AJTWBSLIM2

User Guide

Issue 1

2018 / 07 / 02

1 Introduction

BlueSuite™ is a **BlueCore**® development suite designed to help with your **Bluetooth**® wireless technology development program and with all testing requirements for the BlueSlim 2 module.

BlueSuite contains:

- Graphical and command-line applications to configure, test and program the BlueCore device in your design.
- The **TrueTest™** Toolkit, a set of libraries for developing production test applications for your BlueCore enabled devices.
- Device drivers to communicate with BlueCore.
- Documentation to help development.

This user guide explains:

- The minimum system requirements in order to be able to run Bluesuite. See Section 2.
- How to install BlueSuite Section 3 and Section 4.
- The features and functions Bluesuite contains. See Section 7.

2 System Requirements

Minimum system requirements for BlueSuite are:

- Pentium processor based personal computer
- 64MB of RAM
- 20MB of available hard disk space

2.1 Operating System

BlueSuite can be used on the following operating systems:

- Microsoft Windows XP Service Pack 2 and later (not x64)
- Microsoft Windows Vista (not x64)
- Microsoft Windows 2000 Service Pack 3 and later (only for versions of BlueSuite prior to 2.0)

See the release notes for testing information and detailed operating system requirements.

2.2 Software Required

Your copy of BlueSuite comes with:

- Microsoft .NET Framework Version 2.0 Redistributable Package
- Visual C++ 2005 Redistributable Package

The BlueSuite installer automatically installs these packages on your computer if they are missing.
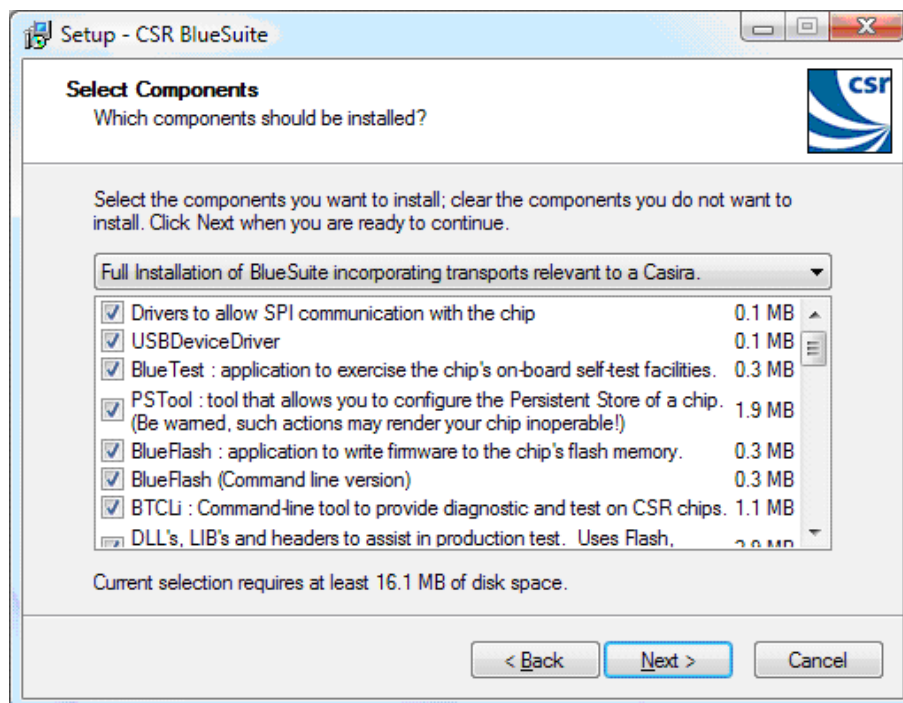
3   Install BlueSuite

Note:

A bug in BlueSuite version 1.23 means that even if a later version of BlueSuite is installed, the version information for the USB driver may still indicate 1.23. If version 1.23 has previously been installed on the PC, unplug all CSR hardware, make sure you have the latest version of BlueSuite installed and run `usbpurger` from the BlueSuite installation directory. When a CSR BlueCore is plugged in, the Hardware Wizard prompts you for drivers.

Run the BlueSuite installer and follow the instructions displayed by the setup wizard.

