Preface

Duncan Glendinning
Director, MCG Mobile Communications Operation, Intel Corp.

Lin Chao
Editor
Intel Technology Journal

Intel's vision of "A Billion Connected Computers" has been the catalyst for a number of Intel-defined programs to connect computers. Wireless connectivity represents the next frontier for mobile computing, and Intel has been investigating the challenges of wireless connections for mobile PC's for some time now.. A key decision that had to be made was which radio technology should be built into mobile computers. There was no existing standard that gave the right performance, was available for worldwide usage, was extremely low powered, and was inexpensive enough to build into a broad range of devices. Consequently, Intel, together with our partners, set about creating a new standard. This resulted in Bluetooth* technology, a technology that allows peripherals to be connected wirelessly.

The need for anywhere anytime connectivity for mobile PC's is about to be answered. We are soon to enter a very exciting era for mobile computing. Over the next one to five years, we are going to see the emergence of new devices, applications, and usage models. The mobile PC as we know it today will evolve to where it PC will affect all types of users in all parts of the world.

The six papers in this issue of the Intel Technology Journal offer a lively discussion on Bluetooth and wireless connectivity. The first and second papers present the hardware and software architectural overviews. The third and fourth papers look at integration details and the issues and benefits of Always On, Always Connected computing.

The fifth and sixth papers address other wireless technologies and directions. The fifth paper specifically addresses security, one of the key issues surrounding wireless connections. If it is convenient for one individual to access business information, how do we prevent others from intercepting the information? The sixth paper looks into the evolution of cellular phone infrastructure as it moves towards being able to support significant data rates for computing applications.

Wireless Mobility and Communications

By Kevin Kahn
Intel Fellow, Director Communications Architecture
Intel Corp.

One of the biggest changes currently ongoing in computing and communications is the emerging importance of mobility. As PCs have become more ubiquitous, users have found it increasingly important to have access to them and the Internet when they are at places other than their desks.

Mobile phones, two-way pagers, PDAs, and other portable gadgets have appeared everywhere. All these devices depend upon the ability to communicate globally with the rest of the user's world and locally with each other. The result of this change is a dramatic growth in interest in wireless communications.

Depending upon the context in which users find themselves, wireless communications may mean connecting to wide area cellular systems, local area networks, or personal area mini nets. As a leader in both computing and communication issues and in their convergence, Intel has been very involved in leading developments for all of these forms of wireless communications. While most of this issue of the *Intel Technology Journal* is devoted to personal area connectivity as enabled by Bluetooth* devices, there are also articles describing some of the activities in third-generation cellular systems that will enable increased data rates and services to users everywhere the cellular phone system reaches and local area networks that will provide secure high-speed services to users in offices and homes.

Bluetooth technology provides a low cost infrastructure that will allow the various devices that a user carries to connect to each other and the various parts of the fixed environment. Cell phones can connect to PCs to deliver mail, PCs can connect to PDAs to exchange schedule and contact information, and PCs can connect to access points into the wired Internet. All this can occur without the pain of wires — critical when one is talking about small devices often carried in pockets and briefcases. By creating a common infrastructure, the user's personal web of devices can work together to make electronic life simpler and more productive.

In the broader space of business local area networking, 802.11 can provide a convenient alternative to wired networks for small businesses and a new level of flexibility for employees in large enterprises who want to be able to connect to the corporate Intranet from cafeterias and conference rooms. Finally, in the public wireless world, the next generation of cellular systems will lift data rates to levels that will make data and Web access from anywhere truly productive.

Taken together, these new wireless capabilities will fuel a revolution in how and where we access the Internet. We hope this issue of the *Intel Technology Journal* will help you understand how the pieces fit together to enable the revolution and how Intel is playing its part in driving this effort forward.

** Bluetooth is a trademark owned by its proprietor and used by Intel under license.*

Bluetooth* Architecture Overview

James Kardach, Mobile Computing Group, Intel Corporation

Index words: Bluetooth, Piconet, IEEE, 802.15, PAN, Wireless, CMOS Radio, Data Access Points, Cable Replacement, WLAN, Global, Frequency Hopping, SIG

ABSTRACT

The Bluetooth* wireless technology was created to solve a simple problem: replace the cables used on mobile devices with radio frequency waves. The technology encompasses a simple low-cost, low-power, global radio system for integration into mobile devices. Such devices can form a quick ad-hoc secure "piconet" and communicate among the connected devices. This technology creates many useful mobile usage models because the connections can occur while mobile devices are being carried in pockets and briefcases (therefore, there are no line-of-sight restrictions). This paper provides a brief description of some of these usage models and explains how the Bluetooth architecture is optimized to enable them. But first, let us answer the question why now.

Original Bluetooth market requirements dictated integration into small handheld devices (mobile phones and computers were key clients), low cost (longterm cost of under $5 per connection point), high security, low power, and ubiquitous global use of the technology. There was no single cellular technology that could meet the global use requirement (there are five wireless phone technologies in the US alone). While WLANs had good ad-hoc networking capabilities, there was no clear market standard to pick (there are at least three varieties of IEEE 802.11 standards and a variety of other proprietary solutions in the market). Moreover, cost was too high for integration; there were no global standards, and integration into small handheld devices (like mobile phones) was a problem. As such it was decided to take a different approach: replace the cable from the "Network Adapter" (WLAN card or cellular phone) with a low-cost RF link that we now call Bluetooth.

Today the Bluetooth technology is the only specification targeted at this new market of cable replacement. Even the IEEE organization has recognized the need for wireless cable replacement technology and started the development of the 802.15 working group that focuses on this market (they call it Wireless Personal Area Networks). This specification is based on the Bluetooth technology!

INTRODUCTION

The Bluetooth technology was developed to provide a wireless interconnect between small mobile devices and their peripherals. Target markets were the mobile computer, the mobile phone, small personal digital assistants and peripherals. These markets were represented by the companies who created the technology: Intel, 3COM, Ericsson, IBM, Motorola, Nokia, and Toshiba, and were further supported by the 1,600 other early adopter companies.

The goals of the technology did not include developing another Wireless Local Area Network (WLAN) technology, for which there were already many in the market and many more being developed. Rather, whereas WLANs are designed to efficiently connect large groups of people over a common backbone, the Bluetooth technology was designed to connect mobile devices over a personal and private connection (in essence, to replace the cables carried by many mobile travelers).

The Bluetooth technology tries to emulate the cost, security, and capabilities of common cables carried by mobile travelers. The technology must be as secure as a cable (supports application/link layer authorization, authentication, and encryption); must be manufactured for about the same cost as a cable (designed for eventual manufacture as single chip CMOS radio giving a long-term cost goal of $5 an endpoint radio); must connect to a variety of devices available to the mobile user (seven simultaneous connections) and support data rates that are

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

consistent with a mobile traveler's needs (1 Mega symbol per second data rates per piconet); must support many simultaneous and private connections (hundreds of private piconets within range of each other); must support the types of data used by mobile users (voice and data); and must be very low power and compact to support the small portable devices into which the technology will be integrated. Finally, the technology must be global as the mobile devices will travel and must work with devices found in other parts of the world.

USAGE MODEL

While the Bluetooth* usage model is based on connecting devices together, it is focused on three broad categories: *voice/data access points, peripheral interconnects,* and *Personal Area Networking (PAN).*

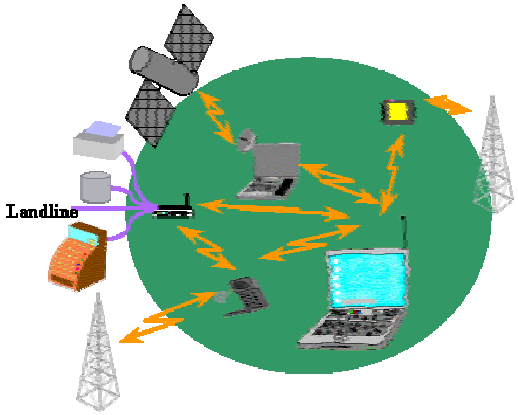

Figure 1: Voice/data access points

Voice/Data Access Points

Voice/data access points is one of the key initial usage models and involves connecting a computing device to a communicating device via a secure wireless link (see **Figure 1**). For example, a mobile computer equipped with Bluetooth technology could link to a mobile phone that uses Bluetooth technology to connect to the Internet to access e-mail. The mobile phone acts as a personal access point. Even more ideal, the notebook can connect to the Internet while the cell phone is being carried in a briefcase or purse. The Bluetooth usage model also envisions public data access points in the future. Imagine the current data-equipped pay phones in airports being upgraded with Bluetooth modems. This would allow any mobile device equipped with Bluetooth technology to easily connect to the Internet while located within ten meters of that access point. These access points could, of course, support much higher data rates than today's modems, as public spaces could connect a variety of private Bluetooth access points via a LAN that is routed to the Internet over a DSL line, allowing each access point a private 1Mbps connection to the Internet.

Figure 2: Peripheral interconnects

Peripheral Interconnects

The second category of uses, *peripheral interconnects*, involves connecting other devices together as shown in **Figure 2**. Imagine standard keyboards, mice, and joysticks that work over a wireless link. The Bluetooth link is built into the mobile computer; therefore, the cost of the peripheral device is less because an access point is not needed. Additionally, many of these devices can be used in multiple markets. For example, a Bluetooth headset used in the office could be connected to a Bluetooth access point that provides access to the office phone and multi-media functions of the mobile computer. When mobile, the same headset could be used to interface with the cellular phone (which can now remain in a briefcase or purse).

Another aspect of a short-range link like Bluetooth is in the area of proximity security devices. In this case, if one device is not within range of another device, the first device will go into a high security mode.

Figure 3: Personal Area Networking (PAN)

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

Personal Area Networking

The last usage model, *Personal Area Networking (PAN*), focuses on the ad-hoc formation and breakdown of personal networks (see **Figure 3**). Imagine meeting someone in an airport and quickly and securely exchanging documents by establishing a private piconet. In the future, Bluetooth kiosks could provide access to electronic media that could be quickly downloaded for later access on the mobile device.

THE DEVELOPMENT OF THE BLUETOOTH* TECHNOLOGY

The Bluetooth technology was developed by members of a Special Interest Group (SIG). The participating companies agree not to charge royalties on any Intellectual Property (IP) necessary to implement the technology. The SIG started initially with the promoters, who were the primary developers of the technology, and then expanded to include early adopters and adoptees.

Environment

The Bluetooth technology was developed to be used within a unique global environment that would not only enable integration into the host devices but would also allow the mobile device to travel easily from one country to another. In addition, due to the personal/confidential data contained on the different types of client devices (e.g., the mobile computer), the link formed between these devices needed to be as secure as the cable it was replacing.

BLUETOOTH ARCHITECTURE

The Bluetooth technology is divided into two specifications: the core and the profile specifications. The core specification discusses how the technology works, while the profile specification focuses on how to build interoperating devices using the core technologies. This paper deals with the core technology, as illustrated in **Figure 4,** and focuses on the lower layers of the Bluetooth architecture (up to the link manager).

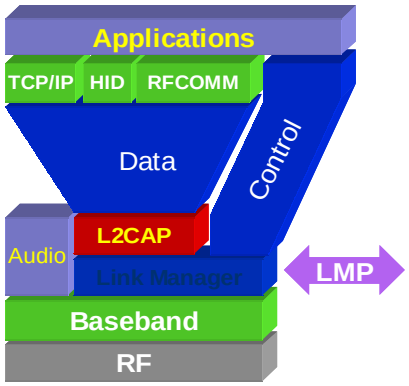

Figure 4: Bluetooth architecture

The Radio Frequency Layer

The Bluetooth air interface is based on a nominal antenna power of 0dBm (1mW) with extensions for operating at up to 20dBm (100mW) worldwide. The air interface complies with most country's ISM band rules up to 20dBm (America, Europe, and Japan). The radio uses Frequency Hopping to spread the energy across the ISM spectrum in 79 hops displaced by 1 MHz, starting at 2.402 GHz and stopping at 2.480 GHz. Currently, the SIG is working to harmonize this 79-channel radio to work globally and has instigated changes within Japan, Spain, and other countries.

The nominal link range is 10 centimeters to 10 meters, but can be extended to more than 100 meters by increasing the transmit power (using the 20dBm option).

The Bluetooth Baseband

As mentioned previously, the basic radio is a hybrid spread spectrum radio. Typically, the radio operates in a frequency-hopping manner in which the 2.4 GHz ISM band is broken into 79 one-MHz channels that the radio randomly hops through while transmitting and receiving data.

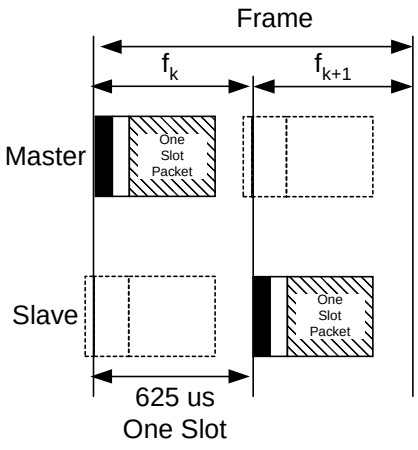

Figure 5: Single slot frame

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

A piconet is formed when one Bluetooth radio connects to another Bluetooth radio. Both radios then hop together through the 79 channels. The Bluetooth radio system supports a large number of piconets by providing each piconet with its own set of random hopping patterns. Occasionally, piconets will end up on the same channel. When this occurs, the radios will hop to a free channel and the data are retransmitted (if lost).

The Bluetooth frame consists of a transmit packet followed by a receive packet. Each packet can be composed of multiple slots (1, 3, or 5) of 625 us. A typical single slot frame is illustrated in **Figure 5**, which typically hops at 1,600 hops/second.

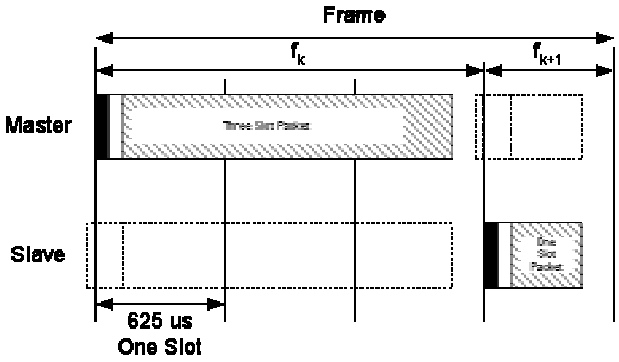

Figure 6: Multi-slot frame

Multi-slot frames allow higher data rates because of the elimination of the turn-around time between packets and the reduction in header overhead. For example, single slot packets can achieve a maximum data rate of 172 Kbits/second, while a 5 slot, 1 slot multi-slot frame will support a 721 Kbits/second rate (in the 5-slot direction) with a 57.6 Kbits/second rate back channel (in the 1-slot direction). A multi-slot frame is illustrated in **Figure 6.**

Network Topology

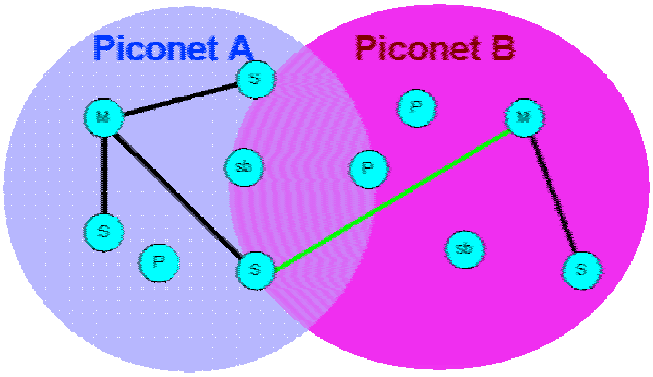

Figure 7: Network topology

Figure 7 illustrates a typical piconet with each small bubble (M, S, P, or Sb) representing a Bluetooth radio. Bluetooth radios connect to each other in piconets, which are formed by a master radio simultaneously connecting up to seven slave radios. The Bluetooth radios are symmetric in that any Bluetooth radio can become a master or slave radio, and the piconet configuration is determined at the time of formation. Typically, the connecting radio will become the master; however, a "master/slave swap" function allows the roles to be reversed. (A device can only be a master in one piconet though.)

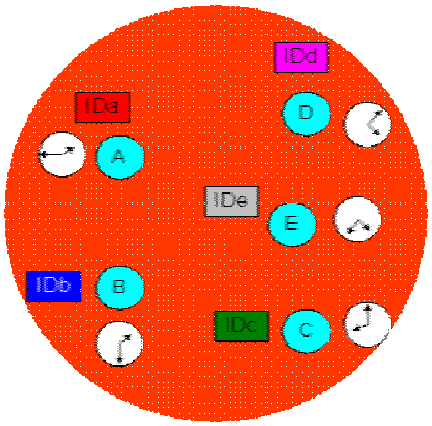

Figure 8: Bluetooth radios in the wild

To form a piconet, the Bluetooth radio needs to understand two parameters: the hopping pattern of the radio it wishes to connect to and the phase within that pattern. Bluetooth radios each have a unique "Global ID" that is used to create a hopping pattern. In forming a piconet, the master radio shares its Global ID with the other radios, which then become slaves and provide all the radios with the correct hopping pattern. The master also shares its clock offset (represented by the clock dial) with the slaves in the piconet, providing the offset into the hopping pattern. This information can easily be exchanged via the FHS packet.

Normally, radios not connected to the piconet exist in "Standby" mode. In this mode, the radios are listening for other radios to find them ("Inquire") and/or are listening for a request to form a piconet ("Page"). When a radio issues an Inquire command, listening radios will respond with their FHS packet (Global ID and clock offset), providing the inquiring radio with a list of Bluetooth radios in the area.

To form a piconet, a Bluetooth radio will page another radio with its Global ID (obtained by a previous inquiry). The paged radio will respond with its Global ID, and the master radio passes the paged radio an FHS packet. The paged radio then loads the paging radio's Global ID and clock offset, thus joining the master's piconet. **Figure 9** illustrates Radio A becoming the master to Radios B and C.

Once a radio joins a piconet, it is assigned a 3-bit Active Member Address (AMA) allowing other radios on the piconet to address it. Once the piconet has eight radios active, the master must then take a radio and "Park" it on the piconet. This radio stays coordinated with the piconet but releases its AMA for an 8-bit Passive Member Address (PMA). The freed AMA can now be assigned to another radio wishing to join the piconet. The combination of AMA and PMA allows over 256 radios to actively reside on a piconet, while only the eight radios with the AMAs can actively transfer data. This is also illustrated in **Figure 9**'s Radio D, which has loaded the master's Global ID and clock offset and is parked on the piconet (prepared to join the piconet when data are ready to be transferred).

