

PLATAFORMA PARA EL APRENDIZAJE DE TECNOLOGÍAS INALÁMBRICAS Y REDES DE SENSORES BASADA EN EL SISTEMA OPEN HARDWARE DENOMINADO OPENMOKO

J.M. HINOJO, D. GUTIÉRREZ, E. MARSAL, M. SOTO, F. CORTES, F. BARRERO, S. TORAL
Departamento de Ingeniería Electrónica. Escuela Superior de Ingenieros. Universidad de Sevilla.
España

jhinojo@us.es, dgutierrezreina@us.es, estebanmarsal@us.es, msoto@us.es, fcortes1@us.es,
fbarrero@esi.us.es, toral@esi.us.es

La enseñanza relacionada con las tecnologías electrónicas requiere acercar al alumno complejos sistemas, como son aquellos que se relacionan con las comunicaciones inalámbricas. Para conseguir este acercamiento se ha desarrollado un sistema para la realización de prácticas basado en un dispositivo comercial “Open Hardware” denominado Openmoko, que permite la caracterización del enlace inalámbrico y, a su vez, potencia la creatividad del alumno en la aplicación de estas tecnologías.. El sistema genera un enlace inalámbrico haciendo uso de un sistema empotrado comercial, un PC y una suite de aplicaciones con las que se pueden realizar prácticas en asignaturas adscritas a la titulación de Ingeniería de Telecomunicación e impartida por el Departamento de Ingeniería Electrónica de la Escuela Superior de Ingenieros de Sevilla.

1. Introducción

Los mecanismos de gestión y transmisión de contenidos e información están sufriendo en la actualidad un desarrollo frenético, favorecidos por el desarrollo de la sociedad de la información y, sobre todo, de las tecnologías electrónicas. Dentro de estos nuevos mecanismos cabe destacar los estándares de comunicaciones inalámbricas, que permiten la interconexión de sistemas dotados de una cierta inteligencia, como pueden ser los sistemas empotrados, formando complejas redes sin cables. Hoy en día, es habitual encontrar gran cantidad de sistemas interconectados entre sí, empleando estándares como GPS, GPRS, WiFi, Bluetooth o Zigbee. Aunque los sistemas de comunicaciones inalámbricos por excelencia son las redes WiFi, utilizadas profusamente en el mundo de la informática de consumo, otros estándares como Bluetooth o Zigbee, de menor alcance y tasa de transferencia de información pero también de mucho menor consumo, se han abierto un importante hueco a nivel industrial (recopilación de información extraída de sensores, acción de control remota, etc.) o incluso comercial (cualquier teléfono móvil o PDA que se precie de serlo incorpora en la actualidad la capacidad de comunicarse empleando el estándar Bluetooth, lo que ha permitido desarrollar nuevos sistemas de información como son los sistemas de navegación en interiores de edificios).

No obstante, la formación que los alumnos de Ingeniería reciben sobre este tipo de sistemas es fundamentalmente teórica y carece de la necesaria visión práctica por la dificultad que supone desarrollar un sistema sencillo con el que los alumnos puedan trabajar y adquirir destrezas profesionales en un tiempo razonable. Otro inconveniente asociado a este tipo de sistemas es su indefinición programática, adscrita en simultáneo a áreas de conocimiento diferentes como son la tecnología electrónica, la teoría de las señales y las comunicaciones o la telemática. Para intentar mejorar la visión práctica que el alumnado adquiere sobre el manejo de sistemas de comunicación inalámbricos, en el Departamento de Ingeniería Electrónica de la Escuela Superior de Ingenieros de Sevilla se está llevando a cabo un importante esfuerzo en pro de la remodelación de las prácticas de las diferentes asignaturas que tiene adscritas. Este es el caso de dos asignaturas denominadas “Complementos de Sistemas Electrónicos Digitales” y “Laboratorio de Instrumentación Electrónica” y, pertenecientes a 3^{er} y 5^o curso, respectivamente, de la titulación de Ingeniería de Telecomunicación. Ambas asignaturas aúnan esfuerzos en la actualidad para desarrollar nuevas prácticas que permitan al

alumno tener una visión más clara y práctica sobre el manejo de los sistemas electrónicos de comunicación y, en concreto, de los estándares inalámbricos. Los estándares WiFi y Bluetooth son algunos de los sistemas de comunicación inalámbrica que centran estos nuevos desarrollos. En este sentido, y dada las asignaturas involucradas en la experiencia, parece adecuado el desarrollo de actividades de tipo práctico que involucren al alumno con el manejo de los sistemas empotrados y el análisis y estudio de sistemas microprocesadores y microcontroladores actuales, con la caracterización de enlaces inalámbricos basados en Bluetooth. En esta dirección se encamina la propuesta docente planteada en este artículo: el desarrollo de un sistema que permita al alumno familiarizarse de forma práctica con el manejo de una tecnología inalámbrica tan extendida como Bluetooth, mediante la programación de sistemas empotrados.

