



Bizkaiko Foru Aldundia
Diputación Foral de Bizkaia
Berrikuntza eta Ekonomi
Sustapen Saila
Departamento de Innovación
y Promoción Económica

Programa Ekinberri 2007

SmartMotes

Nodos inalámbricos de redes de
sensores con inteligencia
semántica

D2.3 Wireless Multimedia Network
Sensors



Tecnológico Fundación Deusto
Teknologikoa Deustu Fundazioa



RESUMEN

A lo largo del documento se estudian las redes de sensores multimedia inalámbricas (WMNS – *Wireless Multimedia Network Sensors*). Se explicarán los factores que influyen en el diseño de redes de sensores y algunas de las diferentes plataformas hardware existentes.

Además, se han realizado algunas pruebas con la plataforma MICAz, para estudiar como se comportan a la hora de enviar grandes paquetes de datos, los retardos que sufren los paquetes y el tamaño máximo de paquete que se puede enviar.

HISTORIAL DE CAMBIOS

Versión	Descripción	Autor	Fecha	Comentarios
V1	Primer borrador del documento	Leire Muguiru	11/03/2008	

TABLA DE CONTENIDOS

Resumen	3
Historial de cambios	4
Tabla de contenidos	5
1 Introducción.....	6
2 Factores que influyen en el diseño de redes de sensores multimedia	8
3 Plataformas hardware	10
4 PRUEBAS realizadas con la plataforma micaz	12
4.1 Escenario	12
4.2 Alteraciones en el código.....	13
5 Resultados obtenidos.....	14
6 Conclusión	15
7 Referencias	16

1 INTRODUCCIÓN

Las redes de sensores inalámbricas están en pleno auge. Los avances en las redes inalámbricas, en procesos de micro-fabricación e integración, y los microprocesadores embebidos han hecho posible la aparición de redes de sensores idóneas para un amplio rango de aplicaciones militares y comerciales.

Una red de sensores es una infraestructura compuesta de elementos de medición, computación, y elementos de comunicación que permiten al administrador observar el entorno y reaccionar ante eventos. Las redes de sensores multimedia inalámbricas están compuestas de dispositivos capaces de capturar contenido multimedia de forma ubicua, como video, audio, etc.

Las redes de sensores inalámbricas más desarrolladas son aquellas que miden fenómenos físicos como la temperatura, la presión, la humedad. Son aplicaciones que requieren anchos de banda reducidos y generalmente toleran el retardo.

La existencia del hardware de bajo coste como las cámaras y micrófonos CMOS, ha impulsado el desarrollo de las redes de sensores multimedia inalámbricas (WMSN).

A medida que avanzan dichas tecnologías, aparecen nuevos desafíos para el procesado de información en las redes de sensores. Se requieren algoritmos, protocolos, metodologías de diseño...

Las redes de sensores multimedia inalámbricas se utilizan en diversas aplicaciones. Además de las aplicaciones existentes para las redes de sensores tradicionales, éstas permiten desarrollar:

- Redes de sensores de vigilancia multimedia: Se complementan los sistemas de vigilancia existentes con la información de video y audio.
- Almacenamiento de actividades relevantes: para poder acceder a él y tenerlo disponible para futuras consultas.
- Control de tráfico: Posibilita monitorizar el tráfico en las grandes ciudades o autopistas para evitar la congestión del tráfico.
- Cuidados de salud avanzados: Los sensores de telemedicina pueden ser integrados

en redes multimedia para proveer servicios de cuidado de salud avanzados.

- Asistencia automatizada para personas mayores: Se pueden emplear los sensores multimedia para analizar y estudiar en comportamiento de la gente mayor para poder identificar las causas de ciertas enfermedades.
- Monitorización del entorno: En ciertas situaciones puede resultar interesante capturar información acústica o de video para la monitorización del entorno.
- Servicios de localización de personas: Se puede hacer uso de estas redes para localizar a personas desaparecidas o para la identificación de criminales.
- Control de procesos industriales: Los contenidos multimedia pueden ser útiles para procesos de control industrial que tienen tiempos críticos.