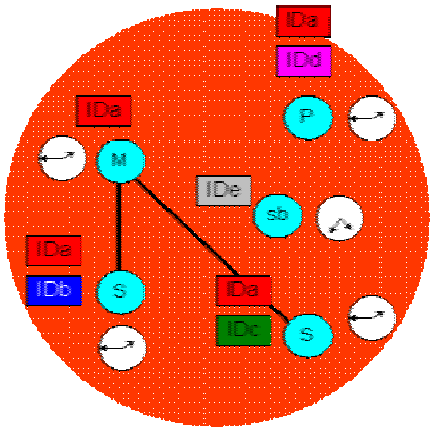

Figure 9: Bluetooth radios in a piconet

Parked radios listen at a beacon interval for information addressed to them. This allows a master to perform a broadcast to all slaves (parked and active).

Radios that are not actively connected to the piconet are in the Standby state (e.g., Radio E in **Figure 9**). These radios listen for Inquires or Pages from other radios. Every 1.25 seconds they will perform a Page Scan and/or an Inquiry Scan to see if such a request is being made.

The inquiry process involves one radio executing a page function on the Inquiry ID (a special global address set aside for the Inquire function), while other radios are performing an Inquiry Scan. This process is performed on a unique sequence of 32 channels. The radio doing an Inquire Scan will listen every 1.25 seconds on one of these 32 channels for 10 ms, then will repeat this scan on the next channel (within this 32-channel sequence). A radio with Inquire Scan enabled will continue this process until the Inquire Scan function is disabled. The inquiring radio will issue a number of pages on the Inquire channels (twice per single slot) and then listen at the corresponding response frequency (twice per slot) for 1.25 seconds for 16 of the 32 frequencies. The listening radio's correlator will fire if it is doing a Page Inquire on the same inquire channel as the inquiring radio and will respond with an FHS packet (containing its Global ID and clock offset). The sequence is then repeated for the second set of 16 frequencies after which the inquiring radio will have a list of FHS packets for all radios within range.

Paging follows a similar sequence. Each radio has a unique sequence of 32 paging frequencies and 32 response frequencies based on its Global ID. A radio in Standby mode doing a Page Scan will listen for a page of its Global ID on each of these paging frequencies for 10 ms, every 1.25 seconds going to the next paging frequency in the sequence. The paging radio will continuously page using the paged radios' Global ID on one of two sets of 16 frequencies within the paging radios' 32 paging frequencies. The paging radio makes an estimate (based on its last known clock offset) of where the paged radios should be listening and then creates an "A Train" of page frequencies around this estimated frequency. The paging radio will then continuously page across these 16 frequencies for 1.25 seconds. If the estimate was wrong (the paging radio received no response), the paging radio will next try the remaining 16 frequencies for the next 1.25 seconds. Radios that have little clock offset will be able to connect very quickly, while radios that have large clock offsets (meaning the radios haven't connected recently) could take up to a maximum of 2.5 seconds to connect (a complete A/B train search).

Once a radio has been found (via Inquiry) and then placed into a piconet (via Page), a piconet is formed and some useful work can now take place**.** **Figure 10,** entitled *Functional Overview*, depicts the different high-level states of a Bluetooth radio.

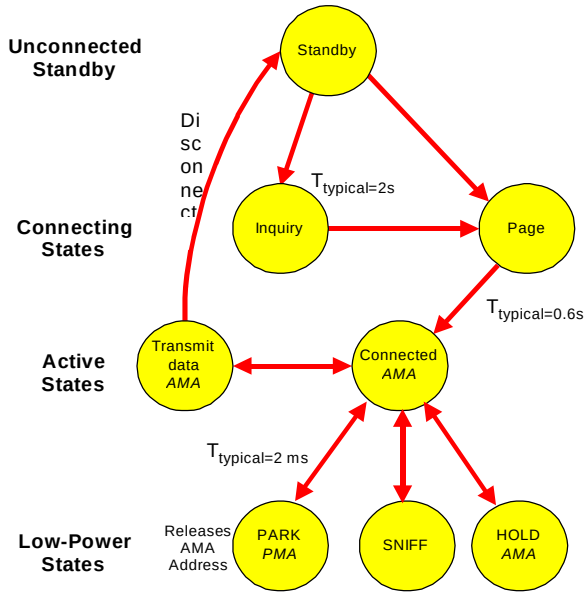

Figure 10: Functional overview

In the connected state, the Bluetooth radio is assigned a 3-bit Active Member Address (AMA) for which it can then direct data to different devices on the piconet (master is always referenced as address 0). Broadcasts to other radios on this piconet can be accomplished by the master sending a packet to address 0. To enable radios to maintain a connected state with the piconet (maintain the piconets hopping pattern and offset) while maintaining a very low-power state, radios can be placed in the Park, Hold, and SNIFF states. For the Hold and Sniff states, the radios are told to wake up at given intervals (go away for x slots); however, in the Sniff state the radio can transfer data on that interval (for example, a keyboard might be told to send/receive data every 20 slots), while in the Hold state no data are transferred. In the Park state, the radio is told to go away and is given the PMA address. A Parked radio will listen on a Beacon interval to see if the master has a) asked the parked device to become an active member, b) asked if any parked device wishes to become an active member, or c) sent any broadcast data.

When in the connected state, the Bluetooth radios can issue two types of packets: a Synchronous Connection Oriented (SCO) type or an Asynchronous Connectionless Type (ACL). The SCO type is associated with isochronous data, and, to date, this is voice. This is typically a symmetrical packet of 1, 2, or 3 slots, and the frames are reserved whether they are used or not within the piconet. In order to have an SCO connection, the radio must have already established an ACL connection. Once an SCO link has been added, a master or slave unit may send SCO packets without being polled. Currently, the voice data links use a CVSD coding that provides very good noise immunity and a high-quality voice link. The CVSD coding enables damaged SCO packets to be thrown away (versus retransmitted) while maintaining a high-quality voice link. One baseband packet type allows both voice and data to be sent in the same packet (DV packet).

The ACL link is packet-oriented and supports both symmetric and asymmetric traffic. As mentioned previously, ACL packets are created with an odd number of slots such that the frame is always an even number of slots (1/1, 1/3, or 1/5, for example).

There are three error correction schemes used in the Bluetooth Baseband: 1/3 rate FEC, 2/3 rate FEC, and Automatic repeat request (ARQ). 1/3 FEC is always applied to the packet header information. To increase the data rate when the link gets noisy, the radio can start adding FEC to the channel: for SCO links, a 1/3 FEC is applied while for ACL links, a 2/3 FEC is applied. As mentioned previously, SCO packets are thrown away when damaged. However, for ACL packets, one packet is directly acknowledged by the recipient in the next packet.

SECURITY

The way that the Bluetooth* radio system is used in mobile devices and the type of data carried on these devices (e.g., a corporate mobile computer) makes security an extremely important factor. While most wireless systems will claim that being a spread spectrum radio provides security, the volumes projected for Bluetooth radios eliminate this barrier. As such, link layer and application layer security are part of the basic Bluetooth radio requirements.

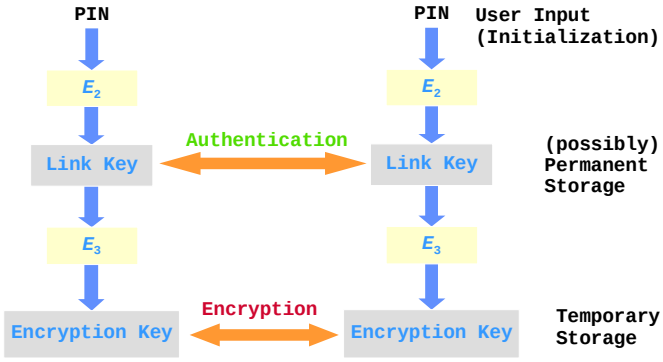

Figure 11: Link layer security architecture

At a link layer, the Bluetooth radio system provides Authentication, Encryption, and Key Management of the various keys involved. Authentication involves the user providing a Personal Identification Number (PIN) that is translated into a 128-bit link key that can be authenticated in a one or two-way direction. Once the radios are authenticated, the link can be encrypted at various key lengths (up to 128-bits in 8-bit key increments). The link layer security architecture provides a number of authentication schemes and a flexible encryption scheme that allows radios to negotiate for key length. This is important, as radios from different countries will be talking to each other. Security policies in these countries will dictate maximum encryption key lengths. Bluetooth radios will negotiate to the smallest common key length for the link (for example, if a USA radio is enabled for a 128-bit encryption key and a Spanish radio is enabled for only a 48-bit encryption key, the radios will negotiate a link with 48-bit encryption key). The Bluetooth architecture also supports authorization of different services to upper software stacks. For example, when two computers have created a Bluetooth link to exchange business cards, authorization must be created to extend these services (such that one computer could not examine other services on that computer unless enabled to do so).

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

The Bluetooth security architecture relies on PIN codes for establishing trusted relationships between devices. While not practical to go through all the combinations of uses of PIN codes, it should be noted that once a trusted pairing is established between devices, these codes can be stored within the device to allow more automatic/simple connections. The key to Bluetooth simplicity will be establishing the trusted relationship between commonly used devices. For random ad-hoc connections that require authenticated connections (such as ensuring you are connecting to who think you are connecting to, something that is not always obvious with invisible radio waves), PINs would have to be exchanged (depending on how the devices are configured).

CONCLUSION

Bluetooth* is a radio system designed for connecting a variety of mobile devices in a secure ad-hoc fashion. Much thought has gone into developing a radio system that provides interoperability between different device types while also meeting the requirements of mobile users. This paper covered a small aspect of the Bluetooth radio system, the lower layers of the Bluetooth radio stack.

ACKNOWLEDGMENTS

Thank you to the teams and companies that worked to develop this radio system in record time. These companies include, but are not limited to, Intel Corporation, Ericsson, IBM, Motorola, Nokia, and Toshiba. Thanks to Laura Mariani for editing this article and Robert Hunter for reviewing it.

REFERENCES

[1] Bluetooth Specifications, Bluetooth SIG at http://www.bluetooth.com/.

AUTHOR'S BIOGRAPHY

Jim Kardach is a Principal Engineer with the Mobile Communications Group (MCG) at Intel Corporation. He earned a B.S. degree in EE from Fresno State University and has been employed by Intel for 14 years. Jim is currently the chairman of the Bluetooth Special Interest Group, which is now developing extensions to the first-generation Bluetooth radio system. Jim holds over 35 patents in the areas of computer systems, power management, and communications and has most recently lead the developments of the ACPI and Bluetooth technologies. His e-mail is jim.kardach@intel.com.

---

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

Architectural Overview of Intel's Bluetooth Software Stack

Kris Fleming, Mobile Computing Group, Intel Corporation
Uma Gadamsetty, Mobile Computing Group, Intel Corporation
Robert J Hunter, Mobile Computing Group, Intel Corporation
Srikanth Kambhatla, Mobile Computing Group, Intel Corporation
Sridhar Rajagopal, Mobile Computing Group, Intel Corporation
Sundaram Ramakesavan, Mobile Computing Group, Intel Corporation

Index words: Bluetooth, mobile computing, WDM, Windows

ABSTRACT

Bluetooth* wireless technology was created to enable many different usage models from networking to cable replacement. Implementation of these usage models and of issues specific to Bluetooth, such as service discovery and security, involve mapping the Bluetooth protocol stack into operating systems' frameworks in order to ensure seamless integration. This paper describes the architecture of the software stack implemented at Intel to support the usage models on Microsoft's* operating systems based on WDM* technology.

The Bluetooth Special Interest Group (SIG) was launched in May 1998. Its goal was to develop the specifications for a low-powered, short range, RF-based wireless communication technology. When added on to a notebook computer, a Bluetooth module enables the notebook computer to talk wirelessly to other Bluetooth devices including cellular phones, PDAs, headsets, access points, and other notebook computers

An implementation based on the architecture described in this paper is available from Intel and was demonstrated at the December 1999 LA Bluetooth Developer's Conference and the Spring 2000 Intel Developer's Forum.

INTRODUCTION

Imagine a person with a PDA walking into an office and automatically waking up the notebook computer as a reaction. Further imagine that the notebook computer automatically synchronizes with the PDA for any calendar and task information entered overnight. When reminded of a meeting, the notebook computer is taken to a nearby conference room for a presentation while maintaining Intranet connectivity using a LAN access point. In the conference room business cards are exchanged electronically amongst the participants using notebook computers, PDAs, and cellular phones. All of this can be achieved with Bluetooth software support in the notebook computers. Bluetooth technology promotes the *Always On, Always Connected* and *Anything, Anywhere, Anytime* vision of seamless mobile computing for mobile personal computers (PCs) [1].

Goals of the Paper

This paper presents an overview of the architecture for Intel's Bluetooth software stack on a PC[1]. It attempts to provide answers to the following questions:

- What does the Intel stack do?
- How do third-party Bluetooth application developers write applications specific to Bluetooth technology?
- How can third-party Bluetooth hardware or device developers develop custom drivers for their hardware that work with the Intel stack?
- What is expected from a Bluetooth device so that it interoperates with the Intel stack?

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

* Other brands and names are the property of their respective owners.

[1] All information contained in this paper relating to Intel's Bluetooth software product and future plans are subject to change without notice.

This paper can also serve as an example for similar development efforts in other operating systems.

Bluetooth Usage Models and Architecture Goals

The Bluetooth Specification [4, 5] addresses a variety of usage models for Bluetooth devices. Intel's implementation of the Bluetooth stack either directly enables or provides the infrastructure for supporting the following Bluetooth SIG 1.0 usage models in its 1.0 release:

- *Object transfer*. This is an ability to transfer an object from one device to another. Objects include files and directories.
- *Data access points*. Access points enable access to the Internet, e-mail, and fax for these types of wireless connections for a notebook computer with Bluetooth: a) to a PSTN plug, b) to a cellular phone, and c) to a LAN access point.
- *Synchronization*. This involves synchronization of business cards, calendars, and task information between Bluetooth devices.
- *Ultimate Headset*. A Bluetooth headset can be wirelessly connected to a PC and appear as an audio playback or recording device.

In addition to the SIG usage models mentioned above, additional goals of Intel's implementation of the Bluetooth stack include the following:

- *Support for legacy PC applications*. There are a variety of legacy comm port applications for file transfer in the market today such as ProComm Plus* and Intellisync*. Supporting these legacy applications for achieving wireless file transfer between PCs enabled with Bluetooth technology would be convenient for end users.
- *Support for applications native to the OS*. This is part of Intel's goal of seamless integration into the OS. When a Bluetooth device comes in range of a notebook computer, this support enables the applications that are already in the OS (such as dial-up networking and direct cable connect) to be able to work with the new device.
- *Support for APIs native to the OS*. This is the other goal of proper integration into the OS: enabling APIs native to the OS wherever possible to access Bluetooth devices. One example is the support for Win32 Comm* API for accessing Bluetooth serial ports.
- *Support for third-party development*. Part of the goal of industry standard implementation is to enable third parties to add value to the software stack without re-implementing the entire stack. To accomplish this goal, Intel plans to publish a user-mode API interface for functionality specific to Bluetooth, which is not already covered in APIs native to the OS, and a new kernel-mode API. The user-mode API is for development of user-mode applications for Bluetooth technology, while the kernel-mode API is for custom drivers for Bluetooth devices.

Bluetooth Hardware Overview

The Bluetooth specification defines a low-power short-range radio and protocol stack supporting a *personal communication bubble*. The radio has an operational range of 10 meters at 0 dbm and operates in the unlicensed 2.4 GHz ISM frequency band. A single connection supports a maximum asymmetric data transfer rate of 721 kbits/s or a maximum of three voice channels. The voice channels use synchronous communication and support mono-audio using a 64 kbit/s CVSD-encoded stream. All connections use a Time-Division Duplex (TDD) scheme to support bi-directional communication.

At the start of any connection, the unit initiating the connection is assigned temporarily as a *master*. This assignment is valid only during this connection. The master may have active connections to up to seven other units, known as *slaves*. The configuration of a master unit connected to one or more slave units is called a *piconet*. Link management mechanisms allow radio units to time-division-multiplex between master and slave, allowing them to act as bridges between piconets, forming a *scatternet*. Mechanisms also exist to allow both masters and slaves to request new connections and accept new connections. The goal of the radio specification is to enable multiple *virtual cables* rather than a single cable-replacement capability.

Spread spectrum technology is used to operate in noisy environments and to allow multiple piconets to co-exist without a noticeable loss in throughput. A frequency-hopping scheme, with a rate of up to 1600 hops per second over 79 one MHz channels, is used to spread the signal over the entire available ISM spectrum.

Various error correction schemes are available to support reliable channels. Packets may be protected using 1/3 and 2/3 rate forward error correction (FEC) schemes to help correct bit errors caused by weak signals near the range

* Other brands and names are the property of their respective owners.

limit. An automatic repeat request (ARQ) scheme helps to reliably transmit link-level frames.

The radio's link-level protocol supports both authentication and privacy, allowing users to develop a domain of trust between their personal devices. Connections may require a one-way, two-way, or no authentication. Authentication is based on a challenge-response algorithm. Encryption is used to protect the privacy of the connection. A stream cipher well suited to a silicon implementation is used with secret key lengths of up to 128 bits (selectable via 8-bit granularity). The authentication and privacy support are appropriate for the short-range nature of the radio, and applications requiring stalwart protection are encouraged to implement stronger security mechanisms.

Bluetooth Protocol Overview

Bluetooth specifications include protocols [4] and profiles [5]. Protocols specify the workings of an individual component (RFCOMM or L2CAP, for example), while the profiles specify how a set of protocols can be used for implementing a particular usage model. Profiles are important to have a common understanding of the protocol stack in order to promote interoperability of usage model implementations.

The Link Management Protocol (LMP), baseband, and radio are typically implemented in the Bluetooth hardware *modules*. These modules can interface to the host using different interfaces. However, all Bluetooth controllers should implement the Bluetooth Host Controller Interface (HCI).

The Logical Link Control and Adaptation Protocol (L2CAP) implements a second link-layer protocol to address protocol multiplexing, segmentation, and re-assembly. L2CAP hosts a set of client protocols. A couple of such protocols are the Service Discovery Protocol (SDP) and a serial cable emulation protocol called RFCOMM. Figure 1 summarizes the Bluetooth protocol stack.

