DVSDK Getting Started Guide

Literature Number: SPRUEG8
August 2006

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products	
Amplifiers	amplifier.ti.com
Data Converters	dataconverter.ti.com
DSP	dsp.ti.com
Interface	interface.ti.com
Logic	logic.ti.com
Power Mgmt	power.ti.com
Microcontrollers	microcontroller.ti.com
Low Power Wireless	www.ti.com/lpw

Applications	
Audio	www.ti.com/audio
Automotive	www.ti.com/automotive
Broadband	www.ti.com/broadband
Digital Control	www.ti.com/digitalcontrol
Military	www.ti.com/military
Optical Networking	www.ti.com/opticalnetwork
Security	www.ti.com/security
Telephony	www.ti.com/telephony
Video & Imaging	www.ti.com/video
Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

EVALUATION BOARD/KIT IMPORTANT NOTICE

Texas Instruments (TI) provides the enclosed product(s) under the following conditions:

This evaluation board/kit is intended for use for **ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY** and is not considered by TI to be a finished end-product fit for general consumer use. Persons handling the product(s) must have electronics training and observe good engineering practice standards. As such, the goods being provided are not intended to be complete in terms of required design-, marketing-, and/or manufacturing-related protective considerations, including product safety and environmental measures typically found in end products that incorporate such semiconductor components or circuit boards. This evaluation board/kit does not fall within the scope of the European Union directives regarding electromagnetic compatibility, restricted substances (RoHS), recycling (WEEE), FCC, CE or UL, and therefore may not meet the technical requirements of these directives or other related directives.

Should this evaluation board/kit not meet the specifications indicated in the User's Guide, the board/kit may be returned within 30 days from the date of delivery for a full refund. THE FOREGOING WARRANTY IS THE EXCLUSIVE WARRANTY MADE BY SELLER TO BUYER AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE.

The user assumes all responsibility and liability for proper and safe handling of the goods. Further, the user indemnifies TI from all claims arising from the handling or use of the goods. Due to the open construction of the product, it is the user's responsibility to take any and all appropriate precautions with regard to electrostatic discharge.

EXCEPT TO THE EXTENT OF THE INDEMNITY SET FORTH ABOVE, NEITHER PARTY SHALL BE LIABLE TO THE OTHER FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

TI currently deals with a variety of customers for products, and therefore our arrangement with the user **is not exclusive**.

TI assumes **no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein**.

Please read the User's Guide and, specifically, the Warnings and Restrictions notice in the User's Guide prior to handling the product. This notice contains important safety information about temperatures and voltages. For additional information on TI's environmental and/or safety programs, please contact the TI application engineer or visit www.ti.com/esh.

No license is granted under any patent right or other intellectual property right of TI covering or relating to any machine, process, or combination in which such TI products or services might be or are used.

Mailing Address:
Texas Instruments
Post Office Box 655303
Dallas, Texas 75265

FCC Warning

This evaluation board/kit is intended for use for **ENGINEERING DEVELOPMENT, DEMONSTRATION, OR EVALUATION PURPOSES ONLY** and is not considered by TI to be a finished end-product fit for general consumer use. It generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment in other environments may cause interference with radio communications, in which case the user at his own expense will be required to take whatever measures may be required to correct this interference.

Preface

About This Guide

The DVSDK software allows you to develop DSP-side applications for the DVEVM board in addition to the ARM-side development supported by the DVEVM software. In addition, the DVSDK provides a full MontaVista Linux license, including support.

The DVEVM (Digital Video Evaluation Module) is an evaluation platform that showcases the DM644x architecture and lets users evaluate the power and performance of the DM644x as a multimedia engine.

This guide assumes that you have already set up the hardware and installed the DVEVM software as described in the *DVEVM Getting Started Guide* (SPRUE66). You install the DVSDK software on top of the DVEVM software, without deinstalling anything in the DVEVM kit.

This document is intended to be used as the initial "getting to know you" document for the DVSDK software. Other documents provide more in-depth information.

Additional Documents and Resources

You can use the following sources to supplement this user's guide. Additional documents are provided with the various DVSDK components in the software installation.

