SISTEMA DE CONTROL AGRÍCOLA

Jorge Rivero Calera

Ingeniería técnica de telecomunicaciones: Especialidad en telemática.

Consultor: Jordi Bécares Ferrés.

12/06/2012

Dedicado a mi familia, amigos y en especial a Clara,
por el apoyo y la confianza recibidos.

Resumen

El presente documento pretende documentar el trabajo realizado en el desarrollo del Sistema de control agrícola. Así mismo se trata de un estudio inicial de los sistemas empotrados, en concreto los destinados a operar en una red de sensores o *WSN* (*Wireless Sensor Network*) por sus siglas en inglés.

El Sistema de control agrícola consta de dos nodos de red, también llamados motas, encargados de recopilar la información del medio-ambiente en el interior de un invernadero, y cambiar el medio-ambiente a las exigencias de una cosecha dada. La carga computacional de las acciones a realizar recae en el PC, que estará conectado a una de las motas para poder formar parte del sistema total.

Palabras clave: Motas, sistemas empotrados, red de sensores inalámbricos, WSN, TinyOS, NesC, invernadero inteligente, control de cosechas.

Área: Sistemas empotrados.

Título: Sistema de control agrícola.

Índice.

1 Introducción 6
- **1.1 Justificación** 8
- **1.2 Descripción** 9
- **1.3 Objetivos del TFC** 10
- **1.4 Enfoque y método seguido** 11
- **1.5 Planificación del proyecto** 12
- **1.6 Recursos empleados** 13
- **1.7 Productos obtenidos** 14
- **1.8 Breve descripción de otros capítulos de la memoria** 14

2 Antecedentes 16
- **2.1 Estado del arte** 16
- **2.2 Estudio de mercado** 19

3 Descripción funcional 21
- **3.1 Sistema total** 21
- **3.2 PC (Diseño de la interfaz de usuario)** 22
- **3.3 Mota (Diseño de la aplicación hecha en la mota)** 24

4 Descripción detallada 28
- **4.1 Sistema total** 28
- **4.2 PC (Diseño de la interfaz de usuario)** 29
- **4.3 Mota (Diseño de la aplicación hecha en la mota)** 30

5 Viabilidad técnica 36

6 Valoración económica 38

7 Conclusiones 40
- **7.1 Conclusiones** 40
- **7.2 Propuestas de mejora** 41
- **7.3 Autoevaluación** 42

8 Glosario 43

9 Bibliografía 45

10 Anexos 47
- **10.1 Ejecución y compilación** 47
- **10.2 Monitorización del sistema** 48
- **10.3 Código de luces** 50

Índice de figuras

Figura 1. Esquema general del Sistema de control agrícola..... 8
Figura 2. Situación de la mota, en el interior del invernadero..... 9
Figura 3. Cronograma inicial del proyecto..... 12
Figura 4. Cronograma final del proyecto..... 13
Figura 5. Imagen de las motas MITes..... 18
Figura 6. Tabla comparativa de las motas mencionadas..... 19
Figura 7. Diagrama de flujo de la aplicación Interfaz..... 23
Figura 8. Diagrama de flujo de la aplicación de la mota-central..... 25
Figura 9. Diagrama de flujo de la aplicación de la mota-invernadero..... 27
Figura 10. Estructura de los mensajes básicos TinyOS..... 28
Figura 11. Estructura de los mensajes del Sistema de control agrícola..... 29
Figura 12. Detalle de los componentes hardware más importantes..... 31
Figura 13. Componentes y conexiones del módulo EstacionBase..... 32
Figura 14. Componentes y conexiones del módulo ModoSupervisado..... 33
Figura 15. Componentes y conexiones del módulo Invernadero..... 34
Figura 16. Detalle de la conexión del módulo Luz y el módulo LecturaADC..... 35
Figura 17. Componentes y conexiones del módulo EfectoHall..... 35
Figura 18. Desglose de precios del Sistema de control agrícola..... 38
Figura 19. Ejemplo de compilación e instalación de la mota-invernadero..... 48
Figura 20. Ejecución de la aplicación Interfaz sincronizada a 6 segundos,
junto con el Serial Forwarder..... 49
Figura 21. Mota-invernadero recién iniciada,
a la espera del mensaje de sincronización..... 51
Figura 22. Mota-central operando en modo supervisado,
a la espera de recibir confirmación de envío..... 52

Capítulo 1

1. Introducción.

La propuesta del sistema de control agrícola, responde a varias necesidades que tratamos de satisfacer basándonos en la tecnología actual.

Más concretamente pretende solucionar o mejorar las necesidades de un invernadero, buscando la comodidad, eficiencia energética y ahorro económico. El sistema tratará de automatizar las tareas básicas y optimizar las condiciones ambientales requeridas para la cosecha.

Se designará una mota para la recolección de datos dentro del invernadero. Y una segunda conectada a un PC; su cometido será recibir el estado de varios elementos básicos (luz, ventilación, y temperatura), calcular las acciones necesarias e informar de las medidas a tomar.

Las funciones específicas de cada mota serán:

- **Mota-estación del invernadero.**

Control de temperatura:

– Control de la temperatura en el interior de el invernadero.

Control lumínico:

- Control de la luz en el interior del invernadero, ya sea natural o artificial.

Control de ventilación interior:

- Control del estado de los ventiladores encargados de renovar el aire del invernadero.

Envío de datos recopilados:

-Establecer comunicación con la estación central para informar del estado en el interior del invernadero.

- **Mota-estación central.**

Recolección de datos:

- Establecer comunicación con la mota del invernadero para la recepción de la información.

Informar de las alarmas:

- Contrastar la información recibida con los valores óptimos para cada tipo de cosecha.

- Informar de las incidencias existentes.

Envío de acciones a desempeñar:

- Envío de datos con la información necesaria para resolver las incidencias según el caso; accionar ventiladores, encender lámparas, etc..

- Estas acciones serán previamente supervisadas por un responsable, por lo que su envío será semi-automático, valiéndonos de la posibilidad de accionar el botón de la mota para ratificarlas.

Cabe destacar que la estación central podrá comunicarse con el PC, por lo que las posibilidades de control se multiplican. Una de las posibilidades es el control de diferentes cosechas, prestando especial atención a las cosechas más sensibles. Para esto, dispondríamos de una base de datos para la consulta de los diferentes periodos de luz, temperatura, ventilación y la frecuencia de monitorización del invernadero recomendadas. Así mismo la base de datos también consideraría las diferentes estaciones del año para actuar en consecuencia.

– Esquema general:

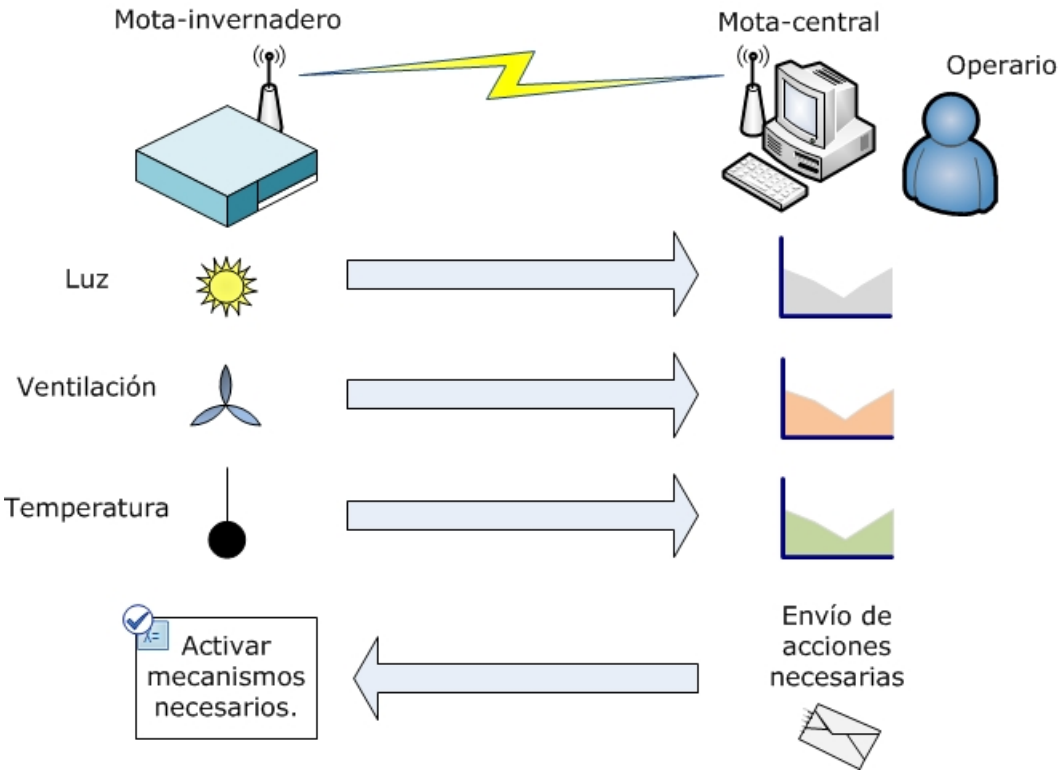

Figura 1. Esquema general del Sistema de control agrícola.

1.1 Justificación.

En los últimos tiempos oímos con frecuencia los conceptos sostenibilidad, ahorro energético; o conceptos más consolidados como ecología. En torno a estos conceptos gira la idea principal del proyecto, buscando satisfacer las necesidades actuales y futuras, de un sector a menudo sofocado por los costes y beneficiario del ecosistema en el que se desarrolla. Para cumplir estos objetivos, y gracias a los conocimientos adquiridos se trata de abordar estos problemas con el Sistema de control agrícola.

La finalidad del Sistema de control agrícola, es la de ahorrar en costos de personal, y de recursos energéticos. Los beneficios aportados son en mayor medida económicos, pero no hay que dejar de lado los beneficios ecológicos, ya que se puede reducir la “huella ecológica” notablemente.

Ya que la legislación aplicable a cultivos ecológicos no contempla los invernaderos en gran medida, el Sistema de control agrícola aporta un componente ético a las certificaciones ecológicas, que difícilmente se verán satisfechas con los equipamientos actuales. En esta linea, Holanda hace avances importantes en la tecnología de los invernaderos, mientras España se encuentra estancada en sistemas obsoletos.

1.2 Descripción.

Para la estructura completa del sistema, son necesarias un mínimo de dos motas y un PC. Este sencillo sistema es aplicable a cualquier invernadero, y fácilmente exportable a todo tipo de cosechas. No obstante, el invernadero que tratamos de abastecer ha de equipar sistemas calefactores, de refrigeración e iluminación artificial.

La primera de las motas, la mota-invernadero, es la encargada de estudiar las variables ambientales del invernadero, de enviar dicha información, y recibir las órdenes para contrarrestar posibles desviaciones de los parámetros óptimos. Situada a una distancia prudencial de los ventiladores y a media altura para recoger datos fiables de luz y temperatura. La proximidad de un electro-imán en paralelo con los ventiladores, permite la correcta lectura de sus estados. Podremos tener tantas motas como cosechas deseemos; también es conveniente apuntar, que a más motas en un mismo invernadero mayor sera la optimización de la cosecha.

