# Capítulo 5

# Fase de pruebas

Se especifican y detallan los distintos controles que se han realizado al sistema implementado, con el fin de comprobar su correcto funcionamiento.

Primeramente se enumeran distintos mecanismos para enviar o recibir información de los sensores, con el fin de poder depurar el código implementado con facilidad. El siguiente bloque son comprobaciones del código de programa, con especial atención a medidas del convertidor analógico-digital, cálculos con ondas de tensión e intensidad, y medida de armónicos.

Después se pasa a las medidas con señales reales, simulando las que se recibirían de una toma eléctrica. Se describen dos procedimientos para llevar a cabo estas pruebas.

El último bloque se centra en las comunicaciones: recepción de datos desde la red y envío de comandos hacia los nodos (con y sin pasarela de datos). Se agrega el monitor desarrollado en MATLAB y se realiza una prueba de conjunto.

## 5.1 Envío y recepción de información

Tanto a la hora de realizar pruebas de los dispositivos, como para recoger/enviar datos, es necesario disponer de una serie de herramientas para tal fin.

#### 5.1.1 Simulador TinyOS

Denominado TOSSIM<sup>54</sup>, permite ejecutar un código compilado tal y como lo haría un sensor inalámbrico. Incluso se puede definir una red con varios nodos, potencia de señal, ruido... Sin embargo, la única plataforma soportada actualmente por este simulador es micaz (y no Telosb / Tmote Sky), por tanto, no podrá usarse en este trabajo.

## 5.1.2 Conexión JTAG<sup>55</sup>

Se trata de una interfaz ampliamente usada para propósitos de testeo y monitorización de programas. El mote incluye un conector (U8, Figura 2.2) para este fin con los terminales TDO (salida de datos de testeo), TDI (entrada datos), TMS (modo de testeo), TCK (reloj de testeo), RESET (igual que el terminal de la CPU) y DVCC (tensión alimentación). No se ha utilizado pues es necesario el dispositivo de interfaz entre el mote (JTAG) y el sistema de monitorización.

#### 5.1.3 Biblioteca Printf

Una de las formas más sencillas para conocer el valor de variables o resultados es mostrarlos por pantalla. La librería de TinyOS denominada Printf ofrece esta funcionalidad.

Para hacer uso de la librería Printf, es necesario modificar el fichero Makefile, indicando la inclusión de dicha librería:

<sup>&</sup>lt;sup>54</sup> TinyOS SIMulator.

<sup>&</sup>lt;sup>55</sup> Acrónimo de Join Test Action Group (norma IEEE 1149.1).

CFLAGS += -I\$(TOSDIR)/lib/printf

Además, en la aplicación en la que se vaya a usar, hay que añadir el fichero de cabecera:

#include "printf.h"

Con esto, ya es posible usar la librería. Por ejemplo, si se tienen las variables vmed e imed, sin signo y de tamaño 32 b, se puede hacer

```
printf("\nVmedia = %lu\tImedia = %lu",vmed,imed);
printfflush();
```

Dando como resultado (en pantalla)

Vmedia = 2540 Imedia = 1023

El delimitador \n provoca un salto de línea, mientras que \t inserta una tabulación; %1u indica que se debe mostrar un entero sin signo y 32 b. Como se aprecia, tiene mucha similitud con la función printf de lenguaje C.

El comando printfflush obliga al buffer de mensajes a enviar los datos que se tengan (sin esperar a que éste se llene).

Para ver estos comandos a través de la interfaz (línea de comandos) del emulador Cygwin, hay que ejecutar

java net.tinyos.tools.PrintfClient -comm serial@COMx:telosb

En COMx hay que indicar el puerto en el cual está conectado el sensor, por ejemplo COM4. Una vez se ejecute el comando, el terminal quedará a la espera de recibir mensajes desde el nodo.

En la aplicación se han creado algunas macros para facilitar la labor de depuración. Por ejemplo, usando la macro perror(cadena):

```
perror("Arranque del sistema radio");
```

Aparecerá en pantalla como

ERROR. Arranque del sistema radio

Estas macros pueden usarse si a la hora de compilar el código se añade la bandera DEBUGMODE, indicando así que se quieren recibir los avisos. En caso contrario, estos comandos no tienen efecto alguno.

#### 5.1.4 Envío de mensajes propios por puerto serie

Cuando se trata de enviar paquetes por puerto serie con una estructura determinada, acorde a las necesidades del problema, es preferible el uso del componente SerialActiveMessageC (ya comentado en 4.12.1) que la biblioteca Printf.

Para recibir estos datos (y posteriormente para enviar) es muy útil un conjunto de aplicaciones desarrolladas en lenguaje C y que se compilan para funcionar bajo Cygwin. Se denominan genéricamente sf (*Serial Forwarder*)<sup>56</sup>. Las más usadas son:

- seriallisten.exe. Recogida de datos a través del puerto serie del PC. La sintaxis es ./seriallisten <dispositivo> <velocidad baud> Por ejemplo, un mote modelo Trnote Sky conectado al puerto COM3 del PC, habría que escribir ./seriallisten /dev/ttyS2 115200.
- serialsend.exe. Envío de datos hacia un nodo conectado al puerto serie.
   Sintaxis: ./serialsend <dispositivo> <velocidad baud> <bytes>
- sflisten.exe. Recepción de datos a través de una conexión TCP en un equipo remoto.

Uso: ./sflisten <dir IP> <puerto>

- Es decir, para recoger los mensajes desde la pasarela Tmote Connect, se podría usar ./sflisten 192.168.1.100 9001.
- prettylisten.exe. Es equivalente al anterior, pero además muestra información más legible para el usuario acerca de la cabecera del mensaje (origen, destino, longitud...).

Sintaxis: ./prettylisten <dir IP> <puerto>

<sup>&</sup>lt;sup>56</sup> Se deben descargar y localizar en el directorio "tinyos2.x\support\sdk\c\sf". Una vez compiladas, los ejecutables pueden llevarse a cualquier otro directorio (Google code).

• sfsend.exe. Utilidad para enviar paquetes de datos hacia el nodo (a través de la pasarela).

Uso: ./sfsend <dir IP> <puerto> <bytes>

## 5.2 Código de programa

Tal y como se comentó en 5.1, existen diversas herramientas para comprobar el estado y funcionamiento del programa. En este bloque de pruebas, bastaría con usar la biblioteca Printf, pues solo hay que mostrar algunos valores de variables y si no son demasiados, esta función debería ser suficiente.