- *DVEVM Getting Started Guide* (SPRUE66)
- DaVinci EVM Home at Spectrum Digital: http://c6000.spectrumdigital.com/davincievm/
- TI Linux Community for DaVinci Processors: http://linux.davincidsp.com
- *Codec Engine Application Developer User's Guide* (SPRUE67)
- *Codec Engine Server Integrator User's Guide* (SPRUED5)
- *Codec Engine Algorithm Creator User's Guide* (SPRUED6)
- *TMS320C6000 DSP/BIOS API Reference* (SPRU403)
- *TMS320C6000 Optimizing C Compiler User's Guide* (SPRU187)
- *TMS320C6000 Programmer's Guide* (SPRU189)

Notational Conventions

This document uses the following conventions:

- Program listings, program examples, and interactive displays are shown in a `mono-spaced font`. Examples use **`bold`** for emphasis, and interactive displays use **`bold`** to distinguish commands that you enter from items that the system displays (such as prompts, command output, error messages, etc.).
- Square brackets ([and]) identify an optional parameter. If you use an optional parameter, you specify the information within the brackets. Unless the square brackets are in a **bold** typeface, do not enter the brackets themselves.

Trademarks

Contents

1 DVSDK Overview 1-1
This chapter introduces the DVSDK (Digital Video Software Development Kit).
1.1 Welcome! 1-2
1.2 What's in this Kit? 1-2
1.2.1 What About Encoders and Decoders? 1-4
1.3 Software Architecture 1-5
1.4 What's Next? 1-5

2 DVSDK Setup 2-1
This chapter explains how to upgrade the DVEVM to a DVSDK installation.
2.1 Introduction 2-2
2.1.1 Command Prompts in This Guide 2-2
2.2 Installation 2-3
2.3 Setting Up the Build/Development Environment 2-5
2.3.1 Testing the Build Environment 2-5
2.4 Using DVSDK Software for DSP-Side Development 2-6
2.4.1 Using the Digital Video Test Bench (DVTB) 2-7

A Additional Procedures A-1
This appendix describes optional procedures you may use depending on your setup and specific needs.
A.1 Installing Components Under Windows A-2

Chapter 1

DVSDK Overview

This chapter introduces the DVSDK (Digital Video Software Development Kit).

Topic **Page**

1.1 Welcome! 1–2
1.2 What's in this Kit? 1–2
1.3 Software Architecture 1–5
1.4 What's Next? 1–5

1.1 Welcome!

By adding the DVSDK (Digital Video Software Development Kit) to your DVEVM, you will be able to develop complete DSP-side and ARM-side applications that make the most of TI's new DaVinci™ Technology and the DM644x architecture.

This guide assumes that you have already set up the hardware and installed the DVEVM software as described in the *DVEVM Getting Started Guide* (SPRUE66). You install the DVSDK software on top of the DVEVM software, without deinstalling anything in the DVEVM kit.

This document is intended to be used as the initial "getting to know you" document for the DVSDK software. Other documents provide more in-depth information.

1.2 What's in this Kit?

This document assumes you have the complete DVSDK kit. It is possible to download and install some of the components of the DVSDK on their own, but this document describes the kit as a whole.

The main components in the DVSDK kit are the following software products:

- **MontaVista Linux Professional Edition v4.** This is the complete, licensed version of MontaVista Linux Pro. The version provided with the DVEVM is a demonstration version. In contrast to the demo version provided with the DVEVM, the complete version includes the DevRocket™ IDE, the Professional Edition documentation, and customer support. See the Quick Start Guide in the MontaVista Linux Pro CD/DVD case for installation and use instructions.
- **DM644x SoC Analyzer (DSA).** This software installs on a Windows host to allow you to collect profiling data from a running system and visually identify situations such as improper load balancing, contention for shared resources, and bottlenecks. It displays information of use to both ARM and DSP developers and to system integrators. See the Quick Start Guide in the DM644x SoC Analyzer CD case for installation and use instructions.

- **TI DVSDK Software.** This CD includes the following sub-components. See the next chapter for installation instructions.
  - **DSP/BIOS for Linux.** DSP/BIOS is a a scalable real-time DSP kernel. It is designed for applications that require real-time scheduling and synchronization, host-to-target communication, or real-time instrumentation. DSP/BIOS provides preemptive multi-threading, hardware abstraction, and real-time analysis. The version provided is DSP/BIOS 5.30.00 for Linux. The Linux version does not include the graphical DSP/BIOS Configuration Tool and the graphical Real-Time Analysis (RTA) tools that are integrated with Code Composer Studio on Microsoft Windows. With the Linux version, you can use a text editor to write configuration scripts and build DSP/BIOS applications.
  - **TI Codegen Tools for Linux.** These are the compilers, linkers, and related build tools for TI DSPs.
  - **Framework Components.** These DSP-side modules add support for managing xDAIS-compliant algorithms, allocating memory and DMA resources and ensuring that those resources can be shared as efficiently (and correctly) as possible. The Framework Components are used internally by the Codec Engine and can also be used directly by DSP programmers. See framework_components_#_##/docs/html/index.html for web-based documentation.
  - **Digital Video Test Bench (DVTB).** This ARM-side application enables script-based (or command-line interactive) exercising of DSP codecs. Using this application from a Linux shell on the ARM processor, you can quickly encode and decode audio and video streams that are either file-based or real-time I/O device-based. For example, a file containing H.264-encoded video frames can be decoded and displayed on the Linux framebuffer with only a few lines of script. You do not have to write any C code that deals with Linux I/O, codec APIs, or threading issues. See the dvtb/docs directory for a preliminary user's guide. In addition, see additional information in Section 2.4.1, *Using the Digital Video Test Bench (DVTB)*.
- **Code Composer Studio and Emulator.** If you purchased the "L3" version of the DVSDK, your kit also includes CDs for Code Composer Studio (CCStudio) and an XDS560 JTAG scan-based emulator for an advanced host-target connection that is fully integrated with CCStudio. CCStudio is a Windows-based development environment for DSP applications.

