

CATC Merlin's Wand™ 1.22

Bluetooth™ Test Generator

User's Manual



Document Revision 1.22

June 18, 2002

730-0019-00

CATC Merlin's Wand 1.22 Bluetooth Test Generator User's Manual, Document Revision 1.22

Document Disclaimer

The information contained in this document has been carefully checked and is believed to be reliable. However, no responsibility can be assumed for inaccuracies that may not have been detected.

CATC reserves the right to revise the information presented in this document without notice or penalty.

Trademarks and Servicemarks

CATC, *Merlin's Wand*, and *Merlin* are trademarks of Computer Access Technology Corporation.

Bluetooth is a trademark owned by Bluetooth SIG, Inc. and is used by Computer Access Technology Corporation under license.

Microsoft, *Windows*, and *Windows NT* are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

RadioShack is a registered trademark of RadioShack Corporation.

GN Netcom is a registered trademark of GN Netcom, Inc.

Motorola is a registered trademark of Motorola, Inc.

Belkin is a registered trademark of Belkin Components.

Coby is a registered trademark of Coby Electronics Corporation.

Plantronics is a registered trademark of Plantronics, Inc.

Intel, *Pentium*, and *Celeron* are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

AMD, *Athlon*, *Duron*, and *AMD-K6* are trademarks of Advanced Micro Devices, Inc.

All other trademarks are property of their respective companies.

Copyright

Copyright 2002, Computer Access Technology Corporation (CATC). All rights reserved.

This document may be printed and reproduced without additional permission, but all copies should contain this copyright notice.

Merlin's Wand Conformance Statements

FCC Conformance Statement

This equipment has been tested and found to comply with the limits for both a Class A and Class B digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial or residential environment. This equipment generates, uses, and can radiate radio frequency energy, and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. The end user of this product should be aware that any changes or modifications made to this equipment without the approval of CATC could result in the product not meeting the Class A or Class B limits, in which case the FCC could void the user's authority to operate the equipment.

Important Notice

This equipment contains Transmitter Module FCC ID PNI8001001. To comply with FCC RF exposure requirements (sections 1.1307 and 1.310 of the Rules) only the antenna supplied by CATC must be used for this device. The antenna must be located at least 20 centimeters away from all persons.

EU Conformance Statement

This equipment complies with the R&TT Directive 1999/5/EC. It has been tested and found to comply with EN55022:1998 Class B (EN61000-3-2:1998, EN61000-3-3:1995), EN55024:1998 (EN61000-4-2:1995, EN61000-4-3:1995, EN61000-4-4:1995, EN61000-4-5:1995, EN61000-4-6:1995, EN61000-4-11:1994), and EN60950:1999. The transmitter module was tested and found to comply with ETS 300 328 (1997).

TABLE OF CONTENTS

Document Disclaimer	ii
Trademarks and Servicemarks	ii
Copyright	ii
FCC Conformance Statement	iii
Important Notice	iii
EU Conformance Statement	iii
Table of Contents	v
Chapter 1 Overview	1
Applications	1
Merlin's Wand User Interface	2
Key Features	3
Audio Connections - Important Information	4
Limitations	5
Specifications	6
Chapter 2 Getting Started	7
System Requirements	7
Setting Up Merlin's Wand	8
Installing the Software and Starting the Program	8
Install from CD-ROM	8
Install from installation download	9
Start the program	9
Displaying the On-Screen Help	9
Application Layout	10
Menus	12
Toolbars	14
Merlin's Wand Toolbar	14
Merlin Bluetooth Analyzer Toolbar	15
Tool Tips	15
Merlin's Wand Keyboard Shortcuts	15
License Keys	16
License Information	16
Chapter 3 Bluetooth Wizard	17
Starting Bluetooth Wizard	17
Connect to a Device: Headset	18
Connect to a Device: Headset Audio Gateway	22
Connect to a Device: Object Push	27
Device Emulation: Headset	30
Device Emulation: Headset Audio Gateway	33
Device Emulation: Object Push	35

	Restarting the Wizard	38
Chapter 4	Command Generator	39
	Layout of the Command Generator	39
	Using Command Generator	40
	Customizing the List of HCI Commands	41
	Tables of Available Commands	42
	HCI Commands	42
	Link Control Commands	42
	Link Policy Commands	44
	Host Controller & Baseband Commands	44
	Informational Commands	46
	Status Commands	46
	Testing Commands	47
	CATC-Specific Commands	47
	L2CAP Commands	48
	SDP Commands	48
	RFCOMM Commands	49
	TCS Commands	50
	OBEX Commands	50
Chapter 5	Script Manager	53
	Layout of the Script Manager Window	53
	Running Scripts	54
	Writing Scripts	55
	Sample Scripts	56
	sample.script	57
	Sample-2.script	59
	Script Controlling Merlin Interface	60
Chapter 6	Device Search and Device List Pop-Up Menu	63
	Device Search	63
	Device List Pop-Up Menu	64
	Create an ACL Connection	64
	Establish an Audio Connection	65
	Display Device Information	66
	Delete a Device	67
	Disconnect All	68
Chapter 7	Data Transfer Manager and Data Pipes	69
	Creating Data Pipes	69
	Using Data Pipes	70
	Transfer Data Using Command Generator	71
	Access Pipes Using Script Manager	72
	Receive Pipes	72

Closing Pipes	73
Saving Data Pipes	73
Saving Data Transmit Pipe Lists	74
Saving a Default List	74
Saving Data Receive Pipes	74
Deleting Pipes	74
Opening Pipe Lists	75
Renaming Pipes	75
Modifying Pipes	76
Chapter 8 Using Merlin to Record Merlin's Wand Traffic	77
Set Up a Remote Machine	77
Windows 98/Me Operating Systems	77
Windows NT®/2000 Operating Systems	78
Set Up Connection Options	80
Start Merlin	81
Connect to Merlin with Merlin's Wand	81
Set Merlin Recording Options	81
Set Merlin Display Options	83
Set Merlin Encryption Options	84
Start a Merlin Recording Session	85
Stop a Merlin Recording Session	85
Disconnect from Merlin Bluetooth Protocol Analyzer	85
Troubleshooting	86
"Server Busy" When Attempting to Launch Merlin	86
"Server Execution Failed" When Attempting to Launch Merlin	86
"The Object Exporter Specified Was Not Found" When Attempting to Launch Merlin	86
Chapter 9 Contact and Warranty Information	89
Contact Information	89
Warranty and License	89
Appendix A: Command Generator Command Descriptions	91
HCI Command Descriptions	91
HCI Link Control Commands	91
Accept_Connection_Request	91
Add_SCO_Connection	91
Authentication_Requested	92
Change_Connection_Link_Key	92
Change_Connection_Packet_Type	92
Create_Connection	93
Disconnect	93

Exit_Periodic_Inquiry_Mode	93
Inquiry	93
Inquiry_Cancel	94
Periodic_Inquiry_Mode	94
PIN_Code_Request_Negative_Reply	95
PIN_Code_Request_Reply	95
Read_Clock_Offset	95
PIN_Code_Request_Negative_Reply	95
PIN_Code_Request_Reply	95
Read_Clock_Offset	95
Read_Remote_Supported_Features	96
Read_Remote_Version_Information	96
Reject_Connection_Request	96
Remote_Name_Request	96
Set_Connection_Encryption	97
HCI Link Policy Commands	98
Exit_Park_Mode	98
Exit_Sniff_Mode	98
Hold_Mode	98
Park_Mode	98
QoS_Setup	99
Read_Link_Policy_Settings	99
Role_Discovery	99
Sniff_Mode	100
Switch_Role	100
Write_Link_Policy_Settings	100
HCI Host Controller & Baseband Commands	101
Change_Local_Name	101
Delete_Stored_Link_Key	101
Host_Buffer_Size	102
Read_Authentication_Enable	102
Read_Class_of_Device	102
Read_Connection_Accept_Timeout	103
Read_Current_IAC_LAP	103
Read_Encryption_Mode	103
Read_Link_Supervision_Timeout	104
Read_Local_Name	104
Read_Number_Of_Supported_IAC	104
Read_Page_Scan_Mode	104
Read_Page_Scan_Period_Mode	105
Read_Page_Timeout	105
Read_PIN_Type	105
Read_Scan_Enable	106

Read_SCO_Flow_Control_Enable	106
Read_Stored_Link_Key	106
Read_Voice_Setting	107
Reset	107
Set_Event_Filter	107
Set_Event_Mask	108
Write_Authentication_Enable	109
Write_Class_of_Device	109
Write_Connection_Accept_Timeout	110
Write_Current_IAC_LAP	110
Write_Encryption_Mode	111
Write_Link_Supervision_Timeout	111
Write_Page_Timeout	112
Write_PIN_Type	112
Write_Scan_Enable	113
Write_Stored_Link_Key	113
Write_Voice_Setting	113
HCI Informational Commands	114
Read_BD_ADDR	114
Read_Buffer_Size	114
Read_Country_Code	114
Read_Local_Supported_Features	115
Read_Local_Version_Information	115
HCI Testing Commands	115
Enable_Device_Under_Test_Mode	115
Read_Loopback_Mode	116
Write_Loopback_Mode	116
CATC-Specific HCI Commands	116
CATC_BER_Test	116
CATC_Change_Headset_Gain	117
CATC_Read_Headset_Gain	117
CATC_Read_Revision_Information	118
CATC_Self_Test	118
CATC_Write_Country_Code	118
Other HCI Events	118
L2CAP Command Descriptions	119
ConfigurationResponse	119
ConfigurationSetup	119
ConnectRequest	120
ConnectResponse	120
DeregisterPsm	120
DisconnectRequest	120
EchoRequest	121

InfoRequest	121
RegisterPsm	121
SendData	122
Other L2CAP Events	122
SDP Command Descriptions	122
AddProfileServiceRecord	122
AddServiceRecord	123
ProfileServiceSearch	123
RequestServiceAttribute	124
RequestServiceSearch	125
RequestServiceSearchAttribute	125
ResetDatabase	125
RFCOMM Command Descriptions	126
AcceptChannel	126
AcceptPortSettings	126
AdvanceCredit	126
CloseClientChannel	127
CreditFlowEnabled	127
DeregisterServerChannel	127
OpenClientChannel	128
RegisterServerChannel	128
RequestPortSettings	128
RequestPortStatus	129
SendData	129
SendTest	130
SetLineStatus	130
SetModemStatus	131
SendATCommand	131
Other RFCOMM Events	132
TCS Command Descriptions	132
RegisterIntercomProfile	132
Open_TCS_Channel	132
Start_TCS_Call	133
Disconnect_TCS_Call	133
Send_Info_Message	133
OBEX Command Descriptions	134
ClientConnect	134
ClientDisconnect	134
ClientGet	134
ClientPut	135
ClientSetPath	135
ServerDeinit	135
ServerInit	136

ServerSetPath	136
Appendix B: Command Generator Examples.....	137
Device Discovery and Remote Name Request.....	138
Procedure	138
Establish Baseband Connection	139
Procedure	139
Baseband Disconnection.....	140
Procedure	140
Create Audio Connection	141
Procedure	141
L2CAP Connection.....	142
Procedure	142
L2CAP Channel Disconnect.....	144
Procedure	144
SDP Profile Service Search	145
Procedure	145
SDP Reset Database and Add Profile Service Record	146
Procedure	146
RFCOMM Client Channel Establishment	147
Procedure	147
RFCOMM Client Channel Disconnection	148
Procedure	148
RFCOMM Register Server Channel.....	149
Procedure	149
Establish TCS Connection	150
Procedure	150
OBEX Server Init	152
Procedure	152
OBEX Client Connection and Client Get & Put.....	153
Procedure	153
Appendix C: Merlin's Wand Scripting Commands	155
Bluetooth Addresses	155
Basic Commands	155
Main()	155
Clock()	156
Connect()	156
Disconnect()	157
DoInquiry()	157
GetDeviceClass()	158
GetRemoteDeviceName()	158
MessageBox()	159
SetDeviceClass()	159

Sleep()	160
Pipe Commands	160
ClosePipe()	160
DeletePipe()	161
OpenPipe()	161
ReadPipe()	162
WritePipe()	163
HCI Commands	164
HCIAcceptConnectionRequest()	164
HCIAddSCOConnection()	164
HCIAuthenticationRequested()	165
HCICatcBerTest()	165
HCICatcChangeHeadsetGain()	166
HCICatcEnterTestMode()	168
HCICatcReadHeadsetGain()	168
HCICatcReadRevisionInformation()	169
HCICatcSelfTest()	170
HCICatcTestControl()	170
HCICatcWriteCountryCode()	171
HCIChangeConnectionLinkKey()	171
HCIChangeConnectionPacketType()	172
HCIChangeLocalName()	173
HCIDeleteStoredLinkKey()	173
HCIEnableDeviceUnderTestMode()	174
HCIExitParkMode()	175
HCIExitSniffMode()	175
HCIHoldMode()	176
HCIMasterLinkKey()	177
HCIParkMode()	177
HCIPINCodeRequestNegativeReply()	178
HCIPINCodeRequestReply()	179
HCIQoSSetup()	180
HCIReadAuthenticationEnable()	181
HCIReadBDADDR()	182
HCIReadBufferSize()	182
HCIReadClockOffset()	184
HCIReadConnectionAcceptTimeout()	184
HCIReadCountryCode()	185
HCIReadCurrentIACLAP()	186
HCIReadEncryptionMode()	187
HCIReadLinkPolicySettings()	187
HCIReadLinkSupervisionTimeout()	188
HCIReadLocalName()	189

HCIReadLocalSupportedFeatures()	189
HCIReadLocalVersionInformation()	190
HCIReadLoopbackMode()	191
HCIReadNumberOfSupportedIAC()	192
HCIReadPageScanMode()	192
HCIReadPageScanPeriodMode()	193
HCIReadPageTimeout()	194
HCIReadPINType()	195
HCIReadRemoteSupportedFeatures()	195
HCIReadRemoteVersionInformation()	196
HCIReadScanEnable()	197
HCIReadSCOWFlowControlEnable()	198
HCIReadStoredLinkKey()	198
HCIReadVoiceSetting()	200
HCIRejectConnectionRequest()	200
HCIRemoveSCOConnection()	201
HCIReset()	201
HCIRoleDiscovery()	202
HCISetConnectionEncryption()	202
HCISetEventFilter()	203
HCISniffMode()	204
HCISwitchRole()	205
HCIWriteAuthenticationEnable()	206
HCIWriteConnectionAcceptTimeout()	207
HCIWriteCurrentIACLAP()	207
HCIWriteEncryptionMode()	208
HCIWriteLinkPolicySettings()	209
HCIWriteLinkSupervisionTimeout()	209
HCIWriteLoopbackMode()	210
HCIWritePageTimeout()	211
HCIWritePINType()	211
HCIWriteScanEnable()	212
HCIWriteStoredLinkKey()	212
HCIWriteVoiceSettings()	213
OBEX Commands	214
OBEXClientConnect()	214
OBEXClientDeinit()	214
OBEXClientDisconnect()	215
OBEXClientGet()	215
OBEXClientInit()	216
OBEXClientPut()	217
OBEXClientSetPath()	218
OBEXServerDeinit()	219

OBEXServerSetPath(Path)	220
RFCOMM Commands	221
RFCloseClientChannel()	221
RFOpenClientChannel()	221
RFRegisterServerChannel()	222
RFSendData()	223
RFSendDataFromPipe()	223
RFReceiveData()	224
RFWaitForConnection()	225
RFAcceptChannel()	226
RFAcceptPortSettings()	226
RFCreditFlowEnabled()	227
RFRequestPortSettings()	228
RFRequestPortStatus()	229
RFSetLineStatus()	230
RFSetModemStatus()	231
RFSendTest()	232
RFAdvanceCredit()	233
TCS Commands	233
TCSRegisterProfile()	233
TCSOpenChannel()	234
TCSStartCall()	234
TCSDisconnectCall()	235
TCSSendInfoMessage()	236
L2CAP Commands	238
L2CAPConfigurationSetup()	238
L2CAPConfigurationResponse()	239
L2CAPConnectRequest()	240
L2CAPConnectResponse()	241
L2CAPDeregisterAllPsm()	241
L2CAPDisconnectRequest()	242
L2CAPEchoRequest()	242
L2CAPInfoRequest()	243
L2CAPRegisterPsm()	244
L2CAPSendData()	244
L2CAPSendDataFromPipe()	245
L2CAPWaitForConnection()	245
SDP Commands	246
SDPAddProfileServiceRecord()	246
SDPQueryProfile()	247
SDPResetDatabase()	247
SDPAddServiceRecord()	248
Merlin Commands	249

MerlinResetAllEncryptionOptions()	249
MerlinSetDisplayOptions()	249
MerlinSetEncryptionLinkKey()	250
MerlinSetEncryptionPIN()	250
MerlinSetRecordingOptions()	251
MerlinStart()	251
MerlinStartRecording()	252
MerlinStop()	252
MerlinStopRecording()	253
Appendix D: CATC Scripting Language	255
Values	255
Literals	255
Integers	255
Strings	256
Escape Sequences	256
Lists	257
Raw Bytes	257
Null	257
Variables	257
Global Variables	258
Local Variables	258
Constants	259
Expressions	259
select expression	260
Operators	261
Operator Precedence and Associativity	262
Comments	268
Keywords	268
Statements	269
Expression Statements	269
if Statements	270
if-else Statements	270
while Statements	270
for Statements	271
return Statements	272
Compound Statements	273
Preprocessing	274
Functions	275
Primitives	276
Call()	277
Format()	277
Format Conversion Characters	278
GetNBits()	279

NextNBits().....	280
Resolve().....	280
Trace().....	281

1. Overview

The CATC Merlin's Wand™ Bluetooth™ Test Generator is the newest member in CATC's industry-leading line of high performance, serial communication protocol analysis tools and test equipment.

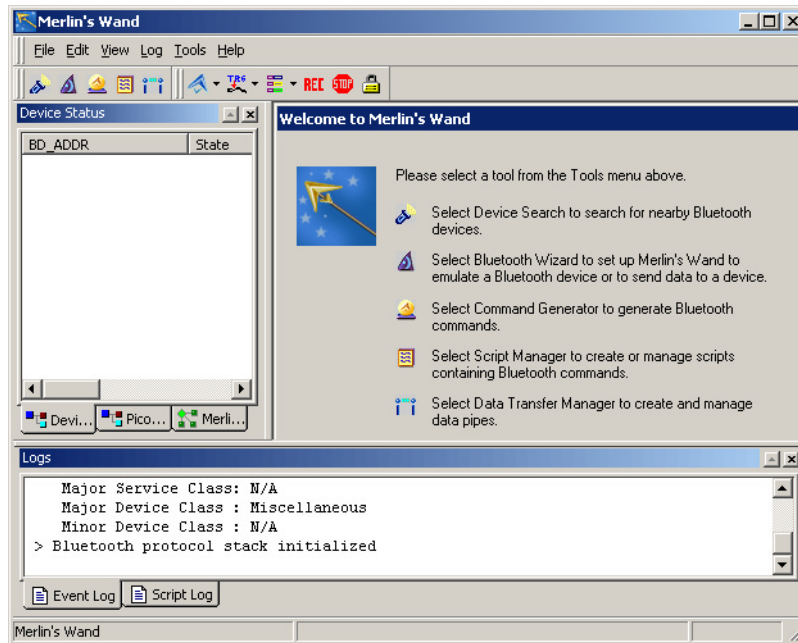
Preceded by CATC's Merlin™, a Bluetooth Protocol Analyzer, Merlin's Wand has been designed as an intelligent Bluetooth wireless technology device that can be used as a verification/validation tester or as an engineering debug and analysis tool. Through its software interface, designers and test technicians will be able to quickly and easily issue protocol commands and test sequences to analyze or validate designs to ensure compliance to the Bluetooth specification. Merlin's Wand can be used in conjunction with the Merlin protocol analyzer, allowing for real-time captures of test sequence results, as is required by the Bluetooth SIG to provide evidence of product compliance to the specification.

1.1 Applications

Merlin's Wand is a combination of hardware and Microsoft® Windows®-based application software. The hardware/software combination is capable of acting as a standard Bluetooth master or slave device within a piconet. By allowing this capability, Merlin's Wand can be used to establish or participate in a piconet and to send or receive data within the piconet. Through the Merlin's Wand Bluetooth Wizard, users can quickly and easily manage Bluetooth wireless traffic generation. Additionally, via its Command Generator, Merlin's Wand can issue individual Bluetooth commands to a device under test, allowing a designer to focus his or her effort on a specific function or group of functions related to the device. Furthermore, users can quickly create test sequences with Script Manager, thus eliminating the difficulties normally associated with the creation of complex test sequences.

1.2 Merlin's Wand User Interface

The Merlin's Wand user interface consists of the Main window, the Logs window at the bottom of the screen, and the Device Status window on the left side of the screen.

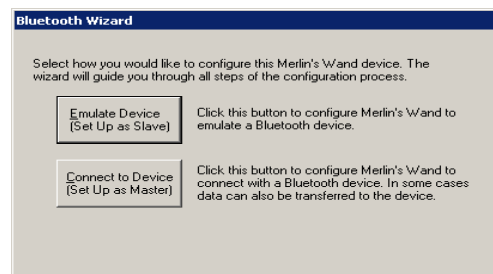


The application's primary tools are run within the Main window: *Bluetooth Wizard*, *Command Generator*, and *Script Manager*. If Command Generator and Script Manager are not enabled on your Merlin's Wand system, you will need to obtain License Keys from CATC in order to use them.

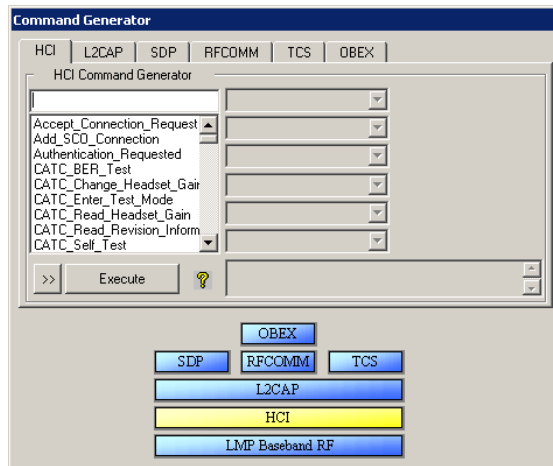
Each tool offers a different means of generating traffic.

Note: Only one tool can be run at a time.

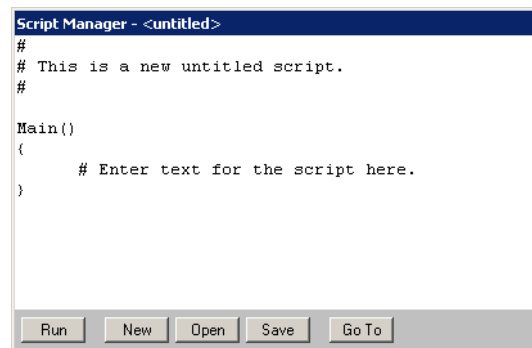
Bluetooth Wizard is a point-and-click tool for creating connections and transferring data between Merlin's Wand and other Bluetooth devices. This tool requires little Bluetooth wireless technology experience and allows you to generate Bluetooth traffic without having to execute specific Bluetooth commands. Bluetooth Wizard is described in Chapter 3, *Bluetooth Wizard*, on page 17.



Command Generator is a tool that presents a menu of protocol commands that can be selected and executed in virtually any sequence. Command Generator thus offers maximum control over the traffic generation process, but also requires familiarity with the Bluetooth commands. Command Generator is described in Chapter 4, *Command Generator*, on page 39.



Script Manager is a tool that provides an editor for writing and/or executing scripts that will generate Bluetooth wireless traffic. With Script Manager, new scripts can be written and saved, or existing scripts may be opened, edited, and run. Script Manager is described in Chapter 5, *Script Manager*, on page 53.



1.3 Key Features

- Plug-and-play USB connection between test system and test module
- External antenna can be removed to create wired piconet
- Audio connector for connecting audio devices, such as headsets
- Can operate as either a master or slave device in a piconet
- Graphical interface allows for easy selection of command parameters
- Wizard provided to reduce learning curve and memorization of command sequences
- Test modes provide for these Bluetooth wireless technology protocols: HCI, L2CAP, SDP, RFCOMM, TCS, and OBEX
- Scripting capability for establishment of predefined test sequences
- System information report provides details regarding device under test

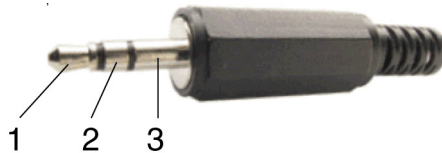
- Can be used with Merlin protocol analyzer
- Power-on self-diagnostics
- No external power required -- obtains power from USB connection
- One year warranty and online customer support

Please refer to the *Bluetooth Specification, version 1.1* for details on the Bluetooth wireless technology protocol. The Bluetooth specification is available from the Bluetooth SIG at its web site <http://www.Bluetooth.com>.

1.4 Audio Connections - Important Information

Merlin's Wand has a 2.5 mm audio stereo jack for plugging in headsets.

Headsets need to have a 2.5 mm plug with the following pinout:



- 1) **Microphone** (signal from headset; bias power of 2.5 V and maximum 1 mA provided by Merlin's Wand on the same pin)
- 2) **Speaker** (signal to headset; speaker impedance needs to be >16 Ohm)
- 3) **Ground**

The following headsets have been successfully tested with Merlin's Wand:

- **RadioShack®** 43-1957 Super Lightweight Hands-Free Headset
- **GN Netcom®** GNX Mobile M200
- **Motorola®** Retractable Hands-Free Headset Model # 98196G
- **Belkin®** Universal 2.5 mm Personal Hands-Free Kit F8V920-PL
- **Coby®** CV-M20 Earphone with Built-In Microphone
- **Plantronics®** CHS122N Hands-Free Headset
- **Plantronics** M110 Headset for Cordless and Mobile Phones

1.5 Limitations

- The only Inquiry Access Code (IAC) supported at inquiry and inquiry scan is the General Inquiry Access Code (GIAC)
- The Scan_Enable parameter value of 0x01 (inquiry scan enabled and page disabled) is not supported
- Page scan intervals and inquiry scan intervals other than 1.28s are not supported
- Page scan windows and inquiry scan windows other than 11.25ms are not supported
- Optional page scan modes are not supported
- More than one SCO connection at a time is not supported

1.6 Specifications

The following specifications describe a Merlin's Wand System.

Package

Dimensions:	3.4 x 2.6 x 1 inches (8.6 x 6.6 x 2.54 centimeters)
Connectors:	Host connection (USB, type 'B') Audio connection (2.5 millimeter audio stereo jack)
Weight:	3.0 oz. (84 g)

Environmental Conditions

Operating Range:	0 to 55 °C (32 to 131 °F)
Storage Range:	-20 to 80 °C (-4 to 176 °F)
Humidity:	10 to 90%, non-condensing

Host Compatibility

Works with any PC equipped with a functioning USB port and a Microsoft Windows 98 SE, Windows Me, Windows 2000, or Windows XP operating system.

Hardware Interfaces

Standard USB Interface -- connects to the host computer
2.4 GHz (ISM band) External Antenna.
2.5 mm audio stereo jack

Product Warranty

CATC offers a one-year limited warranty on its products.

2. Getting Started

This chapter describes how to install Merlin's Wand and its software. Both install easily in just a few minutes. The Merlin's Wand software can be installed on most Windows-based personal computer systems.

2.1 System Requirements

The following is the recommended configuration for the computer that runs the Merlin's Wand application and is connected to the Merlin's Wand hardware unit.

- **Operating system:** Microsoft® Windows® 98 SE, Windows 2000, Windows Me, or Windows XP operating system.

