Integrating microcontroller and RF functions on-chip for wireless out-of-box experience

Wireless is a broad technology encompassing a multitude of applications with different costs and complexity. This article will focus on the integration of Bluetooth, wirelessUSB, and Zigbee radios into eight-bit microcontrollers and mixed-signal arrays to create low-power, low-cost, single-chip wireless solutions that enable interdevice communication without the need for cables.

By Deepak Sharma and Ryan Woodings

As Meg takes the turn to her street in her SUV, she pushes the button on her garage door opener and her garage door opens, the main hallway light comes on, the central alternating current (AC) kicks in to her desired level, the plasma TV comes on to her favorite channel and her automatic blinds come down to provide comfortable viewing. This may sound far fetched, but with today's wireless technologies this scenario is easily within the realm of possibility.

Wireless is a broad technology encompassing everything from extremely low-power, low-cost radio frequency identification (RFID) tags to wireless metropolitan area networks using WiMax and ultra-wideband (UWB) as illustrated in Figure 1. This article will focus on the integration of Bluetooth, wirelessUSB and Zigbee radios into eight-bit microcontrollers and mixed-signal arrays to create low-power, low-cost single-chip wireless solutions that enable interdevice communication without the need for cables.

While 802.11 has made its mark in wireless local area networking, the next revolution is occurring in the areas of wire replacement and sensor networks. Wire replacement covers short-range wireless applications such as keyboard, mouse, presenter tools, in-home light management systems, simple sensor networks, garage door openers, car key fobs, and personal computer (PC) headsets. The heart of these systems consists of a wireless radio (usually Bluetooth or wirelessUSB) and a microcontroller (usually eight bit) along with various external components on the board.

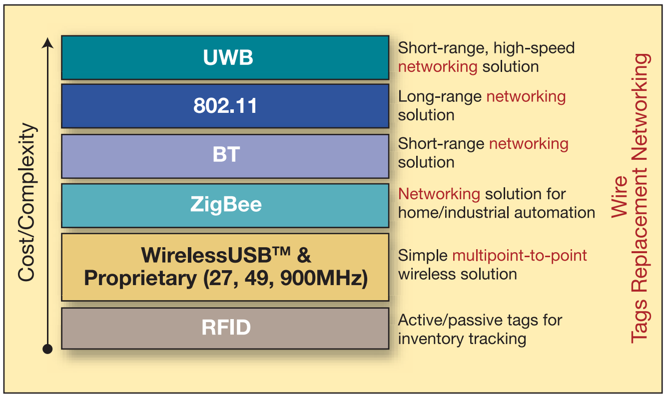

Figure 1. Wireless technologies by application vs. cost/complexity.

Wire replacement technologies

Sensor networks are becoming popular with the recent creation of Zigbee, which has been designed as a standardized solution for sensor and control networks. Zigbee has been optimized for mesh and cluster-tree networks that allow peer-to-peer communication. Most Zigbee devices are extremely power sensitive (thermostats, security sensors, etc.) with target battery life being measured in years. The focus of Bluetooth is ad-hoc interoperability between cell phones, headsets and personal digital assistants (PDAs). Bluetooth is also used as a cable replacement solution for wireless computer peripherals. Most PDAs and high-end cell phones today contain an integrated Bluetooth solution. WirelessUSB is built upon the premise that a complex networking protocol is not required for simple point-to-point applications like PC mice, keyboards, and basic sensor networks.

Bluetooth, wirelessUSB, and Zigbee all operate in the 2.4 GHz industrial scientific and medical (ISM) band. (Zigbee can also operate in the 900 MHz band.) There are several benefits to the 2.4 GHz ISM band compared to other license-free bands:

1. 2.4 GHz is an ISM band and is available worldwide; many other license-free bands are not consistent worldwide. While the end consumer may not care, it is a big advantage to consolidate supply chain logistics for manufacturers.