2. Descripción del sistema

El sistema implementado se basa en un PC dotado de conectividad Bluetooth y ejecutando una distribución de Linux, un dispositivo empotrado conocido como OpenMoko (Fig. 1) y un conjunto de aplicaciones que permiten a los alumnos caracterizar, mediante la obtención de datos experimentales, los enlaces Bluetooth. Algunas de las características más destacables que se pueden estudiar son el establecimiento de conexiones mediante la búsqueda de servicios por parte de la aplicación cliente, cómo afecta a la búsqueda el publicar servicios de carácter privado o cómo se reconocen dichos servicios por dispositivos comerciales como puede ser un teléfono móvil.

Con respecto al PC, se ha utilizado un ordenador de sobremesa de arquitectura x86, arquitectura con la cual el alumno está familiarizado. La distribución de Linux instalada en el equipo se corresponde con una distribución Ubuntu 9.04. La elección de esta distribución se debe, en primer lugar, a que dispone de una extensa comunidad de soporte. De esta forma, si un alumno se encuentra con algún problema puede recurrir a simples buscadores web, como Google, o a foros especializados en esta distribución para resolverlo (se fomentaría así el autoaprendizaje y la búsqueda de información por parte del alumno). En segundo lugar, a la gran cantidad de paquetes software existentes para esta distribución, lo que permite tener actualizado el sistema cómodamente, así como instalar librerías adicionales de una manera muy sencilla y cómoda. Por último, la semejanza de las estructuras de ficheros a las distribuciones de tipo Debian, lo que supone un punto de partida avanzado debido a nuestra experiencia con este tipo de sistemas operativos.

Ubuntu se adapta, por tanto, a las exigencias que se le podrían imponer a una distribución de Linux destinada a fines educativos. En cuanto a las herramientas de compilación para el PC, se suministrará el compilador *gcc*, que cuenta además con la ventaja de que se encuentra incorporado dentro del sistema operativo como una utilidad más. El desarrollo de aplicaciones mediante compilación cruzada permite, además, que el alumno se familiarice con un proceso de diseño económico, ya que la programación y compilación se realiza en una máquina diferente al sistema empotrado, ahorrando tiempo y costes en el desarrollo.



Figura 1. Sistema Openmoko denominado Neo FreeRunner.

Por otro lado, el otro dispositivo que formará parte del sistema de comunicaciones inalámbricas basado en Bluetooth será un teléfono móvil de carácter libre denominado Openmoko, en concreto la versión *Neo FreeRunner*. Este terminal se corresponde con un proyecto de hardware y software

abierto destinado a usuarios avanzados para que puedan personalizar el teléfono móvil para satisfacer sus necesidades. Se caracteriza por disponer de un microprocesador Samsung S3C2442B, cuya principal característica es que presenta una arquitectura ARM920T a 400 MHz. Este tipo de arquitecturas (ARM9x) se están implantando rápidamente en los sistemas empotrados y móviles de comunicaciones debido a su elevada potencia de cálculo, su bajo consumo y el bajo coste de desarrollo que presenta. Por su parte, el uso de la misma supone que, a la hora de realizar los códigos fuentes de los programas a ejecutar en esta plataforma, debamos hacer uso de la compilación cruzada para poder obtener un ejecutable válido para el procesador en cuestión. Otras ventajas adicionales que ofrece Openmoko son los dos acelerómetros disponibles (ubicados en diferentes posiciones para poder realizar las correcciones pertinentes a las medidas obtenidas) o la conectividad GPRS, GSM, Bluetooth con chipset CSR BlueCore 4, 802.11g y GPS, entre otras. Como se puede comprobar, se corresponde con un sistema empotrado que permite su uso en una gran cantidad de aplicaciones diferentes, permitiendo que el alumno pueda investigar y desarrollar tareas en áreas de conocimientos que involucren las comunicaciones inalámbricas o los sistemas de instrumentación, medida y adquisición de datos.

