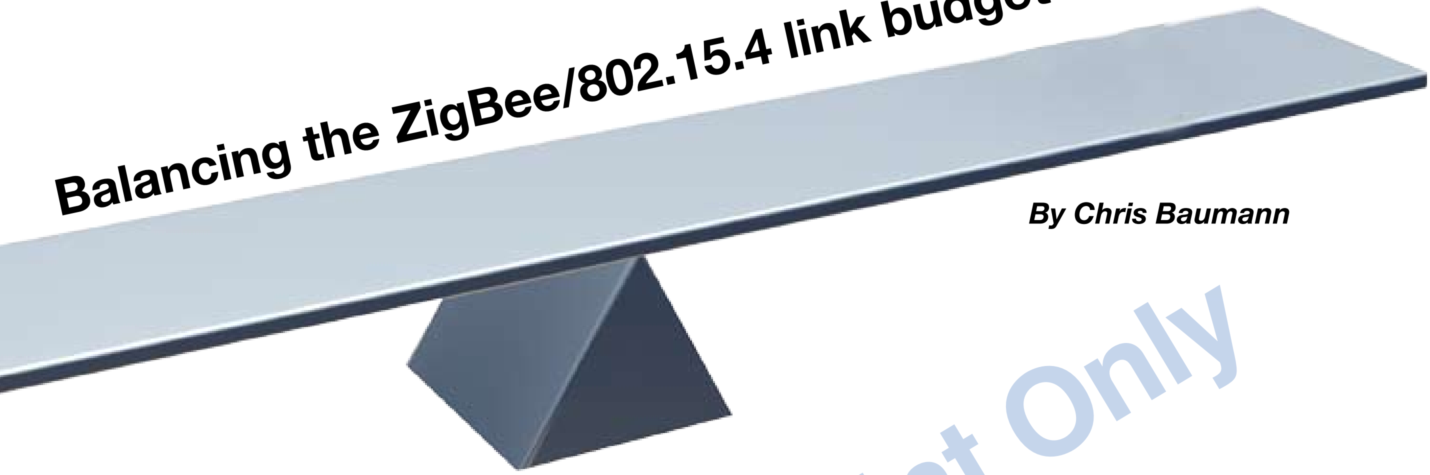
Balancing the ZigBee/802.15.4 link budget



By Chris Baumann

Networking Systems-on-Chip (SoCs) are becoming increasingly important, and ZigBee/802.15.4 systems are not all created equal. Chris describes considerations for choosing solutions and discusses differences in single-chip versus multichip solutions.

ZigBee is a low data rate, wireless networking standard based on IEEE 802.15.4 that can eliminate the need for hard-wiring industrial control networks. All ZigBee networks are 802.15.4 networks, with the ZigBee standard providing security and application layers that ensure interoperability between equipment from different vendors. Interoperability is probably less of an issue in industrial control than it is in security, lighting, or climate control applications. This distinction is important because the ZigBee standard is still evolving, while 802.15.4 is simpler to implement and ready to go now. Figure 1 shows the 802.15.4 hardware/software solution.