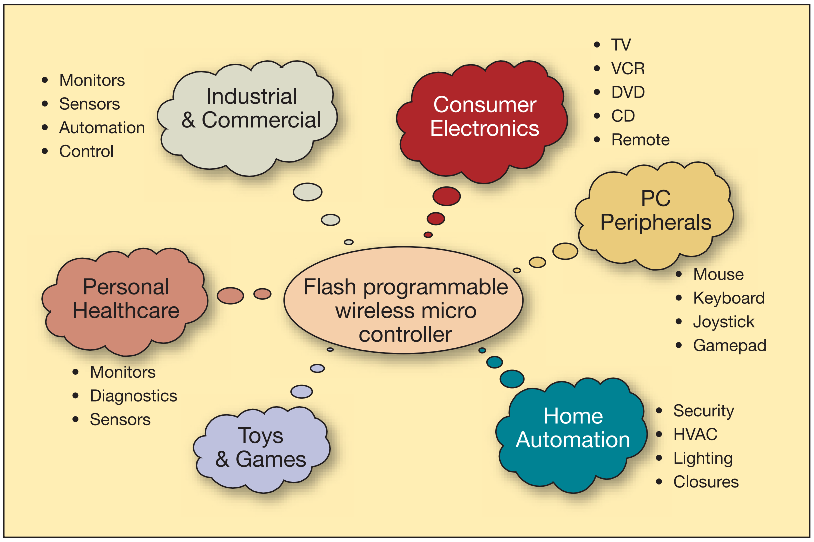

Figure 2. Explosion of applications for a wireless microcontroller.

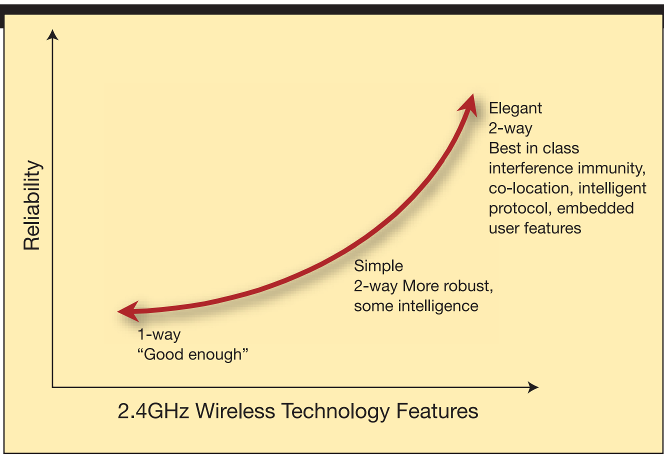

Figure 3. Reliability spectrum for wireless technologies.

2. High bandwidth—27 MHz systems suffer from bandwidth limitation and co-location issues. 2.4 GHz has several channels and combined with intelligent coding schemes can offer virtually unlimited co-location properties.
3. Spread spectrum—Devices using the 2.4 GHz band have to use a spread spectrum technique, either frequency-hopping spread spectrum (FHSS) similar to Bluetooth or direct-sequence spread spectrum (DSSS) similar to Wi-Fi, to enable high interference immunity and rejection. This results in highly robust solutions.
4. Low cost—With the mass adoption and ability of semiconductors to move down the technology curve, we are almost reaching a point where the cost delta between a 27 MHz system and 2.4 GHz system is negligible and the benefits far outweigh the slightly higher cost for designers and marketing groups.

Microcontroller integration

As microcontrollers become smaller and cheaper, the functionality of many external components is being integrated directly into the microcontroller. Eight-bit microcontrollers come in a variety of package sizes, random access memory (RAM) and read-only memory (ROM) sizes, serial communication buses, and analog inputs and outputs, enabling engineers to select a microcontroller that matches their design requirements and cost constraints. Some microcontrollers integrate the microcontroller and almost all related analog and digital peripheral circuits typically found in an embedded design, such as: analog to digital converter (ADC), digital to analog converter (DAC), pulse width modulation (PWM), amplifiers, timers, counters, universal asynchronous receiver-transmitter (UART), small computer serial interface (SCSI), SCSI parallel interface (SPI), intelligent interface controller (I^2C), and USB. This mixed signal integration allows customers to significantly reduce the number of components they have to use, greatly improving system quality and reliability and drastically lowering bill of materials. With this continued integration of external components into the microcontroller it was only a matter of time before the integration of reliable radio technology and advanced mixed-signal array microcontrollers occurred. The result of this integration is the potential to unlock value (read profits for manufacturers and ease of use for end consumers) across a myriad of applications in the consumer world as shown in Figure 2.

The issue now is figuring out the right wireless technology and microcontroller to integrate together. If the right microcontroller is merged with the right radio the resulting technology will enable designers to significantly decrease development time, component count and system cost while improving operating range, power consumption and latency. Besides benefits to the design engineer, the integration also simplifies supply chain logistics by significantly reducing the numbers of components in the device.

So what is the impact on end applications? The biggest impact is in two areas: ease of use and installation costs. Integration leads to a lower learning cycle and complexity level of implementation. By going 'wireless' you are able to carve out huge costs of installation. For example, using a wireless solution to set up carbon dioxide detectors in an existing building enables installation in days and does not require any breaking down of walls or expensive wiring.

