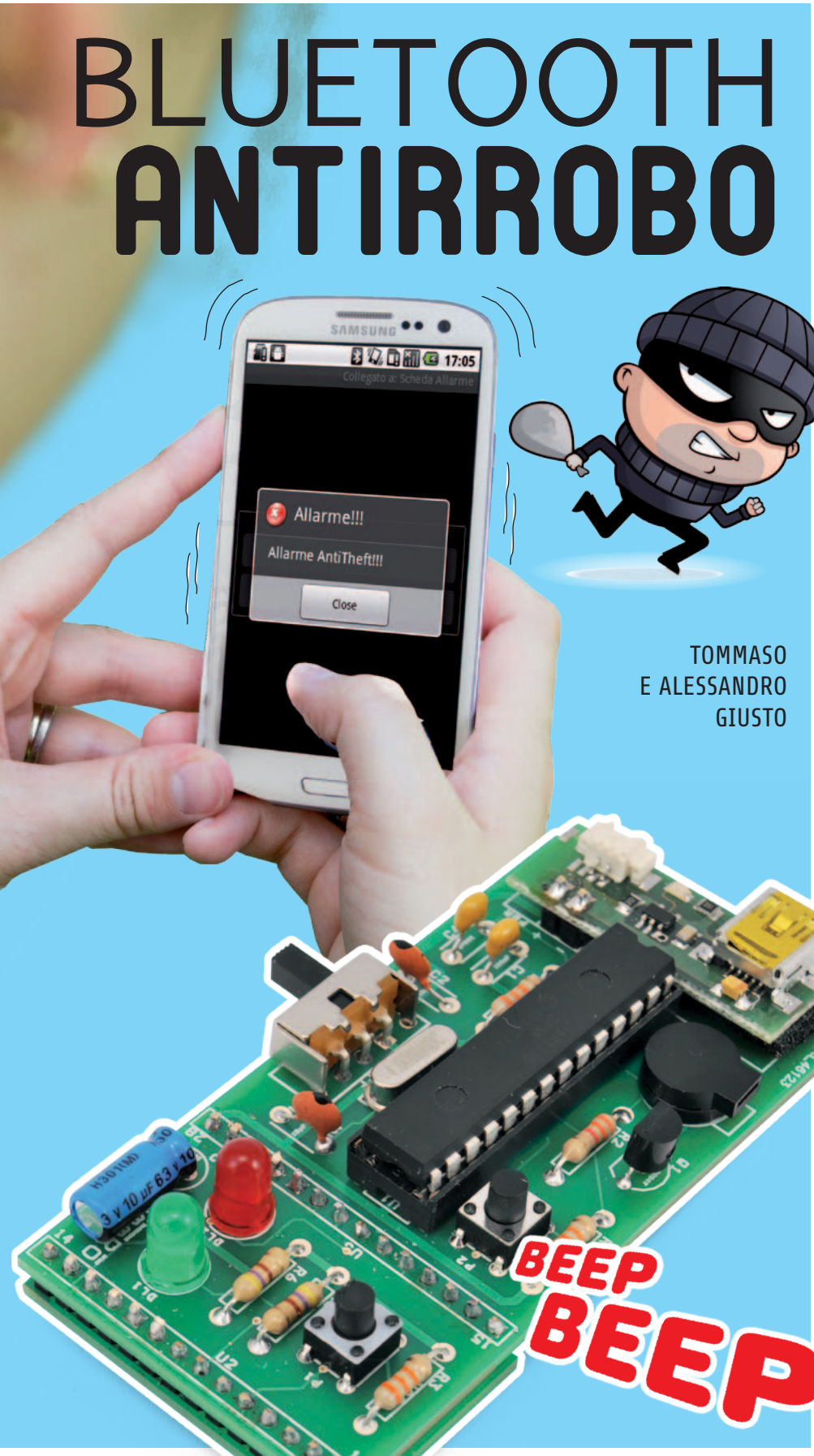
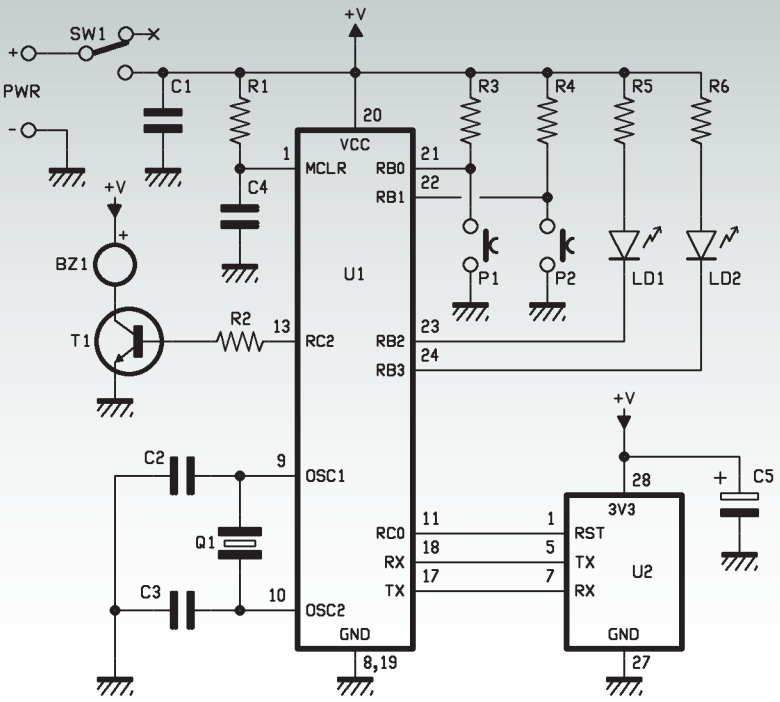
BLUETOOTH ANTIRROBO



TOMMASO
E ALESSANDRO
GIUSTO

Sistema antirrobo con tecnología bluetooth que usa una aplicación en smartphone Android para medir la distancia entre nosotros y un objeto de valor que queremos tener bajo control y, en el caso en que la distancia supere un cierto umbral, lo indica con alarmas a través de un zumbador sonoro y la vibración del móvil.