Muchos investigadores se han esforzado en desarrollar WMSN-s flexibles y eficientes. A lo largo del documento se expondrán los factores que influyen en el diseño de las redes de sensores multimedia, algunas de las plataformas y dispositivos más relevantes que se han desarrollado, y finalmente se expondrá el experimento realizado para estudiar el comportamiento de las motas MICAZ de Crossbow a la hora de enviar grandes tamaños de datos.

2 FACTORES QUE INFLUYEN EN EL DISEÑO DE REDES DE SENSORES MULTIMEDIA

Existen diversos factores que influyen a la hora de diseñar una red de sensores multimedia inalámbrica. A continuación se explica cada una de ellas.

Existe una gran variedad de aplicaciones que se implementan en este tipo de redes. Dichas aplicaciones tienen diferentes requerimientos, en cuanto a la calidad de servicio se refiere. Por ejemplo, las aplicaciones multimedia de tipo *snapshot* son contenidos que se obtienen en cortos periodos de tiempo, en cambio, el contenido multimedia de tipo *streaming* se genera en periodos de tiempo más prolongados. De modo que hay que ser conscientes de los requerimientos específicos de QoS de cada aplicación.

Por otro lado, para que sea posible la transmisión de los contenidos multimedia hay que tener disponible un ancho de banda mayor que el que se requería en las aplicaciones existentes de sensores. Por ejemplo, la velocidad de transmisión de las plataformas MICAz de Crossbow y TelosB, que cumplen el estándar IEEE 802.15.4, es de 250 kbit/s. Para aplicaciones multimedia, sería recomendable disponer de velocidades de transmisión mayores, pero con consumos energéticos similares.

Para que el envío de contenido multimedia a través de redes inalámbricas sea viable es necesario disponer de un ancho de banda muy elevado. Debido a esto, hace falta emplear técnicas de compresión. Tradicionalmente, se intentaba reducir la tasa de transmisión generada por el origen. Para ello, los codificadores empleaban técnicas de compresión intra-frame, para reducir la redundancia dentro de cada uno de los frames y técnicas de compresión inter-frame, para aprovechar la redundancia entre secuencias consecutivas de frames. De este modo, se reducía la cantidad de datos a transmitir y almacenar. Estos codificadores son complejos, requieren potentes algoritmos de procesado, luego no son la mejor elección para sensores multimedia de bajo coste. Se ha demostrado que se puede invertir dicho proceso [1], de modo que el codificador sea sencillo, y el decodificador complejo, estos algoritmos parecen acertados para las WMSN.

Por otro lado, es interesante el procesado del contenido multimedia dentro de la red. Esto permite incrementar la escalabilidad reduciendo la transmisión de información redundante. Es necesario desarrollar arquitecturas independientes de las aplicaciones para poder actuar sobre el procesado del contenido multimedia. Por ejemplo en el caso de IrisNet [2], emplean un filtrado específico basado en la aplicación reduce el ancho de banda consumido, ya que

en lugar de transmitir todo el contenido, envían una pequeña porción procesada de los datos.

Otro aspecto importante de las redes de sensores multimedia inalámbricos es el consumo energético. En las redes formadas con los sensores tradicionales el consumo energético se veía dominado por la parte de comunicaciones inalámbricas, pero esto no tiene porque cumplirse en el caso del contenido multimedia. Al generar altos volúmenes de datos, el desarrollo de protocolos y algoritmos para alargar el tiempo de vida de la red será esencial.

No hay que olvidar la importancia que tiene el uso de arquitecturas flexibles, para poder soportar aplicaciones heterogéneas, la integración con Internet, así como con otras tecnologías inalámbricas.

3 PLATAFORMAS HARDWARE

A continuación se exponen dispositivos de captura y procesado de imagen de bajo consumo y bajo coste que permiten aumentar la interacción con el entorno.