By default the installer does a full installation, as shown in Figure 3.1. If you do not want to install all of the possible components, be sure that you install:

- **Drivers to allow SPI communication with the chip**. You will need these to back-up your BlueCore device.
- **USBDeviceDriver** if you want to connect your device to your computer's USB port.
- **DLL's, LIB's and headers to assist in production test** if you want to do production testing.

Figure 3.1: Full Installation of BlueSuite

When you get to the Additional Tasks dialogue in Figure 3.2, you have to select the default debug transport. This configures the default behaviour of the BlueSuite programs. You can change this after installation by changing environment variable `SPITRANS` to `LPT` or `USB`. See Section 7.1
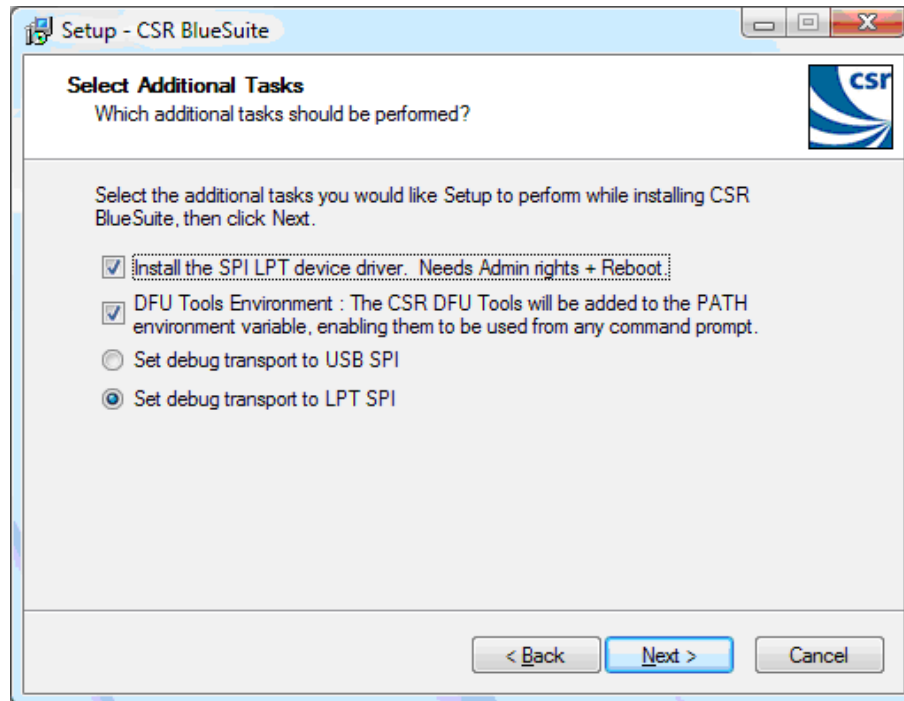
Figure 3.2: Additional Tasks Dialogue

4 Install a USB Driver

BlueCore devices can be used with a USB host interface. The USB host interface is not a debugging interface and can be used by end users or in production modules. If you want to use the USB host interface to BlueCore, you must first configure your BlueCore for the USB host interface, and then install a USB device driver.

5 Firmware Overview

CSR's BlueCore ICs contain control software (called firmware). Firmware in BlueCores with flash can be updated using tools in BlueSuite:

- **BlueFlash** (see Section 7.2) is used during development to update your BlueCore's firmware using the Serial Peripheral Interface (SPI) debugging interface. **TestFlash** (part of **TrueTest**, see Section 8) can be used for the same purpose on the production line.
- **DFUWizard** (see Section 7.9) can sometimes be used to upgrade the firmware both in development and in the field.

  DFU (Device Firmware Upgrade) is a USB standard that allows end users to upgrade the BlueCore's firmware over the USB or UART (Universal Asynchronous Receiver Transmitter) interface.

  DFUWizard requires some firmware (the loader) to already be present in the device. The loader can only be updated by BlueFlash.

The firmware version number is an important piece of information. You can identify your current firmware version using BlueFlash, see Section 7.2.

5.1 Firmware File Formats

Table 5.1 describes the different firmware file formats you may encounter.

