Bluetooth Architecture Overview

Writer: Esteban Alcorn

Technical Reviewers: Jason Farmer, Vasu Pai, Don Dillard

Published: March 2011

Applies To: Windows Embedded Compact 7

Abstract

Describes the Bluetooth layered architecture as implemented on Windows Embedded Compact 7, and how it relates to the software implemented on Bluetooth controllers.

- The layered architecture of the Host Protocol Stack
- The supported Bluetooth Profiles
- The Host Controller Interface that connects the Host to a Bluetooth controller

Contents

Introduction

This article provides an overview of the layered Bluetooth architecture for Windows Embedded Compact 7. It describes the layers of this architecture and how they relate to each other, so that embedded OS designers and developers can identify the components that are referenced in the Bluetooth (http://go.microsoft.com/fwlink/?LinkId=209918) documentation for Windows Embedded Compact 7.

Each layer is a collection of components that include protocol implementations, APIs, applications, and services. The description of each layer includes its contents, purpose, and location in relation to the Bluetooth stack. These layers are divided into three primary components.

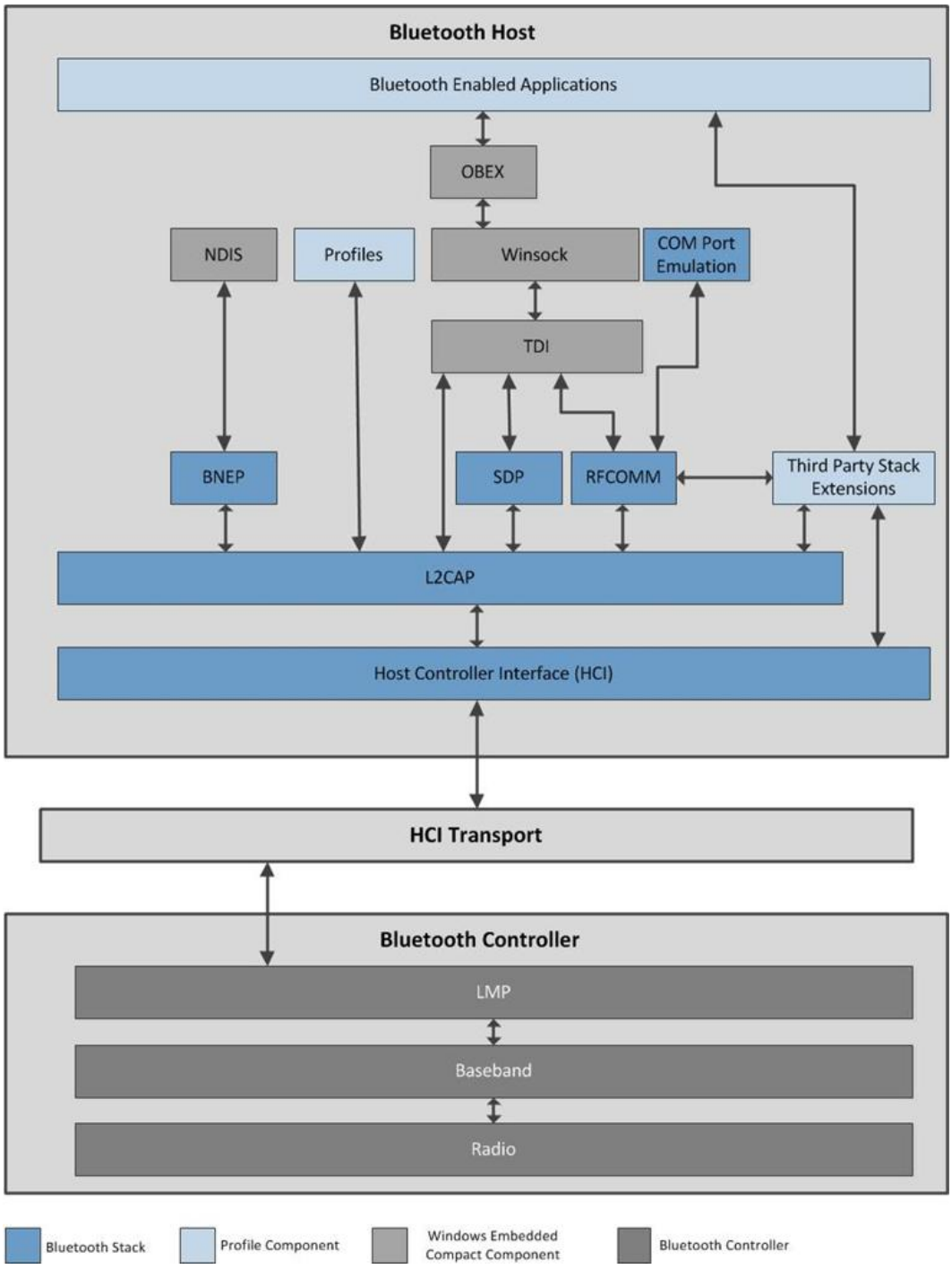
The following table describes these components from the highest to lowest level layers.