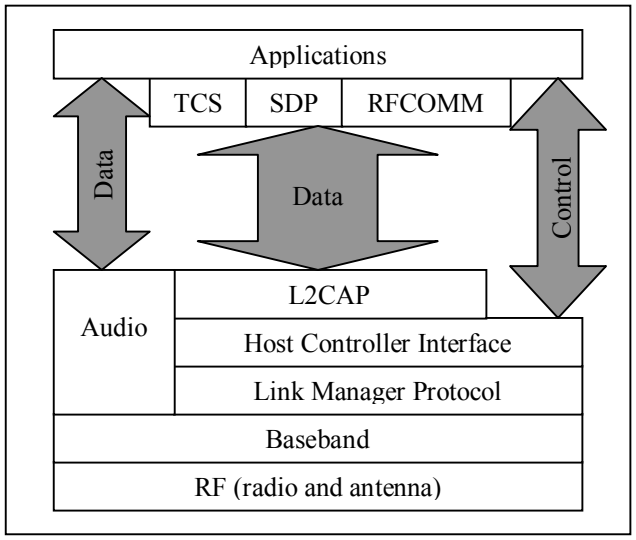

Figure 1: Summary of Bluetooth protocol stack

The L2CAP is a core component of the stack that contributes to the overall throughput that can be achieved.

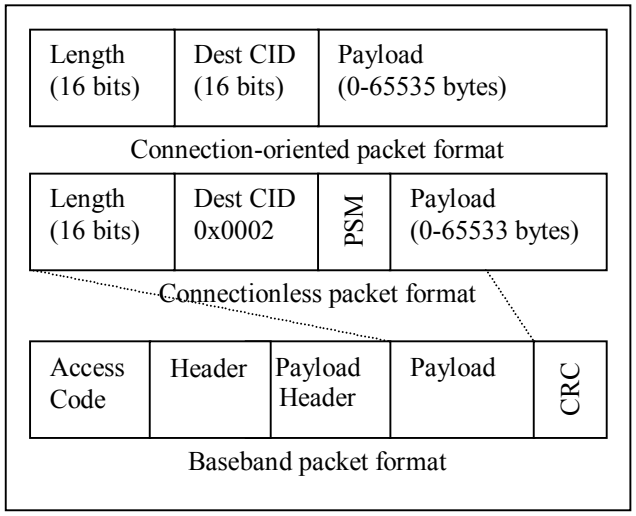

Figure 2: L2CAP and Baseband packet formats

The L2CAP supports both connection-oriented and connectionless packet formats as shown in Figure 2. The L2CAP packet becomes part of the Baseband packet payload as indicated. The L2CAP supports segmentation and re-assembly, protocol multiplexing, group abstraction, and quality of service.

A more complete description of the protocols and profiles can be found in the Bluetooth specifications.

A Note on Implementation Target

The target operating systems are WDM operating systems. For Release 1.0 implementation of the architecture

described here, we are targeting Windows98*, Windows98SE* and Windows2000*. The primary hardware target is notebook computers with embedded Bluetooth USB [2] modules, although appropriate hardware could enable the software stack on desktops also.

FUNDAMENTAL DESIGN

The architecture is based on the notion of treating the radio interface as a logical bus in the PC. Devices dynamically come into or go out of communication range of the Bluetooth module built into the notebook computer. This is analogous to hardware being plugged into or taken out of a physical bus inside the PC. The differences are in terms of the frequency with which Bluetooth devices transition into or out of communication range, and the large number (potentially) of Bluetooth devices that get *virtually* plugged into the PC when compared to a physical bus.

The responsibility of enumerating devices that are in range of the Bluetooth bus rests with an *RF Bus Driver* (RFBD). In addition to WDM bus driver functionality, RFBD processes the results of the *inquiry* process for discovering Bluetooth devices in range: the user drives this inquiry process. This is followed by discovery of services within these devices using Bluetooth SDP. The protocol part of SDP is implemented by RFBD. The top-level searches are handled in the Bluetooth Executive (described in the next section).

The Bluetooth usage models are supported by RFBD loading the appropriate client drivers based on device and service information obtained above.

Custom drivers can be loaded in response to devices discovery using Bluetooth *Plug and Play* [7]. This provides a framework for third parties to leverage unique features in their devices while reusing the Intel stack.

IMPLEMENTING THE BLUETOOTH PROTOCOL STACK

RFBD is at the core of Intel's Bluetooth software stack as shown in Figure 3. It co-exists with an *HCI driver* in the same binary. The HCI driver is responsible for all interactions with the Bluetooth *Host Controller* present on the Bluetooth module interfacing to the PC. It also serves to abstract the Bluetooth hardware interface in terms of a private interface exposed by it.

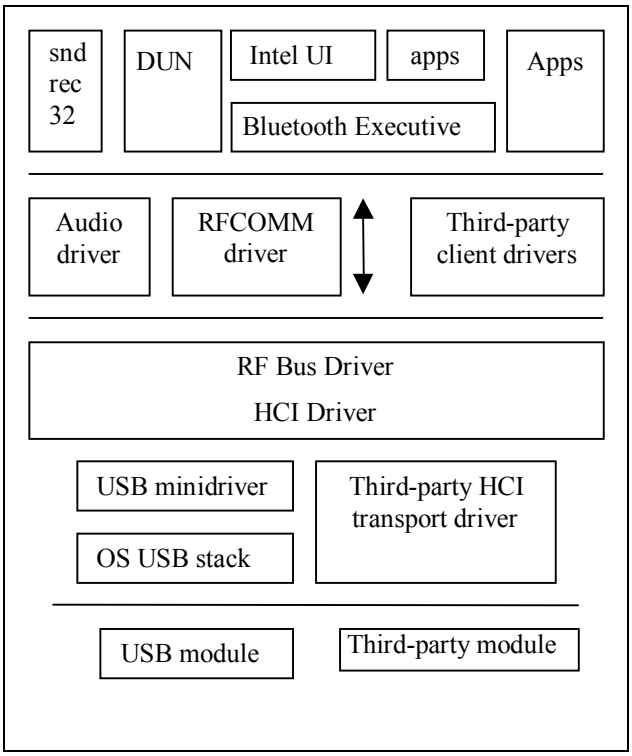

Figure 3: Architecture of Intel's Bluetooth stack

An *HCI transport driver* is typically layered under the HCI driver. It has the responsibility of handling the interface used to connect the Bluetooth hardware module to the PC. Figure 3 shows two such HCI transport drivers for the purposes of exposition: the USB driver stack and the PC Card driver. It is important to note that only one HCI transport driver is to be active at any given time in the stack[2].

In the case of the USB transport, the Bluetooth module interfaces with the PC using USB as shown in Figure 4 below.

* Other brands and names are the property of their respective owners.

[2] While multiple Bluetooth radios can physically be present in the same notebook computer, multiple such radios within a meter of each other are limited by RF physics on how well they work together.

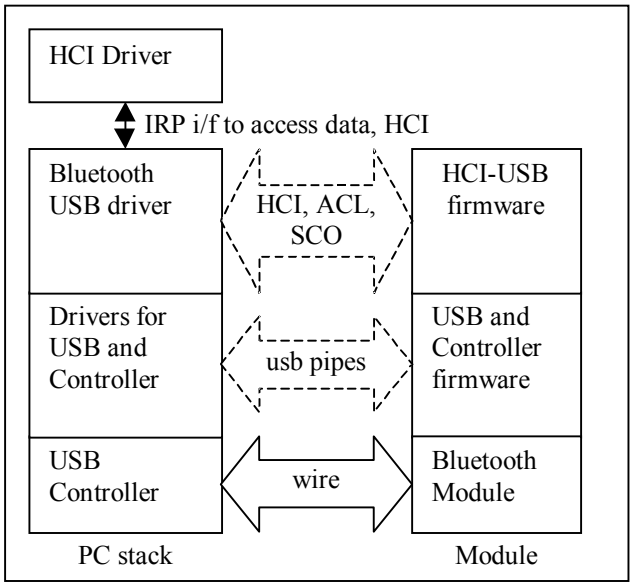

Figure 4: USB interfaces to access HCI and data

The HCI commands are transported using the USB control pipe, HCI events are transported on the USB interrupt pipe, asynchronous data are transported as USB bulk data, and synchronous data are transported over USB isochronous pipes as specified in the HCI USB specification. The Bluetooth USB minidriver on the operating systems' USB stack is responsible for abstracting the HCI transport specifics and presenting the data, events, and commands to the HCI driver in a transport-independent manner. An IRP-based [6] interface is defined under the HCI driver for this purpose, and all HCI transport drivers use this interface to talk to the HCI driver. Having such a common interface enables support for different HCI transport drivers.

The Bluetooth drivers loaded by the RFBD (referred to henceforth as the *client drivers*) talk to the RFBD using a kernel-mode interface called the *RF Bus Driver Interface* (RFBDI). The RFBDI is similar to other bus driver interfaces published by Microsoft such as the USBDI* or the 1394BDI* [6].

Client drivers typically implement minidriver or port driver functionality for some class driver in the OS. This is important to ensure seamless integration into the OS. The class drivers expose interfaces to these drivers as domain-specific APIs in the user mode.

There is one key module in the user mode and it is called the Bluetooth *Executive* (BTEXEC). Among other tasks, the BTEXEC is responsible for exposing a user-mode interface that all user-mode applications can use to access Bluetooth functionality. This interface is called the Bluetooth *API* (BTAPI).

All Bluetooth UI components developed by Intel use BTAPI to interface with the driver stack. The idea is that the BTAPI should contain all the information typically needed from the kernel stack by third-party application developers.

There could be transport-specific, platform-specific, or vendor-specific drivers not covered here.

Kernel-Mode Client Drivers

Two client drivers that are needed for supporting the usage models mentioned in this paper are the RFCOMM driver and the audio driver, as shown in Figure 5 below.

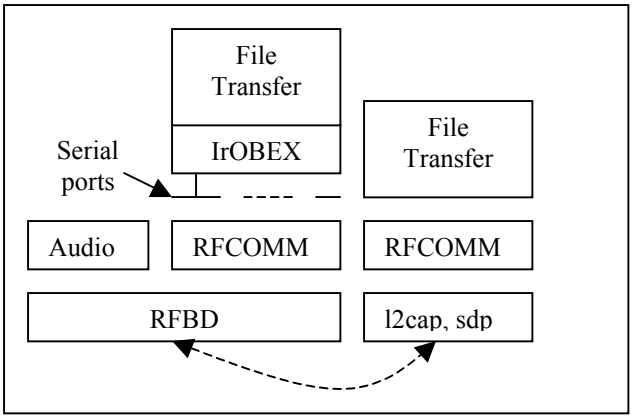

Figure 5: RFCOMM driver emulates serial ports

The RFCOMM driver is responsible for implementing multiple virtual comm ports over RFCOMM connections with each device supporting the protocol. These virtual comm ports are used to support both legacy comm port and IrOBEX-based applications for Bluetooth technology. The latter category includes file transfer and synchronization. RFCOMM is implemented as several drivers that together implement the RFCOMM protocol, comm port emulation functionality, and the interfaces to either VCOMM or serial.sys, depending on the OS.

In addition, PPP over L2CAP is the basis for dial-up networking and PPP-based LAN access points. These can also be supported by the RFCOMM driver set mentioned above.

The audio driver is responsible for exposing the headset as a playback and recording device to the system. It is also responsible for processing the audio stream coming over the air interface and piping it into the streaming class driver.

Programming Interfaces

RFBDI is an IRP-based kernel mode interface similar to USBDI or 1394 BDI interfaces published in the DDKs for

* Other brands and names are the property of their respective owners.

Windows2000*. It supports access to the following features:

- audio data stream
- connection-oriented data stream
- connectionless L2CAP interface
- connection-oriented event handling (examples of these events include those from L2CAP, SDP, and security)
- kernel-mode service discovery functions
- Bluetooth TCI
- L2CAP QoS

BTAPI provides access to features specific to Bluetooth technology in the kernel stack that typically cannot already be accessed through existing Windows* APIs. BTAPI provides access to the following:

- *Information relating to Bluetooth device discovery*. This includes access to local and remote device information and properties, and an ability to initiate device discovery.
- *Radio configuration and status*. This includes an ability to change duration and length of page and inquiry scans, access to RSSI, and an estimation of link quality.
- *Service discovery information*. This includes access to device services and attributes, and an ability to add or remove services on the local host.
- *Security information*. The user can set Bluetooth security modes for the local device, authorize use of a device in an ad-hoc manner, and communicate trust status with the stack.
- *Power management*. The user can set radio power states using BTAPI and control wake-up states in the software.

Applications use the Win32* Comm API for performing open, close, read, write, and flow control functions on the virtual comm ports exposed by the RFCOMM driver. In the case of the server, the application creates an SDP record for its service, which results in the creation of a comm port, opens the port, and "listens" on this port for incoming connections. On the client side, the PC inquires and searches for devices exporting this functionality using SDP, loads drivers to handle the service (this results in a comm port being created), opens the port, and connects to the server.

Usage of virtual comm ports for Bluetooth technology is characterized by the following features:

- Applications need to close and re-open the ports after each session.
- Link loss is indicated by dropping all communication signals low (this includes CD, CTS, DTR, DSR and RTS). The applications need to close the port on a link loss.
- Any reads and writes to the port will fail when the devices are not connected.
- BTAPI functions are needed for addition and deletion of SDP records, inquiry, and service discovery.
- There is no support for baud rate pacing; 5-, 6-, and 7-bit word lengths; parity and stop bits; and xon / xoff flow control.

INTEL UI AND END-USER EXPERIENCE

Intel's UI for controlling and configuring Bluetooth connections and devices is closely integrated into the OS UI components. Some of the UI elements are as follows:

- A Windows Explorer* extension for displaying and managing remote devices and connections. The Explorer extension also supports file transfer by drag and drop of the files to be transferred.
- A control panel applet specifically for managing the local Bluetooth device. It is also used for changing policies applicable to the local device.
- A system tray icon for displaying radio status, security-related pop-ups, and connection statistics.

As noted earlier, several native OS applications are enabled on Bluetooth devices, and (as described in the section on *open architecture* below) third parties have the ability to develop independent Bluetooth applications on the Intel stack.

A few usage scenarios are described below illustrating different aspects of the end-user experience.

File Transfer

Files can be transferred over the air in a couple of different ways:

- The user can perform a drag and drop operation onto the destination device using the Explorer extensions.

* Other brands and names are the property of their respective owners.

- An independent file transfer application could be launched. This application could be a legacy file transfer application such as ProComm Plus or Intellisync, or it could be a Bluetooth application.

To illustrate further the end-user experience with the first scenario, the file transfer application on one notebook computer is typically started, and the Explorer is launched on the second notebook computer. The user would then search for the first computer under the Bluetooth *Devices* folder under the *My Computer* category of the Explorer and drag and drop the desired file onto the first notebook computer.

Device Discovery

Device discovery can be triggered manually or automatically at some preset intervals. This is a policy that needs to be set using the control panel. Once such a policy is set, remote devices are discovered when the appropriate trigger fires. Bluetooth specifications allow a device to be in non-discoverable mode where it will not be scanning for device discovery requests from other devices.

DEVICE INTERACTION

Figure 6 shows the interactions that result when a file is transferred between two devices using Bluetooth technology.

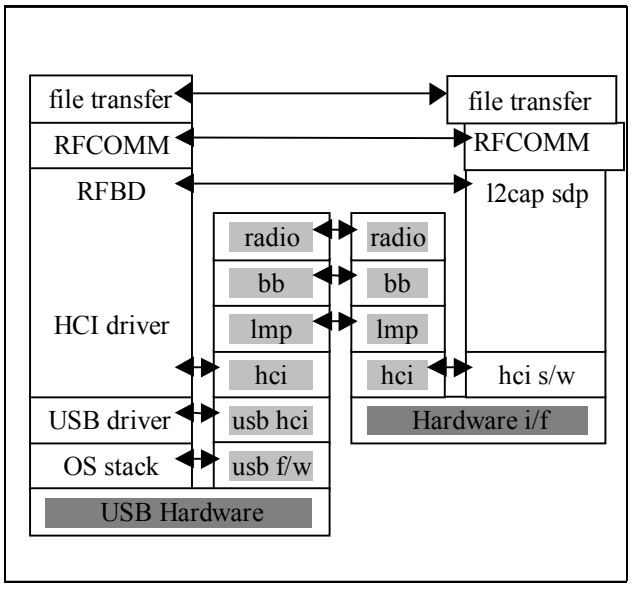

Figure 6: File transfer between devices

On the left-hand side of the picture you see a PC with a Bluetooth module interfacing to the USB controller. The right-hand side of the picture shows another Bluetooth device. For the purposes of this example, it is assumed that both devices implement file transfer profile. An inquiry process results in discovery of the device, and user selection of the device in the UI and an attempt to obtain services result in the PC learning about the file transfer service on the remote device.

The Bluetooth modules in the two devices have the radio, baseband, and LMP implemented in them. They also have HCI implemented as a means of interfacing with the host controller. On the PC side, access to the device HCI is enabled through USB. The PC communicates with the other device using over-the-air packets.

The inquiry process is initiated by a user command (or is initiated automatically after some fixed period of time, based on user policy settings on the PC), and it results in a command to the Bluetooth host controller on the module through the HCI. This results in the LMP and baseband firmware executing the inquiry sequence. The PC discovers the other Bluetooth device, and this is communicated back to the host driver stack through some HCI events carried across USB. Thus the HCI driver communicates virtually with the HCI firmware on the host Bluetooth module to learn about the new device. To achieve this, the USB minidriver interfaces virtually with the USB firmware on the module, and the USB stack, native to the OS, interfaces with the USB controller and interfaces with the USB hardware on the host module.

Once the HCI driver knows about the new device through the event, the RFBD propagates this information to the user. The user drives further actions on the device, particularly those relating to service discovery. In order to perform service discovery, the SDP and L2CAP implementations in the PC and the other Bluetooth device communicate in accordance with the Bluetooth specifications and profiles.

Once the PC discovers that the other device supports file transfer, the user on the PC can initiate file transfers to the other device. In this case, the PC is the client using the file transfer service on the other device. Initiation of file transfer is performed by a drag and drop operation on the file. The RFCOMM driver on the PC stack communicates with the RFCOMM implementation on the device to set up the RFCOMM channels on the L2CAP.

Once RFCOMM channels have been established, the two applications use the IrOBEX protocol to exchange files, in accordance with the Bluetooth profiles.

This example has glossed over details relating to individual implementation features including inquiry, driver loading, service discovery, L2CAP, security, RFCOMM, virtual comm port creation, and user interfaces.

OPEN ARCHITECTURE

There are several ways in which third parties can add value to the stack.

Third-party PC hardware needs an HCI transport driver to work with the Intel stack. Such a driver will interface with the HCI driver and would be responsible for communicating with the hardware to deliver HCI commands from the PC and to return HCI events and data back to the PC.

