



# Telink

## Application Note

# Telink Burning and Debugging Tool(BDT) CMD User Guide

AN\_22010602-E2

---

Ver 1.0.0

2022.1.6

## Keyword

Linux BDT CMD

## Brief

This document is the development guide for Telink Burning and Debugging(BDT) CMD in Linux.

## Acknowledgements

Published by  
Telink Semiconductor

Bldg 3, 1500 Zuchongzhi Rd,  
Zhangjiang Hi-Tech Park, Shanghai, China

© Telink Semiconductor  
All Right Reserved

## Legal Disclaimer

This document is provided as-is. Telink Semiconductor reserves the right to make improvements without further notice to this document or any products herein. This document may contain technical inaccuracies or typographical errors. Telink Semiconductor disclaims any and all liability for any errors, inaccuracies or incompleteness contained herein.

Copyright © 2021 Telink Semiconductor (Shanghai) Co., Ltd.

## Information

For further information on the technology, product and business term, please contact Telink Semiconductor Company [www.telink-semi.com](http://www.telink-semi.com)

For sales or technical support, please send email to the address of:

[telinknsales@telink-semi.com](mailto:telinknsales@telink-semi.com)

[telinknsupport@telink-semi.com](mailto:telinknsupport@telink-semi.com)

## Revision History

-----		
Version	Change	Description
-----		
V1.0.0		Initial release.
-----		

Telink Semiconductor

## Contents

Acknowledgements	2
Legal Disclaimer . . . . .	2
Information . . . . .	2
Revision History	3
1 Telink BDT CMD User Guide	5
2 Introduce	5
3 Command Example	5
3.1 sws . . . . .	6
3.2 active . . . . .	6
3.3 reset . . . . .	6
3.4 read/write flash . . . . .	7
3.5 read/write core . . . . .	7
3.6 read/wirte analog . . . . .	8
3.7 check pc/parameter . . . . .	8
3.8 run stop start stall . . . . .	8
3.9 step . . . . .	9
3.10 up . . . . .	9
3.11 lsusb . . . . .	9
4 FAQ	9

## 1 Telink BDT CMD User Guide

The tool was developed and tested on Ubuntu 20.04.3 LTS, 64-bit operating system.

Version v1.0.

## 2 Introduce

It mainly supports EVK mode, and some chips support USB mode. They are listed in the following table. They are explained in subsequent detailed use cases.

	evk mode	usb mode
read/write flash (rf, wf)	support	only 826x support
read/write sram (rc, wc)	support	only 82xx support
read/wirte analog (ra, wa)	support	only 82xx support
download in flash	support	not support
download in core	support	only 82xx support
erase in flash/core	support	only 82xx support
check pc (pc)	support	only 82xx support
check global parameters (val)	support	only 82xx support
reset in flash or sram	support	only 82xx support
sws	support	not support
run step stop	support	only 82xx support
ac	support	not support

## 3 Command Example

### Command Options

-u : Indicates usb mode, The **default** mode is EVK.  
-s : The number of bytes read and written, which follows -s. eg: -s 16; -s 1k.  
-e : Erasing, used in Flash and core erasing.  
-i : Specifies the input file followed by the file path, often used to specify the  
↳ download file. eg: -i /home/8258\_gpio.bin.  
-o : Specifies the output file, followed by the file path, often used to save read binary  
↳ data to a file. eg: -o /home/readflash.bin  
-b, -d : Bus and devid of usb devices. This parameter is required when multiple USB  
↳ devices exist.

**Supports the function of USB mode. You can add the -u option after the command.**

If there are multiple EVK devices, the VID and PID of EVK devices are the same. You can control a specified EVK device by specifying its **bus**, **dev**.

If you use usb debugging mode, you also need to specify **bus**, **dev** to control the device.