Occasionally, after unplugging the Merlin's Wand hardware on a Windows 2000 system, subsequent attempts to plug in the device cause the computer to recognize the unit as a "USB device" rather than as "Merlin's Wand." If this occurs, you will need to restart the computer so that it will recognize the device properly. To avoid the problem, upgrade to the latest Windows 2000 service pack available from Microsoft.
- **Required setup:** Microsoft Internet Explorer 5 or later must be installed.
- **Processor:** For optimum performance, use processors of the Intel® Pentium® III or Pentium 4 family, the AMD® Athlon® or Duron® family, or other compatible processors with clock speed of 500MHz or higher. Must have, as a minimum, a processor from the Intel Pentium II or Celeron® family, AMD-K6® family, or equivalent with clock speed of 300MHz.
- **Memory:** Minimum of 64 MB of RAM is recommended; more memory generally improves responsiveness.
- **Hard disk:** Minimum 20 GB hard disk is required for installation. Additional disk space is needed for storing user-generated script files and log files that are saved by the user.
- **Display:** Resolution of 1024 x 768 with at least 16-bit color is recommended (resolution of 800 x 600 with 16-bit color is a minimum).
- **Connectivity:** A USB interface is required to connect the computer to the Merlin's Wand hardware unit.

2.2 Setting Up Merlin's Wand

The Merlin's Wand hardware can be set up using the installation CD-ROM or from installation files downloaded from the CATC website.

- Step 1** Attach the external antenna to the Merlin's Wand hardware unit by screwing it onto the connector labelled ANT.
- Step 2** Plug one end of the USB cable into the USB port on the Merlin's Wand hardware, and plug the other end of the USB cable into a USB port on the host computer.

Windows should automatically detect the Merlin's Wand hardware and open the Windows Hardware Wizard to install Merlin's Wand. If the installation doesn't finish automatically, proceed to Step 3.

- Step 3** Follow the Hardware Wizard's on-screen instructions to complete the installation. If the wizard prompts you for driver information, insert the Merlin's Wand installation CD-ROM and direct the wizard to the directory `<drive>:\software\MerlinWand122`. Substitute the drive letter of the CD-ROM drive for `<drive>`. For example, if the CD-ROM drive is drive D, navigate to `D:\software\MerlinWand122`.

Note: If you are using installation files downloaded from the CATC website, you will need to direct the wizard to the Disk 1 directory of the installation files so that it can locate the driver.

2.3 Installing the Software and Starting the Program

The Merlin's Wand software can be installed from the installation CD-ROM or from installation files downloaded from the CATC website.

Install from CD-ROM

- Step 1** Insert the Merlin's Wand installation CD-ROM into the CD-ROM drive of the computer that will be connected to the Merlin's Wand hardware.

The autorun program should start automatically. If it doesn't start, use Windows Explorer or My Computer to navigate to the CD-ROM drive directory and double-click the file **autorun.exe**, and proceed to Step 2. If it still doesn't start, navigate to the `\software\MerlinWand122` directory on the CD-ROM, double-click the file `Setup.exe`, and proceed to Step 3.

- Step 2** Choose **Install Software** to start the setup program.
- Step 3** Follow the on-screen instructions to complete the

installation.

Install from installation download

- Step 1 Select **Start > Run...** from the Windows taskbar and click the **Browse** button, then navigate to the Disk 1 directory of the Merlin's Wand installation download. Select the file **Setup.exe** and click **Open**.
- Step 2 Follow the on-screen instructions to complete the installation.

Start the program

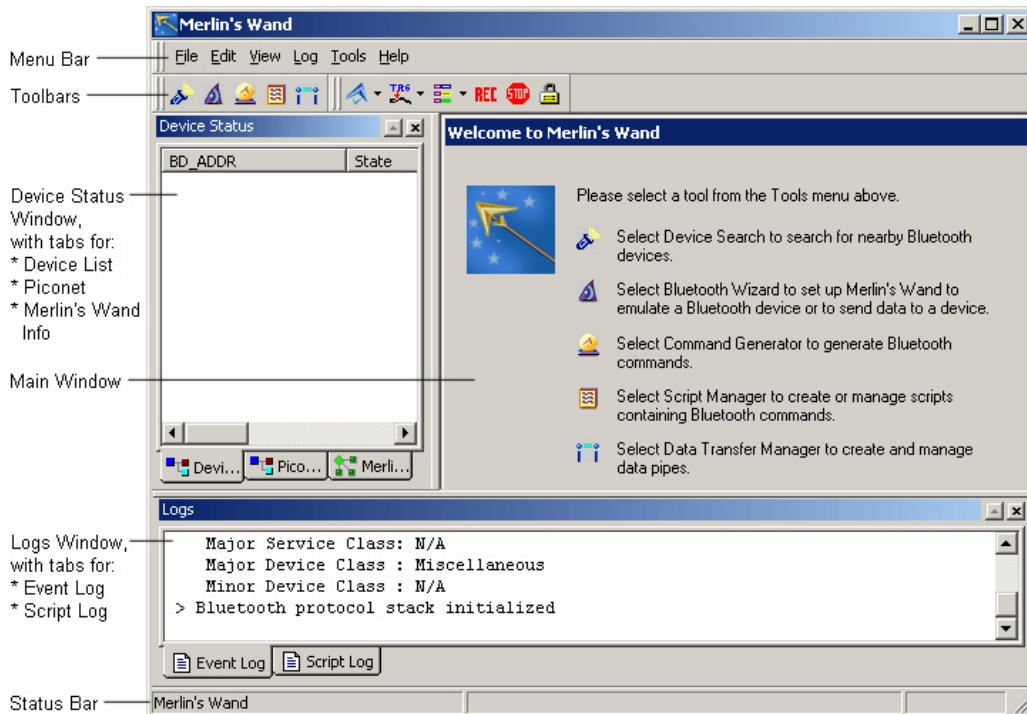
Once the software has been installed, be sure that the Merlin's Wand hardware is connected to the PC via the USB cable before starting the Merlin's Wand application. Otherwise, the application will provide a warning message telling you that the Wand could not be found.

To start the application, select **Start > Programs > CATC > CATC Merlin's Wand**. Note that this is the default location for the Merlin's Wand application. If it was installed in a different folder, select that folder from the Programs menu.

2.4 Displaying the On-Screen Help

Access the on-screen Help included with the Merlin's Wand application by selecting **Help > Help...** from the menu bar.

2.5 Application Layout



The Merlin's Wand window is made up of the following:

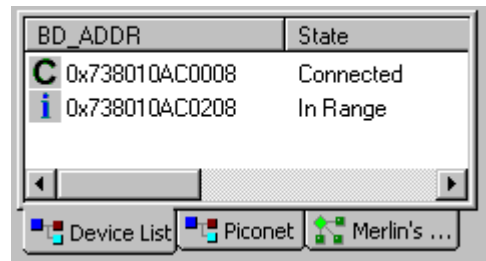
- The Main window, where the primary tools are run: **Bluetooth Wizard, Command Generator, and Script Manager.**
 - **Bluetooth Wizard** -- A simple, easy-to-use tool that guides you through the process of establishing connections and generating traffic between Merlin's Wand and other Bluetooth wireless technology devices.
 - **Command Generator** -- A tool that allows Bluetooth commands to be issued in any chosen sequence. If Command Generator isn't enabled on your Merlin's Wand system, a License Key must be obtained from CATC before it can be used.
 - **Script Manager** -- A notepad-like tool for writing and launching scripts that cause Merlin's Wand to generate traffic. This tool is an optional feature. If Script Manager isn't enabled on your Merlin's Wand system, a License Key must be obtained from CATC before it can be used.

Note: When switching between Bluetooth Wizard, Command Generator and Script Manager, all connections that have been established between Merlin's Wand and another Bluetooth device should be closed. However, expert users may choose to leave the connections open. If a connection is left open and you attempt to switch tools, Merlin's Wand will prompt you to close the connections. Choosing **Disconnect All** will close the connections. Choosing **Cancel** will leave the connections open, but some commands might not work properly in the other tool. When switching to Bluetooth Wizard, any open connections *must* be closed.

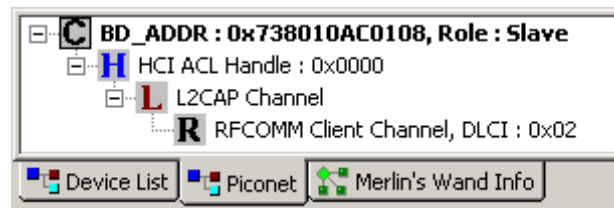
- The **Device Status** window is on the left side of the interface. It contains three tabs: **Device List**, **Piconet**, and **Merlin's Wand Info**.



- **Device List** - Displays a list of devices that Merlin's Wand has discovered. It also contains information about the devices found, such as the Bluetooth address, the state, the role, the class, and the device's local name. This window is open by default. These symbols in the list indicate a device's state: **C** = Connected; **i** = In Range. Right-clicking on a listed device opens the **Device List Pop-Up Menu**. The menu presents the following options: *Connect*, *Add Audio Connection*, *Get Device Information*, *Delete*, and *Disconnect All*. For details on using the Pop-Up Menu, see Chapter 6, *Device Search and Device List Pop-Up Menu*, on page 63.



- **Piconet** - Displays a hierarchical list of all connections between Merlin's Wand and other devices. At the top of the list is the address of the connected device; below it are the various channels established between Merlin's Wand and the device. Symbols: **C** = Connection; **H** = HCI ACL; **S** = HCI SCO; **L** = L2CAP; **R** = RFCOMM; **O** = OBEX.



- **Merlin's Wand Info** - Displays details about Merlin's Wand.

Parameter	Value	Description
Local BD_ADDR	0x738010AC0108	Local Bluetooth device address
Local Name	Merlin's Wand	User-friendly name of the local device
Class of Device	0x000000	The value of the local Class_of_Device
Num Of Supp IAC	1	The number of supported Inquiry Access Codes
Current IAC LAP	0x9E8B33	Currently used IAC for Inquiry Scan
Accessibility Mode	Non-Accessible	Current accessibility mode of the local device

- At the bottom of the interface is the **Logs window**, which contains tabs for the Event Log and the Script Log:
 - **Event Log** - Maintains a log of the commands issued by Merlin's Wand and the events that ensue, such as a reply by another device.

```

Logs
BD_ADDR      : 0x738010AC0008
HCI_Evt> Change_Local_Name_Complete
Local Name   : "Merlin's Wand"
HCI_Evt> Read_Class_of_Device_Complete
COD         : 0x000000
  
```

- **Script Log** - Maintains a record of the commands issued by Script Manager and the events resulting from these commands. If line numbers are referenced in the Script Log, double-clicking on the line number will move the cursor to that line in the Script Manager window.

```

Logs
Close SCO conenction: Not connected
Disconnect returned: Failure

Script execution complete for C:\Program Files\CATC\Merlin's
Script returned: NULL
  
```

2.6 Menus

The menu bar at the top of the application window contains the following menus of pull-down commands:

Table 1: Menu Bar Commands

File Menu

<u>Command</u>	<u>Function</u>
New Script	Creates a new script file
Open Script...	Opens a script file
Close Script	Closes a script file
Save Script	Saves a script file

Table 1: Menu Bar Commands (Continued)

Save Script As...	Saves a script file with a specified name
Print Setup...	Sets up the current or a new printer
Print Script...	Prints a script file
Exit	Exits the Merlin's Wand program

Edit Menu

<u>Command</u>	<u>Function</u>
Undo	Undoes last change
Cut	Cuts text
Copy	Copies text
Paste	Pastes copied or cut text
Select All	Selects all text
Find...	Finds specified string
Find Next	Repeats last find action
Replace...	Searches for a string and replaces it with a new string

View Menu

<u>Command</u>	<u>Function</u>
Merlin's Wand Toolbar	Shows or hides the Merlin's Wand toolbar
Merlin Analyzer Toolbar	Shows or hides the Merlin Analyzer toolbar
Device Status	Shows or hides the Device Status window
Logs	Shows or hides the Logs window
Status Bar	Shows or hides the status bar

Log Menu

<u>Command</u>	<u>Function</u>
Copy Selected Log Text	Copies selected log text to the clipboard
Select All Log Text	Selects all log text in the open log window
Clear Log Window	Clears all text from the open log window
Save Log As...	Saves log file to new name and/or directory
Print Log...	Prints all text from the open log window

Tools Menu

<u>Command</u>	<u>Function</u>
Device Search	Opens the Device Search dialog box
Bluetooth Wizard	Opens Bluetooth Wizard
Command Generator	Opens Command Generator
Script Manager	Opens Script Manager

Table 1: Menu Bar Commands (Continued)

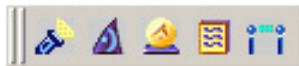
Data Transfer Manager	Opens Data Transfer Manager
-----------------------	-----------------------------





Help Menu

<u>Command</u>	<u>Function</u>
Help...	Displays online Help
Update License...	Opens the Update License dialog, which is used to install License Keys. License Keys must be obtained from CATC.
Display License Information...	Displays maintenance expiration and features data for Merlin's Wand.
About Merlin's Wand	Displays version information about Merlin's Wand.

2.7 Toolbars

There are two toolbars in the Merlin's Wand user interface: Merlin's Wand toolbar and Merlin Analyzer toolbar. The Toolbar buttons provide access to frequently-used program functions. Tool tips describe icon functionality as the mouse pointer is moved over an item.

Merlin's Wand Toolbar

	Device Search Opens Device Search dialog
	Bluetooth Wizard Opens Bluetooth Wizard
	Command Generator Opens Command Generator
	Script Manager Opens Script Manager
	Data Transfer Manager Opens Data Transfer Manager dialog

Merlin Bluetooth Analyzer Toolbar



Connect/Disconnect Merlin Bluetooth Analyzer

Connects to or disconnects from Merlin Bluetooth Analyzer



Set Merlin Recording Options

Displays the Open dialog to choose the Recording Options file for Merlin Bluetooth Analyzer



Set Merlin Display Options

Displays the Open dialog to choose the Display Options file for Merlin Bluetooth Analyzer



Start Recording

Starts a Merlin Bluetooth Analyzer recording session



Stop Recording

Stops a Merlin Bluetooth Analyzer recording session



Set Merlin Encryption Options

Opens the Encryption Setup window

2.8 Tool Tips

For most of the buttons and menus, tool tips provide useful information.

To display a tool tip, position the mouse pointer over an item. If a tooltip exists for the item, it will pop up in a moment.



2.9 Merlin's Wand Keyboard Shortcuts

Several frequently-used operations are bound to keyboard shortcuts.

Table 2: Keyboard Shortcuts

Key Combination	Operation	Key Combination	Operation
Ctrl + A	Select all	Ctrl + V	Paste
Ctrl + C	Copy	Ctrl + W	Close script
Ctrl + F	Find	Ctrl + X	Cut
Ctrl + G	Go to	Ctrl + Z	Undo
Ctrl + H	Replace	Home	Jump to first character of line
Ctrl + I	Indent	End	Jump to last character of line
Ctrl + N	New script	Ctrl + Home	Jump to first character of file

Table 2: Keyboard Shortcuts (Continued)

Key Combination	Operation	Key Combination	Operation
Ctrl + O	Open script	Ctrl + End	Jump to last character of file
Ctrl + P	Print script...	Ctrl + Backspace	Delete previous word
Ctrl + R	Run script	F3	Find next
Ctrl + S	Save script	Alt + F4	Shut down Merlin's Wand application

2.10 License Keys

License Keys are necessary to enable software maintenance, and may be necessary to use the Command Generator and Script Manager tools. If these tools are not enabled, a message will appear if you attempt to access them, stating that a License Key is necessary in order to use them. License Keys must be obtained from CATC.

Follow these steps to install License Keys:

Step 1 Select **Help > Update License...** from the menu bar.

The Update License dialog will come up.

Step 2 Enter the path and filename for the License Key or use the Browse button to navigate to the directory that contains the License Key. Select the .lic file, and then click **Update Device**.

2.11 License Information


Licensing information for Merlin's Wand may be viewed by selecting **Help > Display License Information...** from the menu bar. The License Information window will open, displaying the maintenance expiration and features data for Merlin's Wand.

3. Bluetooth Wizard

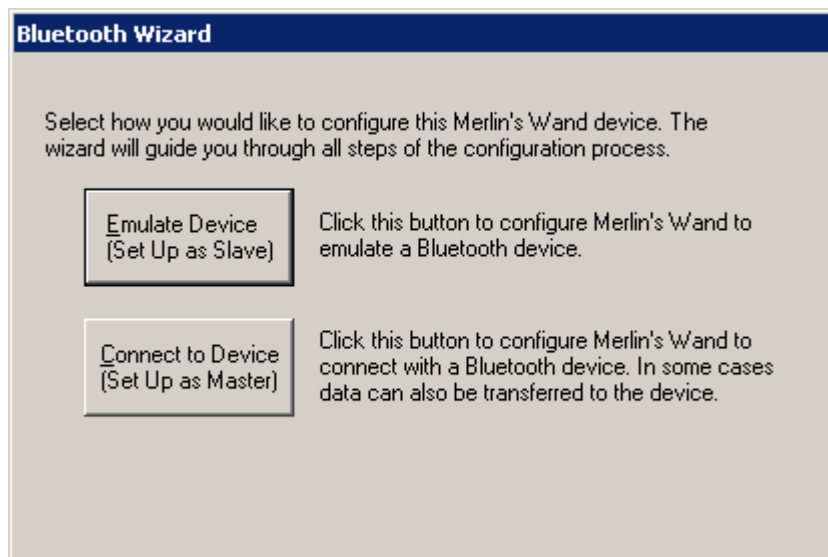
Once Merlin's Wand is installed and running, it is ready to generate traffic.

The easiest way to generate traffic is to use Bluetooth Wizard, a point-and-click tool for creating connections and transferring data between Merlin's Wand and other Bluetooth wireless technology devices. This tool requires little Bluetooth experience and allows the user to generate Bluetooth traffic without having to execute specific Bluetooth commands. Bluetooth Wizard manages the entire traffic generation process. Merely follow the on-screen instructions and Merlin's Wand will execute the Bluetooth commands needed to make the connection.

3.1 Starting Bluetooth Wizard

Step 1 Click the Bluetooth Wizard icon  or select **Tools > Bluetooth Wizard** from the menu.

Bluetooth Wizard will open in the Main Window, displaying the following screen:




When you start Bluetooth Wizard, Merlin's Wand gives you the choice of connecting to a device or emulating a device:

Emulate a Device -- Choose this option to configure Merlin's Wand to emulate a Bluetooth wireless device. Merlin's Wand can emulate two types of devices: Bluetooth headsets that comply with the Headset Audio Gateway profile and devices that comply with the Object Push profile (i.e., devices that are capable of transferring files).

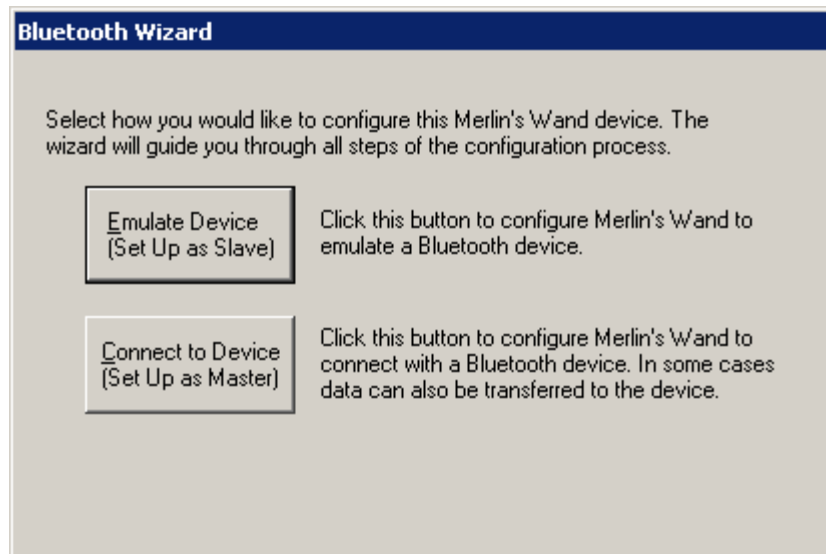
Connect to Device -- Choose this option to configure Merlin's Wand to seek out other Bluetooth devices, connect to one of them, and possibly exchange data with that device. Merlin's Wand can connect to and exchange data with two types of devices: Bluetooth wireless headsets that support the Headset Audio Gateway profile and devices that comply with the Object Push profile.

3.2 Connect to a Device: Headset

The following steps show how to configure Merlin's Wand to connect to and transfer audio with a Bluetooth headset using the Headset profile. An audio headset is connected to Merlin's Wand in order to transmit audio between Merlin's Wand and a Bluetooth headset. A second Merlin's Wand with an audio headset attached can also be used.

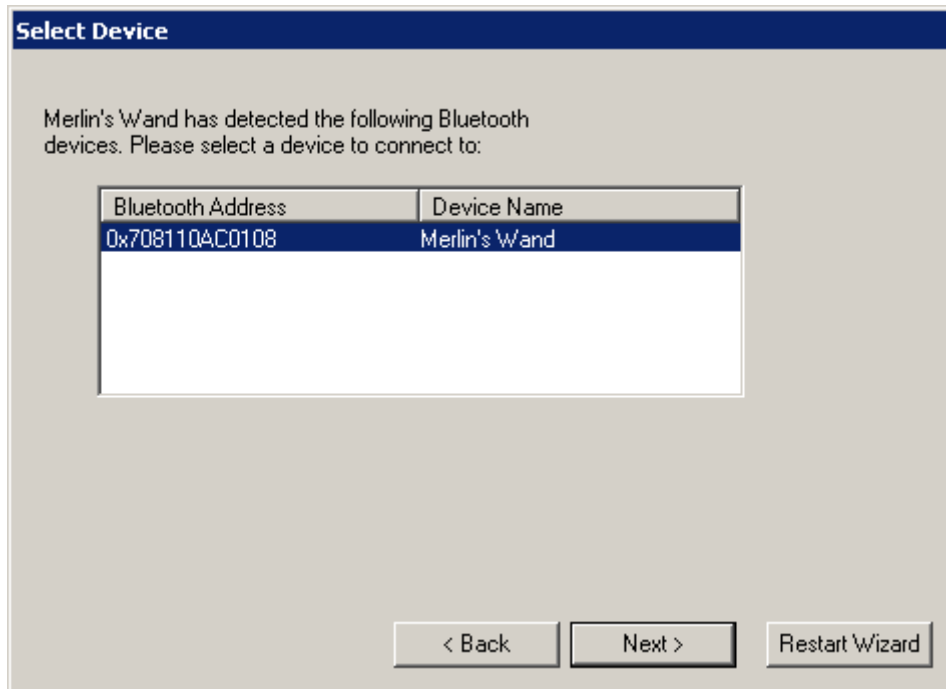
- Step 1** Turn on the Bluetooth headset that Merlin's Wand will be connecting to over the air (i.e., the target device).
- Step 2** Attach a headset to the audio connector on Merlin's Wand.
- Step 3** Click the Bluetooth Wizard icon  on the toolbar or select **Tools > Bluetooth Wizard** from the Menu Bar.

The Bluetooth Wizard opening screen opens in the Main Window:



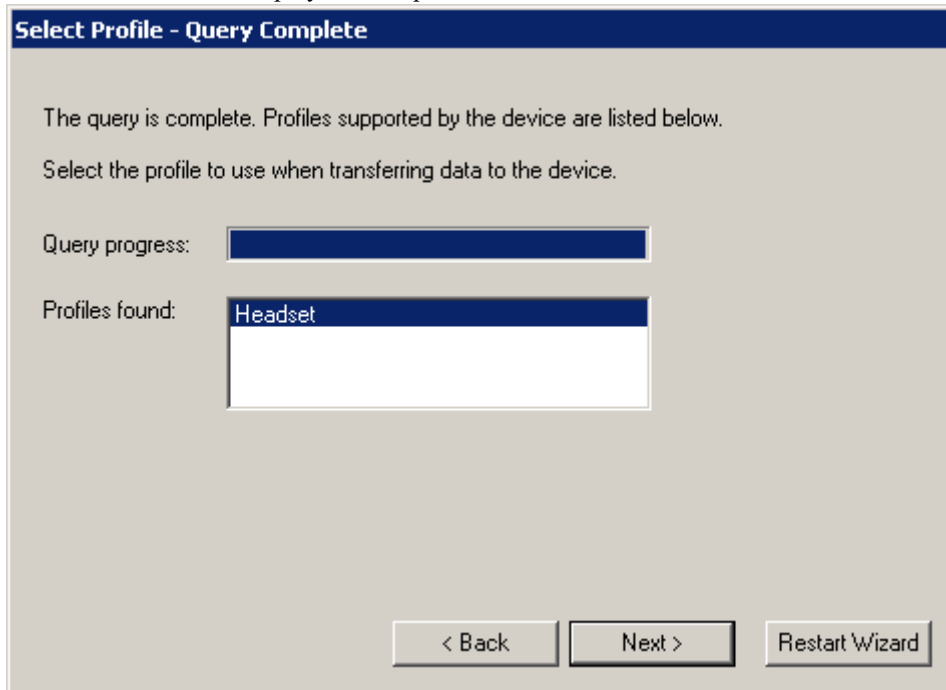
- Step 4** Click the button marked **Connect to Device**.

Clicking this button causes Merlin's Wand to perform a General Inquiry and collect information on local devices. The Select Device screen opens, displaying the devices that are found.



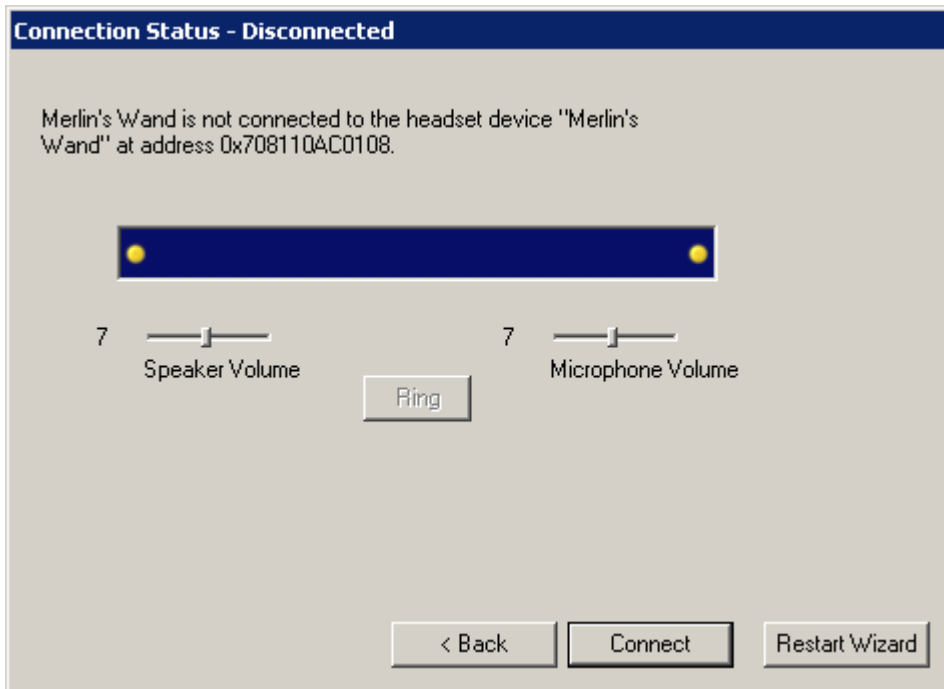
- Step 5** From the list, select the device address to which you want Merlin's Wand to connect, and then click **Next**.

Merlin's Wand will query the selected device to determine its profile. When the query is complete, the Select Profile screen will open and display a list of profiles found.