Los sensores CMOS de captura de imágenes están presentes en diversos sectores tanto industriales como comerciales, como por ejemplo, en teléfonos móviles, PDAs, cámaras digitales. Comparando con los dispositivos tradicionales CCD (Charged-couple device), CMOS es una tecnología acertada para desarrollar sensores de captura de imagen, para poder integrarlos con motas inalámbricas. Elimina la necesidad de tener múltiples chips separados, empleando un único chip. Los dispositivos son de tamaño más reducido, consumen menos potencia y son más ligeros.

Los capturadores de imágenes están diseñados para conectarlos con dispositivos con capacidades computacionales, como teléfonos o PDAs. El módulo Cyclops es un interfaz entre cámaras CMOS y motas MICA2 o MICAz. El firmware está escrito en nesC y permite esconder la complejidad del módulo que captura imágenes de cara a la mota. [3]

Se ha demostrado que los procesadores de 8 bit que se utilizan en la mayoría de las motas no son la mejor elección para el procesado de imágenes. Los procesadores de 32 bit son mejores para este tipo de aplicaciones. Basándose en estas ideas han desarrollado una mota de captura de imagen. [4]

Intel ha desarrollado diferentes plataformas. Stargate, diseñado por Intel y producido por Crossbow, es una plataforma de procesado de alto rendimiento, para aplicaciones de redes de sensores, procesado de señales de sensores, control... Stargate tiene un conector para poder conectarlo a las motas MICA2 o MICAz de Crossbow, a una PCMCIA Bluetooth o tarjetas WiFi [5]. Por otro lado están las plataformas de sensores inalámbricas Imote e Imote2, también desarrolladas por Intel. Emplean un procesador de 32bits y contienen conectores de entrada salida de alta velocidad para poder conectar sensores o cámaras.

Ejemplos de redes de sensores multimedia:

- Panoptes: Desarrollado para aplicaciones de vigilancia y observación del entorno. Utilizan la plataforma String ARM PDA de Intel junto con cámaras web de Logitech que se comunican a través de tarjetas de red 802.11. [6]
- SensEye: Red con tres diferentes niveles formado por nodos inalámbricos

heterogéneos y cámaras. Demuestra las ventajas que aporta una red heterogénea frente a una homogénea. En la primera capa están las motas MICA2 junto con sensores escalares. En la segunda capa están las motas equipadas con cámaras Cyclops o CMUcam. La tercera capa realiza funciones de gateway para comunicarse por un lado con las motas de las capas inferiores y mediante 802.11 se comunica con nodos Stargate de la capa tres. [7]

- IrisNet: Los usuarios realizan peticiones para actuar sobre sensores de video. Para el usuario, la red de sensores es una unidad sobre la que puede interactuar por medio de un lenguaje de alto nivel. [8]

- En el laboratorio de Broadband and Wireless Networking (BWN) de Georgia realizaron unas pruebas para demostrar la eficiencia de los algoritmos y protocolos para comunicaciones multimedia a través de redes inalámbricas. Para ello implementaron una red formada por imotes de Intel y motas MICAz de Crossbow. Incluyeron tres tipos de sensores multimedia:
 - Sensores de gama baja capaces de capturar imágenes como cámaras CMOS, que pueden integrarse con motas MICAz.
 - Sensores multimedia basados en cámaras web Logitech de calidad media interconectadas con las plataformas Stargate.
 - Sensores de video de gama alta instaladas en la plataforma Acromate CARCIA. [9]

4 PRUEBAS REALIZADAS CON LA PLATAFORMA MICAZ

Se han realizado diversas pruebas para ver como se comportan las motas MICAz de Crossbow al enviar paquetes de diferentes tamaños para analizar los retardos sufridos y el tamaño máximo de paquete que se puede llegar a enviar con esta plataforma.

4.1 Escenario

La mota MICAz base envía paquetes XMesh a una mota remota. Para comprobar que la mota remota ha recibido el paquete se comprobará que el contenido es el mismo y se actuará sobre los leds de dicho dispositivo.

A continuación se muestra la estructura de un mensaje y de un paquete XMesh.

