

Hacia una aproximación estructurada al diseño de redes de sensores inalámbricos (2ª Parte)

Por Ilya Bagrak, MeshNetics

Artículo cedido por Next For S.A., distribuidor de MeshNetics en España



www.nextfor.com



www.meshnetics.com

La parte 1 del artículo comenzó con una introducción a las redes de sensores inalámbricos, y examinó los factores relativos a la organización, tamaño y throughput.

La parte 2 del artículo de trucos y técnicas de diseño de redes de sensores inalámbricos continúa con una mirada a la vida de la batería, los periféricos, la seguridad y la puesta en marcha (commissioning) de las redes.

Vida de la batería

Una de las principales promesas de las redes basadas en 802.15.4 es la promesa de un bajo consumo energético y una vida útil de varios años sin cambio de batería. Esta promesa se hace posible por el hecho de que el estándar 802.15.4 define un dispositivo de función reducida (RFD), que puede tanto participar en la red como dormir durante largos periodos de tiempo, dando como resultado un ciclo de trabajo especialmente bajo y una larga duración de la batería.

Las redes ZigBee diferencian entre tres tipos de roles de dispositivos (o nodos). No es de extrañar que los requisitos de consumo energético dependan del tipo de dispositivo, es decir, las restricciones fundamentales impuestas por la especificación ZigBee y la forma en la que el dispositivo es alimentado en una aplicación particular, por ejemplo alimentación con una batería o directamente de la red eléctrica.

Hay tres tipos de dispositivos en una red ZigBee:

1. Coordinador.

Este es el nodo principal de toda la red ZigBee, responsable de iniciar la red y de gestionarla en cierta medida. El coordinador recibe generalmente la alimentación de la red eléctrica, porque genera y recibe gran cantidad de tráfico de radio y debe estar siempre activo para mantener el funcionamiento permanente de toda la red.

2. Router.

Estos nodos son responsables de la transmisión de los datos a través de múltiples saltos en la red, ampliando la cobertura y flexibilidad de red. Puesto que también utilizan el chip de radio activamente, los routers también reciben la alimentación de la red eléctrica en la mayoría de los escenarios.

3. Dispositivo final.

Este es el que lleva a cabo el trabajo, encargándose de obtener los datos a intervalos regulares y/o bajo demanda. Estos nodos pueden ser, y en la mayoría de los casos son, alimentados con baterías y pasan la mayor parte del tiempo dormidos (en modo espera), despertando bien por un temporizador o por un evento externo (interrupción) para obtener los datos de las fuentes externas (tales como sensores, etc) y enviarlos al nodo recolector, que por lo general es el coordinador. Los dispositivos finales también pueden recibir mensajes de control que se encuentran encolados en el router más cercano, y son recibidos en el intervalo de tiempo en el que el dispositivo se encuentra despierto.

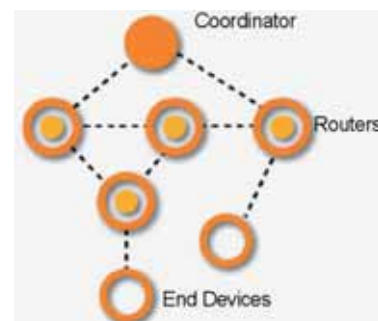
El escenario típico del dispositivo final es el siguiente:

a) El nodo está en modo de ahorro de energía (dormido) hasta que ocurre un evento de despertar o se recibe una interrupción externa

b) El nodo pasa a pleno funcionamiento, opcionalmente re-estableciendo la conexión con la red si esta se ha perdido durante el sueño, obtiene los datos y, a continuación, los envía al nodo padre a través de la radio (si la red aún se encuentra disponible).

c) El nodo apaga el circuito de la radio y vuelve al modo de ahorro de energía.

d) Repetir (a) – (c)



Tipos de dispositivos en una red ZigBee.