File Extension(s)	Description
`.xpv/.xdv`	The standard BlueCore firmware file format holds the binary data for firmware releases in two separate files, each of which hold part of the data.
`.xuv`	This firmware file format is one file containing the contents of both the `.xpv` file and the `.xdv` file.
`.xhv`	Chips with integrated flash have firmware files containing extra production information about the flash.
`.dfu`	File format for use with **DFUWizard**. **Note:** **DFUTools**, see Section 7.12, can convert `.xpv` and `.xdv` files into a `.dfu` file.

Table 5.1: Firmware File Formats

The `.dfu` file format is a flexible container format that can hold multiple different firmware images (for BlueCore2 and later chips) and multiple versions of the Persistent Store. This means you can have a universal `.dfu` file that works for multiple hardware devices.

See *BlueCore Device Firmware Upgrade Overview* for a description of what DFU is and how it is used.

5.2 Persistent Store

As well as containing the firmware image (i.e. control software), firmware files can also contain configuration information known as Persistent Store (PS). DFU files can contain information for a partial or full update of the PS. Any areas of the PS that are not in the DFU file will remain as they were before the update. Other firmware files cannot contain partial information; they either preserve or replace the PS.

Firmware upgrades from the CSR support website do not contain any PS settings, thus the current settings are preserved. Firmware dumps that you make contain PS settings, and so will overwrite any existing ones if you upgrade using a previously dumped file.

Note:

> If firmware with no PS is downloaded onto a blank flash, a new PS is created on the flash memory using default values suitable for a Casira module. However, some keys require calibration per module for optimal performance.

> See `pskeys.html` in the firmware release for details on using individual PS Keys. Some of the same information is available by clicking the **Describe** button in **PSTool**, see the *PSTool User Guide*.

5.3   Firmware Build Types

The Bluecore's Bluetooth stack firmware comes in two forms, either containing the Bluetooth stack layers up to the Host Controller Interface (HCI) layer, or containing all the stack layers up to the Radio Frequency COMMunication (RFCOMM) layer.

Since Firmware version 18, firmware builds are called Unified. They contain all the Bluetooth stack layers up to RFCOMM, but can be configured to present either an HCI interface, or an RFCOMM interface. The behaviour is controlled by `PSKEY_ONCHIP_HCI_CLIENT`, see the firmware release files for more information.

All types of firmware builds can be held in any BlueCore firmware file format.

5.4   Upgrading Firmware

Go to the Bluetooth Firmware section on the CSR support website (www.csrsupport.com/BluetoothFirmware) to get the latest BlueCore firmware files. Use **BlueFlash** or **DFUWizard** to download the firmware to your BlueCore device.

Note:

> Casira users should see the *Casira User Guide* for instructions on upgrading their firmware.

6 Getting Started

When you have connected your BlueCore device(s) to your computer, you can use BlueSuite tools to communicate with them.

To test you have connected your BlueCore device(s) succesfully, you can use:

- The command-line tool **btcli** (Section 7.5).
- One of the graphical programs PSTool (See the *PSTool User Guide*) or BlueTest3 (See the *BlueTest User Guide*).

6.1 Exploring the Bluetooth Stack

Use btcli if you want to explore the Bluetooth stack at all before starting development, as it provides a lower-level interface than the graphical programs in BlueSuite.

The Bluetooth stack is described in the *Specification of the Bluetooth System* from www.bluetooth.com.

7 BlueSuite Programs

This chapter describes the programs included in BlueSuite. The features of each program are briefly described and any relevant documentation is listed. Some programs without separate user guides are described in full detail.

