

Chapter 2

Principle of Wireless Sensor Networks

Keywords IEEE 802.15.4 • ZigBee • 6LowPAN • Wireless sensor networks

2.1 Introduction

Wireless sensor networks are a subset of wireless networking applications, which focus on enabling connectivity without, the need, generally, of wires to connect to the sensors and actuators (Gutierrez et al. 2004). Due to the length of the name “wireless sensor and actuator networks” or “wireless sensor and control networks”, most people have adopted the shorter “wireless sensor networks” instead. In any case, it is important to remember that the design of this type of network is meant to collect information from wireless sensors and send control commands to actuators attached to the wireless network.

Sensor and actuator networks have existed for decades. Computer based control systems are a typical hardwired sensor and actuator network. As shown in Fig. 2.1, sensors and actuators are connected with a central computer or control terminal via a data bus system or other networks and implement control and monitoring functions. This type of hardwired sensor network is simple and reliable, and often seen in industrial control such as process control and manufacturing production control. Because of the involvement of large amount of cabling in the installation, wired sensor networks are hard to extend. The installation cost of hardwired sensor networks is high, which takes in the form of cabling, labor, material, testing, and verification. Furthermore, cables require connectors that can become loose, lost, misconnected, or even break. This problem is commonly known as the last meter connectivity problem and is called this due to the analogous problem in a wide area network.

The use of large number of hardwired sensors networked to a system brings considerable complexity to the system, including cabling deployment, power supply, and configuration, making it impossible in many cases such as forest monitoring and battlefield surveillance. Recent Integrated Circuit (IC) and Micro

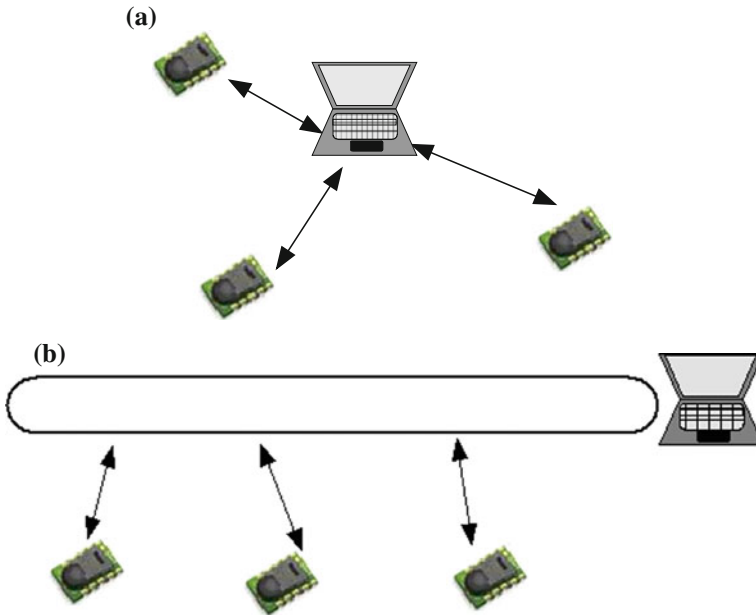


Fig. 2.1 Hardwired sensor and actuator network: **a** Star hardwired sensor and actuator network. **b** Data bus hardwired sensor and actuator network

Electro Mechanical System (MEMS) have matured to the point where they enable the integration of wireless communications, sensors and signal processing together in a single low-cost package, named as a sensor node (Schurgers and Srivastava 2001). Such a sensor node is equipped with data processing and communication capabilities. A set of such sensor nodes forms a wireless sensor network. It is now feasible to deploy ultra-small sensor nodes in many kinds of areas to collect information. The sensing circuitry measures ambient condition related to the environment around the sensor and transforms them into measurable signals. After necessary processing, the signals are sent to a pre-defined destination via a radio transmitter. All of these operations are powered by batteries to ease deployment, since a traditional power supply (i.e. mains power) may not be available.

This type of wireless solutions for sensor networks combines flexible connectivity with ease of installation. The scope of sensors determines the range of applications of wireless sensor networks. There are many types of wireless sensors depending upon the type of sensing required (Lewis 2004; Akyildiz et al. 2002):

- Temperature;
- Humidity;
- Acoustic waves;
- Vehicular movement;
- Lighting condition;
- Pressure;

- Soil makeup;
- Noise levels;
- The presence or absence of certain kinds of objects;
- Mechanical stress levels on attached objects;
- The current characteristics such as speed, direction, and size of an object.

Moreover, there are many applications for the wireless sensor networks, including the following:

- Continues sensing for environmental and condition monitoring;
- Event detection for disaster response;
- Location sensing for mobile target tracking and localization;
- Local control for home automation, industrial automation etc.

Because the reliability and security of hardwired networks can be higher than that of wireless communication systems, wireless sensor networks are not recommended to replace hardwired sensor networks. It is expected that hybrid networks, wired and wireless, will coexist. Wireless sensors will act as extension to wired networks whenever the wireless capability adds value to the applications (Gutierrez et al. 2004).

If we consider only wireless sensor networks with low cost, low energy consumption, low data rate, and short communication range, IEEE 802.15.4 will be the most commonly used communication standard in the design of such wireless sensor networks. ZigBee and 6LowPAN are two most widely adopted IEEE 802.15.4 based communication protocols. This chapter will introduce IEEE 802.15.4 as the foundation of wireless sensor networks and then describe ZigBee and 6LowPAN as two typical wireless sensor networks. A comparison of ZigBee and 6LowPAN will be given at the end of the chapter.

2.2 IEEE 802.15.4 Standard and Wireless Sensor Network

2.2.1 OSI and WSN Stacks

The Open Systems Interconnection (OSI) seven-layer model, proposed by the International Organisation for Standardisation (ISO), forms the basis for the design of the WSN protocol stack. However, unlike the seven-layer OSI model, that consists of the physical layer, the data link layer, the network layer, the transport layer, the session layer, the presentation layer and, the application layer, the WSN protocol stack does not adopt all the seven layers of the OSI model. In reality, the seven-layer OSI model has too many layers making it overly complex and difficult to implement (Aschenbrenner 1986). Consequently, the protocol stack adopted by WSN consists of only five layers, as shown in Fig. 2.2.

Fig. 2.2 WSN protocol stack

Application Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer

The five-layer WSN protocol stack consists of the physical layer, the data link layer, the network layer, the transport layer and the application layer. Each layer is designated a specific set of task to perform independently of the other layers in the protocol stack.

The first layer of the protocol stack, the physical layer, is responsible for defining and managing the connections between individual devices and their communication medium. The physical layer is responsible for frequency selection, carrier frequency generation, signal detection, and modulation and data encryption. Moreover, the physical layer defines the type of connectors and cables compatible with the communication medium.

The second layer of the protocol stack, the data link layer, is responsible for providing services that allow multiple nodes to successfully access and share a communications medium. These services include medium access control, reliable delivery, error detection and error correction.

The third layer of the protocol stack, the network layer, is responsible for establishing the communications paths between nodes in a network and successfully routing packets along these paths. The requirements of different routing protocols can vary and the choice will influence the communication paths set up. Some routing protocols will favour communication paths that help the WSN to deliver the best Quality of Service (QoS), other energy saving protocols may choose the path that enables the WSN to achieve the best lifetime while other will use a hybrid of both objectives.

The fourth layer, the transport layer, is responsible for providing a higher-level layer of the protocol stack and consequently providing the users with transparent and reliable communications between end-users. There are varying forms of transport layer protocols; two of the most popular and contrasting are the transmission control protocol (TCP) and the user datagram protocol (UDP). Connection oriented transport layer protocols, such as TCP, provide a reliable communication service, with extensive error handling, transmission control, and flow control. Whereas, connectionless transport layer protocols, such as UDP, provide an unreliable service but with minimum error handling, transmission, and flow control.

The fifth and final layer, adopted by most WSN, is the application layer. The application layer resides close to the users of the system. There are many potential applications implemented at the application layer including, Telnet, Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP), or Simple Mail Transfer Protocol (SMTP). In terms of WSN, the application layer programming primarily

deals with the processing of sensed information, encryption, the formatting and storage of data. Moreover, the application layer scans the underlying layers to detect if sufficient network resources and services are available to meet the user's network requests.

2.2.2 Overview of IEEE 802.15.4 Standard

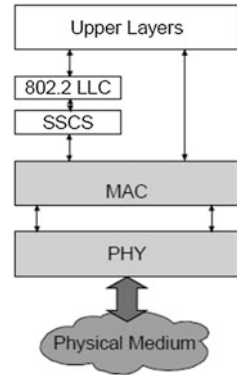
The embedded software design of the wireless sensor networks needs to rely on some standards to ensure that the network system is functional on different hardware platforms. Current standards can be simply divided into two categories public or private, according to the design purpose. Manufacturers of wireless sensor networks will complete the bottom layer development (wireless modulation/demodulation module, MAC layer and network layer protocols, etc.) using the selected standard. Then the developers will build their own applications on top, after purchasing the products from the manufacturers. It is not correct to say that a single standard can suffice for all features required for the wireless sensor networks. Actually, there is no unified standard existing for the concept of WSNs. The existing standards, especially for private standards, usually focus on the specified applications, which might reduce the support available elsewhere. For example, if a standard enables the product to provide a long system lifetime, the support provided for data throughput may be comprised.

The public standards have a much better balanced performance on the above issues than the private standards as their targets are to adopt as much supports from the manufacturers as possible. Any development of a public standard will take into consideration many possible aspects in order to ensure the maximum compatibility. Private standards have a faster development progress than public standards since they only need to improve the content of the standard for their own purpose. However, as indicated by their name, private standards may not be available for public access.

The IEEE 802.15.4 standard (2003) is explicitly designed as a new Low-Rate Wireless Personal Area Network (LR-WPAN) standard for applications that require low data throughput and have limited resource of power and computation capability. It aims to overcome the problems associated with the existing standards such as WiFi and Bluetooth. The standard specifies the physical (PHY) layer and medium access control (MAC) layer for the use of LR-WPANs (IEEE 2003). The first version of IEEE 802.15.4 was published in 2003. Unless we state otherwise, the IEEE 802.15.4 standard described in this chapter is this version.

The IEEE 802.15.4 standard defines the specification of the physical and MAC layers. A comprehensive network layer definition is not directly provided by this standard; instead the standard defines the simplest network topologies—star topology and peer-to-peer topology, which could form the infrastructure for networks based on this standard. Figure 2.3 shows the architecture of the IEEE 802.15.4 standard.

Fig. 2.3 Device architecture defined in IEEE 802.15.4 (IEEE 2003)

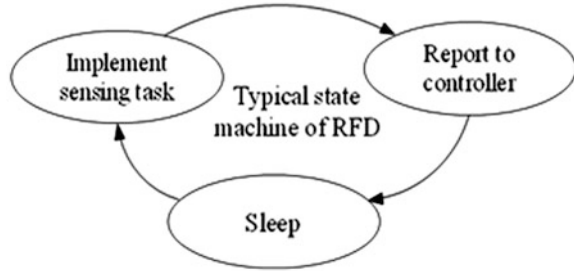


In Fig. 2.3, the architecture consists of a two-layer definition, the PHY and MAC layers. The PHY layer mainly includes the radio transceiver and the corresponding low-level control mechanism. The MAC layer provides the definitions for the data transfer by accessing the PHY layer. The service specific convergence (SSCS) and IEEE 802.2TM Type 1 logical link control (LLC) defines a standard mechanism for the upper layers to access the service of the PHY and MAC layers. Because of the characteristic of limited resource, the wireless sensor network applications normally require the used protocol to be as simple as possible, which can reduce the system overhead. The IEEE 802.15.4 architecture is simple and allows the developers to design the application software at a low-level, which can directly interact with the data transfer. More traditional standards, which comply with the standard Open System Interconnection Reference Model (OSI), might be able to provide reliable and abundant service, but the model's 7-layer definition makes that kind of architecture too complicated to be applicable for WSNs' development.

