

# Alarma Hogareña con tecnología Inalámbrica basada en IEEE 802.15.4

Aguirre, Matias – Diedrichs, Ana – Egea, Mariano – Godoy, Pablo – Mercado, Gustavo  
Grupo en Redes de Sensores Inalámbricos – Facultad Regional Mendoza – UTN  
{pablo.godoy, matias.aguirre, ana.diedrichs, mariano.egea, gustavo.mercado}@gridtics.frm.utn.edu.ar

Ejes Temáticos 1 y 4: Aplicaciones en Comunicaciones - Aplicaciones en dispositivos hogareños

Palabras Clave: Alarma, Domótica, PAN, Redes de Sensores, Redes inalámbricas, 802.15.4

Se ha desarrollado una alarma hogareña, donde los sensores están conectados en forma inalámbrica a través de radios, bajo la norma IEEE 802.15.4 [1], donde se utilizó una pila de protocolos denominada SMAC (Simple MAC) de la empresa FreeScale. Una de las ventajas de estas tecnologías es el bajo consumo, pero se requiere un diseño apropiado tanto de hardware como de software para lograr un funcionamiento “a pilas” de hasta 2 años, como se demuestra en el trabajo. La meta a futuro es utilizar el estándar ZigBee.

El trabajo presentado corresponde a actividades del grupo WSNG (Wireless Sensors Network Group), dependiente del gridTics, de la UTN – Facultad Regional Mendoza. El grupo está compuesto por docentes, alumnos y graduados de las carreras de Ingeniería en Sistemas e Ingeniería Electrónica y tiene como objeto hacer investigación, desarrollo e innovación en redes de sensores inalámbricos y promover el interés por el desarrollo de nuevas tecnologías.

## Introducción

Se ha desarrollado una alarma hogareña [2] de tipo inalámbrica por medio de radios cumpliendo el estándar IEEE 802.15.4. Esta al igual que cualquier alarma hogareña consta de una interfaz con el usuario por medio de displays y teclado. Posee la capacidad de dividir a los sensores por zonas, dando una versatilidad extra como por ejemplo la de una o más zonas temporizadas.

El sistema de alarma inalámbrico se desarrolló bajo las características de topología “maestro-esclavo”. El nodo maestro es el órgano de control general

que consta de una interfaz humana provista de una pantalla LCD y un teclado. Los nodos esclavos son los sensores, sirenas, actuadores, entre otros.

En el desarrollo, se buscó tener a los Nodos Esclavos dormidos y en modo IDLE (radio apagada) la mayor parte del tiempo, reduciendo los tiempos de transmisión y recepción.

La implementación se realizó en lenguaje “C” usando la librería gratuita SMAC de Freescale como “controlador de la radio” o driver, la que incluye primitivas que solucionan el acceso al medio, permiten la selección del canal de comunicaciones

y realizar diferentes pruebas tanto de enlace como a la radio misma. [3]

### Análisis de la topología

Para realizar el esquema Maestro-Eslavo se adopta una topología 'Estrella', ubicando al Maestro en el centro de la misma y a los demás Nodos o Sensores en extremos.

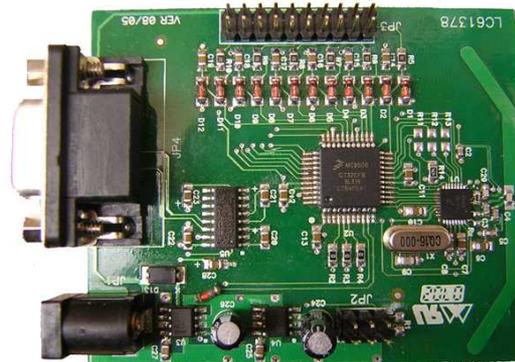
El esquema de comunicación elegido es un enfoque "maestro pasivo", donde los nodos esclavos son los que acusan los eventos al maestro. Al disminuir el tiempo de envío del mensaje de estado lo suficiente, los nodos permanecerán en bajo consumo el mayor tiempo posible. Esto implica que el transceptor está activo sólo una vez por operación, lo cual reduce al mínimo su uso. En cambio un esquema "maestro activo" (pooling) hubiese requerido que los nodos esclavos permanecieran encendidos durante más tiempo que el necesario, lo que implica un gasto energético mayor por parte del nodo, además de la sobrecarga operacional por el continuo sondeo realizado por el maestro.