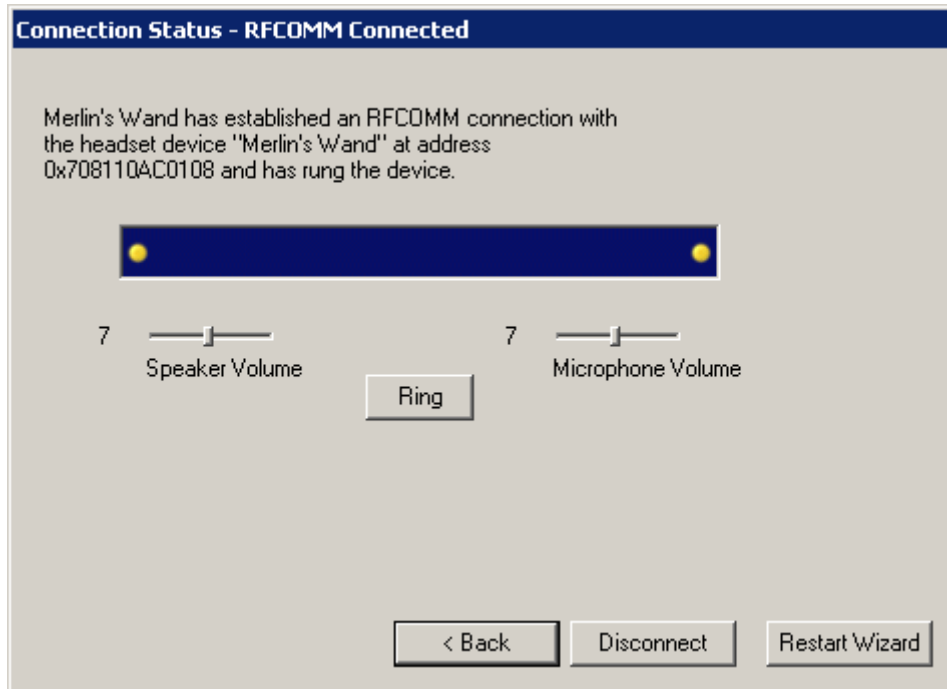
Step 6 Select **Headset Gateway**, then click **Next**.

The Connection Status screen will open, indicating that Merlin's Wand is *not* connected to the device.:



Step 7 Select Connect.

The Connection Status screen will change to show that Merlin's Wand has established an RFCOMM connection with the device. Merlin's Wand will automatically ring the target device and wait for an answer. Pressing the **Ring** button will cause Merlin's Wand to ring the device again.

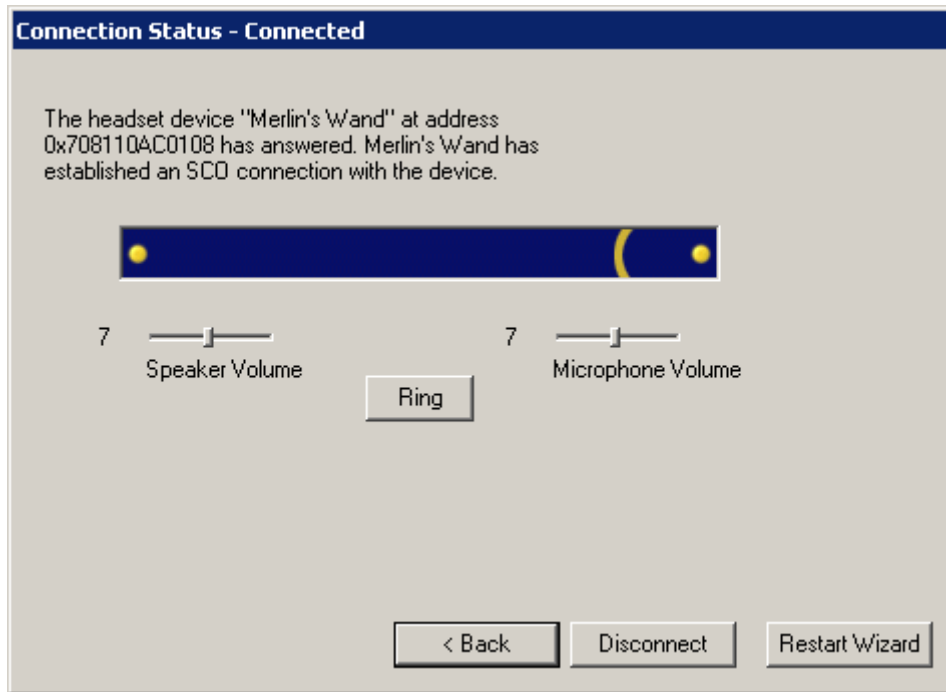


Note The Speaker and Microphone Volume levels can be adjusted by moving the sliders up or down. The level is indicated by a number, from 0 to 15, to the left of each slider.

Note If you cannot establish a connection, you can re-attempt the connection by either pressing **Back** and re-running the previous two steps, or by pressing the **Connect** button again.

Step 8 When the target device answers, Merlin's Wand will

establish an SCO connection with it.




- Step 9** (Optional) Click the **Disconnect** button on the Connection Status screen to close the connection.

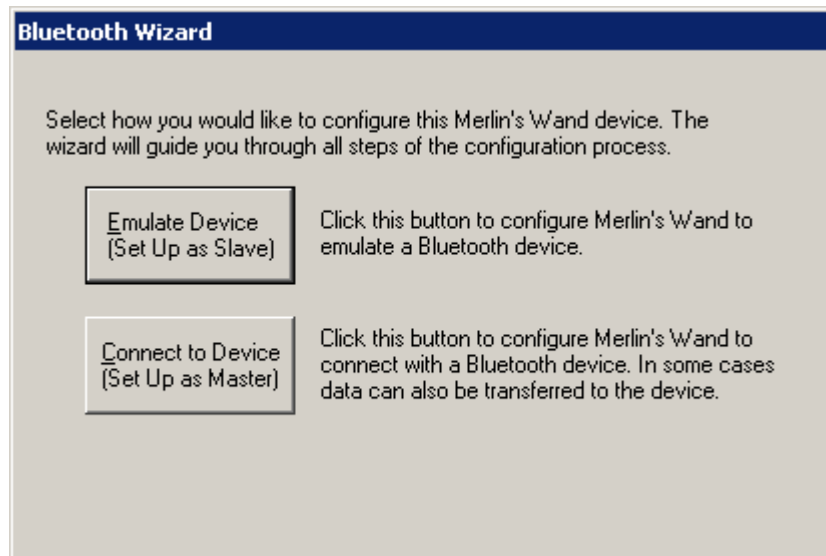
The connection between Merlin's Wand and the device will terminate, and the **Connect** button will again be available. Selecting **Connect** will reestablish the connection.

3.3 Connect to a Device: Headset Audio Gateway

The following steps show how to configure Merlin's Wand to connect to and transfer audio with a Bluetooth headset using the Headset Audio Gateway profile. An audio headset is connected to Merlin's Wand in order to transmit audio between Merlin's Wand and a Bluetooth headset. A second Merlin's Wand with an audio headset attached can also be used.

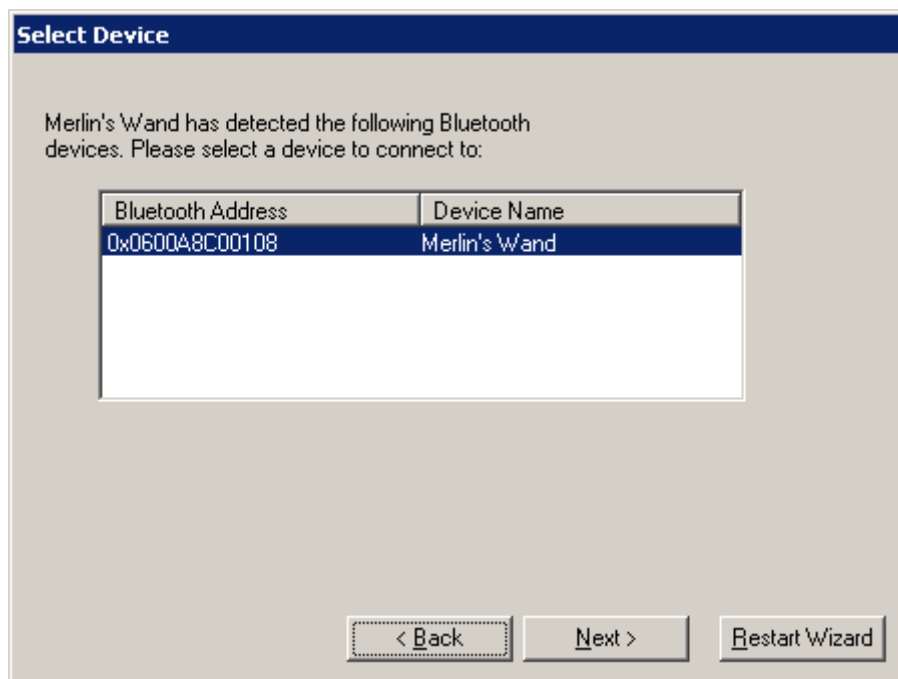
- Step 1** Turn on the Bluetooth headset that Merlin's Wand will be connecting to over the air (i.e., the target device).
- Step 2** Attach a headset to the audio connector on Merlin's Wand.
- Step 3** Click the Bluetooth Wizard icon  on the toolbar or select **Tools > Bluetooth Wizard** from the Menu Bar.

The Bluetooth Wizard opening screen opens in the Main Window:

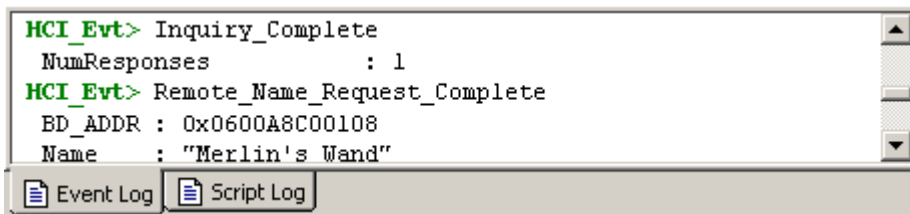


Step 4 Click the button marked **Connect to Device**.

Clicking this button causes Merlin's Wand to perform a General Inquiry and collect information on local devices. The Select Device screen opens, displaying the devices that are found.



Note the messages that appear in the **Event Log**. These messages provide details about the actions and responses taken in each step.

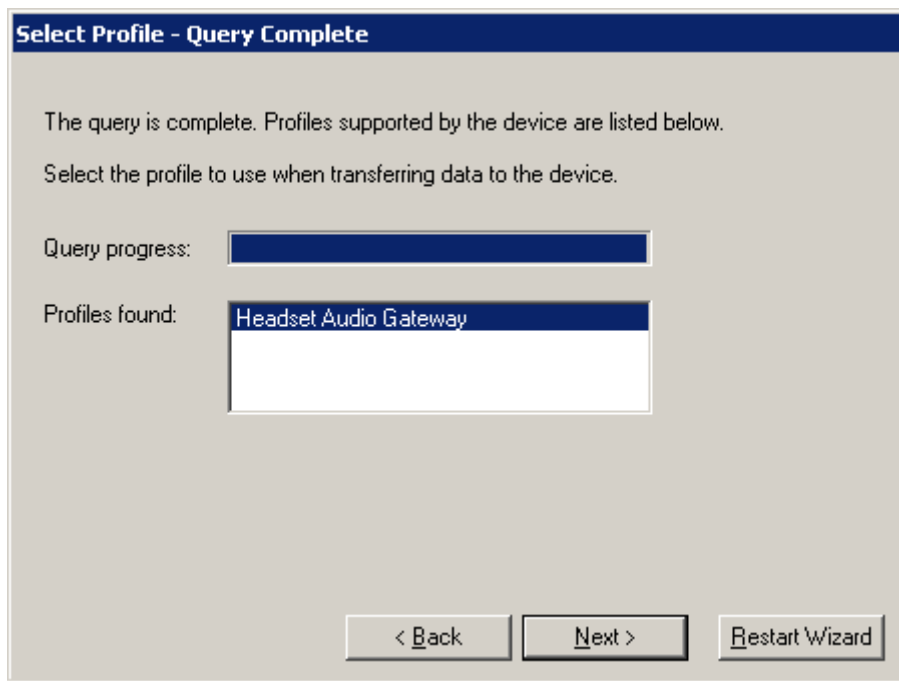


```
HCI_Evt> Inquiry_Complete
  NumResponses      : 1
HCI_Evt> Remote_Name_Request_Complete
  BD_ADDR : 0x0600A8C00108
  Name    : "Merlin's Wand"
```

Event Log | Script Log

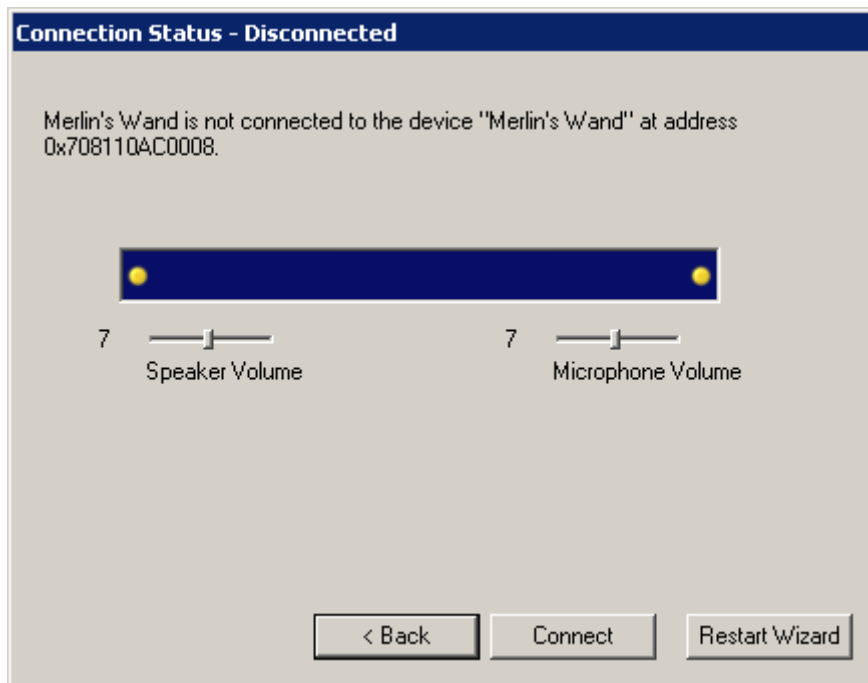
Step 5 From the list, select the device address to which you want Merlin's Wand to connect, and then click **Next**.

Merlin's Wand will query the selected device to determine its profile. When the query is complete, the Select Profile screen will open and display a list of profiles found.



Step 6 Select **Headset Audio Gateway**, then click **Next**.

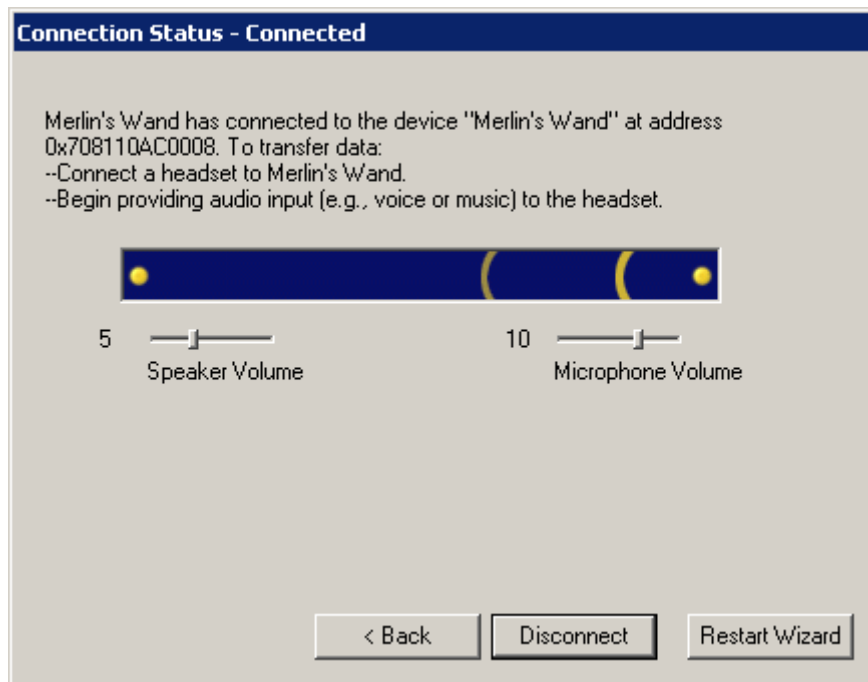
The Connection Status screen will open, indicating that Merlin's Wand is *not* connected to the device.:



Step 7 Select **Connect**.

The Connection Status screen will change to show that Merlin's Wand is connected to the device.

The Speaker and Microphone Volume levels can be adjusted by moving the sliders up or down. The level is indicated by a number, from 0 to 15, to the left of each slider.



Note If you cannot establish a connection, you can re-attempt the connection by either pressing **Back** and re-running the previous two steps, or by pressing the **Connect** button again.

Step 8 To verify that Merlin's Wand and the Bluetooth device are successfully connected, speak into the microphone on one device and listen for audio on the other.


At this point, the audio signal should transfer to the headset. Listen to the headset to see if the data transfer is successful.

Step 9 (Optional) Click the **Disconnect** button on the Connection Status screen to close the connection.

The connection between Merlin's Wand and the device will terminate, and the **Connect** button will again be available. Selecting **Connect** will reestablish the connection.

3.4 Connect to a Device: Object Push

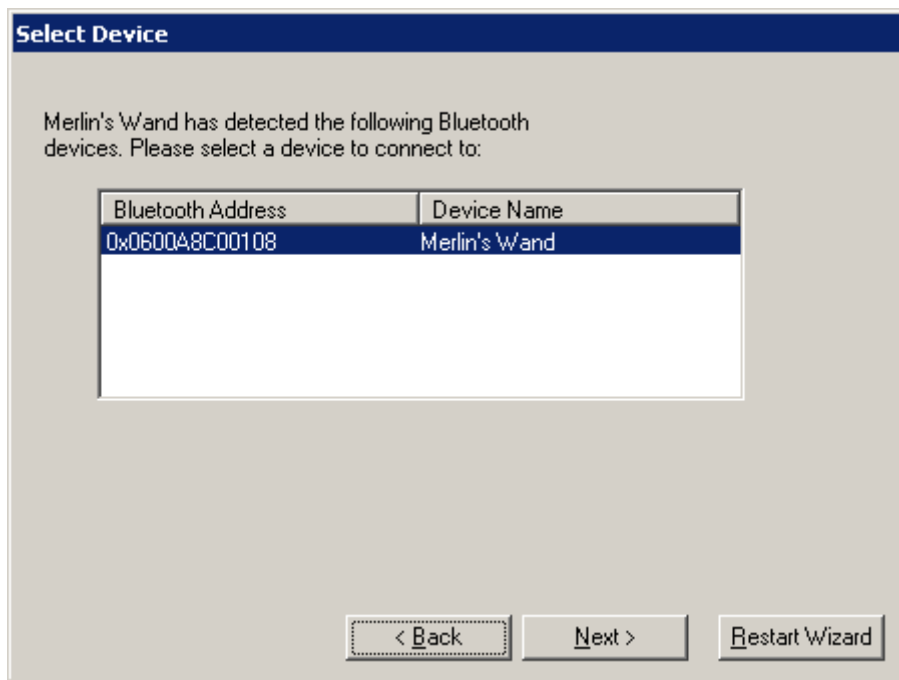
Merlin's Wand can be configured to transfer files to a Bluetooth wireless device that complies with the Object Push profile. This section shows how to configure Merlin's Wand to connect and transfer files to a Bluetooth device that supports Object Push.

- Step 1** Turn on the target device (the device that will receive the data). The target device will act as the slave unit in this example.
- Step 2** Start Bluetooth Wizard by clicking the Bluetooth Wizard icon  or by selecting **Tools > Bluetooth Wizard** from the menu bar.

Bluetooth Wizard will open in the Main Window.

- Step 3** Click the **Connect to Device** button on the Bluetooth Wizard opening screen.

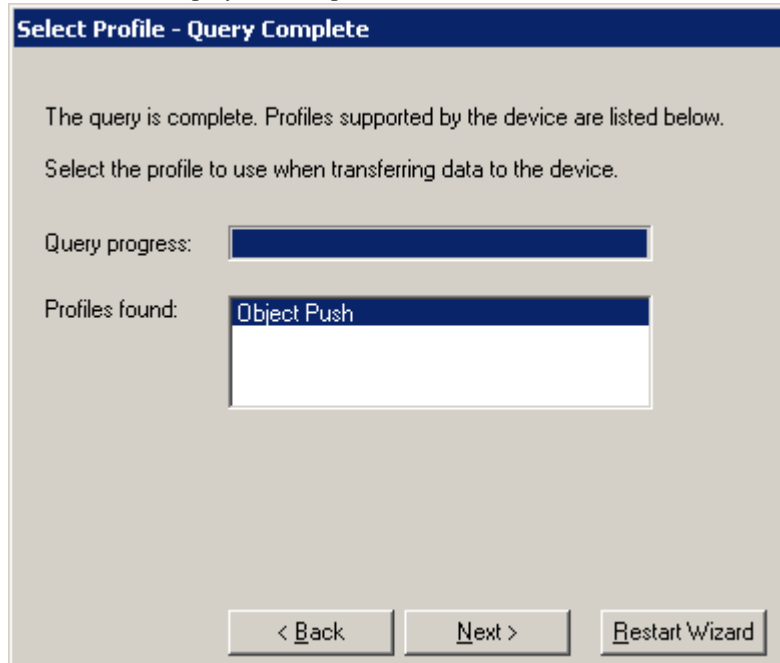
Merlin's Wand will perform a General Inquiry and collect information on local Bluetooth devices. When the search is complete, the Select Device screen will appear with a list of available devices. For details about the actions and responses in each step of the Inquiry, view the **Event Log**.



- Step 4** Select the device address to which you want Merlin's Wand

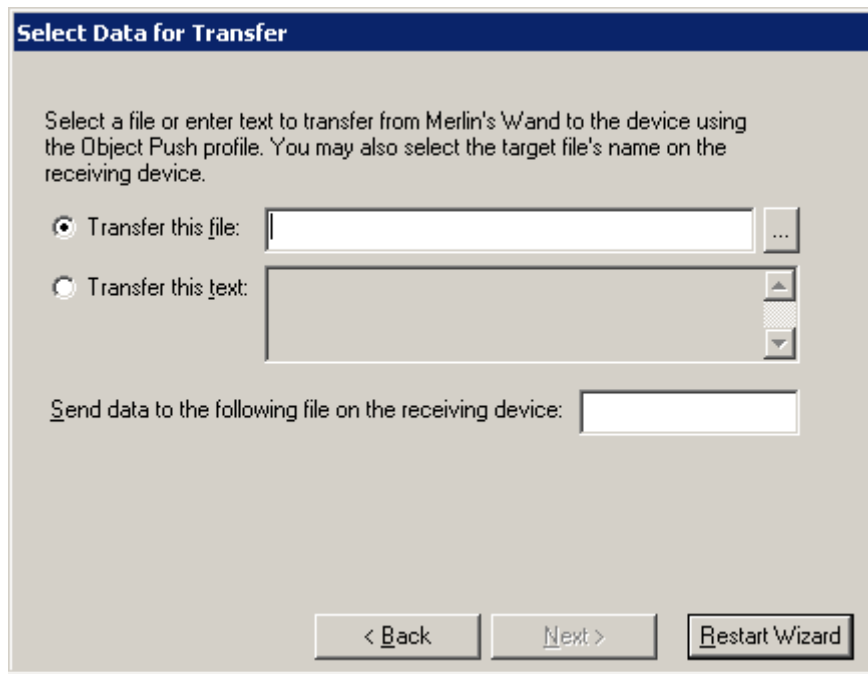
to connect, and then double-click or press **Next**.

Merlin's Wand will query the selected device to determine its profile. When the query is complete, the Select Profile screen will open and display a list of profiles found.



Step 5 Select **Object Push** from the list of displayed profiles, and then double-click or press **Next**.

The Select Data for Transfer screen will open:



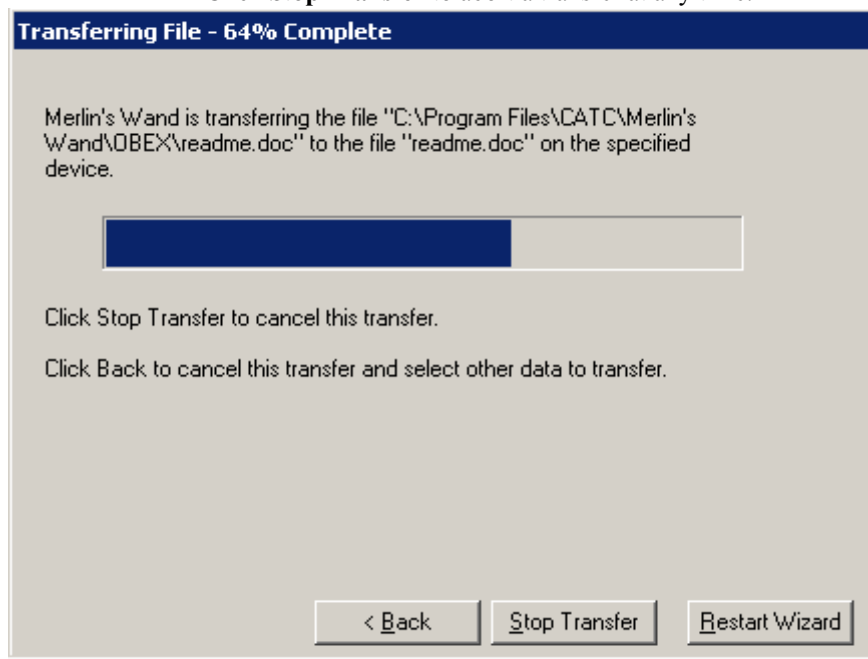
Step 6 On the Select Data for Transfer screen, there are options to transfer a file or to transfer text.

To transfer a file: Select the radio button next to “Transfer this file.” Type in a filename or locate the file by clicking the browse button to access the Open dialog. By default, the filename in the “Send data to the following file on the receiving device” box matches the name of the file to be transferred. If desired, enter a different filename in that box. When the desired file and target file's name have been entered, proceed to Step 7.

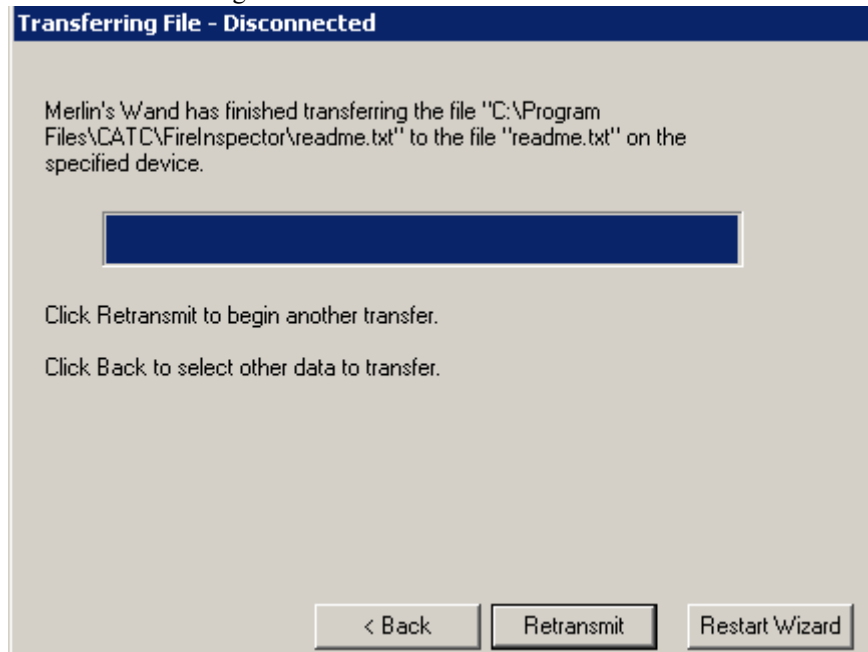
To transfer text: Select the radio button next to “Transfer this text.” Enter text in the text box. By default, the filename in the “Send data to the following file on the receiving device” box is “mw001.txt.” If desired, enter a different target filename. When the desired text and target file's name have been entered, proceed to Step 7.

Step 7 Click **Next**.

The Transferring File screen will appear. First, a connection with the remote device will be established, then data will begin transferring. A progress bar will show what percentage of the transfer has gone through. Click **Stop Transfer** to abort a transfer at any time.



When the transfer is complete, Merlin's Wand will disconnect from the target device:



At this point, you can click **Back** to transfer a different file, click **Retransmit** to send the file again, or click **Restart Wizard** to start a new Bluetooth Wizard session.

