NAME

dfu-util − Device firmware update (DFU) USB programmer

SYNOPSIS

dfu-util −l [**−v**] [**−d** *vid:pid*[,*vid:pid*]] [**−p** *path*] [**−c** *configur ation*] [**−i** *interface*] [**−a** *alt-intf*] [**−S** *serial*[,*serial*]]

dfu-util [**−v**] [**−d** *vid:pid*[,*vid:pid*]] [**−p** *path*] [**−c** *configur ation*] [**−i** *interface*] [**−a** *alt-intf*] [**−S** *serial*[,*serial*]] [**−t** *size*] [**−Z** *size*] [**−w**] [**−s** *addr ess*] [**−R**] [**−D** | **−U** *file*]

dfu-util [**−hV**]

DESCRIPTION

dfu-util is a program that implements the host (computer) side of the USB DFU (Universal Serial Bus Device Firmware Upgrade) protocol.

dfu-util communicates with devices that implement the device side of the USB DFU protocol, and is often used to upgrade the firmware of such devices.

OPTIONS

−l, −−list

List the currently attached DFU capable USB devices.

−d, −−device [*Run-Time VENDOR*]:[*Run-Time PRODUCT*][,[*DFU Mode VENDOR*]:[*DFU Mode PRODUCT*]]

Specify run-time and/or DFU mode vendor and/or product IDs of the DFU device to work with. **VENDOR** and **PRODUCT** are hexadecimal numbers (no prefix needed), "*" (match any), or "-" (match nothing). By default, any DFU capable device in either run-time or DFU mode will be considered.

If you only have one standards-compliant DFU device attached to your computer, this parameter is optional. However, as soon as you have multiple DFU devices connected, dfu-util will detect this and abort, asking you to specify which device to use.

If only run-time IDs are specified (e.g. "**--device 1457:51ab**"), then in addition to the specified run-time IDs, any DFU mode devices will also be considered. This is beneficial to allow a DFU capable device to be found again after a switch to DFU mode, since the vendor and/or product ID of a device usually changes in DFU mode.

If only DFU mode IDs are specified (e.g. "**--device ,951:26**"), then all run-time devices will be ignored, making it easy to target a specific device in DFU mode.

If both run-time and DFU mode IDs are specified (e.g. "**--device 1457:51ab,:2bc**"), then unspecified DFU mode components will use the run-time value specified.

Examples:

--device 1457:51ab,951:26

Work with a device in run-time mode with vendor ID 0x1457 and product ID 0x51ab, or in DFU mode with vendor ID 0x0951 and product ID 0x0026

--device 1457:51ab,:2bc

Work with a device in run-time mode with vendor ID 0x1457 and product ID 0x51ab, or in DFU mode with vendor ID 0x1457 and product ID 0x02bc

--device 1457:51ab

Work with a device in run-time mode with vendor ID 0x1457 and product ID 0x51ab, or in DFU mode with any vendor and product ID

--device ,951:26
Work with a device in DFU mode with vendor ID 0x0951 and product ID 0x0026

--device *,-
Work with any device in run-time mode, and ignore any device in DFU mode

--device ,
Ignore any device in run-time mode, and Work with any device in DFU mode

−p, −−path BUS-PORT. ... .PORT
Specify the path to the DFU device.

−c, −−cfg CONFIG-NR
Specify the configuration of the DFU device. Note that this is only used for matching, the configuration is not set by dfu-util.

−i, −−intf INTF-NR
Specify the DFU interface number.

−a, −−alt ALT
Specify the altsetting of the DFU interface by name or by number.

−S, −−serial [*Run-Time SERIAL*][,[*DFU Mode SERIAL*]]
Specify the run-time and DFU mode serial numbers used to further restrict device matches. If multiple, identical DFU devices are simultaneously connected to a system then vendor and product ID will be insufficient for targeting a single device. In this situation, it may be possible to use this parameter to specify a serial number which also must match.