7.1 Transport Options for Command Line Tools

Some command line tools in BlueSuite take a `-trans` option, which controls the transport options. Table 7.1 indicates which variables can be used with each tool:

Variable	e2bluecmd/ blueflashcmd	pscli	Default	Description	Example Values
CSRTRANS	-	x	SPI	The transport to use	`SPI`, `USB`, `BCSP`, `H4` and `H5`
HCIPORT	-	x	autodetect if USB, otherwise COM1	The port to use for the selected HCI transport if not `CSRTRANS=SPI`	`com1`, `com2`, `csr0`
HCIBAUD	-	x	115200	The baud rate to use for uart based HCI transports	`115200`, `38400`, etc.
SPITRANS	x	x	LPT	The SPI transport to use if `CSRTRANS=SPI`	`LPT`, `USB`
SPIPORT	x	x	autodetect	The port to use for the selected SPI transport	▪ A LPT port number: `1`, `2` etc. ▪ A USB-to-SPI converter's ordinal position: `0`, `1`, `2`. ▪ A USB-to-SPI converter's serial number (as printed on the converter e.g. `185920`)
SPIMUL	x	x	0	The device to use if the SPI port is multiplexed	`0`, `1`, etc.

Table 7.1: Command Line Transport Options

Note:

Where appropriate, BlueSuite tools attempts to prepend `\\.\` to a string in HCIPORT. For example the name of a CSR USB device on WinXP is `\\.\csr0`; BlueSuite tools re-interpret `csr0` as `\\.\csr0`.

Examples

To use USB-SPI (where 185920 is the serial number printed on the USB-SPI):

`-trans "CSRTRANS=SPI SPITRANS=USB SPIPORT=185920"`

(or just `"SPITRANS=USB SPIPORT=185920"`)

To use BCSP on COM1 at 38.4 kbaud:

`-trans "CSRTRANS=BCSP HCIPORT=com1 HCIBAUD=38400"`

It is possible to set these as Windows environment variables, which are then used as the default setting for a variable e.g. `"set SPITRANS=USB"` would change the default SPI transport from LPT-SPI to USB-SPI.

7.2   BlueFlash

BlueFlash is a utility that allows you to download and upload firmware to and from the flash memory on the Casira modules or on your own BlueCore designs.

As described in Section 5, there are several different firmware file formats, and firmware files may contain Persistent Store settings. Read Section 5.2 to make sure you do not accidentally change Persistent Store settings and to see which parts of BlueCore's firmware is updated by BlueFlash.

Important Note:

> Do not erase the Persistent Store with **Flash Erase** (either by selection or via **Full Erase**) unless you are completely sure.

> Fully erasing the flash removes all Persistent Store settings. Use **Full Erase** only in the event that there may be a problem with the Persistent Store and you want to reload to a previous known working version.

7.2.1   User Interface

BlueFlash opens with a user interface window, see Figure 7.1.

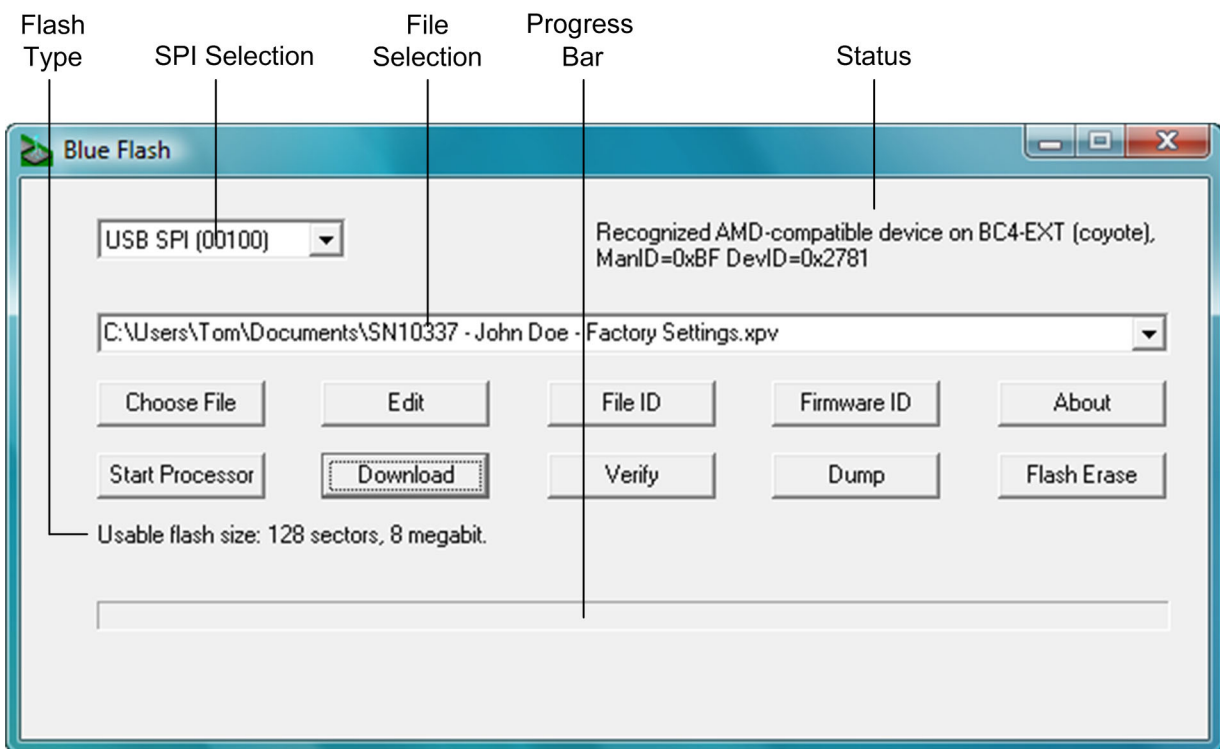Table 7.2 explains the function of each button or area.