#### 5.2.1 Convertidor analógico-digital

Para comenzar con el manejo de este componente interno, se ha diseñado y construido un regulador de tensión (basado en un divisor resistivo) para controlar la tensión que recibe la entrada analógica del sensor. En la Figura 5.1 se muestra el esquema eléctrico del mismo y en Figura 5.2 la construcción física del mismo.



Figura 5.1 Esquemático del regulador de tensión

Se utiliza un conector USB tipo A para conseguir una tensión constante de 5V aprox. El conmutador permite elegir entre dos rangos máximos de tensión. Si se sitúa en A, el máximo está en unos 1.56V, mientras que en B el máximo es de 2.5V. La configuración del convertidor permite usar referencias de tensión de 1.5V y 2.5V, de ahí esos valores máximos. El potenciómetro permite regular la tensión de salida (terminal Vo) desde 0V hasta el máximo elegido.



Figura 5.2 Regulador de tensión USB El selector permite elegir entre un valor máximo de tensión de 1.5 V o de 2.5 V. El potenciómetro giratorio modifica la tensión de salida entre 0 V y el máximo fijado.

Se trata de hacer una serie de medidas secuenciales (usando para ello un *buffer*) y realizar algunos cálculos sobre ellas, como su valor medio.

El convertidor también se usa para adquirir medidas internas del sensor, como temperatura del microcontrolador o la tensión de alimentación que recibe. Usando la función printf es sencillo comprobar que estos valores son correctos.

#### 5.2.2 Tarea de cálculo de potencias

Para comprobar el correcto funcionamiento de esta tarea (CalcPotencia, 4.23) se grabó en memoria una secuencia de valores tal y como habrían sido tomados a través del convertidor analógico-digital<sup>57</sup>.

Debe ser del mismo tipo que la variable que contiene los valores reales raw. Una posible declaración podría ser:

<sup>&</sup>lt;sup>57</sup> Para ello se ha desarrollado un script MATLAB en el cuál se pueden fijar amplitudes y desfases de los 15 primeros armónicos de las señales de tensión e intensidad. A continuación, se generan las ondas, se discretizan y cuantifican con la resolución de 12 bit para que el resultado sea lo más parecido posible al que devuelve el convertidor A-D interno.

Como se explicó, se trabaja con dos *buffer* simultáneamente. En este caso, los dos contienen los mismos datos. Los valores de ondas de tensión e intensidad deben ir alternados.

Con los cambios pertinentes (raw a rawfalse), se puede comprobar que los resultados de valores medios, eficaces, potencias... son los correctos. De esta forma, se puede generar cualquier secuencia de datos para verificar su funcionamiento<sup>58</sup>.

#### 5.2.3 Tarea de cálculo de armónicos

El primer punto a revisar es el algoritmo FFT propiamente dicho, esto es, sin preprocesar datos antes. Simplemente se le pasa una secuencia definida a la cual hay que calcular su DFT. Para compararlo, se usa MATLAB. Debido a que el mote no trabaja en ningún momento con decimales, hay que corregir la DFT calculada con MATLAB (escalado, redondeo...), para que los resultados sean comparables a los del sensor.

Cuando este punto está cubierto y comprobado su correcto funcionamiento, se pasa a añadir el preprocesamiento ya comentado: enventanado, reordenación de las muestras, escalado... Al igual que antes, se genera una secuencia determinada en MATLAB (equivaldría a la generada por el convertidor analógico-digital) y se comprueban los resultados devueltos por el mote.

Por último, se puede pasar a comprobar los cálculos posteriores basados en la DFT, tales como módulos, indicadores de armónicos o tasa de distorsión armónica total.

<sup>&</sup>lt;sup>58</sup> La configuración realizada en MATLAB para generar la señal discreta debe ser acorde a los resultados generados.

## 5.3 Mediciones con señales reales

### 5.3.1 Recogida y procesamiento de datos

Respecto a los sensores, se vuelve a utilizar la biblioteca Printf para enviar datos a través de la interfaz visual de Cygwin. Se añaden comandos printf según lo que se quiera comprobar (valores medios, eficaces, potencias...).

Por otro lado, se ha diseñado una interfaz gráfica o  $VI^{59}$  de LabVIEW, que se ha llamado tester.vi. En la Figura 5.3 y Figura 5.4 se muestra el aspecto en pantalla una vez se ejecuta el VI. Mediante este sistema se comprueban las características de las señales que recibe el sensor.



Figura 5.3 Captura de pantalla VI de comprobación (tensión)

<sup>&</sup>lt;sup>59</sup> Instrumento Virtual de LabVIEW.



Figura 5.4 Captura de pantalla VI de comprobación (potencia)

Partes fundamentales del VI:

- Frecuencia de muestreo. En este campo se fija la frecuencia a la cual la tarjeta de adquisición de datos toma los datos desde las entradas analógicas. Por defecto, está fijado a 1600 Hz, aunque el mote, debido a motivos de precisión, funciona a unos 1598.44 Hz.
- Muestras tomadas. Informa del número de muestras que se recogen en bloque. El tiempo fijado es de 80 ms, correspondiente a cuatro periodos de la señal fundamental 50 Hz. Por tanto, se tomarán 128 muestras (tanto de tensión como de intensidad).
- Lista desplegable TENSIÓN . Permite seleccionar qué se está monitorizando en el VI. Por defecto, aparece todo lo relacionado con la onda de tensión. Las otras opciones son INTENSIDAD y POTENCIA.
- CargaA y CargaB. Indicadores luminosos que muestran cuando los terminales de salida digitales han sido activados (encendidos) o desactivados (apaga-

dos). En la aplicación, se ha elegido CargaA como propósito general y CargaB indica si existe distorsión armónica.