2.2.3 Full Function Device and Reduced Function Device

According to the IEEE 802.15.4 standard, there are two types of devices participating in IEEE 802.15.4 system, a full-function device (FFD) and a reduced-function device (RFD). An FFD is given the capability to implement a full-function IEEE 802.15.4 stack, which makes it be able to become a personal area network (PAN) coordinator (which can initiate and manage the whole network. This includes the establishment of the network, and the acceptance of association requests from other devices, etc.). Alternatively, it can become a coordinator (which has the same functionality as the PAN coordinator, except for initiating a network), or a normal device. An RFD is a device, which can implement the basic functions of the stack, i.e. a minimal implementation of the IEEE 802.15.4 protocol. An RFD cannot be used to initiate and manage a network, but can be used to

Fig. 2.4 Typical state machine of a RFD



execute extremely simple tasks. The common usage of the RFD is to connect to sensors and regularly send the sensor readings to the network. It is defined in the IEEE 802.15.4 standard that a FFD can talk to other FFDs and RFDs. Using this feature; the upper-layer can implement routing protocols to construct a multi-hop network. However, an RFD can only talk to a FFD since the lack of network management capability makes the RFDs unsuitable for participating in complicated network activities such as sending out beacon signals synchronizing network devices. Consequently, a RFD can last longer than a FFD under the same environment condition. Some wireless sensor network applications are for long term and independent monitoring, consequently, frequently changing the power supply for the distributed sensor nodes is not realistic. In order to save energy, RFDs are more suitable for implementing the functions of such sensor nodes.

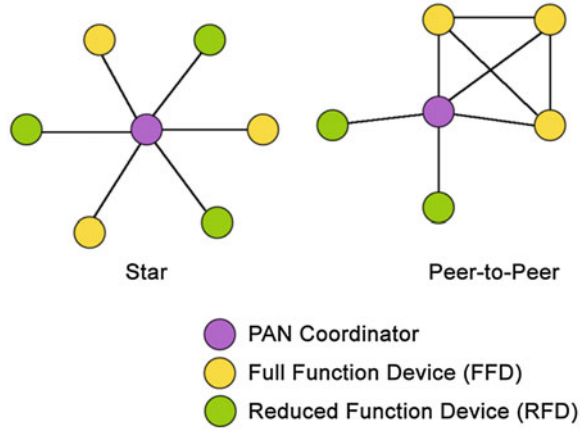
Application code running on FFDs can run more complicated applications than those running on RFDs, e.g. application such as network formation, network maintenance, packet relay, network device management. Application code running on RFDs should be kept as simple as possible. Figure 2.4 illustrates a state machine model of a typical RFD. This RFD regularly implements a sensing task, reports the sensor reading to a controller, and then goes to sleep for a certain period before waking up for the next round of sensing.

2.2.4 IEEE 802.15.4 Topologies

IEEE 802.15.4 supports star, tree, cluster tree, and mesh networks. Figure 2.5 depicts the star and peer-to-peer topologies of IEEE 802.15.4. The star topology is used to form star and tree networks, and the peer-to-peer topology to form cluster tree and mesh networks.

In the star topology, a FFD serving as a coordinator is specified to be the central device, which is called the PAN coordinator, and starts and manage the whole network. Other coordinators and network devices must join the network by associating themselves with the PAN coordinator. The PAN coordinator controls all network communications. The peer-to-peer topology also requires a PAN

Fig. 2.5 IEEE 802.15.4 topologies



coordinator to initialize the network start-up procedure. However, the communications within a network are based on the peer-to-peer topology and are not limited by the PAN coordinator. Any FFD device can freely talk to any other FFD device so long as they are within effective communication range. Any RFD device can talk only to its parent FFD device and cannot directly talk to any other RFD device. RFD devices and their parent FFD device form a tree topology.

Cluster tree topology can be a single cluster network or a multi-cluster network. A single cluster network contains only one cluster-head (CH). All the nodes are connected to the cluster-head with one hop, and the network topology becomes a star topology. A multi-cluster network contains more than one cluster-heads. Each node in a cluster can only talk to its cluster-head. All the cluster-heads form an upper level sub-network, which can directly talk to their head, which might be a sink node, connected to an external network or the head of the cluster heads. Nodes in different clusters do not directly talk to each other but communicate among themselves via their cluster heads. Figure 2.6 illustrates the cluster tree topology, which has hierarchy architecture with the clusters at the bottom level and the cluster-head network at the upper level.

A more complex cluster tree topology is shown in Fig. 2.7 where each cluster illustrated by a dotted cycle connects with another cluster via a border node. The border nodes can be a cluster-head or an ordinary node. A designated device (DD) is required to connect with the network via a border node. The DD device with its border node forms cluster 0 with cluster-head CH0. There are four other clusters with cluster-heads CH1 to CH4 in Fig. 2.7. Cluster-head CH1 is serving as a border node for clusters 0 1, cluster-head CH3 as a cluster-head for clusters 1 and 3. Both CH1 and CH3 have two logical addresses, one as a cluster-head and another one as a border node. The cluster tree topology shown in Fig. 2.6 is different from the one shown in Fig. 2.7, which is a flat network.

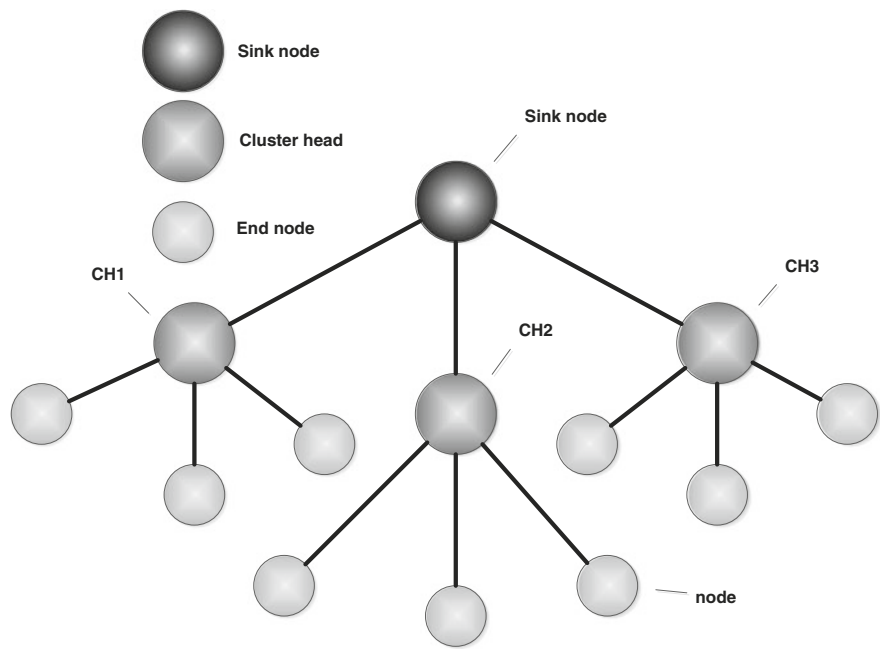
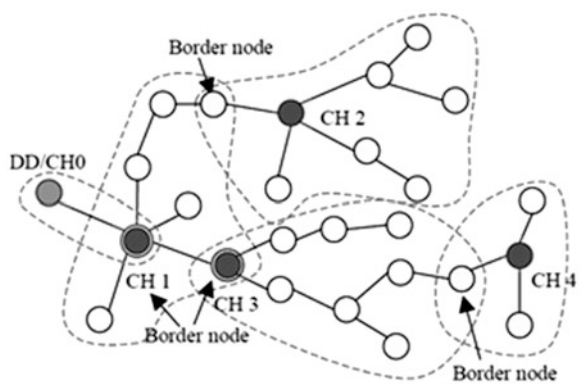


Fig. 2.6 Cluster tree topology

Fig. 2.7 Multi-cluster network connected via border nodes (IEEE 2003)



2.2.5 Multiple Access in IEEE 802.15.4 Wireless Systems

As in all kinds of networks, the wireless nodes in wireless systems have to share a common medium for signal transmission. Multiple Access Control (MAC) protocols in the IEEE 802.15.4 standard defines the manner in which the wireless medium is shared by the participating nodes. This is done in a way that maximizes overall system performance. MAC protocols for wireless networks can be roughly

divided into three categories: fixed assignment (TDMA and FDMA), random access assignment (CSMA/CA), and demand assignment protocols (e.g. polling). In this section, only the most basic concepts of multiple access for wireless networks are presented.

2.2.5.1 Frequency-Hopping/Direct-Sequence Spread Spectrum

Frequency-hopping spread spectrum (FHSS) divides the scientific band in the ISM band into 79 channels of 1 MHz each. The transmitter divides the information and sends each part to a different channel. The process is known as frequency hopping. The order of the channels or hop sequence used by the transmitters is predefined and has already been communicated to the receiver. Bluetooth uses FHSS for its transmission.

Direct-sequence spread spectrum (DSSS) divides each bit into a pattern of bits called a chip. The chip is generated by performing an XOR (exclusive-OR) operation on each bit with a pseudo random code. The output of the XOR operation, i.e. the chip, is then transmitted. The receiver uses the same pseudo random code to decode the original data.

2.2.5.2 FDMA, TDMA, and CDMA

Frequency division multiple access (FDMA) divides the available spectrum into subbands (i.e. channels) each of which is used by one or more users. Using FDMA, each user is allocated a dedicated channel, different in frequency from the channels allocated to other users. The user exchanges information using the dedicated channel. The largest problem with FDMA is the fact that the channels cannot be very close to one another. A separation in frequency is required, in order to avoid inter-channel interference, as transmitters that transmit on a channel's main frequency band also output some energy on sidebands of the channel.

Time division multiple access (TDMA) allows users to share the available bandwidth in the time domain, rather than in the frequency domain. TDMA divides a band into several time slots and each active node is assigned one or more time slots for the transmission of its data.

Code division multiple access (CDMA) follows a different approach. Instead of sharing the available bandwidth either in frequency or time, it places all nodes in the same bandwidth at the same time. The transmission of various users are separated by a unique code that has been assigned to each user. CDMA is often referred to as direct-sequence spread spectrum (DSSS). CDMA can be understood by considering the example of various conversations using different languages taking place in the same room. In such a case, people that understand a certain language listen to that conversation and reject everything else in the other language (Nicopolitidis et al. 2003, p. 59.)

2.2.5.3 CSMA/CA

Carrier-sense multiple access with collision avoidance (CSMA/CA) protocols are the basis of the IEEE 802.11MAC layer. A CSAM node that has a packet to transmit listens to see if another transmission is in progress. If this is true, the node waits for the current transmission to complete and then continues to wait for a span of time known as the short interframe space. Then, if there is still no traffic on the medium, the node will start transmission; otherwise, it has to wait again for the medium to become clear.

2.3 Constructing WSNs with IEEE 802.15.4

Figure 2.8 illustrates the general procedure in terms of which a wireless sensor network is established. The procedure starts with a radio channel assessment, then the network initialization, the network establishment announcement, then several further actions, which take place in parallel. This section introduces the procedures of setting up a wireless sensor network with the corresponding concepts defined in the IEEE 802.15.4 standard. The procedure is shown in Fig. 2.8.

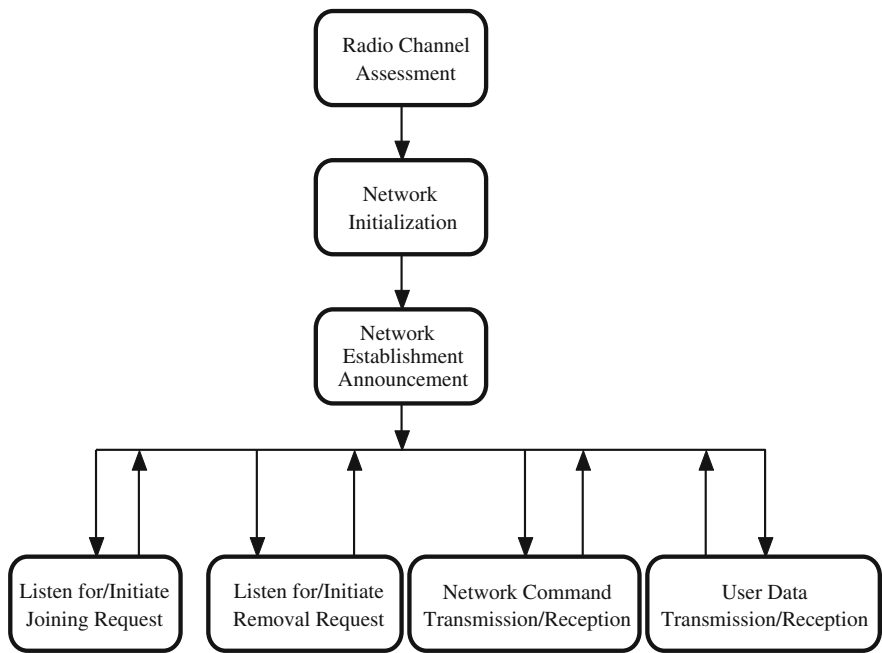


Fig. 2.8 Procedure of establishing a wireless sensor network

2.3.1 Radio Channel Assessment

The first essential task for the construction of a wireless system is always assessing that the desired transmission medium is available. The details of this assessment depend on the characteristics of the wireless network that is to be designed. For networks that utilize frequency hopping, the assessment might focus on the analysis of all available channels and then working out the scheme for hopping. The assessment carried out for networks that utilize frequency division multiple access focuses on searching for the most suitable channel for the network use, such as the cleanest, that which causes the least radio activities, etc. Another important issue in the channel assessment stage is to address how many other systems using the same wireless frequency bands exist in the vicinity. As wireless sensor networks are simple and easy to deploy, multiple networks are highly likely to be operating close by. Trying to avoid conflict with other networks is quite crucial during the assessment stage. [Chapter 7](#) in this book will cover the detail of interference avoidance.