Third-party kernel-mode client drivers are layered above the RFBD. These drivers handle devices external to the PC that communicate with the PC using Bluetooth protocols. The main goal for these drivers is to enable custom handling of the hardware in a way that best uses innovative features in the hardware while still retaining the Bluetooth SIG standard over the air interface. Bluetooth Plug and Play (explained below) is used to load these custom drivers on RFBD.

Third-party user-mode applications are also enabled on the stack to enable custom handling of device or Bluetooth UI. These applications interface to the stack by using BTAPI for functionality specific to the Bluetooth technology and by using standard Microsoft APIs (such as the Win32 comm API*) for data transfers.

These third-party applications can be independent applications or they can be launched from within the Intel UI. An application that falls in the latter category is the file transfer application (when a file is dragged and dropped, for example).

Third-party user-mode applications can also be layered on third-party kernel-mode client drivers. This happens when the kernel-mode drivers export an API that is used by the user-mode applications.

SUPPORT FOR BLUETOOTH FEATURES

This section addresses Bluetooth security, power management, and Bluetooth Plug and Play issues.

Security

Bluetooth security has its basis in encryption, authentication, and authorization. Bluetooth security modes work differently depending on whether they are turned on and on which layer is enforcing them.

The Bluetooth General Access Profile [5] describes the security modes in more detail.

The UI elements, Bluetooth executive, and RFBD work together to create the pop-up messages in response to PIN code requests or authorization requests.

Power Management

BTAPI allows configuration of several power-related items:

- It enables radio power to be automatically set depending on the system power state.
- It allows control of page and inquiry scan policies for the radio.
- It enables control of wake-up policies: the system can wake up when a device is discovered, or it can be woken up when a new connection request has been received. Both of these can be set for a particular device, a particular class of device, or for any device.
- Device discovery policy determines when the discovery process is initiated and whether it is initiated automatically under system control. This policy can be set on a per system power state basis.
- The events indicated by HCI can also be selectively masked on a per system power state basis.

Not all of these are necessarily exposed to the end user, but these are capabilities available for applications through BTAPI. The applications can expose these to the users as appropriate.

BLUETOOTH PLUG AND PLAY

The need for custom drivers for third-party hardware drives the need for Bluetooth Plug and Play (PnP) [7]. Third parties would want to write custom drivers for their hardware to make use of the unique features of their hardware. Bluetooth PnP provides a means for identification of the right driver to load in response to devices discovered.

SUMMARY

This paper presents an overview of what the Intel Bluetooth software stack implements. This could be viewed as an example Bluetooth implementation on the PC platform.

The stack provides for feature enhancements both from ISVs and IHVs.

Third-party Bluetooth application developers can add user-mode components on top of BTAPI for application

* Other brands and names are the property of their respective owners.

management, and Win32 comm* API for data transfer and flow control. In addition, kernel-mode drivers developed on RFBDI can expose a user-mode framework that is different from BTAPI, if needed.

Drivers can be developed on RFBDI for several reasons, with a custom user-mode framework being one of them. Other reasons include the development of support for new usage models and the development of custom support for new hardware devices. The Intel stack provides the infrastructure on which new usage models can be hosted. This infrastructure includes implementation of the base Bluetooth protocols. Implementation of custom drivers for new hardware uses the Bluetooth PnP framework.

In order to interoperate with the Intel stack, device manufacturers need to conform to the Bluetooth air protocol and the applicable profiles.

Following are the key takeaways from Intel's driver stack implementation:

- The driver stack is for WDM-based operating systems like Windows98* and Windows2000*. It is to be noted that in Windows98, we do need a VxD module to interface with VCOMM.
- The driver stack can be enabled on any PC platform. For enabling it on desktops, hardware solutions like a USB dongle are needed.
- The stack supports Bluetooth SIG 1.0 specifications and profiles.
- The stack enables native OS applications on Bluetooth devices where possible.
- BTAPI enables access to functionality specific to Bluetooth technology from user-mode applications.
- RFBDI enables third parties to write kernel-mode drivers for their Bluetooth peripherals.

ACKNOWLEDGMENTS

Ron Mosgrove, Jon Inouye, Steve Lo, Chad Zhu, Bailey Cross, Bruce Clemens, Anurag Prakash, Diane Stevenson, and Jim Kardach were also involved in the architecture definition at various stages. Thanks are also due to Jon, Ron and Steve for detailed review of the final paper.

REFERENCES

[1] Tso, M., Gillespie, D. and Romrell, D. "Always on always connected mobile computing," *Proceedings of the international conference on universal personal communications*, 1996, pp. 918-924.

[2] USB Implementer's Forum, "Universal Serial Bus Specification," *Revision 1.0*, Jan 15, 1996.

[3] Webb, John P., "Bluetooth SIG MRD," Bluetooth *Specification 1.0*.

[4] Bluetooth Special Interest Group, "Specification of the Bluetooth System: Core," *Version 1.0B*. July 26, 1999.

[5] Bluetooth Special Interest Group, "Specification of the Bluetooth System: Profiles," *Version 1.0B*. July 26, 1999.

[6] Microsoft, "Windows2000 Device Driver Kit."

[7] Intel Corporation, "Bluetooth Plug and Play," to be published.

AUTHORS' BIOGRAPHIES

Kris Fleming is a Senior Software Engineer in the Bluetooth Communication Architecture group at Intel Corporation. He has a B.S. degree in electrical and computer engineering and a M.S. degree in computer engineering from the University of Maine. In addition, he has an M.S. degree in information networking from the Information Networking Institute at Carnegie Mellon University. His research work has focused on mobile computing and wireless networking. Kris was one of the initial members of the Intel team, chaired the Bluetooth SIG HCI working group, and is currently the co-chair of the Personal Area Networking Bluetooth 2 SIG working group. His e-mail is kris.d.fleming@intel.com.

Uma Gadamsetty joined Intel in 1995 as a senior software engineer and is a team member of Mobile Communications Operation. He has developed device drivers and protocol implementations under a wide array of operating systems. His e-mail is uma.gadamsetty@intel.com.

Robert J. Hunter is a senior software engineer in the Mobile Communications Operations department. Robert is responsible for Ambler software development and led the Bluetooth USB HCI Specification task force. His email is robert.j.hunter@intel.com.

Srikanth Kambhatla has been working on Windows NT OS and HAL extensions and drivers in many capacities in Intel since 1991. In addition to Bluetooth, he has been involved in the definition of PC Plug and Play, Multiprocessor specs for PCs, IrDA Control and IrDA PnP standards. Srikanth has an M.S. degree in computer science and engineering. His e-mail is srikanth.kambhatla@intel.com.

* Other brands and names are the property of their respective owners.

Sridhar Rajagopal has been at Intel since 1993. He has a M.S. degree in electrical engineering from Arizona State University. He has worked in various software development organizations and projects at Intel and is currently working on the Intel Bluetooth Software suite. His e-mail is sridhar.rajagopal@intel.com.

Sundaram Ramakesavan has an M.S. degree in computer science from Queen's University, Canada. He has worked in various Telecommunications and Data Communications projects at Nortel Networks. He is currently working on the UI elements of the Intel Bluetooth Software suite. His e-mail is ramu.ramakesavan@intel.com.

Enabling Always On, Always Connected (AOAC) Computing with Bluetooth Technology

Kristoffer Fleming, Mobile Computing Group, Intel Corporation
Robert J. Hunter, Mobile Computing Group, Intel Corporation
Jon Inouye, Mobile Computing Group, Intel Corporation
Jeffrey Schiffer, Mobile Computing Group, Intel Corporation

Index words: Bluetooth, personal area networking, wireless communication

ABSTRACT

For the past several years, Intel has been exploring the space of an Always On, Always Connected (AOAC) usage mode for both the home and mobile personal computer. Intel is also one of the promoting companies of the Bluetooth* Special Interest Group, whose purpose is developing specifications for a new short-range wireless technology that, when coupled with advances in power management, smarter applications, cellular networks, and information kiosks, makes this vision a reality for mobile notebook platforms. The IEEE is currently examining the technology for their Wireless Personal Area Networking standard [8].

Other papers in this issue of the *Intel Technology Journal* provide an overview of the Bluetooth wireless technology [4,5]. This paper focuses on key mechanisms provided by the technology to support AOAC applications. These mechanisms are evaluated in terms of how well they support various AOAC applications.

INTRODUCTION

Communication has revolutionized how people use computers. Local area networking (LAN) has enabled information and device sharing, moving the world from a centralized computing structure to a more distributed system approach. Networking has increased the usefulness of the computer by allowing it to share and exchange data seamlessly across a wide variety of physical technologies. We expect advances in mobile networking will similarly change the way people use mobile computers.

The most popular usage model for mobile computers today is based on nomadic computing. Mobile computers are used in stationary locations such as hotel rooms, airport clubs, or conference rooms. Accessing the Internet or communicating with the user's enterprise services is deliberate and infrequent. It depends on the connectivity options at each stationary location. Wireless Wide-Area Networking (WWAN) provides mobile users with another option.

Convenient On-Demand Communication

There are a variety of WWAN technologies and services available today [6]. The emergence of data services on wide-area cellular networks provides a new communication "gateway" for exploitation by mobile users. Wireless modems allow on-demand data connectivity using the cellular network. PC Card cellular modems or cables linking analog cellular phones to mobile computers have been available for several years, but the connection was prone to frequent errors and disconnections. However, the emergence of digital cellular data services on GSM and IS-95 has helped make connectivity more reliable. This paper focuses on data access through cellular handsets equipped with Bluetooth wireless technology.

Wide-area data services make it easier for the mobile user to instantly create a communication channel. This usage model is comparable to that of a home PC user except for the fact that the computer is more portable than the home PC, and the communication pipe is usually thinner, referred to as weak connectivity, and more expensive to use. Bluetooth wireless technology allows the PC to access the cellular network while the cellular handset is within range. There is no need for a cable, or even for taking the phone out of a purse or pocket.

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

Always On, Always Connected

On-demand communication tends to be deliberate, that is, the user always initiates the transfer of data. Tso used the phrase "the data I want finds me" and that is what the AOAC usage model does [9]. This is analogous to the enterprise LAN model where desktop users get e-mail notifications, video conferencing calls, and calendar reminders sent to them.

For the mobile user, AOAC means data continues to find the user even when the user is outside the enterprise. This requires enterprise applications to be able to find the mobile user or to be redirected via some proxy agent. In any case, some form of redirection mechanism is needed to locate the mobile PC.

Besides cost and bandwidth differences as mentioned in on-demand communication, AOAC for mobile PCs needs to take into account battery life. It may be fine to leave a desktop computer on and always connected to the Intranet, but leaving a notebook computer on and connected has serious implications for battery life. Therefore, schemes to enable AOAC technology tend to focus on "virtually connected" solutions where the mobile computer is sleeping the majority of the time to preserve the battery. As described later in this paper, Bluetooth wireless technology allows the phone to "wake up" the mobile computer and transfer data to it. This enables a number of future scenarios where data are delivered to briefcase-enclosed mobile computers while their users are carrying them down the street.

Proximity-Based Computing

Mobile computers have one very significant advantage over stationary desktop computers: they can travel with the user. This ability opens up a variety of new applications or location-based extensions to existing applications. Mobile computers may determine their relative position via a variety of methods including an attached global positioning system (GPS) device or a local cell site ID from a personal cellular phone[1]. This information could be fed to Internet servers providing location-based services such as maps, restaurant guides, and tourist information. Bluetooth wireless technology allows the computer to extract location information from GPS devices and cellular handsets also equipped with the technology and pass this information on to applications running on the computer.

A natural extension to wide-area location-based services is local-area location-oriented services where communication is via information servers or conduits embedded into the local infrastructure. Usage models include local access points for Internet/FAX/voice services, conference room A/V remote control, Point-of-Sale (POS) transaction terminals in stores, electronic white and yellow pages in phone booths, and electronic menus and maps for nearby restaurants in transit terminals and hotels. For example, take the case of business users who have been working while disconnected. As they enter a restaurant, their computers recognize a public data access point is within range and its advertised rate is reasonable. Through a Bluetooth link to the user's cellular phone or personal digital assistant (PDA), the computer sends a prompt to OK a file system synchronization operation. If the user positively acknowledges the operation on the phone or PDA, the computer connects to the data access point, and critical modified files are brought into synchronization with the files on the enterprise server where they can be backed up and made available to other users. To enhance enterprise security, all transferred data should be protected between the computer and the enterprise and should not depend on the security of the Bluetooth link.

In the proximity-based computing model, embedded services advertise themselves to mobile users. Users may either consciously or unconsciously decide to use them. In the example above, a user could create a profile that waives user authorization for activities deemed favorable enough. This type of unconscious proximity-based computing has the promise of simplifying a user's life by doing the rather obvious things automatically. Bluetooth technology supports this usage model by its wireless nature and ability to discover devices that are nearby.

Recap

To summarize, some of the differences between desktop and mobile computing usage models are as follows:

- WWAN provides weaker connectivity than LAN.
- WWAN has higher communication costs.
- Mobile computers have limited battery life.
- Mobile computers may take advantage of location-based applications.
- Proximity use benefits mobile users more than desktop users.

In the remainder of this paper, we focus on low-power, weakly connected mobile computer usage models that use Bluetooth wireless technology. We do not address optimizing communication cost reduction algorithms.

[1] GSM 07.07 defines AT commands that may be used to extract this information from the phone.

SUPPLYING THE MECHANISMS

Bluetooth wireless technology will not enable the above usage models by itself. It is simply one piece in the puzzle. In this section we describe the basic mechanisms supported by the technology.

Low Power

Designed for use in handheld battery-powered devices, Bluetooth radios have relatively low power consumption in an active state and even lower consumption in a standby state. The Bluetooth specification does not specify power requirements, but manufacturers are advertising solutions with standby currents of less than 0.3 mA and active data currents no higher than 30 mA at voltages between 3 and 5 V for a maximum power drain of 0.15 W. A typical notebook Li-Ion battery packs 40 W hours of power. If the battery only had to power the radio, this converts into a notebook battery lifetime of 100+ hours of active radio use and months of standby time before the battery is exhausted. Therefore, the Bluetooth radio will not be a significant drain on notebook battery life compared to the rest of the system.

Discovery Mechanism

Bluetooth wireless technology uses an inquiry procedure to discover the addresses of all the devices in the vicinity. To be discovered, each device needs to enter an Inquiry Scan mode. To discover devices, a device must enter a probing state called Inquiry mode. In this mode, the device collects the remote clock, address, and class of device (COD) information from all units responding to the inquiry message. It can then, if desired, make a connection to any one of them by means of a paging procedure described in the next section.

In the Inquiry state, an inquiry access code message is broadcast by the source. The inquiring device alternatively transmits this message and listens for responses over a 10.24 second period. During this time, it can do little else besides probe the environment and listen for responses. The device listening for inquiry messages has a much simpler task: it listens for only 10.625 ms every 2.56 seconds. In addition to having different scanning times, each responding device uses a random delay before sending the response in order to avoid collisions with responses from other devices. A device typically sends multiple responses (due to receiving multiple messages) during the discovery process, usually on different frequencies, thus yielding a robust discovery mechanism.

The inquiry access code does not contain any information about the source, but does indicate the devices that should respond. There are 64 reserved inquiry access codes and only two are currently defined in the Bluetooth Specification [1]. The first, Generic Inquiry Access Code (GIAC), requires that all *discoverable* devices must respond[2]. In a dense environment like a conference, airport terminal, or classroom there may be many devices that are discoverable. The Limited Inquiry Access Code (LIAC) was created to support the *limited discovery mode*. The objective of the limited discovery mode is to create a user-initiated situation that limits the number of devices discovered. Devices operating in this mode must respond to messages containing the LIAC. By definition, devices must not enter limited discovery mode for more than a minute before they are forced to leave it. Devices in discoverable mode must not respond to LIAC messages. For more details on the Inquiry process consult [1,2].

Connection Mechanism

The connection mechanism is referred to as "paging." The device initiating a connection enters the Page mode. The initiator needs only the 48-bit Bluetooth device address of the target to set up a connection. During paging, the initiator alternatively transmits the target's device access code[3] (DAC) and listens for responses over one set (out of a possible two) of hop frequencies. If no response is detected within 1.28 seconds, the second set of hop frequencies is used. A good estimate of the target's internal clock will accelerate the paging procedure by selecting the correct set of initial frequencies to use. Paging is a very active process, and the initiating radio has little time for anything else while operating in this mode.

To hear paging messages, a device must enter a Page Scan mode. In this mode, the device listens to a single hop frequency band for 11.25 ms every 1.28-second interval, switching to another frequency in a pre-defined sequence during the next interval. Outside this 11.25 ms scan window, the radio may communicate with other radios or enter a power-saving standby state.

For the paging process, several paging schemes can be applied. This section only describes the mandatory paging scheme that has to be supported by each Bluetooth device. For further information on this and alternative paging schemes, consult the Bluetooth Specification [1].

Authentication Mechanism

While untethered access is one of the strengths of wireless technology, it is also a weakness: connectivity to the notebook is no longer restricted to physical access. This becomes a problem when supporting proximity-based computing because battery power is limited. While the

[2] To be discoverable, a device must enter Inquiry Scan mode.

[3] Derived from the 48-bit device address.

notebook operating system may provide protection mechanisms, the involvement of the operating system in proximity-based computing consumes a significant amount of power. This is because the majority of the system, barring the LCD display, needs to migrate to an S0 state (full on). Moreover, involving the operating system in security checks may expose the computer to power drain attacks. If the radio module itself were to filter out undesirable connection requests, then this could be a solution to these power drains.

The Bluetooth Link Manager Protocol (LMP) supports both one-way and two-way authentication by using a challenge-response scheme based on shared secret keys. One-way authentication means that one device authenticates the other by sending a random number and receiving the proper response. Two-way authentication means that both devices authenticate each other. In order to filter wake-up events, it is important for the Bluetooth device to support authentication; not all Bluetooth devices are required to support this feature.

The secret keys are established through a process called "pairing." Pairing starts when there is an assumption of a known shared secret, such as an agreed upon personal identification number (PIN). During the pairing process, devices may prompt users to enter a PIN. If the PIN entered on both devices match, a shared secret key, called a link key, is created for those two devices.

Using the Host Controller Interface described later, incoming connections may automatically trigger link authentication before they are accepted. This allows the radio module to prevent unauthenticated connections requests from waking a "sleeping" computer as described in a later section.

Privacy Mechanism

In addition to the need for authentication, wireless communication also raises concerns about privacy. The communication between the devices should be protected from casual eavesdropping in the same way that a cable provides physical security. To protect the privacy of the communication medium, the Baseband protocol implements a stream cipher using an encryption algorithm called E_0.