- Botón DETENER. Para la ejecución del VI.
- Gráfica de forma de onda (tensión e intensidad). Representa la señal recibida a lo largo del tiempo. El rango de representación es precisamente de 80 ms. El indicador luminoso rojo que aparece en la gráfica indica cuándo la señal de entrada ha excedido los límites impuestos por el convertidor analógico-digital (para el modelo Tmote Sky, la señal deberá estar en el rango [0, 2.5]V).
- Gráfica de forma de onda (potencia). Representa la señal de potencia eléctrica a lo largo del tiempo, es decir, el producto de la tensión y la intensidad. Rango de representación de 80 ms. El rango en el eje de ordenadas se modifica automáticamente.
- Cuadro de información VALORES MEDIDOS (tensión e intensidad). Muestra información extraída de la forma de onda, como valores medios (componente continua) o valores eficaces (RMS).
- Cuadro de información VALORES MEDIDOS (potencia). En este caso se indican parámetros como potencias activa, aparente, reactiva o factor de potencia.
- Cuadro de información VALORES EN MOTE (tensión e intensidad). Informa de los valores que deberían de resultar de los cálculos internos del mote (se elaboran a partir de los valores medidos).
- Cuadro de información VALORES EN MOTE (potencia). De igual forma, muestra los resultados relacionados con la potencia que debe hallar el mote.
- Gráfica FFT. Elabora un espectro tal y como lo haría el mote (nótese que la primera componente es la continua y el algoritmo la devuelve multiplicada por 2, por lo hay que dividirla para que los resultados sean acordes a los del sensor). La frecuencia máxima a representar es la mitad de la fijada en el cuadro Frecuencia de muestreo. Las componentes se situarán con una resolución de 12.5 Hz. El eje de ordenadas muestra el módulo de cada una elevado al cuadrado (así resulta de los cálculos del sensor). También se muestra la tasa de distorsión armónica total (THD) y el resultado que devolvería el mote. El cursor es útil para afinar valores de componentes concretas.

#### 5.3.2 Generación de señales

Para este fin existen varias alternativas. La primera consiste en usar los generadores de ondas existentes en los Laboratorios: uno de ellos para simular una onda de tensión y el otro para la de intensidad.

La segunda opción es usar un software de edición de sonido para generar señales analógicas de audio con unas características determinadas.

#### MONTAJE LABORATORIO

Para recoger los datos y usar la interfaz de LabVIEW se utiliza la tarjeta de adquisición de datos USB-6211 (disponible en las instalaciones) (National Instruments, 2009) y (National Instruments, 2009).



Figura 5.5 Tarjeta adquisición de datos USB-6211

Cuenta con 16 entradas analógicas (single-ended) u 8 diferenciales, con una resolución de 16 bit. La frecuencia máxima de muestreo es de 250 kS/s. Dos salidas analógicas de 16 bit y 250 kS/s. 4 entradas y 4 salidas digitales.

Las conexiones en la tarjeta se realizan de esta forma

- Onda de tensión. Terminales 15 (AI 0) y 28 (AI GND).
- Onda de intensidad. Terminales 17 (AI 1) y 28 (AI GND).

Para el sensor inalámbrico, se pueden utilizar directamente las señales de los generadores de onda (tensión en canal analógico 0 e intensidad en el canal 1). Si bien antes de proceder a activar la señal, con ayuda de un osciloscopio, se deben

comprobar las características de las señales: sinusoidales, componente de continua de unos 1.25 V aproximadamente, y que no sobrepasen los límites impuestos por el convertidor A-D: [0,2.5] V. En principio, se puede fijar una frecuencia de 50 Hz. Hecho esto, se pueden activar las salidas y comenzar las pruebas. En la Figura 5.6 se muestran algunas imágenes del montaje.



Figura 5.6 Montaje Laboratorio

Algunas imágenes del montaje de prueba realizado en los Laboratorios. (A) mote en el cuál se realizan las medidas eléctricas; (B) ordenador que recibe los datos procesados por el mote (a través de Cygwin). (C) generadores de señal (uno para simular la onda de tensión y el otro para la de intensidad). (D) ordenador de monitorización con LabVIEW, conectado a la tarjeta USB-6211; (E) osciloscopio para controlar los niveles de la señal generada.

A continuación se especifican las pruebas realizadas con este equipamiento.

• Con la pestaña TENSIÓN seleccionada, se genera un tono a 50 Hz y máxima amplitud posible. Se comprueba el nivel de continua (DC), así como el valor eficaz (RMS). A continuación, modificar la amplitud de la señal y comprobar los resultados.

- El mismo proceso pero con señal de INTENSIDAD (segundo generador de onda). Se comienza con la señal al máximo de amplitud posible y después se modifica ésta.
- Turno para la pestaña POTENCIA: para ello será necesario utilizar los dos generadores simultáneamente. Se comienza con ambas señales al máximo de amplitud, para después pasar a modificar cada una de ellas por separado. En principio se comprueba la potencia activa (P) y energía eléctrica (símbolo E). Para potencia reactiva, o factor de potencia se debería modificar el desfase temporal entre las señales, lo cual es demasiado complicado hacerlo con el montaje realizado.

Las pruebas anteriores verifican parte de los cálculos relacionados con las ondas en el dominio temporal. La siguiente se centra en dominio frecuencial.

• Verificación de los armónicos. Debido a que un generador de onda no es capaz de generar un tono puro y después añadir armónicos, el procedimiento será comenzar con un tono a 50 Hz y después ir modificando la frecuencia de la señal, por ejemplo, en pasos de 50 Hz (son las frecuencias que interesan). Es igual con onda de tensión que intensidad. Ayudándose del gráfico frecuencia-módulo cuadrado, se pueden comprobar todas las frecuencias que abarca el sensor<sup>60</sup>. El cursor facilita la medida de los valores medidos. En este caso, no tendría sentido utilizar los indicadores de distorsión, ya que la señal solo contiene una frecuencia sinusoidal.

El hecho de que los generadores de onda no puedan generar armónicos sobre una onda principal, y que los dos trabajan de forma independiente (no es posible modificar el desfase entre las señales) imposibilita la realización de pruebas más avanzadas. Con el método propuesto a continuación sí es posible modificar más parámetros de las señales eléctricas.

<sup>&</sup>lt;sup>60</sup> La frecuencia máxima de trabajo está limitada por la frecuencia de muestreo, es decir, si  $F_S = 1600 \,\text{Hz}$  (aprox.), el máximo armónico que podría aparecer sería a frecuencia  $F_S/2 = 800 \,\text{Hz}$ .

#### MONTAJE USANDO AUDACITY

Se trata de aprovechar una señal de audio generada por un ordenador para adaptarla al mote. La salida de audio es en general estéreo, lo que significa que se tienen dos canales independientes. Esto se usa para transmitir por el mismo medio la señal de tensión e intensidad (antes habrá que adaptar correctamente la señal para no dañar el mote).

Será necesario un *software* que trabaje y procese ondas de sonido. El elegido es Audacity. Una vista general del programa se muestra en la Figura 5.7.