XMesh Message Structure

Component	Description
Header	Composed of standard TOS header and XMesh multihop header
Payload	User data
CRC	CRC check (Refer to Appendix E)

XMesh Packet Structure

Type	Name	Description	Component
uint16_t	addr	Destination address (next hop)	TinyOS Header (MICA2). The MICAz header has 5 additional bytes.
uint8_t	type	AM type; defines type of message	
uint8_t	group	AM group	
uint8_t	length	Remaining bytes in message, N/I CRC	
uint16_t	sourceaddr	Address of mote that sent the message	XMesh multihop header
uint16_t	originaddr	Address of mote that originated the message	
uint16_t	seqno	Message sequence number.	
uint8_t	socket	Application ID	
uint8_t	data	Payload	Payload size is set with the TOSH_DATA_LENGTH define. Default size is 29 bytes
uint16_t	crc	CRC check (Refer to Appendix E)	CRC

En principio, el tamaño por defecto de la carga útil de datos es de 29 bytes, pero se puede incrementar. Generalmente el tamaño máximo de datos se limita a 55bytes. Ampliar el tamaño de datos implica un incremento del uso de la SRAM, debido a la creación de múltiples buffers TOS.

`TOSH_DATA_LENGTH` determina el tamaño máximo del payload. Pueden enviarse mensajes de tamaño menor entre las motas sin problemas.

4.2 Alteraciones en el código

Se ha variado el tamaño de *TOSH_DATA_LENGTH* para analizar como se comporta la plataforma MICAz al enviar paquetes de tamaños diferentes.

Los cambios que hay que hacer en el código para variar el tamaño de los mensajes son los siguientes:

En el fichero *Makefile* de las motas involucradas en la comunicación hay que incluir la siguiente línea de código:

```
CFLAGS += -DTOSH_DATA_LENGTH=X
```

Donde X es el tamaño máximo que queremos establecer en bytes.

En el caso de la mota base, en nuestro caso, hemos utilizado Mig para generar código java y a continuación hemos hecho uso de las librerías de Tinyos para establecer la comunicación entre la mota y java.

Cuando alteramos el tamaño del paquete, hay que ser conscientes de que todos los paquetes tienen que coincidir. La definición de los paquetes de datos, en el emisor y en el receptor tiene que ser igual para que se puedan entender.

En nesC, en los ficheros con extensión “.h” se definen los paquetes que enviamos, de modo que al alterar el tamaño de los paquetes, tenemos que cerciorarnos de que los paquetes siguen siendo iguales en el emisor y en el receptor. Será necesario volver a generar el código java con Mig cada vez que alteremos la estructura de datos.

5 RESULTADOS OBTENIDOS

Si en nuestra aplicación queremos utilizar una pantalla de resolución de 128x128 píxeles, y cada píxel está representado por 2bytes, el tiempo estimado para enviar una imagen sería el siguiente:

Datos que queremos transmitir → $(128 \times 128) \text{ píxeles} \times 2 \text{ bytes/píxel} = 32768 \text{ bytes}$

Paquetes que vamos a transmitir → $32768 \text{ bytes} / 100 \text{ bytes/paquete} = 328 \text{ paquetes}$

Tamaño de la carga útil de datos	Instante en el que se ha enviado el primer paquete (milisegundos)	Instante en el que se ha enviado el último paquete (milisegundos)	Tiempo necesario para enviar los paquetes
50 bytes/paquete	Paquete nº0 instante: 1204727562937 ms	Paquete nº50 instante: 1204727564078 ms	50 paquetes en 1141 ms
100bytes/paquete	Paquete nº0 instante: 1204729819859 ms	Paquete nº49 instante: 1204729821343 ms	50 paquetes en 1484 ms
***Es necesario introducir un sleep de 5ms entre el envío consecutivo de mensajes en java para que el puerto serie funcione correctamente.	Paquete nº0 Instante: 1204818937812 ms	Paquete nº328 Instante: 1204818947687 ms	328 paquetes en 9875 ms
	Paquete nº0 instante: 1204732126468 ms	Paquete nº1699 instante: 1204732180062 ms	1700 paquetes en 53594 ms

No es posible incrementar el tamaño del paquete a 110bytes, el receptor no los recibe.