However, you have to be cautious and intelligent about choosing the right solution. Let's begin with the wireless technology. The first step is deciding what kind of system you are building: is the system a high-end consumer electronic (light control system in a house) or a low-end commodity ($12 wireless mouse)? This would help in deciding between a one-way wireless protocol and a two-way wireless protocol. You could move along the reliability spectrum of wireless technology as shown in Figure 3.

Finally, the wireless protocol should be as simple as possible to enable an easy learning curve and implementation in a reasonably sized code space. You should also be evaluating intelligent binding schemes and security algorithms. (You don't want your garage door opener to be a gateway into your PC at home.)

Choosing a microcontroller

The next step is the microcontroller choice. The first thing is to find one that has integrated the wireless radio in it. Beyond that there are several factors to consider:

- **Microcontroller scalability.** Preferably you want to choose a family that allows you to scale both up and down in terms of flash size, inputs and outputs (I/Os), and various analog and digital components. It would be wise to choose a mixed signal array in case you are also looking at temperature sensing and voltage sequencing in your application. (For example, temp sensing in a server farm could trigger fans for the servers and at the same time send a wireless signal to a hub to record data for analytics at a later stage. The analytics could help in energy conservation and optimize power payments.)
- **Toolset integration.** The ideal scenario is to have the wireless radio be a user module/library that would simplify radio communications development in the design environment. The designer software development environment should be GUI based with simple point and click options. It should provide flexibility to code in either C or assembly language and use event triggers and multiple break points in debugging the design.
- **Design time reduction.** A nice to have feature is a toolset witha higher level of abstraction, enabling design at a level

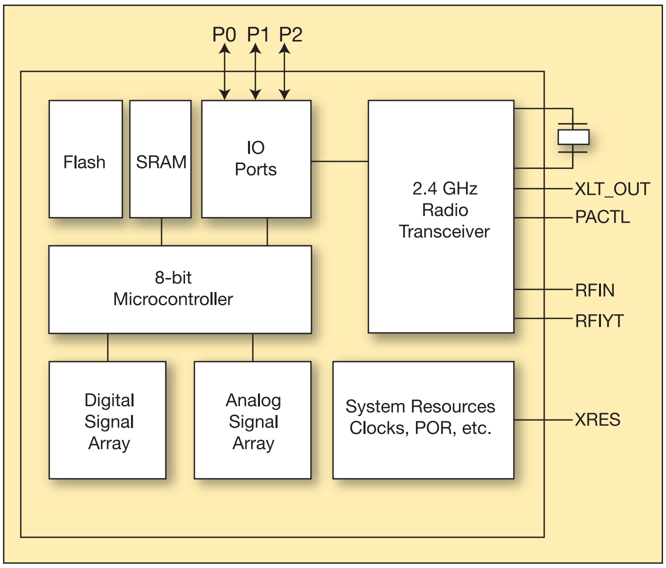

Figure 4. Block diagram of microcontroller with integrated radio.

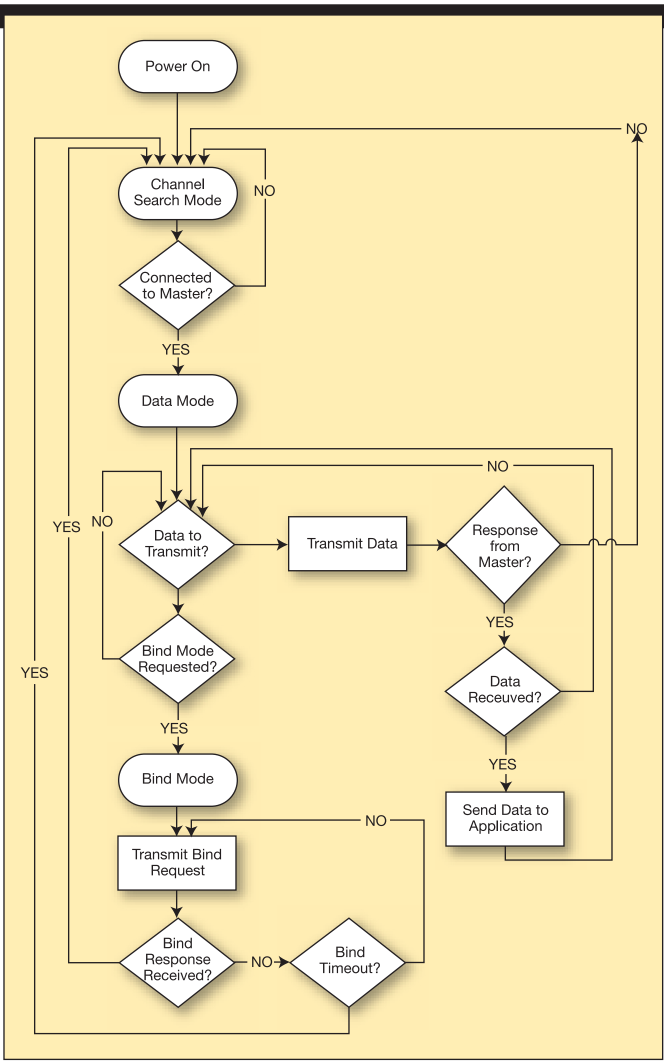

Figure 5. Example protocol state machine.