Figura 5.7 Captura de pantalla de Audacity

Aspecto del software con varias ondas de audio. En la parte superior se encuentra el menú con las distintas opciones del programa. Después, la barra de herramientas principal (controles de reproducción, utilidades de edición, volumen...). El resto son las ondas de sonido separadas por canales.

Para conseguir una señal de tipo alterna, basta seleccionar Tracks > Add New > Stereo Track. En el menú desplegable de la pista, seleccionar la opción Split Stereo Track, y así quedarán divididos los dos canales izquierdo y derecho (Figura 5.8).

🔒 Audacity							
<u>Eile E</u> dit <u>V</u>	iew T <u>r</u> ansport	<u>T</u> racks <u>G</u> enerate	Effe <u>c</u> t <u>A</u> nalyze	e <u>H</u> elp			
•				I ≩ Ø ₽ ↔ ¥	L R ■) ▼		
- 1.0	୍ଦ୍	1.0	2.0	3.0	4.0		
Tens. 50H Left, 96000H 32-bit float Mute S L L L L L L L L L L L L L L L L L L	IZ     1.0       Z     0.5       iolo     +       +     0.0-       -     -       -     -       -     -						
X Int. 50Hz Right, 96000 32-bit float Mute S	▼ 1.0 12 0.5- iolo + 0.0- R -0.5- -1.0						

Figura 5.8 Pistas de audio independientes

Permite trabajar independientemente con la onda de tensión e intensidad.

La opción para generar un tono puro está en el menú Generate > Tone... (Figura 5.9).

Tone Generator 🛛 🔀				
Waveform:	Sine 💌			
Frequency (Hz)	50			
Amplitude (0-1)	1			
Duration	000,120 seconds -			
OK Cancel				

Figura 5.9 Cuadro de diálogo generación de tono

Gracias a que los canales estéreo están separados, se puede trabajar de forma individual con cada señal. Por ejemplo, se puede añadir un desfase temporal entre ellas<sup>61</sup> (Figura 5.10).

<sup>&</sup>lt;sup>61</sup> Esto es especialmente útil para comprobar las diferencias entre la potencia activa y la aparente, cuando existen desfases provocados por cargas capacitivas o inductivas.



Figura 5.10 Ondas audio desfasadas Para desplazar temporalmente una onda de audio, se utiliza la herramienta marcada en rojo (Time Shift Tool).

La gran ventaja de usar este método para generar las señales es que es posible añadir armónicos superpuestos a la señal principal<sup>62</sup>, tal y como pasaría en la red eléctrica real. Simplemente habrá que añadir tantas pistas como armónicos se necesiten, generando después los tonos con la frecuencia necesaria. En la Figura 5.11 se muestra la onda de tensión con dos armónicos (a 100 Hz y 150 Hz). Lógicamente, es posible regular la amplitud de cada uno: al comienzo de cada pista existe un control deslizante que varía su atenuación.



Figura 5.11 Onda de audio con armónicos añadidos

<sup>&</sup>lt;sup>62</sup> Esto no sería posible con un generador de onda, pues en su salida se tiene una onda sinusoidal pura a una sola frecuencia.

La tarjeta de adquisición de datos para LabVIEW usada en este caso será el modelo USB-6009 (National Instruments, 2005).



Figura 5.12 Tarjeta adquisición de datos USB-6009

Entradas analógicas: 8 de tipo single-ended o 4 en modo diferencial. Resolución de 13 bit (single-ended) o 14 bit (diferencial) Tasa máxima de muestreo de 48 kS/s. 2 salidas analógicas con resolución de 12 bit (frecuencia de 150 Hz). 12 terminales de entrada/salida digitales configurables.

Las conexiones se configuran como sigue.

- Onda de tensión. Terminales 2 (AI 0+) y 3 (AI 0-). Modo diferencial.
- Onda de intensidad. Terminales 5 (AI 1+) y 6 (AI 1-). Modo diferencial.
- Señal digital (CargaA). Terminal 17 (P0.0).
- Señal digital (CargaB). Terminal 18 (P0.1).

A su vez, los terminales AI 0-, AI 1- y la referencia del mote (GND) están conectados entre ellos.

La conexión de audio utiliza un conector de audio estándar estéreo (*jack*) de 3.5 mm. El canal izquierdo (blanco) transporta la señal de tensión, mientras que el derecho (rojo) la de intensidad.

La referencia de tensión de la señal de audio y la referencia del mote no pueden ser la misma, pues haría que la señal generada tuviera partes negativas (recuérdese que el convertidor analógico-digital trabaja en el rango [0,2.5]V). Por ello se añade una simple pila eléctrica entre ambas referencias, de forma que se aumente la componente de continua de la señal de entrada (idealmente alrededor de los 1.25V).

En la Figura 5.13 se muestra el montaje completo del sistema de pruebas.



Figura 5.13 Montaje con Audacity

A continuación se describen los test realizados con este procedimiento.

- Seleccionando la pestaña TENSIÓN, se genera un tono puro a 50 Hz y máxima amplitud posible. Se comprueba el nivel de continua (DC), así como el valor eficaz (RMS). Pruebas modificando la amplitud de la señal (esto es, el volumen de audio).
- Se añaden a la señal armónicos, por ejemplo, de frecuencias  $100\,{\rm Hz}$  y  $150\,{\rm Hz}$ . Se modifican sus amplitudes y se comprueban los mismos niveles que antes.
- Se pasa a estudiar la señal de INTENSIDAD. El procedimiento es el mismo descrito antes: primero señal armónica pura y después añadir algunos armónicos.

Consta de las siguientes partes: ordenador de comprobación de valores con Cygwin (A); ordenador de monitorización (a través de la interfaz gráfica de LabVIEW) de la señal generada por el mismo (B); tarjeta de adquisición de datos USB-6009 (C); mote inalámbrico bajo prueba (D); y señal analógica (audio) para ondas de tensión e intensidad (E). En la imagen se recogen (A) datos sobre los primeros armónicos de las ondas. Por ejemplo, en (B) aparece la onda de intensidad con distorsión añadida.