Para la correcta comunicación entre los nodos esclavos y el maestro se desarrolló un protocolo de sincronización basado en TDMA (TDMA – Time Division Multiple Access), sin el uso de balizas (beaconless). Se admiten ventanas de tiempo de breve duración en las cuales el nodo primero transmite un dato, y luego enciende su receptor durante un breve tiempo para recibir cualquier dato que el nodo maestro envíe hacia él. Es decir, la ventana de tiempo asignada a cada nodo se subdivide en dos: una ventana de transmisión seguida de una ventana de recepción. A tales efectos se ha desarrollado un protocolo de sincronía que se explica en las siguientes secciones.

### Placas de Desarrollo

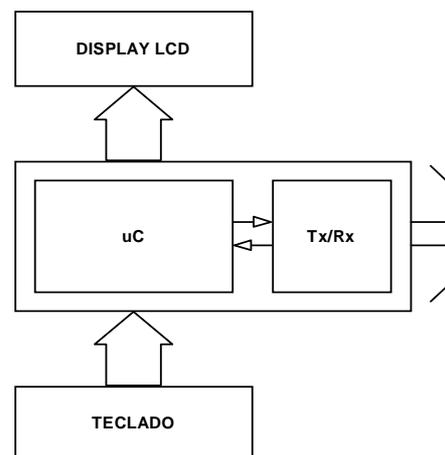
La placa que se usó fue el módulo ANT-Zigbee de Electrocomponentes S.A., el cual cuenta con un integrado microcontrolador MC9S08GT32CFB y

conectada al SCI una radio MC13192, también una interfaz de puerto serie SP232 y dos reguladores de tensión de 5 V y 3.3 V principalmente. Para la conexión con el exterior se dispone de un conector DB-9 conectado al SPI a través del SP232 y un conector de 20 pines correspondientes a los puertos A, B y C del microcontrolador. La siguiente es una imagen de las mismas:



### Nodo Maestro

El nodo maestro o maestro simplemente, incorpora una interfaz de usuario, compuesta por un display LCD y un teclado numérico. A continuación se ve un esquema simplificado.



El Nodo Maestro maneja todas las funcionalidades básicas de una alarma hogareña, como control de contraseña y manejo de activación o desactivación de zonas, detección de nodos ausentes y activación o desactivación total.

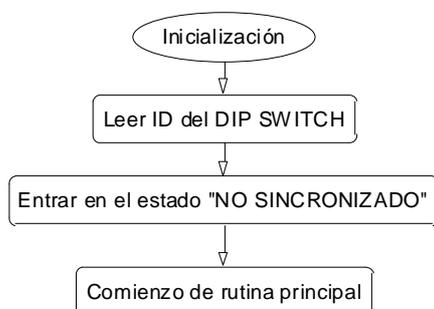
El concepto de Zona es necesario para permitir que el usuario pueda activar o

desactivar la alarma en determinadas áreas del establecimiento o por determinadas circunstancias.

### Nodo esclavo

El nodo esclavo permanece dormido hasta el momento de transmitir su estado. Inmediatamente después pasa al modo de bajo consumo. El nodo puede pasar a estado "no sincronizado" cuando haya pasado un tiempo considerable en que no ha recibido comunicación con el maestro en su ventana. En cada ventana de recepción se recibe un mensaje del maestro con información de sincronización. En este último caso el receptor estará encendido continuamente en modo promiscuo a la escucha de un paquete del maestro para extraer el reloj (ver sincronismo).

Como se observa en el diagrama, los nodos leen su identificación desde un *dip switch* y esta es almacenada en una variable *nodeID* la cual tendrá en el momento a transmitir una notable importancia.



