Avoiding Interference in the 2.4-GHz ISM Band

Designers can create frequency-agile 2.4 GHz designs using procedures provided by standards bodies or by building their own protocol.

By Ryan Winfield Woodings and Mark Gerrior, Cypress Semiconductor

As more and more companies produce products that use the 2.4-GHz portion of the radio spectrum, designers have had to deal with increased signals from other sources. Regulations governing unlicensed parts of the spectrum state that your device must expect interference.

How can designers get the best performance out of their 2.4-GHz solution under these hostile conditions? Often the product works in a controlled lab environment but then suffers performance degradation from the storm of interference from other 2.4GHz solutions in the field. With existing standards like Wi-Fi, Bluetooth, and ZigBee there is little that can be done beyond what the architects of the standard provide. But when the designer controls the protocol there are procedures that will minimize the interference from other sources.

In this article,we'll examine the various interference management techniques provided by 2.4 GHz wireless systems. We'll then show how low-level tools can be used to create frequency-stability in a 2.4 GHz design.

Wi-Fi

The two methods for radio frequency modulation in the unlicensed 2.4 GHz ISM band are frequency-hopping spread spectrum (FHSS) and direct-sequence spread spectrum (DSSS). Bluetooth uses FHSS while WirelessUSB, 802.11b/g/a (commonly known as Wi-Fi), and 802.15.4 (known as ZigBee when combined with the upper networking layers) use DSSS. All of these technologies operate in the ISM frequency band (2.400"2.483 GHz), which is available worldwide **(Figure 1 below)**.

The primary motivation for Wi-Fi is data throughput. Wi-Fi is typically used to connect computers to the local LAN (and indirectly to the Internet). Most Wi-Fi devices are laptops that are recharged daily or wall-powered access points and are therefore not power-sensitive.

Wi-Fi uses DSSS, with each channel being 22 MHz wide, allowing up to three evenly-distributed channels to be used simultaneously without overlapping each other. The channel used by each Wi-Fi access point must be manually configured; Wi-Fi clients search all channels for available access points.

802.11 uses an 11-bit pseudorandom noise (PN) code known as a Barker code to encode each information bit for the original 1 and 2 Mbit/s data rates. In order to achieve higher data rates 802.11b encodes six information bits into an eight-chip symbol using complementary code keying (CCK).

There are 64 possible symbols used in this CCK algorithm, requiring each 802.11b radio to contain 64 separate correlators (the device responsible for turning symbols into information bits), which increases the complexity and cost of the radio, but increases the data rate to 11 Mbit/s.

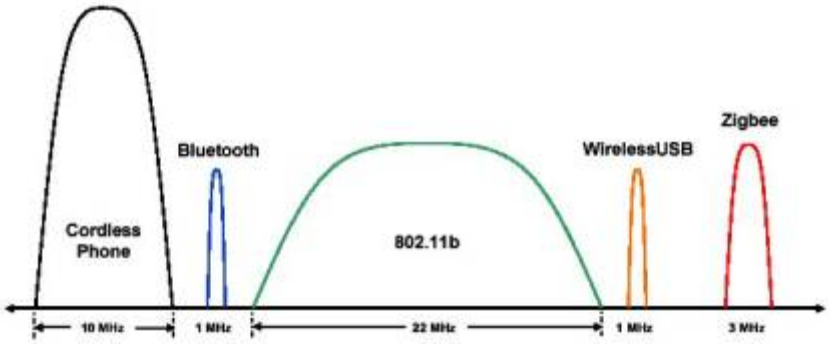

Figure 1: Signal comparison of wireless systems operating in the 2.4-GHz band.

Bluetooth
The focus of Bluetooth is ad-hoc interoperability between cell phones, headsets, and PDA's. Most Bluetooth devices are recharged regularly.

Bluetooth uses FHSS and splits the 2.4 GHz ISM band into 79 1 MHz channels. Bluetooth devices hop among the 79 channels 1600 times per second in a pseudo-random pattern. Connected Bluetooth devices are grouped into networks called piconets; each piconet contains one master and up to seven active slaves. The channel-hopping sequence of each piconet is derived from the master's clock. All the slave devices must remain synchronized with this clock.