El sistema operativo que gobierna Openmoko es una distribución de Linux para dispositivos móviles con un kernel 2.6.24, de forma que puede aprovecharse para que el alumno adquiera abundantes conocimientos sobre la programación de dispositivos móviles basada en el kernel de Linux. El motivo por el que se optó por usar un sistema operativo libre y no uno privado, como puede ser Symbian o Windows Mobile, se debe a que es de código abierto, lo que implica un control absoluto de las aplicaciones externas adquiridas. Además, se dispone de una amplia libertad de uso y desarrollo de aplicaciones (Fig. 2), así como de la existencia de un amplio soporte por parte de desarrolladores externos. Por ejemplo, existencia de una extensa API de programación que permite simplificar tareas como la lectura de acelerómetros o leer la señal del GPS. Por otro lado, se consigue una gran reutilización de las aplicaciones desarrolladas. Finalmente, se estima que la creación de aplicaciones será rápida puesto que al ejecutar Linux, el usuario del sistema únicamente debe conocer el funcionamiento del mismo (algo muy habitual en la actualidad) para empezar a diseñar y construir una primera aplicación.

En cuanto a los dispositivos usados para establecer el sistema destinado a la realización de las prácticas, en principio se ha implementado un enlace Bluetooth empleando dispositivos comerciales, *dongle* USB con chipset CSR BlueCore 4 compatible con la versión 2.1 del estándar Bluetooth. Esto permite realizar tareas como la modificación de la potencia de transmisión durante el proceso de *inquiry*, realizar búsquedas en modo RSSI extendido o aumentar la velocidad de transferencia. El motivo de la elección de un enlace Bluetooth para las primeras experiencias es su profusión a nivel comercial, lo que se prevé provoque cierto interés por parte de los alumnos. En cuanto al tipo de *dongle* seleccionado, se escogió uno cuyo chipset fuese compatible con la versión 2.1 del estándar Bluetooth para que el alumno pudiese experimentar con todos los comandos de la capa HCI que permiten configurar la comunicación inalámbrica a bajo nivel.

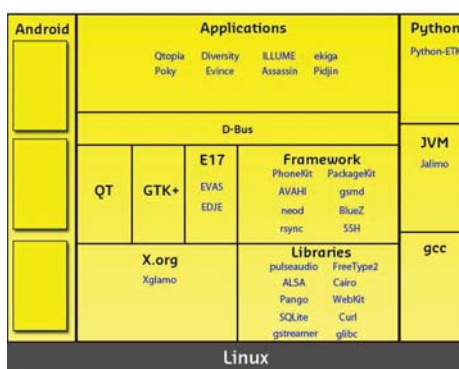


Figura 2. Estructura software del dispositivo Neo FreeRunner.

Finalmente, para dotar de una cierta funcionalidad a esta plataforma de aprendizaje hemos desarrollado una aplicación que sigue una arquitectura de tipo cliente-servidor. Para ello, hacemos uso del protocolo SDP (Service Discovery Protocol) o Protocolo de Descubrimiento de Servicios. Este protocolo permite a los programas clientes descubrir la existencia de diversos servicios proporcionados por uno o varios servidores de aplicaciones, junto con los atributos y propiedades de los servicios que se ofrecen. Estos atributos incluyen el tipo o clase de servicio ofrecido y el mecanismo o la información necesaria para utilizar los mismos.

SDP se basa en una determinada comunicación entre un servidor que contiene los servicios ofrecidos y un cliente que desea algún servicio específico. El servidor mantiene una lista de registros de servicios, los cuales describen las características de estos. Cada registro contiene información sobre un determinado servicio. Para recuperar la información asociada a alguno de estos registros, el cliente debe realizar una petición SDP. Si el cliente o la aplicación asociada con el cliente deciden utilizar un servicio, deben establecer una conexión independiente con el servicio en cuestión. SDP proporciona un mecanismo para el descubrimiento de servicios y sus atributos asociados, pero no proporciona ningún mecanismo ni protocolo para utilizar dichos servicios. Es en este punto donde se establece una comunicación RFCOMM (Fig. 3), que se corresponde con la emulación de un puerto serie para transmitir información a través de él.

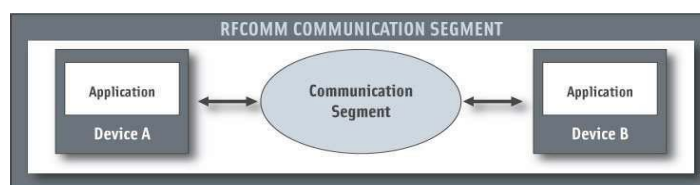


Figura 3. Enlace de comunicación establecido por RFCOMM.

