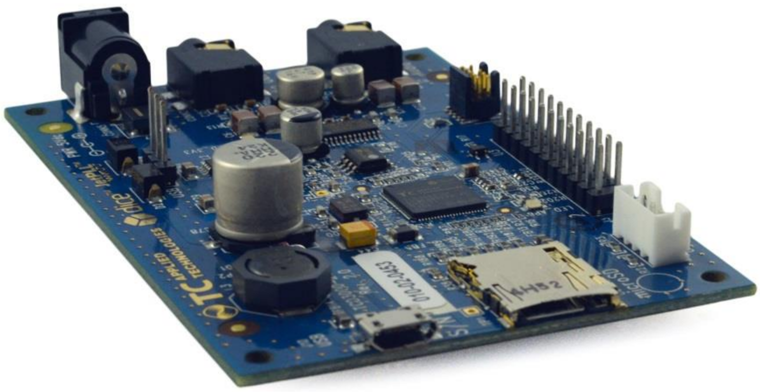
TCD3000-EVM001 Evaluation Module User Guide

Board Revision 2.0



dice™

April 9, 2015

Document Revision 1.0.1-2672

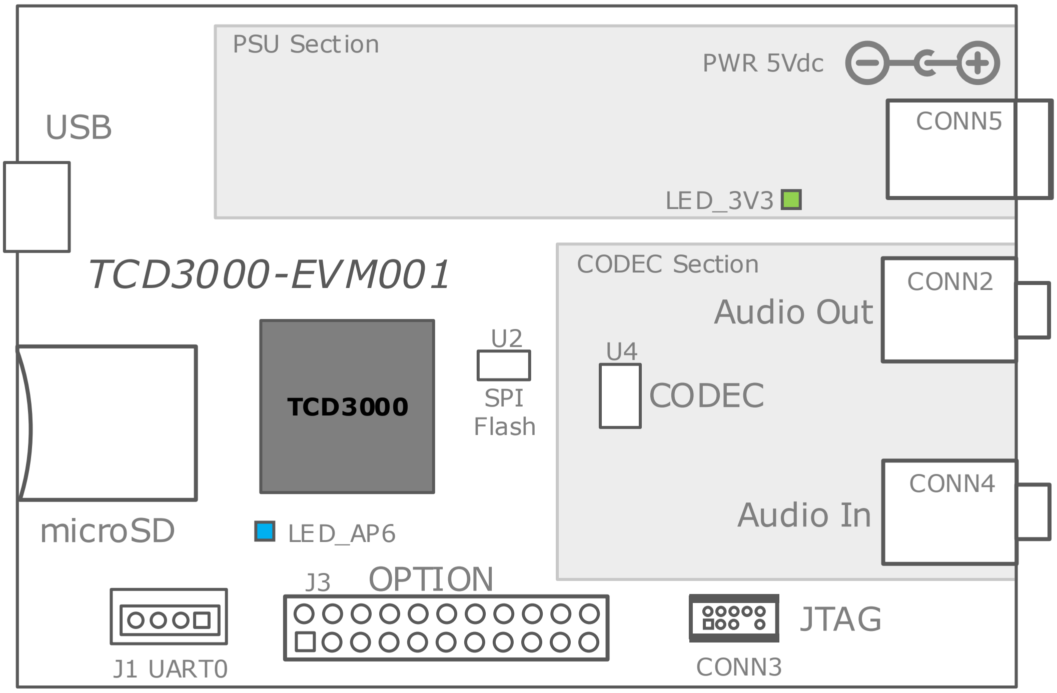
Contents

1 Overview

The TCD3000-EVM001 Evaluation Module (EVM) demonstrates a simple USB interface design using the DICE III TCD3000 IC. It implements a Class Compliant USB 2.0 Audio Interface device, and allows additional functions to be prototyped through an option expansion header. Using this EVM, the DICE III performance and programming can be evaluated and customer devices can be prototyped ahead of custom hardware.