1.2.1 What About Encoders and Decoders?

The Codec Engine software installed with the DVEVM software includes example codec servers that contain a number of encoders and decoders. These are pre-built DSP-side applications.

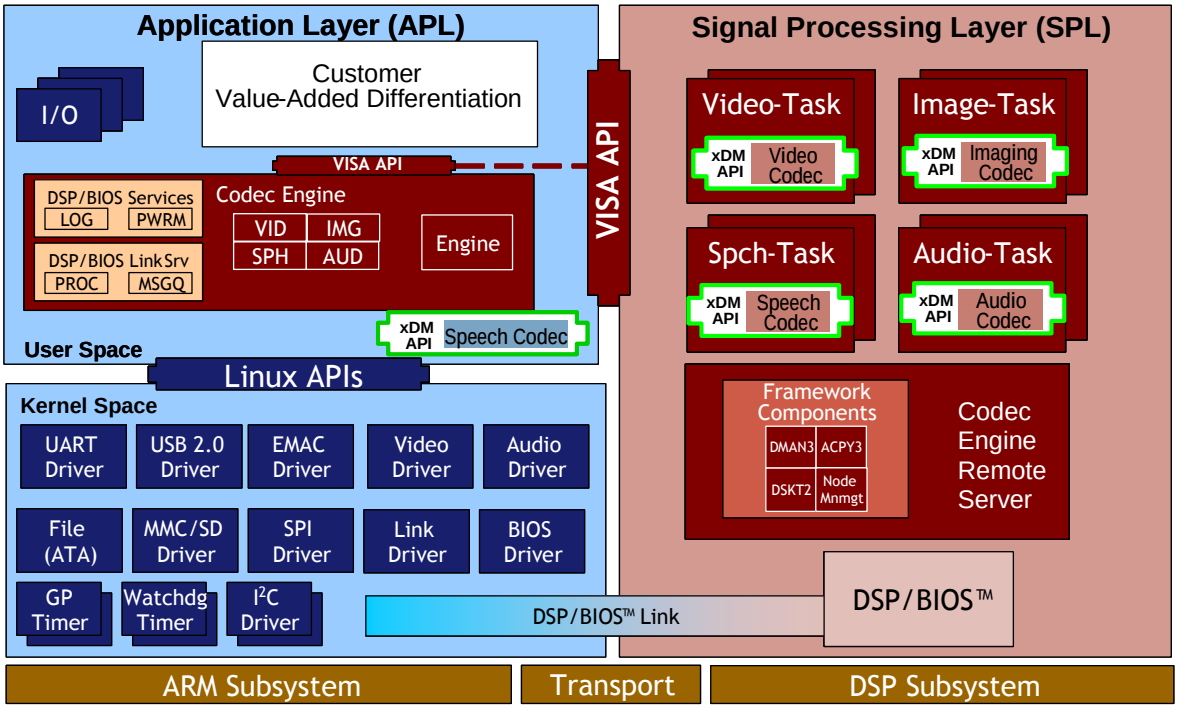
The DVEVM and DVSDK do not include the separate encoder and decoder libraries used to create these servers. Instead, the DVEVM software provides the tools and supporting components required to access a codec server from your GPP-side application. The DVSDK software provides the tools to create your own codec servers using xDM-compliant or xDAIS-compliant algorithms.

The encoders and decoders used in the pre-built codec servers are available through TI's ASP program for a free 60-day evaluation, with up to 4 hours of support from an authorized software provider (ASP) of your choice. Production licenses are available from the same ASPs, with up to 40 hours of support with a signed license contract. A TI sales representative or applications engineer can enter a request for a free evaluation for any customer. You can request more information using the "Contact Me" feature on the TI Digital Media Software website, which is located at www.ti.com/digitalmediasoftware.