- El punto siguiente es trabajar con POTENCIA. En este caso hay que activar ambas señales analógicas. En principio, tanto la señal de tensión como de intensidad serán tonos armónicos puros, y sin desfase temporal. Hay que comprobar valores de potencia activa (P), energía eléctrica (representada por E), potencia reactiva Q al cuadrado, o el factor de potencia (en tanto por ciento al cuadrado). Usando el control de volumen de cada canal de Audacity, se ajustan las amplitudes de las ondas de tensión e intensidad, para observar cómo se modifica la potencia.
- Para observar la variación de reactiva y del factor de potencia, se deben desfasar las ondas de tensión e intensidad, tal y como se mostraba en la Figura 5.10. Por ejemplo, si se retrasa la señal de intensidad respecto de la tensión, se estará simulando un circuito equivalente inductivo; en caso de que la intensidad esté adelantada, será de tipo capacitivo. En ambos casos, el factor de potencia deberá disminuir<sup>63</sup>.
- El siguiente paso es añadir armónicos (típicamente 100 Hz y 150 Hz) a la señal de tensión y/o de intensidad. También es posible modificar los niveles de amplitud de los mismos. Después, se pueden agregar desfases tanto a las ondas principales (50 Hz) como a los armónicos, de forma individual.

Estas pruebas cubren la parte de cálculos relacionados directamente con las ondas medidas en el dominio temporal (valores medios, eficaces, potencias...). Las que siguen se centran en valores significativos en frecuencia.

• Comienzo en onda de tensión. Partiendo de un tono puro a 50 Hz, se añade un armónico (p.ej. a 150 Hz). Con ayuda del gráfico frecuenciamódulo cuadrado, de pueden comprobar todas las frecuencias. El cursor permite ajustar y observar mejor los valores de las medidas. A su vez, se comprueba la tasa de distorsión total armónica tal y como la devuelve el mote. El indicador luminoso que aparece al lado del cuadro de THD indica cuando se está sobrepasando el umbral fijado para la tasa (en este caso está fijado al 8%, ver punto A.4.2).

<sup>&</sup>lt;sup>63</sup> Recuérdese f.d.p. =  $\cos \phi = P/S$ , luego el máximo estará en 1 (toda la potencia es activa) y el mínimo estará en 0, cuando toda la potencia sea reactiva.

- La siguiente prueba es análoga a la anterior, pero aplicada a la onda de intensidad. Primero un tono puro, y después armónicos con diferentes amplitudes. En este caso, el umbral máximo para la THD en intensidad está fijado al 50% (ver A.4.2).
- Comprobado que funcionan los cálculos de armónicos en ambos casos, se pasa al indicador luminoso CargaB, que está conectado a la salida de indicación de distorsión armónica del mote. Se comenzará usando tonos puros, por lo que no debería activarse. A continuación, se añade distorsión en tensión que supere el umbral: en un momento, debería encenderse el indicador. Una vez encendido, pasar a eliminar la distorsión: pasado un tiempo<sup>64</sup>, volverá a apagarse. Este proceso se repetirá para la onda de intensidad. Y después, en ambas ondas.

## 5.4 Comunicación inalámbrica mote-mote (PC). Recepción de datos

Evaluación de la comunicación entre uno o varios motes de medidas de datos y el mote raíz o base. Este último estará conectado (mediante su puerto USB) directamente a un PC, en el cuál se recogen los datos.

Para este fin no es estrictamente necesario que los datos leídos o tratados sean reales. Es decir, se puede tener en memoria un *buffer* de medidas, tal y como se habrían recibido desde el convertidor analógico-digital. El punto clave en esta etapa de pruebas es la comunicación entre nodos.

A la hora de monitorizar la información recibida por el nodo base, se podrían utilizar dos procedimientos fundamentalmente:

<sup>&</sup>lt;sup>64</sup> El tiempo que tarda en activarse o desactivarse el indicador de distorsión depende de los parámetros del algoritmo "leacky-bucket": límite superior (LB\_UPT), inferior (LB\_LWT) y factor de olvido (LB\_D). Véase punto 4.24.7.

- Biblioteca Printf, ya tratada con anterioridad. Es posible enviar mensajes de aviso, error... incluso valores de variables.
- Usar el componente SerialActiveMessageC, que proporciona funcionalidad para el manejo de mensajes a través del puerto serie. De hecho, para enviar (y recibir) determinados tipos de paquetes de datos, se usa este componente.

Para mostrar mensajes de información o error, se prefiere usar Printf, por su sencillez de manejo. En este caso, para mostrar por la línea de comandos de Cygwin estos mensajes, se debe usar

```
java net.tinyos.tools.PrintfClient -comm serial@COMx:telosb
```

Sin embargo, si se requiere de un paquete de datos definido y adecuado para la situación, es preferible usar directamente el componente SerialActiveMessageC. Más concretamente, el mensaje que envía la base hacia el PC a través de la conexión USB tiene la siguiente definición

```
typedef nx_struct SerialMsg {
1
     nx_uint16_t nodeid;
2
    nx_uint16_t vmedia[4];
3
    nx_uint16_t imedia[4];
4
    nx_uint32_t vrms[4];
5
    nx_uint32_t irms[4];
6
    nx_uint32_t potact[4];
7
    nx_uint16_t fdp[4];
8
9
     nx_uint32_t thd[2];
    nx_uint16_t tempvolt[2]
10
11 } SerialMsg;
```

Por ejemplo, el primer campo nodeid (2B) identifica el nodo desde el cual se han recibido los datos; el vector irms (16B) contiene cuatro valores de valor eficaz (al cuadrado) de tensión medidos, etc.

Para mostrar el contenido de este mensaje, se usan las aplicaciones sf (mencionadas en 5.1.4).

./seriallisten <dispositivo> <velocidad baud>

Los mensajes se reciben tal y como se envían por el puerto serie (en crudo, *raw*), agrupados por bytes e incluyendo la cabecera. Para la estructura anterior, el mensaje sería algo como 00 FF FF 00 00 56 00 12 00 17 03 1A 03 19 [...] 03 32 09 E5

- 00. Campo de control para los mensajes de SerialActiveMessageC (siempre el mismo valor) 1B.
- FF FF. Dirección destino del mensaje (2B).
- 00 00. Dirección del enlace que envía el mensaje (2B).
- 56. Tamaño del mensaje sin contar los campos de cabecera (1B). Se expresa en hexadecimal:  $56_h = 86_d$ , esto es, los 86B de la estructura de datos.
- 00. Identificador de grupo de mensajes (1B).
- 12. Identificador de tipo de mensaje serie (1B). Este valor se puede modificar en la constante simbólica SERIALTX\_ID. Si fueran necesarios otros tipos de mensajes, se les asocian diferentes identificadores.
- El resto del mensaje es la carga útil, esto es, los datos de la estructura SerialMsg, en el mismo orden en que están declarados: 00 17 hace referencia a nodeid (2B); 03 1A representa la primera componente (2B) del vector vmedia; 03 19 la segunda componente... 03 32 será la temperatura interna; y 09 E5 la tensión de alimentación (último valor).