The IEEE 802.15.4 standard specifies three functions related to channel assessment: energy detection, active scan, and passive scan. These terms are explained below:

Energy Detection: Energy detection is clearly defined to give the system the ability for determining the energy level on the specified channels.

Any wireless signal activity in the chosen channel increases its energy level. Consequently, using energy detection can locate any potential interfering sources.

Energy detection is the most effective method to assess the channel, particularly, if the unwanted wireless signals do not have the same characteristics of modulation and spreading as the IEEE 802.15.4 transceiver.

Active Scan and Passive Scan: The functions of active and passive scanning are designed to help the system detect how many similar wireless networks exist in the vicinity. Before a FFD coordinator starts an IEEE 802.15.4 network, it should implement at least one active scan. This function is implemented by sending out a beacon (a kind of synchronization signal used to synchronize the network device, normally generated by a network's PAN coordinator) request within the FFD's personal operating space (POS). Then the FFD coordinator will record the received responses, or named beacon frame, containing the network description from any other existing coordinators, as shown in [Fig. 2.9](#). By comparing the output with the received descriptions, the current FFD coordinator is able to determine if it is possible to start the desired network in this area or on the specified channel.

A passive scan implementation is to enable the current FFD's receiver to listen for network beacons on the selected channel over a certain period, as shown in [Fig. 2.9](#). If other coordinators transmit beacons containing their networks' description, the beacons will be recorded and processed using the same method as the active scan.

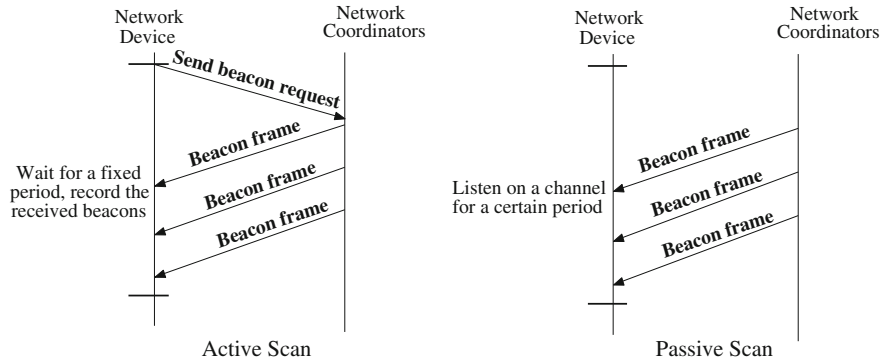


Fig. 2.9 Active/passive scan

The functions of energy detection and active scan are available only for FFD devices, while the passive scan can be applied to both FFD and RFD devices. A typical channel assessment procedure is illustrated in Fig. 2.10, in which energy detection, active scan and passive scan are integrated in a 16 channels assessment.

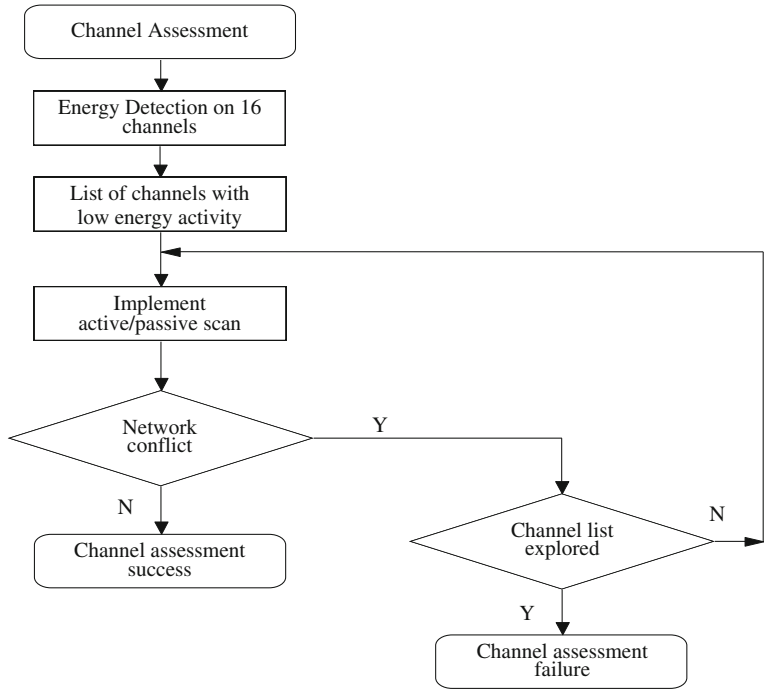


Fig. 2.10 Channel assessment procedure

2.3.2 Network Initialization

Network initialization is implemented by the PAN coordinator. The content of network initialization is to specify various network parameters before actually starting a network. The parameters include the working channel, the network identifier, the network address allocation and setting an IEEE 802.15.4 network beacon.

2.3.2.1 Network Parameter Setting

The working channel is specified according to the results of a channel assessment discussed previously. The IEEE 802.15.4 standard defines the use of the radio frequency and corresponding modulation schemes. The supported data rate is also specified according to the frequency and modulation usage. There are a total of 27 channels across the three frequency bands, which are defined in the standard. Table 2.1 summarizes the allocation of the frequency bands.

Because the IEEE 802.15.4 standard does not support dynamic data rate changing or frequency hopping, a plan for frequency use must be made in advance. Another issue at this stage is the frequency band selection. It needs to comply with the radio regulations, local to where the system is to be deployed.

Once the working channel is decided, the system should select a network identifier by which other devices can identify the network. As a network system, the IEEE 802.15.4 standard supports a 16-bit length network identifier (PAN ID) for labeling each network. The selected PAN ID must be unique and hence cannot be the same as any other network within the radio sphere of influence. Consequently, the active or passive scan can provide useful information for the specified network.

The IEEE 802.15.4 standard defines two basic communication address modes, extended address mode and short address mode. The extended address mode specifies the use of a 64-bit length number, which is fixed in the device’s firmware when it was manufactured. The 64-bit address can ensure the device’s uniqueness. The disadvantage is that the use of the extended address mode will reduce the effective payload size of any data packet. The short address mode specifies the use of a 16-bit length number. The generation of the 16-bit network address is the responsibility of the PAN coordinator when it starts the network. For example, a PAN coordinator can set its own network address as 0x0000. Then any devices joining the network subsequently can be allocated a 16-bit network addresses by

Table 2.1 Allocation of frequency band and data rate

Frequency band (MHz)	Channel	Bit rate (kb/s)	Modulation
868–868.6	0	20	BPSK
902–928	1–10	40	BPSK
2,400–2,483.5	11–26	250	O-QPSK

adding 1 to the PAN coordinator’s address, 0x0001, 0x0002, etc. The length of short address mode decides the theoretical network capacity which cannot exceed 65,535 (i.e. 2^{16}). The use of the short address mode in an IEEE 802.15.4 network can increase the effective payload size in a data packet, but it must be correlated with the PAN ID. Otherwise, the short address’s uniqueness cannot be ensured. The standard has no default short address allocation scheme, and network developers can design an appropriate scheme based on the applications requirement.

2.3.2.2 Superframe Structure

The feature of low power consumption in the IEEE 802.15.4 standard is achieved by a low duty-cycle setting. The component which consumes the most power in a wireless system is the transceiver. A typical working current for an IEEE 802.15.4 transceiver is about 20–30 mA. This is significant energy consumption if the transceiver is kept on for all the time, particularly when the module is powered by battery. The IEEE 802.15.4 standard defines the concept of “Superframe Structure” to allow the system to reduce the transceiver usage, while enabling the network to still function.

The superframe structure is a certain period bounded by the network beacons. Upon receipt of the beacons, the network devices’ transceivers are synchronously functional and start to execute the designed tasks within the range of the superframe. The superframe structure specifies the period within which the transceivers can be active. If an active period is finished, the transceivers should stop working and remain quiet for the following inactive period until the arrival of the next beacon. The mechanism for synchronization means the system has a chance to save energy without losing communication. To ensure the devices synchronize with the same source, the network beacons are sent from the PAN coordinator, which is required to be power on for the whole network lifetime. Figure 2.11 illustrates the superframe structure, where the abbreviations have the meanings shown in Table 2.2.

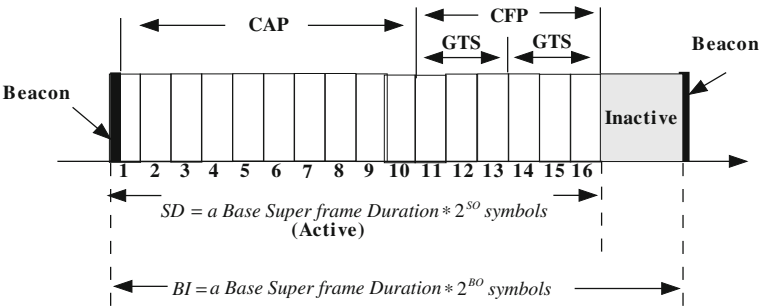


Fig. 2.11 Superframe structure

Table 2.2 Abbreviations in superframe structure

Abbreviation	Meaning
CAP	Contention access period
CFP	Contention free period
GTS	Guaranteed time slot
SD	Superframe duration
SO	Superframe order
BI	Beacon interval
BO	Beacon order

In Fig. 2.11, the superframe structure consists of two main portions: an active period and an inactive period. The length of the active period is denoted as the superframe duration (SD) and calculated by the equation $SD = aBaseSuperframeDuration * 2^{SO}$ where the range of SuperframeOrder (SO) is from 0 to 15, and $aBaseSuperframeDuration$ is calculated as the product of the number of slots (16 in the most case) and the base slot duration (60 in the most case). The whole duration of the superframe structure is called the beacon interval (BI), which includes both the active and the inactive portion and is calculated by the above equation. The range of the beacon order (BO) is from 0 to 15 and the values of SO and BO are related as follows: $0 \leq SO \leq BO \leq 14$. This is because if $BO = SO = 15$, the value of SO should be ignored and the superframe will not exist, consequently the transceivers will be in a state of continuous working with no energy saving.

If $0 \leq SO = BO \leq 14$, the inactive portion will not exist as the length of the beacon interval is equal to the active portion. If $0 \leq SO < BO \leq 14$, the difference between the superframe duration and the beacon interval is the inactive portion, in which all network communications remains silent until the arrival of the next beacon frame.

On receipt of the beacons, network devices can start to implement the designed communications, which must stop before the end of the active portion if $SO < BO$ or before the end of the superframe period if $SO = BO$.

The active portion of the superframe structure is divided equally into 16 slots with the whole duration being $aBaseSuperframeDuration * 2^{SO}$, and further subdivided into two parts: the contention access period (CAP) and the contention free period (CFP). During the CAP, each network device can commence a network communications if required. However, during the CFP, only the devices that have been registered can commence communications. Registrations should be submitted previously to the PAN coordinator by the appropriate network devices. The processed registration information will be contained in the beacon signals. By examining the received beacons, all devices should be able to know if they are registered and hence are allowed to carry out communications in the CFP. Use of the CFP can be allocated to a number of devices, and the communication duration permitted for each registered device is controlled by the guaranteed time slot

(GTS). Further detail can be obtained by reference to the IEEE 802.15.4 standard (2003).

Once the beacon order BO and the superframe order (SO) are chosen, the duty cycle can be calculated. For example, on one of the 16 channels on the 2.4 GHz band, if the beacon order and superframe order are set at 3 and 2 respectively, the beacon interval and superframe duration can be calculated as follows:

$$\begin{aligned}
 BI &= aBaseSuperframeDuration * 2^{BO} symbols \\
 &= numberOfSlots * baseSlotDuration * 2^{BO} symbols \\
 &= 16 \times 60 \times 2^3 \times 16 = 122.88 \text{ ms}
 \end{aligned} \tag{2.1}$$

$$SD = aBaseSuperframeDuration * 2^{SO} * symbols = 960 \times 2^2 \times 16 = 61.44 \text{ ms} \tag{2.2}$$

The PAN coordinator will generate about eight beacons each second ($1,000/122.88 \approx 8$). During each beacon interval, the transceivers of the network devices work for about 61.44 ms and keep quiet for the rest of time. Therefore, the duty cycle is about 50 % ($61.44/122.88 = 0.5$), which briefly means about 50 % of the energy consumption is possibly saved.