Figure 7.1: BlueFlash User Interface in Windows Vista

Button or Area	Explanation
Choose File	Opens a dialogue box for selection of the firmware file to download to the flash on the BlueCore Module. You may open a `.xpv`, `.xuv` or `.xbv` firmware file **Note:** If you open a `.xpv` file, the `.xdv` file must be present in the same folder.
Edit	If pressed, a message indicates that this is no longer supported.
Download	Copies the selected file to the flash on the BlueCore module. This process updates each flash block as required and verifies them in turn.
Verify	Compares the selected file with the code programmed into the flash on the BlueCore module. **Verify** only compares components contained in the file. The status area shows any reported differences.
Status	Provides a text report of the status of the current operation.
Progress Bar	Displays a visual indication of progress for the current operation.
Start Processor	Changes depending on the status of the processor. If it displays **Stop Processor**, the processor is running. It must be stopped before attempting to download code. When the new code has been loaded, the processor can be restarted by clicking **Start Processor**.
Firmware ID	Identifies which version of firmware is currently loaded on the module. If BlueFlash is unable to identify the firmware version, you can still upgrade the firmware using this utility.
Dump	Allows the user to dump the contents of the flash on the BlueCore module to a file. The data can be stored either as a pair of `.xpv`/`.xdv` files, as a combined `.xuv` file or as a raw binary file. The dumped file also contains all of the Persistent Store settings, allowing you to restore to a known firmware build and Persistent Store settings.
Flash Erase	Provides options for erasing some or all of the flash memory including the Persistent Store area.
File ID	Displays the name and version of firmware of the file selected for download on the module.
File Selection	Displays the name and location of the file selected for download.
Flash Type	Displays the type of flash memory that has been identified on the BlueCore module. Flash type is identified after you stop the processor. Different types of flash memory may have different memory block structures that require adjustments to BlueCore's memory map. If the flash type indicates `Unrecognised Flash assuming SST`, either the flash on the BlueCore module is not supported, or there is a problem with the SPI connection. See Section 7.2.2.
SPI Selection	Allows the user to select the LPT or USB port to which the SPI cable is connected.
About	Indicates the version of the BlueFlash application in use. CSR advises using the most up to date version. Refer to the CSR support website for available updates.

Table 7.2: Explanation of the BlueFlash User Interface

7.2.2   Troubleshooting BlueFlash

Potential problems:

- If BlueFlash cannot identify the current build of firmware on BlueCore, check your firmware version.
- If BlueFlash cannot identify flash type, BlueFlash fails to stop processor or BlueFlash times out trying to make a connection to BlueCore, ensure that the following actions have been completed:
    - Run `InstParSpi.bat` (`C:\Program Files\CSR\BlueSuite\drivers`) to copy and register the `parspi.sys` device driver, making sure you have administrative rights . Reboot your PC.
    - Driver has been installed in the registry by looking for
      `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Parspi`
    - SPI cable is correctly connected
    - Check for conflicts with other drivers, especially printer drivers (this includes the PDF converters). Conflicting printer drivers should be removed. To amend or remove a driver, select **Printers and Faxes** in the **Control Panel** and select **File\Server Properties**. Click on the **Ports** tab. Move the port to another location or delete the device.
    - Ensure that another SPI application is not trying to access the SPI port. SPI uses mutexes around port access, therefore another application may lock the port. Close all SPI based applications (use Task Manager if necessary) and retry. Reboot if necessary.