Forward error correction (FEC) is used on all packet headers, by transmitting each bit in the header three times. A Hamming code is also used for forward error correction of the data payload of some packet types. The Hamming code introduces a 50% overhead on each data packet, but is able to correct all single errors and detect all double errors in each 15-bit codeword (each 15-bit codeword contains 10 bits of information).

2.4 GHz Technology Comparison

	Data Rate	Number of channels	Interference Avoidance Method	Minimum Quiet Bandwidth Required
Wi-Fi (802.11b)	11 Mbps	13	Fixed channel collision avoidance	22 MHz (Static)
Bluetooth	723 Kbps	79	Adaptive frequency hopping	15 MHz (Dynamic)
WirelessUSB	62.5 Kbps	79	Frequency agility	1 MHz (Dynamic)
Zigbee	128 Kbps	16	Fixed channel collision avoidance	3 MHz (Static)

WirelessUSB
WirelessUSB has been designed as a cable cutter for computer input devices (mice, keyboards, etc) and is also targeting wireless sensor networks. WirelessUSB devices are not recharged regularly and are designed to operate for months on alkaline batteries.

WirelessUSB uses a radio signal similar to Bluetooth but uses DSSS instead of FHSS. Each WirelessUSB channel is 1 MHz wide, allowing WirelessUSB to split the 2.4 GHz ISM band into 79 1 MHz channels like Bluetooth. WirelessUSB devices are frequency agile, in other words, they use a "fixed" channel, but dynamically change channels if the link quality of the original channel becomes suboptimal.

WirelessUSB uses pseudo-noise (PN) codes to encode each information bit. Most WirelessUSB systems use two 32-chip PN codes allowing two information bits to be encoded in each 32-chip symbol. This scheme can correct up to three chip errors per symbol and can detect up to 10 chip errors per symbol. Although the use of 32-chip (and sometimes 64-chip) PN codes limits the data rate of WirelessUSB to 62.5 kbit/s, data integrity is much higher than Bluetooth, especially in noisy environments.

ZigBee
ZigBee has been designed as a standardized solution for sensor and control networks. Most ZigBee devices are extremely power-sensitive (thermostats, security sensors, etc.) with target battery life being measured in years.

ZigBee also uses a DSSS radio signal in the 868 MHz band (Europe), 915 MHz band (North America), and the 2.4 GHz ISM band (available worldwide). In the 2.4-GHz ISM band sixteen channels are defined; each channel occupies 3 MHz and channels are centered 5 MHz from each other, giving a 2-MHz gap between pairs of channels.

ZigBee uses an 11-chip PN code, with 4 information bits encoded into each symbol giving it a maximum data rate of 128 Kbps. The physical and MAC layers are defined by the IEEE 802.15.4 Working Group and share many of the same design characteristics as the IEEE 802.11b standard.

2.4-GHz Cordless Phones
2.4 GHz cordless phones are becoming increasingly popular in North America and do not use a standard networking technology. Some phones use DSSS; most use FHSS. The phones using DSSS and other fixed channel algorithms typically have a "channel" button on the phone allowing users to manually change the channel. FHSS phones do not have a "channel" button, because they are constantly changing channels. Most 2.4 GHz cordless phones use a channel width of 5 to 10 MHz.

Collision Avoidance
Along with understanding how each of the technologies work, it is also important to understand how each technology interacts in homogeneous and heterogeneous environments.

Wi-Fi's collision-avoidance algorithm listens for a quiet channel before transmitting. This allows multiple Wi-Fi clients to efficiently communicate with a single Wi-Fi access point. If the Wi-Fi channel is noisy the Wi-Fi device does a random back off before listening to the channel again. If the channel is still noisy the process is repeated until the channel becomes quiet; once the channel is quiet the Wi-Fi device will begin its transmission. If the channel never becomes quiet the Wi-Fi device may search for other available access points on another channel.