Siempre es buen momento para unas vacaciones y quien tenga la suerte de poder tomarlas, podrá hacer las maletas y prepararse para el viaje. En tema de maletas, a la vista de las recientes estadísticas, parece que cada año en los aeropuertos, estaciones ferroviarias, metropolitanas, etc... (en general: los lugares de salida públicos) desaparecen alguna que otra de la mano de los "sospechosos habituales". En estos casos los criminales tienen más fácil su trabajo, además de por las condiciones de confusión típicas estos ambientes en esos días, por el hecho que antes de que el legítimo propietario (ocupado en los varios check-in, registros, colas en mostradores, etc….) se dé cuenta del robo normalmente pasa bastante tiempo; los ladrones tienen entonces todo el tiempo necesario para "desaparecer de la circulación", abrir con calma el fruto de su trabajo y apropiarse de todas las cosas de valor que encuentran. Desde todas estas consideraciones ha nacido la idea del proyecto que presentamos en estas páginas. Se trata de un dispositivo que, a través de smartphone Android, permite determinar y monitorizar



la distancia entre nosotros y un objeto que queramos controlar; en el caso en que esta distancia llegue a ser excesiva o supere determinadas condiciones, nos avisa al smartphone a través de mensaje en display y vibración. Además, el dispositivo también emitirá una alarma sonora con un efecto doble: "asustar" y desalentar a los delincuentes (está comprobado que en situaciones de pánico se cometen más errores que en calma) y atraer nuestra atención y la de las personas a nuestro alrededor (nunca se sabe si alguna persona de buena voluntad intervendrá en nuestra ayuda).

Nuestro ejemplo es el primer caso de uso real del sistema que nos ha venido a la mente y es la típica condición para la cual todo ha sido proyectado; nada hay en contra en adaptarlo para otros fines.

Conectado a un animal doméstico podría permitirnos tenerlo bajo control más fácilmente en el parque, sin la necesidad de tener que seguirlo físicamente todo el tiempo; en obras de construcción donde trabajan diferentes grupos de trabajadores y donde se podría controlar con menos estrés una herramienta valiosa, y en tantísimos otros casos. La tecnología utilizada para tener bajo control la distancia es el Bluetooth. Hemos realizado un esquema electrónico (que funciona con batería recargable) que se combinará con el objeto que queremos controlar (mejor oculto y/o insertado en el objeto mismo); a través del software específico Android conectaremos nuestro móvil al dispositivo, activaremos las alarmas y podremos tranquilamente alejarnos. Utilizando una conexión Bluetooth, el software y el periférico cíclicamente se intercambiaran datos como verificación de la conexión; en caso que esta comunicación falle, ambas unidades identificarán que el otro periférico se ha eliminado del campo de cobertura y por tanto ambas activaran las alarmas.

El software permite activar y desactivar las alarmas; de esta manera si debemos alejarnos por un determinado momento dejando el objeto a controlar a personas de confianza, podremos desactivar el sistema evitando inútiles y molestas alarmas; una vez volvamos, retornaremos el sistema a las condiciones normales de uso. Esta prevista además una página de configuración en la cual es posible designar al periférico el nombre Bluetooth (de forma que podamos identificarlo más fácilmente en el caso en que se encuentre en una zona caracterizada por la presencia de distintos dispositivos con Bluetooth activo), el código PIN de acceso y el nivel de potencia de transmisión radio (para poder elegir distintos radios de cobertura y/o alarmas de antirrobo y adaptar el funcionamiento a los espacios abiertos o cerrados). Claramente todos estos ajustes son configurables sin tener que acceder físicamente al objeto, gracias al enlace Bluetooth.

La seguridad de todo el sistema está garantizada intrínsecamente por el Bluetooth: de hecho el acceso al periférico será posible solo si se conoce el correspondiente código PIN Bluetooth ; además la comunicación será cifrada propiamente por los niveles más bajos del mismo protocolo Bluetooth.

TARJETA HARDWARE ANTIRROBO

En el esquema eléctrico del proyecto podemos ver que el hardware del dispositivo está basado principalmente en un microcontrolador PIC18F2520 y el módulo Bluetooth RN-42 (analizado ampliamente en números anteriores de la revista). Las dos partes están interconectadas a través de una conexión serie; a través el puerto UART el microcontrolador conversa, programa y gestiona el módulo radio.

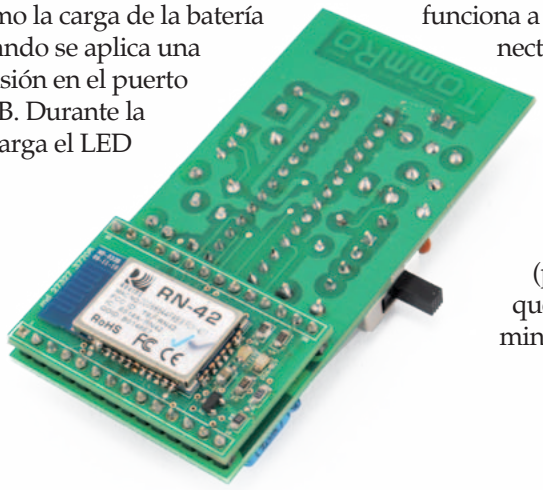
Todo el sistema está alimentado a través de una batería recargable de iones de litio con tensión nominal de alrededor de 3,7 voltios y capacidad de alrededor de 1000 mAh. Como cargador hemos elegido un hardware ya listo, fiable y probado (disponible en la tienda de Nueva Electrónica con el código FT864M).