Por su parte, un cliente SDP normalmente realiza una búsqueda de servicios acotada por determinadas características. No obstante, hay momentos en los que resulta deseable descubrir todos los servicios ofrecidos por un servidor SDP sin que pueda existir ningún conocimiento previo sobre los registros que pueda contener. Este proceso de búsqueda de cualquier servicio ofrecido se denomina navegación o *browsing*.

La tecnología inalámbrica Bluetooth hace un uso muy amplio de procesos de descubrimiento que permiten a los dispositivos identificarse entre sí cuando entran dentro del radio de acción, y establecer las conexiones apropiadas para que los dispositivos puedan ejecutar las aplicaciones comunes y acceder a varios servicios. Por medio del descubrimiento de servicios, las redes personales son básicamente autoconfigurables. Esto, a su vez, hace a los dispositivos portátiles muy sencillos de utilizar. En consecuencia, las conexiones son enteramente transparentes para el usuario y no requieren asistentes de instalación ni configuraciones en línea.

3. Aplicación del sistema

Para ayudar a los alumnos a familiarizarse con el sistema propuesto, se han desarrollado una aplicación de tipo servidor que ofrece servicios y un programa cliente que realiza la búsqueda de estos servicios. En el caso del servidor, el programa se encargará de atender las peticiones del cliente suministrándole los ficheros de las páginas del menú solicitadas. También deberá registrar un servicio SDP que permita al cliente identificar al equipo en concreto. El cliente, por contra, deberá poder buscar un equipo a partir de un determinado servicio, así como solicitar los ficheros asociados a las entradas de los menús visitados y procesarlos. La idea desarrollada ha consistido en aprovechar el protocolo SDP para poder descubrir aquellos dispositivos que pertenezcan a nuestra aplicación para posteriormente solicitarle un inicio de conexión y poder realizar el envío de información a través del enlace establecido (Fig. 4). Por tanto, tendríamos el programa que actúa como servidor en el PC o en una máquina remota. Éste, al arrancar, registrará en una tabla los servicios que dicha máquina es capaz de soportar. Este servicio deberá ser público para que el cliente, la aplicación a ejecutar en el terminal

móvil, pueda realizar una búsqueda del mismo y encontrarlo. Una vez que se haya registrado, el servidor deberá quedar a la escucha de una petición de conexión. Por otro lado, el programa cliente, lo primero que tendrá que realizar al arrancar es un escaneo de los dispositivos que tiene a su alcance para que, una vez detectados todos, pueda realizar una consulta a cada uno de ellos para determinar si el servicio que se solicita está disponible. En caso de encontrarlo, deberá solicitar los atributos que posee dicho servicio para poder configurar la petición de conexión. Una vez configurada, se solicitará el inicio de una comunicación mediante el establecimiento de una conexión RFCOMM al canal donde el servidor se encuentra a la escucha. Tras la detección del equipo remoto con el programa servidor, y el correspondiente establecimiento de conexión, el cliente solicitará el fichero principal de configuración de los menús, que albergará la temática de las entradas, y los procesará, identificando aquellas líneas que hagan referencia a texto plano a mostrar en la ventana y a botones que contengan las entradas de los menús.