En la Figura 5.14 se muestra de forma gráfica una primera configuración sencilla: un mote envía la información hacia otro, que está conectado directamente al PC de monitorización.



Figura 5.14 Configuración de red con dos nodos

El nodo A actúa como nodo raíz (o base) y está conectado al PC mediante conexión serie (USB). Por su parte, el nodo B actúa como recogedor de datos y los envía hacia A (el símbolo del rayo indica alimentación por batería). La línea discontinua indica enlace radio.

El siguiente paso es añadir otro nodo remoto (recogida de datos) a la red. De esta forma, se comprueba el correcto funcionamiento del algoritmo de colección de datos vía radio, Figura 5.15.



Figura 5.15 Configuración de red con tres nodos (I)

El nodo A actúa como nodo raíz (o base) y está conectado al PC mediante conexión serie (USB). Por otro lado, los nodos B y C actúan como sensores de recogida de datos (remotos). El rayo rojo indica alimentación propia. Como se aprecia por las líneas discontinuas, los enlaces radio que se establecen son entre B-A y C-A, esto es, cada nodo directamente con la base.

Esta otra configuración permite evaluar el comportamiento de la red cuando hay dos nodos transmitiendo datos simultáneamente.

Una opción útil es mostrar por pantalla, además de los datos, desde qué nodo se han recibido, con el fin de comprobar que todos están funcionando. Si se está usando la aplicación seriallisten, simplemente hay que fijarse en el primer campo (2B) de la carga útil, que indica el identificador de nodo que calculó esos datos (nodeid).

La próxima configuración es una variación de la anterior (Figura 5.16). Se trata de que un nodo no pueda acceder directamente a la base. El algoritmo de colección de datos usado (LQI) debe ser capaz de encontrar la ruta para cada nodo.



Figura 5.16 Configuración de red con tres nodos (II)

El nodo A actúa como nodo raíz (o base) y está conectado al PC mediante conexión serie (USB). Por otro lado, los nodos B y C actúan como sensores de recogida de datos (remotos). El rayo rojo indica alimentación propia. Como se aprecia por las líneas discontinuas, los enlaces radio que se deberían establecer son C-B y B-A. En este caso, el enlace directo entre C y A no es posible, por ejemplo, están a una distancia demasiado grande.

## 5.5 Comunicación inalámbrica mote-mote (pasarela). Recepción de datos

Pruebas de comunicación similares a las anteriores, si bien ahora se ha introducido la pasarela o *gateway* Tmote Connect, de forma que el PC de recepción de datos está en una red de área local (en el caso particular de estas pruebas, a través de una red local inalámbrica IEEE 802.11g).

Es necesario comprobar el estado de la pasarela. Para ello, es suficiente con acceder a la dirección IP del dispositivo con un navegador web cualquiera. En la página deberá aparecer un cuadro de información como la Figura 5.17.

Device 1 - M4A2M3RK				
Moteiv tmote sky				
Clients	0			
Packets read	0			
Packets written	0			
Serial forwarder	port 9001			
Control, status	port 10001			
Protocol version TinyOS 2.x (autodetect)				
Speed: 115200 (autodetect)				
	Reset device 1: M4A2M3RK			

Figura 5.17 Propiedades nodo conectado a pasarela

La tabla muestra información asociada al nodo conectado al puerto USB de la pasarela Tmote Connect, como el número de clientes conectados, paquetes leídos/escritos, puerto de datos, puerto de control...

La configuración de red (de forma gráfica) para esta prueba se muestra en la Figura 5.18.

Para recibir la información desde un ordenador en red con la pasarela, se hará uso de nuevo de las aplicaciones sf:

./sflisten <dir IP> <puerto>
./prettylisten <dir IP> <puerto>

Es decir, hay que especificar en qué dirección se encuentra la fuente de datos (pasarela) y el puerto por el que se envían dichos datos. Por ejemplo

./sflisten 192.168.1.100 9001



Figura 5.18 Configuración de red con dos nodos y pasarela

El nodo A actúa como nodo raíz (o base). Se encuentra conectado mediante USB a la pasarela Tmote Connect. El nodo B actúa como sensor de recogida de datos (el rayo rojo indica alimentación propia). El enlace radio (línea discontinua) se establece entre B-A. A su vez, la pasarela se conecta mediante Ethernet a un router R. Este router tiene interfaz inalámbrica 802.11g, que usa el PC remoto para recibir datos desde la pasarela. Al encontrarse en una red IP, pasarela, router y adaptador inalámbrico del PC tendrán una dirección IP asignada.

Una vez hecho esto, se comienzan a recibir todos los mensajes que la base envía a través del puerto USB (serie). Como ocurría cuando no existía pasarela, los datos del paquete se reciben sin formato alguno, todos los bytes por orden (y en hexadecimal). Por ejemplo:

00 FF FF 00 00 56 00 12 00 17 0B 1C 0B 18 [...] 00 DA 0A A3

Los campos son idénticos a como se recibían directamente en el puerto serie.

Si se supera la prueba anterior, se pasará a añadir un segundo nodo de recogida de datos. Los datos recibidos por la base desde los dos nodos deberán ser enviados a través de la pasarela hacia el *router*, Figura 5.19.



Figura 5.19 Configuración de red con tres nodos y pasarela (I)

El nodo A actúa como nodo raíz (o base). Se encuentra conectado mediante USB a la pasarela Tmote Connect. Los nodos B y C actúan como sensores de recogida de datos (el rayo rojo indica alimentación propia). Los enlaces radio (líneas discontinuas) se establecen directamente con la base: B-A y C-A. Por otra parte, la pasarela está conectada por Ethernet a un router R. Este router tiene interfaz inalámbrica 802.11g, que usa el PC remoto para recibir datos desde la pasarela.

La última comprobación (Figura 5.20) consiste en que uno de los nodos no sea capaz de acceder directamente al nodo base, por lo que lo tendrá que hacer a través de otro nodo de datos (algoritmo LQI).