Making the transceivers work in the “on or off” mode can save energy, but might cause two problems: firstly, the system may be not able to finish a complete transmission and reception in the time available or secondly, the system response may be delayed. In the first case, it is necessary to calculate the time required for transmitting a single data packet. A full size IEEE 802.15.4 data packet is 133 bytes (IEEE 2003). Using the given data rate, for example 250 kbps at 2.4 GHz, the time required to send an IEEE 802.15.4 data packet is up to 4.256 ms (i.e. $(133 \times 8)/(250 \times 10^3)$). The active portion section in the superframe shown in Fig. 2.11 should be long enough to handle such a transmission within a beacon interval.

Concerning the response delay caused by the mode “on and off” in the data transmission, a proper duty-cycle, i.e. a beacon interval and a superframe interval should be set since a low duty-cycle setting will slow the system response. Table 2.3 summarizes the beacon order and superframe order setting when the duty-cycle is set at 50 % with exception of $BO = SO = 0$.

In Table 2.3, the setting of the superframe order is less than beacon order by 1 (except $BO = SO = 0$). Therefore, the duty-cycle is fixed at 50 %. As defined by the standard, the network will remain quiet during the duration of the inactive period when there is no radio communications allowed. The beacon orders from 1 to 6 in Table 2.3 have the beacon interval set to less than 1 s, which is acceptable for most applications. Beacon orders from 7 to 10 have considerable delay, because the inactive period has duration of about a second (from 983.04 to 7864.32 ms). Increasing the superframe order can reduce the system response delay and decreasing the beacon order, i.e. increasing the duty-cycle. However, a high duty-cycle will consume more power, which is less energy efficient.

Table 2.3 Summary of beacon order and superframe order setting at 50 % duty-cycle

Beacon order (BO)	Superframe order (SO)	Beacon interval (ms)	Superframe interval (ms)	Inactive period (ms)
0	0	15.36	15.36	0
1	0	30.72	15.36	15.36
2	1	61.44	30.72	30.72
3	2	122.88	61.44	61.44
4	3	245.76	122.88	122.88
5	4	491.52	245.76	245.76
6	5	983.04	491.52	491.52
7	6	1,966.08	983.04	983.04
8	7	3,932.16	1,966.08	1,966.08
9	8	7,864.32	3,932.16	3,932.16
10	9	15,728.64	7,864.32	7,864.32

Achieving the balance between the system performance and power consumption is already a challenge to any power supply limited application, and application specific solutions can be achieved. If both BO and SO are set as 15, there will be no power saving issues, as the superframe structure does not exist.

2.3.3 Network Establishment Announcement

Once the network parameters have been initialized, the PAN coordinator can announce the successful establishment of the network. The actual procedure for announcing the establishment of the network is determined by the network protocols used. The purpose of the announcement is to indicate to other devices the existence of the current wireless system. There are two ways to achieve this purpose: announce actively or respond passively upon receiving a requested. Some wireless protocols use the regular beacon signals to synchronize the network operations. This type of network is called a beacon-enabled network. It also informs those newly starting devices with the characteristics of the current wireless systems such as the working channel, frequency band, physical location, etc. If the protocol does not support a regular beacon signal emission, this type of network is called a non-beacon-enabled network, and the PAN coordinator will keep listening on the working channel, and respond to any valid requests which are sent by those devices executing radio channel assessments.

For a beacon-enabled network, after the announcement of the establishment of the network, a beacon signal will be regularly sent out according to the setting of the SO and BO. During the whole working period of the network, the PAN coordinator should ensure the persistent beacon transmission in order to make it detectable by those devices implementing a passive scan, meanwhile any active scan initiated by other devices should also require a response.

2.3.4 Listen for/Initiate Joining Request

After successfully initializing an IEEE 802.15.4 network, the PAN coordinator now becomes the prime network manager. Unless the transceiver of the PAN coordinator is busy on data transmission, it should keep listening on the selected working channel all the time in order to perform the duty of network management.

Any devices wishing to join the network should implement three basic steps: initiate an active scan (FFD only) or a passive scan to locate the desired PAN coordinator, synchronize with the network beacons if applicable ($0 \leq SO \leq BO \leq 14$), request to join the network by issuing an associate request to the located PAN coordinator. Upon receipt of the joining request, the PAN coordinator can implement the designed procedure to validate the request. If the request is granted, the PAN coordinator can decide how to allocate a network address to the device, and it then sends back a response containing the network information (i.e. the network address) and decision to the device. If the joining request is rejected, the PAN coordinator should send back corresponding feedback. Upon the receipt of the response from the PAN coordinator, the network device can use the allocated address to implement network communication, or call a predefined algorithm to deal with the response of “joining failure”.

2.3.5 Listen for/Initiate Removal Request

The way to deal with the removal request is the reverse process to the joining request. The PAN coordinator can delete the device address from the accepted device list and notify the device the removal decision. Alternatively, the PAN coordinator can implement procedures when it receives the disassociate request from the network device. Upon receipt of the notification from the PAN coordinator, the device can ensure that the removal request is permitted.

2.3.6 Network Command Transmission/Reception

The transmission and reception of the network commands are mainly for network management purposes. They are normally invisible to the users without any user interventions. However, sometime a command requiring user interventions will not proceed until the user's instructions are obtained. Therefore, it is necessary to have a processing module for this kind of use in the system design. For example, when a network device notices that there is another IEEE 802.15.4 network in operation in the vicinity and which is using the same network ID, it should send a conflict notification command to the PAN coordinator. Then PAN coordinator should then start an active scan and determine a new PAN ID by broadcasting the coordinator-

realignment command. In this case, the new PAN ID selection is required to be completed by user intervention. Another example is when a network system starts to increase the security level for adopting new devices; the details of any such requesting device need to be reviewed by the upper layer management system. Subsequently network command transmission and reception will occur in this case.

2.3.7 Data Transmission and Reception

The data transmission and reception in the IEEE 802.15.4 standard is categorized according to the use of a beacon. Figure 2.12 illustrates the communication method. There are two communication directions: from a coordinator to a network device, and from a network device to a coordinator.

2.3.7.1 Coordinator → Network Device

In a beacon-enabled network as shown in Fig. 2.12a, when a coordinator has a packet to transmit to a network device, it stores the data in the local buffer, and

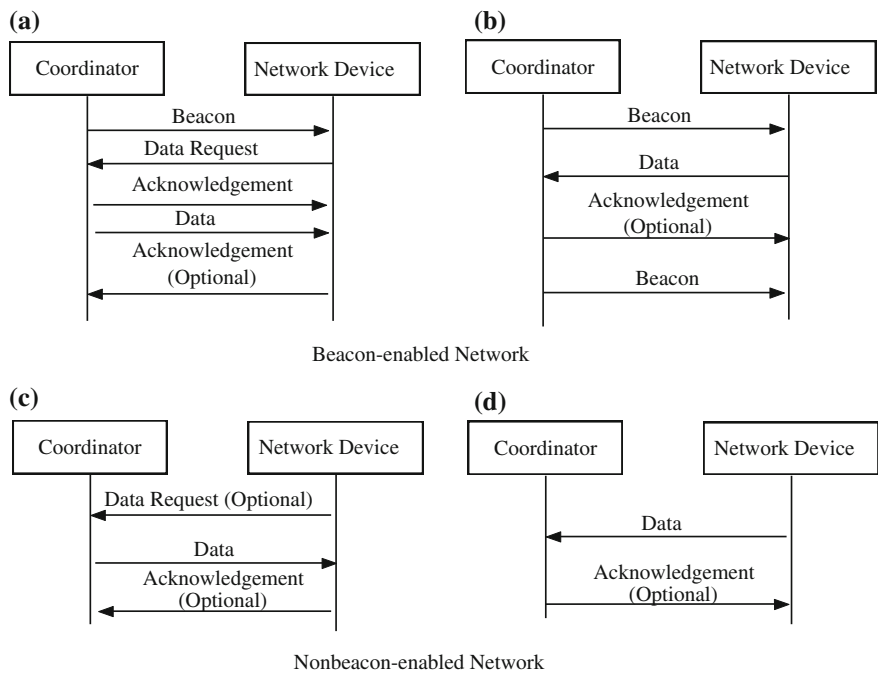


Fig. 2.12 Communication methods defined in IEEE 802.15.4 standard

puts the packet information (i.e. destination address) into the “address pending list” of the beacon frame. On receipt of the beacon frame, the network device will know if there is a packet pending on the coordinator. There are two options for the network device to proceed: if the network device’s *macAutoRequest*, an indicator for the MAC response mode, is set as TRUE, it should automatically send the data request command to the coordinator using the slotted CSMA-CA (carrier sense multiple access with collision avoidance) to request the pending data. If the *macAutoRequest* is set as FALSE, the stack should present the application layer with a primitive of “Beacon Notify”, and let the application decide if it is necessary to send a data request command. On receipt of the data request command, the coordinator will firstly decide the method to send an acknowledgement to the network device. If the coordinator is able to check the local buffer and determine that the pending packet for that network device exists, it then sends the acknowledgement within *macAckWaitDuration*, a predefined duration time. If it is not able to complete the acknowledgement sending within the required time, the coordinator should send the acknowledgement with the data pending field, an indicator of the pending state, set to 1. After sending out the acknowledgement, the coordinator should send the data packet to the network device if the data pending filed is set to 1 in the previous acknowledgement frame. The length of the data payload will be 0 if there is no data pending. On receipt of the acknowledgement, the network device will enable its receiver for the maximum duration of *aMax-FrameResponseTime*, if the data pending field in the acknowledgement is 1. The network device may be required to send back an acknowledgement to indicate the successful reception. The data frame transmission from the coordinator to the network device should use the mechanism of slotted CSMA-CA.

In a nonbeacon-enabled network (Fig. 2.12c), if a coordinator wishes to send a data packet to a network device, it has two options: sends the data packet to the network device directly using the unslotted CSMA-CA, or stores the data into the local buffer and waits for the data request command from the network device, if the network device is programmed to “poll” the coordinator within a certain interval. The process for polling data from the coordinator in the nonbeacon-enabled network is the same as in a beacon-enabled network. However, the CSMA-CA mechanism should use its unslotted version as there is no superframe structure existing.

2.3.7.2 Network Device → Coordinator

In a beacon-enabled network (Fig. 2.12b), the whole network is in an active period. Then on receipt of the regular beacon, if the network device has a packet to transmit to the coordinator, it can commence the communication using the slotted CSMA-CA. It must ensure that the transmission including the acknowledgement can be finished before the end of the active period. Otherwise, the procedure will be temporarily suspended and resume at the start of the next active period. In a nonbeacon-enabled network (Fig. 2.12d), if a network device has a data packet to transmit to the coordinator, it can simply start the communication with the use of unslotted CSMA-CA.

The method of storing the data on the coordinator and sending it out until receiving a request from the network device is called indirect transmission. The indirect transmission is designed for keeping power consumption low. The reason is that the network devices are normally in a sleep state and the radio receiver is off in order to save energy. Storing data on the coordinator can make it convenient for the network devices to obtain the information without keeping the receiver on all the time. The network device can send the data request command when a time slot is available.

2.3.8 Slotted and Unslotted CSMA-CA

As mentioned in the previous section, both beacon-enabled networks and non-beacon-enabled networks use CSMA-CA in their data transmission. Two versions of CSMA-CA are available: slotted CSMA-CA for beacon-enabled access and unslotted CSMA-CA for non-beacon-enabled access. The concept of “slot” is only available in the beacon-enabled network. Figure 2.13 illustrates the “slot” and corresponding “backoff period” in a superframe structure where $BO = SO = 0$.