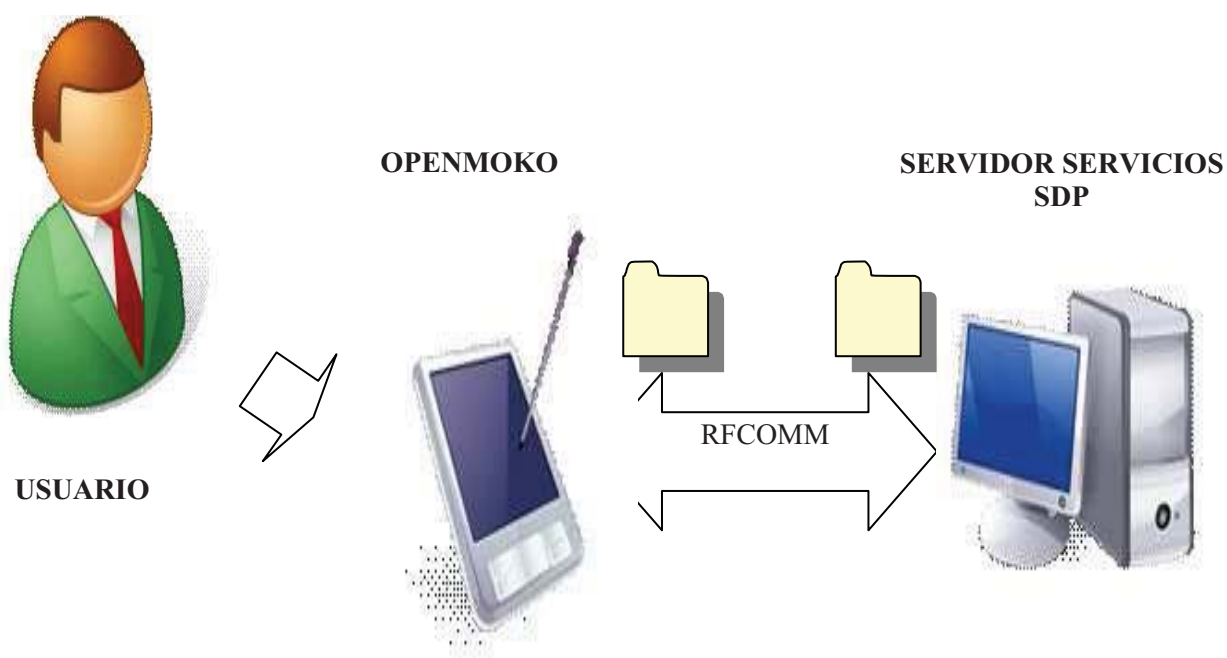


Figura 4. Estructura básica de la aplicación cliente y servidor desarrollada.

El funcionamiento de la aplicación desarrollada, fue comprobado mediante dos pruebas. Una primera en la que se modificó el tiempo de búsqueda del servidor por parte del cliente (buscando minimizar el tiempo que el cliente debe esperar para poder encontrar al servidor), y una segunda en la que se declaró un servicio como privado para comprobar que verdaderamente dicho servicio quedaba oculto al cliente. A continuación, se muestran los resultados obtenidos (Fig. 5)(Fig. 6). Como se puede observar, esta aplicación sirve como una muestra del potencial que este sistema de comunicaciones inalámbricas posee y la diversidad de proyectos en los que se puede aplicar.


```

jose@jose-dell:~/Escritorio/PFC/trabajo_curso$ sudo ./server
Conexión aceptada desde 00:06:6E:17:23:FC
Recibido [main.men]
[T] Laboratorio de Instrumentacion Electronica.

[T] 2008/2009

[T] Programa de prueba

[T] Para salir pulse 0

[T] Bienvenidos al menu de seleccion de nuestra aplicacion:

[B] 15 5 Ocio Ocio.men
[B] 15 5 Restaurantes Restaurantes.men
[B] 15 5 Ropa Ropa.men

FIN

Recibido [Ocio.men]
[T] Menu ocio:

[B] 15 5 Horario a.men
[B] 15 5 Tarifa a.men
[B] 15 5 Sacar billete a.men
[B] 15 5 Atras main.men

FIN

Recibido [main.men]
[T] Laboratorio de Instrumentacion Electronica.

[T] 2008/2009

[T] Programa de prueba

[T] Para salir pulse 0

[T] Bienvenidos al menu de seleccion de nuestra aplicacion:

[B] 15 5 Ocio Ocio.men
[B] 15 5 Restaurantes Restaurantes.men
[B] 15 5 Ropa Ropa.men

FIN

```

Figura 5. Pruebas de las aplicaciones cliente y servidor. Ejecución del servidor.

```

root@om-gta02:/home# ./client
Numero de dispositivos encontrado: 3
[num] [direccion] [nombre]
[ 1] [00:17:E8:F4:7D:7A] [Jose Maria Hinojo]
[ 2] [00:24:91:10:F4:7E] [SGH-1900]
[ 3] [00:23:4D:E8:AC:5C] [jose-dell-0]
No se ha localizado el puerto, 0
No se ha localizado el puerto, 0
Servicio encontrado corriendo en el puerto 10
Nombre: [Laboratorio de Instrumetnacion Electronica]
Descripcion: [Servicio de prueba para una aplicacion piloto]
Proveedor: [Laboratorio LIE]

Laboratorio de Instrumentacion Electronica.
2008/2009
Programa de prueba
Para salir pulse 0
Bienvenidos al menu de seleccion de nuestra aplicacion:
1. Ocio
2. Restaurantes
3. Ropa

Seleccione una opcion: 1
Menu ocio:
1. Horario
2. Tarifa
3. Sacar billete
4. Atras

Seleccione una opcion: 4
Laboratorio de Instrumentacion Electronica.
2008/2009
Programa de prueba
Para salir pulse 0
Bienvenidos al menu de seleccion de nuestra aplicacion:
1. Ocio
2. Restaurantes
3. Ropa

Seleccione una opcion: █

```

Figura 6. Pruebas de las aplicaciones cliente y servidor. Ejecución del cliente.