Figura 5.20 Configuración de red con tres nodos y pasarela (II)

El nodo A actúa como nodo raíz (o base). Se encuentra conectado mediante USB a la pasarela Tmote Connect. Los nodos B y C actúan como sensores de recogida de datos (el rayo rojo indica alimentación propia). El nodo B puede establecer un enlace radio (línea discontinua) con la base: B-A. Sin embargo, el nodo C no puede comunicar directamente con A, por lo que se el algoritmo de encaminamiento toma el enlace C-B. Por su parte, la pasarela está conectada por Ethernet a un router R. Este router tiene interfaz inalámbrica 802.11g, que usa el PC remoto para recibir datos desde la pasarela.

## 5.6 Comunicación inalámbrica mote-mote (PC). Envío de datos

Hasta el momento, las pruebas realizadas en los distintos escenarios posibles han sido para recibir datos que transmitían los nodos de recogida de datos (remotos).

Ahora, se comprobará el funcionamiento de envío de datos desde la base hasta algún (o todos) nodo de la red. Recuérdese que se usaba el algoritmo de diseminación de datos para hacer llegar paquetes a los nodos necesarios.

La estructura del mensaje para enviar hacia los nodos está definida como

```
1 typedef nx_struct DissMsg {
2    nx_uint16_t nodeid;
3    nx_uint8_t function;
4    nx_uint32_t valueA;
5    nx_uint32_t valueB;
6 } DissMsg;
```

Es decir, se tiene un campo nodeid (2B) para identificar al nodo al cuál irá dirigido el mensaje; una variable function (1B), con la que se indica qué operación se debe realizar en el nodo remoto; y dos variables de propósito general valueA y valueB (4B cada una) que se usan para enviar valores numéricos. Por tanto, para enviar información a un nodo se hará según la tupla

(nodeid, function, valueA, valueB)

Una posible configuración de red podría ser la mostrada en la Figura 5.21.



Figura 5.21 Configuración de red con tres nodos para envío

El nodo A actúa como nodo raíz (o base) y está conectado al PC mediante conexión serie (USB). Por otro lado, los nodos B y C actúan como sensores de recogida de datos (remotos). El rayo rojo indica alimentación propia. Como se aprecia por las líneas discontinuas, los enlaces radio que se establecen son entre B-A y C-A, esto es, cada nodo directamente con la base.

La forma más sencilla de comprobar la comunicación es usar la opción de activar/desactivar la salida digital del sensor denominada CargaA. En este caso, la tupla de parámetros quedaría como

(23, 1, 122, 230)

Esto es, el mensaje va dirigido al nodo con identificador 23, operación 1, que representa act./desact. de la salida digital. El siguiente valor indicará si se activa la carga (mayor que cero) o si de desactiva (cero). El último parámetro no tiene efecto alguno para esta operación, aunque es necesario escribirlo.

Para realizar el envío, se usa la aplicación serialsend:

./serialsend <dispositivo> <velocidad baud> <bytes>

Al igual que ocurría con la recepción de datos, el envío a través de la línea de comandos se hace especificando directamente el valor de los bytes (en decimal). El paquete completo (campo bytes) quedaría

<u>0</u> <u>255</u> <u>255</u> <u>0</u> <u>0</u> <u>11</u> <u>0</u> <u>32</u> <u>0</u> <u>23</u> <u>1</u> <u>0</u> <u>0</u> <u>0</u> <u>122</u> <u>0</u> <u>0</u> <u>0</u> <u>230</u>

- O. Campo de control para los mensajes de SerialActiveMessageC (siempre el mismo valor) 1B.
- 255 255. Dirección destino del mensaje (2B).
- Ø 0. Dirección del enlace que envía el mensaje (2B).
- 11. Tamaño del mensaje sin contar los campos de cabecera (1B). Son los cuatro elementos de la estructura anterior DissMsg (expresados en decimal).
- 0. Identificador de grupo de mensajes (1B).
- 32. Identificador de tipo de mensaje serie (1B). Este valor se puede modificar en la constante simbólica SERIALRX\_ID. Si fueran necesarios otros tipos de mensajes, se les asocian diferentes identificadores.
- El resto del mensaje es la carga útil (datos de la estructura en orden). El bloque 0 23 indica el nodo al que va destinado el paquete; 1 indica la operación a realizar; 0 0 0 122 son los cuatro byte de la variable valueA, mientras que 0 0 0 230 corresponden a valueB.

Al enviar el mensaje completo al nodo base, éste lo reenviará (usando diseminación de datos), hacia todos los nodos, si bien solo el que esté identificado como 23 aceptará los comandos y ejecutará la acción determinada. Para este caso, el comando completo sería (suponiendo mote en COM3):

./serialsend /dev/ttyS2 115200 0 255 255 0 0 11 0 32 0 23 1 0 0 0 122 0 0 0 230

Es posible enviar a todos los nodos de la red: en el campo nodeid hay que especificar el valor de la constante BROAD\_ID, definida como FF FF, o en decimal 255 255.

En cualquier caso, cada vez que se envía un mensaje, Cygwin muestra los datos enviados y aparece ack, indicando que ha sido recibido por la base.

Mediante cualquier voltímetro (o usando la interfaz de control realizada en LabVIEW) se comprueba si realmente la salida se activa (nivel alto de tensión) o se desactiva (nivel bajo). Si no se modificara, una de las causas puede ser que esté demasiado alejado del nodo base, por lo que el mensaje no llega a su destino.

## 5.7 Comunicación inalámbrica mote-mote (pasarela). Envío de datos

Consiste en realizar el mismo proceso que el punto anterior, pero ahora a través de la pasarela Tmote Connect. En la Figura 5.22 se muestra un ejemplo de configuración de red.

Para enviar datos a través de una conexión TCP/IP a la pasarela, se utiliza la función

./sfsend <dir IP> <puerto> <bytes>

El campo bytes debe incluir todos los campos del mensaje (cabecera y carga útil). Tiene la misma estructura que el visto en 5.6. Por ejemplo, para activar la salida digital del nodo con identificador 23:

 $\underline{0} \ \underline{255} \ \underline{255} \ \underline{0} \ \underline{0} \ \underline{11} \ \underline{0} \ \underline{32} \ \underline{0} \ \underline{23} \ \underline{1} \ \underline{0} \ \underline{0} \ \underline{0} \ \underline{122} \ \underline{0} \ \underline{0} \ \underline{0} \ \underline{230} \\$ 



Figura 5.22 Configuración de red con tres nodos y pasarela para envío El nodo A actúa como nodo raíz (o base). Se encuentra conectado mediante USB a la pasarela Tmote Connect. Los nodos B y C actúan como sensores de recogida de datos (el rayo rojo indica alimentación propia). Los enlaces radio (líneas discontinuas) se establecen directamente con la base: B-A y C-A. Por otra parte, la pasarela está conectada por Ethernet a un router R. Este router tiene interfaz inalámbrica 802.11g, que usa el PC remoto para enviar datos hacia la pasarela.