7.2.3   BlueFlashCmd

BlueFlashCmd is a command-line tool to download and upload firmware to flash memory on your BlueCore device. It can do everything BlueFlash does.

Type `BlueFlashCmd.exe -help` for usage instructions.

Note:

Running BlueFlashCmd will reset your BlueCore chip

7.3   PSTool

PSTool is an editing suite that allows you to read and modify the Persistent Store. The Persistent Store consists of configuration keys that modify the operation of the firmware. BlueCore is a very flexible device with many setup options. PS Keys must be set correctly for BlueCore to work in a particular design. Some PS Keys must be calibrated for each module.

7.3.1   PsCli

PsCli is a command-line interface for reading and modifying the Persistent Store. Type `pscli.exe -help` to see usage instructions.

7.3.2   Documentation

PSTool and pscli are described in the *PSTool User Guide*.

See `pskeys.html` in the firmware release for details on using individual PS Keys.

7.4   BlueTest3

BlueTest3 is a program that enables you to execute BlueCore's Built In Self Test (BIST) functions for RF testing. The BIST functions consist mainly of low level radio tests (transmitting a continuous wave at a prescribed frequency or receiving pseudo random data and calculating bit error rates). Included are tests for the PCM port and other internal blocks.

7.4.1   Documentation

See the *BlueTest User Guide*.

7.5   BTCli

Bluetooth Command Line Interface (BTCli) is a command-line program that lets you send Host Controller Interface (HCI) commands to your BlueCore device. Host computers use the HCI interface (part of the Bluetooth specification) to communicate with Bluetooth controllers (e.g. your BlueCore chip).

See the *BTCli User Guide* for full details.

7.6   RFCli

RFCli is a command line utility for communicating with RFCOMM firmware builds. It enables you to send and receive RFCOMM primitives to the chip (as well as other protocols). It also has a Tool Command Language (Tcl) interpreter built in so you can write scripts.

See the *RFCli User Guide* for full details.

7.7   BCFMRadio

Important Note:

> Read the document referenced below before running this program, as you must configure your BlueCore device before this program will connect to it.

BCFMRadio is a graphical tool for exercising the FM receiver on Bluecore5-FM. It is able to communicate with the Bluecore using I²C or BCCMDs using BCSP, H4, H4 Deep Sleep, H5, USB and SPI. BCFMRadio provides a user-friendly interface allowing configuration of the BlueCore5-FM radio.

See *Operating the BlueCore5-FM Radio Application Note*.

7.8   E2BlueCmd

E2BlueCmd is a command-line tool which can read and write from/to EEPROM to initialise the BlueCore EEPROM header, read and write blocks, and verify EEPROM contents against a file.

E2BlueCmd works over the SPI interface.

Type `e2bluecmd -help` for usage instructions.

7.9   DFU Wizard

The DFU Wizard is a program for end users to upgrade the firmware on their BlueCore devices.

7.10   DFU Babel

The USB to SPI Converter (or Babel) transforms the USB packets from the host computer to SPI signals sent out over Category 5 cable.

DFU Babel is a program that lets you upgrade your USB to SPI Converter with new firmware. If CSR release a firmware upgrade for the USB to SPI Converter   .

See the *USB-SPI Converter User Guide*.

7.10.1   Upgrading The USB to SPI Converter Firmware

Important Note:

> Do not attempt to upgrade a USB to SPI Converter with firmware designed for different hardware (e.g. Casira firmware). The USB to SPI Converter needs special firmware and any other firmware may make it permanently unusable.

See the *USB-SPI Firmware Upgrade Guide*.