3.5 Device Emulation: Headset

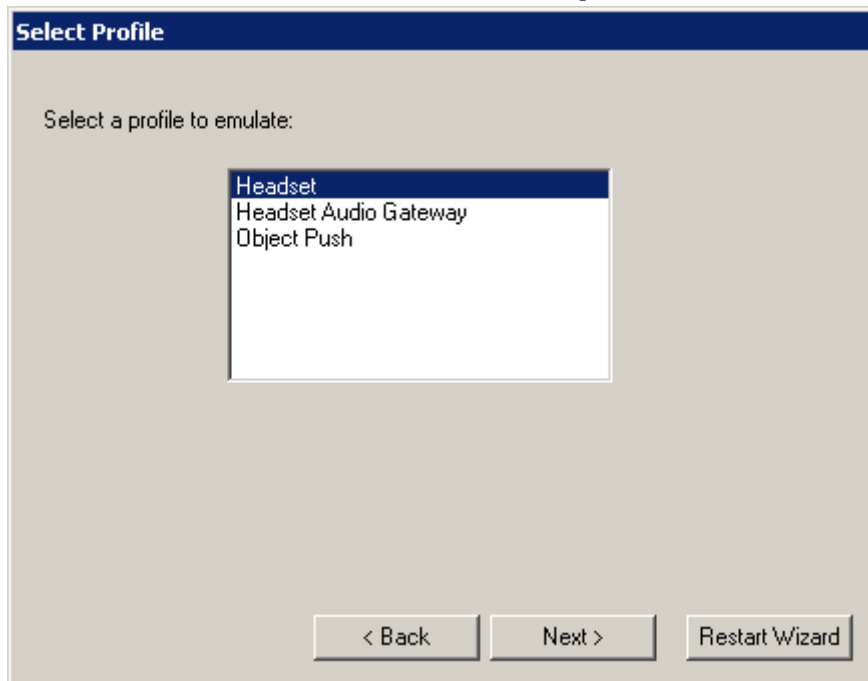
Merlin's Wand can be configured to emulate a wireless device that conforms to the Bluetooth Headset profile. The following steps show how to set up Merlin's Wand as a Headset device and connect to it with a remote Bluetooth headset.

Step 1 Start Bluetooth Wizard by clicking its icon  or by selecting **Tools > Bluetooth Wizard** from the toolbar.

The Bluetooth Wizard screen will open in the Main Window.

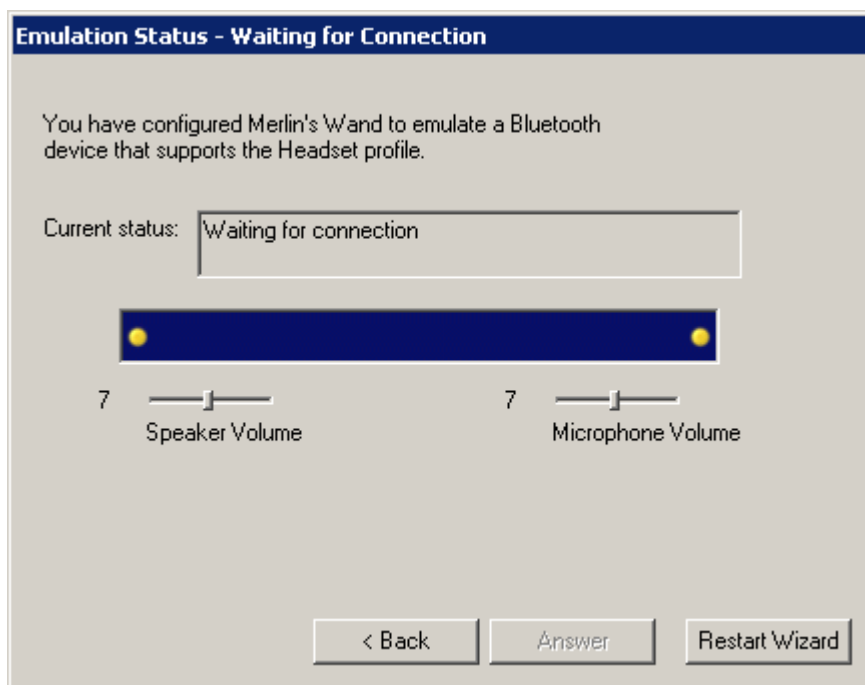
Step 2 Click **Emulate Device**.

The Select Profile screen will open.



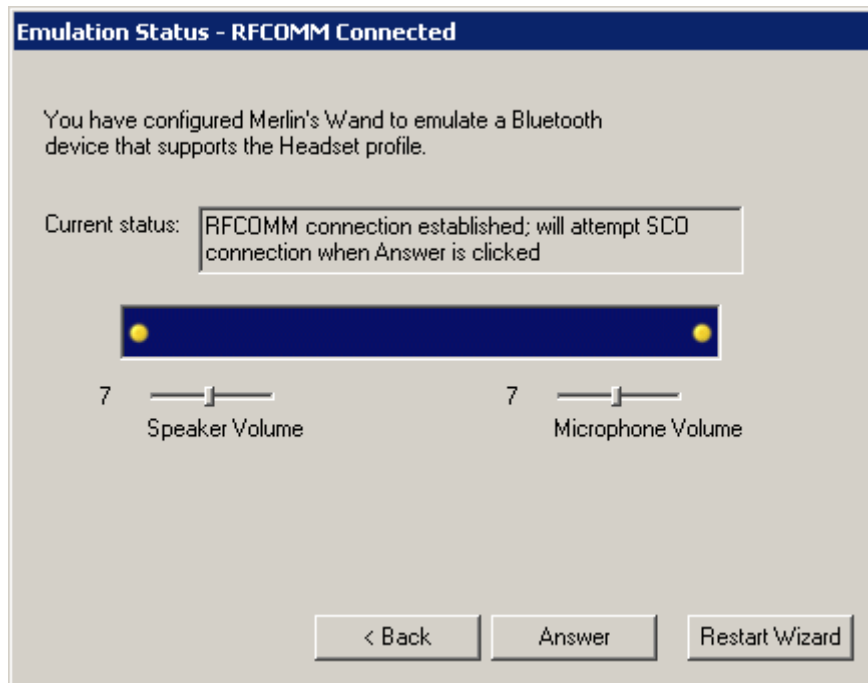
Step 3 Select **Headset**, then press **Next**.

The Emulation Status screen will open, indicating that Merlin's Wand has been configured to emulate a device that supports the Headset profile and is awaiting connection from a device.



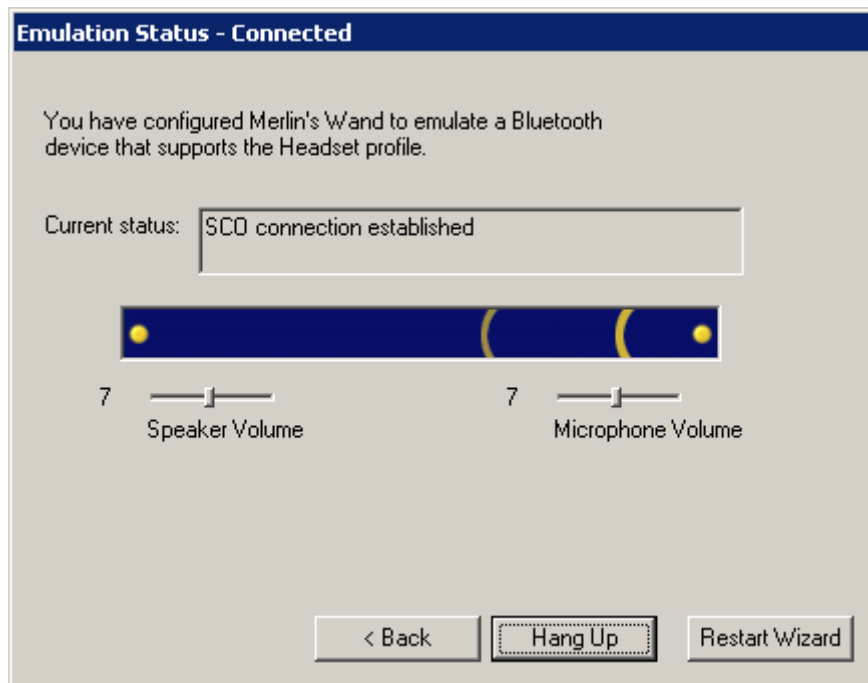
Step 4 Direct a remote Bluetooth device to connect to Merlin's Wand.

Once the connection is established, the Emulation Status screen will indicate that Merlin's Wand has an RFCOMM connection to the device.



Step 5 Click the **Answer** button to make an SCO connection with the remote device.

If the connection attempt is successful, the screen will change to indicate that an SCO connection has been established.



- Step 6** (Optional) Click the **Hang Up** button to close the connection.

The connection between Merlin's Wand and the device will terminate.

3.6 Device Emulation: Headset Audio Gateway

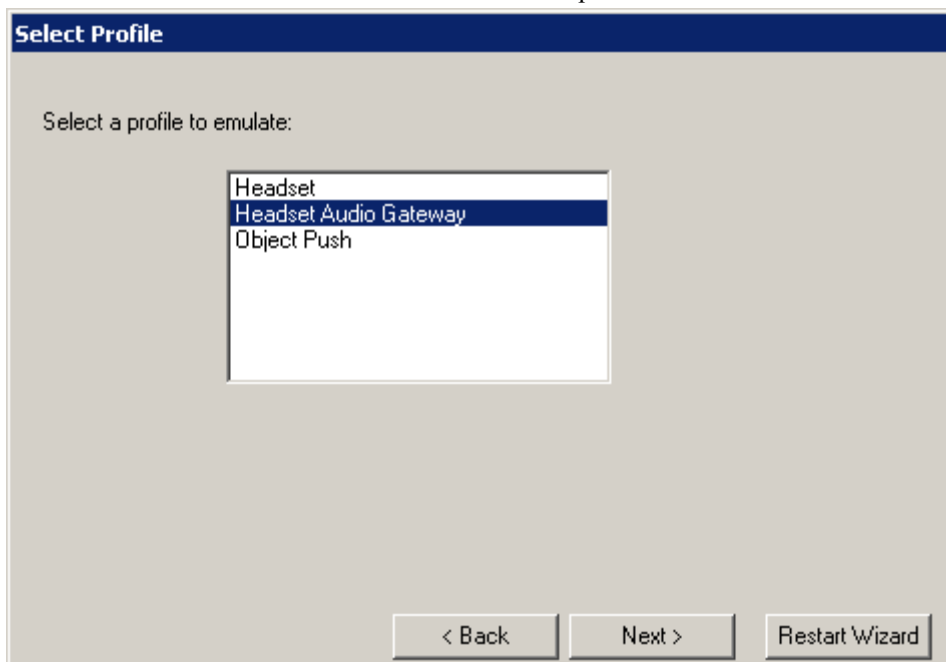
Merlin's Wand can be configured to emulate a wireless device that conforms to the Bluetooth Headset Audio Gateway profile. The following steps show how to set up Merlin's Wand as a Headset Audio Gateway device and connect to it with a remote Bluetooth headset.

- Step 1** **Start** Bluetooth Wizard by clicking its icon  or by selecting **Tools > Bluetooth Wizard** from the toolbar.

The Bluetooth Wizard screen will open in the Main Window.

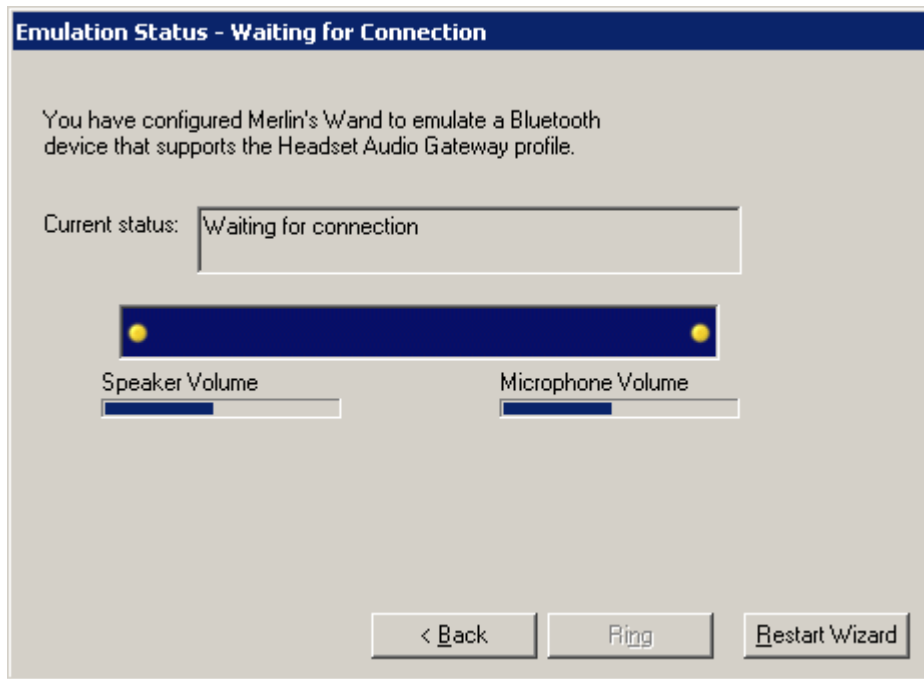
- Step 2** Click **Emulate Device**.

The Select Profile screen will open.



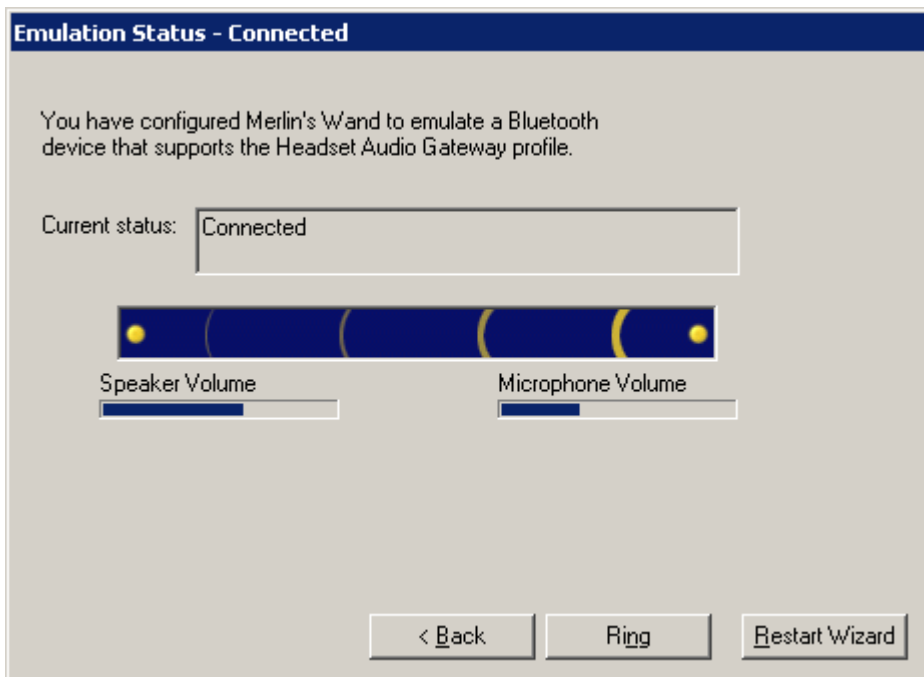
- Step 3** Select **Headset Audio Gateway**, then press **Next**.

The Emulation Status screen will open, indicating that Merlin's Wand has been configured to emulate a device that supports the Headset Audio Gateway profile and is awaiting connection from a device.



Step 4 Direct a remote Bluetooth device to connect to Merlin's Wand.


Once the connection is established, the Emulation Status screen will indicate that Merlin's Wand is currently connected to the device.



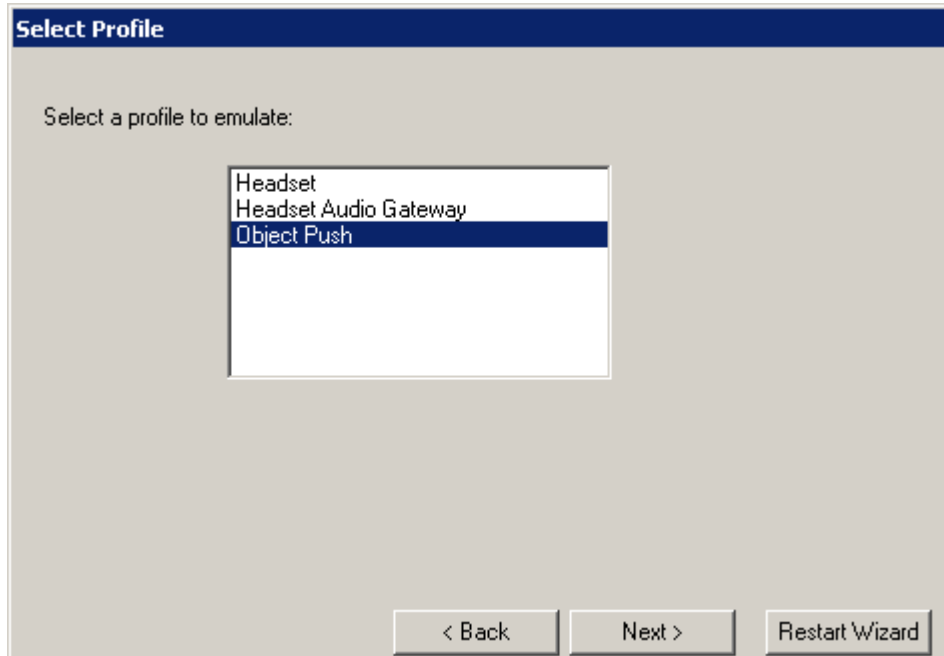
- Step 5** To verify that Merlin's Wand and the remote device are successfully connected, speak into the microphone on one device and listen for audio on the other.

3.7 Device Emulation: Object Push

Merlin's Wand can emulate the file transfer capabilities of wireless devices that support the Object Push profile through the Object Push option. Object Push emulation allows other devices to transfer files to Merlin's Wand.

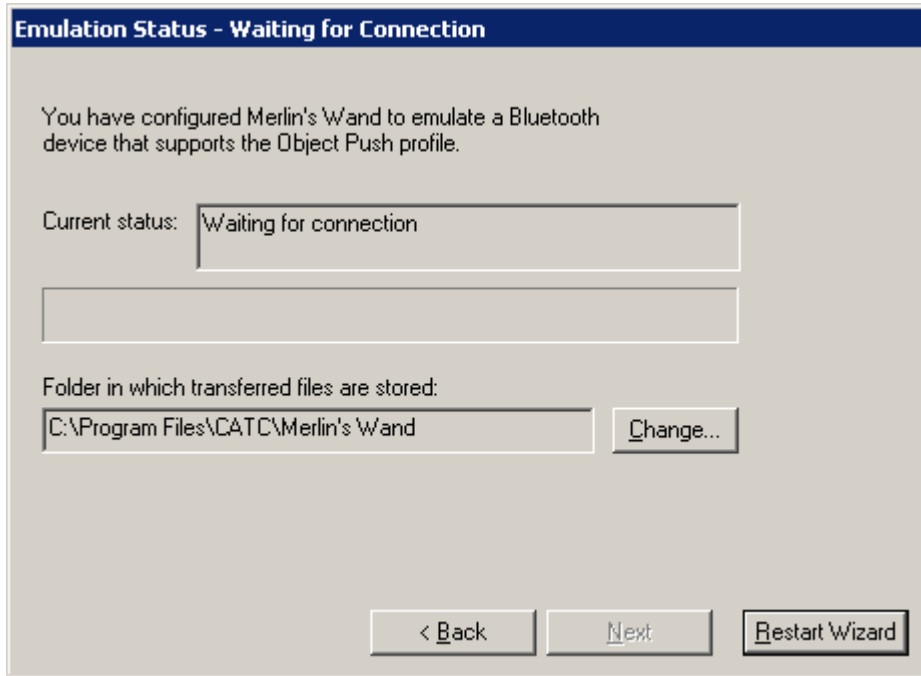
- Step 1** Start  Bluetooth Wizard.
- Step 2** Click the **Emulate Device** button on the Bluetooth Wizard opening screen.

The Select Profile screen will open.



- Step 3** Select **Object Push**, then press **Next**.

The Emulation Status screen will open, indicating that Merlin's Wand has been configured to emulate a Bluetooth device that supports the Object Push profile and is awaiting connection from a device. It is now ready to receive files.

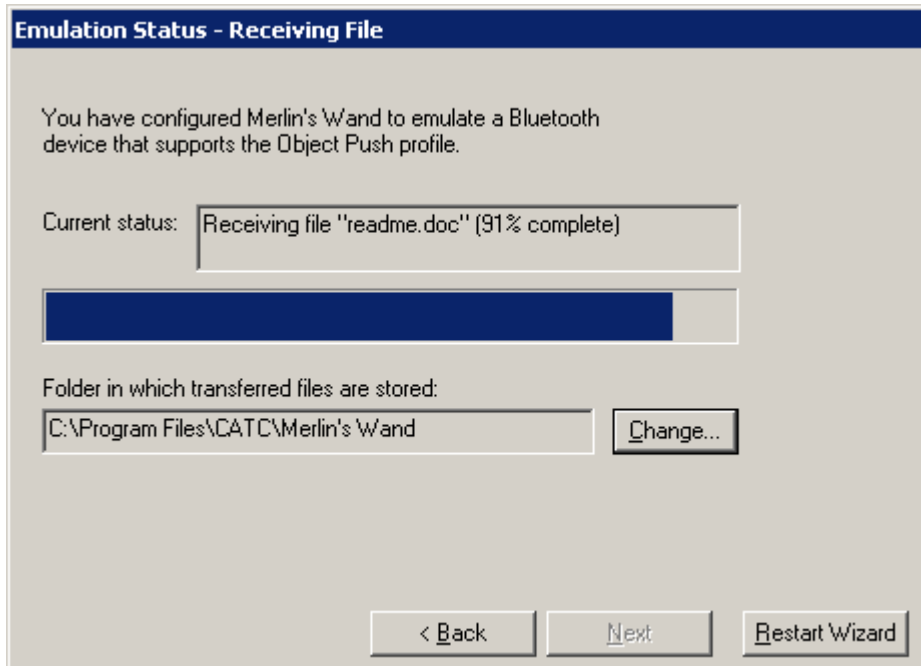


If desired, the folder in which transferred files are stored can be changed. To change it, click the Change button and select a new directory in the Browse for Folder dialog.

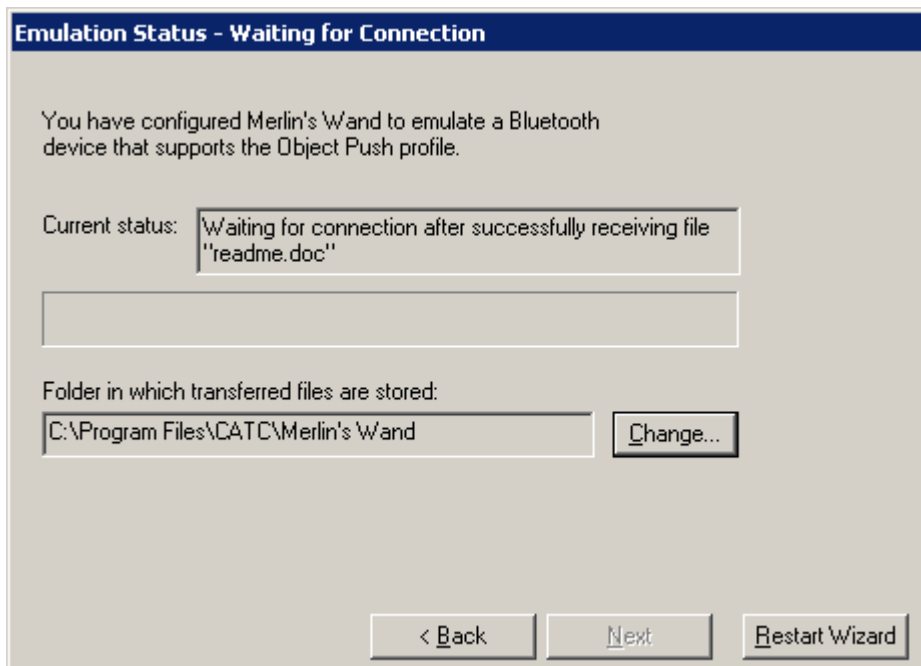
Step 4 Initiate file transfer from the Bluetooth device.

Note: If the Bluetooth device is another Merlin's Wand, follow the steps in "Connect to a Device: Object Push" on page 27.

The Emulation Status screen will show the file transfer progress:



When the transfer is complete, the Emulation Status screen will show that Merlin's Wand is waiting for a connection after having successfully received the file.



3.8 Restarting the Wizard

When working within Bluetooth Wizard, a new session may be started at any time.

Step 1 Click the **Restart Wizard** button on any screen.

- If Merlin's Wand is currently emulating a connected device, the connection will be broken, and Bluetooth Wizard will return to the default Bluetooth Wizard screen.
- If Merlin's Wand is currently connected to a device, a dialog box will ask if the connection should be terminated. Clicking the **Yes** button will cause the connection to be broken, and Bluetooth Wizard will return to the default Bluetooth Wizard screen. Clicking the **No** button will cancel the Restart Wizard request.

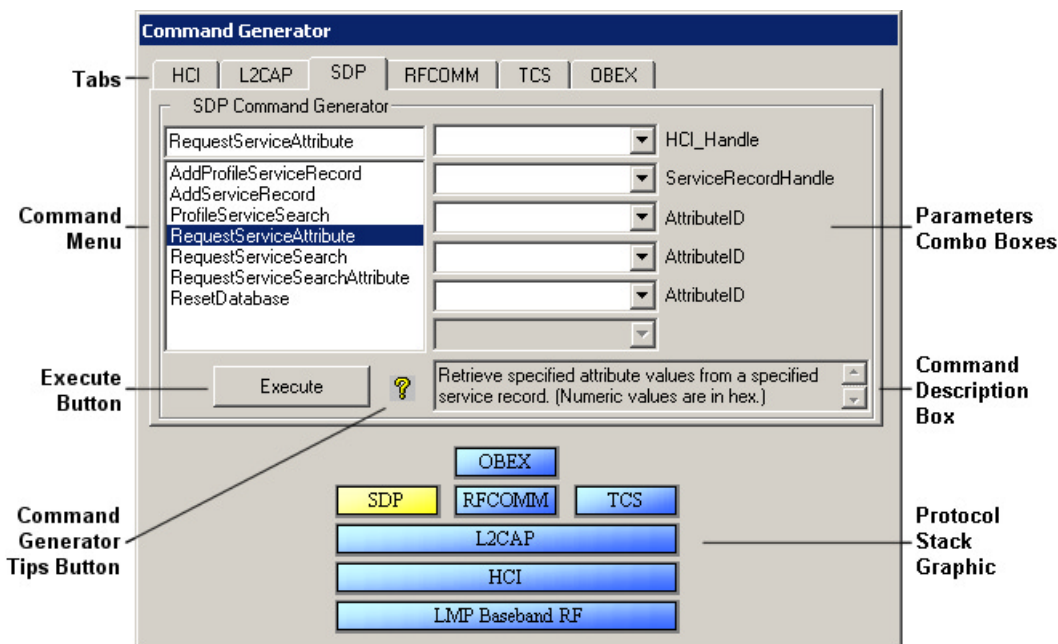
4. Command Generator

The Command Generator is a tool in Merlin's Wand that presents a menu of protocol commands so that you can select and execute any command in virtually any sequence. Command Generator thus gives maximum control over the traffic generation process.

Command Generator requires that you build connections from the Baseband level on up. This means that to establish an OBEX connection, for example, you will need to first start with Baseband and work your way up the protocol stack. You cannot simply start at a higher protocol.

The utility displays a window with tabs for these protocols: HCI (which induces Baseband, LMP and Module-Specific Commands), L2CAP, SDP, RFCOMM, TCS, and OBEX. Clicking a tab or a name in the protocol stack graphic opens a window and presents a menu of commands for that protocol.

4.1 Layout of the Command Generator

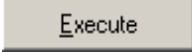


The Command Generator utility is composed of the following:

Tabs -- There is a tab for each of six protocols: HCI, L2CAP, SDP, RFCOMM, TCS, and OBEX. Clicking a tab displays the Command Menu for the chosen protocol.

Command Menus -- A list of commands is provided for each protocol.

Parameters Combo Boxes -- Parameters can be entered via the six combo boxes. One or more of the boxes may be activated, depending on which command is currently selected in the Command Menu. Parameters may either be typed into the box or chosen from a pull-down list within the box.