Protocolo de **comunicación**

El protocolo de comunicación entre tarjeta y smartphone es bastante simple. Como ya hemos dicho en anteriores artículos referentes al módulo Bluetooth RN-42, desde las especificaciones están previstas 2 clases distintas de dispositivos: master y slave. A la primera clase pertenecen todos aquellos dispositivos que, por propia iniciativa, pueden arrancar una nueva conexión mientras a la segunda clase pertenecen aquellos que están siempre en espera de una petición de conexión; haciendo una comparación con la comunicación TCP/IP, los master son equivalentes a los clientes mientras los slave a los servidores. En nuestro proyecto específico el smartphone se comporta como master mientras la tarjeta electrónica como slave.
El protocolo prevé tramas estructuradas así:
- primer byte: tipo de operación (lectura o aplicación estado tarjeta); GETNFO_CMD (byte 0x41) para lectura; SETNFO_CMD (byte 0x42) para aplicación;
- segundo byte: identificador tipo de recurso. Están previstas 4 posibilidades: estado de activación alarmas y actual nivel de la potencia Bluetooth de transmisión (STWRNG_CMD; byte 0x45); nombre Bluetooth (NMEBTH_CMD; byte 0x46); código PIN Bluetooth (PNCBTH_CMD; byte 0x47) y finalmente potencia de transmisión Bluetooth (PTRXTH_CMD; byte 0x48 usado solo en escritura, para la lectura esta ya comprendido en el comando STWRNG_CMD).
- cuarto byte: estado de activación de alarmas o nivel de potencia de trasmisión.

El protocolo prevé además que a cada petición de comandos provenientes del smartphone master, la tarjeta electrónica (slave) responda con paquetes de confirmación.

Este sistema dispone como entrada de un puerto en formato mini USB a utilizar para conectar el cargador (gracias a los populares smartphone este formato se ha convertido en común y está presente e muchos adaptadores para móviles; o la tensión de 5 VDC necesaria para la recarga se puede tomar de los puertos USB de un PC) y de dos puertos de salida: el primero compuesta físicamente por 2 pines de paso 2,54 mm que se utiliza como puerto de alimentación para nuestra tarjeta electrónica (el formato hace simple la integración en los PCB; vasta prever en la fase de diseño de una tira de 2 pinesmacho); el otro se emplea para conectar la batería recargable (esta en realidad tiene funcionalidad tanto de salida como de entrada, dependiendo de is el cargados está conectado o no). Como ya se ha dicho, la tensión de entrada debe ser de 5 VDC (standard USB) mientras aquellas de salida son ambas de 3,7 voltios.
El hecho de utilizar un kit completo de este tipo facilita notablemente la integración en los proyectos propios de un sistema de carga de batería; también para todos vuestros proyectos nos parece inútil "lanzarse" a proyectar una sección específica teniendo ya algo listo y funcionando. Además, el módulo es capaz de comunicar automáticamente si una alimentación está conectada al puerto USB y/o una batería, y gestiona automáticamente tanto el paso de una fuente a la otra para la alimentación, como la carga de la batería cuando se aplica una tensión en el puerto USB. Durante la recarga el LED verde del módulo permanece encendido fijo; cuando la batería está completamente cargada – ósea cuando la tensión en los extremos de la batería alcanza los 4,2 VDC - el LED se apaga.



Analizado el cargador, volvemos al esquema eléctrico de la tarjeta donde está presente un interruptor para el encendido/apagado de la tarjeta (como se ve del esquema conecta y remueve el polo positivo de la tensión de alimentación); y también está presente el condensador C1 para ofrecer un mínimo de energía más en caso de picos breves en la absorción de corriente del sistema general.
Como ya anticipamos, el cerebro está representado por el PIC18F2520 (integrado U1) que funciona a 4MHz (cuarzo Q1) conectado al módulo Bluetooth físicamente a través del específico doble zócalo de paso 2,54 mm de 28 pin y lógicamente a través de la UART (pin 17 y 18 del PIC que respectivamente terminan en los pines 7 y 5 del zócalo) y el pin de reset (pin 11 de U1) usado por el MCU en fase de arranque para llevar el RN-42 a las condiciones por defecto.
Al zócalo para el Bluetooth está conectado el condensador C5, necesario para compensar los picos de absorción, pero solo del módulo RN-42 (típicamente en las fases en las que la sección radio esté activa, es decir, en transmisión y recepción). A diferencia de C1 que es para el sistema general, C5 está sin embargo dedicado al solo Bluetooth y también en la fase de diseño del PCB (como se indica en el datasheet del RN-42) se ha posicionado lo más cercano posible al zócalo.
Volviendo al análisis del esquema eléctrico, se incluye un zumbador (BZ1) para generar las alarmas sonoras. El componente se controla mediante el transistor NPN BC547 (T1) que conecta o no el polo negativo de BZ1 a masa. En la fase de diseño hemos luchado para encontrar un zumbador que funcione a 3,6 voltios; en particular hemos encontrado solo del tipo sin circuito de control interno. En la práctica, para activar el sonido, no

Interfaz **usuario-tarjeta**

La tarjeta dispone de 2 LED (verde y rojo) para dar una indicación visible del propio estado y de 2 pulsadores con los cuales el usuario puede interactuar.
El LED verde indica si una conexión Bluetooth está activa y si el software está en ejecución. Durante el funcionamiento normal en conexión down, el LED parpadea con un intervalo de alrededor de 1 segundo; sin embargo durante todo el tiempo en el que el enlace está activo, el LED permanece encendido fijo.
El LED rojo indica el estado de activación de las alarmas. Alarmas inactivas corresponde al LED rojo apagado; alarmas activas al LED rojo encendido fijo.
El pulsador P1 se emplea para comprobar el zumbador: si las alarmas están desactivadas y no está activa una conexión Bluetooth, pulsando P1 se activa el zumbador alrededor de un 1 segundo.
P2 se utiliza para pasar de un nivel de potencia de transmisión al siguiente. Los posibles niveles seleccionables son 7 (de 1 a 7) en los que 1 significa nivel de potencia mínimo. Pasando de un nivel al siguiente la tarjeta señala el nivel actual a través de parpadeos de los LED y el zumbador en números pares al nivel seleccionado. Además desde el smartphone Android, en la página de configuración, es posible comprobar visiblemente el valor activo.