El número de Zona esta dispuesto en la identificación del nodo esclavo como se ve a continuación:

7	6	5	4	3	2	1	0
I	T	T	Z	Z	N	N	N

Identificación de nodo

Esta tabla nos muestra la distribución en el *dip switch* para los distintos bits, donde:

**ZZ:** Zona a la que pertenece el esclavo

**NNN:** Identificación del nodo esclavo

**I:** 1 Disparo instantáneo, 0 disparo temporizado.

**TT:** Tipo de nodo

00	Nodo de salida
01	Nodo de entrada
10	Nodo Especial
11	<b>NO IMPLEMENTADO</b>

Combinando ZZ y NNN se obtienen 32 posibles direcciones, divididas en cuatro zonas para los nodos esclavos.

### Las Ranuras de Tiempo

A fin de establecer un orden en las comunicaciones del sistema, pensando en el método maestro-pasivo (beaconless), se implementó el esquema de ranuras de tiempo, programando un reloj por software en cada nodo (tanto en los nodos esclavos como en el nodo maestro). Este reloj lleva la cuenta del tiempo en milisegundos y es incrementado mediante una interrupción. La siguiente ecuación indica en que momento un nodo, identificado por su *nodeID*, puede transmitir y recibir:

$$t_A = \underbrace{\text{nodeID} \cdot t_w}_{\text{Constante}} + t_o$$

#### Referencias

- *nodeID*: Identificador del nodo.
- $t_w$ : duración de la ventana de tiempo (incluyendo transmisión, recepción y margen de error).
- $t_o$ : comienzo de la ventana del nodo cero.

El comienzo de la ventana de tiempo  $t_A$  depende de *nodeID*, por lo que *nodeID* da la posición de la ventana de tiempo correspondiente a cada nodo. Se llama Superventana al conjunto de todas las ventanas de tiempo, del nodo 1 al 32, y su duración es:

$$T = n \cdot t_w$$

Donde  $n$  es el número máximo de nodos y la constante  $t_w$  corresponde al tiempo que emplea una ventana (incluyendo transmisión, recepción y un margen de seguridad) y para un nodo cualquiera vale:

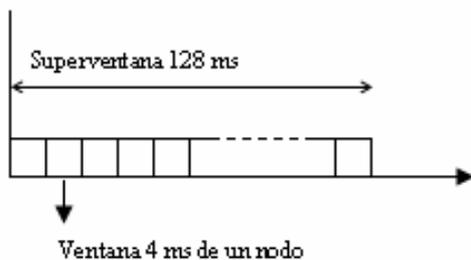
$$t_w = t_{TX} + t_{RX} + t_{MD}$$

Referencias

- $t_{TX}$  : tiempo de transmisión.
- $t_{RX}$  : tiempo de recepción.
- $t_{MD}$  : tiempo de margen de deriva.

Se determinó que la duración de cada ventana de tiempo debe ser de 4 milisegundos. Dicho tiempo es suficiente para que el esclavo le transmita su reporte al nodo maestro, y que el nodo maestro envíe los comandos necesarios al nodo esclavo. Menor tiempo podría provocar que el nodo esclavo no reciba el mensaje completo desde el nodo maestro. Más tiempo implicaría que el receptor del nodo esclavo estaría encendido más que el tiempo necesario, produciendo un consumo de energía innecesario y muy problemático, ya que el uso eficiente de energía es un parámetro clave en este proyecto.

Esto implica que la duración de la superventana completa (32 nodos en esta implementación) es de 128 milisegundos. Sin embargo, no es necesario que cada nodo se reporte cada 128 milisegundos, ya que eso implica derroche de energía. Los nodos no se reportarán en cada superventana, sino que se reportarán arbitrariamente cada 8 superventanas, por lo que cada esclavo se comunicará con el maestro cada 1 segundo aproximadamente (*tiempo entre reportes = 1028 mseg*), que es un tiempo aceptable para el correcto funcionamiento del sistema. Se denomina *superventana útil* a la superventana en la que hay comunicación entre nodos esclavos y el master.