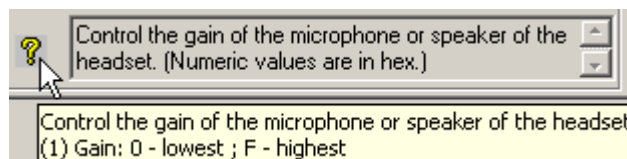
Execute -- Pressing the Execute button  will cause Merlin's Wand to run the selected command.

Command

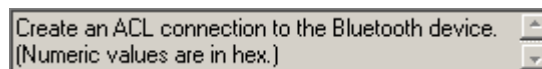
Generator Tips --

Detailed tips for each command are accessible by positioning the


mouse over the question mark  icon. A pop-up window that contains detailed information about the selected command will appear.



Command Description Box -- A short description will display in the Command Description Box when a command is selected from a Command Menu.




Protocol Stack Graphic -- At the bottom of Command Generator is the Protocol Stack Graphic, which illustrates the layers that make up the Bluetooth protocol stack. The protocols in the graphic are also clickable buttons that can be used to access the command menus for each protocol.

HCI Customized List Button -- The HCI tab has an additional button  to the left of the Execute button. It provides access to an interface that allows the user to customize the list of commands displayed in the HCI command menu.

4.2 Using Command Generator

Note: If Command Generator isn't enabled on your Merlin's Wand system, you will need to obtain a License Key from CATC before you can use it. See "License Keys" on page 16 for details.

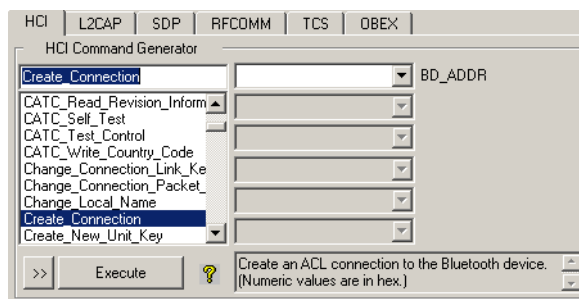
To execute commands with Command Generator:

Step 1 Click the Command Generator button  on the toolbar or select the command **Tools > Command Generator** from the menu bar.

The Command Generator utility will open.

- Step 2 Choose a protocol** to work with by clicking one of the five tabs or a layer in the Protocol Stack Graphic.

The list of available commands for the chosen protocol will display in the Command Menu.



Note: The HCI tab is displayed by default.

- Step 3 Select a command** from the Command Menu.

A description of the command will display in the Command Description Box. If the selected command is not supported, the message in the Command Description Box will read “Not supported.”

- Step 4 Enter parameters**, if required, in the Parameters Combo Boxes. Parameter boxes will be activated as appropriate for the command, with the parameter name(s) appearing to the right of the box(es). Data can be typed directly into the Parameters Combo Boxes, and some of the boxes may offer drop-down lists from which to select the appropriate parameter.


Note: Numeric values should be entered as hexadecimal unless otherwise specified.

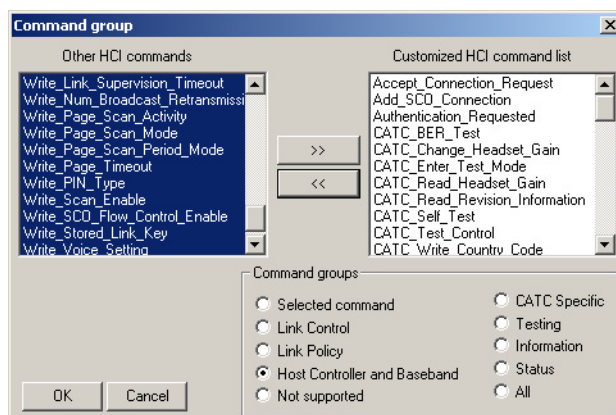
- Step 5 Click the Execute** button to run the command.

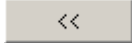
Note: While Command Generator offers maximum control over Merlin's Wand, there are times when command choices may be limited. Some lower-level connections may prevent access to commands for higher-level protocols. For example, if an L2CAP connection has been established between Merlin's Wand and a device, it is not possible to work with OBEX commands in Command Generator. Merlin's Wand will display a message to indicate that L2CAP connections must be closed before working with OBEX commands. Once the L2CAP connection is closed, the OBEX commands will be accessible.


Customizing the List of HCI Commands

The list of commands in the HCI command menu in Command Generator may be customized to display only certain commands. Since there are over 100 commands available in the HCI menu, this feature is a handy way to eliminate scrolling through a lengthy list to find commands.

Clicking the HCI Customized List button , which is located to the left of the Execute button in Command Generator, will open the Command Group interface.



To remove commands from the customized HCI command list, select the radio button beside one of the groups listed in the “Command groups” section of the interface, then press the Remove button . The selected command(s) will move into the “Other HCI commands” list.

To add commands to the customized list, select the radio button next to the group of commands that should be moved, then press the Add button . The selected command(s) will be moved from “Other HCI commands” to the customized HCI command list.

4.3 Tables of Available Commands

The following tables summarize the commands in Command Generator.

For detailed descriptions of the commands, see *Appendix A: Command Generator Command Descriptions*, on page 91.

Note “N/A” means Not Applicable. This indicates that the specified command does not have a parameter.

HCI Commands

Link Control Commands

Two sections of Link Control Commands are presented. The first section lists commands that are supported by Merlin's Wand. The second section presents commands that are *not* supported.

Supported

Commands	Parameters
Accept_Connection_Request	N/A
Add_SCO_Connection	HCI_Handle Packet Type
Authentication_Requested	HCI_Handle
Change_Connection_Link_Key	HCI_Handle
Change_Connection_Packet_Type	HCI_Handle Packet_Type
Create_Connection	BD_ADDR
Disconnect	HCI_Handle
Exit_Periodic_Inquiry_Mode	N/A
Inquiry	Inquiry_Length Num_Responses
Inquiry_Cancel	N/A
Periodic_Inquiry_Mode	Max Period Length Min Period Length Inquiry Length Num of Responses
PIN_Code_Request_Negative_Reply	BD_ADDR
PIN_Code_Request_Reply	PIN Code BD_ADDR
Read_Clock_Offset	HCI_Handle
Read_Remote_Supported_Features	HCI_Handle
Read_Remote_Version_Information	HCI_Handle
Reject_Connection_Request	N/A
Remote_Name_Request	BD_ADDR Page Scan Rep Mode Page Scan Mode Clock Offset
Set_Connection_Encryption	HCI_Handle Encryption_Enable

Not supported

Commands
Link_Key_Request_Negative_Reply
Link_Key_Request_Reply
Master_Link_Key

Link Policy Commands

Supported

Commands	Parameters
Exit_Park_Mode	HCI_Handle
Exit_Sniff_Mode	HCI_Handle
Hold_Mode	HCI_Handle Max_Interval Min_Interval
Park_Mode	HCI_Handle Beacon_Max_Interval Beacon_Min_Interval
QoS_Setup	HCI_Handle ServiceType TokenRate PeakBandwidth Latency DelayVariation
Read_Link_Policy_Settings	HCI_Handle
Role_Discovery	HCI_Handle
Sniff_Mode	HCI_Handle Max_Interval Min_Interval Attempt Timeout
Switch_Role	BD_ADDR
Write_Link_Policy_Settings	HCI_Handle Link_Policy_Settings

Not supported

All Link Policy commands are supported in Command Generator.

Host Controller & Baseband Commands

Supported

Commands	Parameters
Change_Local_Name	Name
Delete_Stored_Link_Key	BD_ADDR Delete_All_Flag
Host_Buffer_Size	ACL_Data_Length SCO_Data_Length Total_Num_ACL Total_Num_SCO
Read_Authentication_Enable	N/A

Commands	Parameters
Read_Class_of_Device	N/A
Read_Connection_Accept_Timeout	N/A
Read_Current_IAC_LAP	N/A
Read_Encryption_Mode	N/A
Read_Local_Name	N/A
Read_Link_Supervision_Timeout	HCI_Handle
Read_Number_Of_Supported_IAC	N/A
Read_Page_Scan_Mode	N/A
Read_Page_Scan_Period_Mode	N/A
Read_Page_Timeout	N/A
Read_PIN_Type	N/A
Read_Scan_Enable	N/A
Read_SCO_Flow_Control_Enable	N/A
Read_Stored_Link_Key	BD_ADDR Read_All_Flag
Read_Voice_Setting	N/A
Reset	N/A
Set_Event_Filter	FilterType FilterConditionType Condition
Set_Event_Mask	Event_Mask
Write_Authentication_Enable	Authentication_Enable
Write_Class_of_Device	CoD
Write_Connection_Accept_Timeout	Timeout
Write_Current_IAC_LAP	IAC_LAP IAC_LAP IAC_LAP IAC_LAP IAC_LAP IAC_LAP
Write_Encryption_Mode	Encryption Mode
Write_Link_Supervision_Timeout	HCI_Handle Timeout
Write_Page_Timeout	Timeout
Write_PIN_Type	PIN_Type
Write_Scan_Enable	Scan_Enable
Write_Stored_Link_Key	BD_ADDR Link_Key
Write_Voice_Settings	HCI_Handle Voice_Setting

Not Supported

Commands
Create_New_Unit_Key
Flush
Host_Number_Of_Completed_Packets
Read_Automatic_Flush_Timeout
Read_Hold_Mode_Activity
Read_Inquiry_Scan_Activity
Read_Num_Broadcast_Retransmissions
Read_Page_Scan_Activity
Read_Transmit_Power_Level
Set_Host_Controller_To_Host_Flow_Control
Write_Automatic_Flush_Timeout
Write_Hold_Mode_Activity
Write_Inquiry_Scan_Activity
Write_Num_Broadcast_Retransmissions
Write_Page_Scan_Activity
Write_Page_Scan_Mode
Write_Page_Scan_Period_Mode
Write_SCO_Flow_Control_Enable

Informational Commands*Supported*

Commands	Parameters
Read_BD_ADDR	N/A
Read_Buffer_Size	N/A
Read_Country_Code	N/A
Read_Local_Supported_Features	N/A
Read_Local_Version_Information	N/A

Not Supported

There are no unsupported Informational commands.

Status Commands*Supported*

There are no supported Status commands.

Not Supported

Commands
Get_Link_Quality
Read_Failed_Contact_Counter
Reset_Failed_Contact_Counter
Read_RSSI

Testing Commands*Supported*

Commands	Parameters
Enable_Device_Under_Test_Mode	N/A
Read_Loopback_Mode	N/A
Write_Loopback_Mode	Loopback_Mode

Not Supported

There are no unsupported Testing commands.

CATC-Specific Commands*Supported*

Commands	Parameters
CATC_BER	HCI_Handle Number_Of_Packets BER_Packet_Type Test_Data_Type Test_Data BER_Interval
CATC_Change_Headset_Gain	Device Gain
CATC_Read_Headset_Gain	Device
CATC_Read_Revision_Information	N/A
CATC_Self_Test	N/A
CATC_Write_Country_Code	Country_Code

Not supported

Commands
CATC_Enter_Test_Mode
CATC_Test_Control

L2CAP Commands

Supported

Commands	Parameters
ConfigurationResponse	Reason
ConfigurationSetup	ServiceType TokenRate TokenBucketSize PeakBandWidth Latency DelayVariation
ConnectRequest	HCI_Handle PSM Receive MTU
ConnectResponse	Response
DeregisterPsm	PSM
DisconnectRequest	CID
EchoRequest	HCI_Handle Data
InfoRequest	HCI_Handle
RegisterPsm	PSM Receive MTU
SendData	CID Data Pipe

Not Supported

Commands
GetRegisteredGroups
GroupDestroy
GroupRegister

SDP Commands

Supported

Commands	Parameters
AddProfileServiceRecord	Profile ServerChannel
AddServiceRecord	Filename Record Name Server Channel
ProfileServiceSearch	HCI_Handle Profile

Commands	Parameters
RequestServiceAttribute	HCI_Handle ServiceRecordHandle AttributeID AttributeID AttributeID
RequestServiceSearch	HCI_Handle ServiceClassID ServiceClassID ServiceClassID
RequestServiceSearchAttribute	HCI_Handle ServiceClassID ServiceClassID ServiceClassID
ResetDatabase	N/A

Not Supported

All SDP commands in Command Generator are supported.

RFCOMM Commands

Supported

Commands	Parameters
AcceptChannel	Accept
AcceptPortSettings	Accept
AdvanceCredit	(HCI/DLCI) Credit
CloseClientChannel	(HCI/DLCI)
CreditFlowEnabled	(HCI/DLCI)
DeregisterServerChannel	ServerChannel
OpenClientChannel	HCI_Handle ServerChannel MaxFrameSize Credit
RegisterServerChannel	N/A
RequestPortSettings	(HCI/DLCI) BaudRate DataFormat FlowControl Xon Xoff
RequestPortStatus	(HCI/DLCI)
SendATCommand	(HCI/DLCI) AT_Command

Commands	Parameters
SendData	(HCI/DLCI) Data Pipe
SendTest	(HCI/DLCI)
SetLineStatus	(HCI/DLCI) LineStatus
SetModemStatus	(HCI/DLCI) ModemSignals Break Length

Not Supported

All RFCOMM commands in Command Generator are supported.

TCS Commands*Supported*

Commands	Parameters
Register_Intercom_Profile	N/A
Open_TCS_Channel	HCI_Handle
Start_TCS_Call	N/A
Disconnect_TCS_Call	N/A
Send_Info_Message	Phone_Number

Not Supported

Current TCS implementation in Merlin's Wand supports only the Intercom profile. The Cordless profile that uses TCS is not currently supported by Merlin's Wand.

OBEX Commands*Supported*

Commands	Parameters
ClientConnect	BD_ADDR
ClientDisconnect	N/A
ClientGet	Object
ClientPut	Filename
ClientSetPath	Path Flags
ServerDeinit	N/A
ServerInit	N/A
ServerSetPath	Path

Not Supported

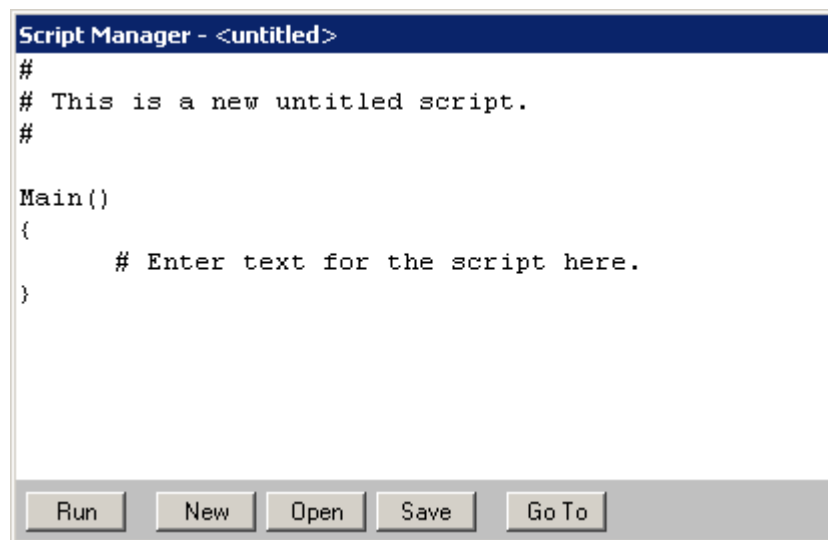
There are no unsupported OBEX commands in Command Generator.

5. Script Manager

Script Manager is a tool within Merlin's Wand that presents a text editor window for writing and executing scripts. Scripts can be used to automate Bluetooth command sequences, making the testing process more efficient.

This chapter introduces the Script Manager interface. There are a number of commands available to you for writing scripts in Merlin's Wand. Command descriptions can be found in *Appendix C: Merlin's Wand Scripting Commands*, on page 155.

5.1 Layout of the Script Manager Window



The Script Manager utility is composed of the following:

Work Area -- The Work Area is a text editor for writing new scripts or displaying and editing opened scripts.

Run Button -- Clicking the Run button saves (if needed) and executes the script that is currently displayed in the Work Area. While the script is running, the label on this button changes to Stop.

New Button -- Clicking the New button brings up a new script template in the Work Area, so that a new script may be composed. If a modified script is open when the New button is clicked, Script Manager will ask if it should be saved.

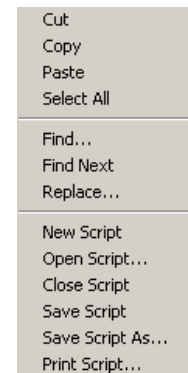
Open Button -- Clicking the Open button brings up the Open dialog, so that a script can be loaded into the Work Area. If a modified script is open when the Open button is clicked, Script Manager will ask if it should be saved.

Save Button -- Clicking the Save button saves the script that is currently open in the Work Area.

Go To Button -- Clicking the Go To button opens the Go To dialog box. Here, users may enter a line number to go to a specific part of an open script. Line numbers are displayed on the bottom right of the Merlin's Wand application, on the status bar.

Script Name and Path -- The name and path of the script that is currently open in the Work Area are displayed along the top of the Script Manager screen.

Script Manager Menu -- Right-clicking within the Work Area brings up the Script Manager menu. All filing and editing commands that can be performed in Script Manager can be accessed via this menu.



5.2 Running Scripts

Note: If Script Manager isn't enabled on your Merlin's Wand system, you will need to obtain a License Key from CATC before you can use it. See "License Keys" on page 16 for details.

Step 1 **Open Script Manager** by clicking the Script Manager icon



on the toolbar or by selecting **Tools > Script Manager** from the menu bar.

Script Manager will open.

Step 2 **Open the script** by clicking the Open button in the Script Manager window or by selecting **File > Open Script...** from the menu bar.



The Open dialog will appear.

Step 3 Navigate to the desired file and click **Open**.

The script will display in Script Manager's Work Area.

Step 4 Click **Run**.

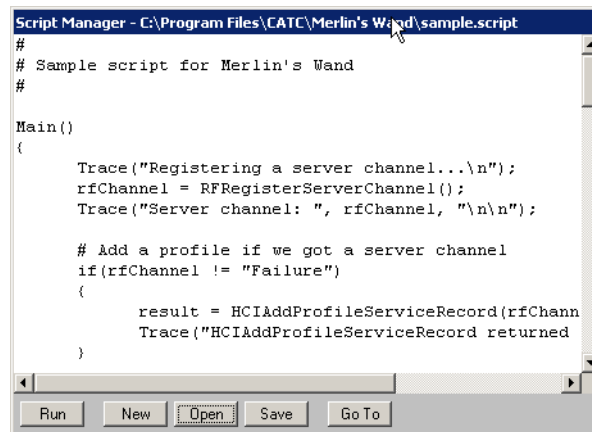
Script execution will begin, and the label of the Run button will change to Stop.

Pressing the Stop button terminates execution of the script.

The script's output can be viewed in the Script Log. If line numbers are referenced in the Script

Log, double-clicking on the line number will move the cursor to that line in Script Manager.

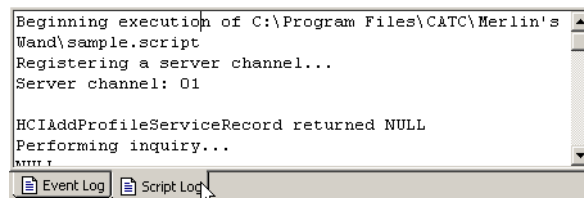
When the script has finished, the Stop button label will change back to Run.



```

Script Manager - C:\Program Files\CATC\Merlin's Wand\sample.script
#
# Sample script for Merlin's Wand
#
Main()
{
    Trace("Registering a server channel...\n");
    rfChannel = RFRegisterServerChannel();
    Trace("Server channel: ", rfChannel, "\n\n");

    # Add a profile if we got a server channel
    if(rfChannel != "Failure")
    {
        result = HCIAddProfileServiceRecord(rfChann
        Trace("HCIAddProfileServiceRecord returned
  
```



```

Beginning execution of C:\Program Files\CATC\Merlin's
Wand\sample.script
Registering a server channel...
Server channel: 01

HCIAddProfileServiceRecord returned NULL
Performing inquiry...
  
```

5.3 Writing Scripts

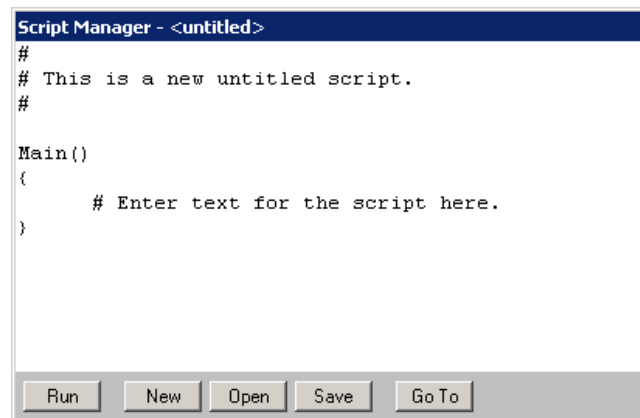
Customized scripts can be written directly in Script Manager using Merlin's Wand Scripting Commands. This allows for automating sequences of commands. There are over 100 commands available for writing custom test sequences, including basic commands and commands for: pipes, HCI,

L2CAP, SDP, RFCOMM, OBEX, and Merlin. Detailed descriptions of the commands can be found in *Appendix C: Merlin's Wand Scripting Commands*, on page 155.

Step 1 **Open Script Manager** by clicking the Script Manager icon



on the toolbar or by selecting **Tools > Script Manager** from the menu bar.



By default, Script Manager opens an “untitled” script template in the Work Area for composing a new script. If Script Manager were already open, the Work Area could be cleared by pressing the **New** button in Script Manager or by selecting **File > New Script** from the menu bar.

Step 2 **Write** the script in Script Manager's Work Area.

Step 3 **Save** the script via the **Save Script As...** command on the File menu or by clicking the **Save** button.

The Save As dialog will open. Enter a name for the script and save it as a Merlin's Wand Script file (*.script).

Step 4 If desired, **Close** the script by selecting **File > Close Script** from the menu bar.

5.4 Sample Scripts

Two sample scripts have been provided with Merlin's Wand to demonstrate how Script Manager works. Running **sample.script** will cause Merlin's Wand to attempt to connect to another device. **Sample-2.script** demonstrates using RFCOMM to wait for and receive data, as well as demonstrating the use of pipes. The default location of the scripts is the directory where the application is installed, which is usually **C:\Program Files\CATC\Merlin's Wand**. An additional script, **Script Controlling Merlin Interface**, is included in this section. It shows how to use a script to control the CATC Merlin Bluetooth Analyzer via Merlin's Wand.

sample.script

This script demonstrates several common Merlin's Wand functions.

```

Main()
{
    Trace("Registering a server channel...\n");
    rfChannel = RFRegisterServerChannel();
    Trace("Server channel: ", rfChannel, "\n\n");

    # Add a profile if we got a server channel
    if(rfChannel != "Failure")
    {
        result = SDPAddProfileServiceRecord(rfChannel,
"ObjectPush");
        Trace("SDPAddProfileServiceRecord returned ", result,
"\n");
    }

    Trace("Performing inquiry...\n");

    Devices = DoInquiry();

    Trace(Devices, "\n");

    class = GetDeviceClass();
    Trace("Device class is: ", class, "\n");

    # Set a new device class
    SetDeviceClass(0x010203);

    class = GetDeviceClass();
    Trace("Device class is now: ", class, "\n");

    # Get the names
    index = 0;
    while(device = Devices[index])
    {
        deviceName = GetRemoteDeviceName(device);
        Trace("Device ", index, ": ", deviceName, "\n");
        index = index + 1;
    }

    # Read the current accessible mode
    Trace("Current accessible mode is: ",
HCIReadScanEnable(), "\n");

    # Set the accessible mode
    result = HCIWriteScanEnable("GENERAL_ACCESSIBLE");
    Trace("HCIWriteScanEnable returned ", result, "\n");

```

```
# Read the new accessible mode
Trace("Current accessible mode is: ",
HCIReadScanEnable(), "\n");

Trace("Connecting to ", Devices[0], "\n");

# Connect to the first device in the list.
ACLHandle = Connect(Devices[0]);
Trace("ACL Handle: ", ACLHandle, "\n");

# Get some SDP query information
Trace("\nSDP query results:\n");
serverChannel = SDPQueryProfile(Devices[0], "Headset");
Trace("Headset: Server channel is: ", serverChannel,
"\n\n");

# Establish an RFCOMM connection
Trace("\nEstablishing RFCOMM connection:\n");
result = RFOpenClientChannel(Devices[0], serverChannel);
Trace("RFCOMM connection: ", result[0], "\n");
DLCI = result[1];

# Send some data over our new RFCOMM connection
data = "ATDT5551212";
Trace("Sending ", data, " to ", Devices[0], "... \n");
result = RFSendData(Devices[0], DLCI, data);

Trace("RFSendData returned ", result, "\n\n");

# Close the RFCOMM connection
result = RFCloseClientChannel(Devices[0], DLCI);
Trace("RFCOMM disconnection: ", result, "\n");

Trace("\nAttempting to make an SCO connection...\n");

result = HCIAddSCOConnection(Devices[0], ["DM1",
"HV1"]);

Trace("HCIAddSCOConnection returned ", result, "\n");

Trace("\nWaiting 2 seconds...\n");

# Wait for 2 seconds
Sleep(2000);

# Close the SCO connection
Trace("Close SCO connection: ",
HCIRemoveSCOConnection(Devices[0]), "\n");

# Disconnect from the device
status = Disconnect(Devices[0]);
```

```

    Trace("Disconnect returned: ", status, "\n\n");
}

```

Sample-2.script

```

# Sample2.script
#
# Demonstrates using RFCOMM to wait for and receive data as
# well as the use of pipes.
#
# A Transmit pipe called "SamplePipe" should be created in
# the Data Transfer Manager
# before the script is run. This pipe should contain a small
# amount of data that will be sent
# back to the host whenever data is received by the script.

Main()
{
    # Do some pipe tests.
    result = DeletePipe("TestPipe", "receive");
    Trace("DeletePipe returned ", result, "\n");

    result = OpenPipe("TestPipe", "receive");
    Trace("OpenPipe returned ", result, "\n");

    # WritePipe only supports receive pipes.
    result = WritePipe("TestPipe", "This is some data.");
    Trace("WritePipe returned ", result, "\n");

    result = ClosePipe("TestPipe", "receive");
    Trace("ClosePipe returned ", result, "\n");

    result = OpenPipe("TestPipe", "receive");
    Trace("OpenPipe (2) returned ", result, "\n");

    result = ReadPipe("TestPipe", "receive", 100);
    Trace("ReadPipe returned ", result, "\n");

    #result = ClosePipe("TestPipe", "receive");
    #Trace("ClosePipe (2) returned ", result, "\n");

    channel = RFRegisterServerChannel();
    Trace("Listening on server channel ", channel, "... \n");

    result = OpenPipe("DataIn", "receive");
    Trace("OpenPipe returned ", result, "\n");

    # Wait for a connection
    status = RFWaitForConnection();
}

```

```

Trace("RFWaitForConnection returned ", status, "\n");
while(1)
{
    results = RFReceiveData();
    Trace("RFReceiveData returned ", results[0], "\n");
    if(results[0] != "Success")
    {
        ClosePipe("DataIn", "receive");
        Trace("Exiting.\n");
        return results[0];
    }
    Trace("Received ", results[1], " bytes:\n");
    Trace(results[2]);
    Trace("\n");
    result = WritePipe("DataIn", results[2]);
    Trace("WritePipe returned ", result, "\n");

    # Send some data back from a pipe.
    result = RFSendDataFromPipe("CONNECTED_DEVICE",
channel, "SamplePipe");
    Trace("SendData returned ", result, "\n");
}
}

```

Script Controlling Merlin Interface

This script demonstrates how to control various Merlin functions via a Merlin's Wand script.

```

Main()
{
    result = MerlinStart();
    Sleep(2000);

    if(result == "Failure")
        return result;

    Trace("Merlin started\n");

    MerlinSetRecordingOptions("C:\\Program
Files\\CATC\\Merlin\\1.rec");
    Trace("MerlinSetRecordingOptions\n");

    MerlinSetDisplayOptions("C:\\Program
Files\\CATC\\Merlin\\1.opt");
    Trace("MerlinSetDisplayOptions\n");

    result = MerlinSetEncryptionPIN('000102030405', "1234");
    if(result != "Success")

```

```
        return result;

    Trace("MerlinSetEncryptionPIN\n");

    result = MerlinSetEncryptionLinkKey('000102030405',
"123456");
    if(result != "Success")
        return result;

    Trace("MerlinSetEncryptionLinkKey\n");

    result = MerlinStartRecording();
    if(result != "Success")
        return result;

    Trace("MerlinStartRecording\n");

    Sleep(10000);
    MerlinStopRecording();
    Trace("MerlinStopRecording\n");

    MerlinResetAllEncryptionOptions();
    MerlinStop();
    Trace("MerlinStop\n");
}
```



6. Device Search and Device List Pop-Up Menu

The Device Search and Device List Pop-Up Menu tools offer shortcut methods for steps that are commonly performed at the beginning of the connection process. They can be used for some commands that would otherwise need to be done in Command Generator.