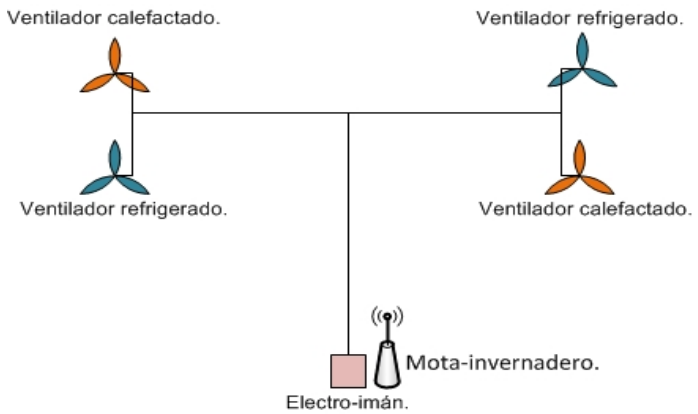

Figura 2. Situación de la mota, en el interior del invernadero.

La segunda mota, la mota-central, actúa a modo de puente de comunicaciones, recibiendo la información enviada desde el invernadero, y enviando las órdenes recibidas desde el PC. La mota-central puede controlar el envío de las órdenes, recibidas desde el PC por un operario si activa el modo supervisado (del cual hablaremos más adelante), por medio de una llave magnética.

El PC es el encargado de calcular las órdenes, en función de unos parámetros establecidos como óptimos para una cosecha determinada. Adicionalmente sirve como monitor de el invernadero y órdenes enviadas.

El sistema tiene dos modos de funcionamiento; automático y supervisado. En modo automático, podemos delegar la toma de decisiones al PC. Mientras que en el modo supervisado, tomaremos nosotros la decisión de enviar o no las órdenes para rectificar los valores.

1.3 Objetivos.

- Recogida de información medio-ambiental en el interior del invernadero.
- Monitorización de sensores.
- Envío y recepción de información entre mota y PC.
- Implementación del modo automático.
- Implementación del modo supervisado.
- Desarrollo de aplicación de PC para el tratamiento de la información recibida.

El proyecto trata de definir varias aplicaciones para cada mota y el PC. La mota-invernadero ha de ser capaz de interpretar la información recibida por sus sensores, la mota-central se ocupara del establecimiento de la comunicación entre invernadero y PC. La aplicación del PC por su parte, se encargará de la visualización y tratamiento de la información recibida. Así mismo deberá informar de incidencias a la mota, distinguiendo los valores incorrectos de los óptimos y calculando las acciones necesarias. Se definen igualmente dos modos de operación, automatizando el envío de acciones; o supervisando el envío de las mismas.

1.4 Enfoque y método seguido.

El proyecto ha partido de la base de una necesidad, reducir la carga de trabajo de agricultores, que de manera ineludible destinan a la atención de una cosecha. A raíz de esta necesidad surge la idea de automatizar estos procesos, y el posterior desarrollo del proyecto.

El enfoque de este proyecto comienza con la definición del sistema global, y responde a las siguientes preguntas:

- ¿Qué necesidades tiene una cosecha?
- ¿Qué tareas podemos automatizar?

Una vez respondidas estas preguntas, se han definido las funcionalidades del sistema. Las partes que forman el sistema y las tareas destinadas a cada una de estas partes. El paso posterior ha sido la implementación del mismo y su posterior puesta a punto. Todo ello con la mirada siempre puesta en la escalabilidad a mayores y diferentes cosechas, punto fundamental si el objetivo final es reducir la carga de trabajo soportada por los agricultores y el consecuente ahorro en recursos económicos, naturales y humanos.

1.5 Planificación.

El ciclo de vida del proyecto se baso en el **modelo en espiral**, con una continua revisión del producto entregado, y mejoras continuas en función de los riesgos asumidos a la hora de llevar a cabo estas.

La planificación inicial del proyecto se presenta en el siguiente diagrama:

Nombre de tarea	Duración	Comienzo	Fin
Memoria	**65 días?**	**mié 07/03/12**	**vie 01/06/**
Crear plantilla	1 día?	mié 07/03/12	mié 07/03/
Recoger información/notas	65 días?	mié 07/03/12	vie 01/06/
Documentar el código	**27 días?**	**mié 18/04/12**	**mar 22/05/**
Documentar el código 1/2	5 días?	mié 18/04/12	mar 24/04/
Documentar el código 2/2	3 días?	sáb 19/05/12	mar 22/05/
Entrega de código 1/2	**21 días?**	**mar 20/03/12**	**mar 17/04/**
Calibrado de sensores	5 días?	mar 20/03/12	lun 26/03/
Envío/recepción de variables	8 días?	mar 27/03/12	jue 05/04/
Envío/recepción de ordenes	8 días?	vie 06/04/12	mar 17/04/
Entrega de código 2/2	**25 días**	**mié 18/04/12**	**sáb 19/05/**
Desarrollo del software del PC	5 días	mié 18/04/12	mar 24/04/
Implementación de la comunicación mota-PC	8 días	mié 25/04/12	vie 04/05/
Implementación de la comunicación PC-mota	7 días	sáb 05/05/12	lun 14/05/
Implementar el modo supervisado	5 días	mar 15/05/12	sáb 19/05/
Presentación	**10 días?**	**dom 03/06/12**	**jue 14/06/**
Elaboración	**7 días**	**dom 03/06/12**	**lun 11/06/**
Elaboración de las diapositivas	5 días	dom 03/06/12	jue 07/06/
Grabación de video	2 días	vie 08/06/12	lun 11/06/
Pruebas y revisión	**2 días?**	**mar 12/06/12**	**mié 13/06/**
Ensayo	2 días?	mar 12/06/12	mié 13/06/
Presentación final	1 día?	jue 14/06/12	jue 14/06/

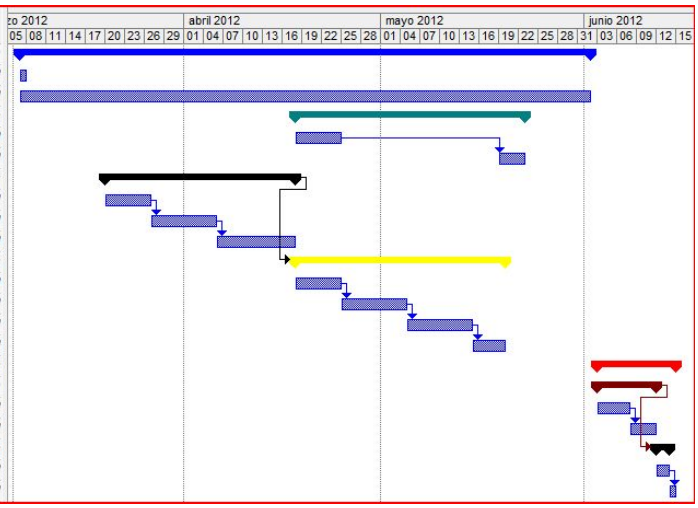

Figura 3. Cronograma inicial del proyecto.

Dado el óptimo avance del proyecto, a mitad del desarrollo decidimos incluir las tareas de desarrollo del software del PC e implementaciones de las comunicaciones, en la primera fase del código a elaborar. Pudiendo retocar más tarde aspectos descuidados y defectos que se venían presentando a lo largo del ciclo del proyecto, y poder así entregar una primera versión aproximada del sistema completo.

La planificación inicial no ha sufrido mayores cambios, si bien no hemos podido añadir objetivos optativos. Estos objetivos incluían la elaboración de una base de datos y la capacidad de la aplicación del PC para consultarla. Por escasez de tiempo no se ha podido implementar esta mejora, la cual habría conllevado trabajar en la capacidad de la aplicación del PC, para comunicarse con la base de datos; distinguir las diferentes cosechas y motas; y la elaboración de la base de datos a consultar.

La planificación final queda tal como se muestra en la siguiente imagen:

Nombre de tarea	Duración	Comienzo	Fin
+ Memoria	65 días?	mié 07/03/12	vie 01/06/
− Entrega de código 1/2	21 días?	mar 20/03/12	mar 17/04/
Calibrado de sensores	5 días?	mar 20/03/12	lun 26/03/
Envío/recepción de variables	8 días?	mar 27/03/12	jue 05/04/
Envío/recepción de ordenes	8 días?	vie 06/04/12	mar 17/04/
− Entrega de código 2/2	25 días	mié 18/04/12	sáb 19/05/
Desarrollo del software del PC	5 días	mié 18/04/12	mar 24/04/
Implementación de la comunicación mota-PC	8 días	mié 25/04/12	vie 04/05/
Implementación de la comunicación PC-mota	7 días	sáb 05/05/12	lun 14/05/
Implementar el modo supervisado	5 días	mar 15/05/12	sáb 19/05/
+ Presentación	10 días?	dom 03/06/12	jue 14/06/

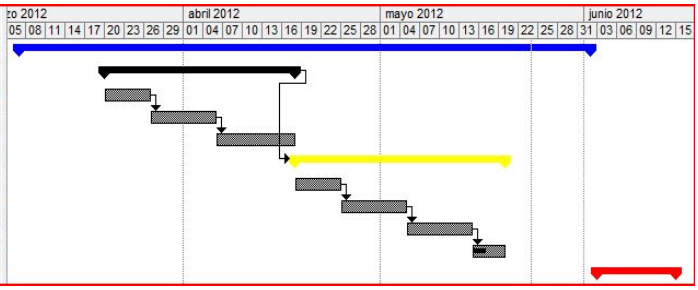

Figura 4. Cronograma final del proyecto.

1.6 Recursos empleados.

- Recursos hardware/software:

Dos **motas COU 24** diseñadas para la recogida de información del medio ambiente, y la interconexión y traspaso de dicha información funcionando en una red de sensores.

PC con sistema operativo **ubuntu 10.04**, con el **entorno de programación Eclipse (versión Ganymede)** más los **plugins Yeti2,** necesario para la programación de las motas. La aplicación **meshprog (versión 0.1.1),** herramienta para la instalación del código compilado en las motas.

Lenguaje **NesC en su versión 1.3.1** y sistema operativo **TinyOS 2.1.1,** lenguaje de programación y sistema optimizados para las redes de sensores. NesC el lenguaje con el que programaremos las aplicaciones de las motas.

Lenguaje **Java** y su kit de desarrollo, para el desarrollo de aplicaciones que nos ayuden a interpretar la información recibida, y el envío de información a las motas.

1.7 Productos obtenidos.

- Mota-invernadero:

La mota-invernadero es la encargada de recoger toda la información sobre el estado del invernadero. Para ello consta de sensores de luz, campo magnético y temperatura. El software instalado se encarga de encapsular todas las variables en un mensaje, y enviarlo a la mota-central. También será la encargada de recibir los mensajes, extraer la información necesaria y tomar las medidas oportunas.