Es importante destacar el hecho de que son los nodos quienes abren su ventana

de tiempo cuando les toca transmitir, para luego recibir y no el maestro. Por eso es importante que los relojes de los nodos se encuentren sincronizados en todo momento, para evitar que dos nodos intenten transmitir en un mismo instante.

### Sincronización

Al momento de la sincronización el nodo maestro envía su propio reloj, y el esclavo al recibir éste, si existe diferencia, toma como correcto el valor del reloj del nodo maestro. Por lo tanto es el reloj del nodo maestro, el que se asume como patrón y es frecuentemente transmitido a los nodos esclavos en sus ventanas para que estos corrijan cualquier corrimiento de su reloj (*deriva* comúnmente en electrónica).

El maestro mantiene un buffer de paquetes a enviar a cada esclavo. En la ventana de cada nodo, primero el maestro recibe los datos del esclavo y luego le envía los paquetes del buffer, incluido la sincronización. En caso de no haber datos a enviar, se envía información de sincronización. Esto simplifica la implementación y no hace falta que el nodo maestro envíe ningún tipo de señalización (como los "beacon" utilizados por el protocolo IEEE 802.11).

Si un nodo pierde por algún motivo la sincronía con el sistema, perderá comunicación con el nodo maestro. Cuando el nodo detecte esta circunstancia, pasará al estado denominado "No sincronizado", en el cual enciende su receptor de forma continua, para escuchar cualquier paquete de datos enviado a cualquier nodo, recuperando del mismo el reloj del sistema. Una vez recuperada la sincronía, el nodo solo encenderá su transmisor o receptor en la ventana de tiempo que le corresponde.

Cuando un nodo esclavo comienza su funcionamiento, que puede ocurrir cuando se adiciona al sistema o se recupera de un inconveniente, se coloca en el estado "No sincronizado", y espera de

este modo recuperar el reloj de alguna transmisión del nodo maestro.

### Reconocimiento de nodos

El maestro lleva una lista de los nodos presentes. Cuando arranca el maestro lo primero que realiza para sincronizar a los nodos esclavos es enviar una ráfaga de paquetes que contienen el reloj del maestro. En un estado inicial todos los nodos esclavos estarán escuchando, por lo que tomarán de la baliza inicial el reloj del maestro. Una vez sincronizados los esclavos se reportan, de esta manera el maestro los reconoce y agrega a la lista. Si el maestro recibe un reporte de estado de un nodo que no estaba presente, lo añade a la lista.

### Trama completa

Como puede verse la trama está comprendida por un header, el reloj y una pila de comandos, siendo como máximo cinco.

HEADER	1er BYTE
RELOJ 1	2do BYTE
RELOJ 2	3er BYTE
RELOJ 3	4to BYTE
RELOJ 4	5to BYTE
COMANDO 1	6to BYTE
...	...
COMANDO 5	10º BYTE

El esclavo forma una pila con los comandos recibidos que se ejecutarán secuencialmente luego de cerrar la ventana. Actualmente, los comandos que el nodo maestro envía a los esclavos son dos. Uno es el de “activar o desactivar”. Si el nodo está activo, ante la presencia de un intruso, se dispara la alarma. Es importante destacar que el estado “desactivado” no implica que el nodo esté apagado, el nodo siempre está encendido y reportando su presencia en la red.

El segundo comando es el de “Activar / Desactivar salidas”, que se le envía a los nodos de tipo “salida”. Es importante mencionar que el sistema se ha diseñado

de forma tal que pueda ampliarse fácilmente, de manera que en un futuro, puedan añadirse otras prestaciones, como ser detección y actuación en caso de presencia de fuego, de monóxido de carbono, etc. (nodos especiales, ver **TT** tipo de nodo).

El “Header” siempre tiene el mismo formato, para el caso del Nodo Maestro o Esclavo, solo difiere en el caso que sea el Maestro quien esté transmitiendo, el campo **TT** (Tipo de Nodo) no es completado.