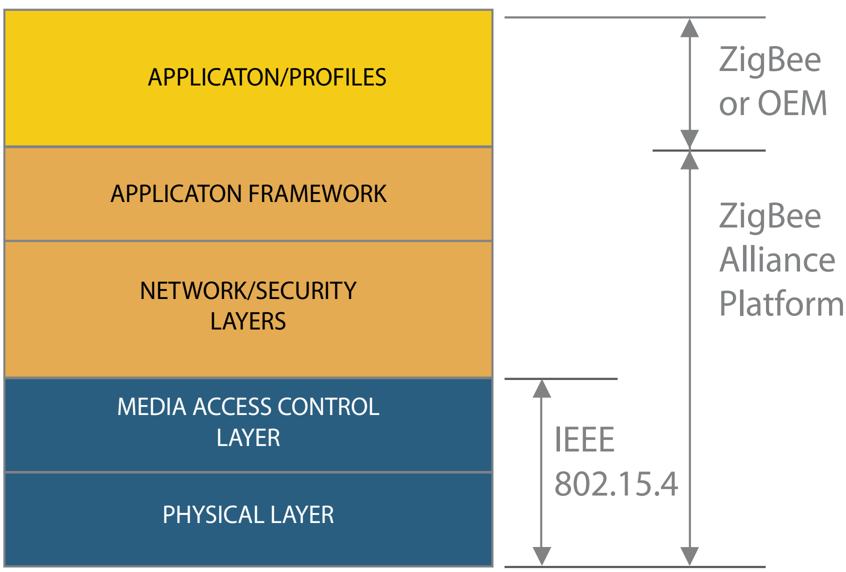


Figure 1

Networks, whether wired or wireless, provide communication for industrial control applications, but generally do not control a chemical or manufacturing process. Compared to WLAN, WiMAX, Bluetooth, and so on, ZigBee/802.15.4 networks are different – their primary application is the industrial control application. The dual sensor-plus-network nature of 802.15.4 applications adds an extra layer of complexity to the design challenge, compounded by the fact that most industrial control engineers are not RF design experts. They may not know which RF parameters are most important, how to evaluate an 802.15.4 software stack, or how to interface the controller with the radio.

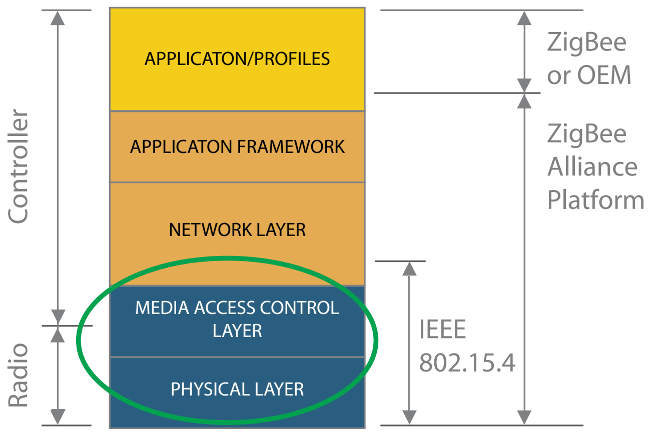
Fortunately, many 802.15.4 vendors, including Atmel, Jennic, Texas Instruments, Freescale, and Ember, are assembling integrated, system-level solutions that include the 802.15.4 radio, controller, all interfaces and 802.15.4/ZigBee software stacks, and development kits. Among these, designers still need some criteria for evaluating which solution best meets cost, performance, topology, and flexibility constraints.

Six basic issues affect the choice of an 802.15.4 solution:

1. How simple or complex the network must be
2. Choosing the 802.15.4 radio frequency for the application
3. RF parameters and system cost
4. The architecture of the Media Access Controller (MAC)
5. Should it be a single-chip or two-chip solution
6. Factors affecting power consumption

Network complexity

Several types of network configurations can be implemented under the 802.15.4



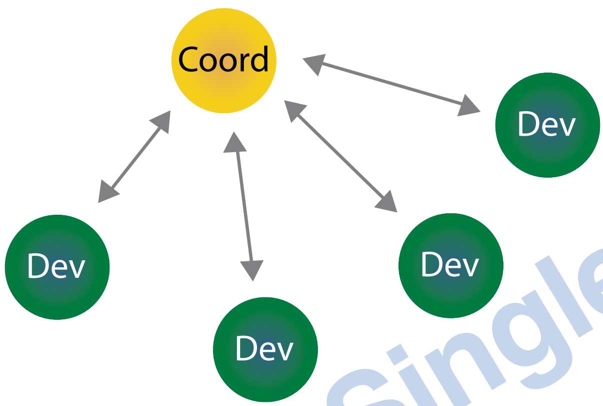


Figure 2

and ZigBee umbrella: point-to-multipoint (star) networks, tree networks, and mesh networks. Star networks (Figure 2), typically used for low-cost gaming or entertainment center control, are the simplest to implement and require the least amount of code for setup and control. They are usually limited in the quantity of nodes and coverage.