Component	Description
Bluetooth Host	Includes implementations of the core Bluetooth protocols: the Bluetooth stack and the high-level layers of the Bluetooth architecture, such as APIs and profiles.
Host Controller Interface (HCI) Transport Layer	Delivers data between the Bluetooth host and the Bluetooth controller.
Bluetooth Controller	A Bluetooth device that implements the lowest levels of the Bluetooth architecture.

Windows Embedded Compact 7 provides core implementations of the Bluetooth host and HCI transport layer that are compatible with Bluetooth 2.1 + EDR compliant Bluetooth controllers.

Figure 1 below displays the location of each layer in the Bluetooth architecture for Windows Embedded Compact 7.

Figure 1: Bluetooth layers



Bluetooth Host

The Bluetooth host contains the logical layers of the Bluetooth architecture. These layers include the core implementation of the Bluetooth stack and the layers that support and extend the functionality of the Bluetooth stack.

The following table describes the components of the Bluetooth host.

Component	Description
Bluetooth Stack	The Windows Embedded Compact implementation of the core Bluetooth protocols. These layers are based on the Bluetooth 2.1 + EDR protocol stack specification described on the Bluetooth SIG (Special Interest Group) website (http://go.microsoft.com/fwlink/?LinkId=182073).
Bluetooth Profile Component	Bluetooth profiles, stack extensions, and applications designed to extend the functionality of the Bluetooth stack. Includes layers implemented by Microsoft. You can also add your own layers to this component.
Windows Embedded Compact Component	The basic layers that support application development over the Bluetooth stack and provide the Bluetooth stack with the most common functionality. Networking and communications protocols and APIs used by the Bluetooth stack and Bluetooth-enabled applications. Not used exclusively for Bluetooth functionality.

Bluetooth Stack

In this article, the term "Bluetooth stack" (also known as "host protocol stack") refers to the Windows Embedded Compact implementation of the Bluetooth protocol stack. The Bluetooth protocol stack is a layered set of protocols that define the core Bluetooth functionality and is managed by the Bluetooth Special Interest Group (SIG). The Bluetooth stack is a Bluetooth 2.1 + EDR compliant stack. For information on the specification for the Bluetooth protocol stack, see the Bluetooth SIG website (http://go.microsoft.com/fwlink/?LinkId=182073).

On Windows Embedded Compact 7, the Bluetooth stack implements the host controller interface (HCI) and the host stack portion of the Bluetooth protocol stack. Each layer of the Bluetooth stack implements a unique protocol or service. For information on adding the Bluetooth stack to your OS, see Bluetooth Catalog Items and Sysgen Variables (http://go.microsoft.com/fwlink/?LinkId=209918). The following table describes the layers of the Bluetooth stack in Windows Embedded Compact 7.

Layer	Description
COM Port Emulation	An API that supports virtual COM ports over RFCOMM channels.
Service Discovery Protocol (SDP)	Manages the discovery and publishing of supported Bluetooth services and parameters between devices. Used to discover and advertise Bluetooth services, and to verify the compatibility of those services. Bound to the Logical Link Control and Adaption Protocol (L2CAP) layer.
Radio Frequency Communications (RFCOMM) Serial Port Emulation	Emulates RS-232 serial ports over the L2CAP layer. This protocol is based on the TS07.10 specification at the ETSI (European Telecommunications Standards Institute) website (http://go.microsoft.com/fwlink/?LinkId=209939). Includes the following functionality: • Base for COM port emulation • Base for derived point-to-point protocols • Implements multiplexing • Implements flow control • Carrier for attention (AT) commands • Transport layer for Object Exchange protocol (OBEX) over Bluetooth
Bluetooth Network Encapsulation Protocol (BNEP)	Encapsulates network packets into a standard format so they can be transmitted over the L2CAP layer. The Personal Area Network (PAN) profile uses BNEP to provide networking capabilities over a Bluetooth connection.
Logical Link Control and Adaption Protocol (L2CAP)	Communicates directly with the HCI. Converts data from high-level layers into a format that is