es suficiente alimentar el zumbador, es necesario hacerlo a través de una señal PWM de determinadas frecuencias y ciclo de trabajo. Por este motivo la base del transistor está conectada al pin 13 del PIC, una de las dos salidas del MCU destinadas a generar señales PWM (en concreto a CCP1 – módulo Compare Capture Pulse 1). En particular para el zumbador que hemos elegido (Part Number: 254-EMB93-RO producido por la empresa Kobitone) la señal PWM a generar deberá tener frecuencia de alrededor 2731Hz ± 200 Hz (será necesario tenerlo presente en la fase de escritura del firmware del PIC).
Para terminar, están los dos LED (LD1 de color verde y LD2 de color rojo, con las respectivas resistencias de polarización R5 y R6) gestionados por firmware a través de los pin 23 y 24 del PIC. En particular hacemos hincapié en que dado que la corriente a través de los LED es bastante baja, no hemos previsto controlarlos mediante un transistor (como sin embargo hemos hecho para el zumbador que típicamente consume más), sus cátodos están directamente conectados a las salidas del micro.
El LED verde señala el estado de la conexión Bluetooth: permanece encendido fijo en el periodo temporal en el que la conexión es activa; para señalar que el firmware del PIC está en ejecución, el LED parpadea con frecuencia de alrededor 0,5Hz cuando la conexión está OFF.
El LED rojo señala sin embargo el estado de activación de las alarmas: si el sistema está habilitado (alarmas activas) el LED rojo está encendido; viceversa si el control de las alarmas está desactivado el LED rojo está apagado.
La última anotación concierne a la presencia de los dos pulsadores P1 y P2 (conectados a los pines 21 y 22 de U1) cuyas funcionalidad viene definida/gestiona por firmware: cuando el control de las alarmas está desactivado y ninguna conexión Bluetooth está en curso, pulsando P1 es posible ejecutar un breve test del zumbador; sin embargo (siempre alarmas y conexiones en las mismas condiciones de activación) pulsando P2 es posible incrementar en una unidad el nivel de la potencia de transmisión (teniendo en cuenta que alcanzado el nivel máximo se vuelve al mínimo y mediante distintos parpadeos de los LED y en el zumbador viene señalado el nivel seleccionado).