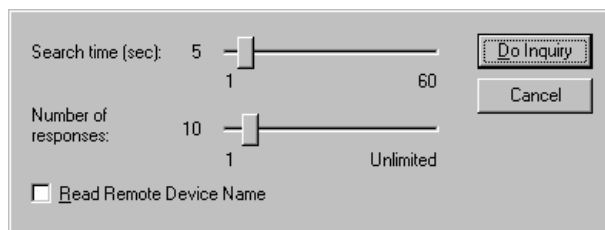
6.1 Device Search

Merlin's Wand can perform an inquiry to find local Bluetooth wireless technology devices via the Device Search tool. Information about the devices that are found are then shown in the Device List.

To perform a Device Search:

- Step 1** Open the **Device Search dialog** by clicking the Device Search icon  on the toolbar or by selecting **Tools > Device Search** from the menu bar.

The Device Search dialog will open.



Device Search presents the following search options:

Search Time -- Sets the duration of the inquiry, in seconds. The default search time is five seconds.

Number of Responses -- Sets the maximum number of responses for which data should be collected. The default number of responses is ten.

Read Remote Device Name -- Selecting this option will cause Merlin's Wand to collect name information from the remote devices it finds. This option is not selected by default.

Note: The reading of names occurs after the search has finished; therefore, processing the entire search will take longer if this option is selected. For example, if the Search Time is set to 5 seconds, and 30 devices are found within 5 seconds, the entire search process will take much longer than 5 seconds because each device will be contacted individually and asked for its name. This could add considerable time to the search, especially if some of the devices found in the search have gone out of range or been turned off.

Step 2 (Optional) Set the values for Search Time, Number of Responses and Read Remote Device Name.

Step 3 Click **Do Inquiry**.

Merlin's Wand will search for devices.

Step 4 To see the results of the search, click the **Device List** tab in the Device Status window. To see the commands and responses from the Inquiry, view the Event Log in the Logs window.

BD_ADDR	State
i 0x738010AC0108	In Range
i 0x718010AC0008	In Range
i 0xC88010AC0008	In Range

6.2 Device List Pop-Up Menu

The Device List Pop-Up Menu presents options for setting up ACL and audio (SCO) connections, displaying remote device information, and terminating connections. The Pop-Up Menu can be accessed by right-clicking on one or more devices in the Device List. It can be used as an alternative to Bluetooth Wizard, Command Generator, and Script Manager for performing some commands.



Note: The Device List Pop-Up Menu is not accessible while the Bluetooth Wizard is running.

Create an ACL Connection

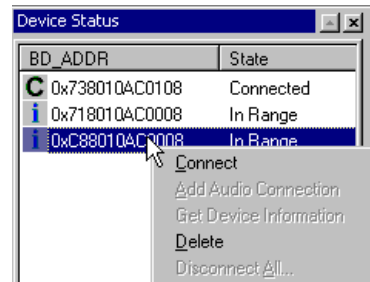
An HCI ACL connection to a remote device can be established via the Device List Pop-Up Menu.

Note: The following instructions assume that a Device Search has been performed and devices are displayed in the Device List. For information about performing a device search, please see Section 6.1, "Device Search," on page 63.



Step 1 **Open the Pop-Up Menu** by right-clicking on the target device in the Device List.

The Device List Pop-Up Menu will open.

Step 2 Choose **Connect** from the menu.



The status of the target device should change from **In Range** to **Connected** in the Device List. The Piconet tab should now indicate that Merlin's Wand has an ACL connection to the target device.

 **BD_ADDR : 0x0600A8C00008, Role : Slave**
 HCI ACL Handle : 0x0000

Establish an Audio Connection

An HCI SCO connection to a device that supports audio connections can be established via the Device List Pop-Up menu.

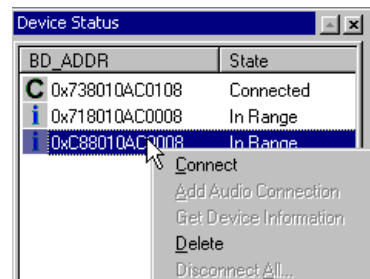
Note: In order to verify that Merlin's Wand and a Bluetooth wireless audio device are successfully connected, a headset will need to be plugged into the audio port on Merlin's Wand. Be sure that the headset is plugged in before initializing the connection between Merlin's Wand and a Bluetooth device.

Note: The following instructions assume that a Device Search has been performed and devices are displayed in the Device List. For information about performing a device search, please see Section 6.1, "Device Search," on page 63.

Step 1 **Open the Pop-Up Menu** by right-clicking on the target device in the Device List.

The Device List Pop-Up Menu will open.

Step 2 Choose **Connect** from the



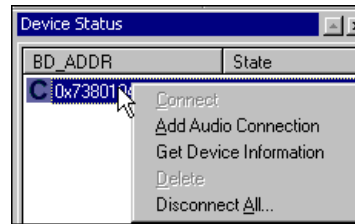
menu.

The status of the target device should change from **In Range** to **Connected** in the Device List. The Piconet tab should now indicate that Merlin's Wand has an ACL connection to the target device.

[-] **C** **BD_ADDR : 0x0600A8C00008, Role : Slave**
 [H] HCI ACL Handle : 0x0000

- Step 3 Reopen the Pop-Up Menu**
 by right-clicking on the target
 device in the Device List.

The Device List Pop-Up Menu will open. If the remote device supports audio connections and Merlin's Wand is connected to it, then the **Add Audio Connection** command should be available.



- Step 4 Select Add Audio Connection from the menu.**

The status of the target device will not change in the Device List; however, the Piconet tab should indicate that Merlin's Wand has an SCO connection to the device.

[-] **C** **BD_ADDR : 0x0600A8C00008, Role : Slave**
 [H] HCI ACL Handle : 0x0000
 [S] HCI SCO Handle : 0x0009

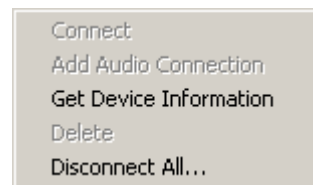
- Step 5 To verify that Merlin's Wand and the Bluetooth device are successfully connected, speak into the microphone on one device and listen for audio on the other.**

Display Device Information

Note: The following instructions assume that Merlin's Wand is currently connected to a remote device.

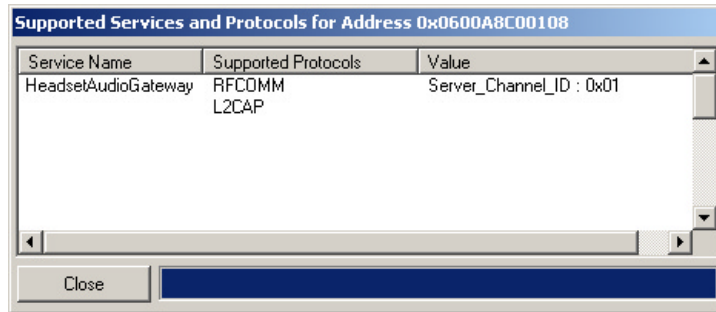
- Step 1 Open the Pop-Up Menu**
 by right-clicking on the target
 device in the Device List.

The Device List Pop-Up
 Menu will open.



Step 2 Select Get Device Information.

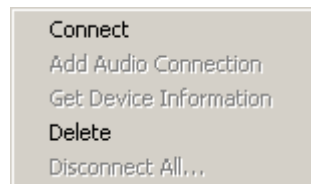
The Supported Services and Protocols window will open. The Service Name, Supported Protocols, and Value for the target device will be displayed in the window.

**Delete a Device**

Devices that are not connected may be removed from the Device List via the Device List Pop-Up Menu. This is useful when there are many devices displayed in the Device List -- non-target devices can be deleted from the list, making it easier to navigate. This option can also be used to remove devices that are no longer in range, but are still displayed in the list.

Step 1 Open the Pop-Up Menu by right-clicking on one or more devices.

The Device List Pop-Up Menu will open.



Note: To delete more than one device at a time, either

- (a) Select non-consecutive devices by Ctrl + clicking on each device to be deleted, or
- (b) Select consecutive devices by Shift + clicking on the first and last devices to be deleted,

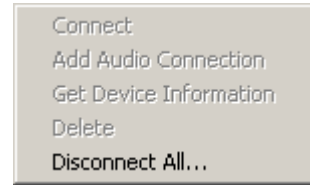
then right-click on one of the selected devices.

Step 2 Select Delete.

The device(s) will be removed from the Device List.

Disconnect All

A fast and easy way to terminate all connections that Merlin's Wand has established with remote devices is to use the **Disconnect All...** command on the Device List Pop-Up Menu.

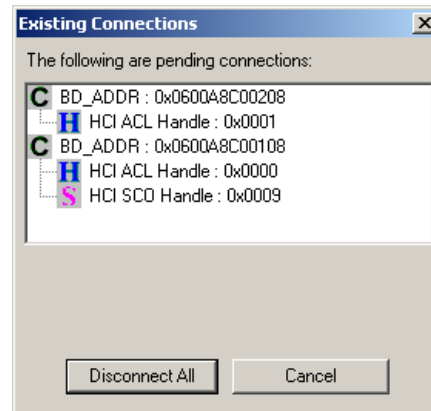


Step 1 Open the **Pop-Up Menu** by right-clicking on a device.

Step 2 Select **Disconnect All...**

The Existing Connections dialog will open, displaying all pending connections.

Step 3 Click the **Disconnect All** button in the Existing Connections dialog to close the connections, or click **Cancel** to leave them open.



Note: When switching between Bluetooth Wizard, Command Generator and Script Manager, all connections that have been established between Merlin's Wand and another Bluetooth device should be closed. However, expert users may choose to leave the connections open. If a connection is left open and you attempt to switch tools, Merlin's Wand will prompt you to close the connections. Choosing **Disconnect All** will close the connections. Choosing **Cancel** will leave the connections open, but some commands might not work properly in the other tool. When switching to Bluetooth Wizard, any open connections *must* be closed.

7. Data Transfer Manager and Data Pipes

Data Transfer Manager is a special tool for creating *pipes*. A pipe is a file or message that has been specially prepared for transmission over an RFCOMM or L2CAP channel. Pipes are necessary with these protocols because RFCOMM and L2CAP only transfer raw data. Pipes are set up to designate the source of the raw data -- either a file or text entered by the user.

Data Transfer Manager is designed to work in conjunction with Command Generator and Script Manager. Use Data Transfer Manager to prepare data, then use Command Generator or Script Manager to transfer it.

Data Transfer Manager is also used to view any data that is being received by Merlin's Wand over L2CAP or RFCOMM channels. If data is being received, the Data Transfer Manager window will automatically create a Receive Pipe and display the arriving data.

Data Transfer Manager contains the following elements:

Data Transmit page -- The Data Transmit page is used to create pipes. Pipes can be created from files or text. For more information on data transmit pipes, see Section 7.1, "Creating Data Pipes," on page 69 and Section 7.2, "Using Data Pipes," on page 70.

Data Receive page -- The Data Receive page is used to view data that has been received by Merlin's Wand. For more information, see Section 7.3, "Receive Pipes," on page 72.

7.1 Creating Data Pipes

Data Transfer Manager is used to prepare data for transmission over RFCOMM and L2CAP channels. To prepare the data, pipe files are created.

- Step 1** **Open Data Transfer Manager** by clicking on the Data Transfer Manager icon  on the Toolbar or selecting **Tools** > **Data Transfer Manager** from the menu bar.

Data Transfer Manager will open, displaying the Data Transmit page.

Step 2 **Name the pipe** by typing a name into the text box labeled **Pipe Name**.

Step 3 *To create a pipe from a file:*

Select the **From file** radio button. Type in a filename and path or navigate to the desired file by clicking the browse button **...** to bring up the Open dialog.

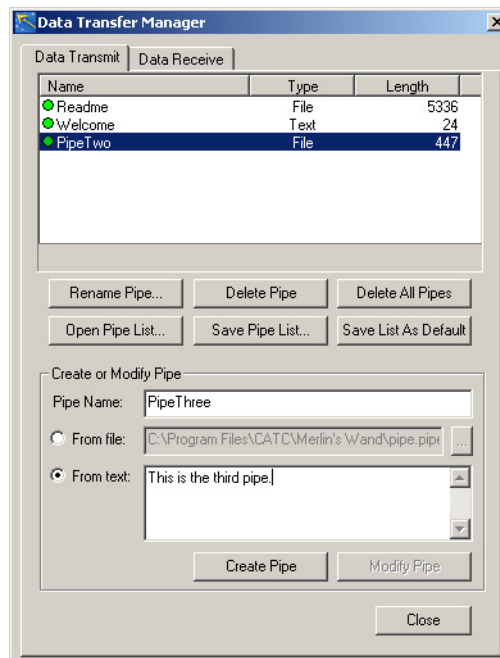
To create a pipe from text:

Select the **From text** radio button and type text into the box to its right.

Step 4 Click the **Create Pipe** button.

The pipe will be created, and its name, type and length will be displayed in the list at the top of the Data Transmit page.

Name	Type	Length
Readme	File	5336
Welcome	Text	24
PipeTwo	File	447
PipeThree	Text	23




7.2 Using Data Pipes

Data Transfer Manager works in conjunction with Command Generator and Script Manager to provide an easy way to transfer files between Merlin's Wand and a Bluetooth wireless device over an RFCOMM or L2CAP channel.

Transfer Data Using Command Generator

Note: A data pipe needs to be created in Data Transfer Manager before the data can be transferred. See Section 7.1, "Creating Data Pipes," on page 69 to learn how to do this.

Step 1 **Open Command Generator** by clicking on the Command Generator icon  on the Toolbar or selecting **Tools > Command Generator** from the menu bar.

Step 2 **Establish an RFCOMM or L2CAP connection** with the target device.

To establish an RFCOMM connection:

- (a) Using the Device List pop-up menu, create an ACL connection. To find out how to do this, see "Create an ACL Connection" on page 64.
- (b) In Command Generator, open the RFCOMM tab.
- (c) Select OpenClientChannel from the list of commands. Enter the HCI_Handle (this is shown on the Piconet tab) and ServerChannel parameters. The default values for MaxFrameSize and Credit can be used. When the parameter values are all entered, press Execute.

The RFCOMM connection will show up in the Piconet.

To establish an L2CAP connection:

- (a) Using the Device List pop-up menu, create an ACL connection. To find out how to do this, see "Create an ACL Connection" on page 64.
- (b) In Command Generator, open the L2CAP tab.
- (c) Select RegisterPsm from the list of commands. Enter the PSM and Receive MTU parameters, then press Execute. Note: RegisterPsm must also be executed for the target device.
- (d) Select ConnectRequest from the list of commands. Enter the HCI_Handle, PSM, and Receive MTU parameters, then press Execute.

The L2CAP connection will display on the Piconet tab.

Step 3 **Select SendData** from the RFCOMM or L2CAP menu, depending on which type of connection was established in Step 2.

The first two Parameters Combo Boxes will become activated, indicating that the (HCI / DLCI) for RFCOMM, or CID for L2CAP, and Data Pipe parameters are required for this command.

Step 4 **Enter or select the appropriate (HCI / DLCI) or CID channel and Data Pipe name** in the Parameters Combo

Boxes.

Step 5 Click the **Execute** button to send the data pipe.

The Event Log will show the transfer of data from Merlin's Wand to the target device.

Access Pipes Using Script Manager

There are functions available in the scripting API to access pipes. They are: OpenPipe, ClosePipe, ReadPipe, WritePipe, and DeletePipe. Please see Section C.3, "Pipe Commands," on page 160 for more information.

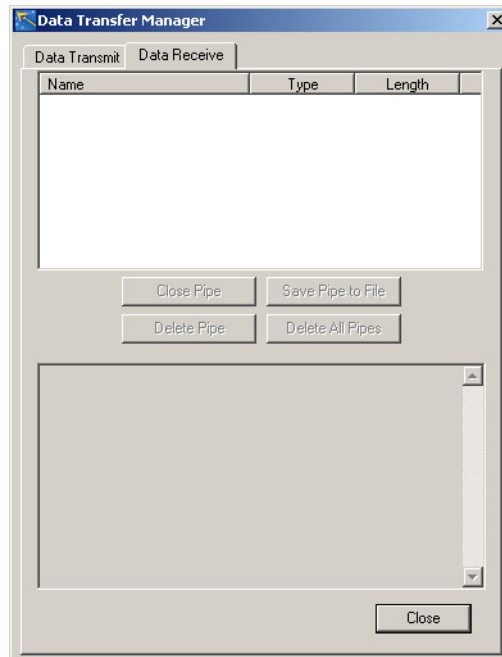
For a demonstration of using RFCOMM to wait for and receive a data pipe, see **Sample-2.script** in Script Manager.

Note: A data pipe needs to be created in Data Transfer Manager before the data can be transferred. See Section 7.1, "Creating Data Pipes," on page 69 to learn how to do this.

7.3 Receive Pipes

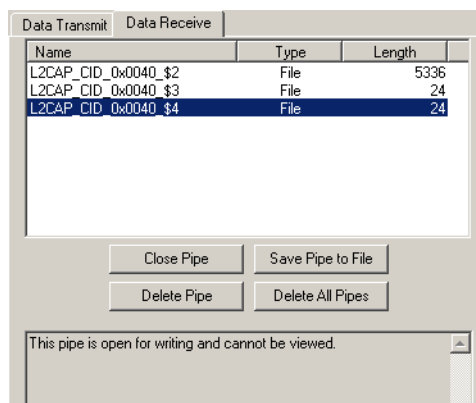
Receive pipes are created automatically in the Data Receive page of Data Transfer Manager when Merlin's Wand receives data from an L2CAP or RFCOMM connection.

A receive pipe is a pipe that is used to receive data until the connection terminates. At that point, the pipe can be closed, saved to a file, or deleted. Note that unless a receive pipe is closed, any additional data that's received will be put into that pipe, even if the new data represents a different file transfer. To ensure that different files will be put into separate pipes, each pipe should be closed after a connection has completed. This way, a new receive pipe will be created when subsequent data arrives.



7.4 Closing Pipes

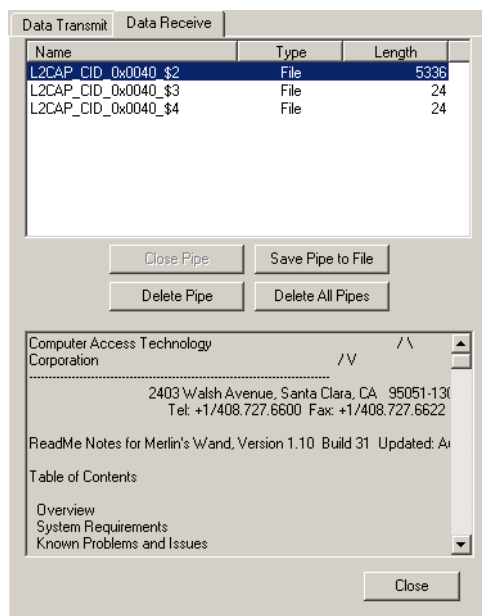
Receive pipes on the Data Receive page of Data Transfer Manager can be closed. Closing a receive pipe prevents additional data from being placed in it. Closing it also allows its contents to be viewed in the bottom window of the Data Receive page. For any pipe that isn't closed, this message will appear in the window: "This pipe is open for writing and cannot be viewed."



To close a receive pipe:

- Step 1 **Select the pipe** that is to be closed.
- Step 2 **Click Close Pipe.**

The pipe will be closed, and its contents will be shown in the bottom window.



7.5 Saving Data Pipes

Data Transfer Manager can save data pipes that are prepared for transmission as well as data pipes that are received.

Saving Data Transmit Pipe Lists

- Step 1** (Optional) **Delete** all pipes. If pipes are already displayed on the Data Transmit page of Data Transfer Manager, any newly created pipes will be added to the displayed list. To create an entirely new list of pipes, the currently displayed list should be deleted.

Note: The default pipe list is automatically loaded into the Data Transmit page when the Merlin's Wand application is opened. If no list has been saved as default, then no list will be loaded.

If the application hasn't been shut down since the last time that Data Transfer Manager was used, then the last list that was open in the Data Transmit page will be displayed the next time the tool is opened.

- Step 2** **Create** one or more data pipes.
- Step 3** Click **Save Pipe List...** to bring up the Save As dialog. Enter a file name and save the list as a Merlin's Wand Pipe File [*.pipe].

Saving a Default List

- Step 1** **Create** one or more data pipes **or open** a pipe list.
- Step 2** Click **Save List As Default**.

The list will be saved as `default.pipe`. That list is loaded into the Data Transmit page when the Merlin's Wand application is opened. However, if the user exits Data Transfer Manager but doesn't exit the application, the last list that was open in the Data Transmit page will be the one displayed when the tool is next accessed.

Saving Data Receive Pipes

- Step 1** **Select a pipe** in the Data Receive page of Data Transfer manager.
- Step 2** Click **Save Pipe to File**.

The Save As dialog will come up. Enter a file name, including the file type extension, then click Save.

7.6 Deleting Pipes

Note: Deleting pipes removes them from the list displayed in Data Transfer Manager. If the pipes were previously saved in a pipe list file, deleting them in Data Transfer Manager won't delete them from the file. To delete pipes from a pipe list file, first delete the pipes, then save the pipe list.

To delete a pipe:

Step 1 Select the pipe to be deleted.

Step 2 Click **Delete Pipe**.

The pipe will be deleted from the displayed pipe list.

To delete all pipes:

Step 1 Click **Delete All Pipes**.

A warning dialog will come up, asking, "Are you sure you want to delete all pipes from the list?" Click **Yes** to delete the pipes.

All pipes will be cleared from the display.

7.7 Opening Pipe Lists

To open a pipe list in the Data Transmit page of Data Transfer Manager:

Step 1 Click **Open Pipe List...**

Note: If pipes are already displayed, a dialog box will ask, "Delete current pipes before adding new pipes from pipe list file?" Choose **Yes** to delete the displayed pipes, or click **No** to leave those pipes displayed.

The Open dialog will come up.

Step 2 Select a file, then click **Open**.

The selected list will be displayed on the Data Transmit page.

7.8 Renaming Pipes

To rename a pipe displayed in the Data Transmit page of Data Transfer Manager:

Step 1 Select the pipe to be renamed.

Step 2 Click **Rename Pipe...**

The Rename Pipe dialog will appear.

Step 3 Enter a new name for the pipe, then click **OK**.

Note: Renaming a pipe changes its name in Data Transfer Manager. To change a pipe's name in a pipe list file, first rename the pipe, then save the pipe list.


7.9 Modifying Pipes

Existing pipes may be modified on the Data Transmit page of Data Transfer Manager. A pipe created from a file can be modified either by associating it with a different file or by changing it to a text-based pipe. A pipe created from text can be modified either by editing the text or by changing the pipe to a file-based pipe.

Note: Modifying pipes changes them in Data Transfer Manager. However, if the pipes were previously saved in a pipe list file, modifying them in Data Transfer Manager won't change them in the saved file. To modify pipes in a pipe list file, first modify the pipes, then save the pipe list.

Step 1 Select the **pipe** that is to be modified.

Step 2 *To associate a pipe with a different file or change a text-based pipe to a file-based pipe:*

- With the “From file” radio button selected, enter a new filename and path or navigate to the new file by clicking the browse button  to bring up the Open dialog. **Select a new file** and click Open.

To modify text or change a file-based pipe to a text-based pipe:

- With the “From text” radio button selected, **change, add or delete text** in the text entry box.

Step 3 Click **Modify Pipe** to implement the changes.

8. Using Merlin to Record Merlin's Wand Traffic

It's possible to control the CATC Merlin Bluetooth Protocol Analyzer via Merlin's Wand. The two can be used together to capture real-time test sequence results, as is required by the Bluetooth SIG to provide evidence of product compliance to the specification.

Merlin's Wand has built-in functionality for controlling the Merlin protocol analyzer. Through Merlin's Wand, a Bluetooth recording session can be set up on Merlin, even if the Merlin application runs on a remote computer.

8.1 Set Up a Remote Machine

If Merlin's Wand will be used to run Merlin on a remote machine, DCOM and accessibility properties on the remote machine must be properly configured. The configuration procedures differ slightly for different operating systems.

Note: If Merlin's Wand will be used only to run Merlin on the same computer that is running Merlin's Wand, skip to "Connect to Merlin with Merlin's Wand" on page 81.

Windows 98/Me Operating Systems

Use this procedure to configure DCOM properties to run a Merlin analyzer remotely on a machine running Windows 98 or Windows Me. All the steps should be performed on the remote machine.

Step 1 **Open the Merlin application** on the remote machine in order to register it with COM, and then close the application.

Step 2 **Download and install** `dcom98.exe` and `dcm95cfg.exe` DCOM configuration utilities from Microsoft. They are available via the following links:

<http://download.microsoft.com/msdownload/dcom/98/x86/en/dcom98.exe>

<http://download.microsoft.com/msdownload/dcom/98/x86/en/dcm95cfg.exe>

Step 3 Select **Start > Run** from the Windows taskbar.

The Run dialog will open.

Step 4 Enter **"dcomcnfg"** in the Open combo box and press **OK**.

The Distributed COM Configuration Properties dialog will open.

- Step 5 On the Applications tab, **select Merlin** from the list of applications.
- Step 6 Select the **Default Properties tab** and make sure that "Enable Distributed COM on this computer" is checked.
- Step 7 Select the **Default Security tab** and make sure that "Enable remote connection" is checked.
- Step 8 **Click OK.**

Windows NT®/2000 Operating Systems

Use this procedure to configure DCOM and accessibility properties to run a Merlin analyzer remotely on a machine running Windows NT 4.0 or Windows 2000. All the steps should be performed on the remote machine.

- Step 1 **Open the Merlin application** on the remote machine in order to register it with COM, and then close the application.

- Step 2 Select **Start > Run** from the Windows taskbar.

The Run dialog will open.

- Step 3 **Enter "dcomcnfg"** in the Open combo box and **press OK.**

The Distributed COM Configuration Properties dialog will open.

- Step 4 On the Applications tab, **select Merlin** from the list of applications.

- Step 5 Click the **Properties** button.

The Merlin Properties dialog will open.

- Step 6 Go to the **Security tab.**

- Step 7 Select the **"Use custom launch permissions"** radio button and click the associated **Edit...** button.

The Registry Value Permissions dialog will open.

- Step 8 Make sure **Allow Launch** is selected in the Type of Access drop-down list.

- Step 9 Click **Add...**

The Add Users and Groups dialog will open.

- Step 10 **Select names** from the Names list and click **Add** to include them in the Add Names list.

- Step 11 When all desired user names have been added, **click OK.**

- Step 12 **Click OK** on each of the three dialogs that are still open.


Step 13 *For Windows NT:*

- (a) Select **Start > Settings > Control Panel** on the Windows taskbar.
The Control Panel window will open.
- (b) Double-click on **Services**.
The Services dialog will open.
- (c) Select **Remote Procedure Call (RPC) Locator** and select **Action > Properties** from the menu bar.
The Remote Procedure Call (RPC) Locator Properties dialog will open.
- (d) Click **Start** on the General tab, then click **OK**.
The status for Remote Procedure Call (RPC) Locator should now be set to "Started" in the Services dialog.
- (e) Select **Remote Procedure Call (RPC) Service** and select **Action > Properties** from the menu bar.
The Remote Procedure Call (RPC) Service Properties dialog will open.
- (f) Click **Start** on the General tab, then click **OK**.
The status for Remote Procedure Call (RPC) Service should now be set to "Started" in the Services dialog.