7.11   DFU ToolTips

BlueSuite comes with DFU ToolTips, a shell extension that shows tool tips for DFU files in Windows. To see the tool tip, point the mouse over a DFU file, and wait for the tool tip to appear. Figure 7.2 shows a DFU tool tip.

If the tool tip does not appear, you may have disabled tool tips in your version of Windows. To enable tool tips for files:
1. Go to the **Control Panel**
2. Open **Folder Options**
3. Select the **View** tab
4. Scroll down the Advanced settings to **Show pop-up description for folder and desktop items**. Tick this box.
5. Click **OK**

DFU ToolTips also lets you right-click on a DFU file, and **Download to BlueCore**. This runs the DFU Wizard to download the DFU file to your BlueCore device.
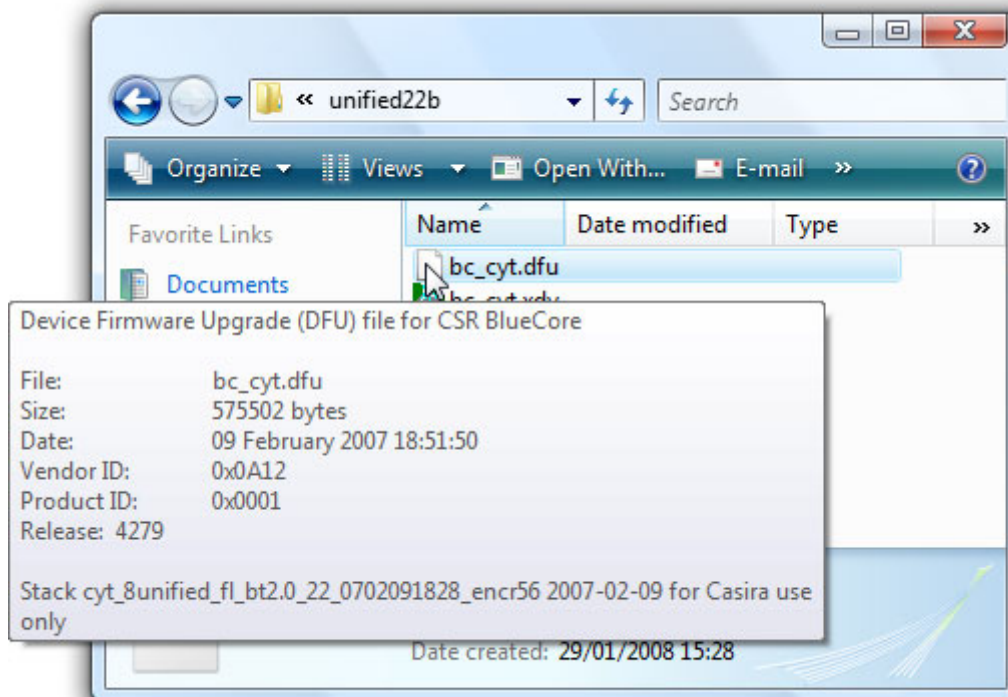
Figure 7.2: A DFU ToolTip

7.12   DFU Tools

DFU Tools are a suite of command-line tools to let your sign and combine firmware and persistent store files to form DFU files.

The DFU Tools documentation can be accessed from the **Start** Menu. Open the **CSR BlueSuite** item, and click **CSR DFU Tools Documentation**.

8 TrueTest Toolkit

The TrueTest toolkit consists of the necessary libraries and documentation to let you develop applications in a variety of languages for production programming and testing of BlueCore enabled devices.

The toolkit is designed for use in production test systems. Instead of writing test scripts to call other BlueSuite programs, you can write a test program that accesses the test APIs (*Application Programming Interface*) directly.

See the *TrueTest Toolkit Introduction*.

9 Document References

Document	Reference
Specification of the Bluetooth System	Version 2.1 + EDR, Core Package, 26 July 2007 www.bluetooth.com
BlueCore Class 2 Example Designs	www.csrsupport.com
BlueCore Production Information Data Books	www.csrsupport.com
BlueCore Device Firmware Upgrade (DFU) Overview	CS-112861-AN
USB Device Firmware Upgrade Specification	Part of the USB Specification http://www.usb.org/developers/devclass_docs/ DFU_1.1.pdf
PSTool User Guide	CS-101505-UG
BlueTest User Guide	CS-102736-AN
USB-SPI Converter User Guide	CS-114776-UG
TrueTest Toolkit Introduction	CS-101531-AN
Casira User Guide	CS-102077-UG
USB-SPI Firmware Upgrade Guide	CS-203416-UG
USB Device Driver User Guide	CS-208306-UG

Appendix A   Regulatory Notes and Information

This device complies with Part 15 of the FCC Rules.

