# 5

# MEDIUM ACCESS CONTROL PROTOCOLS FOR WIRELESS SENSOR NETWORKS

## 5.1 INTRODUCTION

WSNs are typically composed of a large number of low-cost, low-power, multifunctional wireless devices deployed over a geographical area in an ad hoc fashion and without careful planning. Individually, sensing devices are resource-constrained and therefore are only capable of a limited amount of processing and communication. It is the coordinated effort of these sensing devices, however, that bears promise for a significant impact on a wide range of applications in several fields, including science and engineering, military settings, critical infrastructure protection, and environmental monitoring [5.1–5.3].

Harnessing the potential benefits of WSNs requires a high-level of self-organization and coordination among the sensors to perform the tasks required to support the underlying application. At the heart of this collaborative effort to achieve communications is the need for the wireless sensor nodes to self-organize into a multihop wireless network. Consequently, the design of efficient communications and network protocols for WSNs becomes crucial for wireless sensor nodes to carry out successfully the mission for which they are deployed.

The establishment of a multihop wireless network infrastructure for data transfer requires the establishment of communication links between neighboring sensor nodes. Unlike communication over a guided medium in wired networks, however, communication in wireless networks is achieved in the form of electromagnetic signal transmission through the air. This common transmission medium must

*Wireless Sensor Networks: Technology, Protocols, and Applications,* by Kazem Sohraby, Daniel Minoli, and Taieb Znati

therefore be shared by all sensor network nodes in a fair manner. To achieve this goal, a medium access control protocol must be utilized. The choice of the medium access control protocol is the major determining factor in WSN performance. A number of access control protocols have been proposed for WSNs. The objective of this chapter is to discuss the fundamental design issues of medium access control for WSN methods and to provide an overview of these protocols. In Section 5.2, a description of the basic requirements of access control protocols is provided. In Section 5.3 we categorize the major media access control techniques used in shared medium access networks. In Section 5.4 we discuss specific requirements of access control methods for WSNs and describes several media access control (MAC) protocols for these networks.

## 5.2 BACKGROUND

Communication among wireless sensor nodes is usually achieved by means of a unique channel. It is the characteristic of this channel that only a single node can transmit a message at any given time. Therefore, shared access of the channel requires the establishment of a MAC protocol among the sensor nodes. The objective of the MAC protocol is to regulate access to the shared wireless medium such that the performance requirements of the underlying application are satisfied [5.4–5.7]. From the perspective of the Open Systems Interconnection (OSI) Reference Model (OSIRM), the MAC protocol functionalities are provided by the lower sublayer of the data link layer (DLL). The higher sublayer of the DLL is referred as the logical link control (LLC) layer. The subdivision of the data link layer into two sublayers is necessary to accommodate the logic required to manage access to a shared access communications medium. Furthermore, the presence of the LLC sublayer allows support for several MAC options, depending on the structure and topology of the network, the characteristics of the communication channel, and the quality of service requirements of the supported application.

Figure 5.1 depicts the OSI reference model and the logical architecture of the DLL for shared medium access in wireless networks. The physical layer (PHY) typically includes a specification of the transmission medium and the topology of the network. It defines the procedures and functions that must be performed by the physical device and the communications interface to achieve bit transmission and reception. It also coordinates the various functions necessary to transmit a stream of bits over the wireless communication medium. The major services provided by the physical layer typically include the encoding and decoding of signals, preamble generation and removal to achieve synchronization, and the transmission and reception of bits.

The MAC sublayer resides directly above the physical layer. It supports the following basic functions:

- The assembly of data into a frame for transmission by appending a header field containing addressing information and a trailer field for error detection

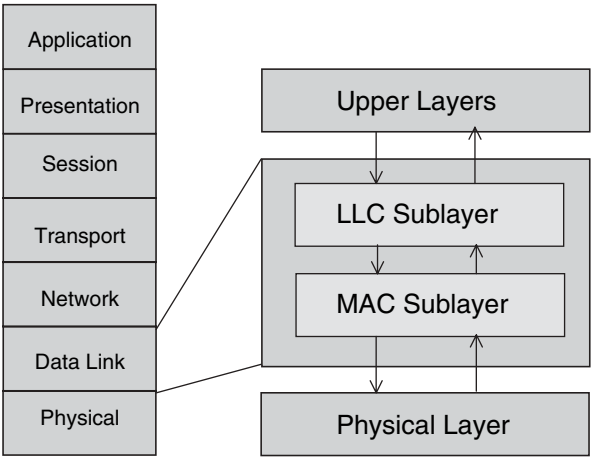



**Figure 5.1** Open systems interconnection reference model and data link layer architecture.

- The disassembly of a received frame to extract addressing and error control information to perform address recognition and error detection and recovery
- The regulation of access to the shared transmission medium in a way commensurate with the performance requirements of the supported application

The LLC sublayer of the DDL provides a direct interface to the upper layer protocols. Its main purpose is to shield the upper layer protocols from the characteristics of the underlying physical network, thereby providing interoperability across different types of networks. The use of the LLC sublayer, however, has been very limited, as interoperability is typically achieved by other network layer protocols.

## 5.3 FUNDAMENTALS OF MAC PROTOCOLS

One major difficulty in designing effective MAC protocols for shared access media arises from the spatial distribution of the communicating nodes [5.8]. To reach agreement as to which node can access the communication channel at any given time, the nodes must exchange some amount of coordinating information. The exchange of this information, however, typically requires use of the communication channel itself. This recursive aspect of the multiaccess medium problem increases the complexity of the access control protocol and consequently, the overhead required to regulate access among the competing nodes. Furthermore, spatial distribution does not allow a given node on the network to know the instantaneous status of other nodes on the network. Any information explicitly or implicitly gathered by any node is at least as old as the time required for its propagation through the communication channel.

Two main factors, the intelligence of the decision made by the access protocol and the overhead involved, influence the aggregate behavior of a distributed multiple-access protocol. These two factors are unavoidably intertwined. An attempt to improve the quality of decisions does not necessarily reduce the overhead incurred. On the other hand, reducing the overhead is likely to lower the quality of the decision. Thus, a trade-off between these two factors must be made.

Determining the nature and extent of information used by a distributed multiple-access protocol is a difficult task, but potentially a valuable one. An understanding of exactly what information is needed could lead to an appreciation of its value. Most of the proposed distributed multiple-access protocols for WSNs operate somewhere along a spectrum of information ranging from a minimum amount of information to perfect information. Furthermore, the information can be predetermined, dynamic global, or local. *Predetermined information* is known to all communicating nodes. *Dynamic global information* is acquired by different nodes during protocol operation. *Local information* is known to individual nodes. Predetermined and dynamic global information may result in efficient, potentially perfect coordination among the nodes. However, there usually is a high price to pay in terms of wasted channel capacity. The use of local information has potential to reduce the overhead required to coordinate the competing nodes, but may result in poor overall performance of the protocol.

The trade-off between the efficiency of the MAC protocol and the overhead required to achieve it has been at the basis of most of the access techniques for shared-medium networks. In the remaining part of this section, the performance metrics for the MAC protocol are described and the major common techniques to regulate access to the medium are discussed.

### 5.3.1 Performance Requirements

In trying to determine the performance requirements of MAC protocols, the scope of research has been very broad [5.9]. Traditionally, issues such as delay, throughput, robustness, scalability, stability, and fairness have dominated the design of MAC protocols [5.10, 5.11]. Following is a brief discussion of these performance metrics.

***Delay*** *Delay* refers to the amount of time spent by a data packet in the MAC layer before it is transmitted successfully. Delay depends not only on the network traffic load, but also on the design choices of the MAC protocol. For time-critical applications, the MAC protocol is required to support delay-bound guarantees necessary for these applications to meet their QoS requirements [5.12]. The precise semantics of the QoS requirements are application-dependent. Guaranteed delay bounds are usually provided through careful message scheduling both locally within a communicating node and globally among all nodes in the network. Two types of delay guarantees can be identified, probabilistic and deterministic. *Probabilistic delay guarantees* are typically characterized by an expected value, a variance and a confidence interval. *Deterministic delay guarantees* ensure a predictable number

of state transitions between message arrival and message transmission. Therefore, deterministic MAC schemes guarantee an upper bound for the access time. Determinism is a crucial requirement in a real-time environment, where the correctness of the application depends on the adherence of its underlying tasks to their specified execution deadline.

***Throughput*** *Throughput* is typically defined as the rate at which messages are serviced by a communication system. It is usually measured either in messages per second or bits per second. In wireless environments it represents the fraction of the channel capacity used for data transmission. Throughput increases as the load on the communication system increases initially. After the load reaches a certain threshold, the throughput ceases to increase, and in some cases, it may start to decrease. An important objective of a MAC protocol is to maximize the channel throughput while minimizing message delay.

***Robustness*** *Robustness*, defined as a combination of reliability, availability, and dependability requirements, reflects the degree of the protocol insensitivity to errors and misinformation. Robustness is a multidimensional activity that must simultaneously address issues such as error confinement, error detection and masking, reconfiguration, and restart. Achieving robustness in a time-varying network such as a WSN is difficult, as it depends strongly on the failure models of both the links and the communicating nodes.

***Scalability*** *Scalability* refers to the ability of a communications system to meet its performance characteristics regardless of the size of the network or the number of competing nodes. In WSNs, the number of sensor nodes may be very large, exceeding thousands and in some cases millions of nodes. In these networks, scalability becomes a critical factor. Achieving scalability is challenging, especially in time-varying environments such as wireless networks. A common approach to achieve scalability is to avoid relying on globally consistent network states. Another approach is to localize interactions among the communicating nodes, through the development of hierarchical structures and information aggregation strategies. Grouping sensor nodes into clusters, for example, allows the design of shared medium access protocols which are highly scalable. Similarly, aggregating information from different sensors allows the development of traffic patterns which can be exploited efficiently to scale the MAC protocol to a large number of sensor nodes.

***Stability*** *Stability* refers to the ability of a communications system to handle fluctuations of the traffic load over sustained periods of time. A stable MAC protocol, for example, must be able to handle instantaneous loads which exceed the maximum sustained load as long as the long-term load offered does not exceed the maximum capacity of the channel. Typically, the scalability of a MAC protocol is studied with respect to either delay or throughput. A MAC protocol is considered to be stable, with respect to delay, if the message waiting time is bounded. These systems can be characterized by a bounded backlog of messages in the transmission

queue. With respect to throughput, a MAC protocol is stable if the throughput does not collapse as the load offered increases. Accommodating load fluctuations while maintaining system stability is difficult to achieve in time-varying large-scale WSNs. One possible approach is for the MAC protocol to adapt to high fluctuations in the traffic load through careful scheduling of bursty traffic.

***Fairness*** A MAC protocol is considered to be *fair* if it allocates channel capacity evenly among the competing communicating nodes without unduly reducing the network throughput. Achieving fairness among competing nodes is desirable to achieve equitable QoS and avoid situations where some nodes fare better than other nodes. As a result, no application is starved or penalized excessively. It is worth noting that the definition of fairness above assumes that the demands of all communicating nodes, expressed in terms of channel capacity, are equivalent. It could be the case, however, that the network must accommodate various traffic sources with different traffic generation patterns and a wide range of QoS requirements. To accommodate heterogeneous resource demands, communicating nodes are assigned different weights to reflect their relative resource share. Proportional fairness is then achieved based on the weights assigned. A MAC protocol is considered to be proportionally fair if it is not possible to increase the allocation of any competing node without reducing the service rate of another node below its proportional fair share.

Fair resource allocation in wireless networks is difficult to achieve, as global information may be required to coordinate access to the communication medium among all contending stations. The time-varying characteristics of the wireless links makes it difficult to compute the fair share of each contending node, even if a centralized resource allocation approach is used.

***Energy Efficiency*** A sensor node is equipped with one or more integrated sensors, embedded processors with limited capability, and short-range radio communication ability as discussed in Chapter 3. These sensor nodes are powered using batteries with small capacity. Unlike in standard wireless networks, wireless sensor nodes are often deployed in unattended environments, making it difficult to change their batteries. Furthermore, recharging sensor batteries by energy scavenging is complicated and volatile. These severe constraints have a direct impact on the lifetime of a sensor node. As a result, energy conservation becomes of paramount importance in WSNs to prolong the lifetime of sensor nodes. One possible approach to reducing energy consumption at a sensor node is to use low-power electronics. The integration of low-power chips in the design of sensor nodes is a necessary step toward achieving high levels of power efficiency. Energy gains resulting from energy-efficient chip design, however, can easily be squandered if the processing and communication capabilities of the sensor node are not operated efficiently. Achieving this goal requires the design of energy-aware communication protocols.

Energy efficiency is one of the most important issues in the design of MAC protocol for wireless sensor nodes. Several sources contribute to energy inefficiency in

MAC-layer protocols [5.44]. The first source of energy waste is *collision*, which occurs when two or more sensor nodes attempt to transmit simultaneously. The need to retransmit a packet that has been corrupted by a collision increases energy consumption. The second source of energy waste is *idle listening*. A sensor node enters this mode when it is listening for a traffic that is not sent. This energy expended monitoring a silent channel can be high in several sensor network applications. The third source of energy waste is *overhearing* which occurs when a sensor node receives packets that are destined to other nodes. Due to their low transmitter output, receivers in sensor nodes may dissipate a large amount of power. The fourth major source of energy waste is caused by *control packet overhead*. Control packets are required to regulate access to the transmission channel. A high number of control packets transmitted, relative to the number of data packets delivered indicates low energy efficiency. Finally, *frequent switching* between different operation modes may result in significant energy consumption. Limiting the number of transitions between sleep and active modes, for example, leads to considerable energy saving.

Energy-efficient link-layer protocols achieve energy savings by controlling the radio to eliminate, or at least reduce, energy waste caused by the sources noted above. Further energy gains can be achieved using comprehensive energy management schemes which focus not only on the sensor node radio, but equally important, on other sources of energy consumption.

### 5.3.2 Common Protocols

The choice of the MAC method is the major determining factor in the performance of a WSN. Several strategies have been proposed to solve the shared medium access problem. These strategies attempt, by various mechanisms, to strike a balance between achieving the highest-quality resource allocation decision and the overhead necessary to reach this decision. These strategies can be classified in three major categories: fixed assignment, demand assignment, and random assignment.

***Fixed-Assignment Protocols*** In fixed-assignment strategies, each node is allocated a predetermined fixed amount of the channel resources. Each node uses its allocated resources exclusively without competing with other nodes. Typical protocols that belong in this category include frequency-division multiple access (FDMA), time-division multiple access (TDMA), and code-division multiple access (CDMA) [5.13].

*FDMA* The FDMA scheme is used by radio systems to share the radio spectrum. Based on this scheme, the available bandwidth is divided into subchannels. Multiple channel access is then achieved by allocating communicating nodes with different carrier frequencies of the radio spectrum. The bandwidth of each node's carrier is constrained within certain limits such that no interference, or overlap, occurs between different nodes. The scheme requires frequency synchronization among communicating nodes. Communication is achieved by having the receiver tune to the channel used by the transmitter.

*TDMA* TDMA is digital transmission technology that allows a number of communicating nodes to access a single radio-frequency channel without interference. This is achieved by dividing the radio frequency into time slots and then allocating unique time slots to each communicating node. Nodes take turns transmitting and receiving in a round-robin fashion. It is worth noting, however, that only one node is actually using the channel at any given time for the duration of a time slot.

*CDMA* CDMA is a spread spectrum (SS)–based scheme that allows multiple communicating nodes to transmit simultaneously. Spread spectrum is a radio-frequency modulation technique in which the radio energy is spread over a much wider bandwidth than that needed for the data rate. Systems based on spread-spectrum technology transmit an information signal by combining it with a noiselike signal of a much larger bandwidth to generate a wideband signal. Consequently, the signal transmitted occupies a larger bandwidth than that normally required to transmit the original information. Using wideband noiselike signals makes it hard to detect, intercept, or demodulate the original signal.

Most common spread spectrum–based systems use either frequency hopping (FH) or direct sequence (DS), although hybrid systems may use some combination of the two types. Frequency-hopping spectrum systems (FHSS) modulate the data signal with a narrowband carrier signal that hops over time from one frequency to another in a pseudorandom but predictable pattern selected from a wideband of frequencies [5.14]. For the signal to be decoded properly, the hopping patterns of the transmitting and receiving radios must be synchronized.

Direct-sequence spread-spectrum systems (DSSSs) divide the stream of information to be transmitted into small chunks, each of which is allocated to a frequency channel across the spectrum. A data signal at the sending node is combined with a higher-data-rate bit sequence, referred to as *chipping code*, which divides the user data according to a spreading ratio. The chipping code is a redundant bit pattern for each bit that is transmitted. This redundancy increases the resistance to interference of the signal transmitted and improves the likelihood of recovering the original data if one or more bits in the pattern are damaged during transmission.

***Demand Assignment Protocols*** The main objective of demand assignment protocols is to improve channel utilization by allocating the capacity of the channel to contending nodes in an optimum or near-optimum fashion. Unlike fixed-assignment schemes, where channel capacity is assigned exclusively to the network nodes in a predetermined fashion regardless of their current communication needs, demand assignment protocols ignore idle nodes and consider only nodes that are ready to transmit. The channel is allocated to the node selected for a specified amount of time, which may vary from a fixed-time slot to the time it takes to transmit a data packet.