Bluetooth privacy mechanisms are targeted at protecting privacy but are limited by global encryption regulations. For corporate confidential communication, an application-based end-to-end security mechanism should be used.

Wakeup

Because battery life is so important, the PC industry has defined a variety of low-power modes in the Advanced Configuration and Power Management Interface (ACPI) specification [3]. Table 1 describes the ACPI sleeping states. To enable proximity-based, unconscious application-level activity while preserving battery life, the PC must migrate from "sleeping" states to a "working" state, and back again to a sleeping state.

The PC may be woken up through an interrupt, and ACPI uses the System Control Interrupt (SCI). Any Bluetooth radio module connecting to the PC uses an SCI to wake the PC from a sleeping state.

State	Description
S0	Working state.
S1	Low wake-up latency sleeping state. No system context is lost.
S2	Low wake-up latency sleeping state. CPU and cache context are not maintained by hardware.
S3	Low wake-up latency sleeping state. All system context is lost except for system memory.
S4	Lowest power, longest wake-up latency sleeping state. This state assumes all devices have been powered off.
S5	Soft off state.

Table 1: ACPI states

To allow intelligent "wake-up" policies, the Bluetooth Host Controller Interface (HCI) allows the host PC to define various "filters". Part of the Bluetooth Specification, the HCI, defines a set of commands that must be supported across all radio devices that claim conformance to the HCI. The HCI Set_Event_Mask command may be used to define which events wake up the host. Table 2 provides some sample events. See the HCI chapter within the Bluetooth Specification for a complete list of events available for this command.

Event Description
Inquiry Complete
Inquiry Result
Connection Request
Disconnection Complete
Authentication Complete
Remote Name Request Complete

Table 2: Sample HCI events

The HCI Set_Event_Filter command may be used to further refine the events allowed to wake up the host. For

the Inquiry Result event, filters may be used to specifically allow only responses from certain classes of devices, e.g., data access points, to wake up the host, while ignoring all other responses.

THE INTEL BLUETOOTH SOLUTION

In this section we describe the Intel solution for enabling AOAC computing. This solution consists of an embedded radio module connected to a USB port and a provision for operating system support for the module.

The radio module connects to one of the unused USB ports on the PIIX4 chipset. This chipset supports two USB ports, and notebook PC manufacturers normally use only one port. Because the operating system recognizes devices physically attached to the USB controller by polling for activity every millisecond, USB devices are not able to wake up the system from an S3 or S4 sleeping state. This severely limits AOAC usage models. As a result, Intel has recommended developers work around this problem through the use of separate sideband signals that provide a Detach and Wake-up event.

The Intel Bluetooth Windows* software stack [5] implements the necessary Bluetooth protocols and configuration software to support the radio module. The following example describes how proximity-based data synchronization is enabled on an ACPI-compliant host.

1) The synchronization application registers with the Intel Bluetooth API to receive connection events.
2) Upon closure of the notebook lid, the operating system suspends the notebook, taking it to an S3 sleeping state.
3) The PDA is used normally, but the synchronization application on the PDA has a timer associated with the change log, authorizing synchronization every six hours if changes are detected. This allows the synchronization to be initiated either consciously by the user, or unconsciously by the expiration of a timer.
4) Synchronization application on the PDA issues a connection request to the notebook.
5) On receiving the page request, the HCI event filter triggers a wake-up event after recognizing the PDA's address.
6) The operating system migrates the notebook to the S0 "working" state, and the Bluetooth software drivers receive a "connection established" event from the HCI. The Intel Bluetooth API alerts all registered applications of this event.
7) The synchronization application on the notebook receives the event and initiates application-level communication with the application on the PDA.
8) Once the synchronization transaction completes, the application on the notebook puts the system back to sleep using an operating system command such as those available through the OnNow Win32 extensions [7].

OUTSTANDING ISSUES

In the previous section we described the mechanisms Bluetooth wireless technology provides to support AOAC mobile computing. In this section, we describe some outstanding issues preventing the wide adoption and use of AOAC computing models.

Thermal Concerns

Many notebook computers are cooled by fans that use forced air currents to prevent overheating. The effectiveness of these fans decreases significantly when the notebook is placed in an enclosure such as a briefcase or portfolio. This may cause temperatures to rise beyond operational limits. However, there is little danger that the system will burn up because ACPI enables thermal management by the operating system. In other words, the operating system can shut down the host device when a critical threshold temperature is reached. This shutdown process may disable the Bluetooth module from further operation until the user directly boots the system.

Communication Costs

While enabling wide-area data communication, cellular services typically cost more than wide-area wired communication technologies. Within the enterprise, most people assume their communication use is unlimited and free. Wireless communication puts a new spin on the AOAC goal, changing it more to a "the data I want finds me as long as I can afford it" model.

There also is the issue of determining what users are willing to pay for. The "usefulness" semantics of data are often hidden from most communication mechanisms though e-mail utilities often allow you to categorize e-mail based on urgency, the sender's identity, and simple content checks.

Once the usefulness of the data has been determined, the next step is to determine how expensive the communication will be. Unfortunately, the cost of the communication medium is not something most networks

* Other brands and names are the property of their respective owners.

provide within the network, or at least it is not extractable by the end user. Accurately budgeting the media, therefore, requires both prioritizing data and determining monetary costs associated with the communication.

Intelligent Applications

The key concept to better mobile user models is intelligent adaptive applications. Applications need to adapt to a variety of mobile profiles, where each profile describes the current operating environment (battery powered, connectivity options, communication urgency, etc.) and the user preferences for that environment.

A well defined set of user profiles would allow users to migrate between a small subset of customized "personalities" and allow applications to adapt appropriately.

CONCLUSION

In this article we describe various mechanisms provided by Bluetooth wireless technology to support AOAC mobile computing. While an important enabler, these additional features will not provide rich new usage models in and of themselves. Additional work is needed to resolve the issues we have discussed here, as well as the issues that we yet have to discover. However, we feel this new technology provides some significant immediate benefits to users of mobile computers: wireless cable-replacement convenience, low-power "unconscious" usage models, and enhanced programming interface for application developers.

ACKNOWLEDGMENTS

The authors thank the other members of the Ambler software development team for many discussions on the topics addressed in this paper. We also acknowledge the input and help from reviewers and editors.

REFERENCES

[1] Bluetooth Special Interest Group, "Specification of the Bluetooth System, volume 1: Core," Revision 1.0b, December 1, 1999.

[2] Haartsen, J., "The Bluetooth Radio System," *IEEE Personal Communications*, volume 7, no. 1, February 2000, pp. 28-36.

[3] Intel, Microsoft, and Toshiba, "Advanced Configuration and Power Interface Specification," Revision 1.0b, February 2, 1999.

[4] Kardach, Jim, "Bluetooth Architecture Overview," *Intel Technology Journal*, Q2, 2000.

[5] Kambhatla, S. et al., "Architectural Overview of Intel's Bluetooth Software Stack," *Intel Technology Journal*, Q2, 2000.

[6] Katz, R, "Adaptation and Mobility in Wireless Information Systems," *IEEE Personal Communications Magazine*, vol. 1, no. 1, 1994, pp. 6-17.

[7] Microsoft, "OnNow Power Management Architecture for Applications," white paper, March 14, 2000. http://www.microsoft.com/hwdev/desinit/onnowapp.htm.

[8] Siep, T., Gifford, I., Braley, R., and Heile, R., "Paving the Way for Personal Area Networking Standards: An Overview of the IEEE P802.15 Working Group for Wireless Personal Area Networks," *IEEE Personal Communications,* vol. 7, no. 1, February 2000, pp. 37-43.

[9] Tso, M. and Gillespie, D., and Romrell, D., "Always On Always Connected Mobile Computing," *Proceedings of the International Conference on Universal Personal Communications*, 1996, pp.918-924.

AUTHORS' BIOGRAPHIES

Kris Fleming is a Senior Software Engineer in the Bluetooth Communication Architecture group at Intel Corporation. He has a B.S. degree in electrical and computer engineering and an M.S. degree in computer engineering from the University of Maine. In addition, he has a M.S. degree in information networking from the Information Networking Institute at Carnegie Mellon University. His research work focused on mobile computing and wireless networking. Kris was one of the initial members of the Intel team, chaired the Bluetooth SIG HCI working group, and is currently the co-chair of the Personal Area Networking Bluetooth 2 SIG working group. His e-mail is kris.d.fleming@intel.com.

Robert Hunter graduated from Oregon State University in 1994 with a degree in computer science. Since then he has worked for Intel Corporation. One of the initial members of the Intel Bluetooth team and active in SIG1, he is currently a Sr. Software Engineer working on the implementation of the Intel Ambler Software Suite. His e-mail is robert.j.hunter@intel.com.

Jon Inouye graduated from Washington State University in 1986 with a degree in electrical engineering. He worked as a component engineer in the parallel processing industry for several years and returned to academia as a Ph.D. candidate in the CSE department of the Oregon Graduate Institute. Jon was awarded an Intel Foundation Graduate Fellowship in 1995 for his work on adaptive

operating systems. He joined Intel in 1997 where he is now a Senior Software Engineer in the Bluetooth Communications Architecture Group. His e-mail is jon.w.inouye@intel.com.

Jeffrey Schiffer is the Engineering Manager of the Subsystem Architecture Group in the Mobile Computing Group at Intel Corporation. He has a B.S. degree in electrical engineering from the University of Massachusetts, Lowell. Since graduation, he has focused his energies on the design and development of RF systems, cable modems, synthesizers, microwave systems, and secure networking devices for both the government and private industry. Jeff was one of the original members of the Intel Bluetooth team and currently chairs the Bluetooth Security and Aviation regulatory task forces. His e-mail is jeffrey.schiffer@intel.com.

Integrating Bluetooth Technology into Mobile Products

Graham Kirby, Mobile Computing Group, Intel Corporation

Index words: Bluetooth, antenna, notebook PC

ABSTRACT

In the past we have seen technologies like PC-Card, PCI, USB, and IrDA introduced to enrichen the mobile computing environment. Bluetooth* technology is the latest such initiative to add a connectivity feature to mobile systems

Bluetooth technology is focused on replacing cables used to connect different mobile devices together. The technology is based on very small, low-cost, light-weight radios that easily form ad-hoc, secure links between various devices, to allow personal connectivity.

With Bluetooth technology we see a combination of several strategies from the PC and cellphone industry that have been used to bring new system features to market.

It is arguably the first time such a new technology has been integrated on the first-generation mobile PC products. To reduce the likelihood of incompatibility issues, significant effort is focused on interoperability testing of products from different vendors before they are launched.

Traditionally, mobile PC manufacturers have focused on reducing RF emissions from their designs. With Bluetooth technology, many manufacturers are exposed to the challenge of integrating intentional emitters into their designs for the first time. Beyond the technical issues, this introduces some new regulatory processes. Bluetooth is a mobile technology. Users can travel with their products equipped with Bluetooth technology to any country in the world. The Bluetooth Special Interest Group (SIG) and Intel are working with regulatory bodies worldwide to explain the technology to them with the goal of establishing simple certification procedures and enabling users to travel freely thoughout the world with such devices.

The airlines and the Federal Aviation Authority (FAA) are taking a very cautious stand on allowing new classes of electronic devices to be used on aircraft. By engaging early with the FAA, its European equivalent, the Joint Aviation Authority (JAA), and the airline community, the Bluetooth SIG has made good progress on educating these bodies about Bluetooth technology so they can make reasoned decisions on whether to allow Bluetooth devices to be operated on aircraft.

INTRODUCTION

There are many aspects to getting Bluetooth technology integrated into a mobile PC. This paper describes the strategies that were adopted to make the integration process as smooth and predictable as possible. How some design decisions were made to simplify the integration tasks for the notebook computer manufacturer is also discussed.

While this paper is geared towards the notebook PC space, many of the ideas and strategies apply equally well to other Bluetooth applications, and beyond that, to other technology integration activities.

INTEGRATION VERSUS EXTERNAL IMPLEMENTATIONS

The long-term promise of Bluetooth technology is wireless connectivity between devices occuring without user involvement. By definition, this precludes plugging things in, or positioning things in a special way so that they can talk to one another. It seems natural, therefore, that for the Bluetooth initiative to achieve this objective, it has to be integrated into the target devices.

A further argument supporting integration is the cost of integration versus that of external add-ons. The natural alternatives to integrating Bluetooth devices into notebook PCs would be either a PC-Card or a USB dongle. Both of these solutions require additional interfacing and packaging that can dwarf the cost of the actual Bluetooth components. They also require a degree of user interaction and add variability to the system environment that can defeat the 'it just works' principle.

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

Usually technologies only get integrated into platforms once they have reached a level of maturity where the integrators can be sure that they will not become obsolete before the host system. A reasonable rule of thumb is about two years. To understand why it makes sense to integrate Bluetooth technology from the start, we need to look at the activities of the Bluetooth SIG.

The Bluetooth Special Interest Group

The Bluetooth SIG has gone to some lengths to ensure that the technology is well defined and reliable on the first instantiation. It follows the model established by GSM in making a clear distinction between the following:

- technical specification development
- definition of usage models (profiles)
- logo testing and certification

Before a product can be shipped with the Bluetooth logo, the manufacturer has to prove that it correctly supports the claimed profiles. The grant to use the Bluetooth Intellectual Property (IP) is contingent on this testing process. The mechanics of the testing process are covered in more detail in a later section.

Components of a Complete Bluetooth System Implementation

There are four distinct pieces to a complete Bluetooth product for mobile PCs: hardware, software, the regulatory approvals required before a product can be marketed and used in different countries, and the approval from the Bluetooth SIG, which allows the OEM to use the Bluetooth IP and logo (see Figure 1).

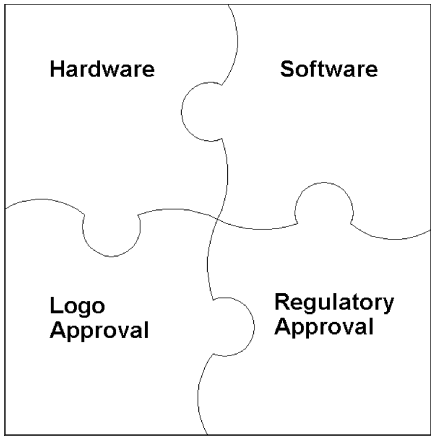

Figure 1: The four pieces of a complete Bluetooth product

The Bluetooth hardware has to be designed to meet the usual mobile system design requirements of low power consumption and small size. Most importantly, and in the spirit of the technology, low cost is also a key feature. While the first systems that incorporate Bluetooth technology will probably be higher-end systems, we expect that over time, products that use Bluetooth technology will come down in cost to the point where this technology will automatically be integrated into every system in much the same way that Infrared is today.

It is recommended that the Bluetooth hardware is contained within a module for ease of integration and regulatory approval By maintaining a clear distinction between the Bluetooth hardware and the rest of the system, the regulatory approval process and some of the manufacturing aspects are simplified. This paper assumes this hardware configuration.

When integrating Bluetooth technology into a design, an antenna needs to be incorporated. The selection of the antenna will be influenced by the specific system design constraints, mainly the space available for the device. Once an antenna is chosen it has to be submitted with the Bluetooth radio to the regulatory bodies for approval in the target countries.

Regulatory approval refers to the intentional emitter aspects of the Bluetooth radio system, and to the legal requirements that must be met within the different countries the product can be used in (this is sometimes referred to as type approval). Part of the strategy of those involved with Bluetooth technology is to provide a module that already meets the above requirements and is globally pre-approved for use in the intended countries (i.e., those where the product is sold and those where the product may roam into). For the purposes of this paper, it is assumed that the Bluetooth module used for integration is of the pre-approved variety.

To ensure interoperability between various classes of devices, the Bluetooth SIG requires products (that contain the entire Bluetooth capabilities stack) to be qualified by a Bluetooth Qualified Test Facility (or similar organization). These test facilities ensure that the various components of the Bluetooth radio system have been assembled correctly and will interoperate with other devices that meet the qualification program. To ensure that products will interoperate, the Bluetooth SIG ties the granting of the IP license grant to certification of the product.

Most notebook systems have very similar configurations so the Bluetooth software developer has the opportunity to do a lot of pretesting of the software/hardware combination before the OEMs final system qualification takes place.

The System Interface

The USB interface was choosen as the primary Bluetooth interface for mobile PCs because of its existing support within the PC software structure, its ability to support both isochronous and asynchronous data (voice/data), and the fact that it consumes no additional resources in the PC system (such as interrupts). Moreover, much of the software infrastructure created for the USB system (such as the Human Interface Device specification) can be reused . USB also provided the flexibility to locate the module anywhere in the system by routing the the 4 USB signals.

Because USB is such a low pin count interconnect, it also reduces the cost of socketing the module if the mobile PC OEM wish to support the use of Bluetooth technology as a build-option on their design.

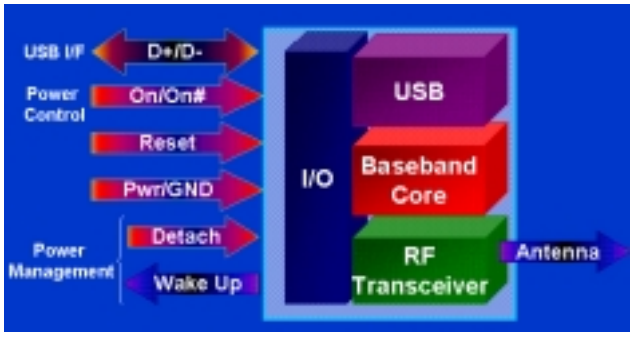

Figure 2: Bluetooth hardware interface

With early USB implementations there was an issue that prevented the CPU from going into its low power C3 state when USB peripherals were attached. (Obviously you cannot unplug an internal device). Our solution was to introduce two sideband signals: WAKE and DETACH as shown in Figure 2.

The DETACH signal is used by the system to tell the module when the USB device can be logically removed from the system. When this is asserted, the module removes itself from the USB bus such that the system thinks the Bluetooth module has been unplugged. The algorithm for asserting DETACH takes into account the current connect state and the time since the last connection.

Once the module is DETACHed, it needs the ability to reconnect itself. This is done by the module asserting the WAKEUP signal to the system. During the DETACHed phase, the module is still powered and able to respond to interrogation by other Bluetooth devices: when something interesting happens (i.e., another Bluetooth device pages the notebook to form a piconet), the module will assert the WAKEUP signal to the notebook indicating it should de-assert the DETACH signal (allowing the Bluetooth module to "reconnect" to the system). The Bluetooth specification includes a mechanism whereby the host system can set these WAKEUP criteria.