In Fig. 2.13, the superframe structure is divided into 16 equal sections, each of which is called a “slot”. The basic element used to locate the appropriate time point in each slot is called the “backoff period”, which is represented by the *aUnitBackoffPeriod* symbol, and is the time that a device must wait before accessing the network again. Because the slot number in a superframe is fixed at 16, according to Eq. (2.2) the duration of a single slot is obtained as:

$$\begin{aligned}
 T_{SuperframeSlot} &= \frac{SD}{16} \\
 &= \frac{60 * 16 * 2^{SO} symbols}{16} = 60 * 2^{SO} symbols
 \end{aligned}
 \tag{2.3}$$

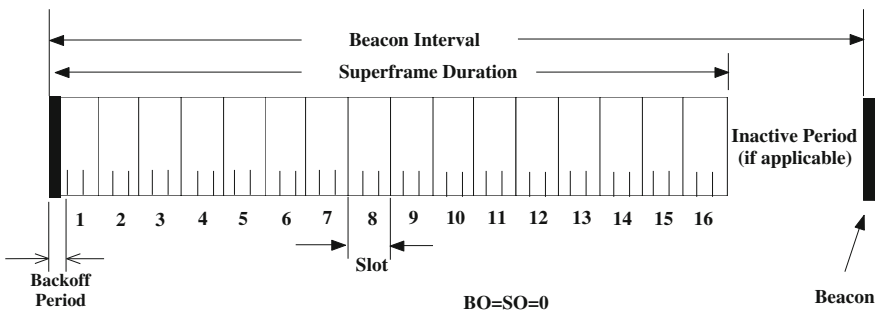


Fig. 2.13 Slot and backoff period in the superframe structure with $BO = SO = 0$

The duration of a single backoff period is defined as: $T_{Backoff_Period} = aUnitBackoffPeriod = 20 \text{ symbols}$

Then the number of backoff period in a single slot $N_{Backoff_Period}$ is defined as:

$$N_{Backoff_Period} = \frac{T_{SuperframeSlot}}{T_{Backoff_Period}} = \frac{60 * 2^{SO} \text{ symbols}}{20 \text{ symbols}} = 3 * 2^{SO} \quad (2.4)$$

In Fig. 2.13, the number of backoff period in each slot is 3, where $SO = 0$.

There are a number of terms used in the operation of CSMA-CA summarized in Table 2.4. Figure 2.14 shows the flowchart of the slotted and unslotted CSMA-CA operation.

In Fig. 2.14, to perform CSMA-CA, the system must first check whether the current network is beacon-enabled. If it is, the slotted CSMA-CA shown in the left hand side is used. Otherwise, the unslotted CSMA-CA shown in the right hand side is used.

In the slotted CSMA-CA, three parameters should be initialized before proceeding, NB , CW and BE . NB is set to an initial value is 0. CW 's initial value is set as 2, and will be reset to 2 each time the channel is assessed to be busy. BE is the backoff exponent, which defines how many backoff periods a device should implement before attempting to assess the channel. If the parameter of $macBattLifeExt$ is set as FALSE, BE should be equal to the value of $macMinBE$. Otherwise, BE should be initialized to the lesser of 2 and the value of $macMinBE$. After completing the parameter initialization, the system should locate the boundary of the next available backoff period (*point 1*). Then it delays for a number of backoff periods, where the number is randomly selected between 0 and

Table 2.4 Terms used in CSMA-CA operations

Term	Meaning
Unit of backoff period	The time that a device must wait before accessing the network again
Backoff exponent (BE)	The time a device must wait before it may attempt to retransmit after previously attempting to transmit a packet over a busy channel
<i>macMaxBE</i>	Maximum backoff exponent, always 5
<i>macMinBE</i>	Minimum backoff exponent, always 3
Number of backoffs (NB)	The number of times a device has tried to access the network via CSMA-CA, the initial value is 0
Contention window (CW)	The length of CW is defined as the number of unit of backoff periods that a channel must be clear before transmission can proceed, the initial value is 2
MacBattery life extension (<i>macBattLifeExt</i>)	When this value is set to true, a device will use the MAC battery life extension period to calculate the number of backoff periods
<i>MacMaxCSMABackoffs</i>	The maximum number of times a device can perform CSMA-CA to access a network, the default value is 4
Backoff period boundary	The start of the beacon in the superframe

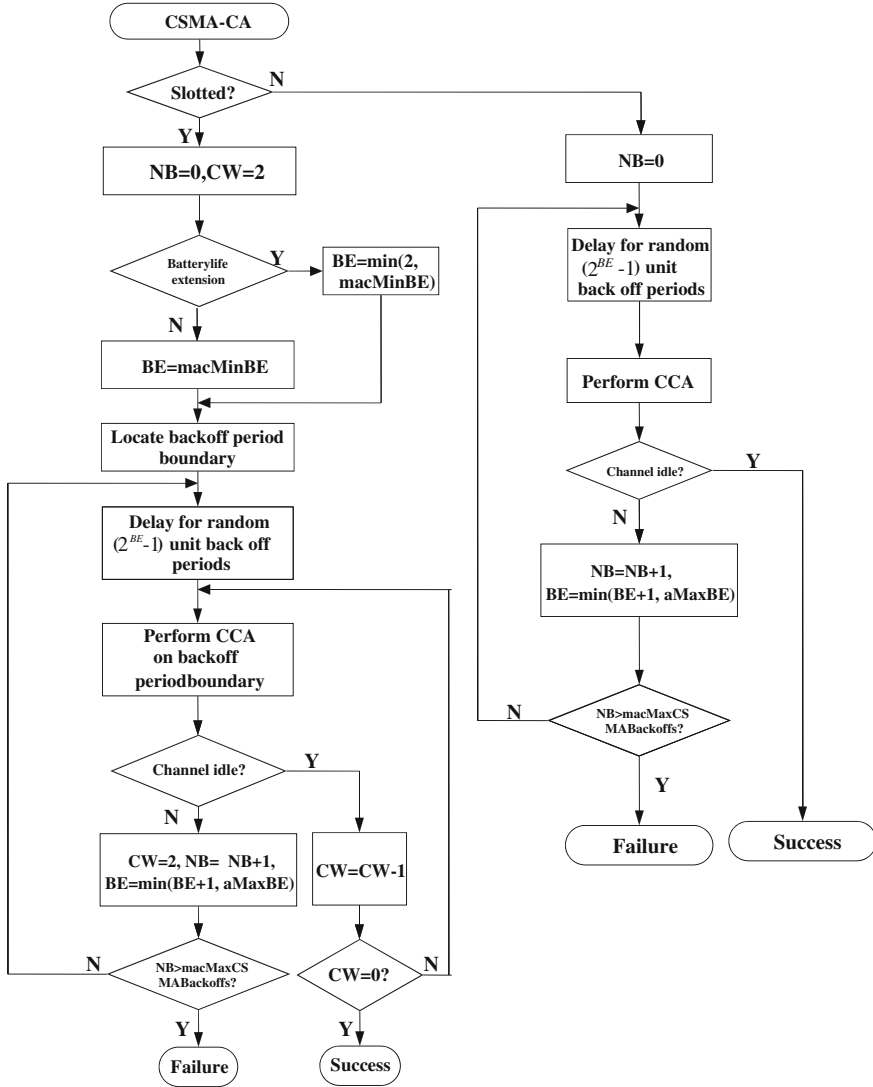


Fig. 2.14 Flow chart of CSMA-CA in IEEE 802.15.4 standard

$2^{BE} - 1$ (point 2). When the delay is finished, the system should perform the CCA on the boundary of the next available backoff period. An important rule defined by the IEEE 802.15.4 standard is that before the first time random delay of the current attempt the MAC layer should evaluate whether the delay can finish before the end of the CAP. If it cannot finish, the system should pause the counter at the end of the CAP, and resume it at the start of the next superframe. If it can finish, the system should apply the backoff delay. When the backoff delay finishes, the

system should evaluate again to determine if the rest of the operations, including two CCA analyses, data frame transmission and the possible acknowledgement reception can be completed before the end of the CAP. If the MAC layer can handle these, the system should start to perform CCA now (*point 3*). If not, the system should stop and wait for the next superframe and repeat the evaluation.

If the channel is assessed to be busy, CW is reset to be 2. The value of NB is increased by 1. BE is reselected from the lesser of $BE + 1$ and $macMaxBE$. If the value of NB is greater than $macMaxCSMABackoffs$, the current attempt is announced to have failed. If not, it should go to *point 2*. If the channel is assessed to be idle, the system should check CW by subtracting 1. If CW is not equal to 0, the system should go to *point 3*. Otherwise, the MAC layer can commence the data transmission on the boundary of the next available backoff period.

In the unslotted system (i.e. nonbeacon-enabled network), two parameters are required to be initialized, NB and BE . NB is set to be 0, and BE is set to be $macMinBE$. After initialization, the system should delay a number of backoff period where the number is randomly selected between 0 and $2^{BE} - 1$ (*point 4*). After the delay, the MAC layer can perform CCA (*point 5*). If the channel is assessed to be busy, increase NB by 1 and reselect BE from the lesser of $BE + 1$ and $macMaxBE$. If NB is greater than $macMaxCSMABackoffs$, the current attempt is announced to have failed. If not, it should go to *point 4*. If the channel is assessed to be idle, the MAC layer can immediately commence the data transmission.

2.3.9 Summary of Data Transmission in IEEE 802.15.4

Because most of the functions used in the data transmission are encapsulated into the stack, the developers may not have been able to access the actual implementation of the mechanisms defined or described in the standard. In spite of that, it is still necessary to know what the system does. Particularly since a lot of stacks will handle the encountered problems and return them to the applications to ask for the users' manual processing. For example, due to the hardware capacity, the coordinator cannot store the pending data permanently in the local buffer. After a certain period, the stack will return an event of "TRANSACTION_EXPIRED" to the application. Or when the coordinator wants to store a new pending data, the stack will also return an event of "TRANSACTION_OVERFLOW" if there is no space available. For the CSMA-CA implementation, the failure of CCA, successful transmission, and missing acknowledgement are all designed to return a corresponding event to the application. Properly handling the event returning from the stack in the function block of "Network Command Transmission/Reception" is essential for the embedded software design.

2.4 ZigBee and Wireless Sensor Networks

2.4.1 ZigBee Stack Structure

The IEEE 802.15.4 standard defines a mechanism to achieve the wireless communication featured with low data rate and low power consumption. It only supports the star and peer-to-peer topologies. There is no definition for a comprehensive network system. Technically, as the IEEE 802.15.4 standard focuses on the development of the PHY and MAC layers, it is mainly suitable for wireless communication, rather than for use in large scale network applications. The ZigBee specification was created in 2004 by the ZigBee Alliance in order to establish large-scale wireless networks on top of the IEEE 802.15.4 standard, which only defines the PHY and MAC layers, for low-rate wireless personal area network (LR-WPAN). The name ZigBee comes from the honeybee, which uses a zigzag type of dance to communicate with other members. ZigBee developers want to emulate this action for solving complex communication tasks simply in LR-WPAN. The ZigBee standard offers a stack profile that defines the network (NWK), security, and application layers. Developers are responsible for creating their own application profiles or integrating with the public profiles provided by the ZigBee Alliance. The publically available ZigBee profiles cover smart energy, building automation, home automation, home and hospital care, telecom applications, consumer electronics control, and industrial process monitoring and control (Elahi and Gschwender 2009). The late version of the ZigBee standard was called ZigBee PRO and published in 2007. Figure 2.15 shows how ZigBee stack sits on top of IEEE 802.15.4.

The characteristics of the ZigBee standard focus on low data rate, low cost, low complexity, low power consumption, ease-to-implement. Table 2.5 shows a comparison of ZigBee characteristics with those of WiFi (IEEE 802.11) and

Fig. 2.15 IEEE 802.15.4 and ZigBee stack

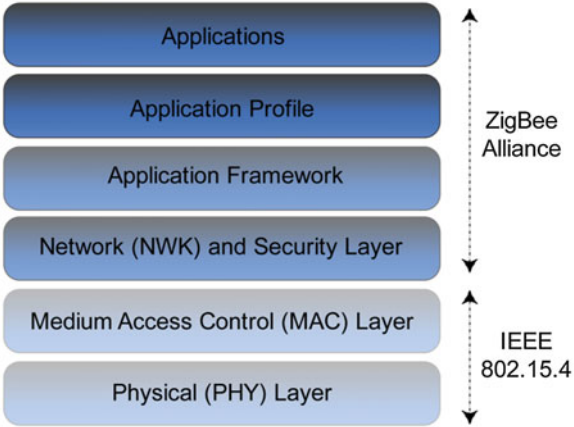


Table 2.5 Comparison of ZigBee, WiFi, and bluetooth (Elahi and Gschwender 2009)

	WiFi (IEEE 802.11)	Bluetooth (IEEE 802.15.1)	ZigBee (IEEE 802.15.4)
Application	Wireless LAN	Cable replacement	Control & monitoring
Frequency bands	2.4 GHz	2.4 GHz	2.4 GHz, 868 and 915 MHz
Battery life (days)	0.1–5	1–7	100–700
Node per network	30	7	65,000
Bandwidth	2–100 Mbps	1 Mbps	20–250 kbps
Range (m)	1–100	1–10	1–75
Topology	Tree	Tree	Star, tree, cluster tree, mesh
Standby current (Amps)	20×10^{-3}	200×10^{-6}	3×10^{-6}
Memory (KB)	100	100	32–60

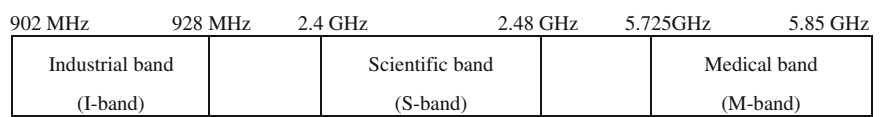


Fig. 2.16 ISM band frequency allocation

Bluetooth (IEEE 802.15.1). Table 2.5 shows that WiFi, Bluetooth, and ZigBee use the ISM (industrial, scientific research, and medical applications) band, which is license free and has a transmission power of less than 1 W. Figure 2.16 illustrates the frequency allocation of the ISM band made by the federal communication committee (FCC). ZigBee can be set at I-band, S-band, or M-band. WiFi and Bluetooth can only be set at S-band.