Demand assignment protocols typically require a network control mechanism to arbitrate access to the channel between contending nodes. Furthermore, a logical control channel, other than the data channel, may be required for contending stations to dynamically request access to the communication medium. Depending

on the characteristics of the protocol, the need to request access to the channel may delay data transmission. Demand assignment protocols may be further classified as centralized or distributed. Polling schemes are representative of centralized control, whereas token- and reservation-based schemes use distributed control.

*Polling* A widely used demand assignment scheme is polling. In this scheme, a master control device queries, in some predetermined order, each slave node about whether it has data to transmit. If the polled node has data to transmit, it informs the controller of its intention to transmit. In response, the controller allocates the channel to the ready node, which uses the full data rate to transmit its traffic. If the node being polled has no data to transmit, it declines the controller's request. In response, the controller proceeds to query the next network node. The main advantage of polling is that all nodes can receive equal access to the channel. Preference can, however, be given to high-priority nodes by polling them more often. The major drawback of polling is the substantial overhead caused by the large number of messages generated by the controller to query the communicating nodes. Furthermore, the efficiency of the polling scheme depends on the reliability of the controller.

*Reservation* The basic idea in a reservation-based scheme is to set some time slots for carrying reservation messages. Since these messages are usually smaller than data packets, they are called *minislots*. When a station has data to send, it requests a data slot by sending a reservation message to the master in a reservation minislot. In some schemes, such as in fixed-priority-oriented demand assignment, each station is assigned its own minislot. In other schemes, such as in packet demand assignment multiple access, stations contend for access to a minislot using one of the distributed packet-based contention schemes, such as slotted ALOHA [5.15,5.16]. When the master receives the reservation request, it computes a transmission schedule and announces the schedule to the slaves.

In a reservation-based scheme, if each station has its own reservation minislot, collision can be avoided. Moreover, if reservation requests have a priority field, the master can schedule urgent data before delay-insensitive data. Packet collisions can happen only when stations contend for the minislot, which use only a small fraction of the total bandwidth. Thus, the largest part of the bandwidth assigned to data packets is used efficiently.

***Random Assignment Protocols*** In fixed-assignment schemes, each communicating node is assigned a frequency band in FDMA systems or a time slot in TDMA systems. This assignment is static, however, regardless of whether or not the node has data to transmit. These schemes may therefore be inefficient if the traffic source is bursty. In the absence of data to be transmitted, the node remains idle, thereby resulting in the allocated bandwidth to be wasted. Random assignment strategies attempt to address this shortcoming by eliminating preallocation of bandwidth to communicating nodes.

Random assignment strategies do not exercise any control to determine which communicating node can access the medium next. Furthermore, these strategies

do not assign any predictable or scheduled time for any node to transmit. All backlogged nodes must contend to access the transmission medium. Collision occurs when more than one node attempts to transmit simultaneously. To deal with collisions, the protocol must include a mechanism to detect collisions and a scheme to schedule colliding packets for subsequent retransmissions.

Random access protocols were first developed for long radio links and for satellite communications. The ALOHA protocol, also referred to as *pure ALOHA*, was one of the first such media access protocols. ALOHA simply allows nodes to transmit whenever they have data to transmit. Efforts to improve the performance of pure ALOHA lead to the development of several schemes, including carrier-sense multiple access (CSMA), carrier-sense multiple access with collision detection (CSMA/CD), and carrier-sense multiple access with collision avoidance (CSMA/CA).

*ALOHA* ALOHA is a simple random assignment protocol developed to regulate access to a shared transmission medium among uncoordinated contending users. The protocol was originally developed for ground-based packet broadcasting networks and was used to connect remote users to mainframe computers [5.15,5.16]. Channel access in pure ALOHA is completely asynchronous and independent of the current activity on the transmission medium. A node is simply allowed to transmit data whenever it is ready to do so. Upon completing the data transmission, the communicating node listens for a period of time equal to the longest possible round-trip propagation time on the network. This is typically the time it takes for the signal to travel between the two most distant nodes in the network. If the node receives an acknowledgment for data transmitted before this period of time elapses, the transmission is considered successful. The acknowledgment is issued by the receiving station after it determines the correctness of the data received by examining the error check sum. In the absence of an acknowledgment, however, the communicating node assumes that the data are lost due to errors caused by noise on the communication channel or because of collision, and retransmits the data. If the number of transmission attempts exceeds a specified threshold, the node refrains from retransmitting the data and reports a fatal error.

ALOHA is simple protocol that requires no central control, thereby allowing nodes to be added and removed easily. Furthermore, under light-load conditions, nodes can gain access to the channel within short periods of time. The main drawback of the protocol, however, is that network performance degrades severely as the number of collisions rises rapidly with increased load. To improve the performance of pure ALOHA, *slotted ALOHA* was proposed. In this scheme, all communication nodes are synchronized and all packets have the same length. Furthermore, the communication channel is divided into uniform time slots whose duration is equal to the transmission time of a data packet. Contrary to pure ALOHA, transmission can occur only at a slot boundary. Consequently, collision can occur only in the beginning of a slot, and colliding packets overlap totally in time.

Limiting channel access to slot boundaries results in a significant decrease in the length of collision intervals, resulting in increased utilization of the underlying

communication channel. Despite this performance improvement, however, ALOHA and pure ALOHA remain inefficient under moderate to heavy load conditions. Furthermore, in networks where the propagation delay is much shorter than the transmission time of a data packet, nodes can become aware almost immediately of an ongoing packet transmission. This observation led to the development of a new class of media access schemes, whereby before a transmission is attempted, a station that has a packet to transmit first "listens" to the channel to determine if it is busy. Carrier sensing forms the basis of the CSMA protocol.

*CSMA* CSMA operates both in continuous time, unslotted CSMA, and in discrete time, slotted CSMA. Furthermore, the class of CSMA protocols can be divided into two categories, nonpersistent CSMA and persistent CSMA, depending on the strategy used to acquire a free channel and the strategy used to wait for a busy channel to become free. In *nonpersistent* CSMA protocol, when a node becomes ready to transmit a packet, it first senses the carrier to determine if another transmission is in progress. If the channel is idle, the node transmits its packet immediately and waits for an acknowledgment. In setting the acknowledgment timeout value, the node must take into account the round-trip propagation delay and the fact that the receiving node must also contend for the channel to transmit the acknowledgment. Estimating the average contention time required for a successful transmission is difficult, as it depends on the traffic load and the number of stations contending. In the absence of an acknowledgment, before a timeout occurs, the sending node assumes that the data packet is lost due to collision or noise interference. The station schedules the packet for retransmission. If the channel is busy, the transmitting node "backs off" for a random period of time after which it senses the channel again. Depending on the status of the channel, the station transmits its packet if the channel is idle, or enters the back-off mode if the channel is busy. This process is repeated until the data packet is transmitted successfully.

The nonpersistent CSMA protocol minimizes the interference between packet transmissions, as it requires stations that find the channel busy to reschedule their transmissions randomly. The major drawback of the nonpersistent CSMA scheme, however, results from the fact that a channel may become idle during the back-off time of a contending station. The unnecessary waste of channel capacity can reduce significantly the overall network throughput. The need to address the shortcomings of nonpersistent CSMA led to the development of a class of $p$-persistent CSMA schemes. These schemes differ in the algorithm they use to acquire a free channel. The 1-persistent scheme never allows the channel to remain idle if a node is ready to transmit. Based this scheme, a node ready to transmit a data packet first senses the channel. If the channel is free, the node transmits its message immediately. If the channel is busy, however, the node *persistently* continues to listen until the channel becomes idle. Transmission is attempted immediately after the channel is sensed idle.

The $p$-persistent algorithm represents a compromise between the nonpersistent and 1-persistent schemes. Based on this algorithm, a node that senses the channel idle transmits its packet with probability $p$. With probability $(1 - p)$, the station

waits for a specific time period before attempting to transmit the packet again. The value of the waiting period is typically set to equal the maximum propagation delay between the most distant nodes in unslotted CSMA or to a slot time in slotted ALOHA. At the end of the waiting period, the node senses the channel again. If the channel is busy, the node continues to listen until the channel becomes idle. When the channel becomes idle, the node repeats the foregoing $p$-persistent channel acquisition algorithm. This process continues until the data packet is transmitted successfully.

The value of $p$ plays a major role in the stability of the protocol. Under heavy traffic load, if the value of $p$ is large, multiple nodes will attempt to transmit, thereby increasing the likelihood of collisions. Since the value of $p$ is high, colliding nodes will probably attempt to retransmit almost immediately after the collision occurs. Worst yet, these retransmissions may have to compete with new transmissions from other nodes, almost guaranteeing more collisions. Eventually, as the number of contending stations increases, the network throughput decreases drastically. To avoid this situation, the value of $p$ must be small. As the value of $p$ is made small, the number of collisions decreases. Under a light traffic load, however, a small value of $p$ may cause a contending station to wait unnecessarily for long delays before transmitting its data packets. Careful consideration of the offered traffic rate is therefore necessary to select a value of $p$ effectively.

*CSMA/CD* In networks where the propagation delay is small relative to the packet transmission time, the CSMA scheme and its variants can result in smaller average delays and higher throughput than with the ALOHA protocols. This performance improvement is due primarily to the fact that carrier sensing reduces the number of collisions and, more important, the length of the collision interval. The main drawback of CSMA-based schemes, however, is that contending stations continue transmitting their data packets even when collision occurs. For long data packets, the amount of wasted bandwidth is significant compared with the propagation time. Furthermore, nodes may suffer unnecessarily long delays waiting for the transmission of the entire packet to complete before attempting to transmit the packet again.

To overcome the shortcomings of CSMA-based schemes and further reduce the collision interval, networks using CSMA/CD extend the capabilities of a communicating node to listen while transmitting. This allows the node to monitor the signal on the channel and detect a collision when it occurs. More specifically, if a node has data to send, it first listens to determine if there is an ongoing transmission over the communication channel. In the absence of any activity on the channel, the node starts transmitting its data and continues to monitor the signal on the channel while transmitting. If an interfering signal is detected over the channel, the transmitting station immediately aborts its transmission. This reduces the amount of bandwidth wasted due to collision to the time it takes to detect a collision. When a collision occurs, each contending station involved in the collision waits for a time period of random length before attempting to retransmit the packet. The length of time that a colliding node waits before it schedules packet retransmission is determined by a probabilistic algorithm, referred to as the *truncated binary exponential back-off*

*algorithm*. The algorithm derives the waiting time after collision from the slot time and the current number of attempts to retransmit.

The major drawback of CSMA/CD is the need to provision sensor nodes with collision detection capabilities. Sensor nodes have a very limited amount of storage, processing power, and energy resources. These limitations impose severe constraints on the design of the MAC layer. Support for collision detection in WSNs is not possible without additional circuitry. In particular, wireless transceivers are typically half-duplex. To detect collision, the sensor node must therefore be capable of ''listening'' while ''talking.'' The complexity and cost of sensor nodes, however, are intended to be low and scalable to enable broad adaptations of the technology in cost-sensitive applications where deployment of large number of sensors is expected. Consequently, the design of physical layer must be optimized to keep the cost low.

Another important factor that works against using a CSMA/CD-based strategy to regulate access to a shared medium in a wireless environment is the difficulty of detecting collision in a wireless environment. In a wired medium, the low attenuation of the signal makes it such that the values of signal-to-noise ratio are nearly the same at the transmitter and the receiver. As a result, the detection of a collision at the transmitter can be used to infer unambiguously that a collision also occurred at the receiver. In wireless environments, the time-varying properties of the wireless channel, coupled with the rapid decrease of the signal power over distance, makes it difficult for the transmitting sensor node to infer unambiguously either the occurrence or the absence of a collision at the receiving node [5.17]. This drawback severely limits the applicability of collision detection–based schemes in WSNs.

*CSMA/CA* Carrier sensing prior to transmission is an effective approach to increase the throughput efficiency in shared-medium access environments. Although applicable in wireless environments, the scheme is susceptible to two problems, commonly referred to as the *hidden-* and *exposed-node problems* [5.4,5.16,5.18]. The hidden- and exposed-node problems result indirectly from the time-varying properties of the wireless channel, which are caused by physical phenomena such as noise, fading, attenuation, and path loss. These interferences, combined with the rapid decrease in the power received with the distance between the sender and receiver, limit the maximum transmission range that can be achieved by a sending node. This limitation and the fact that CSMA is designed to avoid collision by sensing the signal in the vicinity of the transmitter give rise to the hidden- and exposed-node problems.

A *hidden node* is defined as a node that is within the range of the destination node but out of range of the transmitting node. To illustrate this example, consider Figure 5.2, where node B is within the transmission range of nodes A and C. Furthermore, assume that nodes A and C are outside their mutual transmission ranges. Consequently, any transmission from either of the two nodes will not reach the other node. Given this network configuration, assume that node A needs to transmit a data packet to node B. According to the CSMA protocol, node A senses the channel and determines that it is free. Node A then proceeds to transmit its

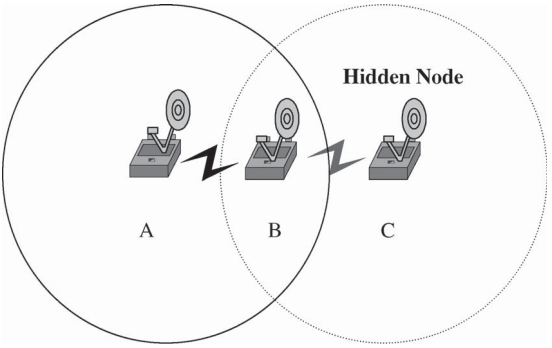



**Figure 5.2** Hidden-node scenario in wireless sensor networks.

packet. Assume now that before node A completes its transmission to node B, node C decides to transmit a data packet to node B. Using the CSMA protocol, node C senses the channel and also determines that the channel is free, since node C, which is outside the transmission range of node A, cannot hear the signal transmitted by node A. As a result, both transmissions collide at node B, thereby causing the loss of both data packets. Notice that neither node A nor node C is aware of the collision, since it happens at the receiver. This feature is intrinsic to wireless networks and constitutes a fundamental difference in the way that collisions are dealt with in wired and wireless environments.

The exposed-node problem is also the result of the intrinsic property of the wireless channel. An *exposed node* is a node that is within the range of the sender but out of the range of the destination. To illustrate the exposed-node problem, consider the network depicted in Figure 5.3, where node B is within the transmission range of nodes A and C, nodes A and C are outside their mutual transmission ranges, and node D is within the transmission range of node C. Assume that node B wants to transmit a message to node A. Node B executes the CSMA protocol to sense the channel, determines that the channel is free, and proceeds to transmit the data packet

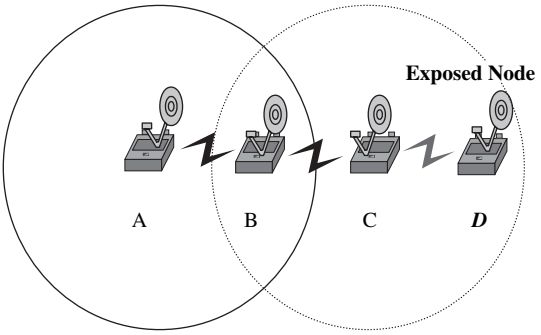



**Figure 5.3** Exposed node scenario in wireless sensor networks.

to node A. Assume now that node C needs to send a packet to D. Node C follows the CSMA rule and first senses the channel. Due to the ongoing transmission between nodes B and A, node C determines that the channel is busy and delays the transmission of its packet to a later time. It is clear, however, that this delay is unnecessary, since the transmission from node C to node D would have been completed successfully, as node D is outside the range of node B.

Several approaches have been proposed to eliminate, or at least reduce, the impact of the hidden- and exposed-node problems on the network throughput. The first approach is based on the use of a *busy tone*. The basic idea of the busy-tone approach stems from the observation that collisions occur at the receiving node whereas CSMA is performed at the transmission node. To address the disparity between the design goals of CSMA as originally specified and application of the protocol to wireless environments, the busy-tone approach requires the use of two separate channels: a data channel and a control channel. The data channel is used to transmit data exclusively. The control channel is used by the receiver to signal to the remaining nodes in the network that it is in the process of receiving data.

Immediately after the node starts to receive a data packet, which carries its address in the destination address field, the node initiates the emission of an unmodulated wave on the control channel, indicating that its receiver is busy. The node continues to transmit the busy tone at the same time that it is receiving the data packet until the packet is fully received. Before transmitting a data packet, the sending node must first sense the control channel for the presence of a busy tone. The node proceeds to transmit the data packet only if the control channel is free. Otherwise, the sending node defers its transmission until the control channel is no longer busy.

The busy-tone approach solves both the hidden- and exposed-node problems, assuming that the busy-tone signal is emitted at a level such that it is not too weak not to be heard by a node within the range of a receiver and not too strong to force more nodes than necessary to suppress their transmissions. The major drawback of the approach, however, is a node's need to operate in duplex mode, to be able to transmit and receive simultaneously. This requirement increases the design complexity of a node significantly, thereby increasing its cost and power consumption.