<b>7</b>	<b>6</b>	<b>5</b>	<b>4</b>	<b>3</b>	<b>2</b>	<b>1</b>	<b>0</b>
M/ S	T	T	Z	Z	N	N	N

**M / S:** 1: destino Master; 0: destino Slave

### Reporte del nodo esclavo

Debido a que cada nodo esclavo debe transmitir su identificador más su estado, se toma una longitud de trama de 2 bytes para la transmisión de datos desde los nodos esclavos al maestro. En cambio, el maestro transmitirá a los nodos esclavos un paquete de 10 bytes, conformado por un identificador de nodo esclavo, el reloj del sistema y comandos.

En cada transmisión el esclavo envía la información de sus entradas/salidas al maestro.

### Consumo de energía

Se realiza un cálculo del consumo promedio en *mA* del nodo debido al uso de la radio. Conociendo este valor y el consumo de los demás periféricos que componen el nodo, se puede hacer el cálculo del tiempo de vida estimativo con dos pilas AA comunes. El tiempo de transmisión se calcula como sigue:

$$\text{Tiempo paquete} = (\text{preámbulo} + \text{sfd} + \text{phr} + \text{tamaño paquete de datos} + \text{checksum}) * 8 \text{ bit} * 1/250$$

*(tiempo de un bit en mseg)*

Donde *sfd*, *phr* y *checksum* corresponde a la norma IEEE 802.15.4. [1]

$$\text{Tiempo paquete} = 0,6 \text{ mseg}$$

$$t_w = t_{TX} + t_{RX} + t_{MD}$$

$$t_w = (0,6 + 0,6 + 2,8)mseg = 4 mseg$$

Esto justifica un tamaño total de la ventana 4 mseg.

De las especificaciones de la radio (MC13192) se obtienen los siguientes datos sobre el consumo: [4]

En transmisión: 35mA

En recepción: 42mA

En modo Hibernate: 6uA máximo

Partiendo del *tiempo entre reportes*, el cual es aproximadamente 1 seg y que la radio está en uso 1,2 mseg, el consumo promedio es (duty cycle):

$$I_{AV-R} = 6 \mu A + (35mA * 0,6mseg + 42mA * 0,6mseg) / 1000 mseg$$

$$I_{AV-R} = 52,2 \mu A$$

El microcontrolador en ejecución consume 6 mA, en modo Stop3 8 uA y con el mismo duty cycle que la radio aproximadamente y despreciando el resto de los componentes, el consumo promedio será: [5]

$$I_{AV-M} = (6mA * 1,2mseg) / 1000 mseg + 8 \mu A$$

$$I_{AV-M} = 15,2 \mu A$$

$$I_{AV-Total} = 52,2 \mu A + 15,2 \mu A$$

$$I_{AV-Total} = 67,4 \mu A$$

Suponiendo pilas AA con capacidad de 1500 mAh (generalmente la capacidad es mayor [6]), podemos calcular el siguiente tiempo de vida:

$$Tiempo\ de\ vida = 1500mAh / 67,4 \mu A$$

$$Tiempo\ de\ vida = 2,5 años$$

## Conclusiones

Se realizó el diseño e implementación de un dispositivo domótico, en donde la principal cualidad es el método de acceso y comunicación optimizado para mantener bajo el consumo. En futuros trabajos se ampliará el sistema a la norma ZigBee.

## Referencias

- [1] IEEE Std 802.15.4a-2007  
www.ieee.org
- [2] Aguirre M, Egea M, Godoy P, Martínez J, Generale S, Mercado G y Robles J, "Nota de Aplicación NA036: Alarma Anti – Robo inalámbrica ZIG BEE con transceiver MC 13192 contenido en módulo de radio (ANT – ZigBee)", Electrocomponentes S.A., Dic 08 (wsng.gridtics.frm.utn.edu.ar)
- [3] SMAC User's Guide, SMACRM/D Rev. 1.0, 07/2004
- [4] MC13192/MC13193 Reference Manual, Rev. 1.5
- [5] MC9S08GB/GT Data Sheet, Rev. 2.3
- [6] Hojas de datos pilas AA LR6 Energizer (<http://data.energizer.com/PDFs/E91.pdf>)