For Windows 2000:

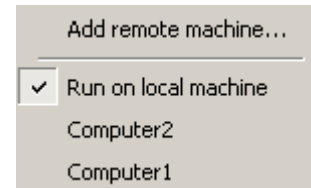
- (a) Select **Start > Settings > Control Panel** on the Windows taskbar.
The Control Panel window will open.
- (b) Double-click on **Administrative Tools**.
The Administrative Tools window will open.
- (c) Double-click on **Services**.
The Services dialog will open.
- (d) Select **Remote Procedure Call (RPC)** and select **Action > Properties** from the menu bar.
The Remote Procedure Call (RPC) Properties dialog will open.
- (e) Click **Start** on the General tab, then click **OK**.
The status for Remote Procedure Call (RPC) should now be set to "Started" in the Services dialog.
- (f) Select **Remote Procedure Call (RPC) Locator** and select **Action > Properties** from the menu bar.
The Remote Procedure Call (RPC) Locator Properties dialog will open.
- (g) Click **Start** on the General tab, then click **OK**.
The status for Remote Procedure Call (RPC) Locator should now be set to "Started" in the Services dialog.

8.2 Set Up Connection Options

In addition to establishing connections, the Connect/Disconnect Merlin Bluetooth Analyzer button  provides several options for configuring the connection between Merlin's Wand and Merlin. To see the options, click on the options arrow on the right side of the button.

Run Merlin on a Local Machine

To run Merlin on a local machine, click the options arrow on the right side of the Connect/Disconnect Merlin button and make sure that “Run on local” is selected in the Connect/Disconnect options menu. When selected, a checkmark appears next to it.

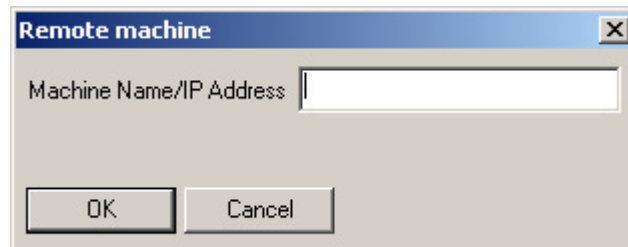


Add a Remote Machine

Merlin's Wand can be configured to control a Merlin analyzer that is running on a remote computer. Before Merlin's Wand can connect to Merlin, the remote machine that runs Merlin must be added to the Connect/Disconnect options menu in Merlin's Wand.

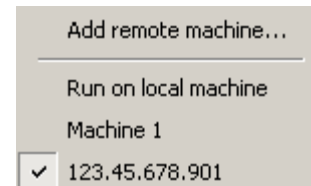
- Step 1** Click the options arrow on the right side of the Connect/Disconnect Merlin button and select “**Add remote machine...**” from the options menu.

The Remote Machine dialog will open.



- Step 2** Enter the **Internet machine name or IP address** for the machine on which Merlin is running and click OK.

Note: If the machine name is used and Merlin's Wand is subsequently unable to connect to Merlin, then the IP address must be used instead.



The machine name/IP address will now be listed on the Connect/Disconnect options menu. By default, it will be selected, as indicated by the checkmark that appears to the left of the name/IP address.


8.3 Start Merlin

This step is required only when running Merlin on a remote machine that uses Windows 98 or Windows Me. In such cases, Merlin's Wand cannot start or stop the Merlin application, although it can control Merlin once it is running. **Be sure to start Merlin before connecting to it on a machine running Windows 98 or Windows Me.**

8.4 Connect to Merlin with Merlin's Wand

Note: Before Merlin's Wand can connect to Merlin, the connection options, DCOM, and accessibility properties may need to be configured. Please refer to "Set Up Connection Options" on page 80 and "Set Up a Remote Machine" on page 77 for more information.

Step 1 To connect to Merlin Bluetooth Protocol Analyzer, click on the **Connect/Disconnect**  button.

Merlin's Wand will connect to Merlin. The status bar at the bottom of the Merlin's Wand application will indicate that Merlin is connected, along with the Merlin software version. The Connect/Disconnect button will remain "pressed down"  while Merlin's Wand and Merlin are connected.

8.5 Set Merlin Recording Options

The recording options file [* .rec] that Merlin should use can be specified through Merlin's Wand. If a recording options file isn't specified through Merlin's Wand, Merlin will use either its default .rec file or the options file that was last loaded into the current instance of Merlin.

Note: The .rec file has to be configured and saved in Merlin before it can be specified through Merlin's Wand.

Step 1 **Share the folder** that contains the file.

(a) In Windows Explorer or My Computer, navigate to the folder that contains the options file.

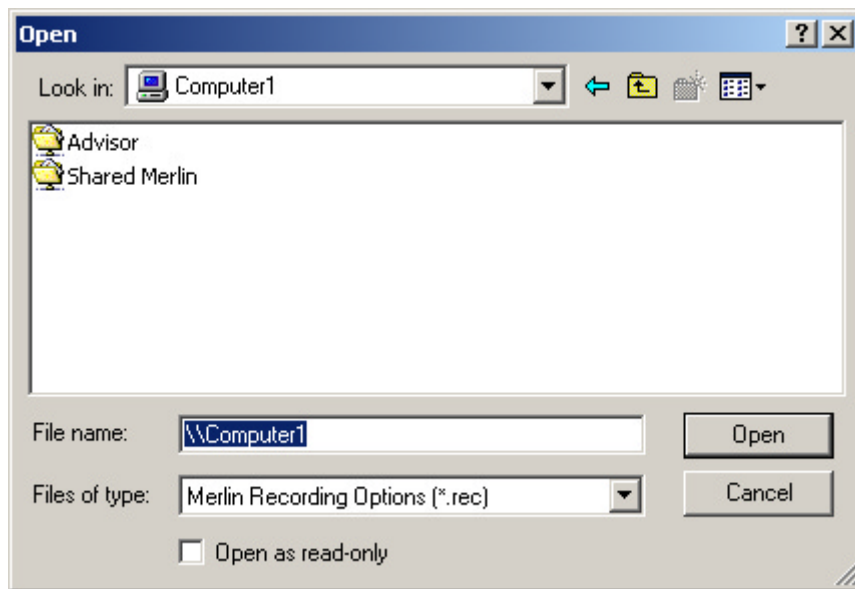
- (b) Right-click on the folder and select Properties, or select File > Properties from the menu bar.

The Properties dialog will open.

- (c) Go to the Sharing tab in the Properties dialog.
- (d) Enable the "Share this folder" option and make sure that the folder is accessible by both the machine running Merlin and the machine running Merlin's Wand.
- (e) Click OK.

Step 2 Click the **Set Recording Options**  button.

The Open dialog will be displayed.



Step 3 Use the **Look in** field at the top of the dialog box to browse to the desired file via Network Neighborhood
-or-

In the **File name** field, type \\ followed by the name of the computer on which the file is located (for example, \\Computer1). Press Enter to display all shared folders, then navigate to the desired file.

Note: A full network path must be used in order to specify the options file through Merlin's Wand, whether the file is local to the machine running Merlin or located on a different computer.

Step 4 Click **Open**.

The path and filename of the recording options file will now be listed on the Set Recording Options drop-down menu. By default, that file will be selected, as indicated by the checkmark that appears to the left of the path and filename.

8.6 Set Merlin Display Options

The display options file [`*.opt`] that Merlin should use can be specified through Merlin's Wand. If a display options file isn't specified through Merlin's Wand, Merlin will use either its default `.opt` file or the options file that was last loaded into the current instance of Merlin.

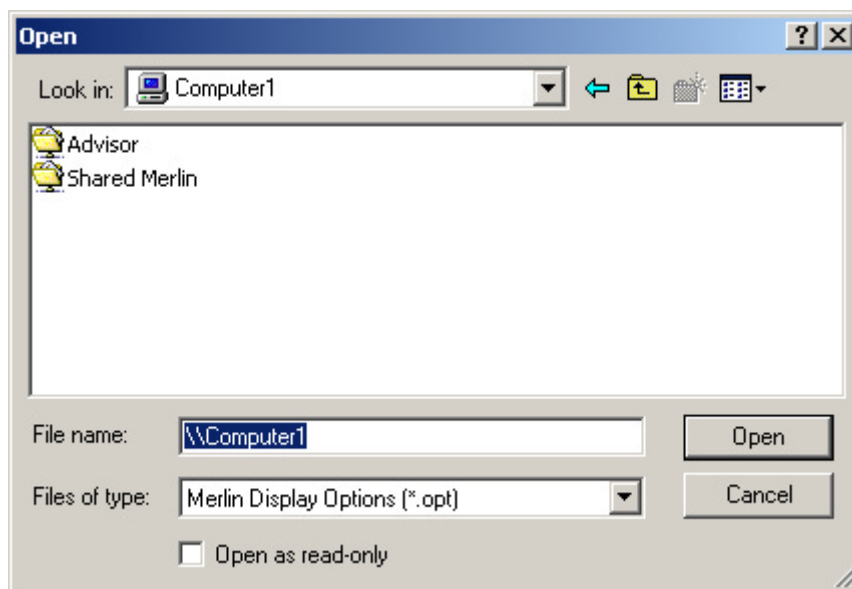
Note: The `.opt` file has to be configured and saved in Merlin before it can be specified through Merlin's Wand.

Step 1 **Share the folder** that contains the file.

- (a) In Windows Explorer or My Computer, navigate to the folder that contains the options file.
- (b) Right-click on the folder and select Properties, or select File > Properties from the menu bar.
The Properties dialog will open.
- (c) Go to the Sharing tab in the Properties dialog.
- (d) Enable the "Share this folder" option and make sure that the folder is accessible by both the machine running Merlin and the machine running Merlin's Wand.
- (e) Click OK.

Step 2 Click the **Set Display Options**  button.

The Open dialog will be displayed.



Step 3 Use the **Look in** field at the top of the dialog box to browse to the desired file via Network Neighborhood
-OR-

In the **File name** field, type \\ followed by the name of the computer on which the file is located (for example, \\Computer1). Press Enter to display all shared folders, then navigate to the desired file.


Note: A full network path must be used in order to specify the options file through Merlin's Wand, whether the file is local to the machine running Merlin or located on a different computer.

Step 4 Click **Open**.

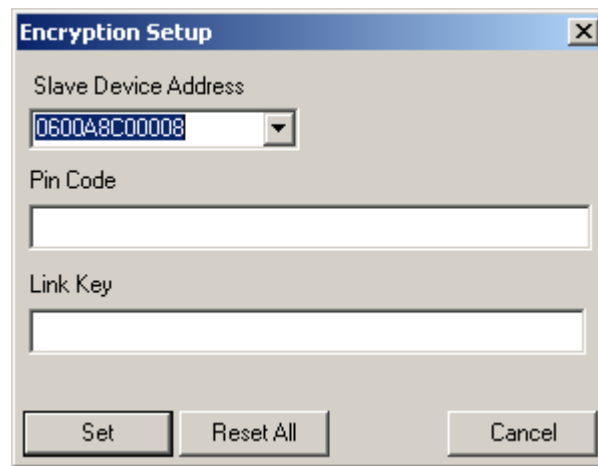
The path and filename of the display options file will now be listed on the Set Display Options drop-down menu. By default, that file will be selected, as indicated by the checkmark that appears to the left of the path and filename.

8.7 Set Merlin Encryption Options

Merlin's Wand can set up Merlin to decode encrypted transmissions.

Step 1 Open the Encryption Setup dialog by pressing the **Set Merlin encryption options**  button.

The Encryption Setup dialog will open.



Step 2 Select the **Slave Device Address** from the drop-down list, or enter it into the combo box.


Step 3 Enter the **PIN Number** for the slave device in the PIN Code text box.

or


Enter the **Link Key** for the master-slave connection in the Link Key text box.

Step 4 Press the **Set** button to apply the encryption setup.


8.8 Start a Merlin Recording Session

To begin a Merlin Bluetooth Analyzer recording session, press the Record  button on the Merlin toolbar.

8.9 Stop a Merlin Recording Session

To stop a Merlin Bluetooth Analyzer recording session, press the Stop  button on the Merlin toolbar.

8.10 Disconnect from Merlin Bluetooth Protocol Analyzer


To disconnect from Merlin Bluetooth Protocol Analyzer, click on the Connect/Disconnect  button.

Merlin's Wand will disconnect from Merlin.

8.11 Troubleshooting

“Server Busy” When Attempting to Launch Merlin

“Server Busy” message appears when attempting to launch Merlin on a remote Windows 98 or Windows Me system.

- Make sure Merlin is running on the remote machine before clicking the Connect/Disconnect Merlin  button in Merlin's Wand. Merlin's Wand cannot start or stop Merlin on a remote Windows 98 or Windows Me system; it can only control Merlin once Merlin is running on the remote machine.
- Make sure DCOM is properly installed and configured on the remote machine, as described in the section DCOM Configuration for Windows 98/Me Systems.
- If the message box won't go away, close Merlin's Wand (press Ctrl+Alt+Delete and close the program directly or through the Task Manager). Then, restart Merlin's Wand and be sure to start Merlin on the remote machine before retrying the operation.

“Server Busy” message appears when attempting to launch Merlin on a remote Windows NT or Windows 2000 system.

- This message may appear the first time Merlin is launched on the remote machine. It can be safely ignored. Merlin will start normally on the remote machine. Clicking the “Switch To” button in the message box will cause the message to disappear and the Start menu to appear. Return to Merlin's Wand and proceed normally.

“Server Execution Failed” When Attempting to Launch Merlin

“Server execution failed” message appears when attempting to launch Merlin on a remote Windows 98 or Windows Me system.

- Make sure Merlin is running on the remote machine before clicking the Connect/Disconnect Merlin button in Merlin's Wand.

“The Object Exporter Specified Was Not Found” When Attempting to Launch Merlin

“The object exporter specified was not found” message appears when attempting to launch Merlin from a Windows 2000 system.

- If Merlin is already running on the remote machine, try to close it. If a message appears indicating that an automation client is connected to the application, close Merlin on the remote machine, close Merlin's Wand on

the local machine, and try again. If the problem persists, restart the remote machine.

- Make sure the local computer (the Windows 2000 system running Merlin's Wand) can reach the remote machine by using its full computer name, such as <computername.domain>. Open a command prompt and use the PING command to determine this. If the local computer cannot communicate with the remote machine, consult a system administrator for assistance.
- If the remote machine is running TCP/IP, make sure it has an assigned IP address. Consult a system administrator for assistance in determining this and, if necessary, assigning an IP address.

9. Contact and Warranty Information

9.1 Contact Information

Mailing address

Computer Access Technology Corporation
Customer Support
2403 Walsh Avenue
Santa Clara, CA 95051-1302
USA

Online support

<http://www.catc.com/>

E-mail address

support@catc.com

Telephone support

+1/800.909.2282 (USA and Canada)
+1/408.727.6600 (worldwide)

Fax

+1/408.727.6622 (worldwide)

Sales information

sales@catc.com

9.2 Warranty and License

Computer Access Technology Corporation (hereafter CATC) warrants this product to be free from defects in material, content, and workmanship, and agrees to repair or replace any part of the enclosed unit that proves defective under these terms and conditions. Parts and labor are warranted for one year from the date of first purchase.



The CATC software is licensed for use on a single personal computer. The software may be copied for backup purposes only.

This warranty covers all defects in material or workmanship. It does not cover accidents, misuse, neglect, unauthorized product modification, or acts of nature. Except as expressly provided above, CATC makes no warranties or conditions, express, implied, or statutory, including without limitation the implied warranties of merchantability and fitness for a particular purpose.

CATC shall not be liable for damage to other property caused by any defects in this product, damages based upon inconvenience, loss of use of the product, loss of time or data, commercial loss, or any other damages, whether special, incidental, consequential, or otherwise, whether under theory of contract, tort (including negligence), indemnity, product liability, or otherwise. In no event shall CATC's liability exceed the total amount paid to CATC for this product.

CATC reserves the right to revise these specifications without notice or penalty.

Appendix A: Command Generator Command Descriptions

A.1 HCI Command Descriptions

Note "N/A" means Not Applicable. This indicates that the specified command does not have a parameter.

HCI Link Control Commands

Accept_Connection_Request

Used to accept a new incoming connection request. Execute this command before connection request from another device. By default, all connection requests are accepted.

Command Parameters	Examples	Comments
N/A		

Return Events
Accept_Connection_Request_Complete

Add_SCO_Connection

Will cause the link manager to create an SCO connection in addition to the existing ACL connection.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Packet Type	0x0080	Possible packet types: HV1=0x0020 HV2=0x0040 HV3=0x0080

Return Events
Add_SCO_Connection_Complete
Add_SCO_Connection_Error

Authentication_Requested

Used to initiate authentication between the two devices associated with the specified HCI_Handle.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Authentication_Error
Authentication_Complete

Change_Connection_Link_Key

Used to force both connected devices to generate a new link key.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	Range: 0x0000-0x0EFF

Return Events
Change_Connection_Link_Key_Error
Change_Connection_Link_Key_Complete

Change_Connection_Packet_Type

Used to change which packet types can be used for a connection that is currently established.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Packet_Type	0x0008	Range: 0x0000-0x0EFF 0x0008 = DM1 0x0010 = DH1 0x0400 = DM3 0x0800 = DH3 0x4000 = DM5 0x8000 = DH5

Return Events
Change_Connection_Packet_Type_Error
Change_Connection_Packet_Type_Complete

Create_Connection

Create_Connection will cause the link manager to create an ACL connection to the Bluetooth wireless device with the BD_ADDR specified by the command parameters.

Command Parameters	Examples	Comments
BD_ADDR	010203040506	Enter in HEX as shown.

Return Events
Create_Connection_Complete
Create_Connection_Error

Disconnect

Disconnect is used to terminate an existing connection.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Disconnection_Complete
Disconnection_Failed

Exit_Periodic_Inquiry_Mode

Exit_Periodic_Inquiry_Mode is used to end the Periodic Inquiry mode when Merlin's Wand is in Periodic Inquiry mode.

Command Parameters	Examples	Comments
N/A		

Return Events
Exit_Periodic_Inquiry_Mode_Complete
Exit_Periodic_Inquiry_Mode_Error

Inquiry

Inquiry will cause Merlin's Wand to enter Inquiry mode and discover other nearby Bluetooth devices.

Command Parameters	Examples	Comments
Inquiry_Length	8	
Num_Responses	10	

Return Events
Inquiry_Complete
Inquiry_Result
Inquiry_Error

Inquiry_Cancel

Inquiry_Cancel will cause Merlin's Wand to stop the current Inquiry if the Bluetooth device is in Inquiry mode.

Command Parameters	Examples	Comments
N/A		

Return Events
Inquiry_Canceled
Inquiry_Error

Periodic_Inquiry_Mode

Periodic_Inquiry_Mode is used to configure Merlin's Wand to perform a periodic Inquiry based on a specified period range.

Note: Max_Period_Length > Min_Period_Length > Inquiry Length.

Command Parameters	Examples	Comments
Max Period Length	10	Range: 0x03 – 0xFFFF
Min Period Length	9	Range: 0x02 – 0xFFFE
Inquiry Length	8	Range: 0x01 – 0x30
Num of Responses	10	Range: 0-255 (0=Unlimited number of responses)

Return Events
Periodic_Inquiry_Mode_Complete
Periodic_Inquiry_Mode_Error

PIN_Code_Request_Negative_Reply

PIN_Code_Request_Negative_Reply is used to reply to a PIN Code Request event from the Host Controller when the Host cannot specify a PIN code to use for a connection. This command should be executed before PIN_Code_Request event is received. By default, the PIN_Code_Request event is rejected.

Command Parameters	Examples	Comments
BD_ADDR	0x010203040506	

Return Events
Command_Complete

PIN_Code_Request_Reply

PIN_Code_Request_Reply is used to reply to a PIN Code Request event from the Host Controller and specifies the PIN code to use for a connection. This command should be executed before Pin_Code Request event is received. By default, the Pin_Code Request event is rejected.

Command Parameters	Examples	Comments
PIN_Code	1234	PIN_Code is a string character that can be up to 128 bits in length
BD_ADDR	0x010203040506	

Return Events
Command_Complete

Read_Clock_Offset

Read_Clock_Offset allows the Host to read the clock offset of remote devices.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Read_Clock_Offset_Complete
Read_Clock_Offset_Error

Read_Remote_Supported_Features

Read_Remote_Supported_Features requests a list of the supported features of a remote device.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Read_Remote_Supported_Features_Complete
Read_Remote_Supported_Features_Error

Read_Remote_Version_Information

Read_Remote_Version_Information command will read the values for the version information for the remote Bluetooth device.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Read_Remote_Version_Information_Complete
Read_Remote_Version_Information_Error

Reject_Connection_Request

Reject_Connection_Request is used to decline a new incoming connection request. Execute this command before connection request from another device. By default, all connection requests are accepted.

Command Parameters	Examples	Comments
N/A		

Return Events
Reject_Connection_Request_Complete

Remote_Name_Request

Remote_Name_Request is used to obtain the user-friendly name of another Bluetooth device.

The BD_ADDR command parameter is used to identify the device for which the user-friendly name is to be obtained. The Page_Scan_Repetition_Mode and Page_Scan_Mode command parameters specify the page scan modes supported by the remote device with the BD_ADDR. This is the information that was acquired during the inquiry

process. The Clock_Offset parameter is the difference between its own clock and the clock of the remote device with BD_ADDR. Only bits 2 through 16 of the difference are used and they are mapped to this parameter as bits 0 through 14 respectively. A Clock_Offset_Valid_Flag, located in bit 15 of the Clock_Offset command parameter, is used to indicate if the Clock_Offset is valid or not.

Command Parameters	Examples	Comments
BD_ADDR	0x010203040506	
Page Scan Rep Mode	0x0	0x00=R0; 0x01=R1; 0x02=R2
Page Scan Mode	0x0	0x00=Mandatory Page Scan Mode 0x01=Optional Page Scan Mode I 0x02=Optional Page Scan Mode II 0x03=Optional Page Scan Mode III
Clock Offset	0x0	Bit Format: Bit 14.0 = Bit 16.2 of CLKslave - CLKmaster. Bit 15 = Clock_Offset_ Valid_Flag where 0= Invalid Clock Offset 1=Valid Clock Offset

Return Events
Remote_Name_Request_Complete
Remote_Name_Request_Error

Set_Connection_Encryption

Set_Connection_Encryption is used to enable and disable the link-level encryption.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Encryption_Enable	1	Range: 0 or 1

Return Events
Set_Connection_Encryption_Complete
Set_Connection_Encryption_Error

HCI Link Policy Commands

Exit_Park_Mode

Stops park mode and enters active mode for the specified ACL link.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Mode_Change
Exit_Park_Mode_Error

Exit_Sniff_Mode

Stops Sniff mode and enters active mode for the specified ACL link.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Mode_Change
Exit_Sniff_Mode_Error

Hold_Mode

Places the specified ACL link into Hold Mode.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Max_Interval	0xFFFF	0x0001 - 0xFFFF
Min_Interval	0x01	0x0001 - 0xFFFF

Return Events
Mode_Change
Hold_Mode_Error

Park_Mode

Places the specified ACL link into Park mode.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Beacon_Max_Interval	0xFFFF	0x0001 - 0xFFFF
Beacon_Min_Interval	0x01	0x0001 - 0xFFFF

Return Events
Mode_Change
Park_Mode_Error

QoS_Setup

Used to specify Quality of Service parameters for a connection handle.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
ServiceType	0x01	0=No traffic; 1=Best effort; 2=Guaranteed
TokenRate	0	Token rate in Bytes/second
PeakBandwidth	0	Bytes per second
Latency	0xFFFFFFFF	In microseconds
DelayVariation	0xFFFFFFFF	In microseconds

Return Events
Quality_of_Service_Setup_Complete
Quality_of_Service_Setup_Error

Read_Link_Policy_Settings

Reads Link Policy setting for the specified ACL link.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Read_Link_Policy_Settings_Complete
Read_Link_Policy_Settings_Error

Role_Discovery

Description:

Role_Discovery is used for a Bluetooth device to determine which role the device is performing (Master or Slave) for a particular connection.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Role_Discovery_Complete
Role_Discovery_Error

Sniff_Mode

Places the specified ACL link into Sniff mode.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Max_Interval	0xFFFF	0x0001 - 0xFFFF
Min_Interval	0x01	0x0001 - 0xFFFF
Attempt	0x3FF6	0x0001 - 0x7FFF
Timeout	0x7FFF	0x0000 - 0x7FFF

Return Events
Mode_Change
Sniff_Mode_Error

Switch_Role

Switches the current role (master/slave) of the calling device with the role of the device specified.

Command Parameters	Examples	Comments
BD_ADDR	0x010203040506	

Return Events
Role_Change_Complete
Role_Change_Error

Write_Link_Policy_Settings

Writes link policy settings for the specified ACL link.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Link_Policy_Settings	0xF	0x0000: Disable all LM modes 0x0001: Enable master/slave switch 0x0002: Enable Hold Mode 0x0004: Enable Sniff Mode 0x0008: Enable Park Mode 0xF: Enable all (Default)

Return Events
Write_Link_Policy_Settings_Complete
Write_Link_Policy_Settings_Error

HCI Host Controller & Baseband Commands

Change_Local_Name

Change_Local_Name allows the user-friendly name to be modified for the Merlin's Wand.

Command Parameters	Examples	Comments
Name	Merlin's Wand	Maximum string length =32 characters

Return Events
Change_Local_Name_Complete
Change_Local_Name_Error

Delete_Stored_Link_Key

Delete_Stored_Link_Key removes one or all link keys stored in the Merlin's Wand.

Command Parameters	Examples	Comments
BD_ADDR	0x010203040506	
Delete_All_Flag	01	00=Delete only the Link Key for specified BD_ADDR 01=Delete all stored Link Keys

Return Events
Delete_Stored_Link_Key_Complete
Delete_Stored_Link_Key_Error

Host_Buffer_Size

Used by the Merlin's Wand to notify the Merlin's Wand Host Controller about its buffer sizes for ACL and SCO data. The Merlin's Wand Host Controller will segment the data to be transmitted from the Host Controller to Merlin's Wand, so that data contained in HCI Data Packets will not exceed these sizes.