The second approach to deal with the hidden-node problem is based on collision avoidance [5.19]. This is achieved using a procedure referred to as the ready-to-send (RTS), clear-to-send (CTS) handshake. Using this handshake procedure, the CSMA/CA scheme requires that nodes apply a standard mechanism to avoid collision of wireless messages. Since a node cannot detect if a collision has occurred, it attempts to avoid collisions by waiting for the wireless medium to be clear for the amount of time it takes for a packet to propagate through the entire medium: the time required to send a packet between the most distant nodes in the network. When a node intends to transmit a data packet, it first senses the carrier to determine if another node is already transmitting. If no other transmissions are sensed, the node sends a short RTS packet to the intended recipient of the data packet. If the recipient is, in fact, idle and senses that the medium is clear, it sends

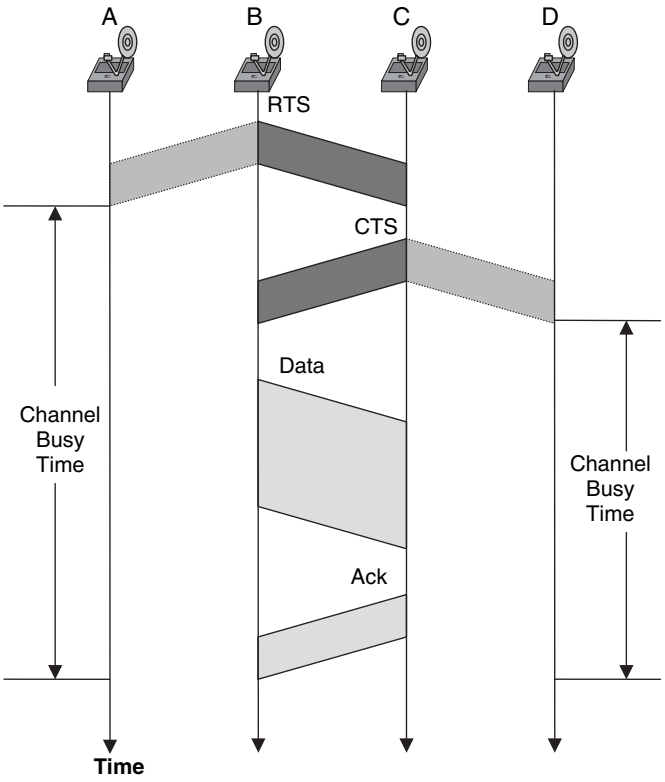



**Figure 5.4** Collision avoidance using RTS/CTS handshake.

a short CTS packet in reply. Upon receiving the CTS packet, the transmitting node sends the actual data packet to its intended recipient. If after a predetermined period of time, the transmitting station does not receive a CTS packet in reply to its RTS packet, it waits a random period of time before repeating the RTS/CTS handshake procedure.

The use of the RTS/CTS handshake procedure in CSMA/CA schemes to avoid collisions is depicted in Figure 5.4. In this scenario, node B intends to transmit a data packet to node C. It senses the carrier to determine if any other node is already transmitting. After it determines that the channel is free, it transmits a RTS packet. In addition to the destination address, the packet also contains the duration field, which indicates the time necessary to complete the transmission of the packet and the receipt of the corresponding acknowledgment. In response, the intended recipient of the packet, node C in this case, transmits a CTS packet, which contains the remaining time until the completion of the transmission. Upon receiving the RTS packet, station A sets an internal timer to the remaining time until completion of the data packet transmission and avoids transmitting any packet until the timer expires. When node B receives the CTS packet, it proceeds to transmit its data

packet. Upon receiving the CTS packet, node D sets an internal timer to the remaining time until completion of the data and defers the transmission of any packets until the timer expires.

In many environments, the RTS/CTS handshake procedure is sufficient to greatly reduce collisions and increase bandwidth utilization. This procedure, however, does not completely solve the hidden-node problem. To illustrate this limitation, consider the following scenarios, depicted in Figure 5.5. In the scenario depicted in Figure 5.5*a*, node A senses the channel to be free and sends an RTS packet to node B. In reply, node B sends a CTS packet. Node C, which is in the transmission range of node B, starts receiving the CTS packet. Before the reception of this packet is complete, however, node D, which is in the transmission range of node C, sends a RTS packet. The latter packet collides with the CTS packet sent by node B. Meanwhile, node A, which receives the CTS packet correctly, proceeds to transmit its data packet to node B. Node D later times-out and retransmits its RTS packet. Since node C never received node B's CTS packet, it assumes that the channel is free and replies with a CTS packet to node D. Since node B is within the transmission range of node C, the latter packet collides with the data packet being transmitted by node A.

The scenario depicted in Figure 5.5*b* shows another case where collision avoidance fails, using the RTS/CTS handshake. In this scenario, node A senses the channel to be free and sends an RTS packet to node B. In reply, node B sends a CTS packet to node A. The CTS packet is received correctly by node A, which allows it to transmit its packet. The CTS packet is also received by node C, which is within the transmission range of node B. Since node C has started transmitting an RTS packet to node D, nearly at the same time that node B is transmitting its CTS packet; node C does not receive correctly the CTS packet sent by B. Node D, however, receives correctly the RTS packet sent by node C. In response, it sends a CTS packet to node C, thereby allowing it to start transmitting its data packet. Since node A did not complete transmission of its data packet to node B, node C's data transmission causes a collision at node B. Despite its failure to solve the hidden-node problem completely, the RTS/CTS handshake is used widely in wireless networks to avoid packet collisions and increase network throughput.

## 5.4 MAC PROTOCOLS FOR WSNs

The need to conserve energy is the most critical issue in the design of scalable and stable MAC layer protocols for WSNs. Several factors contribute to energy waste, including excessive overhead, idle listening, packet collisions, and overhearing. Regulating access to the media requires the exchange of control and synchronization information among the competing nodes. The explicit exchange of a large number of these control and synchronization packets may result in significant energy consumption. Long periods of idle listening may also increase energy consumption and decrease network throughput. In some cases, energy wasted by idle listening accounts for over one-half of the total energy consumed by a sensor during

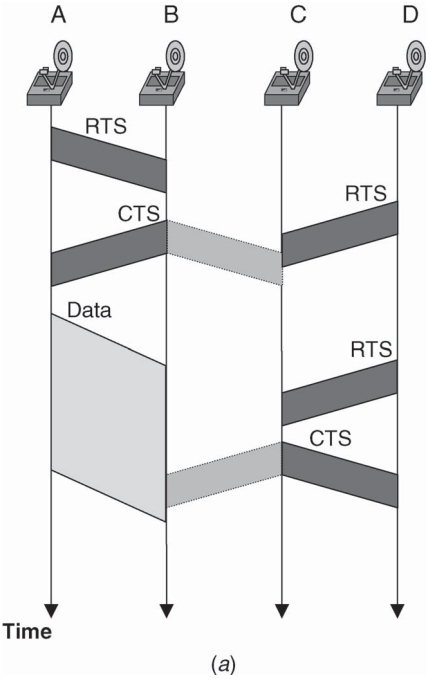



**Figure 5.5** Collision avoidance failure using RTS/CTS handshake.

its lifetime. The retransmission of colliding packets is yet another source of significant energy waste. A high number of these collisions may lead to severe performance degradation of the MAC-layer protocol. Similarly, excessive overhearing, which causes a node to receive and decode packets intended for other sensor nodes, unnecessarily increases energy consumption and can severely degrade the network throughput. These packets are eventually dropped after the node realizes that the destination address is different from its own address.

The main objective of most MAC-layer protocols is to reduce energy waste caused by collisions, idle listening, overhearing, and excessive overhead. These protocols can be categorized into two main groups: schedule- and contention-based MAC-layer protocols. *Schedule-based protocols* are a class of deterministic MAC-layer protocols in which access to the channel is based on a schedule. Channel access is limited to one sensor node at a time. This is achieved based on preallocation of resources to individual sensor nodes. *Contention-based MAC-layer protocols* avoid preallocation of resources to individual sensors. Instead, a single radio channel is shared by all nodes and allocated ondemand. Simultaneous attempts to access the communications medium, however, results in collision. The main objective of contention-based MAC layer protocols is to minimize, rather than completely avoid, the occurrence of collisions. To reduce energy consumption, these protocols differ in the mechanisms used to reduce the likelihood of a collision while minimizing overhearing and control traffic overhead.

Resolving collisions is typically achieved using distributed, randomized algorithms to reschedule channel access among competing sensor nodes. The basic approach used to reduce overhearing is to force nodes into a sleep state when they become inactive. Un-coordinating sleeping, however, can make communications among neighboring nodes difficult. To address this shortcoming, a variety of less restrictive schedules have been proposed by different MAC-layer protocols to coordinate the activity of the network sensors.

In the following section we first discuss schedule-based MAC-layer protocols for WSNs. We then briefly review a variety of contention-based MAC-layer protocols. We conclude the section with two cases studies. The first study focuses on S-MAC, a low-duty-cycle contention-based MAC-layer protocol specifically designed for WSNs. S-MAC strives to retain the flexibility of contention-based MAC-layer protocols, such as IEEE 802.11, while reducing energy waste caused by idle listening, collisions, overhearing, and excessive control overhear. S-MAC uses the concepts of low-duty-cycle coordinated sleep and wakeup time periods to reduce power consumption while achieving high throughput.

The second case study focuses on the IEEE MAC-layer protocol specification for a low-data-rate wireless personal area network standard: IEEE 802.15.4, Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (LR-WPANs). The IEEE 802.15.4 specification supports three traffic types: periodic, intermittent, and repetitive. Furthermore, the protocol specification supports fixed, portable, and moving devices operating at data rates ranging from 20 to 250 kbps. When lines of communication exceed 30ft, the standard allows for the creation of self-configuring

multihop network topologies [5.20]. It also provides features that allow devices operating under the standard to coexist with other wireless devices, such as those that comply with IEEE 802.11 and 802.15.1.

### 5.4.1 Schedule-Based Protocols

Schedule-based MAC protocols for WSNs assume the existence of a schedule that regulates access to resources to avoid contention between nodes. Typical resources include time, a frequency band, or a CDMA code. The main objective of schedule-based MAC protocols is to achieve a high level of energy efficiency in order to prolong the network lifetime. Other attributes of interest include scalability, adaptability to changes in traffic load, and network topology [5.21,5.22]. Most of the scheduled-based protocols for WSNs use a variant of a TDMA scheme whereby the channel is divided into time slots, as depicted in Figure 5.6. A set of $N$ contiguous slots, where $N$ is a system parameter, form a logical frame. This logical frame repeats cyclically over time. In each logical frame, each sensor node is assigned a set of specific time slots. This set constitutes the schedule according to which the sensor node operates in each logical frame. The schedule can be either fixed, constructed on demand on a per-frame basis by the base station to reflect the current requirements of sensor nodes and traffic pattern, or hybrid, in which case the structure varies over different time scales and sensor behavior.

Based on its assigned schedule, a sensor alternates between two modes of operation: active mode and sleep mode. In the active mode, the sensor uses its assigned slots within a logical frame to transmit and receive data frames. Outside their assigned slots, sensor nodes move into sleep mode. In this mode the sensor nodes switch their radio transceivers off to conserve energy.

Many variations on the basic TDMA protocol have been proposed for media access control in WSNs [5.23,5.24]. Next, we provide a brief review of some of these protocols.

***Self-Organizing Medium Access Control for Sensornets (SMACS)*** SMACS is a medium access control protocol to enable the formation of random network topologies without the need to establish global synchronization among the network

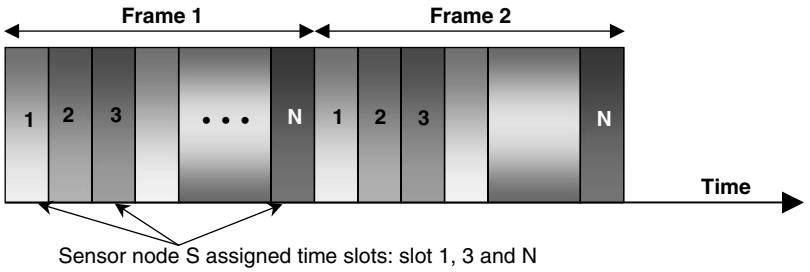



**Figure 5.6** TDMA-based MAC protocols for wireless sensor networks.

nodes [5.11,5.25]. A key feature of SMACS is its use of a hybrid TDMA/FH method referred to as *nonsynchronous scheduled communication* to enable links to be formed and scheduled concurrently throughout the network without the need for costly exchange of global connectivity information or time synchronization. Each node in the network maintains a TDMA-like frame, referred to as a *superframe*, for communication with known neighbors. The length of a superframe is fixed. Furthermore, the superframe is divided into smaller frames. The size of each frame is not fixed and may vary in time for a single node and also from node to node. SMACS requires that each node regularly execute a neighborhood discovery procedure to detect neighboring nodes. Each node establishes a link to each neighbor discovered by assigning a time slot to this link. The selection of time slots is such that the node talks only to neighbors at each time slot. However, since a node and its neighbors are not required to transmit at different slot times, the link establishment procedure must ensure that no interference occurs between adjacent links. This is achieved by randomly assigning a channel, selected from a large number of channels (FDMA), or spreading code (CDMA) to each link. Using the superframe structure, each node maintains its own time slot schedules with all its neighbors, and nodes are required to tune their radios to the proper frequency channel or CDMA code to achieve communication.

***Bluetooth*** Bluetooth is an emerging technology whose primary media access control is a centralized TDMA-based protocol [5.26]. Bluetooth is designed to replace cables and infrared links used to connect disparate electronic devices such as cell phones, headsets, PDAs, digital cameras, notebook computers, and their peripherals with one universal short-range radio link [5.27]. Bluetooth operates in the 2.45-GHz ISM frequency band. Its physical layer is based on a pseudorandom frequency-hopping scheme with a hopping frequency of 1.6 kHz and a scheme for hopping sequence allocation. A set of 79 hop carriers are defined with 1-MHz spacing. Each hop sequence defines a Bluetooth channel, which can support 1 Mbps.

A group of devices sharing a common channel is called a *piconet* [5.28]. Each piconet has a master unit which controls access to the channel, and at most seven slave devices as group participants. Each channel is divided into 625-ms slots. Each piconet is assigned a unique frequency-hopping pattern determined by the master's Bluetooth device address (48 bits) and clock. All slave devices follow their piconet-assigned hopping sequence. Different piconets use different hopping sequences, thereby guaranteeing their coexistence. Piconets can be interconnected, via *bridge* nodes, to form larger ad hoc networks, referred to as *scatternets*. Within a piconet, the master assigns each slave device a unique internal address of 3 bits. Access to the channel is regulated using a slotted time-division duplex (TDD) protocol in which the master uses a polling protocol to allocate time slots to slave nodes. A *Bluetooth frame*, representing one polling epoch, consists of two slots during which a packet can be exchanged between the piconet master and the slave being polled. The master polls the slave devices continuously for communication. A slave can communicate in a slot only if the master has addressed it in a previous slot. Packets

can be one, three, or five time slots long and are transmitted in consecutive slots. A packet can be more than one slot long if the communication is asynchronous.

To reduce energy consumption, Bluetooth specifies four operational modes: active, sniff, hold, and park. In the *active mode*, the slave listens for packet transmission from the master. On receiving a packet, it checks the address and packet length field of the packet header. If the packet does not contain its own address, the slave sleeps for the duration of the remaining packet transmission. The intended slave, however, remains active and receives the packet payload in the following reserved slot. The *sniff mode* is intended to reduce the duty cycle of a slave's listen activity. In this mode, the master transmits to the slave only in specified periodic time slots within a predefined sniff time interval. A slave in sniff mode listens for the master transmissions only during the specified time slots for any possible transmission to it. In the *hold mode*, a slave goes into the sleep mode for a specified amount of time, referred to as the *hold time*. When the hold time expires, the slave returns to the active mode. In the *park mode*, the slave stays in the sleep state for an unspecified amount of time. The master has to awake the slave explicitly and bring it into the active mode at a future time.

Bluetooth specifies four types of communication between nodes within and across piconets: *intra piconet unicast*, for slave-to-slave communication within a piconet; *intra piconet broadcast*, to support broadcasting by a slave to all participants within its piconet; *inter piconet unicast*, for piconet-to-piconet communications; and *inter piconet broadcast*, for piconet-to-all scatternet node communications.

For intra piconet unicast communication, the source slave writes its own MAC address in the corresponding field of the data packet and sets the forward field to 1 and the destination address of the packet to the targeted destination node. Upon receiving the message, the master checks the forward field. If it is set, the master replaces the MAC address field with its MAC address and sends the message to the intended slave device indicated by the destination address of the original packet.

For intra piconet broadcast communication, the source slave writes its own MAC address and sets the forward field to 1 and the destination address to 000. Upon receiving the message, the master notices that the forward field is set. In response, the master replaces the MAC address with its own address and sends the message to all nodes in its piconet.

For inter piconet unicast communication, the source device sends the data packet with its own MAC address and sets the forward field to 1, the broadcast field to 1 and the destination address to the relay of the next piconet. Furthermore, the source device sets the routing vector field (RVF) of the packet to contain the logical path to the targeted destination device in the intended piconet. The RVF is a sequence of tuples of the form (LocId, Mac_Addr), where LocId represents the identity of the local master and Mac_Addr its corresponding piconet MAC address. Upon receiving the message, the master forwards it to the relay node. The relay extracts from the RVF the next pair, containing the local identity and the MAC address of the master, and sends the message to this master. This process is repeated until the RVF becomes empty, signaling that the destination device has been reached.