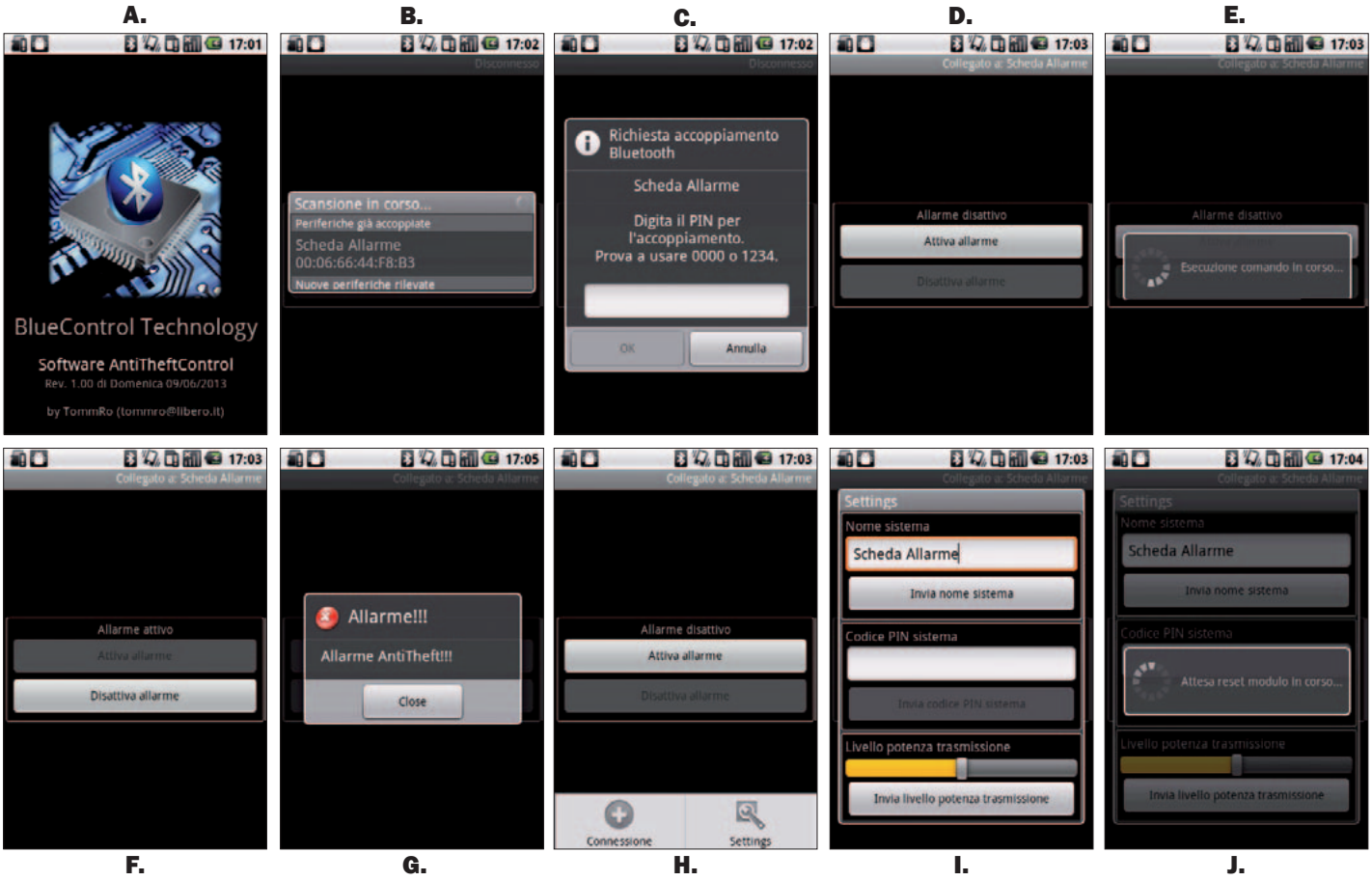
FIRMWARE TARJETA ANTIRROBO

Habiendo analizado ya ampliamente y descrito el módulo Bluetooth RN-42 (tanto como características hardware como protocolo UART de comunicación), entender cuáles son las características y funcionalidades implementadas en el firmware de la tarjeta electrónica resulta bastante intuitivo.
Todos los ajustes de configuración de la tarjeta (nombre Bluetooth, código PIN, nivel de potencia de transmisión radio y finalmente estado de activación de las alarmas) son memorizadas en la EEPROM del microcontrolador de tal manera que no se pierda entre un encendido y otro.
El firmware ha sido escrito para PIC18F2520 utilizando el lenguaje C. En el arranque se inicializan los recursos hardware utilizados (la UART de comunicación de la que se usa la correspondiente interrupción en recepción para memorizar los caracteres de entrada sin ralentizar demasiado el firmware; los pin usados como Entradas/Salidas digitales; el timer 1 cuya interrupción se usa para generar un reloj software para determinar el transcurso del tiempo por distintos timeout y finalmente el timer 2 usado para generar el PWM del zumbador).
Después se inicializa el módulo Bluetooth RN-42: se selecciona la modalidad de funcionamiento slave, se establecen algunos temporizadores de configuración, la potencia de transmisión, el nombre, el PIN y la potencia de transmisión Bluetooth (estos 3 últimos leyendo de la EEPROM el último valor memorizado) y finalmente se requiere el reinicio del módulo de manera que se hagan efectivas las configuraciones (esta operación vendrá ejecutada también más tarde, cada vez que desde el software Android se pida la modificación de algún parámetro). A continuación el software subdivide la ejecución dependiendo que esté activa o no una comunicación Bluetooth;

la condición viene dada gracias a oportunos comandos del protocolo UART.
En la situación en la que no esté presente alguna conexión activa se comprueban continuamente las siguientes condiciones:

- la petición de una nueva conexión para identificar cualquier smartphone Android que esté cercano y/o haya comandos de entrada;
- se verifica que las alarmas están activas; en este caso si han pasado algunos segundos con la conexión continuamente OFF, significa que la distancia entre tarjeta y smartphone se ha convertido en superior a la zona de cobertura y por tanto es el momento de activar las alarmas;
- si la alarma está desactivada se comprueba la presión de un pulsador. En caso afirmativo para P1 (como se ha dicho anteriormente) se efectúa un breve test del zumbador; para P2, sin embargo, se incrementa secuncialmente en modo "circular" el nivel de potencia de transmisión (el nivel seleccionado se señala a través de distintos parpadeos de los LED y el zumbador: 1 parpadeo significa nivel mínimo, 7 nivel máximo).

En la situación en la que hay una conexión activa, sin embargo vienen gestionados los protocolos software de comunicación, tanto el de "bajo nivel" definido por el módulo Bluetooth RN-42, como el del "nivel superior" por definido nosotros para la gestión del envió/recepción de los comandos y control del estado de cercanía entre los dos dispositivos (en el cuadro "Protocolo de comunicación" damos los detalles de las reglas de este último protocolo).
En el firmware señalamos la presencia de distintos timeout (todos gestionados por la única interrupción del módulo hardware Timer 1): el primero se usa como una especie de "condición de bloque software" para comprobar que la comunicación Bluetooth no se está usando inútilmente por cualquier dispositivo no autorizado; de hecho en el caso en el que este temporizador se desborde la comunicación se cierra y se vuelve a las condiciones de espera de nueva conexión.
El segundo temporizador se utiliza (en caso de alarma activa) para mostrar cuanto tiempo ha pasado desde una conexión a la sucesiva; en el caso en que este tiempo sea superior a un determinado umbral, la alarma se debe disparar.



Finalmente, el último temporizador se usa simplemente para generar el parpadeo del LED verde de "sistema vivo" (esto en el caso en el cual la conexión Bluetooth esté desactivada, porque en el tiempo en el cual la conexión está activa el LED verde se mantiene encendido fijo).