The EVM has the following features:

- TCD3000
- SPI Flash, 16Mbit
- USB (micro-B) Connector
- USB bus power or 5V DC input power
- Stereo Codec
- 1/8” Audio input mini jack
- 1/8” Audio output mini jack
- Serial port header
- JTAG header (10-pin)
- LED (PWM programmable)
- SD CARD connector (microSD)
- 22-pin Option Header



The option header routes additional Audio/GPIO, MIDI, and two channels of ADC (i.e. for trim pots or other sensors) to the TCD3000. It also provides 3.3Vdc power and access to the 5V USB bus power.

2 Features

The TCD3000-EVM001 Rev 2.0 is a 4-layer design, small footprint board which can be USB bus-powered or externally powered. The various functional sections are described below.

2.1 PSU

The PSU can use 5V USB Bus power or power from the DC connector. Input power should not exceed 6 Volts. Power is regulated to 1.2V and 3.3V, with the 3.3V supply indicated by the green LED_3V3.

2.1.1 CONN5 – DC Input

CONN5 is a DC power input jack. It is center-positive, with a 2.0 mm center pin. Supplied input should not exceed 6V.

2.2 Reset

2.2.1 Voltage Supervisor

Board power is monitored by a voltage supervisor circuit, where the nCPU_RESET monitor output is routed to the TCD3000 nRESET signal.

2.2.2 JTAG Reset

The nRESET pin on the JTAG connector is also routed to the TCD3000 so the CPU can be reset either if the power supply drops or from the JTAG interface.

2.2.3 Peripheral Reset

The TCD3000 APORT4 signal is routed to the CODEC Power Down pin, which should put the device into minimum power consumption mode (less than 100mA when not configured).

2.3 CODEC

A Cirrus Logic CS4270-CZZ 24-Bit, 192KHz Stereo Audio CODEC is used. SPI0 on the TCD3000 is used for software mode configuration.

2.4 LEDs

2.4.1 Power status

LED_3V3 is connected to the reset supervisor and illuminates when stable power is applied through the DC connector CONN5 or by way of USB bus power.

2.4.2 Programmable/PWM

LED_AP6 is connected to APORT6 of the TDC3000 and is software programmed. The LED is used to indicate software state and as a demonstration of the PWM function of the chip.

2.5 Connectors

2.5.1 USB

A standard 5-pin USB micro B receptacle is used. The DP/DM signals connect directly to TCD3000 and the USB_VBUS bus power is routed to the PSU and option header.

2.5.2 Serial

The TCD3000 UART0 signals are connected to the 4-pin J1 connector. The pinout and connector are standard on all DICE III EVMs.

Table 2.1 J1 - UART0 connector pin assignments

PIN	Signal name	Description
1	GND	
2	+3.3V	
3	UART_TX	UART0 TX
4	UART_RX	UART0 RX

This can be connected to any RS-232 interface using a serial level-shifter device. The included Serial/USB cable can also be used. The cable connects to UART0 on the EVM and to a USB port on your Host computer, where it appears as a standard serial port. The cable has built-in support on most Host operating systems. In serial terminal software, connection settings are 115200 baud, 8 data bits, no parity, one stop-bit and no hardware handshaking.

Please contact TC Applied Technologies for additional cables.

2.5.3 JTAG

The TCD3000 JTAG signals are connected to the 10-pin CONN3 connector. The connector is a standard Cortex ARM 10-pin micro header (0.05”). For JTAG Host interfacing, a cable adapter such as the SEGGER “J-Link Adapter Cortex-M” is used.

Table 2.2 CONN3 - JTAG connector pin assignments

PIN	Signal name	Description
1	VTref	+3.3V
2	JTAG_TMS	
3	GND	
4	JTAG_TCK	
5	GND	
6	JTAG_TDO	
7	N/U	Key pin, should be removed
8	JTAG_TDI	
9	GND	
10	nRESET	

2.5.4 SD Card

CONN8 is a standard SD card microSD receptacle. Signals from this connector are routed directly to the TCD3000. This interface is intended for expansion and is not required for normal use of the EVM.

2.5.5 Option Header – J3 - Pin assignments

J3 is a 22-pin through-hole header (0.1”). This header allows feature expansion.