- Mota-central:

Es la encargada de enrutar la información a la mota invernadero o al PC, y de recibir los mensajes que estos generen. La mota-central no hace uso de los sensores como la mota-invernadero, sin embargo usa el sensor de hall (campo magnético) , para detectar la llave magnética y operar en modo supervisado.

- Aplicación del PC:

Es la encargada de realizar los cálculos, y tomar las decisiones para rectificar valores fuera de los margenes establecidos para cada cosecha. Para ello, cuenta con la capacidad de enviar y recibir mensajes vía conector USB.

1.8 Breve descripción de otros capítulos de la memoria.

Tras el primer capítulo introductorio, se expondrán los siguientes capítulos:

- Antecedentes:

Estudio de los diferentes sistemas operativos, motas, y tecnologías usadas hasta hoy en el mundo de los sistemas empotrados.

- Descripción funcional:

Capítulo en el que se tratará de aclarar el funcionamiento del sistema, así como justificar el diseño del software del mismo.

- Descripción detallada:

Descripción técnica del funcionamiento de las diferentes aplicaciones, arquitectura del software desarrollado.

- Viabilidad técnica:

Estudio de la facilidad para implementar un sistema como el expuesto.

- Valoración económica:

Estimación del valor monetario del sistema en el mercado.

- Conclusiones:

Capítulo en el que se evalúa el rendimiento del proyecto. Resultados finales y opciones de mejoras.

Capítulo 2

2. Antecedentes.

- Introducción:

En la actualidad, los sistemas empotrados han conseguido un nivel de desarrollo que los hacen útiles en infinidad de ámbitos Se diseñan sistemas empotrados para funcionar en electrodomésticos, en la industria automovilística, aeronáutica e incluso en la medicina. Aunque no nos percatemos de ello, todos hemos oído hablar de conceptos como ordenador de abordo en los automóviles, o nos guiamos por un gps a la hora de llegar a un destino. Todos estos son ejemplos cotidianos de sistemas empotrados.

Si nos centramos en las redes de sensores, la monitorización de variables es una de las principales tareas que desempeñan. Pueden monitorizar, en función de los sensores de los que dispongan, el tráfico y polución de una ciudad, las señales sísmicas de un volcán o la composición química de medios acuáticos.

2.1 Estado del arte.

- Protocolos de comunicación:

Algunas de las especificaciones más usados para la transmisión de datos son los conocidos Bluetooth, para entornos domésticos o de corto alcance, o Zigbee más orientado al bajo consumo energético y baja tasa de transferencia de datos. Otras especificaciones menos conocidas a destacar son:

- Knx, estándar europeo y una alternativa a los protocolos inalámbricos, ya que se beneficia del cableado eléctrico del hogar para formar su red. Destinado a entornos domésticos.
- Bluetooth low energy, similar a Zigbee por sus características, pero orientado a redes personales.

- Dash7, un futuro rival de Zigbee, promete menor consumo, mayor alcance y ausencia de interferencias entre otras cualidades.

De todos los métodos presentados, Zigbee es de los más extendidos en redes de sensores en cuanto a implantación Algunas de las características por las que considerar su instalación en redes de sensores son:

- El ahorro energético que supone su bajo consumo.
- Puede formar redes de hasta 64.000 nodos.
- Bajo coste de la tecnología.
- Nodos agrupables en diferentes redes.
- Funciones de broadcast o difusión de información a un grupo de nodos.
- Redes centralizables, pueden recolectar información y enviarla a un nodo central.

- Sistemas operativos:

En cuanto al sistema operativo, podemos contar con diversos sistemas operativos en función de la aplicación final del sistema empotrado. Podemos nombrar entre otros, versiones de Windows para la empresa automovilística como Windows Embedded Automotive, u OSEK específico de la industria automovilística europea. Con una finalidad más sofisticada, y destinados a albergarse en satélites o vehículos espaciales contamos con vxWorks. Y para usos más cotidianos y conocidos, sistemas operativos como Android que planean el salto de los móviles a los electrodomésticos

TinyOS, esta ideado para manejar operaciones de muy bajo consumo en redes de sensores, en las que la longevidad del suministro eléctrico es un parámetro crítico.
Las herramientas de las que dispone están orientadas a reducir, no solo el consumo energético, si no todos los recursos disponibles, como también pueden ser:

- La capacidad operacional del microprocesador.
- O la memoria disponible.

Una de las desventajas con las que tiene que lidiar TinyOS es el consumo de la radio, elemento fundamental en las motas de toda red de sensores inalámbricas. TinyOS ofrece la posibilidad de apagarla cuando no se use, y volver a encenderla cuando vuelva a ser necesaria. La gestión de las interrupciones, como código prioritario al resto, es otro buen ejemplo.

Las interfaces programables (API's) y librerías de hardware de las que dispone, facilitan enormemente la programación de los diferentes elementos hardware, dándonos una visión abstracta de los componentes.

- Motas:

Las motas, los nodos sensoriales de toda red de sensores, han de contar con interfaces que permitan la comunicación, y sensores que recojan la información. Existen infinidad de motas, clasificables según sus sensores, sistema operativo, procesador, etc. Por ello para cada problema diferente, tendremos una solución diferente. Podemos encontrarnos con motas sofisticadas como la iBadge con sensores de movimiento, micrófono, altavoz, y otros sensores ambientales. La sencilla MITes, exclusivamente para la monitorización del movimiento o la potente Medusa MK-II, con dos procesadores para tareas computacionales avanzadas.

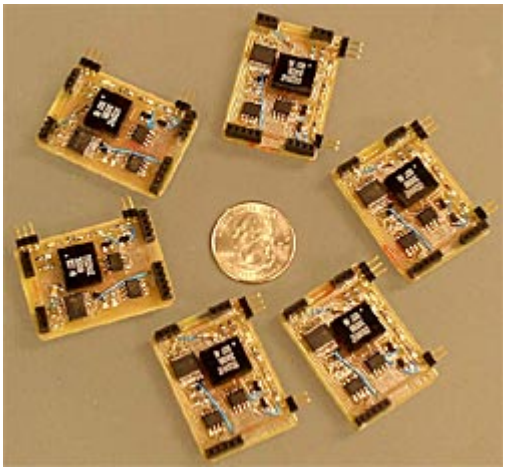

Figura 5. Imagen de las motas MITes.

En la practica las compañías seleccionaran la mota más adecuada para el propósito, en función de las variables a medir, el tamaño o la tasa de transmisión, entre otros criterios. La combinación de estos factores, dará lugar a la mota idónea para la red de sensores que se requiera.

Mota	Procesador	Sensores	Protocolo de radio.
cou24	Atmega 1281 (4MHz).	-Temperatura. -Efecto-Hall. -Luz.	ZigBee como estándar de radio.
iBadge	Atmega 128L (8MHz).	-Acelerómetro. -Humedad. -Magnetismo. -Luz. -Ultrasonidos. -Micrófono. -Presión.	Bluetooth como estándar de radio.
MITes	nRF24E1 (16MHz).	Ninguno en particular. Multisensor.	Banda ISM. Específica para medicina, ciencia e industria.
Medusa MK-II	ATMega128L (4MHz). AT91FR4081 (40MHz).	-Luz. -Temperatura. -Ultrasonido.	Chip RFM TR1000 configurable para varios protocolos.

Figura 6. Tabla comparativa de las motas mencionadas.

2.2 Estudio de mercado.

Los invernaderos inteligentes, son una tecnología relativamente nueva, y poco implantada en algunos países. Sin embargo, en el mercado actual existen sistemas similares al Sistema de control agrícola Citaremos a continuación varios productos y sus principales diferencias:

- **Link4** es una compañía que desarrolla sistemas de control de invernaderos, analizando su **serie 1000**, vemos que incluyen infinidad de sofisticados sensores. La principal diferencia con nuestro sistema es que sus sensores son cableados. Las series 1000 incluyen sensores de ventilación, temperatura, CO_2, humedad, e incluso control del pH. Se trata de una plataforma de hardware totalmente dedicada, aunque existe la posibilidad de adquirir un software para el ordenador. Así mismo es interconectable con otros sistemas de diferentes gamas del fabricante.
- **TNO** es una empresa de investigación Holandesa. Uno de los productos para el control de humedad en los cultivos es **Sensiplant**. Se trata de una red de sensores de humedad, pudiendo ser monitorizados en remoto, la principal desventaja es que por ahora, solo monitoriza la humedad de la tierra. Una de las particularidades técnicas del sistema, es que usa un protocolo de auto-organización desarrollado por la propia TNO, de manera que su instalación resulta muy cómoda. Los diferentes sensores enrutan los datos entre sí, de manera que la red puede ampliarse indefinidamente, sin perdidas debidas a la distancia.

- **Advanticsys** es una empresa española que se dedica a todo tipo de redes de sensores. Trabajan en el sector de la agricultura con diferentes productos, e incluyen en su catálogo sistemas de monitorización de cultivos. El sistema destinado al control de invernaderos incluye sensores de humedad y temperatura, diferentes a los ofrecidos por nuestro sistema. Ofrecen también lo que denominan agricultura de precisión; sensores de medición del grosor del tallo y diámetro del fruto, pluviometría, y estaciones meteorológicas. Desarrollan protocolos de comunicación adaptados al cliente, acceso remoto a través de Internet, y software para la elaboración de gráficas y almacenamiento de datos. En general se trata de una tecnología similar al Sistema de control agrícola, pero con un mayor desarrollo y orientado a grandes explotaciones.

Tras finalizar el estudio, vemos que nuestro sistema encajaría en un mercado nuevo, donde las principales ventajas serían las de operar de forma inalámbrica y con un mayor número de sensores que la media. Encontrado un hueco en el mercado de invernaderos de tamaño pequeño a medio.

Capítulo 3

3. Descripción funcional.

El sistema presentado, se ha diseñado con la intención de controlar y automatizar al máximo las cosechas de los invernaderos. Centrando el diseño en el ahorro energético, y el control remoto de los dispositivos para el control del ambiente en un invernadero.

3.1 Sistema total.

Un enfoque relevante a la hora de desarrollar el sistema, ha sido ver la mota-invernadero como una mota que ejecuta tareas distribuidas, como parte de la aplicación principal o una expansión de la misma. Se considera el extremo de una red de sensores que se encarga de captar información, sin procesamiento alguno de la misma.

En el tándem Mota-central más PC es donde reside la toma de decisiones del sistema, y por la tanto el elemento fundamental que se ramifica en la mota invernadero.

La red que permite la interconexión de ambos elementos, se basa en la radio como interfaz de comunicación. La radio se ayuda del protocolo ZigBee, que ofrece bajos consumos y bajas tasas de transferencia de datos, operando en la banda de los 2'4GHz. La red se empleara para un envío de datos escaso y en ciclos cortos de tiempo, por lo que este protocolo resulta muy apropiado para las características del sistema.