SOFTWARE ANDROID

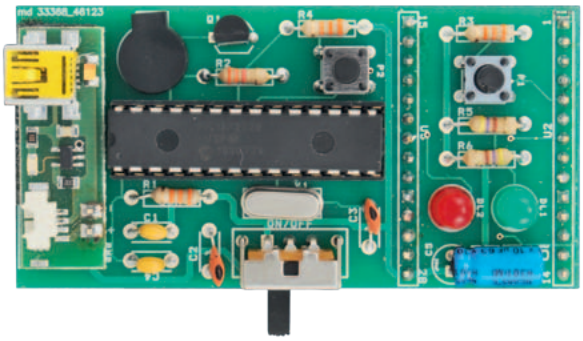
Analicemos ahora el software para smartphone Android; como ya hemos explicado ampliamente, todo el sistema se basa en el uso del Bluetooth (en particular el 2.0) para mostrar la presencia y distancia entre los dos dispositivos. Por tanto resulta obvio que la condición necesaria para funcionar es que el smartphone disponga de tal tecnología (y ahora todos los dispositivos la poseen) y que antes del arranque del software, desde los ajustes del sistema operativo el Bluetooth esté activo.
El software está constituido por una página principal donde es posible activare/desactivar las alarmas y de otras dos páginas menores (rellamada desde el menú Android) donde es posible explorar los dispositivos Bluetooth y elegir a cuál conectarse respectivamente y una página de configuración de la tarjeta (nombre y código PIN Bluetooth de la tarjeta y selección del nivel de potencia de transmisión).
Arrancado el software aparece un primer pantallazo de bienvenida (**Fig. A**) y a continuación la página de arranque y/o selección del dispositivo al que conectarse (**Fig. B**). Seleccionado el dispositivo, si es la primera vez que se conecta, aparece la pregunta de introducción del código PIN (por defecto la tarjeta es programada con 5555 pero, como veremos más adelante, esto puede cambiarse como se quiera por el usuario) necesario para el emparejamiento Bluetooth (**Fig. C**).
Después que se ha establecido la conexión, el software Android se conecta a la tarjeta, lee el estado de activación de alarmas y actualiza en consecuencia grafica de la página principal (**Fig. D** en caso de alarmas desactivas). Pulsando "Activa alarmas" se envía el comando de activación (en el display aparece un mensaje, **Fig. E**) y al ejecutar el comando la gráfica de la página es actualizada de nuevo teniendo cuenta el nuevo estado (**Fig. D** en caso de alarmas desactivada, **Fig. E** en caso de alarmas activadas).
En este punto, con las alarmas activadas, el software continua cíclicamente para conectarse a la tarjeta y envía/recibe datos como verificación de la conexión. Si esta operación falla, el smartphone nos avisa a través de vibración, alarmas sonoras y mensaje en pantalla (**Fig. G**).
Hay que señalar también que en caso de una tentativa fallada, el software continúa su intentando nuevas conexiones a intervalos regulares; si la conexión se ha restablecido lo indica cerrando la señal de alarma y si estos intentos aun fallan nos avisa de nuevo.
En caso de que la alarma esté desactiva, se habiita el menú Android de la aplicación (**Fig. H**). Para evitar falsas alarmas, hemos preferido habilitar este menú solo en el caso que la alarma esté deshabilitada. Nos parece una elección obvia porque si la alarma está activa nos parece poco probable que se decida modificar los ajustes. En el menú están presentes dos posibilidades: "Conexión" y "Settings". La primera visualiza la pantalla de arranque/selección de los dispositivos (ya visto en Fig. B) mientras la segunda abre la pantalla de ajuste (Fig. I). Como se puede observar los ajustes programables son el nombre del sistema, el código PIN y el nivel de la potencia en transmisión. Para las primeras dos están disponibles un campo de texto cada uno, mientras para la última está presente una barra de progreso para la selección. Cada uno de los 3 ajustes dispone de un pulsador de envió del nuevo valor seleccionado.
El envío de una de las tres configuraciones requiere que el módulo Bluetooth sea reseteado para hacer efectivas las modificaciones efectuadas (del software Android se muestra un mensaje de arranque, Fig. J).

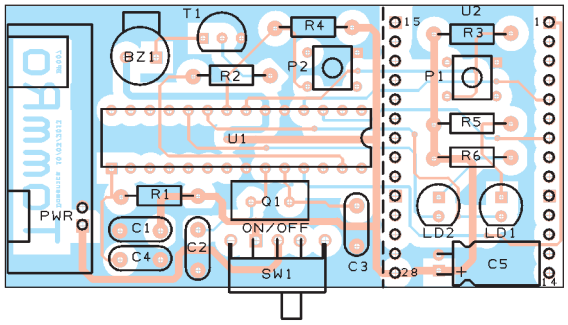
REALIZACIÓN PRÁCTICA

Nota importante para el montaje

Destacamos esta operación de montaje ya que también nosotros, durante la construcción del primer prototipo, nos hemos atascado en un error parecido que ha tenido como consecuencia la destrucción definitiva del prototipo (circuito impreso y módulo Bluetooth). El socket FT1018M es suministrado con los 4 puentes pequeños bajo el circuito impreso predispuestos pero no realizados; esto para dejar más libertad de configuración al usuario final para sus proyectos propios. En nuestro caso estos puentes deben estar cerrados (mediante pequeñas gotas de estaño) de lo contrario la configuración del módulo RN-42 no es compatible con el PIC18F2520 de la tarjeta electrónica.
Durante el montaje del primer prototipo de la tarjeta nos olvidamos de ello y de soldamos los 28 pin de la tarjeta; cuando nos dimos cuenta del error ya era demasiado tarde y no hemos podido desoldar el módulo sin estropear las pistas de la tarjeta y como resultado la hemos estropeado definitivamente y la hemos tenido que tirar a la basura todo.

[plano de **MONTAJE**]





Lista de materiales:

R1, R3, R4: 33 kohm
R2: 3,3 kohm
R5, R6: 470 ohm
C1, C4: 100 nF multicapa
C2, C3: 15 pF cerámico
C5: 10 μF 63 VL electrolítico
Q1: Cuarzo 4 MHz

U1: PIC18F2520 (MF1084)
U2: Módulo RN-42 (FT1018M)
T1: BC547
P1, P2: Microinterruptor
LD1: LED 5 mm verde
LD2: LED 5 mm rojo
BZ1: Zumbador 3,6V

SW1: Conmutador deslizante 90° PCB

Varios:
- Módulo de alimentación FT864M
- Zócalo 14+14
- Tira de 2 pines macho 2,54 mm
- Circuito impreso

