

BIN2TDF

```
{{#switchcategory:MSP430=<McuHitboxHeader/>|C2000=<McuHitboxHeader/>|Stellaris=<McuHitboxHeader/>|TMS570=<McuHitboxHeader/>|MCU=<McuHitboxHeader/>|MAVRK=<MAVRKHitboxHeader/>|<HitboxHeader/>}}
```

Contents

Introduction

Location

Parameters

Required

Required for STM (Optional for DSP and ARM/Cortex trace)

Optional

Examples

Related

Introduction

This page describes usage of BIN2TDF utility. BIN2TDF utility is delivered as part of Code Composer Studio (CCS) and used to convert field captured hardware trace data into a format (.tdf) that can be imported in standalone CCS Trace Visualizer.

Location

This utility gets installed in CCS_DIR/ccsv5/ccs_base/emulation/analysis/bin directory as part of Code Composer Studio installation. To use this utility, open a command line shell and change directory to CCS_DIR/ccsv5/ccs_base/emulation/analysis/bin.

Parameters

Required

-bin <Full path to binary file>

This parameter is used to specify the trace binary file.

-app <Full path to out file>

This parameter is used to specify full path of the application location. Trace decoders use application information to decompress and output the trace information.

If using software messages from STM, this parameter needs to be specified with CPU qualification (i.e. *CPU=file*). More than one of such entries can be specified with comma(,) separations e.g. CPU0=file0,cpu1=file1,...

It is sufficient to give just the name after the slash (e.g., C66XX_o) as CPU name, it is not required to use full CPU name as formed in CCS (e.g., Blackhawk XDS560v2-USB Mexxanine Emulator_o/C66XX_o).

If there is only one processor generating STM SW messages, CPU qualification could be skipped

*Note: If you are using **CCSV4**, this parameter name would be **-outfile**.*

-procid <proc>

This parameter is used to specify processor or trace type. Valid options - 64x, 64x+, 66x, 55x, 55x+, arm7, arm9, arm11, cortex8, and stm.

Note: If you are using CCSv4, this parameter name would be **-cpuid**.

-revr <receiver name>

This parameter is used to specify receiver type used for trace collection. Valid options - ETB, 560T, 560V2, and Pro

Required for STM (Optional for DSP and ARM/Cortex trace)

-devicekey DeviceID

This parameter is used to define the device ID or device key for identifying STM topology in the device. This is a unique ID (TAP ID) and can be read via Ro of ICEPICK or retrieved from the device TRM.

Examples:

Device id for

OMAP4430 ES2 — 0x1B85202F

```

OMAP4430 ES2.1 -- 0x3B95C02F
OMAP4460      -- 0x0B94E02F
OMAP4470      -- 0x0B97502F
C6678         -- 0x0009E02F
C6670         -- 0x0009D02F
Turbo-C6670/TCI6618 -- 0x0b94102F
TCI6614/2     -- 0x0B96202F
C6657/5/4     -- 0xB97A02F
AM335x        -- 0x0B94402F
AM386x        -- 0x0B96802F
AM387x        -- 0x0B8F202F
DM816x        -- 0x0B81E02F

```

Note: If you are using CCSv4, this parameter name would be **-deviceid**.

-dcmfile <Full path to the metadata file>

This parameter is used to pass additional metadata to help decode. Default is none.

For STM ETB, you could create a *foo.dcm* configuration file by copying the text in the box below into a text file and saving it with a name like *foo.dcm* or whatever you prefer.

```

STM_data_flip=1
STM_Buffer_Wrapped=0
HEAD_Present_0=0
HEAD_Pointer_0=
HEAD_Present_1=1
HEAD_Pointer_1=4

```

STM_data_flip=1 --> This field tells decoder that the data is flipped in receiver packing. This is 1 if using ET. Otherwise, its 0.

STM_Buffer_Wrapped=0 --> This field tells decoder that the buffer is wrapped or not. 1 is wrapped, 0 for buffer not wrapped.

HEAD_Present_0=0 --> STM ETB pointer references to efficiently detect start of the buffer (if buffer wrapped). Default to 0 unless you know the HEAD information.

HEAD_Pointer_0=0 --> STM ETB pointer references to efficiently detect start of the buffer (if buffer wrapped). Default to 0 unless you know the HEAD information.

HEAD_Present_1=0 --> STM ETB pointer references to efficiently detect start of the buffer (if buffer wrapped). Default to 0 unless you know the HEAD information.

HEAD_Pointer_1=0 --> STM ETB pointer references to efficiently detect start of the buffer (if buffer wrapped). Default to 0 unless you know the HEAD information.