If only a single serial number is specified, then the same serial number is used in both run-time and DFU mode. An empty serial number will match any serial number in the corresponding mode.

−t, −−transfer-size SIZE
Specify the number of bytes per USB transfer. The optimal value is usually determined automatically so this option is rarely useful. If you need to use this option for a device, please report it as a bug.

−Z, −−upload-size SIZE
Specify the expected upload size, in bytes. Note that the value is only used for scaling the progress bar, the actual upload size is determined by the device.

−U, −−upload FILE
Read firmware from device into **FILE**.

−D, −−download FILE
Write firmware from **FILE** into device. When **FILE** is −, the firmware is read from stdin.

−R, −−reset
Issue USB reset signalling after upload or download has finished.

−e, −−detach
Request that the device re-enumerate out of run-time mode and into DFU mode as when uploading or downloading, but exit immediately after sending the request.

−E, −−detach-delay SECONDS
When uploading or downloading, wait **SECONDS** seconds for the device to re-enumerate after sending the detach request before giving up. Defaults to 5 seconds. This option has no effect with **-e**, since that causes dfu-util to immediately exit after sending the detach request.

−w, −−wait
Wait until matching device appears on the USB bus.

−s, −−dfuse-address [*ADDRESS*][:*LENGTH*][:*MODIFIERS*]
Specify target address for raw binary download/upload on DfuSe devices. Do **not** use this option for downloading DfuSe (.dfu) files. A length can be specified for uploads. Modifiers can be added after the address, separated by a colon, to perform special DfuSE commands such as "leave" DFU mode, "unprotect" and "mass-erase" flash memory. If the device can be expected to reset itself after the operation, "will-reset" should be added. The "force" modifier will override some sanity checks, and is also needed for the "unprotect" and "mass-erase" operations.

−v, −−verbose
Print more information about dfu-util's operation. A second **-v** adds more details. A third **-v** activates verbose logging of USB requests (libusb debug output).

−h, −−help
Show a help text and exit.

−V, −−version
Show version information and exit.

EXAMPLES

Using dfu-util in the OpenMoko project

(with the Neo1973 hardware)

Flashing the rootfs:
$ dfu-util -a rootfs -R -D /path/to/openmoko-devel-image.jffs2

Flashing the kernel:
$ dfu-util -a kernel -R -D /path/to/uImage

Flashing the bootloader:
$ dfu-util -a u-boot -R -D /path/to/u-boot.bin

Copying a kernel into RAM:
$ dfu-util -a 0 -R -D /path/to/uImage

Once this has finished, the kernel will be available at the default load address of 0x32000000 in Neo1973 RAM. **Note:** You cannot transfer more than 2MB of data into RAM using this method.

Using dfu-util with a DfuSe device

Flashing a **.dfu** (special DfuSe format) file to the device:
$ dfu-util -D /path/to/dfuse-image.dfu

Reading out 1 KB of flash starting at address 0x8000000:
$ dfu-util -a 0 -s 0x08000000:1024 -U newfile.bin

Flashing a binary file to address 0x8004000 of device memory and ask the device to leave DFU mode:
$ dfu-util -a 0 -s 0x08004000:leave -D /path/to/image.bin

BUGS

Please report any bugs to the dfu-util bug tracker at **http://sourceforge.net/p/dfu-util/tickets/**. Please use the *--verbose* option (repeated as necessary) to provide more information in your bug report.

SEE ALSO

The dfu-util home page is **http://dfu-util.sourceforge.net/**

HISTORY

dfu-util was originally written for the OpenMoko project by Weston Schmidt <weston_schmidt@yahoo.com> and Harald Welte <hwelte@hmw-consulting.de>. Over time, nearly complete support of DFU 1.0, DFU 1.1 and DfuSe ("1.1a") has been added.

LICENCE

dfu-util is covered by the GNU General Public License (GPL), version 2 or later.

COPYRIGHT