Las típicas fases de la operación de un dispositivo final componen lo que comúnmente se conoce como ciclo de trabajo. Cuando se diseña un sistema para tener una larga duración de las baterías la principal preocupación son los dispositivos finales, ya que los otros nodos de la red se espera que obtengan la energía de la red eléctrica. Para los dispositivos finales, el ciclo de trabajo es de suma importancia. Es el parámetro que afecta directamente a la vida de la batería, y el único parámetro que ingeniero del sistema puede manipular en el software para lograr una mayor operación de la batería. El ciclo de trabajo es la proporción de tiempo que un dispositivo está encendido de su tiempo total de sueño. Por ejemplo, si un dispositivo duerme durante 99 segundos, y 1 segundo lo pasa despierto, entonces el ciclo de trabajo de dicho dispositivo es del 1%.

Las fórmulas utilizadas a continuación no tienen en cuenta varios parámetros importantes, por ejemplo, la auto-descarga de la batería y las fugas internas, que afectan al consumo energético real y al tiempo de vida de la batería. Por lo tanto, la fórmula sólo debe utilizarse como una guía aproximada para la estimación de la capacidad de la batería en la aplicación objetivo. Además, la fórmula calcula la vida útil de la batería basándose únicamente en la energía consumida por el propio nodo de la red

(se ha utilizado el módulo ZigBit de MeshNetics como referencia). En aplicaciones reales, la energía consumida por el propio nodo es a menudo eclipsada por los periféricos conectados, por lo tanto, cualquier diseño debe tener en cuenta la potencia total consumida por todos los componentes activos en los estados de sueño y activo.

Los siguientes cálculos muestran cómo calcular la vida útil de la batería en función de la capacidad de la batería y del ciclo de trabajo.

Supongamos:

La capacidad de la batería:
 $W = 2000\text{mAh}$

Por requisitos de voltaje, los diseños pueden requerir más de una batería conectada en serie. Por ejemplo, una tensión mínima necesaria de 1,8V requerirá al menos de dos pilas de tamaño AA de 1,25V cada una. A medida que las pilas se gastan su tensión disminuye, por lo tanto, una batería de 1,8V durará menos que dos pilas de 1,25V cada una.

Definidos:

- Vida total de la batería: T_{work} (segundos)
- Tiempo pasado en el modo activo: T_{awake} (segundos)
- Intervalo de sueño: T_{sleep} (segundos)
- Consumo de energía en modo activo (módulo ZigBit únicamente, sin circuitos externos o periféricos): $I_{awake} = 19\text{mA}$
- Consumo de energía en modo dormido (módulo ZigBit únicamente, sin circuitos externos): $I_{sleep} = 0,006\text{mA}$
- Período de trabajo (sólo un ejemplo, varía según cada caso): $T_{period} = 3,6\text{seg}$

Ahora vamos a encontrar T_{work} :

Aquí puede ver que el factor más importante es T_{awake} (duración del ciclo activo, cuando el consumo de energía es máximo), ya que T_{sleep} es constante en la mayoría de los casos. Para ilustrar esto, vamos a calcular el tiempo de trabajo para 2 valores de T_{awake} , 20ms y 200ms.

Para $T_{awake} = 200\text{ms}$:

$$T_{sleep} = T_{period} - T_{awake} = 3,6 - 0,2 = 3,4\text{seg}$$

$$T_{work} = \frac{2000}{\frac{(0,006 \times 3,4) + (19 \times 0,2)}{3,6}} = \frac{2000 \times 3,6}{0,0204 + 3,8} = 1884\text{horas} \approx 78\text{días}$$

Para $T_{awake} = 20\text{ms}$:

$$T_{sleep} = T_{period} - T_{awake} = 3,6 - 0,02 = 3,58\text{seg}$$

$$T_{work} = \frac{2000}{\frac{(0,01 \times 3,58) + (20 \times 0,02)}{3,6}} = \frac{2000 \times 3,6}{0,0358 + 0,4} = 16521\text{horas} \approx 688\text{días}$$



Figura 2. Módulo ZigBit A2.