Wi-Fi networks using the same or overlapping channels will co-exist due to the collision avoidance algorithm, but the throughput of each network will be reduced. If multiple networks are used in the same area it is best to use non-overlapping channels such as channels 1, 6, and 11. This allows each network to maximize its throughput since it will not have to share the bandwidth with another network.

Interference from Bluetooth is minimal due to the hopping nature of the Bluetooth transmission. If a Bluetooth device transmits on a frequency that overlaps the Wi-Fi channel while a Wi-Fi device is doing a "listen before transmit", the Wi-Fi device will do a random back off during which time the Bluetooth device will hop to a non-overlapping channel allowing the Wi-Fi device to begin its transmission.

Interference from 2.4 GHz cordless phones can completely stop a Wi-Fi network, even if the cordless phones use FHSS as opposed to DSSS. This is partially due to the wider channel (5 to 10 MHz) compared to Bluetooth (1 MHz) and also due to the higher power of the cordless phone signal. An FHSS cordless phone that hops into the middle of a Wi-Fi channel can corrupt the Wi-Fi transmission, causing the Wi-Fi device to repeat its transmission. 2.4 GHz FHSS cordless phones will most likely cause interference with all Wi-Fi devices in close proximity; therefore, these phones are not recommended for use around Wi-Fi networks. If the cordless phone is DSSS the channels used by the cordless phone and Wi-Fi access point may be configured to not overlap, thus eliminating interference.

Handling Interference in Bluetooth

In Bluetooth, interference from other Bluetooth piconets is minimal, because each piconet uses its own pseudo-random frequency-hopping pattern. If two co-located piconets are active the probability of a collision is 1/79. The probability of a collision increases linearly with the number of co-located active piconets.

Bluetooth originally relied on its frequency-hopping algorithm to handle interference, but people have realized that a single active Wi-Fi network can cause heavy interference on 25% of the Bluetooth channels. Packets lost due to overlap have to be retransmitted on quiet channels, thereby greatly reducing the throughput of Bluetooth devices.

Bluetooth specification version 1.2 addresses this issue by defining an adaptive frequency hopping (AFH) algorithm. This algorithm allows Bluetooth devices to mark channels as good, bad, or unknown. Bad channels in the frequency-hopping pattern are then replaced with good channels via a look-up table. The Bluetooth master may periodically listen on

bad channels to determine if the interference has disappeared; if so, the channel is then marked as a good channel and removed from the look-up table. Bluetooth slaves, when requested by the master, can also send a report to the master informing the master of the slave's assessment of channel quality. For instance, the slave may be able to hear a Wi-Fi network the master cannot. The Federal Communications Commission (FCC) requires at least fifteen different channels be used.

The AFH algorithm allows Bluetooth to avoid channels occupied by DSSS systems such as Wi-Fi and WirelessUSB. 2.4 GHz FHSS cordless phones may still cause interference with Bluetooth since both systems are hopping over the entire 2.4 GHz ISM band, but since the Bluetooth signal is only 1 MHz wide the frequency of collisions between the FHSS cordless phone and Bluetooth is significantly less than the frequency of collisions between Wi-Fi and FHSS cordless phones.

Bluetooth also has three different packet lengths that translate into different dwell times on a given channel. Bluetooth has the option to reduce the packet length in an effort to increase data throughput reliability. In this scenario it is better to get smaller packets through at a slower data rate than losing larger packets at the normal data rate.

Handling Interference in WirelessUSB, ZigBee
In WirelessUSB, each network checks for other WirelessUSB networks before selecting a channel. Therefore interference from other WirelessUSB networks is minimal. WirelessUSB checks the noise level of the channel at least once every 50 ms. Interference from a Wi-Fi device will cause consecutive high noise readings causing the WirelessUSB master to select a new channel. WirelessUSB peacefully co-exists with multiple Wi-Fi networks, because WirelessUSB is able to find the quiet channels between the Wi-Fi networks **(Figure 2)**.