ANTENNA TYPES AND TRADEOFFS

One of the key design points in implementing Bluetooth wireless technology into a system is the selection of an appropriate antenna. Generally, the larger antennas perform better than the smaller ones but the smaller devices are easier to hide within a portable product. An additional consideration is that the distance between the radio and the antenna should be kept to a minimum to reduce the loss of signal strength between the two.

Antenna design is a very mature science and while the notebook PC is a challenging environment in which to locate an antenna, there are very few new tricks left. The initial strategy was to narrow in on antennas that had the following characteristics:

- good gain characteristics
- small size
- a good Omni-directional radiation pattern
- low cost

Keeping these characteristics in mind, we evaluated a number of antennas and came to these conclusions.

Ceramic. We evaluated a range of small ceramic antennas all of which were approximately 10mm x 5mm x 5mm. This is a very interesting size for integration into notebook PCs as these antennas can be placed in many different locations. However, our initial findings indicated that their efficiency is not as high as some of the other antenna types. Therefore, empirical testing is recommended to ensure a notebook PC equipped with such an antenna has adequate range.

Half Wave Dipole. These antennas were found to have good RF characteristics, but the form factor was less than ideal for integration into notebook systems. These long and thin devices are usually made to stick outside the platform, though a better location, for reasons of mechanical reliability, is in the side of the display.

Planer Inverted F Antennas (PIFA). These can measure approximately 45mm x 10mm x 10mm. While they are undoubtedly large compared to most ceramic devices, we found that their radiation pattern and performance were without question the best of the three types.

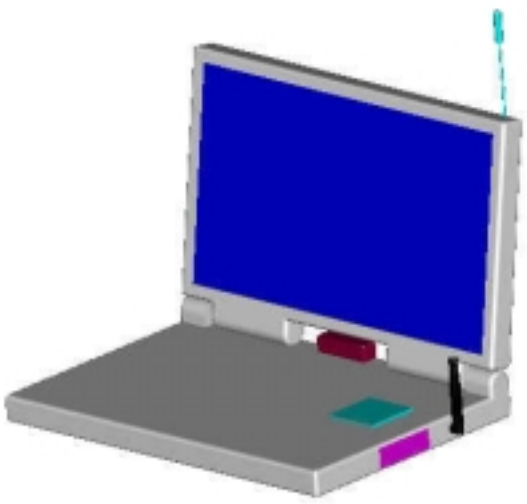

Figure 3: Antenna placement examples (good and bad)

Antenna and Module Placement

As previously mentioned, the distance between the antenna and module should be kept to a minimum in order to stay within effective range. There are no such restrictions on where the Bluetooth module can be located in relationship to the system board due to the flexibility of the USB interface. From an RF standpoint, the notebook system appears as two large ground planes: the screen and the base. Each of these will act as a shield, so you cannot mount an antenna in the center of the backside of the display and expect it to connect with another Bluetooth device held in front of the display. Furthermore, these two ground planes are hinged together and, depending on how open or closed the system is, can drastically affect the radiation pattern. Another consideration is that the user sitting in front of the system is effectively another ground-plane. So placing the antenna in the palm rest area in front of the keyboard is not an efficient location.

For most antenna types we found that the best location was in the hinge area. In this location, the screen and system unit present a uniformly small shadow irrespective of whether the system is open or shut or at some point in between. It is also in a part of the system that is reasonably free of obstruction.

As a second choice, the edge, or the top of the display area has some advantages for long thin antennas such as the ½ wave dipole. It is possible to conceal the antenna in the edge of the display. However, as the display goes from open to closed, the polarity of the pattern moves by 90 degrees. The disadvantage of this location is that as displays get thinner and bigger such a location is mechanically challenging. Some display manufacturers are at the point of molding the plastics around the panel so that they can get the maximum screen side for a given notebook form factor.

System Mechanical and Material Effects

As notebook PC technology has developed over the past ten years, manufacturers have become expert in eliminating RF emissions from their systems. They have improved plating techniques for the plastics; developed carbon-filled polymers, and have found many other ways to prevent their designs from leaking EMI.

With the integration of Bluetooth technology, we are now asking them to allow one piece of the system to radiate, so the strategy of wrapping the entire system in EMI shielding material is no longer appropriate.

The recommended solution is to position the antenna behind an RF transparent window within the system, but if it is recessed into the system, the walls of the cavity will restrict the radiation pattern. It is better to locate the antenna in a bubble that stands above the surface of the design.

Traditional protruding rod antennas, whether they are extendable or flexible, do not fit well into typical notebook computer designs. The issue is reliability. Corporate notebook PCs are handled very roughly, and experience has shown that things that stick out get broken off. While the cost of the antenna may be insignificant, the cost of repairing the damaged system is not.

Figure 3 shows several of the antenna locations that we experimented with. For the reasons described above, our first choice is the hinge area and our second choice is the region on the side of the base unit.

Antenna Testing

The process of evaluating antenna performance in a notebook system is relatively well defined. Companies have been making and testing similar products for many years and Bluetooth technology poses few new problems.

The measurements must be done in an anechoic chamber using a reference generator or software that will set the Bluetooth antennas to a single frequency. By changing the orientation of the system under test relative to the receiving antenna for the test equipment, a three-dimensional plot of signal strength can be generated. This gives a relative indication of range in each direction and can help the system designer tune the antenna placement to get close to a uniformly strong signal in all directions.

In the real world, there is an effect called multi-path that helps to fill in the nulls and weak areas in these measured radiation patterns. This occurs when radio waves bounce around, reflecting off walls and other surfaces, to fill in

the areas that are not covered well by direct radiation from the device in the system.

It should be pointed out that in a system where one antenna is used for both the receiver and transmitter, these measurements apply to both uses.

Build to Order/Configure to Order

Notebook computer manufacturers have moved away from the fixed configuration model to an environment where they can deliver systems built to the customer's specification. There are two ways that this is done:

- Build to Order (BTO)
- Configure to Order (CTO)

Both of these ways allow for features to be added to a system to meet a customer's specific request at some stage during the manufacturing process. CTO is a model where a technician simply adds devices after the system is basically complete. BTO is when these configuration decisions are made earlier in the assembly process. BTOs can result in greater cost savings to the manufacturer but are more complex and result in longer turnaround times.

When evaluating the integration of Bluetooth technology as a build option, there are three design pieces to consider:

1. mounting a Bluetooth module
2. mounting the antenna
3. the interconnect between the two

The module should be designed with sockets, and sockets need to be available that can carry the 2.4GHz RF signal if the antenna is not actually on the hardware module.

Ideally these sockets should be available from multiple vendors and have the same pinout as the module. In this way, the notebook manufacturer has the ability to start manufacturing using sockets, and then to switch to directly mounting the module when the Bluetooth product attach ratio makes sense. This should be able to be done without changing the system board.

Depending on the cost of the antenna, it could either be a fixed part of the system or it could be added during the assembly process, if the customer wants the Bluetooth feature. The easiest way to do this is to include the antenna and module on the same small plug-in board to be attached as a single unit when ordered. Two added benefits of this approach is that it automatically keeps the interconnect between the module and antenna to a minimum, thus maximizing performance, and secondly, it does away with the cost of complex cabling and RF connectors that could be required if the antenna and module were attached separately. The downside is the impact it has on the system's mechanical design.

BLUETOOTH PRODUCTS ON AIRCRAFT

In the United States, the Federal Aviation Administration states in the section of its regulations covering portable electronic devices (FAR 91.21)[1] that

"a) Except as provided in paragraph (b) of this section, no person may operate, nor may any operator or pilot in command of an aircraft allow the operation of, any portable electronic device on any of the following U.S-registered civil aircraft:

> (1) Aircraft operated by a holder of an air carrier operating certificate or an operating certificate; or
>
> (2) Any other aircraft while it is operated under IFR.

(b) Paragraph (a) of this section does not apply to--

> (1) portable voice recorders;
>
> (2) hearing aids;
>
> (3) heart pacemakers;
>
> (4) electric shavers; or
>
> (5) any other portable electronic device that the operator of the aircraft has determined will not cause interference with the navigation or communication system of the aircraft on which it is to be used.

(c) In the case of an aircraft operated by a holder of an air carrier operating certificate or an operating certificate, the determination required by paragraph (b)(5) of this section shall be made by that operator of the aircraft on which the particular device is to be used. In the case of other aircraft, the determination may be made by the pilot in command or other operator of the aircraft."

To paraphrase: it is up to the **airlines** to decide whether Bluetooth devices can be operated on their aircraft. To this end the Bluetooth SIG has been running tests with many of the major airlines with the intent of proving that Bluetooth devices are safe for use on board aircraft. Although these tests are going well, we do not expect that all airlines will approve Bluetooth technology before it is introduced.

To avoid the situation where airlines would consider banning the use of notebooks because they could contain Bluetooth technology, we have driven a strategy of having a cellphone-like on/off mechanism with a positive indication of when the Bluetooth device is powered or not. This provides the user an easy mechanism for preventing the operation of the radio.

There are, however, still some conflicts between the Bluetooth strategy of devices talking to each other automatically without user interaction, and the requirement to prevent operation of personal electronic devices below 10,000 feet. By working closely with the airlines and regulatory bodies, we are optimistic that these issues will be successfully closed. One indication of this is the recent approval in Germany of an 802.11 wireless

network for use on board Lufthansa aircraft. The approval was granted to a system that allows cabin crew to communicate with an on-board server while they are moving around the aircraft. The most interesting part of this approval is that it was granted for all altitudes—not just above 10,000 feet (as one would assume).

REGULATORY: THE MODULE APPROVAL CONCEPT

Bluetooth technology is intended to be a worldwide initiative, but wireless (intentional emitters) is one of the more regulated technologies in the world. Countries tend to allocate their radio spectrum differently and police that allocation very aggressively.

By using the same frequency as Microwave ovens (Industrial, Scientific and Medical, ISM band) and by being low power, Bluetooth systems are easier for countries to accept, but Bluetooth devices will still have to be tested and registered with the individual bureaus in each country.

For a manufacturer of a Bluetooth product to have to go through the regulatory process for every country that the product is to be sold in is, to say the least, very burdensome and would add time to the critical period between when a product is complete and the time when it can be launched.

To address this problem, the "Module Approval" (MA) process for Bluetooth devices has been developed. This process allows a Bluetooth radio module and an antenna to be approved once in each country. Later, when a system manufacturer integrates that module and antenna into a design, he/she can avoid having to repeat the process by referring to the prior approval.

To support this, the module is buffered from the main system in the following ways:

- The module has an on-board voltage regulator so that irrespective of how the module gets powered, it cannot affect the power of the transmitter.
- A clock on the module generates the radio frequencies. This guarantees that the radio will always operate within the target spectrum.
- Transmitted data on the module are buffered to remove the possibility of the host system affecting the behavior of the module by feeding it data at different rates.

Under the MA process, the system that is submitted for testing has an extremely short connection between the antenna and the module. This gives the integrator of the module scope to increase the length of this connection without impacting the right to use the MA.

At the time of this writing, there are still some regulatory challenges being worked out by the Bluetooth SIG and some of its member companies. The MA concept has not been adopted worldwide, and in some countries other procedures are likely to be required. However, the SIG has had major successes in influencing changes in Japan and Spain to open up the 2.4GHz spectrum, and this activity is ongoing. For the current status refer to the Bluetooth SIG web page (www.Bluetooth.com).

Qualification

While the MA is under the jurisdiction of the regulatory bodies of the various governments, the Bluetooth SIG runs the qualification program. When a company signs on as a member, there is an agreement that in order to have access to the intellectual property in the technology, that company must pass the SIGs Qualification Program.

There are two types of testing that a product can be subjected to. The first is the very specific testing of low-level interfaces and protocols. The second is the higher level testing of "profiles." A system manufacturer will claim that a product supports certain usage models, for example, file transfer. Specific tests must be done to ensure that the product will 'do' file transfer with all types of other Bluetooth devices that this product would be expected to work with.

A formal qualification process has been designed for Bluetooth products. From an integrator's perspective, the two key acronyms are BQTF and BQB.

BQTF is the Bluetooth Qualification Test Facility. This is a test body that has been accredited by the SIG to carry out the prescribed tests. The BQTF prepares documentation showing how a submitted system fared and this is handed onto a BQB.

BQB is the Bluetooth Qualification Body. This is an individual who will review the results generated by the BTQF and either grant or deny the Bluetooth logo to a product.

Again, this is still very much a work in progress. For current information, the best resource is the SIG's web page.

CHALLENGES

The Bluetooth initiative has generated a huge amount of interest and there is a lot of excitement about getting products to market. However, getting products to market is contingent on the SIG completing and solidifying some of its work. The three main areas that stand out as

needing work prior to products being launched are as follows:

- *Worldwide RF regulatory process*. At present, as an alternative, manufacturers can submit products to every country for approval but this is a long and arduous activity.
- *Bluetooth SIG qualification process*. While this has been defined, it needs to be tested and tuned with early products.
- *Documented position on airline acceptance of Bluetooth technology*. We do not expect it to be legal to operate Bluetooth devices on aircraft for some time, but we do need written closure on the cellphone-like power switch.

With so many companies and industries involved it is important to keep the perspective of a work-in-progress. Companies willing to maintain involvement through specification changes and process development will be in a great position to capitalize on being the first to market when the current issues are all closed.

RESULTS

In this paper, we discussed all aspects of integrating Bluetooth technology into a notebook computer system and have hopefully given some insight into a program with so many interrelated activities running in parallel.

On the one hand, the excitement generated by the technology and the consequent desire to get products to market is driving the SIG activities. On the other hand, the technology has such broad-based support from its member companies that you can speculate that this is helping the SIG to drive some of the more complex political issues such as worldwide regulations.

Another behavior which is interesting to note is the focus placed on interoperability. The Bluetooth SIG realizes the penalties that other technology initiatives have paid when first products have been rushed to market without ensuring they interoperate correctly.

DISCUSSION

From the perspective of integrating Bluetooth technology into a notebook PC, it is interesting that much of the technology can be regarded as a black box. The integrator does not need to understand the detailed behavior of the hardware or software if the external interfaces are clear and the integrator knows how to hook it up to the rest of the system. This is actually very similar to the way other system components are regarded. A graphics controller is probably similar in that there is a piece of hardware and a piece of software. They can both be extremely complex, but if the integration process is well understood, there is no need for the notebook PC manufacturer to understand the details of how the device operates internally.

It is the external approval aspects of the Bluetooth technology that present the biggest challenge. To get products to market as quicky as possible, these tasks need to be structured in such a way that notebook OEMs do not have to become experts. The detailed procedures have to be converted into simple, crisp activities with known durations so they can be factored into existing system development schedules.

Undeniably, there is also some risk associated with being on the leading ege of this technology. Early this year, some issues were identified with the Host Controller Interface specification. This required changes to the specification and the hardware that was designed to go with it. If this kind of change had occurred later it would have had a bigger effect. The goal of the program is that all such issues will be identified before or during the first wave of qualification tests and that the extent of this testing is related to the number of devices and profiles that can be fully tested at that time.

It is therefore interesting to note that it is in the interests of all Bluetooth SIG members to have a large number of different products ready for the first wave of qualification tests before the official launch.

CONCLUSION

As one of the founding members of the Bluetooth SIG, Intel wants to help and encourage Notebook PC OEMs to integrate the technology into their designs.

Using the results of the SIG activities, Intel is working to put together a clear set of detailed instructions on how notebook computer OEMs can integrate Bluetooth technology into products.

ACKNOWLEDGMENTS

I thank Gordon Chin, Jim Kardach, and Jeff Schiffer for reviewing and providing valuable inputs to this paper.

REFERENCES

[1] Federal Aviation Administration, “Federal Aviation Regulations.”

AUTHOR’S BIOGRAPHY

Graham Kirby is the Manager of the Bluetooth Enabling Program in the Mobile Computing Group at Intel Corporation . He holds a B.S. degree in CS from Loughborough University of Technology (UK) and has been in the PC industry for 18 years, the last 8 of which have been at Intel working on product and technology

development for mobile PCs. His e-mail is graham.d.kirby@intel.com.

Overview of IEEE 802.11b Security

Sultan Weatherspoon, Network Communications Group, Intel Corporation

Index words: 802.11b, wireless, WLAN, encryption, security

ABSTRACT

There is much regulatory and standards work in the area of security, especially in wireless. The wireless LAN standard IEEE 802.11b provides a mechanism for authentication and encryption. This paper describes the 802.11b security protocols and the implications they have for user privacy, ease of use, and import/export issues.

INTRODUCTION

Because wireless is a shared medium, everything that is transmitted or received over a wireless network can be intercepted. Encryption and authentication are always considered when developing a wireless networking system. The goal of adding these security features is to make wireless traffic as secure as wired traffic. The IEEE 802.11b standard provides a mechanism to do this by encrypting the traffic and authenticating nodes via the Wired Equivalent Privacy (WEP) protocol.

Whenever encryption and authentication are implemented in any system, three things must be considered:

1. *The customers need for privacy*. How strong do the protocols need to be and how much should they cost (MIPS, dollars, and time).
2. *Ease of use*. If the security implementation is too difficult to use, then it will not be used.
3. *Government regulations*. Encryption is viewed as munitions by many governments, including the US, so all encryption products are export controlled.

The WEP protocol used in 802.11b balances all the above-mentioned considerations.

WIRED VS. WIRELESS IMPLICATIONS

The main security issue with wireless networks, especially radio networks, is that wireless networks intentionally radiate data over an area that may exceed the limits of the area the organization physically controls. For instance, 802.11b radio waves at 2.4GHz easily penetrate building walls and are receivable from the facility's parking lot and possibly a few blocks away. Someone can passively retrieve all of a company's sensitive information by using the same wireless Network Interface Card (NIC) from a distance without being noticed by network security personnel.

This problem also exists with wired LAN networks, but to a lesser degree. Current flow through the wires emits electromagnetic waves that someone could receive by using sensitive listening equipment. However, a person would have to be much closer to the cable to receive the signal.

Most LAN adapters, wired and wireless, on the market today offer a "promiscuous mode" that, with off-the-shelf software, enables them to capture every packet on their segment of the LAN.

Many of the security issues facing wireless LANs are also issues facing wired LANs. Data transmitted on the wired LAN are incorrectly assumed to be protected because one needs to be physically in the building to access the network. This is largely untrue with corporate access to the Internet. Often, if users from inside can get out to the Internet, then hackers from outside can get into a network if proper precautions are not taken. There is also remote access for corporate travelers and telecommuters.