not requiring C or assembly language. This enables designers to focus on application expertise and create a custom solution using a system similar to choosing from a catalog and linking the parts in a logical manner. There would be an additional bonus if the tool could create datasheets, schematics and bill of materials to reduce total design cycle time to hours instead of weeks or months. Some of this may not appeal to the 'die-hard' engineers who prefer to hand code assembly and debug it using logic analyzers, but most modern engineers welcome tools that will reduce design time and increase the reliability of their product.

It is clear from a business and technology perspective that the time for integrating reliable 2.4 GHz radios with flexible mixed signal array microcontrollers is here. The rest of this article will explore an application example of a garage door opener and describe some of the real world design issues faced in implementing this using an integrated solution such as PRoC from Cypress Semiconductor Corp. vs. a discrete multichip solution.

Hardware

A discrete multichip solution typically requires additional external components in addition to separate microcontroller and radio chips. This adds additional size and cost to the design. By using an integrated solution, the design can be extremely small, consume less power, cost less, and take less time to develop. Figure 4 shows a block diagram of an eight-bit microcontroller with an integrated 2.4 GHz radio. Observe that due to the integration of the two chips the interface between the microcontroller and the radio is completely internal, which reduces the number of external pins required or frees external pins to become generic I/O pins instead of being dedicated to the radio interface.

Firmware

Integrated solutions are able to take advantage of the tight coupling between the microcontroller and the radio to create an easy-to-use firmware library for radio access. Some solutions even provide a complete protocol stack that provides a robust two-way link between devices. Depending on the target application the right protocol may be Bluetooth, wirelessUSB or Zigbee. By providing a complete protocol stack customized for the specific radio and microcontroller these solutions make it easy to create a connection between two or more devices. A simple API, such as the one shown below, is used to interact with the radio. After creating the connection the protocol will send packets to the target device, retransmitting the packets if an error is detected. If a connection is lost to the target device the protocol will re-establish the connection or find another route to the target device.

```
protocol_init()
protocol_create_connection()
protocol_send_packet(int packet_length, char* packet)
protocol_get_packet(int max_packet_length, char* packet)
```

Figure 5 shows an example of a protocol state machine that handles creating a wireless connection, and provides guaranteed packet delivery and interference immunity, without requiring additional design effort by engineers. This allows the engineers to treat the wireless link like a wired serial bus such as SPI, UART or I^2C.

System

With the integration of radio and microcontroller it is now trivial to create small wireless temperature sensors that can be placed in each room of the house, with each sensor periodically reporting its temperature to the main thermostat in order to more accurately control the heating and air conditioning in the home. A floodlight that lights up the driveway can now be connected to additional motion detectors covering the front walkway and side of the house so that the floodlight turns on when you walk out the door instead of requiring you to walk in darkness to the driveway before it turns on.

As Meg pulled into her garage, she smiled as she looked forward to watching her favorite show on her new plasma TV and wished that somehow her intelligent garage door opener would also heat up the TV dinner in a microwave and serve it on her table! Not a reality yet, but dream on since now is the time to bring those dreams to reality. RFD

ABOUT THE AUTHORS

Deepak Sharma is senior product marketing manager-wireless for Cypress. He joined the company in 1998 as a marketing engineer and has progressed over the years to manage several products. He managed PLDs, datacom and video products in the DCD division. In early 2005, he moved into the CCD division to take over the wireless portfolio. His expertise is in marketing and brand strategy for hi-tech products with experience in organizational and business strategy. He holds an M.B.A. from Rochester Institute of Technology (NY) and a B.Tech from B.I.T.S. (Pilani), India.

Ryan Woodings is a staff software engineer at Cypress Semiconductor, specializing in short-range wireless protocol design for the company's wirelessUSB products. He has six years experience in short-range wireless technologies including IrDA, Bluetooth and wirelessUSB. Woodings holds a B.S. and M.S. in Computer Science from Brigham Young University.