Optional

-output <TDFFileName>

This parameter is used to specify output TDF file name. Default is <BinaryFileName>.tdf.

Note: If you are using CCSv4, this parameter name would be **-tdf**.

-sourcepaths <"Source paths">

This parameter is used to specify source path location to help Trace Visualizer locate the source files. Multiple paths can be specified with comma seperated string. The Trace Visualizer automatically searches in the sub-directories. Default is empty source path string.

Examples

DSP Trace

```
bin2tdf -bin C:/temp/etbdata.bin -app C:/Examples/bin/csetbexample.tci6488.out -procid 64x+ -rcvr ETB -output C:/Examples/bin/mytrace.tdf
```

ARM Trace

```
bin2tdf.exe -bin c:/temp/ETM_etbdata.bin -app C:/Examples/bin/etm_csetb_A8.c6a816x.out -procid cortexa8 -rcvr ETB -dcmfile c:/temp/ETM_etbdata.dcm -output C:/Examples/bin/etm_csetb_A8.c6a816x.tdf
```

Note: If the device has more than one ARM core, the -cpuname parameter will need to be specified.

```
bin2tdf.exe -bin c:/temp/ETM_etbdata.bin -app C:/Examples/bin/ptm_tietb_d_A15.c66ak2hxx.out -procid cortexa15 -rcvr ETB -dcmfile c:/temp/ETM_etbdata.dcm -output C:/temp/ptm_tietb_d_A15.c66ak2hxx.tdf -cpuname CortexA15_o
```

System Trace

```
bin2tdf -bin C:/temp/etbdata.bin -app C66X_o=C:/Examples/bin/tietbexample_d.c66xxstm.out -procid stm -rcvr ETB -devicekey 0x0009D02F -dcmfile C:/temp/foo.dcm -output C:/Examples/bin/mytrace.tdf
```

```
bin2tdf -bin C:/temp/etbdata.bin -app CortexA8_o=C:/Examples/bin/csetbexample_d.ti816xstma8.out -procid stm -rcvr ETB -devicekey 0x0B81E02F -dcmfile C:/temp/foo.dcm -output C:/Examples/bin/mytrace.tdf
```

Related

- CToolsLib
- Embedded Trace Buffer or ETB

1. switchcategory:MultiCore=

- For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum
- For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum

Please post only comments related to the article **BIN2TDF** here.

Keystone=

- For technical support on MultiCore devices, please post your questions in the C6000 MultiCore Forum
- For questions related to the BIOS MultiCore SDK (MCSDK), please use the BIOS Forum

Please post only comments related to the article **BIN2TDF** here.

C2000=For technical support on the C2000 please post your questions on The C2000 Forum. Please post only comments about the article **BIN2TDF** here.

DaVinci=For technical support on DaVincoplease post your questions on The DaVinci Forum. Please post only comments about the article **BIN2TDF** here.

MSP430=For technical support on MSP430 please post your questions on The MSP430 Forum. Please post only comments about the article **BIN2TDF** here.

OMAP35x=For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **BIN2TDF** here.

OMAPL1=For technical support on OMAP please post your questions on The OMAP Forum. Please post only comments about the article **BIN2TDF** here.

MAVRK=For technical support on MAVRK please post your questions on The MAVRK Toolbox Forum. Please post only comments about the article **BIN2TDF** here.}}

For technical support please post your questions at <http://e2e.ti.com> Please post on comments about article **BIN2TDF**

Links

[Amplifiers & Linear Audio](#)
[Broadband RF/IF & Digital Radio](#)
[Clocks & Timers](#)
[Data Converters](#)

[DLP & MEMS High-Reliability Interface](#)
[Logic](#)
[Power Management](#)

[Processors](#)

- ARM Processors
- Digital Signal Processors (DSP)
- Microcontrollers (MCU)
- OMAP Applications Processors

[Switches & Multiplexers](#)
[Temperature Sensors & Control ICs](#)
[Wireless Connectivity](#)

```
{{#switchcategory:MSP430=<McuiHitboxFooter/>|C2000=<McuiHitboxFooter/>|Stellaris=<McuiHitboxFooter/>|TMS570=<McuiHitboxFooter/>|MCU=<McuiHitboxFooter/>|MAVRK=<MAVRKHitboxFooter/>|<HitboxFooter/>}}
```

Retrieved from "<https://processors.wiki.ti.com/index.php?title=BIN2TDF&oldid=166119>"

This page was last edited on 13 December 2013, at 13:07.
Content is available under [Creative Commons Attribution-ShareAlike](#) unless otherwise noted.

<https://processors.wiki.ti.com/index.php/BIN2TDF>

3/3