For inter piconet broadcast communication, the source device creates a packet containing its own MAC address and sets the forward and broadcast fields of the packet to 1 and the destination address to 000. The packet is then sent to the master. When the master notices that the broadcast field is set to 1, it sends the packet to all the slaves within its piconet, including relay nodes. When a relay node receives the broadcast packet, it forwards it to all masters to which it is connected, except the one from which it came.

***Low-Energy Adaptive Clustering Hierarchy (LEACH)*** LEACH takes a hierarchical approach and organizes nodes into clusters. Within each cluster, nodes take turns to assume the role of a cluster head. LEACH uses TDMA to achieve communication between nodes and their cluster head [5.29–5.31]. The cluster head forwards to the base station messages received from its cluster nodes.

The cluster head node sets up a TDMA schedule and transmits this schedule to all nodes in its cluster. The schedule prevents collisions among data messages. Furthermore, the schedule can be used by the nodes to determine the time slots during which they must be active. This allows each cluster node, except for the head cluster, to turn off their radio components until its allocated time slots. LEACH assumes that cluster nodes start the cluster setup phase at the same time and remain synchronized thereafter. One possible mechanism to achieve synchronization is to have the base station send out synchronization pulses to the all the nodes [5.32,5.33].

To reduce intercluster interference, LEACH uses a *transmitter-based code assignment* scheme. Communications between a node and its cluster head are achieved using direct-sequence spread spectrum (DSSS), whereby each cluster is assigned a unique spreading code, which is used by all nodes in the cluster to transmit their data to the cluster head. Spreading codes are assigned to cluster heads on a first-in, first-served basis, starting with the first cluster head to announce its position, followed by subsequent cluster heads. Nodes are also required to adjust their transmit powers to reduce interference with nearby clusters.

Upon receiving data packets from its cluster nodes, the cluster head aggregates the data before sending them to the base station. The communication between a cluster head and a base station is achieved using fixed spreading code and CSMA. Before transmitting data to the base station, the cluster head must sense the channel to ensure that no other cluster head is currently transmitting data using the base station spreading code. If the channel is sensed busy, the cluster head delays the data transmission until the channel becomes idle. When this event occurs, the cluster head sends the data using the base station spreading code.

In general, schedule-based protocols are contention-free, and as such, they eliminate energy waste caused by collisions. Furthermore, sensor nodes need only turn their radios on during those slots where data are to be transmitted or received. In all other slots, the sensor node can turn off its radio, thereby avoiding overhearing. This results in low-duty-cycle node operations, which may extend the network lifetime significantly. Schedule-based MAC protocols have several disadvantages, however, which limit their use in WSNs. The use of TDMA requires the organization

of nodes into clusters. This hierarchical structure often restricts nodes to communicate only with their cluster head. Consequently, peer-to-peer communication cannot be supported directly, unless nodes are required to listen during all time slots. Most of the schedule-based schemes depend on distributed, fine-grained time synchronization to align slot boundaries. Achieving time synchronization among distributed sensor nodes is difficult and costly, especially in energy-constrained wireless networks. Schedule-based schemes also require additional mechanisms such as FDMA or CDMA to overcome intercluster communications and interference. Finally, TDMA-based MAC-layer protocols have limited scalability and are not easily adaptable to node mobility and changes in network traffic and topology. As nodes join or leave a cluster, the frame length as well as the slot assignment must be adjusted. Frequent changes may be expensive or slow to take effect.

### 5.4.2 Random Access-Based Protocols

Traditional random access MAC-layer protocols, also known as contention-based protocols, require no coordination among the nodes accessing the channel. Colliding nodes back off for a random duration of time before again attempting to access the channel. These protocols, however, are not well suited for WSN environments. The enhancement of these protocols with collision avoidance and request-to-send (RTS) and clear-to-send (CTS) mechanisms improves their performance and makes them more robust to the hidden terminal problem [5.34]. The energy efficiency of contention-based MAC-layer protocols, however, remains low due to collisions, idle listening, overhearing, and excessive control overhead. To address this shortcoming, efforts in the design of random access MAC-layer protocols focused on reducing energy waste in order to extend the network lifetime.

The power aware multiaccess protocol with signaling (PAMAS) avoids overhearing among neighboring nodes by using a separate signaling channel [5.4,5.35]. The protocol combines the use of a busy tone with RTS and CTS packets to allow nodes currently not actively transmitting or receiving packets to turn off their radio transceivers. The protocol does not, however, provide mechanisms to reduce energy waste caused by idle listening.

The sparse topology and energy management (STEM) protocol trades latency for energy efficiency [5.36]. This is achieved using two radio channels: a data radio channel and a wake-up radio channel. A variant of STEM uses a busy tone instead of encoded data for the wake-up signal. STEM is known as a pseudoasynchronous scheduled scheme. Based on this scheme, a node turns off its data radio channel until communication with another node is desired. When a node has data to transmit, it begins transmitting on the wake-up radio channel. The wake-up signal channel acts like a paging signal. The transmission of this signal lasts long enough to ensure that all neighboring nodes are paged. When a node is awakened from its sleeping mode, it may remain awake long enough to receive a ''session'' of packets. A node can also be awakened to receive all of its pending packets before going into the sleep mode again. The STEM protocol is general and can be used in conjunction with other MAC-layer scheduling protocols. The scheme is, however, effective only

in network environments where events do not happen very frequently. If events occur frequently, the energy wasted by continuously transmitting wake-up signals may offset, or may exceed, the energy gained in sleeping modes.

A variety of IEEE 802.11-inspired contention-based protocols prevent overhearing by using RTS and CTS packets [5.37–5.40]. A common feature of these protocols is to use the overhearing of RTS and CTS packet exchange between two other contending nodes to force a contending node to go into sleep mode. These protocols also rely on synchronized schedules between neighboring nodes to avoid idle listening. These protocols differ in the way they maintain low duty cycles and the way they achieve energy efficiency, especially when the size of the data packets is of the same order of magnitude as the size of the RTS and CTS packets. They also differ in the mechanisms used to reduce packet latency, as a sending node may have to wait a significant period of time before the receiver wakes up. Finally, the protocols also differ in the level and the way in which they achieve fairness among nodes.

The timeout-MAC (T-MAC) is a contention-based MAC-layer protocol designed for applications characterized by low message rate and low sensitivity to latency [5.5]. To avoid collision and ensure reliable transmission, T-MAC nodes use RTS, CTS, and acknowledgment packets to communicate with each other. Furthermore, the protocol uses an adaptive duty cycle to reduce energy consumption and adapt to traffic load variations. The basic idea of the T-MAC protocol is to reduce idle listening by transmitting all messages in bursts of variable length. Nodes are allowed to sleep between bursts. Furthermore, the protocol dynamically determines the optimal length of the active time, based on current load. Since messages between active times must be buffered, the buffer capacity determines an upper bound on the maximum frame time.

Based on the T-MAC protocol, nodes alternate between sleep and wake-up modes. Each node wakes up periodically to communicate with its neighbors. A node keeps listening and potentially transmitting as long as it is in the active period. An active period ends when no active event occurs for a predetermined time interval. Active events include the hearing of a periodic frame timer, the reception of data over the radio, the sensing of an activity such as collision on the channel, the end of transmission of a node's own data packet or acknowledgment, and the end of a neighboring node's data exchange, determined through overhearing of prior RTS and CTS packets. At the end of the active period, the node goes into sleep mode.

The Berkeley media access control (B-MAC) is a lower-power carrier-sense media access protocol for WSNs [5.41–5.43]. In contrast to traditional IEEE 802.11-inspired MAC-layer protocols, which include mechanisms for network organization and clustering, the B-MAC protocol embodies a small core of media access functionality. B-MAC uses clear channel assessment (CCA) and packet back-offs for channel arbitration, link-layer acknowledgments for reliability, and listening for low-power communication. B-MAC does not provide direct support for multipacket mechanisms to address the hidden terminal problem, handle message fragmentation, or enforce a particular low-power policy. However, in addition to the standard message interface, provides, B-MAC, a set of interfaces that allow

services to tune its operation. By exposing a set of configurable mechanisms, protocols built on B-MAC make local policy decisions to optimize power consumption, latency, throughput, fairness, or reliability.

To achieve low-power operation, B-MAC employs an adaptive preamble sampling scheme to reduce duty cycle and minimize idle listening. Each time the node wakes up, it turns on the radio and checks for activity. If it detects activity, the node powers its radio transceiver up and stays awake for the time required to receive the incoming packet. After reception, the node returns to sleep. If no packet is received within the specified timeout, the node goes to sleep. B-MAC supports on-the-fly reconfiguration and provides bidirectional interfaces for system services to optimize performance, whether it is for throughput, latency, or power conservation.

In the remaining sections of this chapter, we first discuss in detail a common IEEE 802.11-inspired protocol MAC-layer protocol, referred to as S-MAC. We then describe the basic architecture and protocols of the IEEE 802.15.4 wireless MAC- and physical-layer specifications for low-data-rate wireless personal area networks.

## 5.5 SENSOR-MAC CASE STUDY

The sensor-MAC (S-MAC) protocol is designed explicitly to reduce energy waste caused by collision, idle listening, control overhead, and overhearing [5.44–5.46]. The goal is to increase energy efficiency while achieving a high level of stability and scalability. In exchange, the protocol incurs some performance reduction in per-hop fairness, and latency S-MAC uses multiple techniques to reduce energy consumption, control overhead, and latency, in order to improve application-level performance. In the following we provide an overview of the S-MAC-layer protocol and discuss the techniques it proposes to achieve energy efficiency while keeping latency low.

### 5.5.1 Protocol Overview

The protocol design assumes a large number of sensor nodes, with limited storage, communication, and processing capabilities. The nodes are configured in an ad hoc, self-organized, and self-managed wireless network. Data generated by sensors are processed and communicated in a store-and-forward manner. The applications supported by the network are assumed to alternate between long idle periods, during which no events occur, and bursty active periods, during which data flow toward the base station through message exchange among peer sensor nodes. Furthermore, the applications are assumed to tolerate increased latency for an extended network lifetime. Typical applications that fall into this category include surveillance and monitoring of natural habitats and protection of critical infrastructure. In these applications the sensors must be vigilant over long periods of time, during which they remain inactive until some event occurs. The frequency at which these events occur is typically orders of magnitude slower than the time it takes to transmit a message across the network toward the base station.

S-MAC exploits the bursty profile of sensor applications to establish low-duty-cycle operation on nodes in a multihop network and to achieve significant energy savings. During the long periods of time during which no sensing occurs, S-MAC nodes alternate periodically between listening and sleep modes. Each node sets a wakeup time and sleeps for a certain period of time, during which its radio is turned off. At the expiration of the timer, the node becomes active again. To further reduce control overhead while keeping message latency low, the protocol uses coordinated sleeping among neighboring nodes. Periodic sleeping reduces energy consumption at the expense of increased latency. The importance of message latency strongly depends on the requirements of the sensing application. S-MAC focuses on applications that can tolerate latency on the order of seconds. However, when nodes follow their schedule strictly, latency can increase significantly. To address this shortcoming and keep message delay within the targeted-second-level latency, S-MAC uses adaptive listening.

As stated previously, S-MAC design is focused on cooperating applications, such as monitoring and surveillance applications. The applications cooperate to achieve a common single task, such as protecting a critical infrastructure. The nature of these applications is such that at any particular point in time, one sensor node may have a large amount of information to communicate to its neighbors. To accommodate this requirement while further reducing overhead, S-MAC sacrifices channel access fairness and uses the concept of message passing, whereby a node is allowed to send a long message in burst. Message passing reduces control overheard and avoids overhearing.

### 5.5.2 Periodic Listen and Sleep Operations

One of the S-MAC design objectives is to reduce energy consumption by avoiding idle listening. This is achieved by establishing low-duty-cycle operations for sensor nodes. Periodically, nodes move into a sleep state during which their radios are turned off completely. Nodes become active when there is traffic in the network. The basic periodic listen and sleep scheme is depicted in Figure 5.7. Based on this scheme, each node sets a wake-up timer and goes to sleep for the specified period of time. At the expiration of the timer, the node wakes up and listens to determine if it needs to communicate with other nodes. The complete listen- and-sleep cycle is referred to as a *frame*. Each frame is characterized by its *duty cycle*, defined as the listening interval-to-frame length ratio. Although the length of the listening interval can be selected independently by sensor nodes, for simplicity the protocol assumes the value to be the same for all nodes.

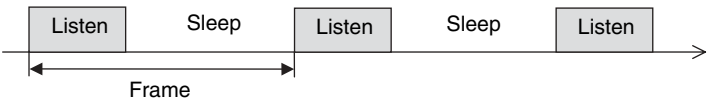



**Figure 5.7** S-MAC period listen and sleep modes of operations [5.44].

Nodes are free to schedule their own sleep and listen intervals. It is preferable, however, that the schedules of neighboring nodes be coordinated in order to reduce the control overhead necessary to achieve communications between these nodes. Contrary to other protocols in which coordination is achieved through a master node such as a cluster head, S-MAC nodes form virtual clusters around schedules but communicate directly with their peers to exchange and synchronize their sleep and listen schedules.

### 5.5.3 Schedule Selection and Coordination

The neighboring nodes coordinate their listen and sleep schedules such that they all listen at the same time and all sleep at the same time. To coordinate their sleeping and listening, each node selects a schedule and exchanges it with it neighbors during the synchronization period. Each node maintains a schedule table that contains the schedule of all its known neighbors.

To select a schedule, a node first listens to the channel for a fixed amount of time, at least equal to the synchronization period. At the expiration of this waiting period, if the node does not hear a schedule from another node, it immediately chooses its own schedule. The node announces the schedule selected by broadcasting a SYNC packet to all its neighbors. It is worth noting that the node must first perform physical carrier sensing before broadcasting the SYNC packet. This reduces the likelihood of SYNC packet collisions among competing nodes. If during the synchronization period the node receives a schedule from a neighbor before choosing and announcing its own schedule, the node sets its schedule to be the same as the schedule received. The node waits until the next synchronization period to announce the schedule to its neighboring nodes.

It is worth noting that a node may receive a different schedule after it chooses and announces its own schedule. This may occur if the SYNC packet is corrupted by either collision or channel interference. If the node has no neighbor with whom it shares a schedule, the node simply discards its own schedule and adopts the new one. On the other hand, if the node is aware of other neighboring nodes that have already adopted its schedule, the node adopts both schedules. The node is then required to wake up at the listen intervals of the two schedules adopted. This is illustrated in Figure 5.8. The advantage of carrying multiple schedules is that border nodes are required to broadcast only one SYNC packet. The disadvantage of this approach is that border nodes consume more energy, as they spend less time in the sleep mode.

It is to be noted that neighboring nodes may still fail to discover each other, due to the delay or loss of a SYNC packet. To address this shortcoming, S-MAC nodes are required to perform frequent neighbor discovery, whereby a node listens periodically to the entire synchronization period. Nodes that currently do not have any neighbors are expected to perform neighbor discovery more frequently.

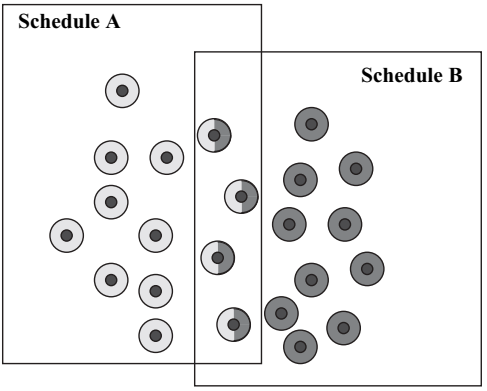



**Figure 5.8** Border node schedule selection and synchronization.

### 5.5.4 Schedule Synchronization

Neighboring nodes need to synchronize their schedules periodically to prevent long-term clock drift. Schedule updating is accomplished by sending a SYNC packet. For a node to receive both SYNC packets and data packets, the listen interval is divided into two subintervals as depicted in Figure 5.9. This figure illustrates three cases. In the first case the sender sends only a SYNC packet; in the second the sender sends only a data packet; and in the third the sender sends a SYNC packet in addition to the data packet.

Access to the channel by contending nodes during these subintervals is regulated using a multislotted contention window. The first subinterval is dedicated to the transmission of SYNC packets; the second subinterval is used for the transmission of data packets. In either of these subintervals, a contending station randomly selects a time slot, performs carrier sensing, and starts sending its packet if it detects that the channel is idle. Transmission of data packets uses the RTS/CTS handshake

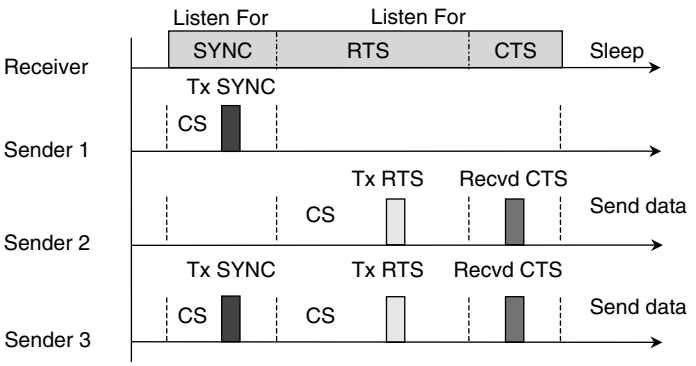



**Figure 5.9** Timing relationship between a receiver and a variety of senders [5.44].

to secure exclusive access to the channel during transmission of the data. This access procedure guarantees that the neighboring nodes receive both the synchronization and data packets.