Additional encoders, decoders, and codecs developed by TI 3rd parties are also available. For an encoder, decoder, or codec to be supported by the Codec Engine, it should be compliant with the xDM (eXpressDSP Digital Media) algorithm standard and should include the appropriate XDC packaging required by the eXpressDSP configuration kit's scripting utilities. For more details on available 3rd party algorithms, consult the TI website.

1.3 Software Architecture

Installing the TI DVSDK software allows you to modify the items on the DSP Subsystem side of this component architecture diagram:



A DSP/BIOS application runs the Codec Engine Remote Server, which runs threads for any signal processing to be performed on the DSP. A number of xDAIS framework components are used. Communication between the GPP and DSP is managed by DSP/BIOS Link.

1.4 What's Next?

Follow the instructions in the Quick Start Guides for MontaVista Linux Pro and the DM644x SoC Analyzer to install and use those products.

See Chapter 2 for instructions on installing and setting up the software on the TI DVSDK CD. See the documents installed with each product components for instructions on using those components.

Chapter 2

DVSDK Setup

This chapter explains how to upgrade the DVEVM to a DVSDK installation.

Topic **Page**

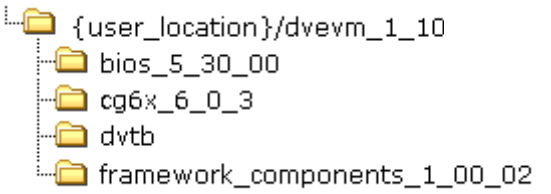
2.1 Introduction 2–2
2.2 Installation 2–3
2.3 Setting Up the Build/Development Environment 2–5
2.4 Using DVSDK Software for DSP-Side Development 2–6

2.1 Introduction

This guide assumes that you have already set up the DVEVM hardware and installed the DVEVM software as described in the *DVEVM Getting Started Guide* (SPRUE66). You must have already installed the **same version** of the DVEVM software as the DVSDK software you will install on top. For example, if you want to install the DVSDK 1.10 software, you must first install the DVEVM 1.10 software.

Use the steps in this chapter to install the TI DVSDK software on top of the DVEVM software, without deinstalling anything in the DVEVM kit.

When you install the DVSDK software, the following subdirectories are typically added to the base directory (~/dvevm_#_##) of the DaVinci DVEVM installation (the version numbers may differ):



Note: The installers for DSP/BIOS and Code Generation Tools (codegen) have a different default installation location. However, we strongly recommend that you change the default installation locations to place the components together (if you have not already installed the Linux versions of these components elsewhere). This simplifies the build setup steps.

The resulting base directory has approximately 15 subdirectories.

2.1.1 Command Prompts in This Guide

In this guide, commands are preceded by prompts that indicate the environment where the command is to be typed. For example:

- **`host $`**
  Indicates command to be typed into the shell window of the host Linux workstation.
- **`EVM #`**
  Indicates commands to be typed into the U-Boot shell in a console window connected to the EVM board's serial port. (Section 2.2)
- **`target $`**
  Indicates commands to be typed into the Linux shell in the terminal window connected to the EVM board's serial port or telnet session.

2.2 Installation

You install the TI DVSDK software on a Linux host. The DVSDK software is packaged to install in the same directory ("on top of") the DVEVM Linux host software directory (as described in Step 6 and Step 9 that follow).

Some Microsoft Windows installers are provided for customers who already have the Windows version of MontaVista Linux Tools. For information about installing on Windows, see Section A.1, *Installing Components Under Windows*.

Follow these steps to install the Linux version of the TI DVSDK software:

1) You must have first installed the DVEVM software as described in the *DVEVM Getting Started Guide* (SPRUE66).

   You must installed the same version of the DVEVM and DVSDK software. For example, if you want to install the DVSDK 1.10 software, you must first install the DVEVM 1.10 software. You cannot install the DVSDK 1.10 software on top of the DVEVM 1.00 software.

2) If your version of the DVSDK includes a fully licensed version of MontaVista Linux Pro, install that software as described in the Quick Start Guide in the CD/DVD case.

3) On your Linux host system, mount the TI DVSDK CD and copy the following files to a temporary location with at least 200 MB of available space. Since you can delete the installation files after installing the software, a directory like /tmp is recommended. (The # signs reflect a digit of the version number.)