Command Parameters	Examples	Comments
ACL_Data_Length	0x2A0	Only ACL is valid
SCO_Data_Length	0xFF	The value of the Host_SCO_Data_Packet_Length must be > 399
Total_Num_ACL	10	
Total_Num_SCO	0xFF	

Return Events
Host_Buffer_Size_Complete
Host_Buffer_Size_Error

Read_Authentication_Enable

Read_Authentication_Enable will read the value for the Authentication_Enable parameter, which controls whether Merlin's Wand will require authentication for each connection with other Bluetooth devices.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Authentication_Enable_Complete
Read_Authentication_Enable_Error

Read_Class_of_Device

Read_Class_of_Device will read the value for the Class_of_Device parameter for Merlin's Wand, which is used to indicate its capabilities to other devices.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Class_of_Device_Complete
Read_Class_of_Device_Error

Read_Connection_Accept_Timeout

Read_Connection_Accept_Timeout will read the value for the Connection_Accept_Timeout parameter so that Merlin's Wand can automatically deny a connection request after a specified period has occurred, and to refuse a new connection.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Connection_Accept_Timeout_Complete
Read_Connection_Accept_Timeout_Error

Read_Current_IAC_LAP

Read_Current_IAC_LAP will read the LAP(s) used to create the Inquiry Access Codes (IAC) that Merlin's Wand is simultaneously scanning for during Inquiry Scans.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Current_IAC_LAP_Complete
Read_Current_IAC_LAP_Error

Read_Encryption_Mode

Read_Encryption_Mode will read the value for the Encryption_Mode parameter, which controls whether Merlin's Wand will require encryption for each connection with other Bluetooth devices.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Encryption_Mode_Complete
Read_Encryption_Mode_Error

Read_Link_Supervision_Timeout

Reads link supervision timeout setting for the specified ACL link.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Read_Link_Supervision_Timeout_Complete
Read_Link_Supervision_Timeout_Error

Read_Local_Name

Read_Local_Name reads the stored user-friendly name for Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Local_Name_Complete
Read_Local_Name_Error

Read_Number_Of_Supported_IAC

This command will read the value for the number of Inquiry Access Codes (IAC) that Merlin's Wand can simultaneously listen for during an Inquiry Scan.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Number_Of_Supported_IAC_Complete
Read_Number_Of_Supported_IAC_Error

Read_Page_Scan_Mode

Read_Page_Scan_Mode command is used to read the current Page_Scan_Mode of Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Page_Scan_Mode_Complete
Read_Page_Scan_Mode_Error

Read_Page_Scan_Period_Mode

Read_Page_Scan_Period_Mode is used to read the Page_Scan_Period_Mode of Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Page_Scan_Period_Mode_Complete
Read_Page_Scan_Period_Mode_Error

Read_Page_Timeout

Read_Page_Timeout will read the value for the Page_Reply_Timeout configuration parameter, which allows Merlin's Wand to define the amount of time a connection request will wait for the remote device to respond before the local device returns a connection failure.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Page_Timeout_Complete
Read_Page_Timeout_Error

Read_PIN_Type

Read_PIN_Type will read the PIN type specified in the host controller of Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_PIN_Type_Complete
Read_PIN_Type_Error

Read_Scan_Enable

Read_Scan_Enable will read the value for the Scan_Enable configuration parameter, which controls whether or not Merlin's Wand will periodically scan for page attempts and/or inquiry requests from other Bluetooth devices.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Scan_Enable_Complete
Read_Scan_Enable_Error

Read_SCO_Flow_Control_Enable

The Read_SCO_Flow_Control_Enable command provides the ability to read the SCO_Flow_Control_Enable setting. By using this setting, Merlin can decide if the Merlin's Wand Host Controller will send Number Of Completed Packets events for SCO HCI_Handles.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_SCO_Flow_Control_Enable_Complete
Read_SCO_Flow_Control_Enable_Error

Read_Stored_Link_Key

Read_Stored_Link_Key will read one or all link keys stored in the Merlin's Wand Host Controller.

Command Parameters	Examples	Comments
BD_ADDR	0x010203040506	
Read_All_Flag	01	00=Return Link Key for specified BD_ADDR 01=Return all stored Link Keys

Return Events
Read_Stored_Link_Key_Complete
Read_Stored_Link_Key_Error

Read_Voice_Setting

Read_Voice_Setting will read the values for the Voice_Setting parameter in Merlin's Wand, which controls all the various settings for the voice connections.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Voice_Setting_Complete
Read_Voice_Setting_Error

Reset

Resets the Bluetooth Host Controller, Link Manager, and the radio module of Merlin's Wand. After executing this command, the application has to be restarted.

Command Parameters	Examples	Comments
N/A		Will reset to default values for the parameters

Return Events
Reset_Complete

Set_Event_Filter

Set_Event_Filter is used by the Host to specify different event filters. The Host may issue this command multiple times to request various conditions for the same type of event filter and for different types of event filters.

Command Parameters	Examples	Comments
FilterType	0x00	0x00=Clear All Filters 0x01=Inquiry Result 0x02=Connection Setup
FilterConditionType	0x00	0x00=New device responded to the Inquiry process 0x01= Device with a specific Class of Device responded to the Inquiry process. 0x02=Device with specific BD_ADDR responded to the Inquiry process.

Command Parameters	Examples	Comments
Condition	0x01	0x00=Allow Connections from all devices 0x01=Allow Connections from a device with a specific Class of Device 0x02=Allow Connections from a device with a specific BD_ADDR

Return Events
Set_Event_Filter_Complete
Set_Event_Filter_Error

Set_Event_Mask

Set_Event_Mask is used to control which events are generated by the HCI for the Host.

Command Parameters	Examples	Comments
Event_Mask		NO_EVENTS 0x0000 INQUIRY_RESULT 0x0001 INQUIRY_COMPLETE 0x0002 INQUIRY_CANCELED 0x0004 LINK_CONNECT_IND 0x0008 SCO_CONNECT_IND 0x0010 LINK_DISCONNECT 0x0020 LINK_CONNECT_CNF 0x0040 LINK_CON_RESTRICT 0x0080 MODE_CHANGE 0x0100 ACCESSIBLE_CHANGE 0x0200 AUTHENTICATED 0x0400 ENCRYPTION_CHANGE 0x0800 SECURITY_CHANGE 0x1000 ROLE_CHANGE 0x2000 SCO_DISCONNECT 0x4000 SCO_CONNECT_CNF 0x8000 ALL_EVENTS 0xffff

Return Events
Set_Event_Mask_Complete
Set_Event_Mask_Error

Write_Authentication_Enable

This command will write the value for the Authentication_Enable parameter, which controls whether Merlin's Wand will require authentication for each connection with other Bluetooth devices.

Command Parameters	Examples	Comments
Authentication_Enable	0x0	0x00=Authentication disabled. Default 0x01=Authentication enabled for all connection

Return Events
Write_Authentication_Enable_Complete
Write_Authentication_Enable_Error

Write_Class_of_Device

Write_Class_of_Device will write the value for the Class_of_Device parameter, which is used to indicate its capabilities to other devices.

Command Parameters	Examples	Comments
CoD	0x000000	Class of Device for the device

Return Events
Write_Class_of_Device_Complete
Write_Class_of_Device_Error

Write_Connection_Accept_Timeout

Write_Connection_Accept_Timeout will write the value for the Connection_Accept_Timeout configuration parameter, which allows Merlin's Wand to automatically deny a connection request after a specified period has occurred, and to refuse a new connection.

Command Parameters	Examples	Comments
Timeout	0x00	Connection Accept Timeout measured in Number of Baseband slots. Interval Length = N * 0.625 msec (1 Baseband slot) Range for N: 0x0001 – 0xB540 Time Range: 0.625 msec - 29 seconds Default: N = 0x1FA0 Time = 5 Sec

Return Events
Write_Connection_Accept_Timeout_Complete
Write_Connection_Accept_Timeout_Error

Write_Current_IAC_LAP

Will write the LAP(s) used to create the Inquiry Access Codes (IAC) that the local Bluetooth device is simultaneously scanning for during Inquiry Scans.

Command Parameters	Examples	Comments
IAC_LAP	0x9E8B33	0x9E8B00-0x9E8B3F
IAC_LAP		0x9E8B00-0x9E8B3F
IAC_LAP		0x9E8B00-0x9E8B3F
IAC_LAP		0x9E8B00-0x9E8B3F
IAC_LAP		0x9E8B00-0x9E8B3F
IAC_LAP		0x9E8B00-0x9E8B3F

Return Events
Write_Current_IAC_LAP_Complete
Write_Current_IAC_LAP_Error

Write_Encryption_Mode

Write_Encryption_Mode command will write the value for the Encryption_Mode parameter, which controls whether Merlin's Wand will require encryption for each connection with other Bluetooth devices.

Command Parameters	Examples	Comments
Encryption Mode	0x0	0x00=Encryption disabled. Default 0x01=Encryption only for point-to-point packets 0x02=Encryption for both point-to-point and broadcast packets

Return Events
Write_Encryption_Mode_Complete
Write_Encryption_Mode_Error

Write_Link_Supervision_Timeout

Writes link supervision timeout setting for the specified ACL link.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Timeout	0x7D00	0x0001 - 0xFFFF

Return Events
Write_Link_Supervision_Timeout_Complete
Write_Link_Supervision_Timeout_Error

Write_Page_Timeout

Write_Page_Timeout command will write the value for the Page_Reply_Timeout configuration parameter, which allows Merlin's Wand to define the amount of time a connection request will wait for the remote device to respond before the local device returns a connection failure.

Command Parameters	Examples	Comments
Timeout	0x10	0=Illegal Page Timeout. Must be larger than 0 N = 0xXXXX Page Timeout measured in Number of Baseband slots. Interval Length = N * 0.625 msec (1 Baseband slot) Range for N: 0x0001 – 0xFFFF Time Range: 0.625 msec -40.9 Seconds Default: N = 0x2000 Time = 5.12 Sec

Return Events
Write_Page_Timeout_Complete
Write_Page_Timeout_Error

Write_PIN_Type

Write_PIN_Type will specify whether the Host supports variable PIN or only fixed PINs.

Command Parameters	Examples	Comments
PIN_Type	0	0x00=Variable PIN 0x01=Fixed PIN

Return Events
Write_PIN_Type_Complete
Write_PIN_Type_Error

Write_Scan_Enable

The Write_Scan_Enable command will write the value for the Scan_Enable configuration parameter into Merlin's Wand, which controls whether or not Merlin's Wand will periodically scan for page attempts and/or inquiry requests from other Bluetooth devices.

Command Parameters	Examples	Comments
Scan_Enable	3	0x00=No Scans enabled. 0x01=Inquiry Scan enabled Page Scan disabled. 0x02=Inquiry Scan disabled. Page Scan enabled. 0x03=Inquiry Scan enabled Page Scan enabled. (Default)

Return Events
Write_Scan_Enable_Complete
Write_Scan_Enable_Error

Write_Stored_Link_Key

Write_Stored_Link_Key command will write a link key to the Merlin's Wand host controller.

Command Parameters	Examples	Comments
BD_ADDR	0x010203040506	
Link_Key	0x01020304	

Return Events
Write_Stored_Link_Key_Complete
Write_Stored_Link_Key_Error

Write_Voice_Setting

The Write_Voice_Setting command will write the values for the Voice_Setting parameter into Merlin's Wand, which controls all the various as settings for the voice connections.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Voice_Setting	0x0062	0x0060=CVSD coding 0x0061=u-Law coding 0x0062=A-law coding

Return Events
Write_Voice_Setting_Complete
Write_Voice_Setting_Error

HCI Informational Commands

Read_BD_ADDR

Read_BD_ADDR will read the value of Merlin's Wand's address. The BD_ADDR is a 48-bit unique identifier for a Bluetooth device.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_BD_ADDR_Complete
Read_BD_ADDR_Error

Read_Buffer_Size

Read_Buffer_Size returns the size of the HCI buffers in Merlin's Wand. These buffers are used by Merlin's Wand's Host Controller to buffer data that is to be transmitted.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Buffer_Size_Complete
Read_Buffer_Size_Error

Read_Country_Code

Read_Country_Code command will read the value for the Country_Code status parameter in Merlin's Wand. The Country_Code defines which range of frequency band of the ISM 2.4 GHz band will be used by the device.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Country_Code_Complete
Read_Country_Code_Error

Read_Local_Supported_Features

Read_Local_Supported_Features will request a list of the supported features for Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Local_Supported_Features_Complete
Read_Local_Supported_Features_Error

Read_Local_Version_Information

Read_Local_Version_Information command will read the values for the version information for Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Local_Version_Information_Complete
Read_Local_Version_Information_Error

HCI Testing Commands**Enable_Device_Under_Test_Mode**

The Enable_Device_Under_Test_Mode command will allow Merlin's Wand to enter test mode via LMP test commands. Merlin's Wand issues this command when it wants to become the DUT for the Testing scenarios as described in the Bluetooth Test Mode.

Command Parameters	Examples	Comments
N/A		

Return Events
Enable_Device_Under_Test_Mode_Complete
Enable_Device_Under_Test_Mode_Error

Read_Loopback_Mode

Read_Loopback_Mode will read the value for the setting of the Merlin's Wand Host Controller's Loopback_Mode. The setting of the Loopback_Mode will determine the path of information.

Command Parameters	Examples	Comments
N/A		

Return Events
Read_Loopback_Mode_Complete
Read_Loopback_Mode_Error

Write_Loopback_Mode

The Write_Loopback_Mode will write the value for the setting of the Host Controller's Loopback_Mode into Merlin's Wand.

Command Parameters	Examples	Comments
Loopback_Mode	0	0x00=No Loopback mode enabled. Default 0x01=Enable Local Loopback 0x02=Enable Remote Loopback

Return Events
Write_Loopback_Mode_Complete
Write_Loopback_Mode_Error

CATC-Specific HCI Commands**CATC_BER_Test**

This command will measure Bit Error Rate (BER) when fully loaded DH1, DH3, DH5, DM1, DM3 or DM5 packets are sent from master to slave on the link.

Command Parameters	Examples	Comments
HCI_Handle Number_Of_Packets	0000	0x0000=Unlimited number of packets will be sent 0x0001=0xFFFF number of packets will be sent

Command Parameters	Examples	Comments
BER_Packet_Type	03	0x00=DH1 0x01=DH3 0x02=DH5 0x03=DM1 0x04=DM3 0x05=DM5
Test_Data_Type	00	0x00=Send PBRS (same as in Bluetooth test mode) 0x01=Every octet that is sent equals Test_Data
Test_Data	FF	Data to send
BER_Interval	10	A packet is sent every BER_Interval frame

Return Events
CATC_BER_Test_Complete
CATC_BER_Test_Error

CATC_Change_Headset_Gain

Controls the gain of the microphone or speaker of the headset.

Command Parameters	Examples	Comments
Device	"Speaker"	"Microphone" or "Speaker" ("Speaker" is default)
Gain	0	0x00 - 0x0F

Return Events
CATC_Change_Headset_Gain_Complete
CATC_Change_Headset_Gain_Error

CATC_Read_Headset_Gain

Reads the gain of the microphone or speaker of the headset.

Command Parameters	Examples	Comments
Device	"Speaker"	"Microphone" or "Speaker" ("Speaker" is default)

Return Events
CATC_Read_Headset_Gain_Complete
CATC_Read_Headset_Gain_Error

CATC_Read_Revision_Information

Merlin's Wand uses this command to read the revision number of the Merlin's Wand baseband controller.

Command Parameters	Examples	Comments
N/A		

Return Events
CATC_Read_Revision_Information_Complete
CATC_Read_Revision_Information_Error

CATC_Self_Test

This command will perform self-test of Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
CATC_Self_Test_Complete
CATC_Self_Test_Error

CATC_Write_Country_Code

Writes the value of the Country_Code. **Note:** this command will not take effect until the device is reset.

Command Parameters	Examples	Comments
Country_Code	0x00	0x00 North America and Europe (default) 0x01 France

Return Events
CATC_Write_Country_Code_Complete
CATC_Write_Country_Code_Error

A.2 Other HCI Events

Events
Command_Complete
PIN_Code_Request
Paring_Complete

Events
Encryption_Change
Disconnect_Complete
Link_Key_Request_Complete

A.3 L2CAP Command Descriptions

ConfigurationResponse

Response to an incoming configuration request.

Command Parameters	Examples	Comments
Reason	"Accept"	"Accept" (Default) "Reject" "Reject - unacceptable params" "Reject - unknown options"

Return Events
ConfigurationResponse_Complete

ConfigurationSetup

Sets L2CAP connection options.

Command Parameters	Examples	Comments
ServiceType	0x01	
TokenRate	0x00	
TokenBucketSize	0x00	
PeakBandWidth	0x00	
Latency	0xFFFFFFFF	
DelayVariation	0xFFFFFFFF	

Return Events
ConfigurationSetup_Complete
Error

ConnectRequest

Requests establishment of an L2CAP channel in the remote Bluetooth device.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
PSM	0x1001	
Receive MTU	0x1B6	

Return Events
Connection_Complete
Connection_Failed

ConnectResponse

Indicates the response to the incoming connection request. This command should be executed before Connection Request.

Command Parameters	Examples	Comments
Response		Accept (Default) Reject_Pending Reject_PSM_Not_Supported Reject_Security_Block Reject_No_Resources

Return Events
ConnectResponse_Complete

DeregisterPsm

Deregisters the specified PSM.

Command Parameters	Examples	Comments
PSM	0x1001	

Return Events
DeregisterPsm_Complete
DeregisterPsm_Failed

DisconnectRequest

Requests the disconnection of the specified L2CAP channel.

Command Parameters	Examples	Comments
CID	0x0040	

Return Events
Disconnection_Complete
Disconnection_Failed

EchoRequest

Sends an Echo Request over the L2CAP channel.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
Data	"echo"	

Return Events
EchoRequest_Complete
EchoRequest_Failed

InfoRequest

Sends an Info Request over the L2CAP channel. Info requests are used to exchange implementation-specific information regarding L2CAP's capabilities.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
InfoRequest_Complete
InfoRequest_Failed

RegisterPsm

Registers a PSM identifier with L2CAP. Once registered, the protocol can initiate connections and data transfers as well as receive connection requests and data.

Command Parameters	Examples	Comments
PSM	0x1001	
Receive MTU	0x1B6	

Return Events
RegisterPsm_Complete
RegisterPsm_Failed

SendData

Sends data on the specified L2CAP channel.

Command Parameters	Examples	Comments
CID Data Pipe	0x0040 "Pipe1"	Data_Pipe should be created in the Data Transfer Manager.

Return Events
SendData_Complete SendData_Failed

A.4 Other L2CAP Events

Events
Connection_Indication Disconnection_Indication Data_Indication Write_Configuration_Complete Command_Complete Error

A.5 SDP Command Descriptions**AddProfileServiceRecord**

This command will add a pre-defined Service Record according to one of the Bluetooth wireless technology profiles to the SDP database.

Command Parameters	Examples	Comments
Profile	HeadSet	The following are values of the Profile parameter: HeadsetAudioGateway Headset SerialPort Dialup Fax LAN FileTransfer ObjectPush Sync SyncCommand InterCom Cordless

Command Parameters	Examples	Comments
ServerChannel	0x01	Server channel has to be entered for all profiles except for InterCom and Cordless.

Return Events
AddProfileServiceRecord_Complete
AddProfileServiceRecord_Error

AddServiceRecord

This command will add a pre-defined Service Record according to one of the Bluetooth wireless technology profiles to the SDP database.

Command Parameters	Examples	Comments
Filename	"C:/Records.sdp"	Click the "..." button to choose a file. Choosing a file will automatically load the records within that file, and those record names will be in the drop-down for Record Name.
Record Name	"FTP Test Record"	Record Names loaded from the Filename file will be in the drop-down for this parameter.
ServerChannel		To leave the server channel as is, enter 0 or leave this blank.

Return Events
AddServiceRecord_Complete
AddServiceRecord_Failed

ProfileServiceSearch

This command will search for support of one of the Bluetooth wireless technology profiles.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Command Parameters	Examples	Comments
Profile	HeadSet	The following are values of the Profile parameter: HeadsetAudioGateway Headset SerialPort Dialup Fax LAN FileTransfer ObjectPush Sync SyncCommand InterCom Cordless

Return Events
ProfileServiceSearch_Complete
ProfileServiceSearch_Failed

RequestServiceAttribute

This command will retrieve specific attribute values from a specific service record. The Service Record Handle from a specific Service Record and a list of AttributeIDs to be retrieved from that Service Record are supplied as parameters. Up to three AttributeIDs can be searched in one request. Service Record Handle is usually retrieved by using RequestServiceSearch command.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	You can specify between 0 and 3 AttributeIDs.
ServiceRecordHandle	0x00010000	
AttributeID	0x1108	
AttributeID	0x1203	

Return Events
RequestServiceAttribute_Response
Search_Failed

RequestServiceSearch

This command will locate Service Records that match the ServiceSearch Pattern of Service Class IDs. The SDP server will return all Service Record Handles of Service Records that match the given Service Search Pattern. Up to three ServiceClassIDs can be searched in one request.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	Between 0 and 3 Service Class IDs can be specified.
ServiceClassID	0x1204	
ServiceClassID	0x1110	
ServiceClassID	0x1000	

Return Events
RequestServiceSearch_Response
Search_Failed

RequestServiceSearchAttribute

This command combines the capabilities of the RequestServiceAttribute and RequestServiceSearch into a single request. This command will retrieve all Attribute values that match the ServiceSearch pattern.

Command Parameters	Examples	Comments
HCI_Handle		Note that all Attributes are requested since a range of 0x0000-0xFFFF is specified by default.
ServiceClassID		
ServiceClassID		
ServiceClassID		

Return Events
RequestServiceSearchAttribute_Response
Search_Failed

ResetDatabase

This command will remove all Service Records from the database in Merlin's Wand.

Command Parameters	Examples	Comments
N/A		

Return Events
ResetDatabase_Complete
ResetDatabase_Failed

A.6 RFCOMM Command Descriptions

AcceptChannel

This command will accept or reject incoming request to open an RFCOMM channel from RFCOMM server. This command should be executed before RFCOMM connection request from another device. By default, all connection requests are accepted.

Command Parameters	Examples	Comments
Accept	TRUE	(Values: TRUE/FALSE)

Return Events
AcceptChannel_Complete

AcceptPortSettings

This command will accept or reject PortSettings received during RequestPortSettings event. This command should be executed before PortSettings request from another device. By default, all requests are accepted.

Command Parameters	Examples	Comments
Accept	TRUE	(Values: TRUE/FALSE)

Return Events
AcceptPortSettings_Complete

AdvanceCredit

Advances a specified number of credits to a particular RFCOMM connection.

Command Parameters	Examples	Comments
(HCI / DLCI)	(0x0001, 0x02)	
Credit	20	Number of credits to advance

Return Events
AdvanceCredit_Complete
AdvanceCredit_Error

CloseClientChannel

This command will close an established RFCOMM channel between Merlin's Wand and a remotely connected device.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	The DLCI value is returned by the OpenClientChannel command

Return Events
CloseClientChannel_Complete
CloseClientChannel_Error

CreditFlowEnabled

This command is used to check if credit-based flow control has been negotiated for the current RFCOMM session.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	The DLCI value is returned by the OpenClientChannel command

Return Events
CreditFlowEnabled_Complete

DeregisterServerChannel

Deregisters an RFCOMM server channel.

Command Parameters	Examples	Comments
ServerChannel	0x01	ServerChannel must first be registered via RegisterServerChannel command.

Return Events
DeregisterServerChannel_Complete
DeregisterServerChannel_Error

OpenClientChannel

This command will open an RFCOMM channel.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	
ServerChannel	0x01	Range: 1-30
MaxFrameSize	0x7F	Range: 23-32767 Default is 127
Credit	0x20	The number of frames that the sender has available

Return Events
OpenClientChannel_Complete
OpenClientChannel_Failed

RegisterServerChannel

This command will register ServerChannel with an RFCOMM server so that the server can respond to incoming OpenClientChannel requests.

Command Parameters	Examples	Comments
N/A		

Return Events
RegisterServerChannel_Complete
RegisterServerChannel_Error

RequestPortSettings

This command will request a change to the current PortSettings.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	For values, see below.
BaudRate	9600	
DataFormat	0x02	
FlowControl	0x00	
Xon	0x11	
Xoff	0x13	

Return Events
RequestPortSettings_Complete
RequestPortSettings_Failed

Parameter Values:

- BaudRate: Specifies the baud rate. Note that the baud rate setting does not actually affect RFCOMM throughput.

Values: 2400, 4800, 7200, 9600, 19200, 38400, 57600, 115200, 230400

- DataFormat: The following values identify the number of data bits.

Values:

0x00 -- DATA_BITS_5

0x02 -- DATA_BITS_6

0x01 -- DATA_BITS_7

0x03 -- DATA_BITS_8

- Flow Control:

Values:

0x00 -- FLOW_CTRL_NONE

0x01 -- XON_ON_INPUT

0x02 -- XON_ON_OUTPUT

0x04 -- RTR_ON_INPUT

0x08 -- RTR_ON_OUTPUT

0x10 -- RTC_ON_INPUT

0x20 -- RTC_ON_OUTPUT

Xon:

Value: Default Xon char -- 0x11

- Xoff:

Value: Default Xoff char -- 0x13

RequestPortStatus

This command will request the status of the PortSettings for the remote device.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	

Return Events
RequestPortStatus_Complete
RequestPortStatus_Error

SendData

Causes Merlin's Wand to send data by pipe value to remote device over the specified channel.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	

Command Parameters	Examples	Comments
Data Pipe	Pipe1	Data Pipe should be created in the Data Transfer Manager

Return Events
SendData_Complete SendData_Failed

SendTest

This command causes Merlin's Wand to sent a test frame to a remote device over the specified channel.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	

Return Events
SendTest_Complete SendTest_Failed

SetLineStatus

This command will send the LineStatus command to the remote device. It allows the RFCOMM user to communicate overrun framing and parity errors to the remote device.

Command Parameters	Examples	Comments
(HCI/DLCI) LineStatus	(0x0001, 0x02) 0x0F	LineStatus Values: 0x01 -- Set to indicate an error. 0x02 -- Set to indicate an overrun error. 0x04 -- Set to indicate a parity error. 0x08 -- Set to indicate a framing error.

Return Events
SetLineStatus_Complete SetLineStatus_Failed

SetModemStatus

This command will send the ModemStatus to the remote device. It allows the user to send Flow Control and V.24 signals to the remote device.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	
ModemSignals	0x8C	Modem Signal Values: 0x8c - Ready to communicate, ready to receive, and data valid 0x02 (FLOW) - Set when sender is unable to receive frames 0x04 (RTC) - Set when sender is ready to communicate. 0x08 (RTR) - Set when sender is ready to receive data. 0x40 (IC) - Set when a call is incoming. 0x80 (DV) - Set when valid data is being sent.
BreakLength	0x00	0-15 indicates the length of the break signal in 200ms units. 0 indicates no break signal.

Return Events
SetModemStatus_Complete
SetModemStatus_Failed

SendATCommand

This command will send a selected AT command.

Command Parameters	Examples	Comments
(HCI/DLCI)	(0x0001, 0x02)	
AT_Command	RING	Values: AT + CKPD = 200: Headset Button pressed. AT + VGM = 1: Microphone gain. +VGM = 1: Microphone gain 1. RING OK ERROR BUSY CONNECT NO_CARRIER NO_DIAL_TONE

Return Events
Send_AT_Command_Complete
Send_AT_Command_Error

A.7 Other RFCOMM Events

Events
OpenClientChannel_Request
CloseClientChannel_Indication
Data_Indication
PortNegotiation_Indication
RequestPortStatus_Indication
ModemStatus_Indication
LineStatus_Indication
Flow_Off_Indication
Flow_On_Indication

A.8 TCS Command Descriptions

RegisterIntercomProfile

Registers an Intercom identifier with TCS. Once registered, the protocol can initiate connections as well as receive connection requests.

Command Parameters	Examples	Comments
N/A		

Return Events
Register_Intercom_Profile_Complete
Register_Intercom_Profile_Error

Open_TCS_Channel

This command opens an L2CAP channel with TCS PSM and initializes a TCS state machine into NULL state.

Command Parameters	Examples	Comments
HCI_Handle	0x0001	

Return Events
Open_TCS_Channel_Complete
Open_TCS_Channel_Failed

Start_TCS_Call

This command must be called right after TCSOpenChannel. It automatically sends a sequence of TCS messages according to the Intercom profile specification of the TCS state machine. After successful execution of this command, TCS state machine is in ACTIVE state and SCO connection is opened.

Command Parameters	Examples	Comments
N/A		

Return Events
Start_TCS_Call_Complete
Start_TCS_Call_Error

Disconnect_TCS_Call

This command is called to close an existing TCS connection according to the Intercom profile specification of the TCS state machine, close the L2CAP connection, and close the SCO connection.

Command Parameters	Examples	Comments
N/A		

Return Events
Disconnect_TCS_Call_Complete
Disconnect_TCS_Call_Error

Send_Info_Message

This command can be called after a TCS channel is opened. It sends an INFORMATION TCS message with a called party number.

Command Parameters	Examples	Comments
Phone_Number	408 727 6600	Phone number may contain up to 10 digits.

Return Events
Send_Info_Complete
Send_Info_Error

A.9 OBEX Command Descriptions

ClientConnect

This command will create an OBEX connection with a remote device.

Command Parameters	Examples	Comments
BD_ADDR	0x010203040506	An HCI Connection has to be established before calling this command.

Return Events
ClientConnect_Complete
ClientConnect_Error

ClientDisconnect

This command will cause the remote device to close the established OBEX channel.

Command Parameters	Examples	Comments
N/A		

Return Events
ClientDisconnect_Complete
ClientDisconnect_Error

ClientGet

This command will initiate an OBEX Get operation in the remote device for the object named in the store handle. This operation is only valid over an OBEX connection.

Command Parameters	Examples	Comments
Object	"VCard.vcf"	

Return Events
ClientGet_Complete
ClientGet_Error