Figure 2.17 illustrates the architecture of the ZigBee stack, which is divided into three sections, as follows:

- IEEE 802.15.4, which consists of the MAC and PHY layers.
- The ZigBee section, which consists of the network (NWK) layer, the application support sublayer (APS), the security service management, and the ZigBee device object (ZDO). The endpoint is an application object, which can have up to 240 separate application objects. An endpoint defines input and output to the APS. The ZDO performs control and management of application objects.
- The ZigBee application section, in which developers can use the ZigBee application profiles or develop their own application profile.

In the ZigBee specification, the network devices are categorized into three types: ZigBee coordinator, ZigBee router, and ZigBee end device. A ZigBee coordinator is an IEEE 802.15.4 PAN coordinator, which is a fully function device. Any ZigBee network should have one and only one ZigBee coordinator. The ZigBee coordinator should be capable of selecting an available channel and

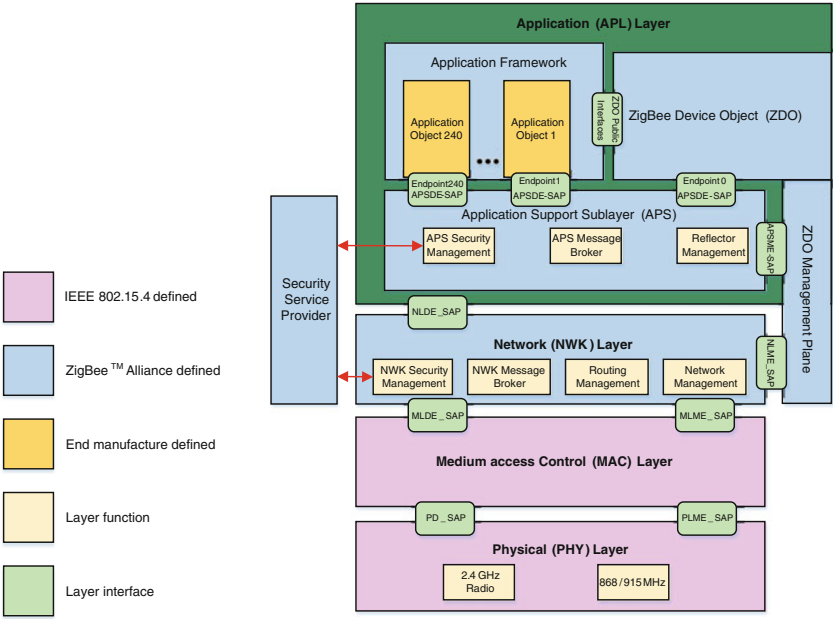


Fig. 2.17 Stack structure of ZigBee (2004)

appropriate network identifier (16-bit length) for the creation of a new network. As the first device that starts the ZigBee network, the ZigBee coordinator takes the responsibility of adopting the new devices and allocating network address.

A ZigBee router is an IEEE 802.15.4 full function device. The ZigBee router should provide the capability to select an existing ZigBee network to join and extend the ZigBee network by adopting new devices, which are out of the communication range of the ZigBee coordinator. Meanwhile, the ZigBee routers construct the backbone of the network by implementing the designed routing protocols.

A ZigBee end device is an IEEE 802.15.4 full function device or reduced function device. As the ZigBee end devices are often deployed at the end of the network to implement the sensing tasks, a reduced function device is more suitable for such use, as they are more energy efficient.

2.4.2 ZigBee Topologies

2.4.2.1 ZigBee Topologies

ZigBee is on top of IEEE 802.15.4 and uses the IEEE 802.15.4 specification for its MAC and PHY layers. ZigBee network supports star, tree and mesh topologies by

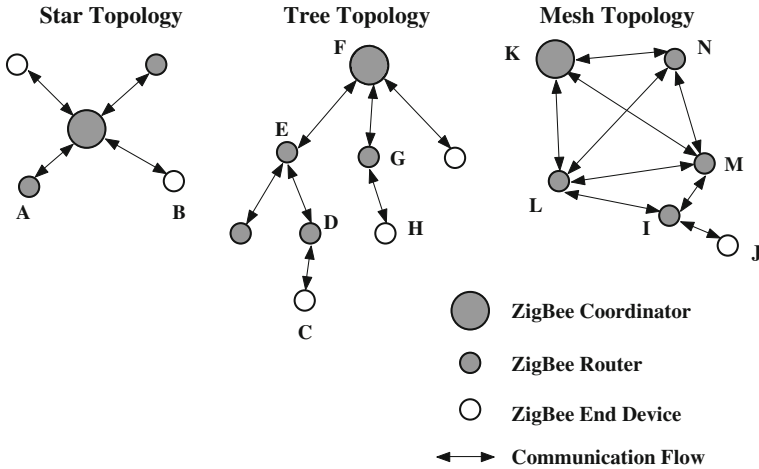


Fig. 2.18 Three topologies in ZigBee specification

extending the use of peer-to-peer topology. Figure 2.18 illustrates the three topologies.

In Fig. 2.18, the star topology is the easiest one to achieve. The ZigBee coordinator is the centre node of the network, other ZigBee devices including the ZigBee routers and ZigBee end devices are required to connect to the ZigBee coordinator to form the network. The star topology is not suitable for large-scale applications due to the limitation of the ZigBee coordinator. Because all the devices must join the network through the ZigBee coordinator, devices, which are out of the radio range of the coordinator, cannot be networked. The main shortcoming of the star topology is that the failure of the centre node (ZigBee coordinator) will affect the whole network. Devices in a star network cannot communicate with each other directly. For example, if device A is to send a message to device B in the start topology shown in Fig. 2.18, the message will be sent to the ZigBee coordinator first, and then relayed to the destination.

The tree topology is more flexible compared with the star topology. Its deployment is not limited by the coordinator, and can be extended by using ZigBee routers to adopt sub-devices. The criteria to form a tree network are: an end device joins the tree via a router device, and a router device joins the tree via another router device (the ZigBee coordinator can be used as a router device as well). The difference is that a router device can adopt end devices or other router devices as its sub-devices, which are also called children. An end device cannot have children. Therefore an end device cannot be a parent device. The network communications in a tree network must comply with these rules. For example, if device C is to send a message to device H, the message should be sent back to device F by passing through devices D and E. Then device F sends the message down to the device H through device G. The criterion is that the message must travel from the

source node up the tree to the nearest common ancestor and then down the tree to the destination node (ZigBee 2004). The disadvantage is that there is no alternative route available if any one of links on the route fails. However, the implementation of the routing protocol is fairly easy as each device just needs to maintain a tree table and simply pass the message to the parent node or to the descendent node which points to the destination.

The mesh topology has the same structure as the tree topology, but its network communications are more flexible. All routers are allowed to communicate with each other without needs to send message to the parent device first. For example, if device J is to send a message to device K, the possible routes can be $J \rightarrow I \rightarrow L \rightarrow K$, $J \rightarrow I \rightarrow M \rightarrow K$, $J \rightarrow I \rightarrow M \rightarrow N \rightarrow K$, $J \rightarrow I \rightarrow L \rightarrow N \rightarrow K$ and $J \rightarrow I \rightarrow L \rightarrow M \rightarrow N \rightarrow K$. The network routing algorithm would pick up an alternative route from the available options when some of them fail.

2.4.2.2 ZigBee Hybrid Network

By appropriately using star, tree, and mesh topologies, or a combination of them, ZigBee network can form various architectures. Among them the mesh topology is the most popular network topology with flexible network configuration and the capability of self-healing in network communication.

- *Flexible Network Configuration*

From the view of logic relationship, the mesh topology is the same as the tree topology. Because the mesh topology employs the same criteria as the tree topology to form the network, it can be thought of as a special version of the tree topology. Meanwhile a mesh topology is an amplified star topology. Since the router nodes can commence communications with each other, it is possible to program the communication flows to make them aggregate at a single point, if required by the application. Therefore, the mesh topology is flexible in its network configuration. However, a star or tree network cannot be extended to a mesh network.

- *Capability of self-healing*

As discussed above, the star network and tree network have the same critical defect: if any link on the route or the centre node fails, the whole network will breakdown. Using the dynamic routing protocol, the mesh network can solve the problem by walking around the failed links or nodes.

- *Hybrid structure*

A ZigBee network has normally a hybrid structure rather than one of the three topologies discussed above. It consists of two layers as shown in Fig. 2.19. Layer 1 consists of the ZigBee coordinator and ZigBee router devices. The router devices construct the backbone of the network within which the routing protocols can be implemented. The ZigBee end devices, which are categorized into layer 2, join

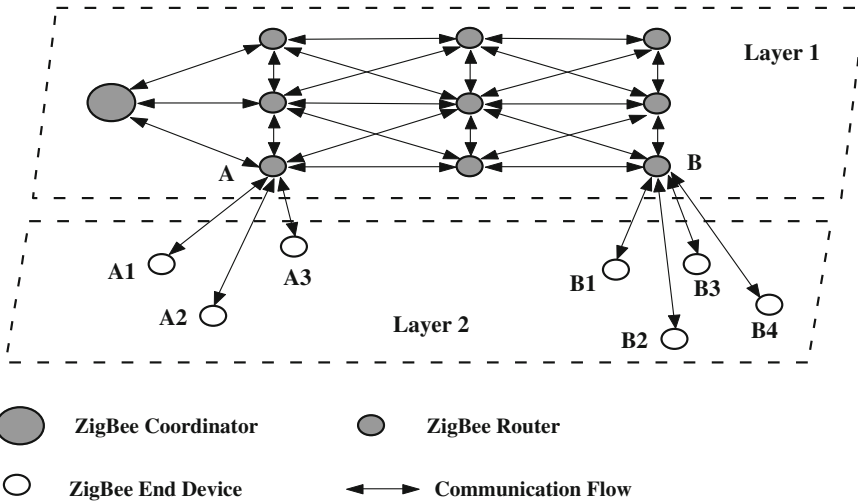


Fig. 2.19 ZigBee hybrid network

the network through their parent ZigBee router devices. According to the ZigBee specification, a ZigBee end device can only talk to its parent device. Therefore, the ZigBee end devices are not involved into the network communication relay. Any network communications between a ZigBee end device and other ZigBee network devices must be sent to the corresponding parent device, and then routed to the destination. The parent ZigBee router device and its connected ZigBee end devices form a star topology.

2.4.3 ZigBee Address Allocation Scheme

The ZigBee specification also defines a practical network address distributed allocation scheme for the network use, named as “Cskip”, which is not specified in the IEEE 802.15.4 standard. Briefly, the network capacity i.e. the number of available 16-bit addresses in a tree topology is decided by four parameters:

- C_m —Total number of children that any parent device may have
- R_m —Total number of router children that any parent device may have
- L_m —Maximum depth of the network (i.e. the level at which parent devices may no longer have children)
- d —Actual depth of the device under consideration

The above four parameters are stored in the network information base of the coordinator. The ZigBee coordinator assigns a block of addresses to each router

based on the maximum number of children C_m . The allowed number of end devices accepted by a router device is calculated as:

$$MaxEndDevices = MaxChildren - MaxRouters = C_m - R_m \quad (2.5)$$

In the sequence, when a router device successfully joins a network, its parent device allocates a block of address for its use, which means the joined router device becomes a potential parent device. Each joined router device can accept a certain number of children devices whose number cannot exceed C_m . The joining of the new router device is considered to extend the depth of the network, and the depth should not be greater than L_m .

If an end device successfully joins a network, it will be allocated a network address by its parent device. The joined end device does not have the capability to accept new children devices.