Mota-central y PC, se comunican por un sencillo cable USB, igualmente apto para los requisitos del sistema. Aquí destaca el papel de la mota-central, al tener las interfaces de radio y cable serie, lo que la convierten en el puente entre ambos canales de comunicación.

3.2 PC.

La Interfaz de usuario es un software escrito en Java y con librerías para la interpretación de TinyOS, es la encargada de la toma de decisiones y el cerebro del sistema. Una tarea primordial de la interfaz del PC es la sincronización de todo el sistema, pudiendo optar por valores expresados en milisegundos, para una mayor precisión en el control. Por razones de seguridad y para evitar paradas del sistema, la mota-central solo acepta valores de sincronización menores de 8.000 milisegundos, por lo que los valores mayores no se aceptan como parámetro de la interfaz. Otras tareas específicas son; la recepción de las variables del estado del invernadero; el cálculo de las acciones para corregir los valores que estén fuera del rango óptimo para la cosecha; y el envío de las acciones a la mota-central, encargada del envío al invernadero por su interfaz de radio.

Todas estas tareas son fácilmente visualizables en la consola de comandos, donde las variables recibidas y el código enviado para que la mota-invernadero realice las acciones necesarias, se muestran conforme se reciben y calculan respectivamente.

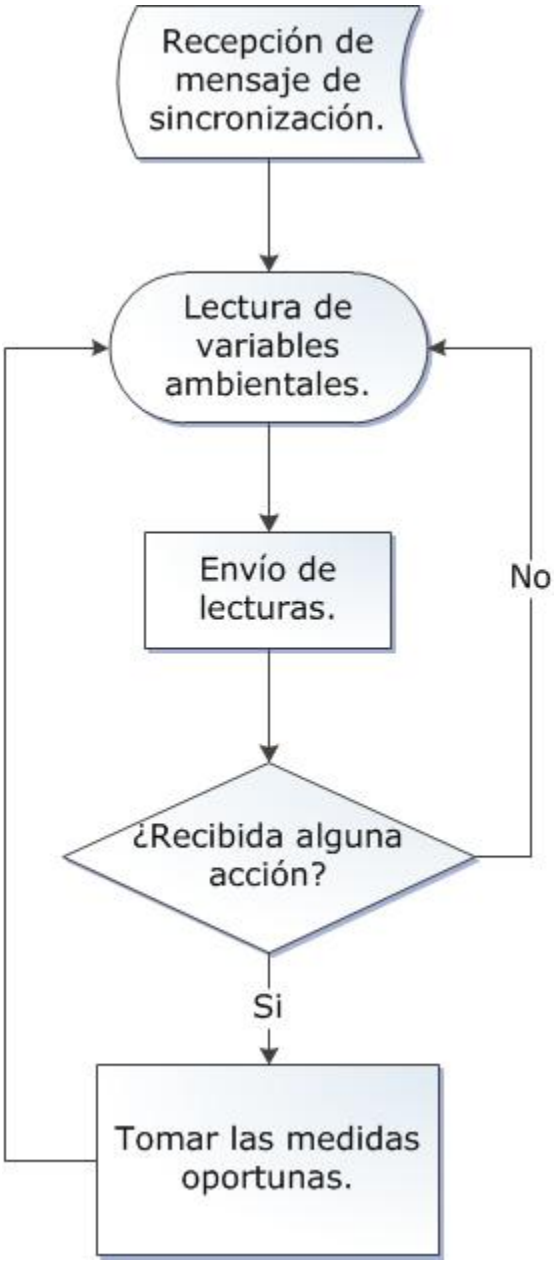

Figura 7. Diagrama de flujo de la aplicación Interfaz.

3.3 Mota.

Para las aplicaciones de la mota, es necesario distinguir entre las dos elementos de la red de sensores, mota-central y mota-invernadero.

- Mota-central:

La aplicación de la mota-central, ha sido diseñada para actuar como nexo de unión entre la red inalámbrica y el PC. Por lo que esta configurada con los módulos de comunicación inalámbrica y serie. Puede activar los leds para dar información adicional. Encenderá el led ámbar, para indicar el envío de mensajes con la información del estado del invernadero, desde la mota-invernadero hacia el PC, apagando el led una vez realizada la transferencia. Para informar del envío de mensajes con el código de acciones y el mensaje inicial de sincronización, desde el PC hacia la mota-invernadero, lo hará encendiendo y apagando el led rojo. El tercer led se usa para anunciar el modo de funcionamiento, ya sea modo automático o supervisado.

El modo de funcionamiento por defecto es el modo automático Este modo envía las órdenes a la mota-invernadero de manera automática, permitiendo una supervisión ininterrumpida de forma automática.

El modo supervisado se activa al pasar la llave magnética a la mota-central, podemos verificar el modo gracias al led verde de la mota, que se apagara para indicar el cambio de modo automático a supervisado. Este modo es útil si queremos comprobar el funcionamiento del sistema, o ratificar las órdenes que enviamos al invernadero. Por medio del botón USR podremos ordenar el envío de las órdenes a la mota-invernadero.

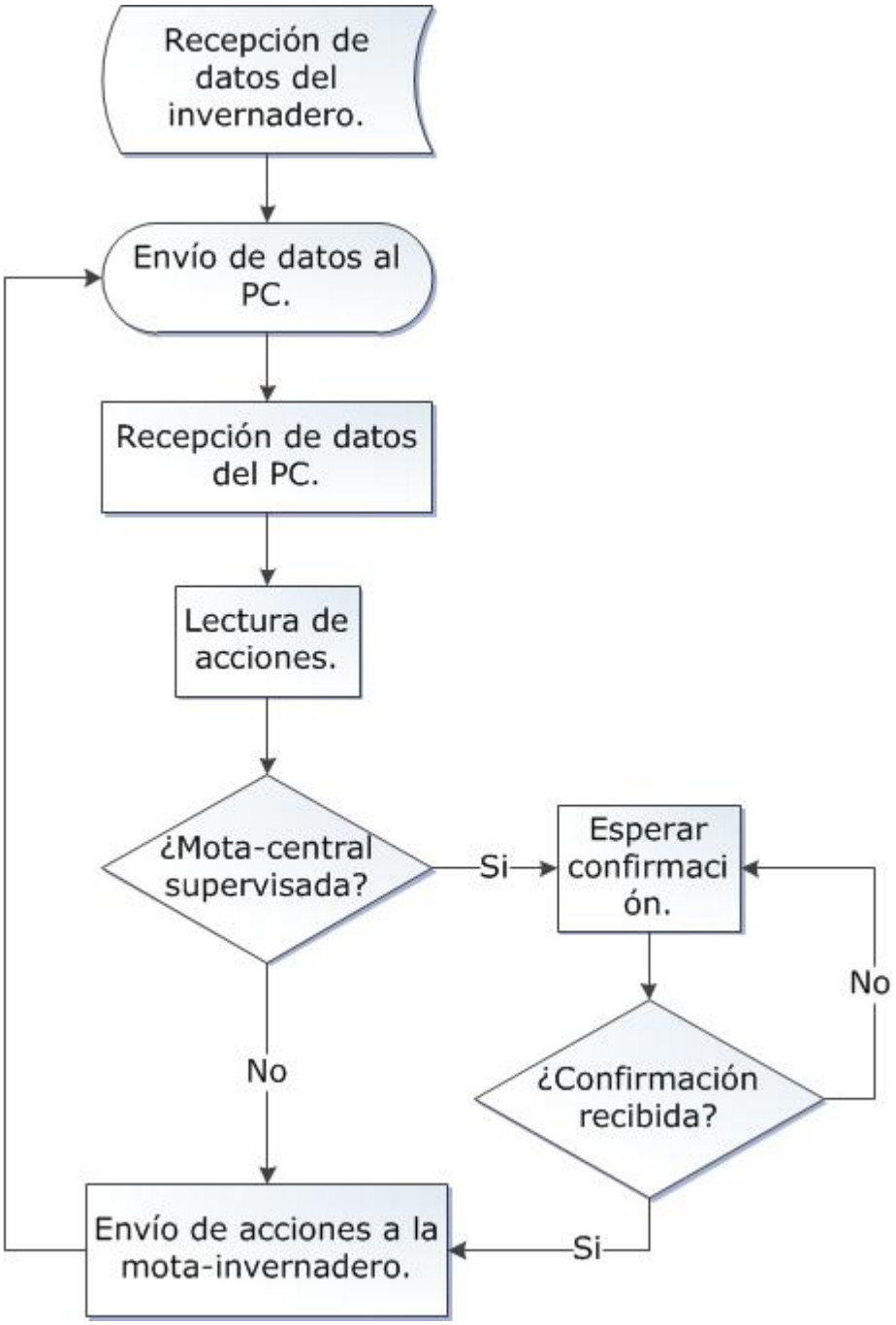

Figura 8. Diagrama de flujo de la aplicación de la mota-central.

- Mota-invernadero:

La aplicación de la mota-invernadero realiza sus tareas de lectura de sensores, y envío/recepción de mensajes. Los dos módulos más importantes con los que cuenta son el lector de sensores, y el de la comunicación vía radio. La mota-invernadero, al igual que la mota-central, muestra información de su estado por medio de los leds. Una vez iniciada la mota, podremos observar como todos los leds permanecen encendidos, hasta que enviemos el mensaje de sincronización. Una vez enviado los leds se apagan, indicando así que ha comenzado a hacer lecturas de los sensores.

Una vez sincronizada, podremos observar las alarmas detectadas por el PC gracias a los leds, siendo el led rojo el encargado de alertar sobre un exceso de temperatura; led verde para la temperatura baja; led ámbar para un nivel inadecuado de luz; y luz verde y roja para apagar los ventiladores de calor o frío. La idea es manejar estas alertas para contrarrestar las incidencias, mediante la activación o desactivación de los mecanismos necesarios.

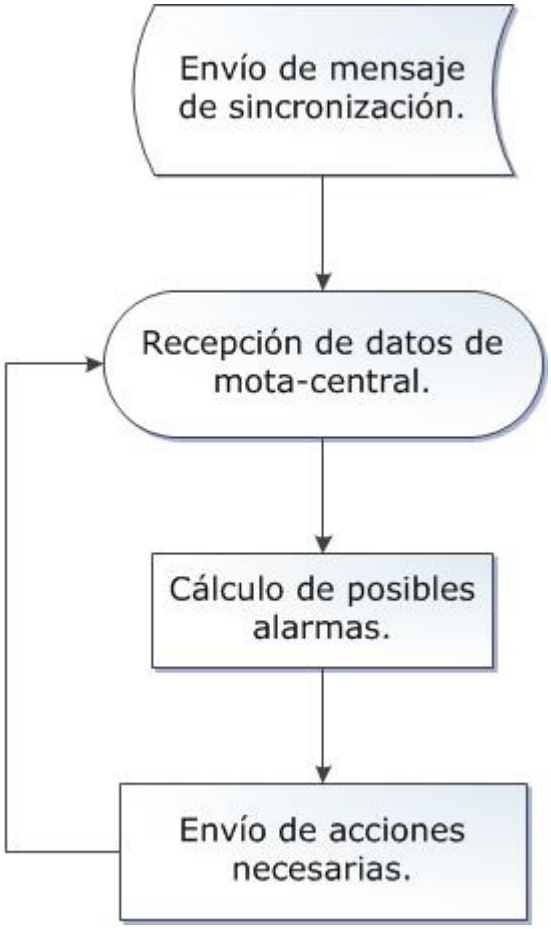

Figura 9. Diagrama de flujo de la aplicación de la mota-invernadero.