These examples illustrate that both wireless and wired networks are subject to the same security risks and have the same issues. These include the following:

- Threats to physical security of a network (e.g., denial of service and sabotage).
- Unauthorized access and eavesdropping.
- Attacks from within the network's (authorized) user community, (e.g., disgruntled, current, and ex-employees have been known to read, distribute, and alter valuable company data).

In addition, measures taken to ensure the integrity and security of data in the wired LAN environment are also applicable to the wireless LAN's as well. Wireless

LAN's, such as 802.11b, include an additional set of security elements, namely WEP, which are not available in the wired world. Therefore, some people are of the opinion that a properly implemented protected wireless LAN is more protected than a wired LAN.

802.11 WEP Protocol

The WEP algorithm was selected to meet the following criteria:

- *Reasonably strong*. It must meet customers' needs.
- *Self-synchronizing*. Stations quite frequently go in and out of coverage.
- *Computationally efficient*. The WEP algorithm may be implemented in hardware or software. If it is efficient, it allows low-MIPS devices to still implement it in software.
- *Exportable*. It can be exported outside the US and imported to other countries.
- *Optional*. It is an option not required in an 802.11-compliant system.

The same key is used to encrypt and decrypt the data. The WEP encryption algorithm (see Figure 1) works as follows:

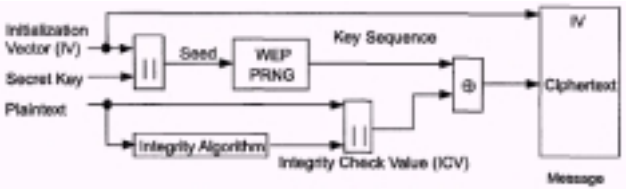

Figure 1: WEP encryption

Two processes are applied to the plaintext data. One encrypts the plaintext; the other protects against unauthorized data modification.

The secret key (40-bits) is concatenated with an initialization vector ("IV", 24-bits) resulting in a 64-bit total key size. The resulting key is input into the pseudorandom number generator (PRNG). The PRNG (RC4) outputs a pseudorandom key sequence based on the input key. The resulting sequence is used to encrypt the data by doing a bitwise XOR. This results in encrypted bytes equal in length to the number of data bytes that are to be transmitted in the expanded data plus 4 bytes. This is because the key sequence is used to protect the integrity check value (ICV, 32-bits) as well as the data.

To protect against unauthorized data modification, an integrity algorithm (CRC-32) operates on the plaintext to produce the ICV. The ciphertext is accomplished by doing the following. See Figure 3 for an example.

1. Compute the ICV using CRC-32 over the message plaintext.
2. Concatenate the ICV to the plaintext.
3. Choose a random initialization vector (IV) and concatenate this to the secret key.
4. Input the secret key+IV into the RC4 algorithm to produce a pseudorandom key sequence.
5. Encrypt the plaintext+ICV by doing a bitwise XOR with the pseudorandom key sequence under RC4 to produce the ciphertext.
6. Communicate the IV to the peer by placing it in front of the ciphertext.

The IV, plaintext, and ICV triplet forms the actual data sent in the data frame.

In decryption, the IV of the incoming message is used to generate the key sequence necessary to decrypt the incoming message (see Figure 2). Combining the ciphertext with the proper key sequence yields the original plaintext and ICV. The decryption is verified by performing the integrity check algorithm on the recovered plaintext and comparing the output ICV' to the ICV transmitted with the message. If ICV' is not equal to ICV, the received message is in error, and an error indication is sent to the MAC management and back to the sending station. Mobile units with erroneous messages (due to inability to decrypt) are not authenticated.

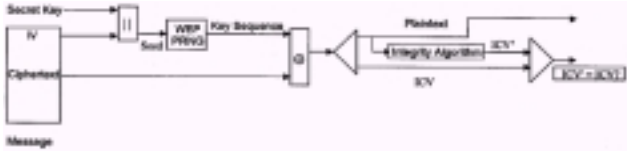

Figure 2: WEP decryption

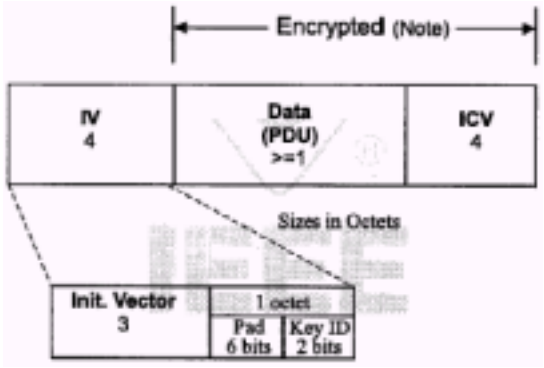

Figure 3: WEP data frames

The same shared key used to encrypt/decrypt the data frames is also used to authenticate the station. It is considered a security risk to have the encryption and

authentication keys be the same. There is also a method where stations and AP's can utilize WEP alone without shared key authentication, essentially using WEP as an encryption engine only. This is done in open system mode. This is considered to be the most protected implementation in 802.11 thus far, and it still enables reasonable authentication.

There are two types of 802.11 authentication:

- *Open system authentication*. This is the default authentication service that doesn't have authentication.
- *Shared key authentication*. This involves a shared secret key to authenticate the station to the AP.

The open system authentication is "null" authentication. The station can associate with any access point and listen to all data that are sent plaintext. This is usually implemented where ease-of-use is the main issue, and the network administrator does not want to deal with security at all.

The shared key authentication approach provides a better degree of authentication than the open system approach. For a station to utilize shared-key authentication, it must implement WEP. Figure 4 illustrates the operation of shared-key authentication. The secret shared key resides in each station's MIB in a write-only form and is therefore only available to the MAC coordinator. The 802.11 standard does not specify how to distribute the keys to each station, however.

The process is as follows:

1. A requesting station sends an Authentication frame to the access point ("AP").
2. When the AP receives an initial Authentication frame, the AP will reply with an Authentication frame containing 128 bytes of random challenge text generated by the WEP engine in standard form.
3. The requesting station will then copy the challenge text into an Authentication frame, encrypt it with a shared key, and then send the frame to the responding station.
4. The receiving AP will decrypt the value of the challenge text using the same shared key and compare it to the challenge text sent earlier. If a match occurs, the responding station will reply with an authentication indicating a successful authentication. If not, the responding AP will send a negative authentication.

The WEP PRNG (RC4) is the critical component of the WEP process, since it is the actual encryption engine. The IV extends the useful lifetime of the secret key and provides the self-synchronous property of the algorithm. The secret key remains constant while the IV changes periodically. Each new IV results in a new key sequence, thus there is a one-to-one correspondence between the IV and the output. The IV may change as frequently as every message, and since it travels with the message, the receiver will always be able to decrypt any message. Therefore the data of higher layer protocols (e.g., IP) are usually highly predictable. An eavesdropper can readily determine portions of the key sequence generated by the (Key, IV) pair. If the same pair is used for successive messages, this effect may reduce the degree of privacy. Changing the IV after each message is an easy way to preserve the effectiveness of WEP.

RC4 was developed in 1987 by Ron Rivest for RSA Data security. RC4 is a stream cipher that takes a fixed length key and produces a series of pseudorandom bits that are XOR'ed with the plaintext to produce ciphertext and vice versa. RC4 is used in the popular SSL Internet protocol and many other cryptography products. The benefits of using RC4 are as follows:

- The keystream is independent of the plaintext.
- Encryption and decryption are fast, about 10 times faster than DES.
- RC4 is simple enough that most programmers can quickly code it in software.
- RSADSI claims that RC4 is immune to differential and linear cryptoanalysis.

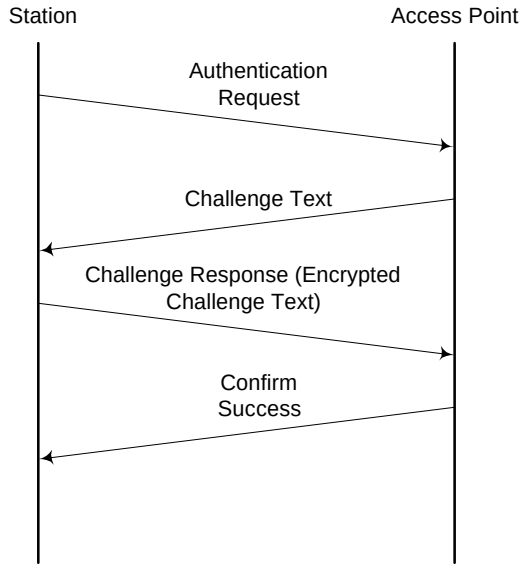

Figure 4: Shared-key authentication

SECURITY IMPLEMENTATION FOR BLUETOOTH*

Intel is a key contributor and supporter of the Bluetooth standard. This is a brief description of the implementation of its security system.

Bluetooth Security Implementation

Bluetooth technology provides three modes of security:

- *Security mode 1 (non-secure).* A device does not initiate any security procedure such as encryption or authentication.
- *Security mode 2 (service-level enforcement security).* A device does not initiate security procedures before channel establishment at the L2CAP (service) level. This mode allows different and flexible access policies for applications, and is used especially for running applications with different security requirements in parallel.
- *Security mode 3 (link-level enforced security).* A device allows only authenticated connections.

Bluetooth technology has three security attributes: authorization, authentication, and encryption.

Since there are many services that a Bluetooth device might have, there is a database of which services a device has authorization to use. The user can choose to "auto" trust devices or "manually" trust devices.

Since the identity of the remote device is used as a condition for authorization, authentication is performed. Authentication is accomplished using a challenge-response scheme using symmetric link keys. If the devices do not share a link key, one is created through a process called "pairing" and it is based on a shared secret association, such as a PIN code. If a device does not have a mechanism to enter a PIN, a restricted form link key, called a unit key, is generated based on the device's address and random number.

Encryption can only be activated after authentication. Encryption is based on a stream cipher easily implemented in hardware or software.

SPREAD SPECTRUM

Both 802.11b and Bluetooth operate in the worldwide available 2.4GHz spectrum and use spread spectrum technologies. Spread spectrum was initially developed by the US military to send unbreakable codes that were either hard to detect or hard to jam. Some vendors might say their systems are secure because they use spread spectrum technology but when the spreading sequence is known to everyone, there is very little security gained from spread spectrum by itself.

Impact on Ease of Use

For 802.11b and Bluetooth, ease of use was always considered to be a key factor in choosing the implementation. As mentioned before, if security is not transparent to the application and easy to use, it won't be used. This is proven by the fact that both 802.11b and Bluetooth security implementations have encryption as an option. There are clearly scenarios where encryption is not required nor wanted and others where encryption is needed.

Both of these implementations also have varying levels of security. The greater the level of security, the more complex the implementation seems to be. If users want very strong security, they will need to possess some knowledge of security mechanisms (shared secret entered by hand). If users do not want any security, they need not possess any knowledge of the underlying security, because there is none.

Another issue that affects ease of use is how to distribute the initial shared secret key. The shared secret in 802.11b is 40-bits. Shared secret keys are normally passwords or the like. They either need to be told to everyone in the peer group or distributed in a secure fashion.

Regulatory Issues

The United States government is constantly changing its export laws. The laws on encryption are no different. An interesting note is that the US government will let a company export any sort of product with authentication without regulation. Up to this point, the US government views encryption as munitions, similar to a nuclear bomb!

However, the government has recently relaxed the rules on export. Now a company can export an encryption product with an infinite size encryption key as long as the company complies with the following rules:

- It must have a one-time technical review with the Department of Export.
- It must not sell to terrorist countries.
- It must not sell directly to foreign governments. (However, if the product obtains "retail" classification, then the company can sell to foreign governments.)

The law used to be that one could only export up to 56-bit encryption. The 802.11b standard specified up to only

* Bluetooth is a trademark owned by its proprietor and used by Intel under license.

40-bit solely for export reasons. When exporting, though, one also needs to comply with a country's import rules.

Many of the encryption algorithms are well known and thus implemented outside the US. So some of the export/import rules can inhibit a US company from competing effectively in the world market.

FUTURE WORK

Intel and other companies are working together to propose increased security extensions to the 802.11 standard to increase the authentication, encryption, and key exchange.

From a security point of view it is more secure to have separate keys for encryption and authentication. It is even suggested that a different encryption key be used for sending and receiving.

Since 802.11 is a true wireless LAN, meaning it is designed to connect to the wired LAN, many of the security protocols applied to the wired LAN's are directly applicable to 802.11.

Here are some of the security mechanisms that will likely be implemented:

- 128 bit WEP encryption. This is already implemented by all of the major vendors but has not been standardized yet. It is a 104-bit key with a 24-bit IV.
- Standard key exchange and key distribution. Currently, 802.11 uses shared secrets that are physically entered. This can be done by a number of different protocols like RADIUS, Kerberos, SSL, and IPSEC.
- Improved data integrity via keyed MAC. This is a mechanism to ensure each packet of data has not been tampered with and is authentic. The CRC in 802.11b was designed to detect errors in a data frame while a keyed MAC will provide better authenticity and integrity.
- Compatibility with existing wired LAN infrastructure standards such as manageability and wake on LAN.

CONCLUSION

When defining a security system, companies need to first define what type of attacks to protect against and what they are willing to pay in MIPS, dollars, and time. As users' needs change and the requirements for security become more aggressive, 802.11 is poised to accept the challenge.

ACKNOWLEDGMENTS

The author acknowledges the help of Duncan Kitchin, Jesse Walker, and Jon Inouye.

REFERENCES

[1] Jim Geir, *Wireless LANs: Implementing Interoperable Networks*, MacMillan Technical Publishing, USA, pp. 25-26, 138-142.

[2] Wireless LAN Security White Paper, http://www.wlana.com.

[3] HomeRF Technical Committee, "Shared Wireless Access Protocol Specification," Revisions 1.09.

[4] Bluetooth White paper, "Bluetooth Security Architecture," Version 1.0.

[5] IEEE Standards Board, "802 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications."

[3] Bruce Schneier, *Applied Cryptography*, John Wiley & Sons, Inc, USA and Canada, pp. 397-398.

AUTHOR'S BIOGRAPHY

Sultan Weatherspoon is a technical marketing engineer with the Wireless LAN Operation in Intel Oregon. He is one of the original members of WLO that investigated WLAN, which eventually lead to the formation of the group and a $100 million investment in Symbol Technologies. He earned his B.S. degree in electrical engineering from the University of Michigan. His e-mail is sultan.weatherspoon@intel.com.

The Evolution of Third-Generation Cellular Standards

Phillip Ames, Wireless Communications and Computing Group, Intel Corporation
John Gabor, Wireless Communications and Computing Group, Intel Corporation

Index words: cellular, wireless, standard, generation, radio

ABSTRACT

This paper provides an overview of the evolutionary path of mobile/terrestrial cellular standards, leading up to the now defined third-generation cellular standards.

Commercial mobile cellular systems first became available in the early 1980's. These first systems were deployed, utilizing analog technology over circuit-switched networks. They had very limited features, poor voice quality, and limited radio coverage, although they have vastly improved over the last two decades and are still widely deployed around the world. In addition, data transfer was limited to 9600 baud.

In the early 1990's, the second generation of mobile cellular systems was introduced. Based upon digital technology and still utilizing the circuit-switched network, new features and services were introduced. Speech quality, although not equivalent to that of analog, was digitized through a low 8kbps bitrate vocoder that used Code Excited Linear Projection (CELP) technology. This has also been enhanced, over the past five years, to the extent that speech quality now exceeds that of the FM analog systems.

The immediate motivating factor for the third-generation communications systems is to increase system capacity. The overall number of users is exceeding the radio spectrum allocated to the second-generation; however, receiving and sending data are the essential building blocks to widespread mobile Internet access and to mobile data transfer, capabilities enabled with the third-generation. The third-generation is a generic term used for the next generation of mobile communications systems, often referred to simply as "3G." 3G mobile systems will provide enhanced services such as voice, text, and high-speed data, with 144kbps as an overall goal. The technology involved in the deployment of 3G systems and services is currently under development throughout the industry.

Attaining the goals of 3G will be an evolutionary migration from the installed 2G systems. Operators are phasing in new enhanced 2G capabilities, preparing for 3G services, and attempting to provide a seamless transition from existing digital systems, including full backwards compatibility. From a consumer perspective, this integration of system and service profiles, along with multi-mode terminals will mean worldwide roaming possibilities. 3G systems offer up to fifteen times the network capacity of analog networks. With the initial third-generation networks due to be launched in Japan in early 2001, and with European countries following in early 2002, 3G is already in sight. To enable third-generation capabilities, especially worldwide roaming, the radio interface specifications need to be defined and adopted, and complete interoperability needs to be finalized.

INTRODUCTION

While no one can predict the future, it is certain that the way we communicate in the future will be vastly different from today. Video-on-demand, high-speed multimedia, and mobile Internet are just a few of the communication possibilities. Third-generation systems will expand the possibilities of information transfer and communication. "Third Generation" is a term given to wireless services that, for example, allow users to make video calls from a mobile terminal, while simultaneously accessing a remote database, or while receiving e-mails and phone calls. The foundation for these services has already been laid in the existing structure of today's digital mobile phone networks. What is needed in order to support these advanced multimedia services is to expand the information capacity, or "bandwidth" of the wireless links.

While conversational speech is still the main service of today's mobile systems, support for the data communications over-the-air interface is quickly increasing. To implement this capability for the market, radio interface standards must be defined and adopted worldwide. The process of developing these standards has taken years of effort by hundreds of participating

companies and government agencies around the world. The First-Generation (1G) systems used analog technology. The current handsets, widely deployed today, use Second-Generation (2G) technology, often referred to as "digital." The Third-Generation systems extend the voice-only digital from 2G (as enhanced), and incorporate additional data capability. During the transition from 2G to 3G there will be an interim deployment of 2.5G digital technology with limited data capabilities, such as short messaging services (ability to send and receive short text messages from a cellular system).

The process of developing standards provides independent companies with an opportunity to influence the standards in such a way that their respective Intellectual Property Rights (IPR's) will be adopted. This process also provides companies insight into the future direction of communications, and it gives them information that allows them to prepare for and to make advanced engineering and marketing decisions.

Developing international standards to which all participating members can agree requires a good deal of time, study, and patience. The process is slow and is consensus-driven. This paper addresses the Mobile/Cellular Standards process, highlights the key Standards Organizations, and provides an overview of the 3G Radio Interface Standards (see Figure 1).