Resulta notoria la drástica diferencia en el tiempo de vida estimado de la batería entre 200ms y 20ms de tiempo activo del nodo. Nuestros cálculos muestran que la vida de la batería es aproximadamente proporcional al tiempo activo. La razón de ello es que la corriente consumida en el estado activo domina sobre la del estado de sueño (las separan unos tres órdenes de magnitud). Evidentemente, el ingeniero del sistema hará bien centrando sus esfuerzos en optimizar el tiempo en el que el equipo está activo en lugar de intentar aumentar el tiempo que pasa dormido.

Periféricos

Los cálculos anteriores realizan una estimación de la vida de la batería basada en el consumo del nodo de red por sí solo. No es de extrañar que la elección de los periféricos pueda afectar al consumo general de energía, tanto mientras el dispositivo está dormido como despierto.

Otra consideración importante es el tiempo que tarda en despertarse el periférico en sí. Si un sensor tarda 500ms en calentarse, entonces claramente, el tiempo mínimo activo no será inferior a 500ms, lo que resulta menos que deseable para muchas aplicaciones con restricciones de consumo. Hay estrategias que un ingeniero de sistemas puede usar para reducir aún más el tiempo activo, como es encender los periféricos en paralelo, así el tiempo total para calentar todos los sensores será el máximo tiempo para calentar el sensor más lento, y no la suma de los tiempos de calentamiento de todos los sensores.

Además de programar el calentamiento de los sensores adecuadamente, los desarrolladores pueden elegir entre una serie de modos de consumo de los nodos para reducir aún más la energía consumida mientras estos están activos. Por ejemplo, una buena estrategia es la de no encender el radio hasta que sea absolutamente necesario hacerlo, es decir, después de que los sensores hayan calentado, se haya obtenidos las lecturas y los datos estén listos para ser enviados. Haciéndolo aseguraría que el transceptor se mantendría encendido durante el menor tiempo posible.

$$T_{Work} = \frac{W}{I_{average}}$$

$$I_{average} = \frac{(I_{sleep} \times T_{Sleep}) + (I_{awake} \times T_{awake})}{T_{period}}$$

$$T_{period} = T_{sleep} + T_{awake}$$

Cabe señalar que el transceptor tiene un consumo de al menos diez veces el del microcontrolador. Los nodos que incluyen el microcontrolador y transceptor separados pueden sufrir de altos consumos innecesarios si ambos se alimentan en el momento en que el nodo se despierta. Lamentablemente, esta estrategia no es aplicable en situaciones en las que los nodos de la red se construyen en torno a un sistema en chip (SoC), que combina tanto el transceptor como el microcontrolador en un solo chip (die).

Más allá de las cuestiones básicas de consumo de los periféricos, estos deben ser escogidos cuidadosamente porque finalmente tienen que conectarse con el nodo de red. Un típico nodo de red expondrá una variedad de interfaces a los que se pueden conectar una variedad de sensores y actuadores. Entre estos se encuentran los canales ADC, I2C, SPI, UART, 1-wire y muchos otros. Los diseñadores deben tener cuidado de recordar que en ocasiones varios interfaces son mutuamente excluyentes, ya que utilizar los mismos recursos hardware. Por ejemplo, las interfaces UART y SPI suelen compartir los recursos hardware, de modo que sólo puede utilizarse una de las interfaces a la vez.

Los sensores y actuadores deben ser cuidadosamente adaptados a los tipos de interfaces, su velocidad, precisión y disponibilidad en combinación con otros interfaces necesarios. Debe tenerse en cuenta que los buses de alta velocidad pueden afectar al rendimiento de pila si se utiliza el mismo microcontrolador para el bus de los periféricos y para ejecutar la pila de red. Puesto que los protocolos inalámbricos suelen requerir un estricto cumplimiento de los tiempos de respuesta a los comandos de radio, la utilización de un periférico de alta velocidad puede tener un efecto negativo en la habilidad de la pila de procesar el tráfico de la red, lo que a su vez resultará en la pérdida de paquetes, y un rendimiento de la red poco fiable.

En los casos en los que la única opción sea un interfaz de alta velocidad con el sensor/actuador, se recomienda incorporar un microcontrolador dedicado específicamente a la manipulación de los periféricos. Observar, sin embargo, que un controlador adicional casi seguramente resultará en un aumento del consumo.