Tree networks are used for applications such as access or industrial control sensing. Since they allow more nodes, they can cover a larger area than star networks. However, they may suffer from latency effects that can cause unacceptable data delays for critical applications. Tree networks may be subject to critical node failure leading to system failure, and usually need more code to implement than multipoint systems.

Mesh networks (Figure 3) represent the highest level of 802.15.4/ZigBee configuration and require the most network level code. Mesh networks can *self-heal* critical node failures, making them ideal for large building control systems or wide area sensing. They are by far the most difficult 802.15.4/ZigBee networks to design and implement.

In terms of system implementation, the simpler the network, the better. A simpler network allows the design team to focus on the industrial control application rather than the network. For the vast majority of industrial control applications, star or tree networks should suffice. These simpler networks do not require a full ZigBee implementation and are therefore easier to build and integrate into the application using IEEE 802.15.4 alone.

Extremely large and/or self-healing networks require ZigBee to manage the mesh networking functionality.

868 MHz, 902 MHz, or 2.4 GHz

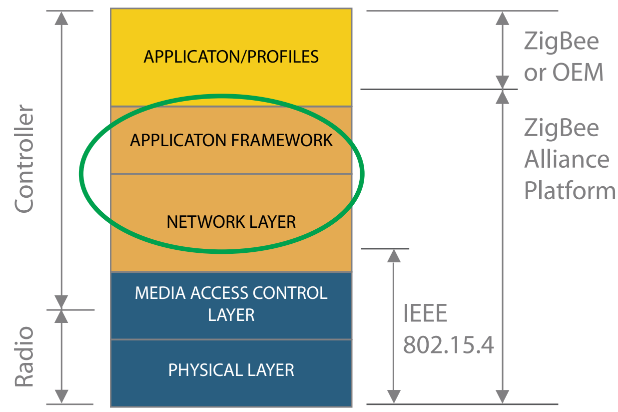
The 802.15.4 standard defines three radio frequencies: 868 MHz (available only in the European Union), 902 MHz (available in the United States), and 2.4 GHz (worldwide). Table 1 summarizes each radio frequency's data rate.

Frequency	Data rate	Channels
868 MHz	20 mbps	1
902 MHz	40 mpbs	16
2.4 GHz	250 mbps	40

Table 1

The majority of 802.15.4 radios on the market today operate in the 2.4 GHz band. This unlicensed frequency is available all over the world, so an application that requires worldwide interoperability should definitely use the 2.4 GHz band.

However, 2.4 GHz radios have some disadvantages. For one, the 2.4 GHz band is crowded. Bluetooth, WLAN, microwave ovens, and garage door openers all operate in this unlicensed band, increasing the likelihood of interference. There is virtually no interference in the 868/902 MHz bands except for some older cordless phones and keyboard mice. The higher sensitivity



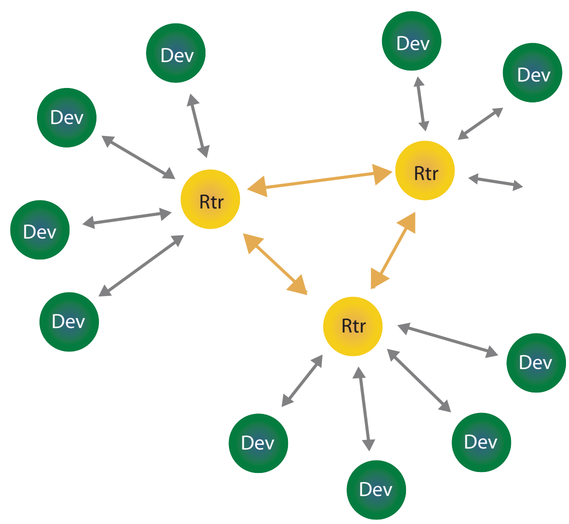


Figure 3

(-92 dBm versus -85 dBm) and the inherently better wall penetration of the 868/902 MHz radios allow them to be spaced farther apart, potentially lowering the cost of the network. At the same distance, lower-band radios also consume less power than 2.4 GHz radios due to their better sensitivity and wall penetration.