Capítulo 4

4. Descripción detallada.

En este capítulo, se presentarán las estructuras de las aplicaciones de todo el sistema desde un punto de vista más técnico. Se representan según la estructura propia del lenguaje NesC, haciendo hincapié en los módulos que conforman cada aplicación. Se trata por tanto de una exposición de la arquitectura del software, propia de TinyOS.

4.1 Sistema total.

El sistema total, es capaz de comunicar los diferentes nodos gracias a los interfaces de comunicación. Por ello es interesante entrar en detalle en la estructura de los mensajes. Para el Sistema de control agrícola, se han definido unas estructuras propias en base a las características de TinyOS. Estas características comunes se definen para todo mensaje emitido o recibido, independientemente de la estructura interna de su carga útil.

Cabecera del mensaje	Carga útil	Metadatos
11 bytes.	28 bytes.	7 bytes.

Figura 10. Estructura de los mensajes básicos TinyOS.

Es en el contenido de esta carga útil donde reside la singularidad de los mensajes del sistema. Podemos definir tres grupos de mensajes, el mensaje con el estado del invernadero **estado_msg**; el mensaje con el código de acciones computadas **codigoAcciones_msg**; y el mensaje con el valor de sincronización **temporizador_msg**. En la siguiente tabla se desglosan los contenidos de cada mensaje, junto con el código AM o identificador de cada mensaje.

Mensaje	Identificador AM	Variables contenidas	Longitud total
temporizador_msg	44	-miliseg (uint 8 bits)	8 bits.
estado_msg	42	-ventilador (bool 8 bits) -temperatura (uint 16 bits) -luz (uint 16 bits) -batería (uint 16 bits)	56 bits.
codigoAcciones_msg	43	-codigo (uint 8 bits)	8 bits.

Figura 11. Estructura de los mensajes del Sistema de control agrícola.

Si bien todas las variables contenidas en cada mensaje son bastante aclaratorias, conviene esclarecer el funcionamiento de la variable código. Se ha optado por codificar las órdenes, de manera que las sean descifradas en la mota-invernadero, siguiendo un código de valores. El código define las siguientes acciones:

- "0" para no realizar ninguna acción
- "1" para activar la ventilación refrigerada.
- "2" para indicar una iluminación inadecuada, y activar/desactivar la luz.
- "4" si se requiere activar la ventilación calefactada.
- "5" para apagar cualquier tipo de ventilación.

Estos códigos son calculados por la aplicación **Interfaz** residente en el PC, que pasaremos a explicar a continuación.

4.2 PC.

- Interfaz de usuario:

Esta es la aplicación que toma las decisiones sobre qué acciones se han de tomar, sincroniza todo el sistema a la temporización indicada, y guarda el rango de valores de cada variable ambiental considerada óptima para la cosecha. Programada en Java, incluye los paquetes necesarios para poder interpretar los mensajes de TinyOS antes descritos. Estos mensajes pueden ser exportados a otros entornos diferentes a TinyOS con el *MIG (Message Interface Generator),* con lo que tendremos no solo la información de su estructura, también dispondremos de métodos para acceder a su contenido, y trabajar con el.

Otros paquetes que exportamos para la aplicación, que permiten la interacción con el resto del sistema son:

- Message: Clase base de Java para la interpretación de los mensajes generados en TinyOS.
- MessageListener: La interfaz que nos notificara la llegada de mensajes TinyOS, desde la mota-central.
- MoteIF: Con esta interfaz podremos enviar y recibir los mensajes, vía serie o conexión TCP. En en el caso de la Interfaz de usuario, los enviaremos vía conexión TCP, cediendo la gestión de la conexión serie al *SerialForwarder,* la aplicación que nos retransmite los mensajes de la mota-central.
- Murceguillo: Esta clase permitirá la creación del PhoenixSource con sus parámetros de conectividad.
- PhoenixSource: Clase para crear un receptor y/o emisor de paquetes TinyOS.
- PrintStreamMessenger: Básicamente la usaremos para la escritura de datos por un canal.

Todas estas clases e interfaces permiten la integración de la aplicación con TinyOS, de manera que pueda enviar el código de acciones en un formato legible para el resto de elementos del sistema.

4.3 Mota.

Ambas motas aunque con tareas dispares, comparten una misma configuración de hardware. Por lo tanto sensores, microprocesador e interfaces de comunicación son idénticos en las dos motas.

La lista de sensores instalados es la siguiente:

- Sensor fotoconductivo **PDV-P9003-1**: Sensor que reacciona a la luz detectada, indicando un voltaje distinto para cada nivel lumínico.
- Sensor analógico de temperatura **MCP9700**: Es un sensor capaz de medir un rango de temperatura desde los 0º a los +70º.

– Sensor de efecto-Hall **BU52011HFV**: Sensor para detectar los cambios producidos por la presencia de un campo magnético. Se usará para el control del accionamiento de ventiladores en la mota-invernadero, y para la detección de la llave magnética en la mota-central.

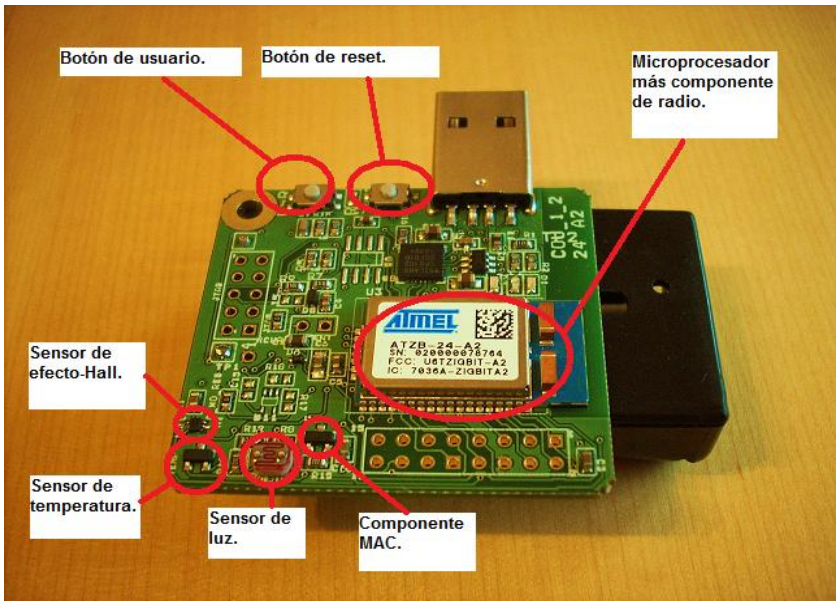

Figura 12. Detalle de los componentes hardware más importantes.

Además de los sensores podemos ver otros elementos no menos importantes como son microprocesador y el componente de radio. Como microprocesador dispone de un ATmega1281V de 8 bits, fabricado por la compañía ATMEL. Capaz de operar a frecuencias de 4MHz y con una memoria SRAM de 8 Kbytes, limita su capacidad de computar datos a tareas sencillas. Cuenta también con el componente de radio de bajo consumo AT86RF230 con antena incorporada. Este componente opera en la frecuencia de 2'4GHz para comunicaciones por radiofrecuencia, compatible con el protocolo ZigBee. Ambos componentes (microprocesador y componente de radio) incorporados en un mismo módulo inalámbrico llamado ATZB-24-A2. Otro componente de interés es el componente para almacenar su dirección MAC.

- Mota-central:

Como hemos visto antes, la mota-central es la encargada de hacer de puente entre PC y mota-invernadero. Para esta tarea necesita unos módulos determinados, para tratar los mensajes serie y radio, y para el control del modo de operación.

Los componentes de comunicaciones del modulo **EstacionBase** están parametrizados, lo que supone un componente destinado a cada tipo de mensaje en particular, sea el mensaje de temporización, estado o código de acciones. La lista se completa con el uso de componentes destinados al control los leds, control del encendido y apagado de componentes, y componentes para el manejo de los paquetes de comunicación de TinyOS.

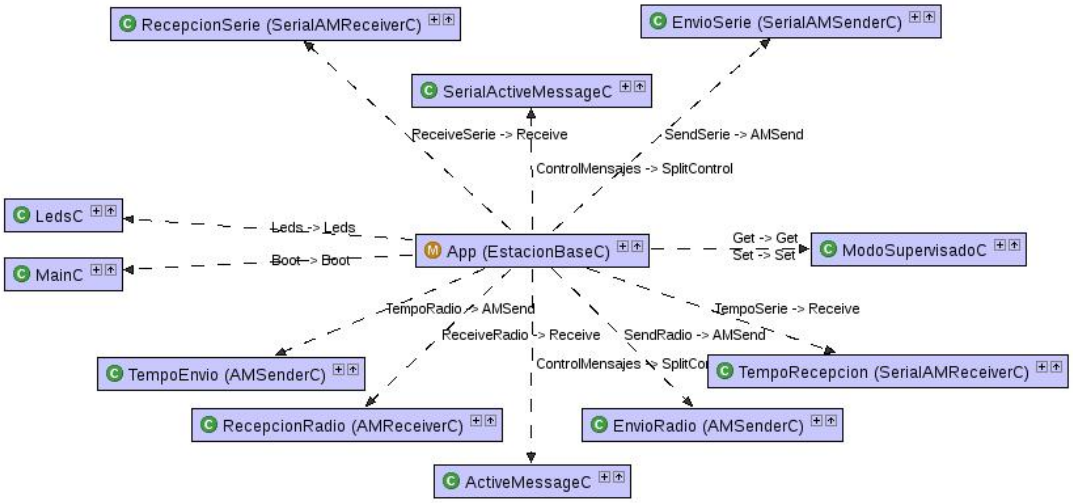

Figura 13. Componentes y conexiones del módulo EstacionBase.

El modulo **ModoSupervisado** sirve para controlar el uso de llave magnética, por lo que en su lista de componentes figuran los dedicados a este control, como el control de interrupciones de hardware para el uso del botón de usuario o uso del efecto-Hall, control de temporizador del muestreo de estado, y control de los leds.