### 5.5.5 Adaptive Listening

A closer look at the periodic listen and sleep scheme reveals that a message may incur increased latency as it is stored and forwarded between adjacent network nodes. If a sensor is to follow its sleep schedule strictly, data packets may be delayed at each hop. To address this shortcoming and improve latency performance, the protocol uses an aggressive technique referred to as *adaptive listening*. Based on this technique, a node that overhears, during its listen period, the exchange of a CTS or RTS packet between a neighboring node and another node assumes that it may be the next hop along the routing path of the overheard RTS/CTS packet, ignores its own wake-up schedule, and schedules an extra listening period around the time the transmission of the packet terminates. The overhearing node determines the time necessary to complete the transmission of the packet from the duration field of the overheard CTS or RTS packet. Immediately upon receiving the data packet, the node issues an RTS packet to initiate an RTS/CTS handshake with the overhearing node. Ideally, the latter node is awake, in which case the packet forwarding process proceeds immediately between the two nodes. If the overhearing node does not receive an RTS packet during adaptive listening, it reenters its sleep state until the next scheduled listen interval.

### 5.5.6 Access Control and Data Exchange

To regulate access to the communication channel among contending sensor nodes, S-MAC uses a CSMA/CA-based procedure, including physical and virtual carrier sensing and the use of RTS/CTS handshake to reduce the impact of the hidden and exposed terminal problems. Virtual carrier sensing is achieved through use of the network allocation vector (NAV), a variable whose value contains the remaining time until the end of the current packet transmission. Initially, the NAV value is set to the value carried in the duration field of the packet transmitted. The value is decremented as time passes and eventually reaches zero. A node cannot initiate its own transmission until the NAV value reaches zero. Physical carrier sensing is performed by listening to the channel to detect ongoing transmission. Carrier sensing is randomized within a contention window to avoid collisions and starvation. A node is allowed to transmit if both virtual and physical carrier sensing indicate that the channel is free.

To perform virtual carrier sensing effectively, nodes may be required to listen to all transmissions from their neighbors. As a result, nodes may be required to listen to packets that are destined for other nodes. Packet overhearing may lead to significant energy waste. To avoid overhearing, S-MAC allows nodes to move into sleep mode after they hear the exchange of an RTS or a CTS packet between two other nodes. The node initializes its NAV with the value contained in the duration field of

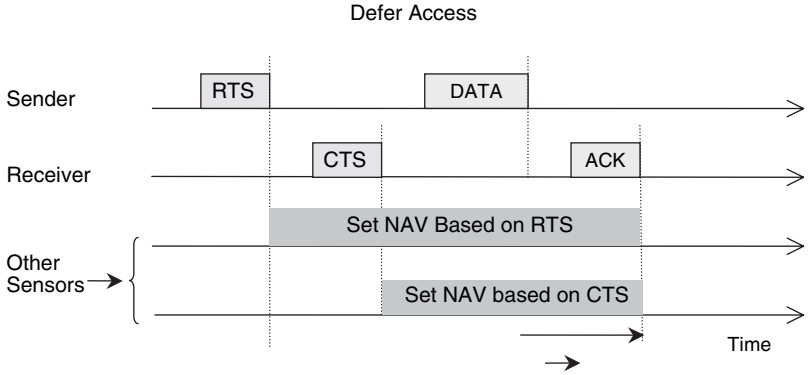



**Figure 5.10** S-MAC collision avoidance scheme [5.46].

the RTS or CTS packets and enters the sleep state until the NAV value reaches zero. Since data packets are typically larger than control packets, the overhearing avoidance process may lead to significant energy savings. The scheme used by S-MAC to avoid collisions is illustrated in Figure 5.10.

A node attempting to transmit a message must first sense the channel. If the channel is busy, the node goes to sleep and wakes up when the channel becomes free again. If the channel is idle, a node, sending a data packet, first issues an RTS packet and waits for a CTS packet from the receiver. When it receives the CTS packet, the node sends its data packet. The transaction is completed when the node receives an acknowledgment from the receiver. It is worth noting that after successful exchange of the RTS and CTS packets, the communicating nodes use their normal sleep time to exchange data packets. The nodes do not resume their regular sleep schedule until the data transmission is completed. Furthermore, the transmission of a broadcast packet, such as a SYNC packet, does not require the exchange of the RTS and CTS packet.

### 5.5.7 Message Passing

To improve application-level performance, S-MAC introduces the concept of *message passing*, where a message is a meaningful unit of data that a node can process. Messages are divided into small fragments. These fragments are then transmitted in a single burst. The fragments of a message are transmitted using only one RTS/CTS exchange between the sending and receiving nodes. At the completion of this exchange, the medium is reserved for the time necessary to complete the transfer of the entire message successfully. Furthermore, each fragment carries in its duration field the time needed to transmit all the subsequent fragments and their corresponding acknowledgments. This procedure is depicted in Figure 5.11.

Upon transmitting a fragment, the sender waits for an acknowledgment from the receiver. If it receives the acknowledgment, the sender proceeds with transmission

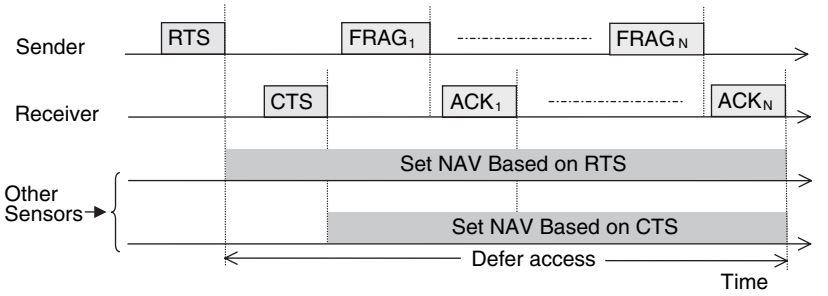



**Figure 5.11** S-MAC message passing [5.46].

of the next fragment. If it fails to receive the acknowledgment, however, the sender extends the time required to complete transmission of the segment to include the time to transmit one more fragment and its corresponding acknowledgment and retransmits the unacknowledged frame immediately. It is worth noting that sleeping nodes can hear about this extension only if they hear extended fragments or their corresponding acknowledgments. Nodes that only heard the initial RTS and CTS packet exchange remain unaware of the transmission extension. The S-MAC has the potential to achieve significant energy savings. It is well suited for applications where fairness is not a critical design goal and increased latency is tolerable.

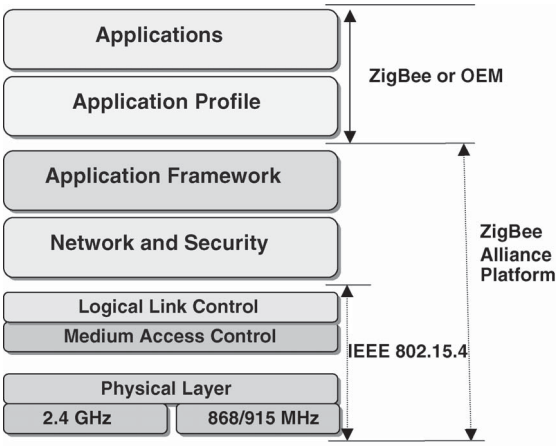
## 5.6 IEEE 802.15.4 LR-WPANs STANDARD CASE STUDY

The IEEE 802.15.4 specification complements the IEEE 802 set of wireless standards [5.47]. The main design objective of the IEEE 802.15.4 open standard is to support the wireless connectivity of a vast number of industrial, home, and medical applications, including automotive monitoring and control, home automation, ubiquitous and pervasive health care, gaming, and sensor-rich environments. Such applications require a small, low-cost, highly reliable technology that offers long battery life, measured in months or even years, and automatic or semiautomatic installation. The IEEE 802.15.4 standard supports these requirements by trading off higher speed and performance for architectures that benefit from low power consumption and low cost, as noted in Chapter 4.

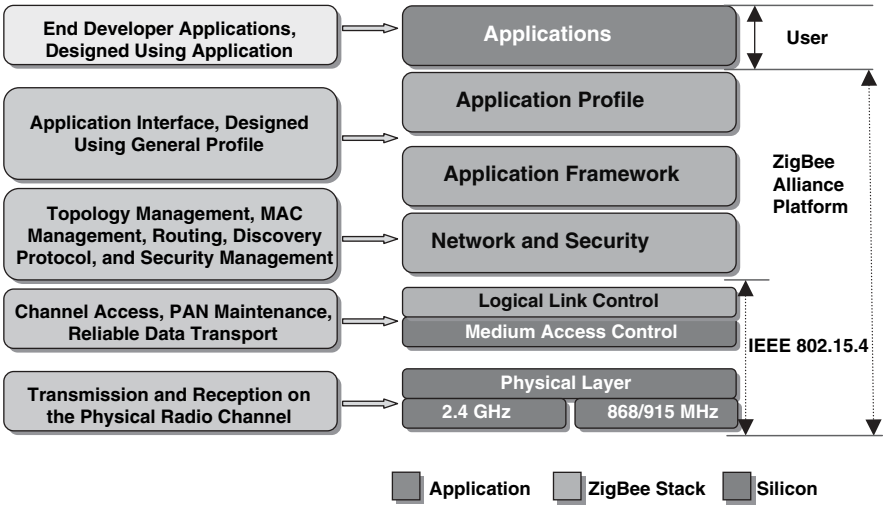
The IEEE 802.15.4 standard has been adopted by the ZigBee Alliance for wireless personal area network technology [5.48]. The alliance is an association of hundreds of members from around the world, working together to enable the reliable and cost-effective networking of wireless devices for monitoring and control, based on an open global standard. The reference model, depicted in Figure 5.12(*a*), shows the various layers of the ZigBee wireless technology architecture the relationship of the IEEE 802.15.4 standard to the ZigBee alliance MAC layer protocol model. These layers facilitate the features that make ZigBee very attractive: low cost, very low power consumption, reliable data transfer, and easy implementation. Using the IEEE 802.15.4 specifications, the alliance focuses on the design issues

related to the network, security, and applications layers. It also provides specification for interoperability and testing. Figure 5.12(*b*) shows the ZigBee stack reference model and lists the basic functionalities at each layer.

The physical layer (PHY) of the reference model specifies the network interface components, their parameters, and their operation. Furthermore, to support operation of the MAC layer, the PHY layer includes a variety of features, such as receiver energy detection (RED), link quality indicator (LQI), and clear channel assessment




(*a*)




(*b*)

**Figure 5.12** (*a*) IEEE 802.15.4 and ZigBee reference model; (*b*) ZigBee stack reference model.

**TABLE 5.1 IEEE 80215.4 PHY Layer Main Parameters**

| Parameter | 2.4-GHz PHY | 868/915-MHz PHY |
|---|---|---|
| Sensitivity @ 1% PER | −85 dBm | −92 dBm |
| Receiver maximum input level | −20 dBm | |
| Adjacent channel rejection | 0 dB | |
| Alternate channel rejection | 30 dB | |
| Output power, lowest maximum | −3 dBm | |
| Transmission modulation accuracy | EMV < 35% for 1000 chips | |
| Number of channels | 16 | 1/10 |
| Channel spacing | 5 MHz | NA[a]/2 MHz |
| Transmission rates | | |
| Date rate | 250 kbps | 20/40 kbps |
| Symbol rate | 62.5 kilosymbols/sec | 20/40 kilosymbols/sec |
| Chip rate | 2 megachips/sec | 300/60 kilochips/sec |
| Chip modulation | O-QPSK with half-sine pulse shaping (MKS) | BPSK with raised cosine pulse shaping |
| RX-TX and TX-RX turnaround time | 12 symbols | |

[a]Single channel.

(CCA). The PHY layer is also specified with a wide range of operational low-power features, including low-duty-cycle operations, strict power management, and low transmission overhead. These parameters are listed in Table 5.1.

The MAC layer handles network association and disassociation. It also regulates access to the medium. This is achieved through two modes of operation: beaconing and nonbeaconing. The beaconing mode is specified for environments where control and data forwarding is achieved by an always-active device. The nonbeaconing mode specifies the use of unslotted, nonpersistent CSMA-based MAC protocol.

The network layer provides the functionality required to support network configuration and device discovery, association and disassociation, topology management, MAC-layer management, routing, and security management. Three network topologies—star, mesh, and cluster tree—are supported.

The security layer leverages the basic security services specified by the IEEE 802.15.4 security model to provide support for infrastructure security and application data security. The first security service of the IEEE 802.15.4 security model provides support for access control. This basic security service prevents unauthorized parties from participating in the network. It allows a legitimate device to maintain a list of trusted devices in the network. A legitimate device uses this list to detect and reject messages from unauthorized devices. The second security service

supports message integrity protection to prevent an adversary from tampering with a data message from an authorized sender, while the message is in transit. The third security service provides data confidentiality to keep the information carried by a data message secret from unauthorized parties. This is achieved using the advanced encryption standard (AES) as its core cryptographic algorithm. The encryption scheme uses a 128-bit key to encrypt messages. The fourth security service deals with sequential data freshness to prevent replay attacks. An adversary engages in a replay attack by eavesdropping on legitimate messages sent between authorized users and replaying them at a later time.

Using the basic security services, the MAC layer describes a variety of security suites. Each suite offers a different set of security properties and guarantees. By default, security is not enabled. The application must therefore explicitly set the appropriate control parameters into the radio stack to enable the desired level of security.

The application layer consists of the application support sublayer (APS), the ZigBee device object (ZDO), and the manufacturer-defined application objects. The responsibilities of the APS sublayer include maintaining tables for binding devices together, based on their services and their needs, and forwarding messages between bound devices. The ZDO can be thought of as a special application object that is resident on all nodes. It has its own profile, referred to as the ZigBee device profile (ZDP), which user application endpoints and other ZigBee nodes can access. The ZDO is responsible for overall device management and security keys and policies, including defining the role of the device within the network, initiating and responding to binding requests, and establishing a secure relationship between network devices. The manufacturer-defined application objects implement the actual applications according to the ZigBee-defined application descriptions.

In the following section we focus on physical- and MAC-layer design issues, mechanisms, and protocols. First, the overall characteristics of the PHY layer are highlighted. Following, a description of the different components and operations of the MAC layer is provided.

### 5.6.1 PHY Layer

The design of the PHY layer is driven by the need for a low-cost power-effective physical layer for cost-sensitive low-data-rate monitoring and control applications. Under IEEE 802.15.4, wireless links can operate in three unlicensed frequency bands: 858 MHz, 902 to 928 MHz, and 2.4 GHz. Based on these frequency bands, the IEEE 802.15.4 standard defines three physical media:

1. Direct-sequence spread spectrum using BPSK operating in the 868-MHz band at a data rate of 20 kbps
2. Direct-sequence spread spectrum using BPSK operating in the 915-MHz band at a data rate of 40 kbps
3. Direct-sequence spread spectrum using O-QPSK operating in the 2.4-GHz band at a data rate of 140 kbps

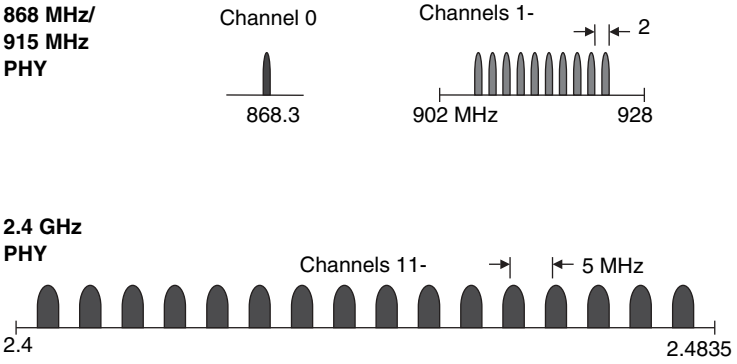



**Figure 5.13** IEEE 802.15.4 PHY-layer operating frequency bands.

These operating frequency bands are depicted in Figure 5.13. The spreading code of the 868- and 915-MHz PHY layers is a 15-chip m-sequence. Both specifications use BPSK with a differential encoding data modulation scheme. The data rate of 868-MHz layer is 20 kbps while the data rat of the 915 MHz specification is 40 kbps. The chip modulation used by both specifications is BPSK with raised cosine shaping ($a = 1.0$). The resulting chip rate is 300 kilochips/sec for the 868-MHz PHY layer and 600 kilochips/sec for the 915-MHz PHY layer.

The data modulation of the 2.4-GHz PHY layer is a 16-ary orthogonal modulation. Consequently, 16 symbols are an orthogonal set of 32-chip PN codes. The resulting data rate is 250 kbps (4 bits/symbol, 62.5 kilosymbols/sec). The specification uses O-QPSK with half-sine pulse shaping, which is equivalent to minimum shift keying. The resulting chip rate is 2.0 megachips/sec. The packet structure of the IEEE 802.15.4 PHY layer is depicted in Figure 5.14. The first field of this structure contains a 32-bit preamble. This field is used for symbol synchronization. The next field represents the start of a packet delimiter. This field of 8 bits is used for frame synchronization. The 8-bit PHY header field specifies the length of the PHY service data unit (PSDU). The PSDU field can carry up to 127 bytes of data.