Layer	Description
	supported by lower-level layers of a Bluetooth controller. Provides the following services: • Packet SAR (Segmentation and Reassembly), which converts packets from high-level layers into packets supported by the Baseband layer • Protocol multiplexing • Group abstraction, which maps groups of addresses from high-level layers to piconets, which are supported by the Baseband layer
Host Controller Interface (HCI)	Provides the Bluetooth stack with access to a Bluetooth controller. An interface for Bluetooth hardware that is responsible for controller management, link establishment, and maintenance. Has direct access to the L2CAP layer. Communicates with a Bluetooth controller through the HCI transport layer.

Bluetooth Profile Component

The Bluetooth profile component contains the application layer and the layers for extending the functionality of the Bluetooth stack. This component contains layers that are implemented by Microsoft and all of the third-party layers of the Bluetooth architecture.

The following table describes the types of layers of the Bluetooth profile component.

Type	Description
Bluetooth-Enabled Applications	Applications that implement Bluetooth functionality. Includes Microsoft and third-party applications.
Profiles	Wireless interface specifications that describe possible Bluetooth applications and define how Bluetooth devices communicate. A number of Bluetooth profiles are available on Windows

Type	Description
	Embedded Compact 7. You can also create your own profiles.
Stack Extensions	Provides Bluetooth-enabled applications with access to the layers of the Bluetooth stack. You can extend the functionality of the Bluetooth stack by providing your applications with access to additional profiles.

Bluetooth-Enabled Applications

The primary way for an application to use Bluetooth technology is through the Winsock interface, which exposes the RFCOMM layer to the application. A Bluetooth-enabled application must then interface with certain profiles to support specific types of Bluetooth activity. For example, an application must interface with the human interface device (HID) profile to use a wireless keyboard over a Bluetooth connection.

Profiles

A Bluetooth profile is a specification that describes how devices must use Bluetooth protocols to implement a particular task. These descriptions detail how Bluetooth devices must communicate when an application implements specific functionality. For specifications on Bluetooth profiles, see the Bluetooth SIG website (http://go.microsoft.com/fwlink/?LinkId=182073).

The following table describes the profiles that are supported by Windows Embedded Compact 7.

Profile	Description
Generic Access Profile (GAP)	Defines the generic requirements for detecting and establishing a connection to a Bluetooth device. All profiles are based on GAP. A core profile that is available by default.
Generic Object Exchange Profile (GOEP)	Defines the requirements for OBEX usage models. A core profile that is available by default.
Serial Port Profile (SPP)	Defines the requirements for virtual serial port connections over the RFCOMM layer. Each serial port connection can connect two Bluetooth-enabled devices.

Profile	Description
	A core profile that is available by default.
Personal Area Network (PAN)	Defines standard IP-based network services over Bluetooth. Permits BNEP to run over a Bluetooth connection using Network Layer (OSI model) protocols. Defines the following roles and services: • PAN User (PANU) • Network Access Point (NAT) • Group Ad-Hoc Network (GN)
Human Interface Device (HID)	Defines communication with wireless HID devices over a Bluetooth connection.
Headset Profile (HSP)	Defines procedures that support interoperability between a headset and a mobile device, such as a cellular phone, that the headset controls through AT commands. Depends on SPP.
Object Push Profile (OPP)	Defines how to exchange small data objects, such as calling card exchanges, between Bluetooth devices using the OBEX protocol. Depends on GOEP.
Hands-Free Profile (HFP)	Defines how to use a mobile device, such as a cellular phone, in conjunction with a hands-free device over a Bluetooth link, so that the hands-free device can act as an audio input or output device for the mobile device. Depends on SPP.
File Transfer Profile (FTP)	Defines how to transfer data and manipulate objects on or with a peer Bluetooth device. Depends on GOEP.

Bluetooth Stack Extensions

You can create a Bluetooth stack extension layer that implements built-in data structures for accessing the layers of the Bluetooth stack. A stack extension layer exposes an interface from the Bluetooth stack to your Bluetooth-enabled applications.

Windows Embedded Compact Component

The Windows Embedded Compact component contains the layers of the Bluetooth host that support Bluetooth functionality but are not used exclusively by Bluetooth. These layers contain APIs and implement protocols for applications to interface with the Bluetooth stack. The layers in the Windows Embedded Compact component are some of the basic layers required to develop Bluetooth-enabled applications. The layers of the Windows Embedded Compact component also provide the Bluetooth stack and Bluetooth profiles with the networking and communications functionality of the OS.