Sabiendo entonces que los paquetes se envían ahora a la pasarela IP y no directamente al sensor (puerto serie), el procedimiento de testeo es idéntico al realizado anteriormente: modificar el estado de la salida digital CargaA (nivel alto o bajo). Resultaría entonces:

./sfsend 192.168.1.100 9001 0 255 255 0 0 11 0 32 0 23 1 0 0 0 122 0 0 0 230

## 5.8 Monitorización y control con MATLAB

Si se han pasado las pruebas anteriores, se puede decir que las comunicaciones, tanto inalámbricas (entre los nodos remotos y la base) y la comunicación a través de IP funcionan correctamente.

En este punto, se añade la aplicación desarrollada en MATLAB para monitorizar los valores recibidos y para poder enviar mensajes de control a los nodos de forma más sencilla. Esta interfaz también soportaba un servidor web, que lleva estos datos a cualquier navegador en formato web *html*. El manejo de la misma está desarrollado en el punto 4.29 de este documento.



Un escenario de pruebas completo se muestra en la Figura 5.23.

Se comienza cargando el programa monitor en la ventana de MATLAB. Después, hay que definir los parámetros de la conexión con la pasarela: su dirección IP y dónde está conectado el nodo base; se intentará establecer una conexión con la pasarela.

Para monitorizar datos recibidos (los nodos remotos deberán estar encendidos y transmitiendo), seleccionar la opción de Recibir datos. Mientras la pasarela reciba datos a través del nodo base, se irán registrando en la ventana de MATLAB. Además, si se ha activado la opción de gráficas, se irán mostrando los resultados temporales de cada sensor.

El siguiente punto a comprobar es el envío de mensajes. Igual que antes, establecer una conexión si no existe ninguna activa. En este caso, se selecciona la opción Enviar datos del menú principal. En ese momento, MATLAB quedará a la espera que se introduzcan los cuatro valores necesarios.

Figura 5.23 Configuración de red con tres nodos, pasarela y monitor MATLAB El nodo A actúa como nodo raíz (o base). Se encuentra conectado mediante USB a la pasarela Tmote Connect. Los nodos B y C actúan como sensores de recogida de datos (el rayo rojo indica alimentación propia). Los enlaces radio (líneas discontinuas) se establecen directamente con la base: B-A y C-A. Por otra parte, la pasarela está conectada por Ethernet a un router R. Este router tiene interfaz inalámbrica 802.11g, que usa el PC remoto para enviar/recibir datos hacia/desde la pasarela. Los datos se manejan a través de la interfaz MATLAB, que establece una conexión TCP/IP con la pasarela.

Tras pulsar OK, se envía el paquete de datos desde MATLAB hacia la pasarela (TCP/IP) y desde ésta hacia el nodo base. A su vez, el nodo base enviará por radio el mensaje a los nodos de su red.

Si un nodo no respondiese a los mensajes enviados puede deberse a problemas en la comunicación establecida entre MATLAB y la pasarela, o que el nodo remoto está demasiado alejado de la red.

## 5.9 Prueba de conjunto

Esta fase se utiliza para verificar el funcionamiento del conjunto de componentes que forman el sistema. En la Figura 5.24 se muestra una descripción gráfica de la configuración de la red y equipos usados; en Figura 5.25 el montaje real.

Para comenzar, comprobar el estado correcto de la pasarela, y arrancar los distintos motes. En PC1, arrancar LabVIEW con el instrumento de testeo, a la vez que se ejecuta el programa monitor elaborado en MATLAB.

Por su parte, en PC2 se arrancará Cygwin para recibir los mensajes de depuración provenientes del nodo marcado en las figuras como C. También puede abrirse un explorador web con la dirección IP y puerto del servidor devuelto por MATLAB.

En este punto, pueden generarse las señales para tensión y potencia con el software Audacity. Estas señales pueden verificarse con el VI de LabVIEW. Se usarán las pruebas ya tratadas anteriormente.

En MATLAB se muestran los valores procesados por los distintos nodos, así como gráficas de evolución temporal. En Cygwin pueden observarse los resultados calculados por el nodo C. En el navegador web, deberán aparecer las últimas medidas recibidas (iguales que en MATLAB).



Figura 5.24 Montaje para realizar prueba de conjunto

El nodo A actúa como nodo raíz (o base). Se encuentra conectado mediante USB a la pasarela Tmote Connect. Los nodos B y C actúan como sensores de recogida de datos. El nodo B tiene alimentación propia (símbolo rayo rojo). Los enlaces radio (líneas discontinuas) se establecen directamente con la base: B-A y C-A. Por otra parte, la pasarela está conectada por Ethernet a un router R, con interfaz inalámbrica 802.11g. El PC1 genera la señal (audio) que recibe tanto el sensor C como la tarjeta de adquisición de datos de LabVIEW. Este PC monitoriza la señal por LabVIEW y contiene el programa monitor de MATLAB (monitorización, control y servidor web). Por su parte, el PC2 puede recoger datos (depuración) provenientes de C a través del puerto USB. También se accede a la página web de toma de datos soportada por el servidor de MATLAB (PC1).

Cuando se haya probado el funcionamiento del sistema en recepción de datos, se prueba el envío de los mismos. Desde MATLAB pueden enviarse mensajes a los distintos nodos (o a todos a la vez). Por ejemplo, activar/desactivar la salida digital, o establecer nuevos umbrales para la distorsión armónica (recuérdese que LabVIEW también monitoriza las salidas digitales del mote C).



Figura 5.25 Montaje prueba de conjunto

Ordenador de comprobación de valores con Cygwin (D), también se reciben los datos procesados a través de la página web HTTP (A), esta página se recibe con una conexión inalámbrica 802.11g. El segundo ordenador comprueba la señal generada a través de LabVIEW (E); la generación de dicha señal también se realiza en ese ordenador, con Audacity (F). El servidor web implementado en MATLAB muestra información a través de la ventana de comandos (G). Tarjeta de adquisición de datos USB-6009 (H). Los dos nodos que envían información hacia la pasarela (donde se encuentra el nodo raíz) son (B) y (C). Las señales analógicas de tensión e intensidad se envían por el conector de audio (I).