4. Aplicación académica

El sistema desarrollado tendrá como finalidad su aplicación en prácticas tanto en cursos introductorios a los sistemas microprocesadores basados en DSPs como puede ser “Complementos de Sistemas Electrónicos Digitales” como en las asignaturas de último curso, donde se le presuponen al alumno los conocimientos suficientes como para poder sacarle el máximo provecho al sistema mediante la realización de trabajos prácticos. En función del nivel del curso en el que se utilice se pueden adaptar las prácticas.

Así, por ejemplo, en las asignaturas de cursos inferiores se puede emplear el sistema para introducir a los alumnos en el proceso de compilación cruzada, de tal forma que sean ellos los que realicen una aplicación sencilla que haga uso del algún módulo del sistema desarrollado. Esto les permitirá adquirir pericia con el proceso de realización de programas para microprocesadores. Con este fin, tendrán que codificar y compilar en el PC un programa mediante el uso del compilador cruzado y las librerías necesarias para, finalmente, descargarlo en el sistema Openmoko donde se ejecutará y depurará.

Para facilitar esta labor y como parte del sistema diseñado, se desarrolló una API de libre disposición para los alumnos, interfaz de programación que permite leer los valores de los acelerómetros del terminal Openmoko. De esta forma, el alumno puede hacer uso de las funciones incluidas en la API para realizar una lectura del acelerómetro y en función del valor obtenido generar una respuesta u otra en la pantalla del dispositivo.

Por otra parte, en los últimos cursos, los alumnos poseen un mayor conocimiento de diversas disciplinas como telemática, teoría de la señal o electrónica. Esto permitiría sacarle un mayor provecho al sistema puesto que en este caso, pueden aplicar los conocimientos adquiridos en estas parcelas para aplicarlos en la creación de complejas aplicaciones. Como ejemplo de posibles proyectos que se les podría asignar a los alumnos se podría mencionar el diseño de redes distribuidas de procesamiento basados en un sistema inalámbrico de comunicaciones como el diseñado, aplicaciones de tipo cliente/servidor que hagan uso de servicios web o el diseño e implementación de algoritmos de posicionamiento.

5. Conclusión

Se ha desarrollado una plataforma de comunicaciones inalámbricas, basada en el estándar Bluetooth, para motivar a los alumnos de Ingeniería mediante el uso de dispositivos electrónicos de consumo. El sistema desarrollado es una herramienta novedosa y versátil que permite realizar prácticas a alumnos de los primeros cursos y a aquellos que se haya cerca de finalizar sus estudios. El hecho de hacer uso de un sistema “open hardware” como es Openmoko permite ofrecer al alumno un sistema atractivo al alumnos, además de ofrecerle un entorno (incluido el sistema operativo) totalmente configurable por él. La existencia de una comunidad de desarrolladores detrás del dispositivo seleccionado permite reducir los tiempos de aprendizaje, aumentar los conocimientos adquiridos y fomentar el autoaprendizaje y la búsqueda de información por parte de los alumnos, cualidades que resultarán muy beneficiosas en su futura actividad profesional.

Referencias

- [1] Página oficial del proyecto Openmoko. http://wiki.openmoko.org/wiki/Main_Page
- [2] Albert S. Huang, Larry Rudolph. *Bluetooth essentials for programmers*,. Cambridge University Press (2007).
- [3] Robert Morrow. *Bluetooth operation and use*, McGraw-Hill (2002).
- [4] H. Schildt. *C:Manual de referencia*. McGraw-Hill (2002).
- [5] Bluetooth SIG. *Core Specification v2.1 + EDR*. <http://www.bluetooth.com>
- [6] Ian McLoughlin. *Secure embedded systems: the threat of reverse engineering*. 14th IEEE International Conference on Parallel and Distributed Systems. pp. 729-736. (2008).