Cskip is the method for calculating the total number of possible descendants that exist down any branch in the network. It is defined as follows:

$$Cskip(d) = \begin{cases} 1 + C_m \times (L_m - d - 1) & \text{if } R_m = 1 \\ \frac{1 + C_m - R_m - C_m \times R_m^{L_m - d - 1}}{1 - R_m} & \text{if } R_m > 1 \end{cases} \quad (2.6)$$

Figure 2.20 illustrates the address allocation with $C_m = 5$, $R_m = 2$, and $L_m = 2$. According to Eq. (2.6), $Cskip(0)$, $Cskip(1)$, and $Cskip(2)$ are obtained to be 6, 1, and 0 respectively. $Cskip(0)$ is the maximum possible number of descendants that lie down each router-branch from the coordinator. $Cskip(1) = 1$ means that each branch from the router maximally has one descendants. $Cskip(2) = 0$ means that any router at this level has no descendant.

The total number of potential nodes in the network can be calculated as follows:

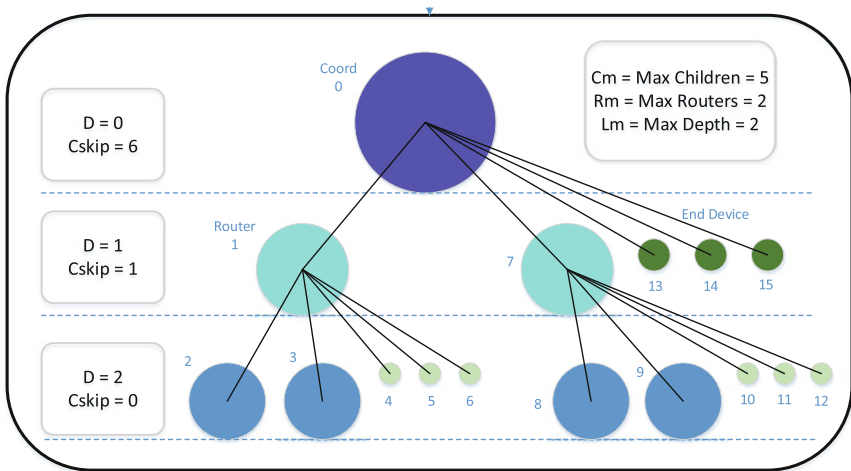


Fig. 2.20 Cskip address allocation scheme (ZigBee 2008)

Table 2.6 Network capacity evaluation under certain parameters (ZigBee 2008)

Increasing value	C_m	R_m	L_m	Total number of nodes
None	20	6	5	31,101
C_m	21	6	5	32,656
R_m	20	7	5	56,021
L_m	20	6	6	186,621

$$Node_{total} = Cskip(0) \times R_m + (C_m - R_m) + 1 \quad (2.7)$$

According to Eq. (2.7), the total number of nodes in Fig. 2.20 is obtained as $6 \times 2 + (5 - 2) + 1 = 16$, in which includes the coordinator. Changing the parameters C_m , R_m , and L_m can change the maximum number of nodes in a network. Table 2.3 shows a summary of network capacity according to the given parameters of C_m , R_m , and L_m . Each ZigBee device is assigned a logical 16-bit address by the Zigbee coordinator or router when joining a ZigBee network. Therefore, the maximum number of network nodes in any ZigBee network is $2^{16} = 65,535$. The last row in Table 2.6 is not realistic.

$Cskip(d)$ is also used as the offset value to assign addresses to routers and its end devices. Assuming the address of the coordinator is fixed. The address of the first router R_1 is equal to the address of the coordinator +1. The following equation is used to assign an address to router R_n :

$$R_n = R_1 + (n - 1) \times Cskip(d) \quad (2.8)$$

Where d is the depth of the network at the upper level, i.e. the coordinator level. In the same way, each router device assigns an address to each of its descendants. For example, router R_1 initially assigns an address to one of its children, which is one greater than its own address. It then uses $Cskip(d + 1)$ as the offset value to assign addresses to the other children connected to it. The rest of the routers use the same procedure.

ZigBee PRO offers stochastic address assignment. This means that each node, when joining the network, is assigned a random number for 0–65,536 as the address. If the new address of the node has been used for any existing device, a conflict notification will be announced and another address should be assigned to the new device.

2.4.4 ZigBee Management Mechanisms

The main feature of the ZigBee standard is its efficient and effective management mechanism designed for the application layer. The definition for the ZigBee management mechanism consists of the address management, profile management, device & service discovery and binding.

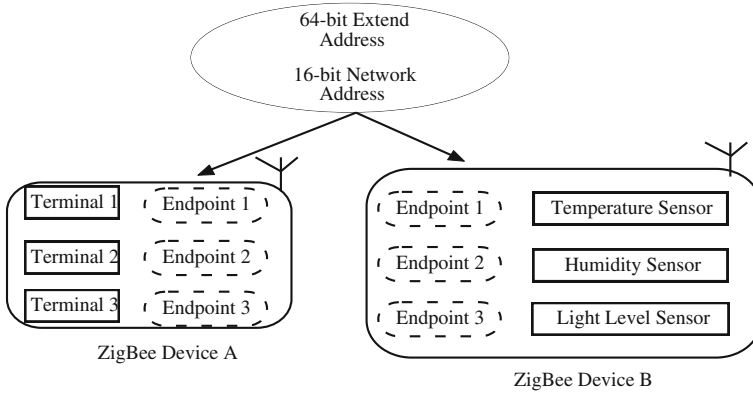


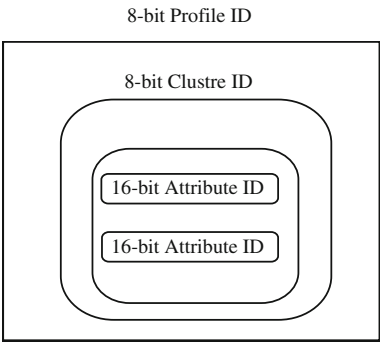
Fig. 2.21 Address management in ZigBee

2.4.4.1 ZigBee Address Management

As the IEEE 802.15.4 standard is used to construct the bottom layer (PHY and MAC) of the ZigBee stack, the 64-bit extended address and the 16-bit network address are both available for a ZigBee network's use. However, they are not sufficient to identify multiple objects sharing the same physical address. In the ZigBee specification, the concept of Endpoint addressing is specified to solve the problem.

Figure 2.21 shows the address usage in a ZigBee network, in which there are two ZigBee devices A and B need to communicate with other. Device A has three terminals, which correspond to the three sensors on devices B respectively. If terminal 1 on device A wishes to establish communication with the temperature sensor on device B, it can request device A to establish a wireless communication channel to the device B, using either its IEEE 802.15.4 64-bit extend address or its 16-bit network address. The problem is how to make device B recognizes that the communication is for the temperature sensor, rather than one of the other two sensors? The ZigBee specification defines a sub-level addressing mode—Endpoint, to help the system distinguish the multiple objects existing on one physical device. “Endpoint” is a kind of categorization, which virtually exists in the stack. Each ZigBee device can support up to 240 virtual objects (endpoint 0 is used for endpoint management). Each virtual object has its own property and can be independent from other objects. If the starter of the communication specifies which endpoint it is looking for, the ZigBee stack running on the destination ZigBee device can easily locate the target object. The concept of Endpoint in the ZigBee specification is useful, particular for wireless sensor networks. A sensor node is normally equipped with more than one sensor for executing multiple sensing tasks.

Fig. 2.22 ZigBee profile management



2.4.4.2 ZigBee Profile Management

Profile management is the communication fundament in the ZigBee specification. It consists of the agreements on messages, message formats and processing actions that have been well defined to enable the cooperation in the system. By following the same profile, the different components are able to create an interoperable, distributed application. And also, the products from different manufacturers can seamlessly communicate without worrying about compatibility. Figure 2.22 shows the concept of a profile.

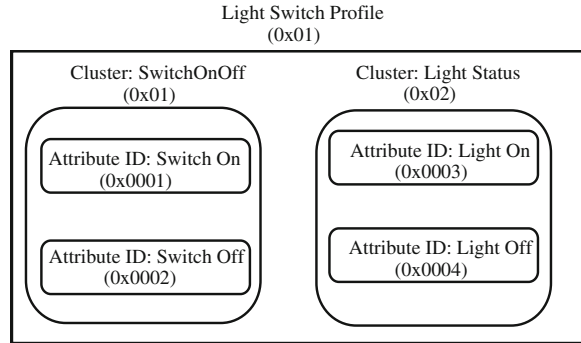
In Fig. 2.22, the Profile ID is an 8-bit number, which specifies the property of the current profile. The ZigBee Alliance has specified a number of profiles (Home Control Stack Profile, Building Automation Stack Profile, Plant Control Stack Profile, etc.) which are called public profiles. The manufacturers can define their own profiles for the specified applications, which are called private profiles. The developers can obtain a profile ID by applying to the ZigBee Alliance. The profile ID must be unique due to administrative issues. If the profile is defined for research purpose, users do not have to apply for permission. Table 2.7 shows the ZigBee public profiles and corresponding IDs. The public profile ID range is from 0x0000 to 0x7fff. The private profile ID ranges from 0xbfff to 0xffff.

Attributes indicate the function or data of a device connected to a node. For example, the attribute of a light switch represents the position (status) of the

Table 2.7 ZigBee public profiles (Elahi and Gschwender 2009)

Profile name	Profile ID
Industrial process control and monitoring (IPM)	0x0101
Home automation (HA)	0x0104
Commercial building management (CBM)	0x0105
Telecom applications (TA)	0x0107
Personal, home, and hospital care (PHHC)	0x0108
Advance metering initiative (AMI)	0x0109

Fig. 2.23 Example of using ZigBee profile



switch, which can be on or off. The attribute identifier is a 16-bit number used to specify the actual data item, i.e. the attribute, passes between the ZigBee devices.

A cluster is a collection of attributes and commands that is used to perform a specific function, which associates with data flowing out of, or into the device. From example, a SwitchOnOff cluster is used to turn on or off a switch device. The cluster identifier is an 8-bit number which lies on at the second level of the profile management. The cluster identifier is unique in the scope of a specific profile. An application profile could have more than one cluster. By associating with the cluster id, the attribute id expresses the actual command to the applications.

Figure 2.23 illustrates an example of using profile management. In Fig. 2.23, a light switch profile with profile id 0x01 is set to manage the lighting system. Two clusters are defined: SwitchOnOff and LightStatus. The cluster SwitchOnOff is for controlling the instruction implementation from the users. By recognizing the incoming command containing the corresponding cluster id and attribute id, the local system is able to correctly execute the instruction of switching on or off a light. The cluster LightStatus is for outputting the light status upon receiving a request from the users. By examining the formatted message containing the cluster id (LightStatus) and the attribute id (Light On/Off), the users will be shown the current status of the light.

2.4.4.3 Device and Service Discovery

To simplify and standardize the service provision, the ZigBee specification provides the mechanism of discovering devices and services in the network. The device discovery command supports both the IEEE 64-bit and 16-bit network address and can be sent either as a broadcast or unicast message. A network should have a primary discovery cache device, which can be a router or a coordinator, for storing node descriptors of the devices that are in sleep mode. Before any device goes into sleep mode, it transmits its descriptor information to the primary discovery cache. It is the primary discovery cache device that responds when the requested device is in sleep mode. The actual implementation is performed by

the ZigBee Device Objects (ZDO). For example, if a ZigBee device newly joining the network wishes to know the network address of the ZigBee coordinator but it only knows the coordinator’s 64-bit extend MAC address, or if the device wishes to know the network address of the devices which can provide the function of controlling a light, the ZDO can help send out the formatted broadcast queries to the network, or a unicast query to a specified device, and obtain the results when the discovery is finished. The ZDO is a kind of protocols defined in the ZigBee stack. Each ZigBee device running a ZigBee stack has it’s own ZDO instance, which can deal with the information processing under the ZDO management without any user intervention. What the developers should consider is how to design the query submission and process the returned results.

2.4.4.4 ZigBee Binding

A useful feature of the ZigBee specification is its support for the concept of binding, which is a logical relation between two endpoints located in different devices. In the sensor network application development, the situation of sending control message from a single point to multiple destinations or from multiple destinations to a single point, or from a single point to a single destination is often encountered. The situation is shown in Fig. 2.24, in which three situations are present:

- A single light switch controls more than one light. The situation often happens when a central switch is designed for use in a warehouse.
- A single light switch controls a single light. This is the normal situation happening in daily life.
- More than one light switches control a single light. This is often used for the design of the light control in the corridor or on the stairs.