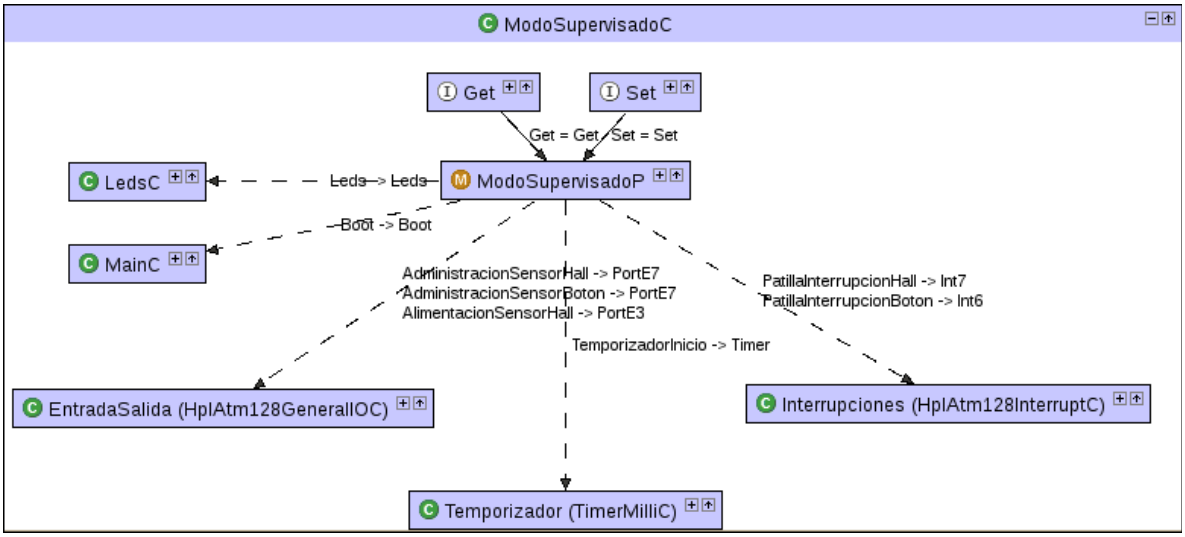

Figura 14. Componentes y conexiones del módulo ModoSupervisado.

Se hace uso de una librería que contiene la estructura de los mensajes de temporización, estado y código de acciones, común en ambas motas. La mota-central implementa en su modulo EstacionBase, una librería adicional para evitar el colapso del sistema, esta librería llamada **Watchdog**, establece un sistema de reinicio en caso de no obtener respuesta del sistema, evitando un parada del sistema.

- Mota-invernadero:

La mota-invernadero encargada de medir las variables ambientales, y recibir las órdenes, cuenta con varios módulos para satisfacer estas tareas. El módulo principal es **Invernadero,** está conectado con los demás módulos por interfaces que facilitan el traspaso de información entre componentes. Los componentes específicos de radio, al igual que en la mota-central, están parametrizados para la recepción del tiempo de sincronización o el código de acciones, y para el envío del estado del invernadero. Cuenta con componentes para el control del encendido, y para el control de temporización.

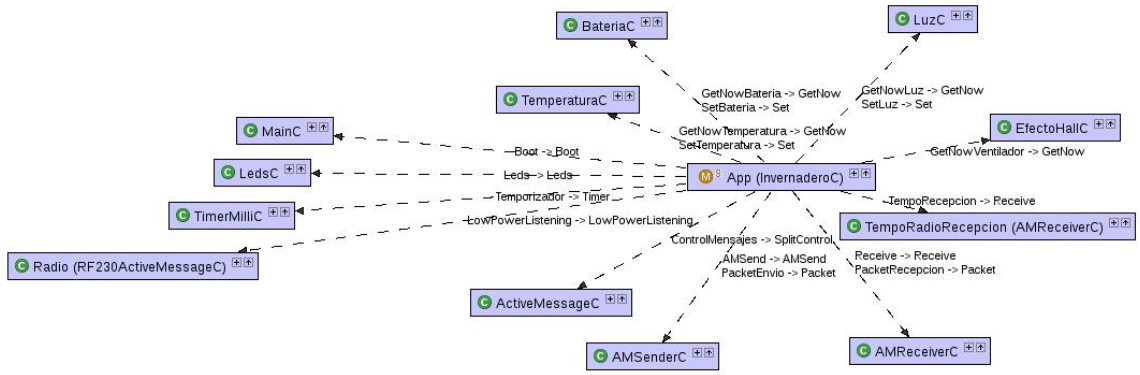

Figura 15. Componentes y conexiones del módulo Invernadero.

Una interfaz de especial es la **LowPowerListening**. Conectada al componente específico de la radio de la mota, permite el consumo eficiente de energía. Una vez configurado en base al temporizador, permite el uso eficiente de las baterías.

Los componentes que facilitan la lectura de sensores (Luz, temperatura, y autonomía de la batería), están conectados a **LecturaADC**. LecturaADC es el modulo principal que accede al registro de lecturas y devuelve los datos a cada módulo de sensor. En las siguientes imágenes mostramos como ejemplo el modulo **Luz** y su interacción con el modulo LecturaADC.

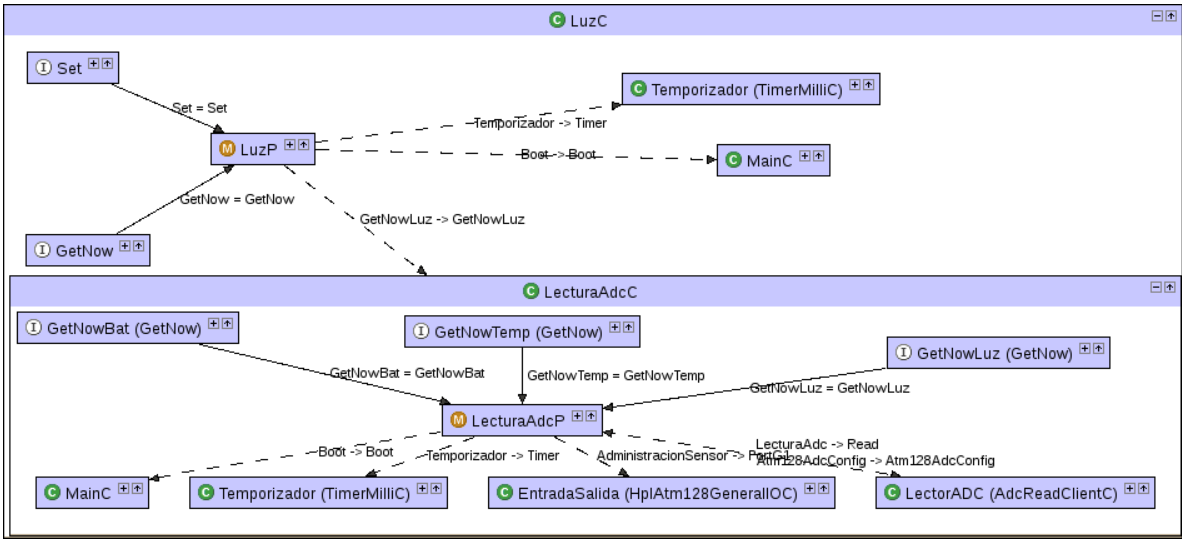

Figura 16. Detalle de la conexión del módulo Luz y el módulo LecturaADC.

El módulo EfectoHall, no sigue la misma configuración que el resto de módulos de sensores. Esto es debido a que no se ha pedido calibrar correctamente, para convivir con el modulo LecturaADC, dado que es un componente sensible en cuanto el funcionamiento para detectar el estado de los ventiladores.

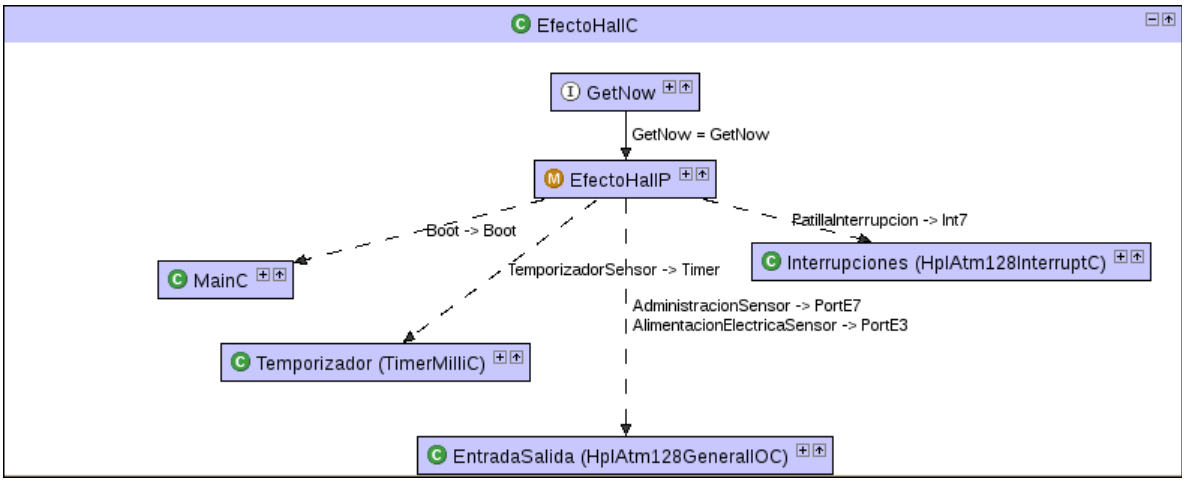

Figura 17. Componentes y conexiones del módulo EfectoHall.

Como en el caso de la mota-central, en la mota-invernadero existe una librería idéntica con la información referente a los mensajes, su nombre es Invernadero.h, aunque la información contenida es exactamente la misma.

Capítulo 5

5. Viabilidad técnica.

El sistema es técnicamente viable. La tecnología es accesible, no requiere grandes desembolsos por parte del cliente final, y con ella se consiguen resultados de forma sencilla, rápida, y efectiva. Tanto su desarrollo, como instalación y mantenimiento no resultan complicados. El desarrollo del sistema o sistemas similares, a partir de los conocimientos adquiridos, no debería alargarse más allá de los 3 meses, por lo que el tiempo de desarrollo no es excesivo. Tampoco los trabajos de instalación son ningún agravante, hay que recordar que la comunicación vía radio, facilita este aspecto, así como su mantenimiento. Tampoco necesitaremos antenas, repetidores u otros elementos de comunicación por radio adicionales.

El mercado lejos de estar debilitado, es un mercado en auge, lo cual abre la puerta a futuras mejoras. Es seguro, que podremos disponer de mayores y más eficientes componentes y motas, que mejoren la experiencia obtenida con el sistema. Por lo que respecta al momento actual, la porción del mercado tecnológico al que se destina, no está saturado ni es tan competitivo como otros, por lo que el sistema desarrollado encontrará un hueco tecnológico importante por cubrir.

Por lo tanto se considera viable implementar el Sistema de control agrícola, tanto por la tecnología requerida, como por las ventajas que supone.