Operation is subject to the two following conditions:

1. This device may not cause harmful interference
2. This device must accept any interference received, including interference that may cause undesired operation.

Changes or modifications to the transmitter not expressively approved by CSR could void the authority ability to operate the equipment.

User Label Warning

The transmitter is in compliance with the Part 15 FCC Rules. A label which demonstrates compliance with the FCC is located on the bottom of the laptop. The label contains the following information:

- Contains FCC ID:- PIWBLUESLIM2

Transmitter Warning

Installers of the Bluetooth transmitter should note that the Bluetooth transmitter should not be co-located with any other transmitter, for example WLAN interfaces in the lap top. The distance between the Bluetooth transmitter and antennas of other radio equipment must be at least 20cm in distance away from each other. Any closer installations require additional certification filings to examine co-location issues.

A.1   Industry Canada Statements

Operation is subject to the following two conditions:

1. This device may not cause interference
2. This device must accept any interference, including interference that may cause undesired operation of the device.

To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the equivalent isotropically radiated power (EIRP) is not more than that permitted for successful communication.

This device has been designed to operate with the antennas listed below and having a maximum gain of 3dBi. Antennas not included in this list or having a gain greater than 3dBi are strictly prohibited for use with this device. The required antenna impedance is 50Ω.

Module BLUESLIM2 uses a printed dual-meander antenna designed by CSR.

A.2   OEM Responsibilities

The BLUESLIM2 Bluetooth Module has been certified for integration into products only by OEM integrators under the following conditions:

1. The antenna(s) must be installed such that a minimum separation distance of 20cm is maintained between the radiator (antenna) and all persons at all times.
2. The transmitter module must not be co-located or operating in conjunction with any other antenna or transmitter.

As long as the two conditions above are met, further transmitter testing will not be required. However, the OEM integrator is still responsible for testing their end-product for any additional compliance requirements required with this module installed (for example, digital device emissions, PC peripheral requirements, etc.).

Important Note:

In the event that these conditions can not be met (for certain configurations or co-location with another transmitter), then the FCC and Industry Canada authorizations are no longer considered valid and the FCC ID and IC Certification Number can not be used on the final product. In these circumstances, the OEM integrator will be responsible for re-evaluating the end product (including the transmitter) and obtaining a separate FCC and Industry Canada authorization.

Terms and Definitions

Term	Definition
API	Application Programming Interface
BCHS	BlueCore Host Software
BIST	Built-In Self-Test
BlueCore®	Group term for CSR's range of Bluetooth wireless technology ICs
Bluetooth®	Set of technologies providing audio and data transfer over short-range radio connections
BTCli	Bluetooth Command Line Interface
CD ROM	Compact Disk Read Only Memory
CSR	Cambridge Silicon Radio
DFU	Device Firmware Upgrade
DLL	Dynamically Linked Library
e.g.	*exempli gratia*, for example
EDR	Enhanced Data Rate
EEPROM	Electrically Erasable Programmable Read Only Memory
etc	*et cetera*, and the rest, and so forth
HCI	Host Controller Interface
i.e.	*Id est*, that is
LPT	Line Printer Terminal
PC	Personal Computer
PCM	Pulse Code Modulation
PDF	Portable Document Format
Persistent Store	Store of configuration values in non-volatile memory
PsCli	Persistent Store Command Line Interface
RAM	Random Access Memory
RF	Radio Frequency
RFCli	RFCOMM Command Line Interface
RFCOMM	Radio Frequency COMMunication. Protocol layer providing serial port emulation over L2CAP
SPI	Serial Peripheral Interface
Tcl	The Tool Command Language, a scripting language
UART	Universal Asynchronous Receiver Transmitter
USB	Universal Serial Bus