The 900 MHz band is not widely available in the European Union, thus it is not practical for applications that must be interoperable between the United States and Europe. However, the relative emptiness of this band in non-European geographies, combined with low power and high sensitivity make 900 MHz radios good candidates for industrial or other applications that do not need global interoperability.

RF parameters and system costs

It is probably not necessary to be an RF expert to implement an 802.15.4 industrial control application. However, some radio-related issues have significant implications for system cost: *receiver sensitivity*, *transmit power*, and the *link budget*.

Receiver sensitivity is the minimum power in decibels (dBm) at which a radio can reliably receive data. A large negative dBm number indicates *higher* receiver sensitivity. The 802.15.4 standard specifies a minimum receiver sensitivity of -85 dBm for 2.4 GHz radios and -92 dBm for 900 MHz radios. All 802.15.4 radio vendors exceed these standards, offering radios with receiver sensitivities that range between -90 dBm and -100 dBm. Higher sensitivity allows radios to be spaced farther apart, directly cutting system costs. Higher sensitivity also can reduce or eliminate the need for power amplifiers used to boost signal strength when receiver sensitivity is low.

Very small differences in sensitivity result in very large differences in the number of radios required. Although 6 dBm may not seem like much, improving the receiver sensitivity of an 802.15.4 radio from -94 dBm to -100 dBm effectively doubles its line-of-sight range and allows half as many radios to cover the same area. For example, if a radio with -94 dBm receiver sensitivity has a 100-meter range, increasing that sensitivity to -100 dBm will increase its range to 200 meters.

Transmit power drives radio range – the higher the power, the longer its range for a desired signal strength. The 802.15.4 standard requires radios to have a minimum output power of -3 dBm, or 0.5 mW. Radios on the market today have output power of between 0 dBm (1 mW) and 3 dBm (2 mW).

The *link budget* is determined by adding together the absolute values of receiver sensitivity and transmit power. It influences both the line-of-sight range and the robustness of non line-of-sight transmissions of a transmitter/receiver pair. The better the receiver sensitivity and the higher the transmit power, the higher the link budget and the longer the range for both line-of-sight and non line-of-sight. A vendor comparison of link budgets for 802.15.4 radios is shown in Table 2.

For example, Chipcon's CC2420 2.4 GHz 802.15.4 radio has transmit power of 0 dBm (1 mW) and receiver sensitivity of -94 dBm, while Atmel's Z-Link radio has transmit power of 3 dBm (2 mW) and receiver sensitivity of -100 dBm. The Chipcon radio's link budget is 94 dBm and the Atmel radio has a link budget of 103 dBm.

Under the same conditions, if the Chipcon radio's range was 100 meters, the Atmel radio's range would be 280 meters. This means that about one-third as many nodes could be required to cover the same network area, reducing system cost dramatically in a large network.

Media access controller architecture

A MAC is software that provides the interface between the network security layer and the 802.15.4 radio. MAC implementations can affect system complexity, performance, power consumption, cost, and scalability of system features.

An 802.15.4 MAC can require up to 24 kB of memory, creating a trade-off between the *fullness* of the MAC and system cost. Some vendors optimize the MAC code to produce the smallest memory footprint for the target controller while keeping the full MAC feature set. Others eliminate features, such as *Guaranteed Time Slot* (GTS), that are deemed noncritical. While this latter approach may reduce costs by allowing the use of a microcontroller with a smaller flash memory, it can have an adverse impact on system scalability to next-generation applications. For example, if an application that has a MAC without GTS evolves to a future generation that needs GTS, the MAC and possibly the entire network layer will have to be redesigned and ZigBee recertified.

Another, potentially more effective means of addressing the code density issue is to select a C/C++ friendly microcontroller and compiler that provide compact compiled code. For example, the compiled code for an 802.15.4 application that requires 55 kB of flash on an 8051-based microcontroller needs only 30 kB on an AVR-based MCU. Choosing the higher-density MCU cuts total code size by almost half. On the compiler side, IAR compilers are known to compile 802.15.4 code that is 20 percent denser than the same code compiled using GCC's GNU compiler.