Como puntos fuertes destacan:

- El sistema ofrece evidentes mejoras en el trabajo diario de manera eficaz. Resultados efectivos en el control de las cosechas, mejores y con mayor rendimiento.
- Sencillez en el manejo del sistema, sin grandes conocimientos técnicos y de fácil mantenimiento, no requiere mantenimiento de una infraestructura propia.
- El hecho de disponer de interfaz de red, añade al sistema la posibilidad de expandirse, abarcando nuevas posibilidades. Cosechas de mayor tamaño, sistema fácilmente ampliable a más motas o invernaderos.

- El valor añadido que supone utilizar el aire como medio en cuanto a costes y sencillez, en lo referente a infraestructuras. Se abarata el precio final de la instalación y del mantenimiento.
- Al ser un mercado en auge, garantiza mejoras y nuevas posibilidades. Tecnología en crecimiento y mercado en estado de expansión.

Como puntos débiles podemos señalar:

- La falta de algunos sensores convenientes para el control de cosechas. Sensor de humedad, necesario para el control del riego.
- Configuración complicada para otras cosechas, o falta de una base de datos extensa de diferentes cosechas. El sistema actual solo considera un tipo de cosecha.

Capítulo 6

6. Valoración económica.

Una vez finalizado el Sistema de control agrícola, estudiaremos el costo total de trasladar la idea a un cliente final interesado en el producto. Se analiza el presupuesto, como una aproximación a un hipotético contrato para implantar un invernadero inteligente. Los conceptos a contabilizar son los siguientes:

- Equipamiento, donde incluimos motas y PC.
- Desarrollo del sistema, incluyendo mano de obra calculada en euros/hora y análisis de las particularidades del cliente.
- Pruebas con el cliente, valorado en euros/hora.
- Mantenimiento de la aplicación, como paquete básico para posibles fallos posteriores a la implantación.

En la siguiente tabla presentamos el precio total del sistema y desglosamos el precio de cada concepto.

Concepto	Cantidad del concepto	Precio	Precio total
Equipamiento			
PC	1	500	500
Motas	2	35	70
Desarrollo			
Estudio del caso	8	20	160
Mano de obra	160	20	3200
Pruebas del sistema			
Mano de obra	40	20	800
Mantenimiento			
Mantenimiento básico	1	200	200
Total general			4930

Figura 18. Desglose de precios del Sistema de control agrícola.

El trabajo se estima de una duración aproximada de mes y medio, sin contar el mantenimiento básico por seis meses, necesario para asegurar el funcionamiento en una cosecha.

Tras la implantación del sistema, podemos esperar una mejora en la calidad de la cosecha y el control de pérdidas ocasionadas por heladas, olas de calor y demás situaciones adversas. La amortización calculada a un año queda en unos 410€ mensuales aproximadamente, una cifra asumible si contemplamos que las pérdidas por hectárea pueden suponer unos 3.000€ por cosecha. A partir de estos datos, consideramos que el sistema resultaría rentable en grandes cosechas, donde la recuperación de la inversión sería más rápida.

Capítulo 7

7. Conclusiones.

Los objetivos más importantes se han conseguido, estos eran objetivos básicos para el control de toda cosecha:

- Recogida de información medio-ambiental en el interior del invernadero.
- Monitorización de sensores.
- Envío y recepción de información.
- Implementación del modo automático y supervisado.
- Desarrollo de aplicación de PC para el tratamiento de la información recibida.

Los objetivos secundarios, implicaban una base de datos con diferenetes cosechas. Por falta de tiempo para llevarlo a cabo, no se ha podido ultimar este detalle. Se ha optado por centrar los esfuerzos en los principales objetivos, para el control de la cosecha.

7.1 Conclusiones.

Las motas utilizadas, son aparatos sumamente versátiles capaces de infinidad de tareas, asequibles, y muy útiles para todo tipo de tareas. No implica un gran coste su instalación y hacen muy cómodas las tareas repetitivas, con el ahorro en costes que ello supone. Como todos los sistemas empotrados en general, realizan tareas de forma eficiente y añaden el factor de las comunicaciones inalámbricas, pudiendo operar en red para tareas remotas lo cual es una ventaja sustancial.

TinyOS cubre muchas de las posibilidades que podamos plantearnos, por lo que se muestra como una opción de código abierto importante en el mundo de los sistemas empotrados. El nivel de abstracción conseguido con los componentes y las interfaces definidas, facilitan mucho el trabajo. También destaca el funcionamiento enfocado a eventos, que concuerda con las necesidades de los sistemas empotrados.

7.2 Propuesta de mejoras.

El sistema carece de un sensor que facilite el control del riego, y este es un elemento fundamental a controlar en toda cosecha. Existen riegos automáticos que cumplen bien estas tareas y se consideran un dispositivo muy extendido en los invernaderos. Sin embargo, el riego puede escapar al control del sistema, lo cual es una carencia a tener en cuenta. No obstante, su implementación en el sistema no está carente de dificultades. Los sensores de humedad analizados, requieren muchas veces su implantación en el interior de la tierra. Si se necesita un control de riego exhaustivo, se requiere a su vez una reconfiguración de la mota utilizada u otra mota completamente nueva que añada este sensor.

Dotar la mota de carcasa protectora seria interesante. No hay que obviar que su instalación en un entorno expuesto a humedades, podría dañar la circuitería, oxidar o incluso hacer lecturas erróneas del entorno. Estos inconvenientes, conseguirían rebajar la vida útil o inutilizar el dispositivo completamente.

Otra mejora, aludida con anterioridad es la base de datos. Esta permitiría la manipulación de varias cosechas y hacerlo simultáneamente. El concepto tendría sentido si fuese accesible desde la aplicación del PC, ya que centralizamos en él la mayor carga computacional. Se podría igualmente adjudicar una cosecha a cada mota, algo posible si tenemos en cuenta el chip de MAC que permitirá realizar estas adjudicaciones. Además de ahorrar la incomodidad de configurar la aplicación para cada cosecha, conseguiríamos explotar al máximo las capacidades del sistema, ampliando su radio de acción.

7.3 Autoevaluación.

El mundo de los sistemas empotrados a pesar de ser complejo, es muy atractivo. Al indagar en él he visto muchas posibilidades, es la herramienta justa para tareas muy específicas de la manera más eficiente. He conocido un potencial enorme en estos aparatos, lo más alentador para seguir estudiándolos es poder hacer un uso más cotidiano, útil y cercano de la tecnología actual. Personalmente me ha enriquecido mucho, los conocimientos a los que aspiraba me han animado a seguir investigando en los sistemas empotrados.

Como punto negativo, quizás el tiempo no me haya permitido aprovechar al máximo las posibilidades de la asignatura, pero siempre podré optar a volver con ello sin mayor diligencia que las ganas de aprender.

Capítulo 8

8. Glosario.

Sistema empotrado: Sistema computacional concebido para tareas concretas, dotado de los elementos fundamentales para tal fin.

Mota: Sistema empotrado diseñado para operar como nodo en una red de sensores, dotado de sensores e interfaces de comunicación.

Nodo: Punto o dispositivo conectado a una red de telecomunicaciones.

ZigBee: Protocolo de comunicaciones inalámbricas ideado para bajas tasas de transferencia y bajo consumo energético

Efecto-Hall: Alteración de la corriente eléctrica por la presencia de un campo magnético.

SRAM: Un tipo de memoria RAM llamada estática, para almacenar los datos durante el tiempo de ejecución.

Frecuencia de procesamiento: Habitualmente medido en hercios (Hz), mide el número de operaciones realizadas por segundo.

MAC: Identificador de red para interfaces de comunicación, expresada en 6 bloques de caracteres hexadecimales.

Watchdog: Mecanismo de postergación de reinicio para evitar la parada o bloqueo de un sistema.

Llave magnética: Llave dotada de un imán, lo suficientemente potente como para activar el sensor de efecto-hall.

Modelo en espiral: Modelo para el desarrollo de proyectos, caracterizado por un bucle de actividades, la continua revisión de los resultados obtenidos y la elasticidad de su ciclo de vida.

TinyOS: Sistema operativo enfocado a nodos en redes de sensores.

NesC: Dialecto del lenguaje de programación C, utilizado para el desarrollo de aplicaciones en TinyOS.

Arquitectura del software: Estructura del software o abstracción para representar una aplicación.

Entorno de programación: Aplicación informática diseñada específicamente para la programación de software.

Plugins: Mejoras o complementos para aplicaciones informáticas

Meshprog: Programa para instalar las aplicaciones elaboradas para TinyOS en motas.

SerialForwarder: Aplicación escrita en el lenguaje Java para la comunicación entre motas en entorno TinyOS y un PC.

Capítulo 9

9. Bibliografía.

- **URL's:**

– Protocolos de comunicación:

Zigbee: http://www.zigbee.org/Specifications/ZigBee/Overview.aspx

Dash7: http://www.dash7.org/index.php?option=com_content&view=article&id=9&Itemid=11

Estándar KNX: http://www.knx.org/knx-standard/interworking/

BLE: http://www.bluetooth.com/Pages/Low-Energy.aspx

– Sistemas operativos en los sistemas empotrados:

Análisis y ejemplos: http://onsoftware.softonic.com/sistemas-embebidos-y-usos-cotidianos

– Ejemplos de sistemas empotrados:

Control del tráfico y la polución: http://www.libelium.com/Smart_City_Air_Quality_Urban_Traffic_Waspmote/

Monitorización de un volcán: http://www.eecs.harvard.edu/~mdw/proj/volcano/tungurahua2007/Tungurahua%202007%20Sensor%20Deployment.html

Monitorización de la calidad de las aguas: http://flux.ocean.washington.edu/riser_web/web-pubs/chem-reviews.paper.pdf

- Motas del mercado:

  Comparativa de motas: http://ubimon.doc.ic.ac.uk/bsn/m206.html

  Wiki de la COU 24: http://cv.uoc.edu/app/mediawiki14/wiki/Inici24

- Estudio de mercado:

  Link4: http://www.link4corp.com/products/igrow-environmental-controllers.html

  TNO: http://www.tno.nl/content.cfm?context=thema&content=markt_product&laag1=896&laag2=346&laag3=397&item_id=888

  Advanticsys: http://www.advanticsys.com/services/agricultural-environment-monitoring/?lang=es

- Documentación sobre el sistema operativo de las motas:

  Wiki de TinyOS: http://docs.tinyos.net/tinywiki/index.php/Main_Page

- **Recursos bibliográficos:**

Peckol, James K..(ed.) (2008). ***Embedded systems: a contemporary design tool*** (1.ª ed.). Hoboken, NJ: John Wiley.

Levis, Philip; Gay, David.(eds.) (2009). ***TinyOS Programming*** (1.ª ed.). New York: Cambridge University Press.

Capítulo 10

10. Anexos.

10.1 Compilación e instalación.

Se entregan los códigos necesarios para cada mota, más la aplicación Java “Interfaz” en tres archivos comprimidos. Es necesario descomprimir cada archivo en carpetas separadas.