El sistema presentado ha sido diseñado utilizando solo componentes de montaje convencional, de taladro pasantes (y por tanto realizable fácilmente disponiendo de un soldador); el único componente en SMD es la placa adaptadoraRN-42 ,que aconsejamos adquirir ya montado y comprobada en la tienda de Nueva Electrónica (código de articulo FT1018M).
Como se puede ver en las imágenes presentes en el artículo (que por claridad aconsejamos consultar continuamente durante el montaje), todos los componentes van montados por la cara de componentes con excepción del módulo individual Bluetooth que debe montarse por la cara de soldadura (y por tanto en orden temporal será montado el último).
Después habernos procurado el circuito impreso y todos los componentes, comenzaremos soldando los componentes más pequeños (resistencias, condensadores, pulsadores, interruptor de encendido y LED, posiblemente en este orden).
Dado que al final la placa se alojará en el interior de una caja de plástico, hay realizar el montaje intentando mantener la altura mínima posible. Intentar montar las resistencias y los condensadores (todos los componentes en general) pegados lo máximo posible al circuito impreso; en particular aconsejamos montar el condensador C5 doblado 90° (es decir, en posición horizontal); señalamos además que, siempre por el mismo motivo, hemos elegido un cuarzo no tradicional, de altura reducida.
Terminado el montaje de estos componentes, es el momento de montar el PIC; en nuestro ejemplo, por comodidad, hemos elegido utilizar un zócalo; en vuestro caso debéis elegir vosotros entre utilizarlo o soldar directamente el PIC (en este caso no será ya posible, o muy difícil, actualizar/programar nuevamente su firmware).
A continuación consideramos el cargador integrado: aconsejamos montarlo con una tira de 2 pines de paso 2,54mm. Aconsejamos además, para evitar inútiles oscilaciones del módulo, pegar sobre sobre el circuito impreso (en la zona bajo este componente) de una esponja de tal manera que se amortigüen los movimientos.
Como primer paso soldaremos la tira de pines de dos polos sobre el impreso; después pegaremos la esponja; a continación insertaremos el módulo cargador en los dos polos y soldamos la tira de pines sobre el lado superior del cargador.
Terminaremos el montaje soldando el módulo Bluetooth. Antes sin embargo hay que recordar realizar bajo el mismo módulo las 4 soldaduras relativas a los 4 puentes "LED STATO", "LED CONNECT", "RESET" y "BAUD 9600" sino el módulo RN-42 y el PIC no se "hablaran"; además el PIC durante la programación no conseguirá resetear el Bluetooth (para los detalles referidos al primer artículo relativo al Bluetooth en el cual mostramos y describimos el esquema del socket).
El módulo Bluetooth será posicionado sobre la cara de soldadura teniendo en cuenta la numeración de los pines (para evitar errores fijaros en las imágenes del articulo).
Las soldaduras se harán sobre la

Tile, encontrar los objetos perdidos con una App

Tile incorpora un pequeño zumbador que nos ayuda a encontrarlo en ambientes cerrados; la batería no puede ser sustituida y tiene una duración de un año, después de este es necesario adquirir otro Tile cuyo precio es de 18,95 dólares (alrededor de 15 euros).
La aplicación, ideada por Nick Evans y Mike Farley, ha sido financiada por una campaña de crowdfunding que ha visto la adhesión de 49.586 personas para una financiación total de 2.681.297 dólares. En este caso no ha sido utilizada una de las típicas plataformas de crowdfunding (Kickstarter, Indiegogo, etc.) si no Selfstarter (http://selfstarter.us/), una solución open source para cargar sobre el propio server para gestionar en completa autonomía la propia campaña.

Ni haciéndolo aposta, una idea parecida a la de este proyecto es la base de uno de los más grandes sucesos de crowdfunding de los últimos meses: el proyecto Tile (www.thetileapp.com), que consiste en un tag miniatura con tecnología Bluetooth 4.0 completo a batería con autonomía de un año y en una App que trabaja en iPhone 4S, iPhone 5, iPad mini, iPad de tercera y cuarta generación y iPod Touch de quinta generación.
El pequeño tag se puede fijar a cualquier objeto, llaveros, monederos, maletas, bicicletas, etcétera, mientras la App nos permite localizar el lugar en el cual hemos dejado el objeto. Todo funciona dentro de un radio de 15-45 metros y con un máximo de 10 tag asociados a cada móvil. Aun si el objeto ha sido robado, podemos habilitar a nuestros amigos para la búsqueda del tag, y por lo tanto del objeto, de tal manera que el radio de búsqueda se puede expandir como se quiera.

cara de componentes; al no haber en las cercanías componentes, la operación resulta bastante ágil.
Finalmente, insertaremos la tarjeta en la caja de plástico. Para evitar que el hardware sufra golpes inútiles, aconsejamos recortar de la esponja tanto las formas del circuito impreso como de la caja: insertarla en la caja de manera "cojín" para la tarjeta (avisamos que es mejor abundar en algún milímetro, ya que una vez que la tarjeta esta insertada difícilmente se moverá).
La tarjeta ha sido proyectada teniendo cuenta de las dimensiones de la batería recargable utilizada. Como se ve en las fotos publicadas, sobre el lado de soldaduras el módulo RN-42 se ha colocado lo más exterior posible cercano a un borde, dejando cerca del lado opuesto el espacio necesario para contener la batería que, teniendo altura parecida al componente Bluetooth, creara con esto un conjunto perfecto generando un borde inferior prácticamente plano.
La batería debe estar posicionada bajo la tarjeta (sobre el lado de soldaduras donde están presentes las partes terminales de las soldaduras mismas); además también inserta en la pequeña caja. Con el tiempo las partes terminales de las soldaduras podrían generar pequeños arañazos que podrían crear cortocircuitos y estropear entonces la electrónica. Sugerimos por tanto aislar el lado de la batería que estará en contacto con la tarjeta pegando encima una etiqueta de papel bastante gruesa (también se pueden superponer más etiquetas).
Una vez que tarjeta y batería están en el interior de la caja, cerramos la cubierta con el tornillo. Por comodidad, en correspondencia de los LED y del interruptor, es posible hacer unos agujeros o aberturas en el contenedor de plástico haciendo accesible y visible sin tener que quitar cada vez la cubierta superior. Esto mismo sirve para la toma USB de carga de batería, también si esta se utilizará menos frecuentemente y entonces podría no ser necesario.
Finalmente, si queréis aumentar la potencia sonora de la alarma, realizar una apertura en correspondencia de la salida del zumbador.