Table 2.3 Option Header – J3 - Pin assignments

PIN	Signal name	Description
1	GND	
2	APORT0	MIDI TX, or I2C SCL (i.e. for MFI)
3	APORT1	MIDI RX, or I2C SDA (i.e. for MFI)
4	VDD +3.3V	
5	APORT2	InS Out (CODEC SDIN)
6	APORT3	InS In (CODEC SDOUT)
7	GND	
8	APORT4	PSU_POWER_ON
9	APORT5	Unassigned
10	APORT6	LED_AP6 (Blue)
11	APORT7	Chip select for CODEC SPI
12	SPI0_CLK	
13	SPI0_MOSI	
14	SPI0_MISO	
15	GND	
16	BCLK0	
17	FCLK0	
18	MCLK0	
19	GND	
20	VDD +5V, USB_VBUS	5V power from the USB Port
21	ADC Input 1	DICE III ADC input 1
22	ADC Input 2	DICE III ADC input 2

2.5.6 Audio I/O

CONN4 is a 1/8” stereo mini jack, line level audio input.

CONN5 is a 1/8” stereo mini jack, line level audio output.

3 CLI

3.1 Overview

The SDK includes an extensible Command Line Interpreter (CLI), which is used on UART0 by default. Serial settings are 115200 8-N-1, with no hardware flow-control. TC Applied Technologies EVMs include a USB Serial adapter cable for this purpose. The CLI is covered in more detail elsewhere, and for evaluation purposes, using the CLI to update firmware is covered here.

There are a number of built-in commands and a number of commands which are optionally installed by each module at compile time. Once connected with a serial terminal program, you can see your application info by issuing the splash command.

```
> splash
**********************************************************
* TC Command Line Interface System                       *
* Copyright 2004-2015 TC Applied Technologies            *
**********************************************************
* Organization    : TCAT                                 *
* Product         : TCD3000_EVM001                       *
* Serial no.      : 124                                  *
* SDK Version     : 1.0.0.0 local beta                   *
* Application Ver.: 0.9.0.0                              *
*                   - built on 11:15:02, Mar 10 2015     *
**********************************************************
>
```

To find out which commands are available, enter a single question mark.

```
>?
[Root]:
------------------------------------------------------------
[fss]       [midi]      [dice]      [aesrx]
[ins]       [hpll]      [mix]       [dal]
[kernel]    [usb]

get                 (built-in)
set                 (built-in)
setm                (built-in)
gms                 (built-in)
dump                (built-in)
splash              (built-in)
exit                (built-in)
reset               (built-in)
echo                (built-in)
prompt              (built-in)
>
```

Commands that have categories are listed with square brackets. To list the commands for a category, enter the category name.

```
>fss
[fss]:
------------------------------------------------------------
list                (Dump FSS Information)
erase               (Erase FSS segment)
load                (Load FSS segment)
dump                (Reading and displaying flash memory)
>
```

To show usage for a command, enter the category and command separated by a period or colon and then a question mark.

```
>fss.load ?
------------------------------------------------------------
fss.load <name> <erase> <verify>
Load a segment in the FSS via X-modem-1k
>
```

Note that some commands are not intended to be used to configure normal operation of your device and host as system, but rather they are there for specific debugging scenarios that most developers will not encounter. In many cases normal operation is accomplished by interaction with the host driver and protocol state machines, so be aware of this when experimenting.

