza, M., Lopez-Amo, M., “Fiber optic and KNX sensors network for remote monitoring a new building cladding system”, Automation in Construction 30, pp. 9-14, 2013.

[5] Nazabal, J.A., Gómez, J., Falcone, F., Fernández-Valdivielso, C., Branchi, P.E., Matías, I.R., “Accessing KNX devices using USB/KNX interfaces for remote monitoring and storing sensor data”, International Journal of Smart Home 7 (2) , pp. 105-110, 2013.

[6] Nazabal, J.A., Falcone, F., Fernández-Valdivielso, C., Matías, I.R., “Development of a low mobility IEEE 802.15.4 compliant VANET system for urban environments”, Sensors (Switzerland) 13 (6), pp. 7065-7078, 2013.

Libros

[1] Fernández, C., Matías, I.R., Gabilondo, A., Ruiz, C., Falcone, F., Castells, I., Del Villar, I., Militino, J., Nazabal, J.A., “Instalaciones de telecomunicaciones para edificios”, Ed. Marcombo, ISBN: 9788426718150.

Proyectos de Investigación

[1] "Desarrollo de sistemas industrializables de cerramientos inteligentes para su aplicación en el sector de la construcción, para la empresa AH y Asociados", Gobierno de Navarra, Departamento de industria. Ref. IIM010566.RI1.

[2] "Urbótica. Desarrollo y planificación de nuevos modelos de movilidad urbana sostenible mediante el uso de tecnologías TIC", Ministerio de Fomento. Acción estratégica en energía y cambio climático- Proyecto E23/08.

[3] “Navarra-Asistencia-TIC (NASISTIC)”, Gobierno de Navarra, Departamento de industria. Ref. IIM14089.RI1.

Premios recibidos

[1] Premio al mejor trabajo de Arquitectura Sostenible en los KNX Awards Spain 2010.