Vendor	Part number	Receiver sensitivity (dBm)	Transmit power (dBm)	Link budget (dBm)	Approximate line-of-sight range (m)
Atmel	AT86RF230	-100	3	103	1,450
Chipcon	CC/EM 2420	-94	0	94	500
	CC 2430	-90	0	90	300
Freescale	MC1320x	-94	0	94	500
	MC1321x	-94	0	94	500
Ember	EM250	-94	0	94	500
	EM260	-94	0	94	500
Jennic	JN5121	-93	1	94	500

Table 2

The MAC architecture also can affect system performance. Processor resources must be shared between the 802.15.4 MAC sublayer, the 802.15.4 network layer (which provides network configuration, manipulation, and message routing), and the industrial control application. The two basic approaches to resource scheduling are cooperative multitasking and preemptive multitasking.

In cooperative multitasking, every task voluntarily cedes the microcontroller to the next, resulting in lower program code size because complicated scheduling algorithms are avoided. In addition, context switching isn't required, so there is less latency and smaller memories can be used. The drawback of cooperative multitasking is the amount of *trust* involved. Each process must regularly give processor time to other processes. A poorly designed program or a *hung* task can effectively bring the system to a halt. Designing a system so that it avoids these pitfalls can be onerous and may result in irregular or inefficient use of system resources.

Preemptive multitasking initiates a context switch that satisfies the scheduling policy's priority constraint. It preempts the active task and prevents a hung task from halting the system. This requires more code and introduces latencies into the system. While cooperative multitasking gives the application designer control over scheduling, preemptive multitasking gives scheduling control to the operating system and software stack. A typical 802.15.4 application will not usually need this level of protection and can generally go with the smaller code size and lower latency of a cooperative multitasking scheme.

Single-chip or chipset

Although it may seem intuitive that a single-chip solution would be preferable to a multichip solution, in 802.15.4 applications, this isn't the case because there are different types of nodes for different functions. Full function devices act as gateway servers or routers and can be quite complex, while Reduced Function Devices (RFDs) can be as simple as a sensor or a switch. Obviously the amount of processing, code size, peripherals, context switching, and memory required will be much more substantial than for an end node with a sensor.

The two single-chip offerings on the market today are overkill for many end nodes and may not have the horsepower to execute the industrial control application itself, mandating additional controllers for the primary application and increasing system complexity substantially. Someday there will be single-chip 802.15.4 devices that can cost effectively address all the various node types, but for now it is preferable to select an 802.15.4 vendor offering multiple microcontrollers with a range of memory densities optimized for their radios.

Controller and full function nodes, such as those in gateway servers or electrical equipment, are usually hardwired to a power source. Reduced function nodes, connected to sensors and switches, are usually battery powered. All battery-operated nodes should have a very long battery life – if possible longer than the life of the end product. The ZigBee standard mandates a two-year battery life for battery-powered nodes, and longer is always better. Imagine how annoying (not to mention expensive) it would be to replace all the sensors and switches in a process control system every few years.

A wide variety of vendors offer 802.15.4/ZigBee radios or controllers or both. These can be integrated on a single chip or come as a complete chipset. Any engineer who is not an expert in the integration of radios with controllers (that is, most engineers) should probably choose a complete solution from a single vendor. This path will vastly simplify product development and will give engineers much more freedom to develop the differentiating features of the end application.

Using a single-chip solution provides a small footprint and may lower power consumption. However, it also weds the engineer to a microcontroller that might not be the best for the target application. The embedded controller may not have all the necessary peripherals. Furthermore, although the embedded controller may have enough flash memory for the first-generation design, it may not offer a migration path to devices with bigger memories to accommodate the addition of new, software-based features. If there is no migration path to a controller with 128 kB or even 256 kB of flash, external chips may be required, increasing system cost, board size, and power consumption. Conversely, single-chip solutions do not offer the option of reducing costs by opting for a controller with smaller flash or fewer peripherals.