Cuando se conecta un nodo con un periférico externo se necesitará un controlador software. Afortunadamente, la mayoría de los proveedores proporcionan controladores de referencia para su plataforma, los cuales pueden incorporarse en el diseño final con poca o ninguna modificación. Los ingenieros de sistemas están bien servidos por una amplia disponibilidad de controladores de referencia que evitan el desarrollo de estos a partir de cero, ya que puede ser una tarea difícil, sobre todo en una plataforma con recursos limitados como la utilizada en la mayoría de los nodos típicos ZigBee.

Seguridad

Una consideración crítica en el diseño global del sistema es la seguridad de aplicación. La especificación ZigBee ofrece una variedad de modos de seguridad entre los que elegir, cada uno de ellos ofreciendo diferentes niveles de protección.

La encriptación puede implementarse por software (por ejemplo, mediante la aplicación de un sistema de cifrado de bloque AES128 utilizando un método de tablas) o soportadas por el hardware del nodo. Naturalmente, el cifrado hardware incurrirá en menor carga. La encriptación software puede ser un sustituto viable en los casos donde la encriptación hardware no está disponible. Con independencia de la aplicación, un cierto nivel de seguridad siempre es recomendable. En nuestra experiencia, las redes ZigBee no han sido objeto de muchos ataques maliciosos, pero a medida que ZigBee va ganando popularidad es sólo una cuestión de tiempo que empiecen.

El modo más simple de seguridad es usar una clave pre-configurada para cifrar los datos a nivel de aplicación de modo que no se transfiere la carga útil abiertamente. Evidentemente, este esquema de seguridad ofrece la protección más débil ya que la divulgación de la clave sería catastrófica para la seguridad de toda la red. Si la clave se filtra o es adivinada una vez, todos los mensajes puede ser descifrados y leídos por un atacante malicioso.

Una de las ventajas de utilizar una clave pre-configurada es la facilidad de la implementación y la mínima carga que esta implementación impone. El uso de claves pre-configuradas puede

seguir siendo práctica, incluso cuando se utiliza el cifrado software, porque el cifrado/descifrado de mensajes es el menos frecuente de todos los esquemas de seguridad. Tenga en cuenta que la encriptación de un paquete de tamaño máximo en un nodo funcionando con una frecuencia de 4MHz puede llevar del orden de 20-40ms. Huelga decir que dicho funcionamiento es incompatible con la operación de aplicaciones que requieren un alto rendimiento de datos y/o de una baja latencia. Por el contrario, se tarda alrededor de 2ms en cifrar un paquete de tamaño máximo cuando se dispone de soporte hardware AES128.

En la especificación ZigBee también se describe el centro de confianza, se trata de un dispositivo conocido que monitoriza la incorporación de nuevos dispositivos y administra la asignación periódica de claves para permitir las comunicaciones entre ellos. La implementación del centro de confianza acarrea una sobrecarga software significativa y es mejor que se encuentre en un nodo dedicado, donde la operación del centro de confianza no interferirá con la operación de la propia aplicación. Además, los ingenieros del sistema deben asegurarse de que un nodo con las funciones de centro de confianza esté siempre accesible, lo que requiere la cuidadosa colocación física del mismo.

Como componente crítico de la red, un centro de confianza, al igual que el coordinador de la red, debe estar especialmente asegurado contra manipulaciones físicas e interferencias RF. El nivel de seguridad con centro de confianza ofrece el mejor nivel de protección en relación con la sobrecarga software y la complejidad de la aplicación introducidas.

El modo de alta seguridad ofrece el más alto nivel de protección. Este modo de seguridad requiere el cifrado completo de todos los paquetes de red (incluidos los comandos del sistema, las confirmaciones, y las respuestas). Ese sería el modo más apropiado para aplicaciones militares o aplicaciones que requieran el más alto nivel de confidencialidad. En nuestra experiencia, el modo ZigBee de alta seguridad rara vez se utiliza en la práctica.