PRUEBA

Terminado el montaje del hardware, veamos como comprobar el sistema y conectar la electrónica al software Android.
Primero de todo aconsejamos conectar la batería al conector correspondiente del módulo cargador y dejar cargando el sistema al menos u de 15 minutos (mejor hasta la carga completa; señalamos además que el conector de la batería tiene posición, así que si no entrara en el correspondiente conector del cargador, no forcéis nunca la inser-

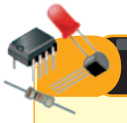
ción y probad a girarlo 180°).
Para encender el sistema mover el interruptor de la tarjeta de OFF a ON (ver serigrafía del circuito impreso) y verificar que el LED verde se enciende o parpadea señalando así que el software está activo.
Ahora ya podéis efectuar un primer test a través de los dos pulsadores: pulsar P1 y verificar que se activa el zumbador y que los LED respondan correctamente; pulsar P2 y verificar que sea reconocido correctamente y que los LED/zumbador señalen el paso de un nivel de potencia de transmisión al siguiente.
Si todo está correcto, pasar al smartphone: lo primero es ir a la página de configuración Wifi Android, sección Bluetooth y iniciar un análisis. Verificar que la tarjeta se detecta (nombre por defecto: "Scheda Allarme") y si queréis probar a ejecutar el acoplamiento (PIN por defecto: 5555).
Después arrancar el software *AntiTheftControl* que habréis descargado de nuestra web *www.nuevaelectronica.com* e instalado en vuestro smartphone; desde la sección de conexión ejecutar una exploración de los dispositivos Bluetooth activos y seleccionar vuestra tarjeta. Esperar algún segundo para la conexión y verificar que se active el pulsador "Activa alarma", y confirma que la primera conexión se ha realizado correctamente y que actualmente las alarmas están desactivadas.
Si queréis, a través del menú podéis visualizar la página de configuración y modificar las configuraciones según vuestras necesidades. **IMPORTANTE: Cuidado si modificáis el PIN, porque si os equivocáis al escribirlo y/o lo olvidáis, después no podréis usar más la tarjeta (a menos que dispongáis de un programador de PIC para reprogramarlo; sino tendréis que volver a pedir uno nuevo).**

De vuelta a la página principal, activar las alarmas y comprobar que se enciende el LED rojo. Probar a moveros en la habitación y en el espacio alrededor y comprobar que el enlace este siempre activo y no haya falsas alarmas.
Después probar una alarma. Si no queréis alejaros demasiado, haced esto: cerrar el software Android y comprobar que después de un poco de tiempo la tarjeta activa el zumbador al establecer la condición de alarma. Reiniciar nuevamente el software, conectaros nuevamente a la tarjeta y comprobar que cuando la conexión es restablecida las alarmas se paran.
Ahora apagad la tarjeta y comprobad que en este caso dispara las alarmas en el smartphone.
También en este caso probad a encender de nuevo la tarjeta y las alarmas del smartphone deben cesar.
Antes de concluir el test, acordaos de desactivar las alarmas, sino en el próximo arranque se activaran en automático y la tarjeta activará el zumbador.
Para concluir, debemos señalar que el sistema, al funcionar con batería, tiene una autonomía limitada. Por nuestras pruebas podemos decir que esta es más que suficiente para el uso normal al cual el sistema es destinado; claramente hemos utilizado una batería nueva y correctamente recargada. Probablemente con el paso del tiempo las prestaciones de la batería empeoran; si haces un uso intensivo del sistema aconsejamos sustituir la batería con una nueva cada cierto tiempo.

POSIBLES MEJORAS

El proyecto presentado en estas páginas permite gestionar un único periférico Bluetooth a la vez. En efecto, sin embargo adquiriendo más kits, si podréis pensar en modificar el software de manera que desde un único smartphone sea posible controlar distintos módulos insertados en el interior de distintos objetos.
El sistema de desarrollo Android permite gestionar varios enlaces Bluetooth hacía distintos dispositivos o - y en nuestra opinión más simple de implementar - gestionar una sola conexión a la vez y recorrer secuencialmente las distintas tarjetas.
Eligiendo este segundo camino, el firmware del PIC podría no necesitar modificación alguna (como mucho un aumento mínimo del timeout de notificación de alarmas para darse cuenta que el smartphone tiene más periféricos a controlar y por tanto podría pasar algún momento más antes de que nuevamente termine el ciclo); el software Android sin embargo si habría que modificarlo, pero no mucho.
Por el momento hemos decidido dejar esta tarea a algún lector voluntario; hacednos saber de todas formas si os puede interesar el desarrollo, podríamos volver sobre este argumento.

(179049)



el MATERIAL

Todos los componentes utilizados en este proyecto son fáciles de encontrar. El master del circuito impreso puede descargarse de la web de la revista así como el firmware utilizado para programar el microcontrolador PIC18F2520 y el software para el móvil Android. El módulo Bluetooth con RN-42 (cod. FT1018M) cuesta 29,00 Euros, mientras el módulo de alimentación FT864M cuesta 15,00 Euros.

Precios IVA incluido sin gastos de envío.
Puede hacer su pedido en:
www.nuevaelectronica.com
pedidos@nuevaelectronica.com