The 802.15.4/ZigBee market is in its infancy. Nobody knows yet which applications will get traction in the market or how those applications will evolve. There are probably dozens of applications that no one has even imagined. Therefore, at present, it makes sense to design an application with a discrete radio coupled with a family of microcontrollers that provides the flexibility to let applications evolve as the market evolves.

Power consumption considerations

Reduced function 802.15.4 nodes are often battery powered.

Active	Percent time on	Active power (mA)	Power consumed per transmission (mA)	Percent of total power consumption
Controller	0.00017	8.0000	0.0013	
Radio TX	0.00008	17.0000	0.0014	
Radio RX	0.00008	15.0000	0.0013	
Total active			0.0040	64.6
Sleep				
Controller	0.99983	0.0015	0.0015	
Radio	0.99983	0.0007	0.0007	
Total sleep			0.0022	35.4
0.00400 mA active mode + 0.0022 mA sleep mode power = 0.0062 mA				

Table 3

Any battery-powered node should have a battery life that outlasts the system itself because changing the battery inside a piece of industrial equipment, pipeline, or flow control valve can bring the system to a halt and be very expensive.

The factors most affecting power consumption include:

- Supply voltages of the radio and microcontroller
- Active current drawn by the radio and microcontroller
- Clock frequency at which the controller operates
- Number of external components required in the system (particularly power amplifiers)
- Code size, as it affects the MCU clock frequency

This indicates engineers should use low-voltage devices, avoid the use of power amplifiers, and strive for compact code. A less obvious factor, sleep-mode power consumption, is perhaps more important than these concerns. RFDs are likely to spend 99.9 percent of their time in sleep mode, waking up periodically for a few microseconds to check a sensor or poll other radios, then going right back to sleep. As a result, the total power consumption of an RFD is likely to approach sleep-mode power consumption. Vendors and engineers tend to emphasize active power consumption, and sleep-mode power consumption may be buried deep inside the data sheet.

Consider an 802.15.4 end node that wakes up once a minute, performs a task that takes 2 milliseconds, spends equal amounts of time on transmission and reception, and then goes back to sleep. Although the radio consumes 17 mA during transmit and 15 mA during receive and the controller consumes 8 mA active, the total power consumed in this scenario is 0.0062 mA, with sleep-mode power representing more than one-third of the total. This is shown in Table 3.

In a real-life temperature sensor node application, a microcontroller with active current of 8 mA and sleep current of 1.5 µA (with watchdog timer on), and a radio with transmit and receive currents of 17 mA and 15 mA and sleep current of 0.7 µA together consume 0.0011 mAh, for wakeup, sense, ADC conversion, data transmission, receive acknowledgement, and transition back to sleep mode. At a rate of one transmission per minute, this node would consume 0.0706 mA per hour of operation, allowing two AA 2,700 mAh lithium-ion batteries to last about 5.2 years. Adding just 1 µA to sleep mode power consumption would cut the battery life by about 10 percent to 4.8 years.

It's the link budget, folks

Industrial control engineers need not become RF experts to implement 802.15.4 or ZigBee-based industrial control systems. There are plenty of good radio and controller vendors that offer solutions for a variety of target applications.

When evaluating 802.15.4/ZigBee system solutions:

- Strive to achieve the highest possible link budget
- Pay particular attention to the robustness of the architecture and size of the media access controller, as well as the diversity and flexibility of the microcontrollers supported by the radio vendor
- Recognize that receiver sensitivity and transmit power can have a huge impact on the system
- Keep an eye on sleep-mode power consumption and the voltage operating range of both the radio and controller in applications with battery-powered end nodes IES

Chris Baumann *is director of Atmel's BiCMOS Products business unit. Prior to joining Atmel in 1989, he maintained various positions at Texas Instruments and Honeywell. Chris received his BS in Electrical Engineering and his MSEE from the University of Notre Dame.*

To learn more, contact Chris at:

Atmel
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
Tel: 719-540-7326
E-mail: cbaumann@cso.atmel.com
Website: www.atmel.com