Cualquiera que sea el modo elegido, la API de programación de la seguridad es bastante sencilla. Como regla general, la clave pre-configurada

se suministra en tiempo de compilación, y si se utiliza un centro de confianza, su funcionamiento es en gran medida transparente a la aplicación. En la transmisión de paquetes, la habilitación de la seguridad normalmente se habilita fijando los flags adecuados en la cabecera del paquete, lo que también se gestiona adecuadamente por la pila con poco aporte adicional de la aplicación.

Puesta en marcha

La puesta en marcha de una red inalámbrica abarca la instalación de la red y toda la configuración post-instalación que se requiere para ajustar los parámetros de red para la operación en ese lugar concreto. Así pues, la puesta en marcha generalmente implica, además de la solución de problemas, por lo menos una forma de particionamiento de red y pruebas de rendimiento in-situ. Se recomienda que antes de hacer la puesta en marcha de una red se lleve a cabo una instalación de prueba para prevenir todos y cada uno de los problemas en la instalación de definitiva. Dicho sitio de ensayo puede ayudar a poner de manifiesto cualquier cuestión imprevista, y puede ayudar a decidir sobre una topología de red y configuración apropiadas.

La especificación ZigBee ofrece ayudas para un procedimiento efectivo de puesta en marcha. Las facilidades forman parte del cluster ZigBee de commissioning - un conjunto de comandos para la puesta en marcha de dispositivos ZigBee por aire. El procedimiento de puesta en marcha consistirá en una herramienta de puesta en marcha conectada a una red ya instalada, y la operación usando la herramienta de puesta en marcha para asignar una variedad de parámetros (por ejemplo, PANID, máscara de canal) a los dispositivos que encuentra en la red que se está poniendo en marcha.

Un paso crítico en la puesta en marcha es garantizar que todos los dispositivos que se instalaron están realmente presentes en la red. Porque la instalación se suele hacer como un paso aparte, cuando se lleva a cabo la puesta en marcha tiene que haber una manera de garantizar que se contabilizan todos los dispositivos instalados. Una forma de hacerlo es interactuar físicamente con todos los dispositi-

vos para comprobar su presencia. Por ejemplo, un operador de la puesta en marcha presionaría un botón en cada dispositivo y comprobaría que el dispositivo se iluminaría en la vista de la topología de red de la herramienta. Del mismo modo, un dispositivo marcado en la herramienta de puesta en marcha debe iluminarse físicamente. Esta forma de comunicación en ambos sentidos puede ejecutarse eventualmente en todos los dispositivos instalados e identificar aquellos nodos que se han instalado pero no han aparecido en la red.

El uso de la seguridad durante el procedimiento de puesta en marcha es de suma importancia ya que cualquier manipulación maliciosa que tenga lugar durante la etapa de la puesta en marcha es una amenaza mucho mayor para la seguridad. Si los nodos reciben parámetros incorrectos pueden llegar a quedarse inaccesibles.

Conclusiones

El diseño de redes de sensores inalámbricos es un reto multifacético que implica las aportaciones de muchos profesionales de diversas tecnologías y gran riqueza de opciones de diseño. Estos artículos intentan dar algo de luz sobre algunas de las cuestiones en juego, ofreciendo enfoques prácticos, consejos y trucos, que pueden ahorrar muchos días del ciclo global de diseño. Un enfoque estructurado hacia el diseño, donde las preocupaciones más relevantes son identificadas a priori y son tratadas una a una, pueden beneficiar en gran medida al ingeniero de sistemas y dirigirle hacia una solución óptima.

Lo que hemos presentado en estos artículos es una síntesis de muchos años de experiencia ayudando a los clientes a diseñar, implementar y desplegar redes de sensores inalámbricos basados en tecnología ZigBee y en nuestros propios productos inalámbricos. En el proceso, hemos reunido las opiniones de los usuarios e identificado los aspectos más problemáticos del diseño de sistemas para abordarlos en estos artículos. Nuestras recomendaciones son prácticas y, como tales, implican la ocasional regla general. No obstante, estas recomendaciones han sido comprobadas que han funcionado para muchos clientes en la vida real. 