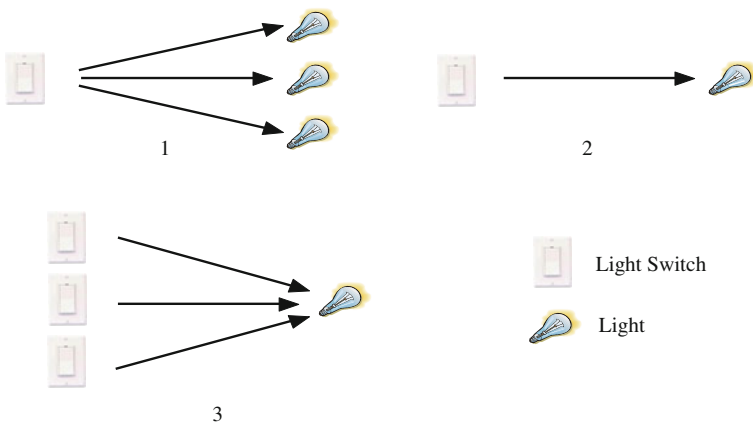


Fig. 2.24 Message sending in wireless sensor network for lighting control

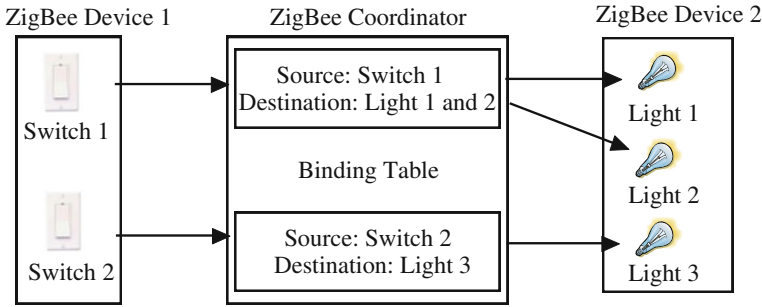


Fig. 2.25 Implementation of binding in ZigBee networks

As the procedure for handling the situations described in Fig. 2.24 will introduce massive duplicated of work using the traditional method, the mechanism of binding can make the operations much more convenient. The implementation of binding is shown in Fig. 2.25.

In Fig. 2.25, the coordinator is selected to store the binding table as it is supposed to be power on during the whole network's lifetime. There are two entries created and recorded in the table, the first entry records the source address and endpoint of Switch 1 and the matched addresses and endpoints of Light 1 and Light 2. The second entry records the address and endpoint of Switch 2 and the matched address and endpoint of light 3. If Switch 1 needs to turn on Lights 1 and 2, it can send the instruction and its own address to the coordinator. Upon receipt of the instruction, the coordinator will search the table and find out the two addresses of Lights 1 and 2. Then the coordinator replaces the destination addresses of the instruction with those of Lights 1 and 2 and automatically sends the instruction out. Therefore, Switch 1 can control Lights 1 and 2 via the binding between them. And the instruction can be processed quickly, which improves the overall execution efficiency.

2.5 6LoWPAN and Wireless Sensor Network

6LoWPAN stands for IPv6 over IEEE 802.15.4 low-power wireless personal area networks (L-WPAN) and was developed by the Internet Engineering task Force (IETF) in 2007 (Kushalnagar et al. 2007). IPv6 is the newest version of the Internet Protocol. 6LoWPAN enables IPv6 directly working over IEEE 802.15.4 low-power wireless sensor networks. Consequently, individual wireless node in a 6LoWPAN based wireless sensor network become accessible from the Internet. A straightforward technical definition of 6LoWPAN given by Shelby and Bormann (2009) is *6LoWPAN standards enable the efficient use of IPv6 over low-power, low-rate wireless networks on simple embedded devices through an adaptation layer and the optimisation of related protocols.*

The benefits of making wireless sensor networks Internet enable include (Kushalnagar et al. 2007):

- It brings interoperability as it allows the use of existing network infrastructure based on IP-based protocols.
- Wireless devices can be connected easily to the Internet without the need for gateways.
- Enabling IP also enables the network to use all of the IP-based technologies such as proxies, which are well known and proven to work for higher level services in a large-scale network.
- Established Application protocols and data models such as HTTP, SNMP and DPWS etc. can be used.
- Transport protocols can be used to provide some reliability in a network with unreliable links.
- IP technology promotes innovation by providing all the standards and related documents available to anyone.
- Many protocols are already available for commissioning and managing the IP-based networks which can be used.

Figure 2.26 shows the protocol architecture for 6LoWPAN, where an adaptation layer, or called a LoWPAN layer, is added between the MAC layer and the IPv6 network layer. The function of adaptation layer is to perform the following tasks:

- Compress the IPv6 header
- Fragment the IPv6 payload
- Compress the UDP header

The detail can be found in the 6LoWPAN specification (Kushalnagar et al. 2007). The UDP (user datagram protocol) and the Internet control message protocol (ICMP) are used in 6LoWPAN. Adaptation between IPv6 and IEEE 802.15.4 is performed by routers at the edge of 6LoWPAN, referred to as edge routers.

Figure 2.27 shows the position of edge routers in the integration of WSN and the Internet. Each LoWPAN consists of an edge router, a number of LoWPAN routers (R) and a number of LoWPAN hosts (H). Additionally, there is a remote server on the Internet. 6LoWPAN enables IPv6 for simple embedded devices over low-power wireless networks by efficiently compressing headers and simplifying IPv6 requirements. When connecting a LoWPAN to the Internet or another IP network, there are several issues to be considered (Shelby and Bormann 2009):

Fig. 2.26 6LoWPAN protocol stack

Application protocol	
UDP	ICMP
IPv6	
LoWPAN	
IEEE 802.15.4 MAC	
IEEE 802.15.4 PHY	

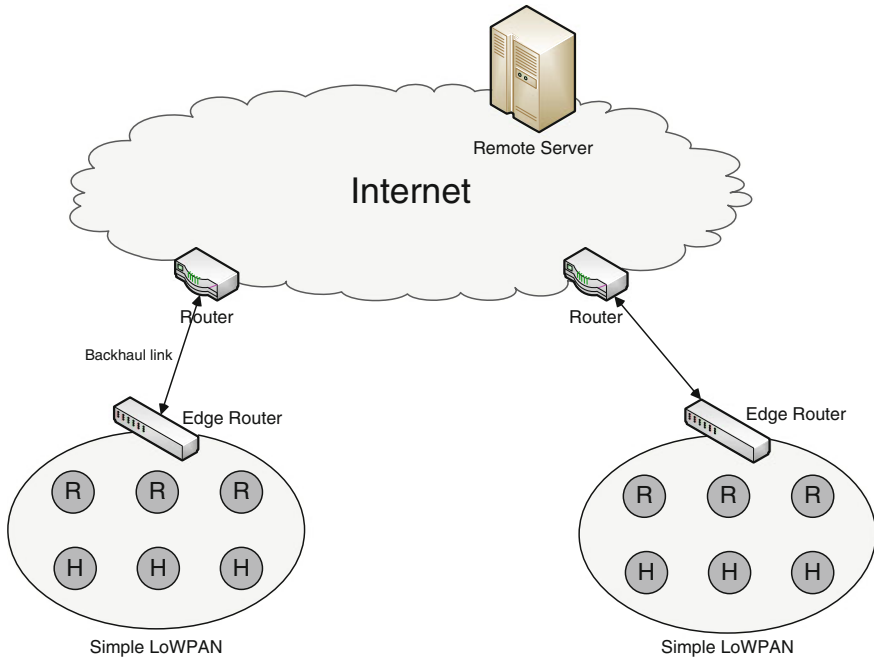


Fig. 2.27 Example of 6LoWPAN deployment

- **Maximum transmission unit:** Applications of 6LoWPAN should minimize packet sizes and avoid forcing a LoWPAN to fragment IPv6 packets.
- **Application protocols:** End-to-end application protocols should make use of UDP and compact payload formats to suit for use with 6LoWPAN nodes.
- **Firewalls and network address translators:** When connecting 6LoWPAN with the Internet, the issues of firewalls and network address translators are not avoidable.
- **IPv4 interconnectivity:** IPv4 and IPv6 are concurrently used in the Internet. It might be necessary for 6LoWPAN nodes to interact with IPv4 nodes or across IPv4 networks.
- **Security:** Connecting 6LoWPAN nodes with the Internet brings benefits and risks as well. Security should always be a major concern.

2.6 Summary

This chapter introduces the principle of wireless sensor networks, particularly IEEE 802.15.4, ZigBee and 6LoWPAN. Both ZigBee and 6LoWPAN are built on top of IEEE 802.15.4. Therefore, IEEE 802.15.4 lays down the foundation for low-rate, low-power wireless sensor networks.

It might be true that developers of wireless sensor networks often deal with ZigBee and 6LoWPAN stacks rather than IEEE 802.15.4 stack. These two LoWPAN stacks are the most popular and are currently independent of each other. ZigBee cannot communicate directly with the Internet due to its lack of native IP stack processing. There are a few approaches, which aim to interconnect 6LoWPAN with ZigBee. The first approach is to put the IPv6 stack on top of a ZigBee network layer. A global unicast IPv6 address is assigned to every ZigBee node; conversely every IPv6 node is assigned with a ZigBee short address. The gateway is responsible for dealing with sending and receiving all network traffic. It will then handle encapsulation and decapsulation of all the transmitted packets from and to an IPv6 network, or WAN respectively (Wang et al. 2007). The second approach is a dual stack design, with both the 6LoWPAN stack and the ZigBee stack working on the same IEEE 802.15.4 MAC layer. This approach allows both the 6LoWPAN and ZigBee stack to coexist on the same 802.15.4 MAC. Although it enables both IPv6 and ZigBee functions to be applied to a same node, however only one of the specifications can be used at any time. Finally, a gateway that allows both 6LoWPAN and ZigBee devices to be converted to IPv6 would be desirable (Hossen et al. 2010).

References

- Akyildiz, I.F., Su, W., Sankarasubramaniam, Y., Cyirci, E.: A survey on sensor networks. *IEEE Commun. Mag.* **40**(8), 102–114 (2002)
- Aschenbrenner, J.R.: Open systems interconnection. *IBM Syst. J.* **25**(3/4), 369–379 (1986)
- Elahi, A., Gschwender, A.: *ZigBee Wireless Sensor and Control Network*. Prentice Hall, NJ (2009)
- Gutiérrez, J.A., Callaway, E.H., Barrett, R.L.: *Low-Rate Wireless Personal Area Networks Enabling Wireless Sensors with IEEE 802.15.4*. IEEE Press, New York (2004)
- Hossen, M.S., Kabir, A.F.M.S., Khan, R.H., Azfar, A.: Interconnection between 802.15.4 devices and IPv6: implications and existing approaches. *Int. J. Comput. Sci.* **7**(1), 19–31 (2010)
- IEEE: Part 15.4: Wireless medium access control (MAC) and physical layer (PHY) specifications for low-rate wireless personal area networks (LR-WPANs) (2003)
- Kushalnagar, N., Montenegro, G., Schumacher, C.: IPv6 over low-power wireless personal area networks (6LoWPANs): Overview, assumptions, problem statement, and goals. RFC4919, Internet Engineering Task Force (2007)
- Lewis, F.L.: Smart environments: Technology, protocol and applications. In: Cook, D.J., Das, S.K. (eds.) *Wireless Sensor Networks*, 1st edn, pp. 13–46. Wiley, New York (2004)
- Nicopolitidis, P., Obaidat, M.S., Papadimitriou, G.I., Pomportsis, A.S.: *Wireless Networks*. Wiley, New York (2003)
- Schurgers, C., Srivastava, M.B.: Energy efficient routing wireless sensor networks. In: *Military Communications Conference on Communications for Network-Centric Operations: Creating the Information Force*, vol. 1, pp. 357–361
- Shelby, Z., Bormann, C.: *6LoWPAN—The Wireless Embedded Internet*. Wiley, New York (2009)
- Wang, R.C., Chang, R.S., Chao, H.C.: Interworking between ZigBee/802.15.4 and IPv6/802.3 Network. In: *SigComm Conference on IPv6 (IPv6'07)*, Kyoto, Japan
- ZigBee: ZigBee specification, version 1.0. Available at www.zigbee.org (2004)
- ZigBee: ZigBee stack advanced user guide, JN-UG-3045 Revision 1.2, 6 Mar 2008

Wireless Sensor Networks

Principles, Design and Applications

Yang, S.-H.

2014, XVII, 293 p. 175 illus., 80 illus. in color.,

Hardcover

ISBN: 978-1-4471-5504-1