3.2 Updating Firmware

Firmware images are stored using a fault tolerant, redundant method called BOOT2. The on-chip boot program will look for a program at address zero. That program is in a segment reserved as the 'boot' segment. The boot program looks for a valid image in the App_Upgrade segment and loads it. If App_Upgrade is not valid it looks for a valid application in App_Gold. Therefore you will update the App_Upgrade flash memory segment in your device. The boot and App_Gold segments are usually programmed during factory production. Developers may program all segments of course. Use of the various segments is discussed elsewhere.

```
> fss.list
===========================================================
FSS Structure:
Flash Base:      0x00000000
Flash Size:      0x000e0000
Flash Segments:  5
Flash Prg. Time: 2 ms/64kB
Flash Ers. Time: 1000 ms/64kB
===========================================================
Segment[0]: boot
Base Address: 0x00000000, Size: 0x00010000 Good
Flags: 0x00000009
Segment[1]: App_Gold
Base Address: 0x00010000, Size: 0x00050000 Good
Flags: 0x0000000c
Segment[2]: App_Upgrade
Base Address: 0x00060000, Size: 0x00050000 Good
Flags: 0x00000002
Segment[3]: App_Env
Base Address: 0x000b0000, Size: 0x00010000 Good
Flags: 0x00000040
Segment[4]: App_Settings
Base Address: 0x000c0000, Size: 0x00020000 Good
Flags: 0x00000030
===========================================================
>
```

```
>
>fss.load App_Upgrade yes yes
Erasing flash...
start XModem-1k transfer ...
CCCC
Loaded 169572 bytes
>
```

Once you start seeing the ‘C’ characters, you can start the transfer using the X-Modem 1K protocol. The binary file to transfer must be a ‘TCA’ image, which is a firmware binary that has a BOOT2 header prepended and will normally have a `.tca` file extension. If the transfer was successful, you’ll see a message about the number of bytes that were received, and you can reset the board.

The firmware X-Modem protocol will timeout in approximately 45 seconds if a transfer has not started, at which time you’ll have an unprogrammed App_Upgrade segment.

```
>fss.load App_Upgrade yes yes
Erasing flash...
start XModem-1k transfer ...
CCCCCCCCCCCCCCCCCCCCCXModem error: Timed out
CLI ERROR: The tool "load" returned the error 0xff000007 - E_FAIL
>
```

At that point, you can repeat `fss.load` to try again. If you restart with the unprogrammed App_Upgrage segment, it will boot from the App_Gold segment. You can see which segment was loaded in the initial splash screen when the board is booted.

```
>reset
**********************************************************
* Secondary Boot - ver 0.9.0                             *
* Application Invalid, found Golden - execute            *
**********************************************************

**********************************************************
* TC Command Line Interface System                       *
* Copyright 2004-2014 TC Applied Technologies            *
**********************************************************
* Organization    : TC Applied Technologies              *
* Product         : TCD3000 EVM001                       *
* Serial no.      : 124                                  *
* SDK Version     : 1.0.0.0 local beta                   *
* Application Ver.: 0.9.0.0                              *
*                   - built on 13:55:08, Mar 11 2015     *
**********************************************************
```

```
>fss.load App_Upgrade yes yes
Erasing flash...
start XModem-1k transfer ...
CCC
Loaded 169572 bytes
>reset
**********************************************************
* Secondary Boot - ver 0.9.0                             *
* Application found - execute                            *
**********************************************************

**********************************************************
* TC Command Line Interface System                       *
* Copyright 2004-2014 TC Applied Technologies            *
**********************************************************
* Organization    : TC Applied Technologies              *
* Product         : tcd3xxx_usb                          *
* Serial no.      : 124                                  *
* SDK Version     : 1.0.0.0 local beta                   *
* Application Ver.: 0.9.0.0                              *
*                   - built on 13:06:56, Mar 12 2015     *
**********************************************************
>
```

As shown above, the Upgrade segment is now programmed. As mentioned before, it isn’t necessary to update the App_Gold or boot segments, but if you try this and find that you have corrupted these segments, you can recover the board using a JTAG connection. This is covered in a separate document.

4 Resources

Design Files (schematic, BOM, layout) for this and other EVMs are available from TC Applied Technologies.

Firmware binaries are also available as well as information about developing firmware for the TCD3xxx family, and the complete firmware source code tree (SDK). TC Applied Technologies provides Host Drivers (built with vendor IDs and strings) for Windows and Mac OS X where applicable, including installers and brand-able Control Panels.

5 Document History

Table 5.1 Document revision history

Date	Rev.	By	Change
April 9, 2015	1.0.1-2672	BK	Initial release