Example, added after the command. -b:bus -d:dev

```
./bdt 8258 sws -b 1 -d 1
./bdt 8258 sws -b 1 -d 2

./bdt 8258 sws -b 1 -d 1 -u
./bdt 8258 sws -b 1 -d 2 -u
```

### 3.1 sws

Set the rate, and detect whether the EVK and the target board connection is normal.

```
# Sets the specified SWS value.
# b0:address 10:Rate parameter value.The first two (b0 10) are set evK SWire CLK values; The
↪ last two (B0 10) are the target development board swire CLK values.
./bdt 8258 sws b0 10 b0 10

# If no value is specified, the default SWS value is B0 10 b0 10.
./bdt 8258 sws
```

Writing SWS values must be followed by SWS command arguments.

### 3.2 active

Run this command when the program is in low power mode.

```
./bdt 8258 active
```

### 3.3 reset

Restart, the program starts from Flash or SRAM.

```
# Restart the device from the Flash
./bdt 8258 reset

# Restart the device from the Sram
./bdt 8258 reset -c
```

### 3.4 read/write flash

#### read flash(rf)

If the read quantity is less than 1KB, the read data will be printed. Larger than 1KB will be saved to the default file.

Default file name example: save1020-11294102.bin

```
# Read 16 bytes of flash address 0x00
./bdt 8258 rf 0x00 -s 16
./bdt 8258 rf 0x00 -s 1k

# Reads the data output to the specified file
./bdt 8258 rf 0x00 -s 16 -o readflash.bin
```

#### write flash(wf)

flash Erasure is required before writing, and the default unit of erasure is 4K.

```
# Write 4 bytes of data to flash 0x00.
./bdt 8258 wf 0x00 01 02 03 04 -s 4

# Erase first, then write data.
./bdt 8258 wf 0x00 01 02 03 04 -s 4 -e

# Write a file to Flash, download function.
# Write files without the -e and -s option.
./bdt 8258 wf 0x00 -i bin/USB_Demo.bin
```

### 3.5 read/write core

#### read core(rc)

If the read quantity is less than 1KB, the read data will be printed. Larger than 1KB will be saved to the default file.

Default file name example: save1020-11294102.bin

```
# Read 16 bytes of sram address 0x40000
./bdt 8258 rc 0x40000 -s 16
./bdt 8258 rc 0x40000 -s 1k

# Reads the data output to the specified file.
./bdt 8258 rc 0x40000 -s 16 -o readsram.bin
```

#### write core(wc)

```
# Write 4 bytes of data to sram 0x40000
./bdt 8258 wc 0x40000 01 02 03 44 -s 4

# Write a file to sram, download function.
# Write files without the -e and -s option.
./bdt 8258 wc 0x40000 -i bin/USB_Demo.bin
```

### 3.6 read/wirte analog

#### read analog(ra)

```
# Read 16 bytes of analog address 0x40000
./bdt 8258 ra 0x00 -s 16
```

#### write analog(wa)

```
# Write 4 bytes of data to analog 0x00.
./bdt 8258 wa 0x00 01 02 03 44 -s 4
```

### 3.7 check pc/parameter

View the PC pointer value, global parameter list (VAR).

You need to configure the. LST file to view the PC pointer value.

```
# Prints program run pointer.
./bdt 8258 pc

# Print the current PC pointer in detail.
./bdt 8258 pc -i USB_PRINT_LOG.lst

# Prints a list of current program parameters (address, length, value).
/bdt 8258 var -i USB_PRINT_LOG.lst
```

### 3.8 run stop start stall

Run, stop the program.

```
./bdt 8258 run
./bdt 8258 stop
```

start, stall the program



```
./bdt 8258 start  
./bdt 8258 stall
```

### 3.9 step

Step through the program.

```
./bdt 8258 step
```

### 3.10 up

Update EVK firmware

-i : Specifies the firmware file path to update.

-v : Query evk version number

```
# The chip used by burning evk is 8266  
./bdt 8266 up -i fw/Firmware_v3.4.bin  
./bdt 8266 up -i fw/Firmware_v3.4.bin -v
```

### 3.11 lsusb

List connected USB devices.

```
./bdt lsusb  
  
# -v : View usb descriptors  
./bdt lsusb -v
```

## 4 FAQ