Hemos comprobado que hacen falta 10segundos para transmitir los 32,8kB de la imagen.

6 CONCLUSIÓN

El contenido multimedia, como por ejemplo el vídeo, sin comprimir consume mucho ancho de banda y energía, de modo que resulta necesario aplicar compresiones que tengan alta eficiencia. No hay que olvidar, que los codificadores, irán embebidos en los dispositivos de sensorización, de modo que tienen que ser lo menos complejos posibles, para que el dispositivo sea de bajo coste, no consuma mucha potencia y no se vea muy afectado el factor de forma.

Otro desafío que hay que abordar es el desarrollo de sistemas software capaces de abstraer la información recogida por los sensores escalares y multimedia, a las capas de aplicación, para proporcionar un fácil y rápido desarrollo de las aplicaciones de redes de sensores. Es decir, exportar los servicios a capas de aplicaciones más altas a través de máquinas virtuales, sistemas operativos, y middleware. Las características que tiene que tener el sistema software son:

- Proveer un interfaz de alto nivel que especifique el comportamiento de los sensores.
- Permitir al usuario manejar algoritmos específicos de cada aplicación para que pueda actuar sobre el contenido multimedia.
- Un largo tiempo de vida.
- Aplicable a diferentes aplicaciones, sensores y plataformas.
- Escalable.

Sdlib es una librería para redes de sensores y comunicaciones implementada en nesC, para aplicaciones que requieren grandes paquetes de datos, como por ejemplo aplicaciones de monitorización de video. Permite abstraer operaciones comunes en redes de sensores y además permite acceder a operaciones de bajo nivel, que están implementadas como componentes nesC.

Por último, comentar que ZigBee permite usar ranuras de tiempo de forma dedicada para cada comunicación, pero la baja tasa de transferencia limita su aplicabilidad a aplicaciones multimedia. ZigBee no implementa soporte para dividir paquetes grandes en más pequeños, luego será necesario introducir código adicional en la capa de aplicación.

7 REFERENCIAS

- [1] B. Girod, A. Aaron, S. Rane, D. Rebollo-Monedero, Distributed video coding, Proc.IEEE 93 (1) (2005) 71-83.
- [2] S. Nath, . Ke, P.B. Gibbons, B.Karp, S. Seshan, A distributed filtering architecture for multimedia sensors, Intel Research Technical Report IRP-TR-04-16, August 2004.
- [3] M.Rahimi, R. Baer, O. Iroezi, J. Garcia, J.Warrior, D. Estrin, M. Srivastava, Cyclops: in situ image sensing and interpretation in wireless sensor networks. in: Proc. Of the ACM Conf. on Embedded Networked Sensor Systems (SenSys), San Diego, CA, November 2005.
- [4] I.Downes, L.B. Rad, H. Alhajan, Development of a mote for wireless image sensor networks, in: Proc. Of COGNitive systems with Interactive Sensors (COGIS), Paris, France, March 2006.
- [5] The Stargate Platform. <
http://www.xbow.com/Support/Support_pdf_files/Stargate_Manual.pdf >
- [6] W. Feng, B. Code, E. Kaiser, M. Shea, W. Feng, L. Bavoil, Panoptes: scalable low-power video sensor networking technologies, in : Proc of ACM Multimedia, Berkeley, CA, USA, November 2003.
- [7] P. Kulkarni, D. Ganesan, P.Shenoy, Q. Lu, SensEye: a multi-tier camera sensor network, in: Proc. Of ACM Multimedia, Singapore, November 2005.
- [8] S. Nath, Y. Ke, P.B. Gibbons, B.Karp, S. Seshan, A distributed filtering architecture for multimedia sensors Intel Research Technical Report IRP-TR-04-16, August 2004.
- [9] I. Akyildiz, T. Melodia, K. Chowdhury, A survey on wireless multimedia sensor networks, Computer Networks, August 2006.