La compilación es sencilla si partimos de un entorno preparado para TinyOS:

1. Nos situamos en la carpeta que contiene el código de cada mota e introducimos el comando **make cou24.** Esto creará el ejecutable que más adelante instalaremos en las motas.
2. El comando **meshprog -f build/cou24/main.srec -t /dev/ttyUSB0** realizará esta instalación. Es importante comprobar que efectivamente la mota corresponde al **USB0**, podemos asegurarnos de esto con el comando **ls /dev/ttyUSB0** que nos mostrará una breve descripción del dispositivo. Si no fuera así, deberíamos incrementar el número hasta verificar el número del puerto USB, **ls /dev/ttyUSBX.** También es importante resetear la mota antes del envío, basta con pulsar el botón de reset para ello.

Siguiendo estos pasos, ya tendremos las motas configuradas y listas para el uso.

```
jorge@jorge-laptop:~/workspace/Proyecto/src2$ make cou24
mkdir -p build/cou24
    compiling InvernaderoAppC to a cou24 binary
ncc -o build/cou24/main.exe  -Os -fnesc-separator=__ -Wall -Wshadow -Wnesc-all -target=cou24 -fnesc-cfile=build/cou24/app.c -
board= -DDEFINED_TOS_AM_GROUP=0x22 --param max-inline-insns-single=100000 -DLOW_POWER_LISTENING -DDELAY_AFTER_RECEIVE=500 -DI
DENT_APPNAME=\"InvernaderoAppC\" -DIDENT_USERNAME=\"jorge\" -DIDENT_HOSTNAME=\"jorge-laptop\" -DIDENT_USERHASH=0x10b39c32L -D
IDENT_TIMESTAMP=0x4fccd909L -DIDENT_UIDHASH=0x66e9a738L -fnesc-dump=wiring -fnesc-dump='interfaces(!abstract())' -fnesc-dump=
'referenced(interfacedefs, components)' -fnesc-dumpfile=build/cou24/wiring-check.xml InvernaderoAppC.nc -lm
/opt/tinyos-2.x/tos/chips/rf2xx/rf230/RF230RadioC.nc:169:3: warning: #warning "*** USING LOW POWER LISTENING LAYER"
    compiled InvernaderoAppC to build/cou24/main.exe
           13964 bytes in ROM
             579 bytes in RAM
avr-objcopy --output-target=srec build/cou24/main.exe build/cou24/main.srec
avr-objcopy --output-target=ihex build/cou24/main.exe build/cou24/main.ihex
    writing TOS image
jorge@jorge-laptop:~/workspace/Proyecto/src2$ meshprog -f build/cou24/main.srec -t /dev/ttyUSB0
reading from file build/cou24/main.srec
using tty /dev/ttyUSB0
sending build/cou24/main.srec -> /dev/ttyUSB0

Opened file build/cou24/main.srec
Opened device /dev/ttyUSB0
.....,[i??&]
Starting transmission.
finished!
jorge@jorge-laptop:~/workspace/Proyecto/src2$
```

Figura 19. Ejemplo de compilación e instalación de la mota-invernadero.

Para la compilación del programa de Java:

1. Nos situamos en la carpeta donde se encuentra el archivo Interfaz.java.
2. Compilamos con el comando **javac Interfaz.java**.

Automáticamente creará la clase Interfaz.class, con lo que ya tendremos el ejecutable.

10.2 Monitorización del sistema:

Podemos conectar la mota-central a un PC y con ayuda de la aplicación SerialForwarder verificar el correcto funcionamiento del sistema.

1. Verificamos el puerto USB al que tenemos conectada la mota-central. Podemos hacerlo tal como se indica en la fase de compilación e instalación.
2. Una vez conectada, lanzaremos el comando **java net.tinyos.sf.SerialForwarder -comm serial@/dev/ttyUSBX:19200.**

3. Nos situamos en la carpeta con el ejecutable de la aplicación “Interfaz”, y por medio del comando **java Interfaz XXXX**, donde las X's representan los milisegundos de sincronización, iniciamos el sistema. Conviene destacar que el sistema no acepta una sincronización mayor de 8.000 milisegundos, por lo que si intentamos superar este tiempo se configurará automáticamente a 8.000 milisegundos.

Hecho esto, ya habremos lanzado la aplicación para visualizar las variables del invernadero y enviar las acciones de control. Podremos ver, el envío y recepción de datos entre ambas motas en el terminal de linea de comandos.

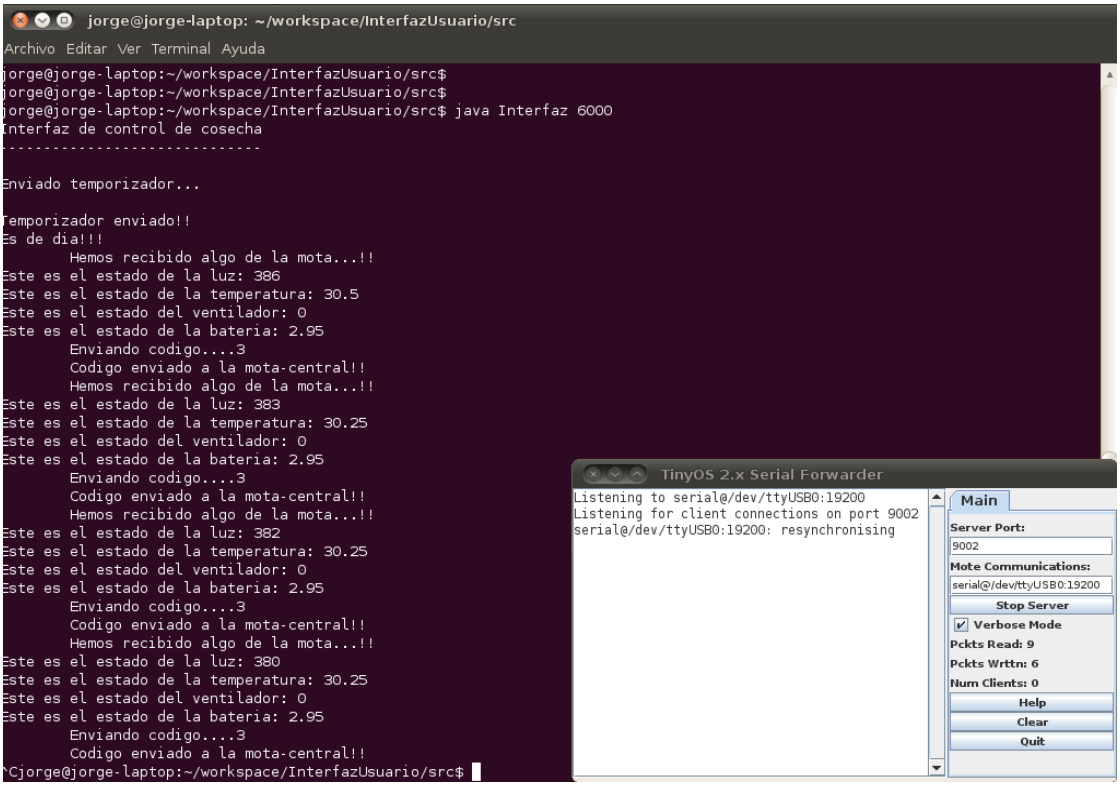

Figura 20. Ejecución de la aplicación Interfaz sincronizada a 6 segundos, junto con el Serial Forwarder.

Para cambiar el modo de operación de la mota:

1. Bastará con pasar una llave magnética sobre la mota-central, si la mota cambia a modo supervisado, lo notificará apagando el led verde.
2. En este modo de operación, los mensajes con el código de acciones esperan la confirmación para poder ser enviados a la mota-invernadero.
3. Podemos ratificar el envío de la acción pulsando el botón de usuario.
4. Si queremos reactivar el modo automático, tendremos que volver a pasar la llave magnética por encima de la mota.

10.3 Códigos de luces:

Mota-invernadero:

- Led rojo+verde+ámbar: Mota sin sincronizar; pendiente de recepción del mensaje inicial de sincronización.
- Led verde: alarma de temperatura baja detectada; recibida orden de encender la ventilación de aire caliente.
- Led roja: Alarma de temperatura alta detectada; recibida orden de encender la ventilación de aire frío.
- Led rojo+verde: Alarma por ventilador encendido; apagar el ventilador, temperatura correcta.
- Led ámbar: Alarma de iluminación inadecuada; cambiar estado de la iluminación artificial, encender o apagar.

Figura 21. Mota-invernadero recién iniciada, a la espera del mensaje de sincronización.

Mota-central:

- Led verde: Modo automático en funcionamiento si lo vemos encendido y supervisado si lo vemos apagado.
- Led amarillo: Recibido mensaje de estado del invernadero al encender el led y enviado al PC al apagarse.
- Led rojo: Se encenderá en el momento de recibir el código de acciones de PC y se apagara en el momento de entregar el mensaje a la mota-invernadero.

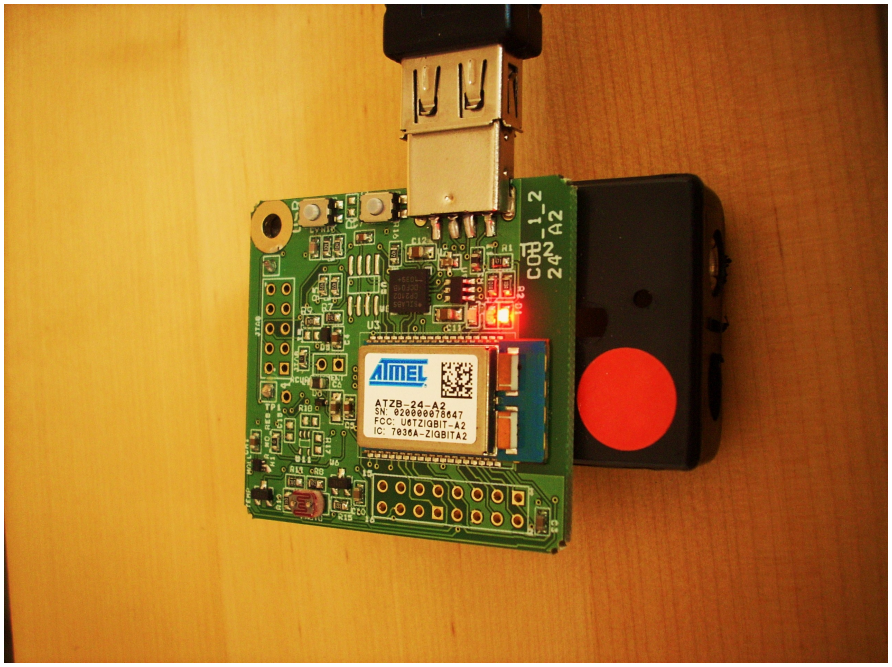

Figura 22. Mota-central operando en modo supervisado, a la espera de recibir confirmación de envío.