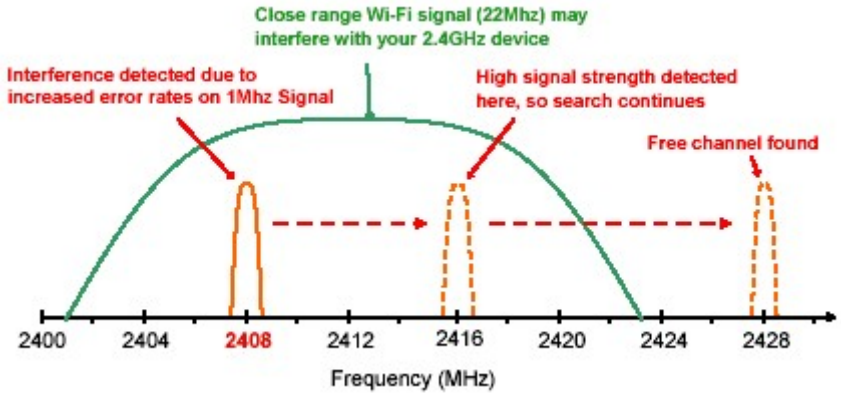

Figure 2: Diagram showing the frequency agility of a WirelessUSB design.

Interference from Bluetooth may cause WirelessUSB packets to be retransmitted. Due to the hopping nature of Bluetooth, WirelessUSB retransmissions will not collide with the next Bluetooth transmission because the Bluetooth device will have moved on to a

different channel. Bluetooth networks will not cause enough consecutive high noise readings for the WirelessUSB master to change channels.

ZigBee specifies a collision-avoidance algorithm similar to 802.11b; each device listens to the channel before transmitting in order to minimize the frequency of collisions between ZigBee devices. ZigBee does not change channels during heavy interference; instead, it relies upon its low duty cycle and collision-avoidance algorithms to minimize data loss caused by collisions. If ZigBee uses a channel that overlaps a heavily used Wi-Fi channel field tests have shown that up to 20% of all ZigBee packets will be retransmitted due to packet collisions.

What Can Be Done?

When developing Bluetooth, Wi-Fi, or ZigBee, designers must use the methods provided in the specification. When developing a proprietary system based on 802.15.4, WirelessUSB or other 2.4 GHz radio, designers can use lower-level tools to create frequency agility.

DSSS systems have the most to lose because of the danger of overlapping with another DSSS system. But there are things DSSS systems can do to obtain the frequency agility of FHSS systems. One approach is network monitoring. If the DSSS system uses a polled protocol (where packets are expected at specified intervals) then the master can switch channels after a number of failed transmit attempts or bad received packets.

Another approach is to take a reading of the energy level on the air if the radio has this capability. A receive strength signal indicator (RSSI) can be used to proactively measure the amount of energy on the air and if that level is too high over a period of time switch to a clearer channel. A period of time is taken into account so as not to change channels if a FHSS system is passing through.

Network monitoring and RSSI readings assume that both radios are transceivers — they can transmit and receive packets. In a DSSS system, in which one side is a transceiver and one side is a receiver, a multiple transmit approach can be used to obtain frequency agility. The transmitter sends the same packet at multiple frequencies and the receiver cycles though the receive channels at a much slower rate. This system works when the receiver is connected to power and the battery-powered transmitter is used less frequency. A wireless remote might use this approach.

Wrap Up

Each of the standard 2.4-GHz networking technologies has made design tradeoffs to mitigate the effects of interference or to avoid it altogether. Designers can create their systems to be frequency agile either by using the procedures provided by the standard being implemented or by building their own protocol using the methods mentioned here in conjunction with radio features like RSSI when available. While it will never be possible to completely eliminate interference from outside 2.4-GHz systems, designers can create their systems to be frequency agile and give their product the best chance to survive in today's hostile 2.4-GHz ISM band environment.

About the Authors

Mark Gerrior *is a principal software engineer in the Consumer and Computation Division of Cypress Semiconductor. He holds a MA in computer science from Marlboro College and can be reached at mgt@cypress.com.*

Ryan Winfield Woodings *is a systems engineer in the Consumer and Computation Division of Cypress Semiconductor. He holds an MS and BS in computer science from Brigham Young University and can be reached at rww@cypress.com.*