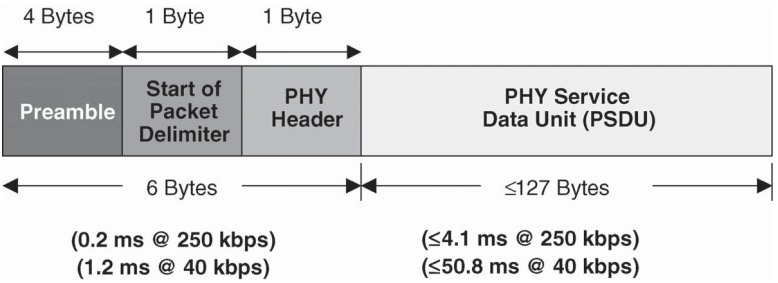



**Figure 5.14** IEEE 802.15.4 PHY-layer packet structure.

### 5.6.2 MAC Layer

The IEEE 802.15.4 MAC-layer specification is designed to support a vast number of industrial and home applications for control and monitoring. These applications typically require low to medium data rates and moderate average delay requirements with flexible delay guarantees. Furthermore, the complexity and implementation cost of the IEEE 802.15.4 standard compliant devices must be low to minimize energy consumption and enable the deployment of these devices on a large scale.

To address the needs of its intended applications while enabling the deployment of a large number of monitoring and control devices at a reduced implementation cost, the IEEE 802.15.4 MAC-layer specification embeds in its design several unique features for flexible network configurations and low-power operations. These features include:

- Support for various network topologies and network devices
- The availability of an optional superframe structure to control the network devices' duty cycle
- Support for direct and indirect data transmissions
- Contention- and schedule-based media access control methods
- Beaconed and nonbeaconed modes of operation (In the beacon mode, the protocol uses a superframe structure to coordinate access to the medium—both contention-based access and guaranteed time slots allocation are supported; in the nonbeaconed mode, the protocol uses an unslotted CSMA/CA-based access scheme.)
- Efficient energy management schemes for an extended battery life, including adaptive sleep for extended period of time over multiple beacons
- Flexible addressing scheme to support the deployment of large-scale networks, theoretically over 65,000 nodes per network

In the following, the classes of network devices supported by the IEEE 802.15.4 MAC standard and the network topologies that can be achieved using these devices are discussed. The optional superframe structure is then described and the two modes of operations, beaconed and nonbeaconed modes, are discussed. Depending on the mode of operations used, two MAC layer protocols are specified. The basic operations of these two MAC layer protocols, including the procedures that govern the data transfer between the network devices in each mode of operations, are described.

***Device Types and Network Topologies*** To accommodate the MAC protocol, the IEEE 802.15.4 standard distinguishes devices based on their hardware complexity and capability. Accordingly, the standard defines two classes of physical devices: a full-function device (FFD) and a reduced-function device (RFD). These device types differ in their use and how much of the standard they implement. An FFD

**TABLE 5.2 Device Types in ZigBee Networks**

| Physical Device Type | Logical Device Type | | |
|---|---|---|---|
| | Coordinator | Router | End Device |
| Full-function device | Yes | Yes | Yes |
| Reduced-function device | No | No | Yes |

is equipped with the adequate resources and memory capacity to handle all the functionalities and features specified by the standard. It can therefore assume multiple network responsibilities. It can also communicate with any other network device. An RFD is a simple device that carries a reduced set of functionalities, for lower cost and complexity. It typically contains a physical interface to the wireless modem and executes the specified IEEE 802.15.4 MAC layer protocol. Furthermore, it can only associate and communicate with an FFD.

Based on these physical device types, ZigBee defines a variety of logical device types. These logical devices are distinguished based on their physical capabilities and the role they play in the network deployed. Table 5.2 illustrates the possible combinations of device types that can be deployed in a ZigBee-enabled network.

There are three categories of logical devices:

1. *Network coordinator*: an FFD device responsible for network establishment and control. The coordinator is responsible for choosing key parameters of the network configuration and for starting the network. It also stores information about the network and acts as the repository for security keys.
2. *Router*: an FFD device that supports the data routing functionality, including acting as an intermediate device to link different components of the network and forwarding message between remote devices across multihop paths. A router can communicate with other routers and end devices.
3. *End devices*: an RFD device that contains just enough functionality to communicate with its parent node: the network coordinator or a router. An end device does not have the capability to relay data messages to other end devices.

Based on these logical devices types, a ZigBee wireless personal area network (PAN) can be organized in one of three possible topologies: a star, a mesh (peer-to-peer), or a cluster tree. The three network configurations are illustrated in Figure 5.15. The star network topology supports a single coordinator, with up to 65,536 devices. In this topology configuration, one of the FFD-type devices assumes the role of network coordinator. All other devices act as end devices. The coordinator selected is responsible for initiating and maintaining the end devices on the network. Upon initiation, the end devices can only communicate with the coordinator. The mesh configuration allows path formation from any source device to any destination device, using tree and table-driven routing algorithms. The table-driven routing algorithm employs a simplified version of the on-demand distance vector routing (AODV) and Internet Engineering Task Force

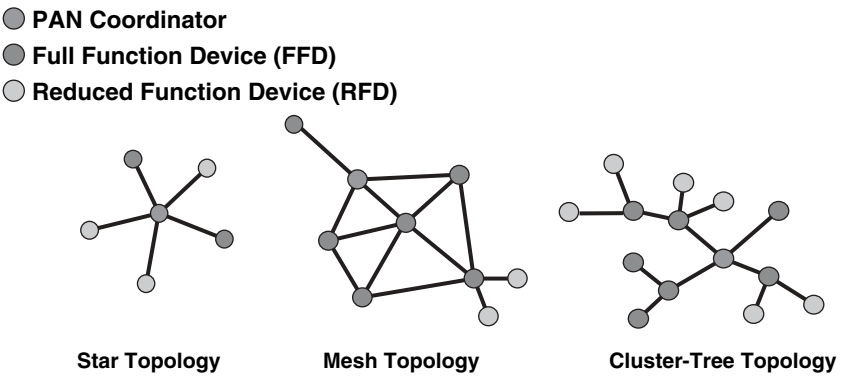



**Figure 5.15** Network topologies.

(IETF) proposal for mobile ad hoc networking (MANET). In the mesh topology, the radio receivers of the coordinator and the routers must always be on.

Cluster tree networks enable a peer-to-peer network to be formed with a minimum of routing overhead, using multihop routing. The topology is suitable for latency-tolerant applications. A cluster tree network is self-organized and supports network redundancy to achieve a high degree of fault resistance and self-repair. The cluster can be significantly large, comprising up to 255 clusters of up to 254 nodes each, for a total of 64,770 nodes. It may also span large physical areas. Any FFD can be a coordinator. Only one coordinator is selected for the PAN. The PAN coordinator forms the first cluster and assigns to it a cluster identity (CID) of value zero. Subsequent clusters are then formed with a designated cluster head for each cluster.

Each PAN is uniquely identified by a 16-bit identifier. A PAN coordinator is the designated principal controller of the WPAN. Every network has exactly one PAN coordinator, selected from within all the coordinators of the network. A coordinator is a network device configured to support network functionalities and additional responsibilities, including:

- Managing a list of all associated network devices
- Exchanging data frames with network devices and a peer coordinator
- Allocating 16-bit short addresses to network devices (The short addresses, assigned on demand, are used by the associated devices in lieu of the 64-bit addresses for subsequent communications with the coordinator.)
- Generating beacon frames on a periodic basis (These frames are used to announce the PAN identifier, the list of outstanding frames, and other network and device parameters.)

***Superframe Structure*** The IEEE 802.15.4 MAC standard defines an optional superframe structure. It is initiated by the PAN coordinator. Furthermore, its format is decided by the coordinator. As Figure 5.16(*a*) shows, the superframe is bounded

by network beacons and divided into 16 equally sized slots. The first time slot of each superframe is used to transmit the beacon. The main purpose of the beacon is to synchronize the attached devices, identify the PAN, and describe the superframe structure. The remaining time slots are used by competing devices for communications during the contention access period (CAP). The devices use a slotted CSMA-CA-based protocol to gain access to compete for the time slots. All communications between devices must complete by the end of the current CAP and the beginning of the next network beacon.

To satisfy the latency and bandwidth requirements of the supported applications, the PAN coordinator may dedicate groups of contiguous time slots of the active superframe to these applications. These slots are labeled as guaranteed time slots (GTSs). The number of GTSs cannot exceed seven. A single GTS allocation, however, may occupy more then one time slot. Together the GTSs form the contention-free period (CFP). As shown in Figure 5.16(*b*), the CFP always appears at the end of the active superframe and starts at a slot boundary immediately following the CAP. The CAP time slots remain for contention-based access between networked devices and new devices wishing to join the network. All communication transactions using contention-based access and GTS-based access must complete before the end their associated CAP and CFP, respectively.

Network devices, which need GTS allocation, can send requests during the CAP period to reserve a desired number of contiguous time slots. The requested slots can be of either the ''receive'' or the ''transmit'' type. The receive slots are used by the device to fetch data from the coordinator, while the transmit slots are used to send data to the coordinator. Devices that have no data to exchange with the coordinator can switch off their power and go into a sleep mode. Devices are expected to remain active, however, during their allocated GTSs. Devices are allowed to go into a sleep mode during the rest of the GTSs.

To reduce energy consumption, the coordinator may also issue a superframe containing both an active period and an idle period, as shown in Figure 5.16(*c*). The active period, composed of 16 time slots, contains the frame beacon, the CAP time slots, and if applicable, the CAP slots. The inactive period defines a time period during which all network nodes, including the coordinator, can go into a sleep mode. In this mode, the network devices switch off their power and set a timer to wake up immediately before the announcement of the next beacon frame.

It is worth noting that to accommodate a wide range of application requirements and network deployment, the length of the active and inactive periods, the time slot duration, and the number and usage of the slots designated as GTSs are configurable network parameters. Consequently, depending on the network activity, the types of devices connected to the network, and the nature of the application supported by the network, the length of the inactive period varies and may be set to zero.

***Frame Types*** The general MAC frame format of the IEEE 802.15.4 MAC-layer standard is depicted in Figure 5.17(*a*). It is composed of three basic components: the MAC header, the MAC payload, and the MAC footer. The MAC header

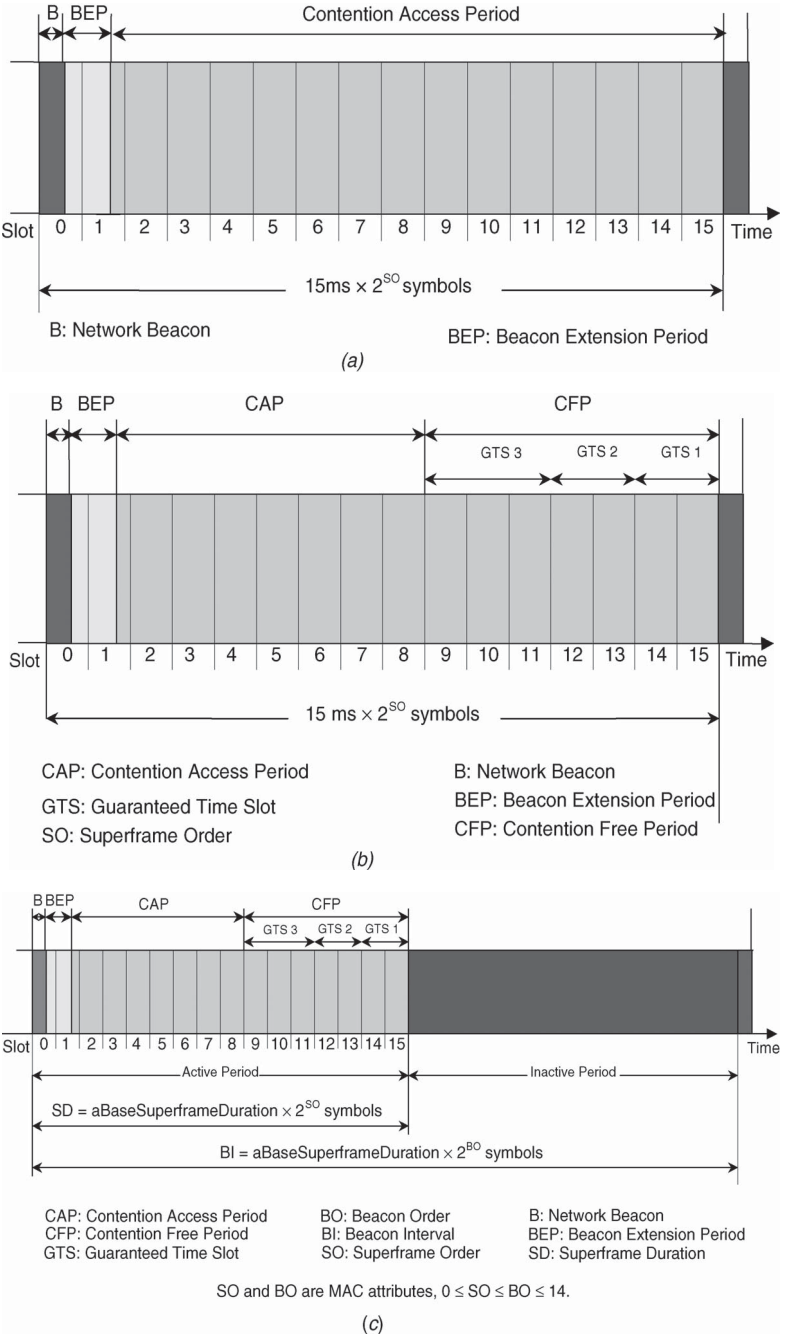
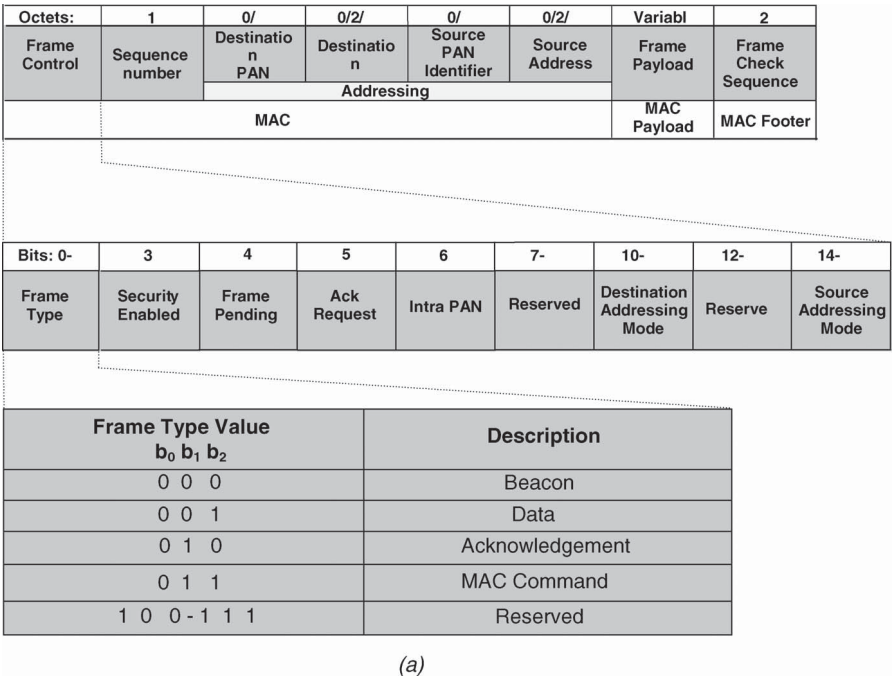



**Figure 5.16** (*a*) Superframe structure; (*b*) QoS frame structure; (*c*) Superframe structure with energy saving.



| Octets: | 1 | 0/ | 0/2/ | 0/ | 0/2/ | Variabl | 2 |
|---|---|---|---|---|---|---|---|
| Frame Control | Sequence number | Destination PAN | Destination | Source PAN Identifier | Source Address | Frame Payload | Frame Check Sequence |
| | | Addressing | | | | | |
| MAC | | | | | | MAC Payload | MAC Footer |

| Bits: 0- | 3 | 4 | 5 | 6 | 7- | 10- | 12- | 14- |
|---|---|---|---|---|---|---|---|---|
| Frame Type | Security Enabled | Frame Pending | Ack Request | Intra PAN | Reserved | Destination Addressing Mode | Reserve | Source Addressing Mode |

| Frame Type Value $b_0\ b_1\ b_2$ | Description |
|---|---|
| 0 0 0 | Beacon |
| 0 0 1 | Data |
| 0 1 0 | Acknowledgement |
| 0 1 1 | MAC Command |
| 1 0 0 - 1 1 1 | Reserved |

(a)

| Octets:2 | 1 | 4 or 10 | 2 | Variable | Variable | Variable | 2 |
|---|---|---|---|---|---|---|---|
| Frame Control | Beacon Sequence Number | Source Address | Superframe Spec | GTS Fields | Pending Address Fields | Beacon Payload | Frame Check Sequence |
| MAC Header | | | MAC Payload | | | | MAC Footer |