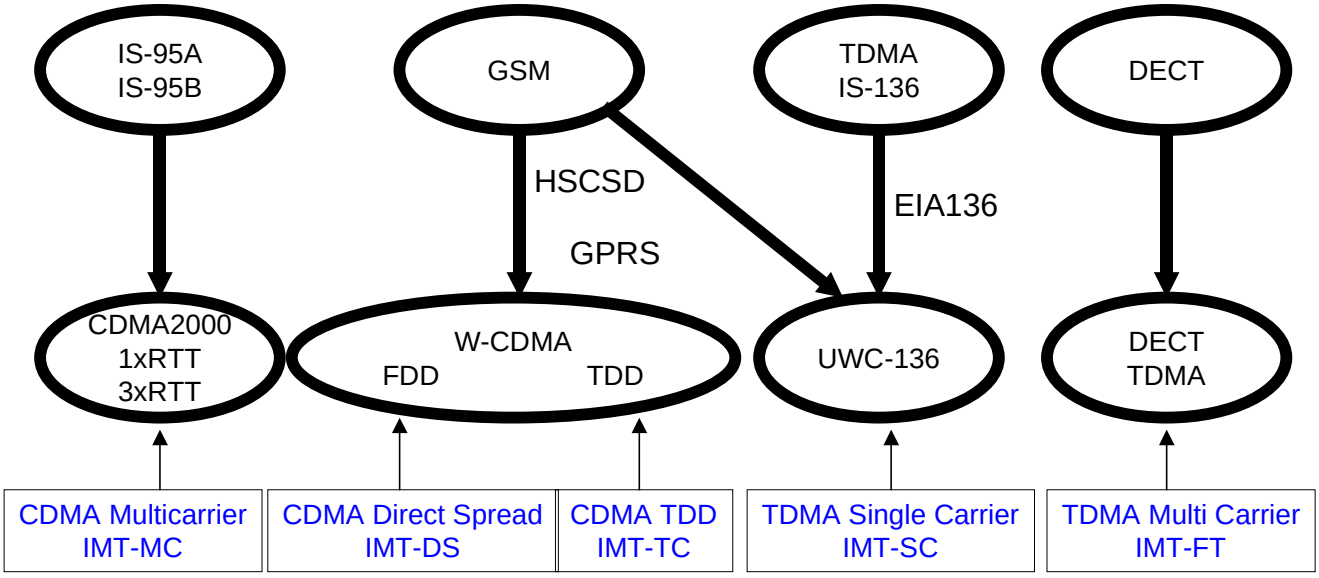

Figure 1: The evolution of mobile cellular standards

MOBILE/CELLULAR STANDARDS PROCESS

The standards organizations and partnership projects provide technical input to the global standards developer for ratification and approval.

The International Telecommunications Union (ITU), a charter organization of the United Nations, is the pre-eminent global standards developer for telecommunications. The ITU-R (radio communications sector) addresses terrestrial and space (satellite) radio communication. Standards development organizations and partnership projects, listed below, provide technical input to the ITU for ratification and approval.

Standards Development Organizations

Standards development organizations (SDOs) are national or multi-national organizations, actively involved in defining the next-generation wireless standards, along with refining the ongoing remedial editing of existing standards SDOs are comprised of various companies who work together to promote specification proposals. SDOs also study radio spectrum utilization, including such subsets as intersystem issues, emergency services, and accommodations for the disabled. They also coordinate and cooperate with the ITU on standardization of radio systems in the field of telecommunications. The coordination and cooperation issues are managed by "Harmonization" groups. The following is a list of Western standard development organizations, along with their respective areas of geographical and technical interests:

- The European Telecommunications Standards Institute (ETSI) is defining a technology standard for 3G called the Universal Mobile Telecommunications Systems (UMTS).
- The Japan Association of Radio Industries and Business (ARIB) primarily focuses on WCDMA for IMT-2000.

- The primary Canadian SDO is the Telecommunications Standards Advisory Council of Canada (TSACC).
- The American National Standards Institute (ANSI) is a US repository for standards considered to be semi-permanent, a nebulous term for "longer than interim."
- The United States Telecommunications Industry Association (TIA) and T1 have presented several technology proposals on WCDMA, TDMA UWC-136 (based upon D-AMPS IS-136), and cdma2000 (based upon IS-95). The American National Standards Institute (ANSI) accredits both TIA and T1. The primary standards working groups are TR45 (Mobile & Personal Communications 900 & 1800 Standards and TR46 (Mobile & Personal Communications 1800 only Standards).

The Asian standards development organizations include the Korean Telecommunications Technology Association (TTA) and China Wireless Telecommunications Standards Group (CWTS), Partnership Projects.

The Third-Generation Partnership Project (3GPP) was formed by SDOs and other related standards' bodies to harmonize European, Asian, and North American standards proposals, and to define a complete set of global technical specifications for third-generation mobile systems based upon the evolved GSM core networks and radio access technologies. The project is better known as "3GPP."

3GPP is comprised of the following SDOs: ARIB (Japan), CWTS (China), ETSI (Europe), T1 (USA), and TTA (Korea). The project is divided into several technical specification groups (TSG's), with each TSG having multiple working groups, each responsible for defining an aspect of the third-generation standard. 3GPP is chartered to define the radio access network, the core network (including mobility management and global roaming), terminal access to the network (including specifications for the user identification module), and systems and services. (Additional information is available at the 3GPP web site *http://www.3gpp.org*.

The Third-Generation Partnership Project 2 (3GPP2) was organized by the SDOs that were concentrating on the development and evolution of the American National Standard (ANSI/TIA-41 core networks) and the relevant radio access technologies. The five SDOs are ARIB (Japan), CWTS (China), TIA (USA), TTA (Korea), and TTC (Japan). Similar to 3GPP, 3GPP2 is also comprised of several technical specification groups, each with multiple working groups. (Additional information is available at the 3GPP2 web site *http://www.3gpp2.org*.)

Both 3GPP and 3GPP2 organizations recognize the ITU as the preeminent organization for international telecommunications standardization, and they have agreed to submit all results to the ITU for consideration and approval.

THE MIGRATION TO 3G

The genesis of today's wireless technology began in the early 1980's with the introduction of the first mobile cellular handsets. These systems utilized analog interface technology and supported voice-only capabilities. This technology is still used in many parts of the world; however, it is limited in bandwidth and is low in quality. With the high demand for cell phones and the increased need for enhanced quality and more features, the Second Generation was introduced. 2G is primarily voice only, but does provide higher bandwidth, better voice quality, and limited data services that use packet data technology. 2G systems are currently in wide deployment with enhanced 2G systems currently available on the market. The 3G standardization process is coming to closure, with the recent completion of the IMT-2000 radio interface recommendations.

First-Generation Mobile Standards

The first generation of cellular wireless communications was based on analog technology and progressively became available to the consumer during the late 1970's and early 1980's. The most successful analog systems are based on the following standards, all of which are still in demand today:

Nordic Mobile Telephone (NMT) was the first commercially available analog system, introduced in Sweden and Norway in 1979.

Advanced Mobile Phone Service (AMPS) was launched in 1982. This has proven to be the most successful analog standard of all. AMPS networks are widely deployed and can be found on all continents.

Total Access Communications System (TACS) was originally specified for the United Kingdom and is based on AMPS. The original TACS specification was extended and is known as ETACS. ETACS is primarily deployed in Asia Pacific regions.

Second-Generation Mobile Standards

The second-generation (also known as 2G) introduced digital wireless standards that concentrated on improving voice quality, coverage, and capacity. The 2G standards were defined and designed to support voice and low-rate data only—Internet browsing was in its infancy during the definition stage. The world's four primary mobile digital wireless standards currently deployed around the world

are GSM, TDMA (IS-136), CDMA (IS-95-B), and PDC, all supporting data rates up to 9.6kbps.

Global System for Mobile phone communications (GSM) was the first commercially available digital standard, introduced in 1992. GSM relies on circuit-switched data. The basic development of supporting data at low bit-rates (<9.6 kbps) was introduced at the beginning of commercial services and has been predominantly used for e-mailing from laptop computers. [2]

Time Division Multiple Access, originally IS-54 and now IS-136 (TDMA IS-136), is sometimes referred to as the "North American" digital standard; however, it is also deployed in Latin America, Asia Pacific, and Eastern Europe.

Personal Digital Communications (PDC) is the primary digital standard in Japan.

IS-95 is based on "narrowband" (referred to as narrowband because of the limited amount of information that can flow through these networks) Code Division Multiple Access (CDMA) technology. It has become popular in South Korea and North America.

Enhanced Second-Generation Mobile Standards

Enhanced second-generation (sometimes referred to as 2.5G or 2+G) builds upon the second-generation standards by providing increased bit-rates and bringing limited data capability. Data rates range from 57.6kbps to 171.2kbps.

High-Speed Circuit-Switched Data (HSCSD) provides access to four channels simultaneously, theoretically providing four times the bandwidth (57.6) of a standard circuit-switched data transmission of 14.4kbps.

D-AMPS IS-136B Time Division Multiple Access (TDMA) is the intermediate step to Universal Wireless Communication (UWC-136), a third-generation standard. The first phase of D-AMPS will provide up to 64kbps. The second phase will provide up to 115kbps in a mobile environment.

General Packet Radio System (GPRS) is an evolutionary path for GSM and IS-136 TDMA to UWC-136. It is a standard from the European Telecommunications Standards Institute (ETSI) on packet data in GSM systems. The Telecommunications Industry Association (TIA), as the packet-data SDO for TDMA-136 systems, has also accepted GPRS. GPRS supports theoretical data rates up to 171.2kbps by utilizing all eight channels simultaneously. This data rate is roughly three times faster than today's fixed telecommunication networks and about ten times as fast as current circuit-switched data services on GSM networks. GPRS is a universal packet-switched data service in GSM. It involves overlaying a packet-based air interface on the existing circuit-switched GSM network. Packet switching means that GPRS radio resources are used only when users are actually sending or receiving data. Using GPRS, the information is split into separate but related packets before being transmitted and subsequently reassembled at the receiving end. GPRS is a non-voice-added service that allows information to be sent and received across multiple mobile telephone networks. It supplements today's circuit-switched data and short messaging service. GPRS uses packet data technology, a fundamental change from circuit-switched technology, to transfer information. It also facilitates instant connection capability, sometimes referred to as "always connected." Immediacy is one of the key advantages of GPRS. Immediacy enables time-critical application services [5][6].

Third-Generation Mobile Standards

Third-generation systems will provide wide-area coverage at 384kbps and local area coverage up to 2Mbps. The primary motivation for the development of third-generation wireless communications is the ability to supplement standardized 2G and 2G+ services with wideband services. Essentially, this offers voice plus data capability.

The existing array of incompatible second-generation technologies, together with the restricted amount of information that can be transferred over these narrowband systems, prompted the ITU to work towards defining a new global standard for the next-generation broadband mobile telecommunication systems. Known as IMT-2000 (International Mobile Telecommunications-2000), the project was started to attain authorship of a set of globally harmonized standards for broadband mobile communications. The first set of IMT-2000 recommendations was recently approved by the ITU.

IMT-2000 is the term used by the International Telecommunications Union for this set of globally harmonized standards. The initiative was to define the goal of accessing the global telecommunication infrastructure through both satellite and terrestrial mobile systems. IMT-2000 has reflected the explosion of mobile usage and the need for future high-speed data communications, with wideband mobile submissions. IMT-2000 is a flexible standard that allows operators around the world the freedom of radio access methods and of core networks so that they can openly implement and evolve their systems. How they do it depends on regulations and market requirements.

The recent IMT-2000 recommendation highlights five distinct mobile/terrestrial radio interface standards:

1. IMT-MC: CDMA Multi-Carrier (known as cdma2000 or IS-2000).
2. IMT-DS: CDMA Direct Spread (known as Wideband CMDA or WCDMA-FDD). This standard is intended for applications in public macro-cell and micro-cell environments. The Frequency Division Duplex (FDD) mode is used for symmetrical applications, i.e., those requiring the same amount of radio resources in the uplink as in the downlink. This standard is well supported by Japan's ARIB and GSM network operators and vendors.
3. IMT-TC: CDMA TDD (WCDMA-TDD). Time Division Duplex (TDD) targets public micro-cell and pico-cell environments, and, due to severe interference-related considerations, is intended primarily for indoor use. This standard is optimized for symmetrical and asymmetrical applications with high data rates.
4. IMT-SC: TDMA Single Carrier (known as UWC-136 and EDGE). UWC-136 (Universal Wireless Communications) and EDGE (Enhanced Data Rates for GSM Evolution) will provide extended data services, with no changes to channel structure, frequency, or bandwidth. IMT-SC is the evolutionary path for GSM and TDMA-136, achieved by building upon enhanced versions of GSM and TDMA-136 technology. EDGE is a radio-based high-speed mobile data standard with aggregate transmission speeds of up to 384kbps when all eight timeslots are used.
5. IMT-FT: TDMA Multi-Carrier (well known as DECT, Digital Enhanced Cordless Telecommunication).

The IMT-2000 recommendations encompass three CDMA and two TDMA radio air interface standards.

Wideband Code Division Multiple Access (WCDMA) should not be confused with narrowband CDMA; they are completely different protocols. WCDMA is a younger technology, defined specifically to deliver high-speed data services and Internet-based packet-data at 3G data rates. WCDMA supports both packet and circuit-switched communications, such as Internet access and landline telephone services; however, WCDMA was defined with no requirements on second-generation backward compatibility.

WCDMA makes very efficient use of the available radio spectrum. No frequency planning is needed, since one-cell re-use is applied. Using techniques such as adaptive antenna arrays, hierarchical cell structures, and coherent demodulation, network capacity can be increased. In addition, circuit and packet-switched services can be combined on the same channel, allowing true multimedia services with multiple packet or circuit connections on a single terminal. WCDMA capacity is approximately double that of narrowband CDMA. The wider bandwidth and the use of both coherent demodulation and fast power control in the uplinks and the downlinks allow a lower receiver threshold. WCDMA uses a network protocol structure (signaling) similar to that of GSM; therefore, it will be able to use the existing GSM network as the core network infrastructure. [4]

In CDMA2000, a range of RF channel bandwidths are supported: 1.25, 3.75, 7.5, 11.25, and 15MHz. This range allows for support of a range of data rates as well as a high number of users.

In order to support higher bandwidth channels, CDMA2000 has defined two configuration options: Direct Spread (DS) and Multi-Carrier (MC). The DS option is similar to IS-95B and uses the entire bandwidth to spread the data for radio transmissions. In the MC option, user data is encoded as a single stream and de-multiplexed into multiple streams. Each stream carries part of the user data using a different carrier frequency signal, hence the name Multi-Carrier. The receiver will multiplex the received signals together before demodulation is carried out. Both the DS and MC options are available in the forward link only. The reverse link supports only the DS option. [3]

Time Division Multiple Access

One approach to reducing the number of confusing options to the end user and to improve the overall functionality of time-division cellular technology is to combine TDMA and CDMA radio air interface technology into one system. This combined approach, referred to as TD-CDMA, would retain some of the fundamental GSM-TDMA design parameters, such as frame and time-slot structure, which are key factors for interoperability and evolution. At the same time, the CDMA technology would add better interference averaging and frequency diversity. The combined approach would also merge the excellent spectral efficiency of CDMA, while retaining the robustness, planning principles, and well understood characteristics of TDMA-based GSM. [1]

In addition to the improvements of data throughput and interworking, 3G will provide an additional spectrum for the operators. The increase in 3G spectrum efficiency will also provide the operator with more throughput over limited resources. The transition from the existing 2G networks to 3G capabilities will evolve over time. Dual-mode terminals will attempt to provide seamless hand-over and roaming capabilities.

CONCLUSION

The goal of an unqualified single standard for implementation worldwide is not a reality. Operators have too much invested in their existing infrastructure and subscriber base; however, limited worldwide roaming will be possible with 3G. Even though the radio interfaces may be different, the handsets will support dual- or tri-mode operation, making the transition seamless from the subscriber's perspective.

Even with 3G radio interfaces, handsets will require dual- or even tri-mode operation, combining two or more radio standards to enable worldwide roaming. Intel is providing the building blocks to enable second- and third-generation wireless capabilities. The requirements to meet these increasing capabilities are higher performance, low-power microprocessors, highly integrated FLASH memory, and ASICs that support dual- or tri-mode standards. Intel is developing these high-performance semiconductor devices for use in RF equipment base-stations and cellular phones.

ABBREVIATIONS

The following table will help you navigate through the multiple acronyms in this paper.

2G	second-generation
3G	third-generation
3GPP	third-generation partnership project
3GPP2	third-generation partnership project 2
AMPS/ D-AMPS	advanced mobile phone system
ARIB	Association of Radio Industries and Broadcasting
EDGE	enhanced data rates for GSM and TDMA-136
ETSI	European Telecommunications Standards Institute
FDD	frequency division duplex
GPRS	general packet radio services
GSM	global system for mobile communication
HSCSD	high-speed circuit-switched data
IMT-2000	International Mobile Telecommunication-2000
ITU-R	International Telecommunication Union—Radio Communications
NMT	Nordic Mobile Telephone
PDC	personal digital communication
TACS	total access communications system
TDD	time division duplex
TDMA	time division multiple access
TIA	Telecommunications Industry Association
SDO	standards development organization
WCDMA	wideband code division multiple access
UMTS	universal mobile telecommunications system
UWC-136	universal wireless communications

REFERENCES

[1] Dropman, Ulrich, "A real step toward UMTS," http://w2.siemens.de/telcom/articles/e0497/497drop.htm.

[2] 3G – The Future of Communications, http://www.gsmworld.com/technology/3g_future.htm.

[3] Gorham, Peter, "Strategic Technology Steps on the CDMAONE Evolution Path to 3G CDMA2000," 3G Mobile Broadband Conference, August 10, 1999.

[4] Mercer Management Consulting, "3G Investment: How will it Prove in?," 3G Mobile Broadband Conference, August 10, 1999.

[5] Buckingham, Simon, "High Speed Circuit-Switched Data (HSCSD), http://www.mobileipworld.com/wp/ffz_wp3.htm.

[6] "An Overview of GPRS, http://www.gsmworld.com/technology/gprs.html.

AUTHORS' BIOGRAPHIES

Phil Ames is an Intel Sr. Staff Engineer with the Wireless Communications and Computing Group. He is currently the Mobile/Wireless Standards Manager and is contributing to the development of the industry's third-generation cellular standards. He has been with Intel since 1986 after graduating from Boston University. His email is phil.h.ames@intel.com.

John Gabor is Manager of Industry Relations for Intel's Cellular Communications Division. He is the chair for various groups within the TIA and 3GPP2. His background includes positions as Vice-President, Sales and Marketing, for two cellular telephone manufacturers, and as President of Houston-based Astro Business Communications, a Bell Atlantic acquisition. He has a Ph.D. in Managerial Psychology from University of Miami. His e-mail is john.gabor@intel.com.