The following table describes the layers of the Windows Embedded Compact component.

Layer	Description
Object Exchange (OBEX)	A communications protocol used to exchange binary objects between devices while using minimal resources. Works over Bluetooth and is defined by the Infrared Data Association (IrDA). Is the base for many high-level Bluetooth profiles such as Generic Object Exchange Profile (GOEP), Object Push Profile (OPP), and File Transfer Profile (FTP). For more information on OBEX, see Object Exchange Protocol Reference (http://go.microsoft.com/fwlink/?LinkId=209918).
Windows Sockets (Winsock)	An API that defines how software accesses networking services such as TCP/IP. Facilitates communication between networking applications, the TCP/IP, and Bluetooth protocol stacks. One of the primary interfaces for Bluetooth application development. For more information on Winsock, see Windows Sockets (http://go.microsoft.com/fwlink/?LinkId=209918).
Transport Driver Interface Layer (TDI)	An adaption layer for Winsock-based APIs. It is used by Winsock to communicate with the transport stack. This layer enables Winsock APIs to access the callback mechanism that is built into the Bluetooth stack.
Network Driver Interface Specification (NDIS)	An API for communicating with network interface cards. Used by the PAN profile. For information on NDIS, see Network Drivers

Layer	Description
	(http://go.microsoft.com/fwlink/?LinkId=209918).

HCI Transport Layer

The Bluetooth host and Bluetooth controller communicate through an intermediate layer called the HCI transport layer. This layer (also known as the host controller transport layer) contains a set of transport drivers that abstract and transfer data between the HCI layer of the Bluetooth host and the Bluetooth controller. The transport drivers implement communication between the Bluetooth Host and Bluetooth controller with a small set of functions that send and receive commands, data packets, and events.

The following table describes the transport drivers that make up the HCI transport layer.

Driver	Description
Bluetooth Universal Transport Manager (BthUniv)	An intermediate transport driver that detects and loads the appropriate built-in or third-party transport driver.
Built-In Transport Drivers	Built-in transport drivers that are supported by the HCI transport layer. The following list displays the supported transport drivers that can be found in %_WINCEROOT%\Public\Common\OAK\Drivers\Bluetooth\Transports. • USB – Bthusb.dll • UART – Bthuart.dll • BCSP – Bthcsr.dll

For more information on the HCI transport layer, see HCI Transport Layer Reference (http://go.microsoft.com/fwlink/?LinkId=209918).

Bluetooth Controller

The Bluetooth controller is a Bluetooth device such as a USB dongle. The Bluetooth controller implements the lowest level protocols from the Bluetooth protocol stack, including the physical layer and radio transceiver. This component is not provided by Windows Embedded Compact 7. For the specifications of these protocols, see the Bluetooth 2.1 + EDR protocol stack specification at the Bluetooth SIG website (http://go.microsoft.com/fwlink/?LinkId=182073).

The Bluetooth controller receives commands from the Bluetooth host through the HCI transport layer.

The following table describes the layers of the Bluetooth controller.

Layer	Description
Link Manager Protocol (LMP)	Manages logical link establishment between Bluetooth devices, which includes authentication and encryption. Creates, updates, and removes logical links and logical transports, and updates parameters for physical links to Bluetooth devices.
Baseband (Baseband)	A link controller that forms the physical layer of the Bluetooth architecture. Establishes and manages the physical radio frequency (RF) link between Bluetooth units that form a piconet.
Radio	A Bluetooth RF transceiver. Transmits data to and from the Baseband. This layer is the lowest level of the Bluetooth architecture.

Conclusion

The Bluetooth architecture for Windows Embedded Compact 7 is a layered architecture that consists of two primary components that are implemented by Microsoft: the Bluetooth host and the HCI transport layer. The Bluetooth host contains the logical layers such as the Bluetooth stack, Bluetooth profile component, and Windows Embedded Compact component. The HCI transport layer transfers data between the Bluetooth host and Bluetooth controller. The Bluetooth controller is the hardware component of the Bluetooth architecture and is not provided by Microsoft. The Bluetooth controller contains the physical layers of the Bluetooth architecture, such as the LMP, Baseband, and Radio layers.

Additional Resources

- Windows Embedded website (http://go.microsoft.com/fwlink/?LinkID=183524)

© 2011 Microsoft. All rights reserved.