Superframe Specification

| Bits: 0-3 | 4-7 | 8-11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|
| Beacon Order | Superframe Order | Final CAP Slot | Battery Life Extension | Reserved | PAN Coordinator | Association Permit |

GTS Fields

| Octets: 1 | 0/1 | Variable |
|---|---|---|
| GTS Spec | GTS Directions | GTS List |

Pending Address Fields

| Octets: 1 | Variable |
|---|---|
| Pending Address Spec | Address List |

GTS Specification

| Bits: 0-2 | 3-6 | 7 |
|---|---|---|
| GTS Descriptor Count | Reserved | GTS Permit |

GTS Directions

| Bits: 0-6 | 7 |
|---|---|
| GTS Directions Mask | Reserved |

GTS List

| Bits: 0-15 | 16-19 | 20-23 |
|---|---|---|
| Device Short Address | GTS Starting Slot | GTS Length |

(b)

**Figure 5.17** (*a*) General MAC frame format; (*b*) Beacon frame format; (*c*) Data and acknowledgment frame format; (*d*) MAC command frame format. (*Continued*)

**Data Frame Format**

| Octets:2 | 1 | 4 to 20 | variable | 2 |
|---|---|---|---|---|
| Frame Control | Data Sequence Number | Address Information | Data | Frame Check Sequence |
| MAC | | | MAC Payload | MAC Foote |

**Acknowledgement Frame Format**

| Octets: | 1 | 2 |
|---|---|---|
| Frame Control | Data Sequence Number | Frame Check Sequence |
| MAC | | MAC Foote |

*(c)*

| Octets: | 1 | 4 to | 1 | variabl | 2 |
|---|---|---|---|---|---|
| Frame Control | Data Sequence | Address Information | Command | Command | Frame Check Sequence |
| MAC | | | MAC | | MAC foote |

Command Frame Types

| Command Frame Identifier | Command Name | RFD | |
|---|---|---|---|
| | | Tx | Rx |
| 0 x 01 | Association Request | X | |
| 0 x 02 | Association Response | | X |
| 0 x 03 | Dis-association Notification | X | X |
| 0 x 04 | Data Request | X | |
| 0 x 05 | PAN ID Conflict Notification | X | |
| 0 x 06 | Orphan Notification | X | |
| 0 x 07 | Beacon Request | | |
| 0 x 08 | Coordinator Realignment | | X |
| 0 x 09 | GTS Request | | |
| 0 x 0a – 0 | Reserved | | |

(*d*)

**Figure 5.17** (*continued*)

contains a frame control field and the addressing field. The control field carries the frame type and other information necessary for network control and operation. The addressing specifies the source PAN identifier, the source node address, the destination PAN identifier, and the destination address. The MAC payload contains the data frame to be exchanged between the communicating devices. The MAC footer contains the frame check sequence field. This field is used to detect frame errors.

The IEEE 802.15.4 defines four basic frame types: the beacon frame, the data frame, the acknowledgment frame, and the MAC command frame. The *beacon frame* is transmitted periodically by the coordinator. The frame serves multiple purposes, including identifying the network and its structure, waking up devices from the sleep mode to the listening mode, and synchronizing network operations. The beacon frames are particularly important in mesh and cluster-tree network topologies. They keep all the network nodes synchronized without requiring these nodes to remain awake for long period of times, thereby reducing considerably energy consumption and extending battery lifetime. The beacon frame and its fields are described in Figure 5.17(*b*).

The *data frame* carries a payload of up to 104 octets. Each data frame carries a sequence number that identifies the frame uniquely. The sequence number ensures that all frames are accounted for and are received in order. The FCS field is used to detect frames in error.

The *acknowledgment frame* is used by a receiver to acknowledge the receipt of a data frame. The receipt of the acknowledgment by the sender constitutes a confirmation that the corresponding data frame was received without error and in order. A data frame and its corresponding acknowledgment frame are matched by their respective sequence numbers. The data and acknowledgment frame formats are depicted in Figure 5.17(*c*).

The *MAC command frame* is used by the MAC entities in different devices for negotiation and communication. The frame provides the mechanism for a centralized network manager to control and configure devices remotely, irrespective of the network size and topology. Typical commands include device association and disassociation request and notification, data request, PAN ID conflict notification, orphan notification, beacon request, GTS request, and coordinator realignment. The MAC command frame format is shown in Figure 5.17(*d*). Upon receiving a frame, the MAC-layer entity must process the received frame to determine the actions required to handle the frame properly. To provide enough time for the MAC-layer entity to process the frame, the IEEE 802.15.4 MAC-layer standard requires that an interframe spacing (IFS) be inserted between two consecutive frames. The IFS duration depends on whether the transmission transaction is acknowledged or unacknowledged.

If the transmission is acknowledged, the IFS follows the acknowledgment frame. Furthermore, if the frame length does not exceed the threshold, aMaxSIFSFrameSize, the acknowledgment must be followed by a short IFS (SIFS) period, the duration of which should be at least aMinSIFSPeriod. If the frame length exceeds aMaxSIFSFrameSize, the acknowledgment must be followed by a long IFS (LIFS). The duration

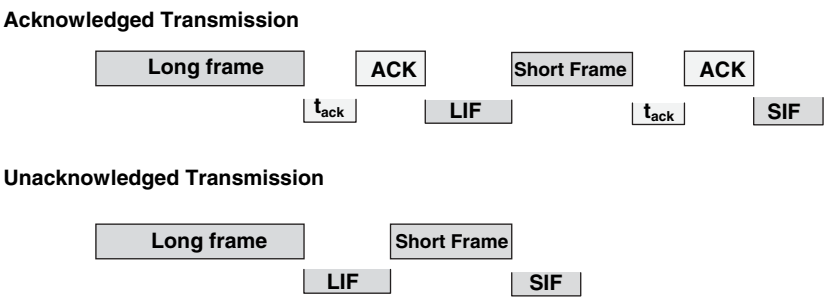



**Figure 5.18** Interframe spacing.

of the LIFS must be at least aMaxLIFSPeriod. The a MinSIFSPeriod is typically 12 symbols long, while the aMaxLIFSPeriod is 40 symbols long.

If the transmission is unacknowledged, the IFS immediately follows the data frame. Depending on whether the size of the frame exceeds or does not exceed the aMaxSIFSFrameSize threshold, a LIFS or a SIFS is used. The IFS procedure is depicted in Figure 5.18. As shown in this figure, an acknowledgment of the transmitted frame is expected to be received within a time interval, $t_{\text{ack}}$, such that aTurnaroundTime $\leq t_{\text{ack}} \leq$ (aTurnaroundTime + aUnitBackoff Period); aTurnaroundTime is typically 12 symbols long, while the aUnitBackoff Period is 20 symbols long.

***Modes of Operation*** The IEEE 802.15.4 MAC protocol is designed to meet the requirements of multiple types of traffic. Each traffic type is characterized by its unique characteristics in terms of the data profile and latency requirement. Based on this characterization, the IEEE 802.15.4 standard identifies three types of traffic: periodic data, intermittent data, and repetitive low-latency data. *Periodic data* characterize wireless sensor applications, whereby the sensor alternates between two modes of operations, active and idle. In the active mode, the sensor wakes up and exchanges data with a coordinator or another network device. In the idle mode, the sensor goes to sleep. *Intermittent data* are defined by an external stimulus or by an application such as a wireless light switch controlling a lamp. In this example, the lamp, typically mains powered, can monitor the channel in a continuous manner. On the other hand, the switch remains idle until it is toggled, in which case it transmits the information to the lamp. *Repetitive low-latency data* are defined by critical applications such as security monitoring systems. This type of application requires time slot allocations to guarantee access to the channel within its latency tolerance. To accommodate the three types of traffic, the IEEE 802.15.4 MAC-layer standard specifies a beaconed and a beaconless mode of operation. In the following we discuss the basic operations of these two modes.

*Beacon Mode Operation* The beacon mode allows devices within an extended network, such as mesh or cluster tree, to synchronize their actions and coordinate

communications with each other. Devices wake up only when a beacon is broadcast. A device registers with the network coordinator and looks for any messages addressed to it. If no messages are pending, the device returns to sleep. The coordinator may also go to sleep when the communications with the end devices are completed.

To regulate access to the channel, the network coordinator uses a superframe structure. As discussed above, the superframe is divided into 16 equally sized slots, the first of which is dedicated to the transmission of the beacon frame. Network devices can compete to access the channel during the contention access period (CAP), using a slotted CSMA-CA mechanism. Applications requiring low latency or a specific data rate may issue a request to the PAN coordinator for the allocation of GTSs during the contention-free period (CFP). The allocation of these GTSs are such that all contention-based transactions complete before the start of the CAP, and all GTS-based transactions finish within their allocated time slots and before the end of the CFP. In the following we first discuss the GTS allocation. We then describe the CSMA-CA access mechanism used by network devices to compete for the channel during the CAP.

*GTS Allocation* To reserve GTS for data exchange, a device sends an explicit request to the network coordinator. The request specifies the type, transmit or receive, and the number of the contiguous slots desired. A transmit slot is used by the device to send data to the coordinator. A request of a receive slot, on the other hand, expresses the readiness of the device to receive data from the coordinator. Immediately upon receiving a GTS request, the coordinator acknowledges the reservation frame but the acknowledgment constitutes neither a confirmation nor a denial of the reservation request. Upon receiving the acknowledgment, the device sets a timer to a specific value referred to as a GTSDescPersistenceTime and monitors transmission of the subsequent coordinator's beacon.

Depending on slot availability, three scenarios are possible. In the first scenario, the coordinator responds favorably to the request within the GTSDescPersistenceTime interval and inserts a GTS descriptor into a subsequent beacon. The GTS descriptor contains the short address of the requesting device, the number of the allocated GTSs, and their position within the GTS interval. These slots remain allocated to the device and are used every time they are announced in the GTS descriptor until either voluntarily relinquished by the device or explicitly revoked by the coordinator.

The device can request deallocation of the GTSs by sending an explicit control frame. The coordinator can also revoke the GTSs reserved if it observes that the slots have not been used within a specified number of superframes. The coordinator informs the device of its decision to deallocate the GTSs reserved by generating a GTS descriptor with a start slot of value zero. The second scenario occurs when not enough slots are available to honor the device's requests. In this case, the coordinator generates a GTS descriptor with an invalid time slot value of zero, but specifies the amount of available slots in the descriptor length field. Depending on the type of data to be exchanged and the nature of the transaction with the coordinator,

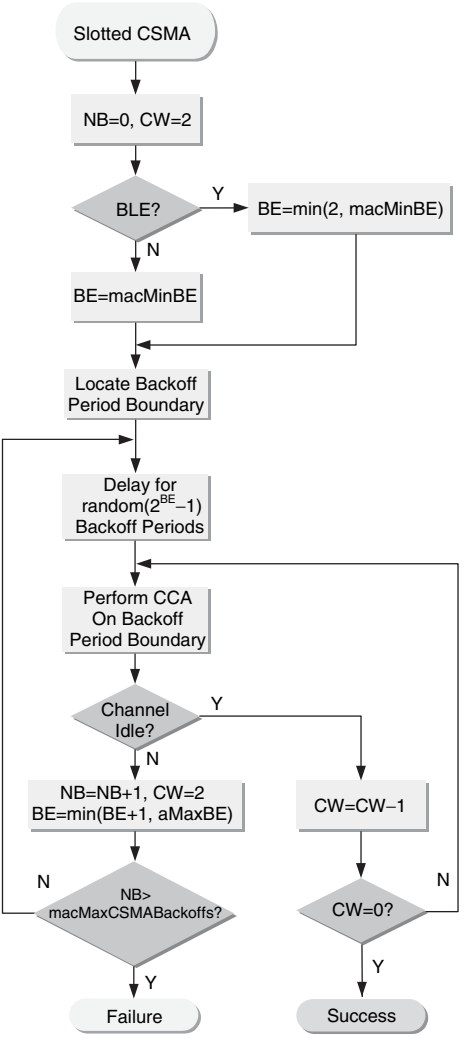
the device may renegotiate its request. The third scenario occurs when the interval GTSDescPersistenceTime elapses and the coordinator does not insert an appropriate GTS descriptor into one of the beacon frames. The device concludes that the resource request failed.

If the GTS allocation request is confirmed by the coordinator, the device uses these slots for communication. Depending on the type of the request, the GTSs allocated can be either transmit or receive. In the case of a transmit GTS, the device wakes up before the start of the allocated GTSs and uses the reserved contiguous slots to transmit its data to the coordinator. The latter acknowledges the receipt of the data immediately. Similarly, if the allocated GTS is of the receive type, the device wakes up at the beginning of the reserved time slots and receives the data transmitted by the coordinator at the start of the same slot. Upon receiving the data, the device completes the data transfer transaction by acknowledging receipt of the data. It is worth noting, however, that the receive transaction can be carried out successfully only if the device has reserved enough time slots to cover the transmission of the data packet and its corresponding acknowledgment, along with the required interframe spacing.

If the number of allocated time slots for a transmit transaction is not sufficient to cover the entire cycle of the transaction, the device must send its data during the CAP of the active period. This is also the case when the device does not have an allocated slot. In the case of a receive transaction, if the coordinator cannot use a receive GTS allocation, it announces the buffered packet to the intended recipient by including its address in the pending address field of the beacon frame. In response, the receiving device sends a special data packet request during the CAP of the active period and sets its transceiver on in preparation of the incoming packet. The coordinator acknowledges the request and proceeds to transmit the packet immediately. Notice that the coordinator continues to include the device's address in the pending address field, as long as the packet is still buffered or until its associated timer expires. If the data request fails to trigger an acknowledgment from the coordinator, the device may reiterate its attempt in one of the subsequent superframes. It can also switch off its transceiver until the next beacon transmission.

*Contention-Based Channel Access* The access to the medium during the CAP of an active period is regulated using a slotted, nonpersistent CSMA protocol hardware. The protocol, however, does not address the hidden terminal problem. As such, it does not use any mechanisms, such as a RTS/CTS handshake, to alleviate this problem. To reduce the likelihood of collisions, the protocol uses random delays. The basic steps of the contention-based MAC protocol are described in Figure 5.19.

Contrary to the traditional protocols, the CAP access protocol does not use the superframe slots for its back-off procedure. Instead, it uses the back-off period to accelerate the contention resolution process. A device attempting to transmit a date packet during the CAP of an active period first synchronizes its transceiver with the coordinator's beacon. It then initializes and maintains three main variables: NB,




BLE: Battery Life Extension

**Figure 5.19** Slotted CSMA algorithm.

BE, and CW. The variable NB, initially set to zero, counts the number of back-offs. The variable BE denotes the current back-off exponent. The initial value of this variable is set to macBinBE, a protocol-specific parameter. The last variable, CW, represents the current congestion window.

Prior to transmitting the new packet, the device first aligns itself with the boundary of the next back-off period. It then draws an integer number, $i$, between $[0, 2^{BE}-1]$ and waits for $i$ back-off periods. At the expiration of this waiting interval,

the contending device performs a clear channel assessment (CCA) operation. If the channel is idle, the contending device decrements CW by 1. If the value of CW is positive, the device performs a second CCA operation to determine the current state of the channel. If the channel is clear, the device decrements CW, which then becomes zero, declares the channel idle, and proceeds with the transmission of the packet.

If the second CCA operation indicates a busy channel, however, the device increases the number of back-offs, BE, by 1 and sets CW to 2. It then computes a new BE as min*(BE+1, aMaxBE)*, where *aMaxBE* is a protocol-specific parameter. If the number of back-offs, NB, exceeds the maximum back-off threshold, MaxCS MABack-offs, the device drops the frame and declares failure. Otherwise, the device waits for a number of back-off periods, randomly drawn from $[0, 2^{\mathrm{BE}}-1]$, before it makes a new attempt to transmit the frame.

It is worth noting that a contending device must assert that the channel is clear for two consecutive back-off periods to account for the hardware nonzero receive-to-transmit turnaround time. Failure to do so may cause the contending device to senses the medium as clear during the turnaround time of an exchange between two devices and wrongly declares the channel idle. The attempt to transmit the frame results in a collision.

As stated previously, the CAP access control protocol is designed to greatly reduce, when adequate, the device duty cycle, especially in low-activity networks. For applications running on low-activity networks, even the minimum duration of a CAP interval exceeds the length of the time interval necessary to perform the low activity of the network. To address this situation, the CAP access control protocol offers a battery life extension (BLE) mode of operation. The BLE mode allows devices to go into sleep mode in the presence of low activity. To trigger a BLE mode of operation, the coordinator sets a BLE flag in its beacon frame. It then limits its monitoring of the CAP to six back-off periods. If no activity is heard within this period, the coordinator returns to sleep. Upon detecting the BLE flag, a device attempting to communicate with the coordinator sets the initial value of the back-off exponent to 2 or less, thereby reducing considerably the channel sensing period. Although such an action may increase the likelihood of a collision, it has been shown that using BLE with the superframe order (SO) set to 14 greatly reduces the total system duty cycle.

*Beaconless Mode of Operation* A beaconless mode is better suited for applications where remote units such as intrusion sensors and motion detectors wake up on a regular, yet random basis to report on events as they occur. The network coordinator, mains powered, is continuously awake waiting to hear from each of these units. In the beaconless mode, the network coordinator does not send a beacon frame on a regular basis. Furthermore, this mode of operation does not support allocation of guaranteed slots for low-latency applications and applications requiring a specific data rate. Instead, devices compete for channel access using an unslotted, nonpersistent CSMA/CA protocol. When a device wishes to transmit data, it waits for a random number of back-off periods before sensing the channel. If the channel

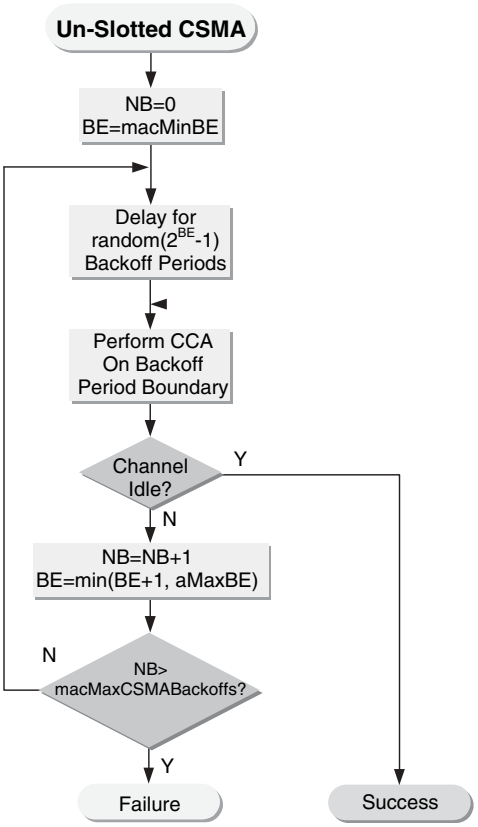



**Figure 5.20** Unslotted CSMA algorithm.

is busy, the device increases the number of attempts by one and checks if the maximum number of attempts has been reached. If the limit is exceeded, the device generates a channel access error and reports this event to upper layers. If the number of attempts is below the limit, the device reiterates this procedure until it either captures the channel successfully or the number of attempts exceeds the limit. The basic steps of this protocol are described in Figure 5.20. It is worth noting that the absence of a synchronization mechanism, such as the beacon, coupled with the unslotted nature of the access protocol, devices are no longer required to locate the back-off period boundaries, as was the case in the slotted CSMA access protocol. Furthermore, the devices are no longer required to execute only a single CCA operation. In summary, the IEEE 802.15.4 MAC-layer standard is designed for secure, low-power operations. Due to the low data throughput of the applications envisioned to use the IEEE 802.15.4 standard, the MAC-layer protocol includes several mechanisms to keep the average power consumption of the network devices low. This is achieved largely by allowing nodes to operate at low duty cycles while maintaining network-level connectivity.

By adopting a loosely synchronized sleep and wake-up cycle, network nodes can operate for extended periods of time with minimum energy consumption. To accommodate different data types, the standard defines different modes of operations. Support for these traffic types is achieved by the use of an optional superframe for the beacon-enabled operation mode, with the possibility of guaranteed slot allocation to accommodate low-latency applications. The standard also offers a beaconless mode of operations to support star-based topologies for monitoring and security applications.

## 5.7 CONCLUSION

Sensor networking is an emerging technology that has a wide range of potential applications, including critical infrastructure protection, environmental monitoring, smart spaces, ubiquitous and pervasive health care, and robotic exploration. A WSN normally consists of a large number of distributed, battery-operated nodes equipped with one or more sensors, embedded processors, and low-power radios. These nodes cooperate to organize themselves into a multihop wireless network. The design of efficient MAC-layer protocols for WSNs is crucial for the wireless sensor nodes to carry out successfully the mission for which they are deployed.

The choice of the medium access control protocol is the major determining factor in WSN performance. Several attributes must be considered in the design of an efficient MAC layer protocol for WSN. Sensor network are likely to be battery powered, and it is often difficult, if not impossible, to change or recharge the batteries on these nodes. An efficient design of a MAC-layer protocol for a WSN must therefore be energy efficient to extend the lifetime of the network [5.10]. The MAC-layer protocol must also be scalable to accommodate changes in he network size, node density, and network topology. Finally, access fairness, reduced latency, high throughput, and bandwidth utilization are also important attributes in the design of MAC layer protocols for WSNs.

A number of access control protocols have been proposed for WSNs. In this chapter we discuss the fundamental design issues of medium access control for WSN methods and provide an overview of these protocols. In the first part of the chapter, a description of the basic requirements of access control protocols is provided. The following section categorizes the major media access control techniques used in shared medium access networks. In the last part of the chapter we discuss specific requirements of access control methods for WSNs and describe several MAC layer protocols for these networks. Two cases studies focused on a detailed description of two IEEE 802.11 inspired protocols, S-MAC and IEEE 802.15.4. S-MAC is designed explicitly for WSNs. The MAC-layer protocol uses periodic and coordinated sleeping to reduce energy consumption. It also uses message passing to reduce contention latency for delay-sensitive sensor applications.

The IEEE 802.15.4 standard is designed to provide support for low-data-rate connectivity among relatively inexpensive, minimally powered devices, typically

residing within 30 ft or less of each other. The type of these devices ranges from sensors and switches for industrial and residential use to interactive toys, inventory tracking, smart tags, and badges. The IEEE 802.15.4 wireless MAC- and PHY-layer specifications allow a collection of portable and moving devices, which operate at a data rate of 10 to 250 kpbs, to form ad hoc personal area networks within which these devices and interact directly.

The development of a MAC-layer protocol for sensor networks is likely to continue to be a topic of interest as new WSNs continue to emerge. Furthermore, recent developments in cognitive radio are likely to bring a new perspective to the design of MAC-layer protocol for WSNs. The ability of a cognitive radio to interact directly with its environment will enhance the ability of a wireless network device equipped with such a radio to better adapt to and interact with its environment while carefully managing its energy consumption.

## REFERENCES

[5.1] J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, K. Pister, ''System Architecture Directions for Networked Sensors,'' *Proceedings of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems*, Cambridge, MA, Nov. 2000, pp. 93–104.

[5.2] Y.-C. Tseng, C.-S. Hsu, T.-Y. Hsieh, ''Power-Saving Protocols for IEEE 802.11-Based Multi-hop Ad Hoc Networks,'' *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies* (InfoCom'02), New York, June 2002, pp. 200–209.

[5.3] J. Zhao, R. Govindan, ''Understanding Packet Delivery Performance in Dense Wireless Sensor Networks,'' *Proceedings of the 1st ACM Conference on Embedded Networked Sensor Systems* (SenSys'03), Los Angeles, Nov. 2003, pp. 1–13.

[5.4] V. Bharghavan, A. Demers, S. Shenker, L. Zhang, ''MACAW: A Media Access Protocol for Wireless LANs,'' *Proceedings of the ACM SIGCOMM*, Portland, Oregon, 1994, pp. 212–225.

[5.5] T. V. Dam, K. Langendoen, ''An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks,'' *Proceedings fo the 1st ACM Conference on Embedded Networked Sensor Systems* (SenSys'03), Los Angeles, Nov. 2003.

[5.6] J. Ding, K. Sivalingam, R. Kashyapa, L. J. Chuan, ''A Multi-layered Architecture and Protocols for Large-Scale Wireless Sensor Networks,'' *Proceedings of the IEEE 58th Vehicular Technology Conference* (VTC'03), Oct. 2003, Vol. 3, pp. 1443–1447.

[5.7] P. Karn, ''MACA: A New Channel Access Method for Packet Radio,'' *Proceedings of the ARRL/CRRL Amateur Radio 9th Computer Networking Conference*, Sept. 1990, pp. 134–140.

[5.8] ''Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications,'' ANSI/IEEE Std. 802.11-1999, 1999.

[5.9] L. Bao, J. J. Garcia-Luna-Aceves, ''A New Approach to Channel Access Scheduling for Ad Hoc Networks,'' 2001, *Proceedings of the 7th ACM International Conference on Mobile Computing and Networking* (MobiCom'01), Rome, Italy, July 2001. pp. 210–221.

[5.10] J. Monks, V. Bharghavan, W.-M. Hwu, "A Power Controlled Multiple Access Protocol for Wireless Packet Networks," *Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies* (InfoCom'01), Anchorage, AK, Apr. 2001.

[5.11] K. Sohrabi, G. J. Pottie, "Performance of a Novel Self-Organization Protocol for Wireless Ad Hoc Sensor Networks," *Proceedings of the IEEE 50th Vehicular Technology Conference* (VTC'99), 1999, pp. 1222–1226.

[5.12] G. D. Bacco et al., "A MAC Protocol for Delay-Bounded Applications in Wireless Sensor Networks," *Proceedings of the 3rd Annual Mediterranean Ad Hoc Networking Workshop*, June 2004.

[5.13] A. El-Hoiydi, "Spatial TDMA and CSMA with Preamble Sampling for Low Power Ad Hoc Wireless Sensor Networks," *Proceedings of the 7th IEEE International Symposium on Computers and Communications* (ISCC'02), July 2002, pp. 685–692.

[5.14] E. McCune, "DSSS vs. FHSS Narrowband Interference Performance Issues," *RF Signal Processing*, Sept. 2000.

[5.15] L. Kleinrock, Fouad Tobagi, "Packet Switching in Radio Channels, Part I: Carrier Sense Multiple Access Modes and Their Throughput Delay Characteristics," *IEEE Transactions on Communications*, Vol. 23, No. 12, Dec. 1975, pp. 1400–1416.

[5.16] F. A. Tobagi, L. Kleinrock, "Packet Switching in Radio Channels: Part II: The Hidden Terminal Problem in Carrier Sense Multiple Access and the Busy Tone Solution," *IEEE Transactions on Communications*, Vol. 23, Dec. 1975, pp. 1417–1433.

[5.17] T. S. Rappaport, *Wireless Communications: Principles and Practice*, 1996, Prentice Hall, Upper Saddle River, NJ.

[5.18] S. Xu, T. Saadawi, "Does the IEEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?" *IEEE Communications*, June 2001, pp. 130–137.

[5.19] Y. C. Tay, K. Jamieson, H. Balakrishnan, "Collision-Minimizing CSMA and Its Applications to Wireless Sensor Networks," *IEEE Journal on Selected Areas in Communications*, Vol. 22, No. 6, Aug. 2004, pp. 1048–1057.

[5.20] D. J. Baker, J. Wieselthier, "A Distributed Algorithm for Scheduling the Activation of Links in a Self-Organizing, Mobile, Radio Network," *Proceedings of the International Conference on Communications*, 1982, pp. 2F.6.1–2F.6.5.

[5.21] P. Lin, C. Qiao, X. Wang, "Medium Access Control with a Dynamic Duty Cycle for Sensor Networks", *IEEE Wireless Communications and Networking Conference* (WCNC'04), Mar. 2004, Vol. 3, pp. 1534–1539.

[5.22] G. Lu, B. Krishnamachari, C. Raghavendra, "An Adaptive Energy-Efficient and Low-Latency MAC for Data Gathering in Sensor Networks," presented at the Workshop on Energy-Efficient Wireless Communications and Networks (EWCN'04), held in conjunction with the IEEE International Performance Computing and Communications Conference (IPCCC), Apr. 2004.

[5.23] S. S. Kulkarni, "TDMA Services for Sensor Networks," *Proceedings of the 24th International Conference on Distributed Computing Systems Workshops*, Mar. 2004, pp. 604–609.

[5.24] K. Sohrabi, G. J. Pottie, "Performance of a Novel Self-Organization Protocol for Wireless Ad Hoc Sensor Networks," *Proceedings of the IEEE 50th Vehicular Technology Conference* (VTC'99), 1999, pp. 1222–1226.

[5.25] K. Sohrabi, J. Gao, V. Ailawadhi, G. J. Pottie, ''Protocols for Self-Organization of a Wireless Sensor Network,'' *IEEE Personal Communications*, Vol. 7, No. 5, Oct. 2000, pp. 16–27.

[5.26] J. C. Haartsen, ''The Bluetooth Radio System,'' *IEEE Personal Communications*, Feb. 2000, pp. 28–36.

[5.27] ''Specification of the Bluetooth System: Core,'' http://www.bluetooth.org/, 2001.

[5.28] F. Bennett, D. Clarke, J. B. Evans, A. Hopper, A. Jones, D. Leask, ''Piconet: Embedded Mobile Networking,'' *IEEE Personal Communications*, Vol. 4, Oct. 1997, pp. 8–15.

[5.29] W. R. Heinzelman, A. Chandrakasan, H. Balakrishnan, ''Energy Efficient Communication Protocols for Wireless Microsensor Networks,'' *Proceedings of the 33rd Hawaii International Conference Systems Sciences* (HICSS'00), Maui, HI, Jan. 2000, pp. 3005–3014.

[5.30] W. Heinzelman, A. Sinha, A. Wang, A. Chandrakasan, ''Energy-Scalable Algorithms and Protocols for Wireless Microsensor Networks,'' *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00)*, June 2000.

[5.31] W. Heinzelman, ''Application-Specific Protocol Architectures for Wireless Networks,'' Ph.D. dissertation, Massachusetts Institute of Technology, June 2000.

[5.32] J. Elson, D. Estrin, ''Time Synchronization for Wireless Sensor Networks,'' *Proceedings of the 15th International Symposium on Parallel and Distributed Processing*, San Francisco, CA, Apr. 2001.

[5.33] S. Ganeriwal, R. Kumar, M. B. Srivastava, ''Timing-Sync Protocol for Sensor Networks,'' *Proceedings of the 1st ACM Conference on Embedded Networked Sensor Systems* (SenSys'03), Los Angeles, Nov. 2003.

[5.34] C. L. Fullmer, J. J. Garcia-Luna-Aveces, ''Solutions to Hidden Terminal Problems in Wireless Networks,'' *Proceedings of the ACM SIGCOMM Conference*, 1997, pp. 39–49.

[5.35] S. Singh, C. S. Raghavendra, ''PAMAS: Power Aware Multi-access Protocol with Signalling for Ad Hoc Networks,'' *ACM Computers in Communications Review*, Vol. 28, No. 3, July 1998, pp. 5–26.

[5.36] C. Schurgers, V. Tsiatsis, S. Ganeriwal, M. Srivastaval., ''Optimizing Sensor Networks in the Energy-Latency-Density Design Space,'' *IEEE Transactions on Mobile Computing*, Vol. 1, No. 1, Jan.–Mar. 2002.

[5.37] A. El-Hoiydi, J.-D. Decotignie, ''Wisemac: An Ultra Low Power Mac Protocol for Multi-hop Wireless Sensor Networks,'' *Algorithmic Aspects of Wireless Sensor Networks: 1st International Workshop* AlgoSensors'04, Turku, Finland, July 2004.

[5.38] C. C. Enz, A. El-Hoiydi, J.-D. Decotignie, V. Peiris, ''WiseNET: An Ultralow-Power Wireless Sensor Network Solution,'' *IEEE Computer*, Vol. 37, No. 8, Aug. 2004.

[5.39] K. Jamieson, H. Balakrishnan, Y. C. Tay, ''Sift: A MAC Protocol for Event-Driven Wireless Sensor Networks,'' Technical Report 894, MIT Laboratory for Computer Science, Cambridge, MA, http://www.lcs.mit.edu/publications/pubs/pdf/MIT-LCS-TR-894.pdf, May 2003.

[5.40] V. Rajendran, K. Obraczka, J. J. Garcia-Luna-Aceves, ''Energy-Efficient, Collision-Free Medium Access Control for Wireless Sensor Networks,'' *Proceedings of the 1st ACM Conference on Embedded Networked Sensor Systems* SenSys'03, Los Angeles, Nov. 2003, pp. 181–192.

[5.41] J. Polastre, J. Hill, D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks," *Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems* (SenSys'04), Baltimore, MD, Nov. 2004.

[5.42] A. Woo, D. Culler, "A Transmission Control Scheme for Media Access in Sensor Networks," *Proceedings of the 7th ACM/IEEE International Conference on Mobile Computing and Networking* (MobiCom'01), Rome, Italy, July 2001, pp. 221–235.

[5.43] A. Woo, T. Tong, D. Culler, "Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks," *Proceedings of the 1st ACM Conference on Embedded Networked Sensor Systems* (SenSys'03), Nov. 2003, Los Angeles, pp. 14–27.

[5.44] W. Ye, J. Heidemann, D. Estrin, "A Flexible and Reliable Radio Communication Stack on Motes," Technical Report ISI-TR-565, Information Sciences Institute, University of Southern California, Los Angeles, Sept. 2002.

[5.45] W. Ye, J. Heidemann, D. Estrin, "Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks," *IEEE/ACM Transactions on Networking*, Vol. 12, No. 3, June 2004, pp. 493–506.

[5.46] W. Ye, J. Heidemann, D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies* (InfoCom'02), New York, June 2002, pp. 1567–1576.

[5.47] IEEE 802.15.4 Standard-2003, "Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LRWPANs)," IEEE-SA Standards Board, 2003, http://grouper.ieee.org/groups/802/15/pub/TG4b.html.

[5.48] "ZigBee Specification," Document 053474r05, Version 1.0, ZigBee Alliance, Bishop Ranch, CA, Dec. 14, 2004, available (since June 2005) at http://www.zigbee.org/.