   - dvsdk_setuplinux_#_##_##_##.bin
   - dsp_bios_setuplinux_#_##_##_##.bin
   - cg_setuplinux_#_##_##_##.bin

4) Ensure that an X graphical display is available, and point your DISPLAY environment variable to this value. For example:

   csh:

   ```
   host $ setenv DISPLAY mylinuxhost:0
   ```

   ksh:

   ```
   host $ export DISPLAY=mylinuxhost:0
   ```

5) Log in using the same **user account** you used to install the TI DVEVM software. In the following steps, we refer to the home user directory as "~".

6) Install the DVSDK software from the DVSDK CD. For example (your version numbers may differ):

```
host $ cd /tmp
host $ ./dvsdk_setuplinux_1_00_00_00.bin
```

When the installer prompts for an installation location, *do not* use the default installation location. Instead, install in the home directory for the account you are using. (The same directory where you installed the DVEVM software.) For example, if your home directory is /home/ada, enter that in the installation location dialog. The DVSDK software would then be installed under /home/ada/dvevm_1_10.

7) Install the DSP/BIOS software. For example:

```
host $ ./dsp_bios_setuplinux_5_30_00_11.bin
```

When the installer prompts for an installation location, *do not* use the default location. Instead, install in the dvevm_#_## directory. For example, /home/ada/dvevm_1_10.

8) Install the Code Generation Tools software. For example:

```
host $ ./cg_setuplinux_6_00_03.bin
```

When the installer prompts for an installation location, *do not* use the default location. Instead, use the **entire** path to the dvevm_#_## directory. For example, /db/jtree/ada/daVinci/dvevm_1_10. The Codegen executables are installed in a subdirectory of the location you specify called "cg6x_6_0_3".

9) You can now delete the .bin files that you loaded into the temporary directory.

> **Note:** You can uninstall one of these components by using the `rm -rf` command on its directory. You should ignore the _uninstall directories created by InstallShield.

10) If you have installed the complete, licensed version of MontaVista Linux Pro, set up aliases for the Linux online documentation provided with MontaVista Linux. These aliases avoid conflicts between the host and target Linux man and info pages. Add the following aliases to your shell resource file (e.g., .bashrc):

```
alias mvman="MANPATH=/opt/mv_pro_4.0/montavista/pro/man man"
alias mvinfo="INFOPATH=/opt/mv_pro_4.0/montavista/pro/info info"
```

To test the alias, type the following to get the man page for gcc:

```
mvman gcc
```

2.3 Setting Up the Build/Development Environment

This section talks about how to configure the build environment to build DSP applications. You should have first performed the build setup instructions in the *DVEVM Getting Started Guide* (SPRUE66) and Section 2.2 of this guide.

The top-level Rules.make controls much of the behavior of the build environment. This file is included by the makefiles in the various component sub-directories. It is installed by the DVEVM software installation, and you will likely have modified it as part of the DVEVM software setup.

Follow these steps to configure the build and development environment:

1) Use a text editor to open the Rules.make file in the top-level DVEVM software directory (~/dvevm_#_##).
2) If necessary, modify the definitions of the following to match the locations of these components on your Linux host. Although we recommend that these components be installed within the ~/dvevm_#_## directory, you may have placed them elsewhere.
   - BIOS_INSTALL_DIR
   - FRAMEWORK_COMPONENTS_INSTALL_DIR

After making these modifications, you can run make as described in the *DVEVM Getting Started Guide*.

2.3.1 Testing the Build Environment

To test your installation of the DVSDK software, you can build one of the Codec Engine servers. This server is a DSP-side application. Building it tests the installation of DSP-side development components.

To build the video_copy server, follow these steps:

1) Go to the ~/dvevm_#_##/codec_engine_#_##/examples directory and open the build_instructions.html file.
2) Follow the step-by-step instructions for building examples. When you are editing the xdcpaths.mak file, note that the DVEVM / DVSDK installation *does not* include the cetools directory, so you will need to modify additional variables to point to the locations of xDAIS, DSP/BIOS Link, CMEM, and Framework Components.
3) When you are ready to build, go to the video_copy server directory. (~/dvevm_#_##/codec_engine_#_##/examples/servers/video_copy) and follow the steps in build_instructions.html for building a server.
4) If the build is successful, you have successfully installed the DVSDK software.

2.4 Using DVSDK Software for DSP-Side Development

After you have installed the DVSDK software, you can begin to create and modify DSP-side applications for your DM644x. See Section 1.3, *Software Architecture* for an architecture diagram that includes DSP-side applications.

The following table lists places to look for documentation on using each component of the DVSDK. Documents in PDF, HTML, and text format are included in the installations with each product.

Table 2-1. Documentation for DVSDK Components

Component	Title	Location
DSP/BIOS	*TMS320C6000 DSP/BIOS API Reference* (SPRU403)	~/dvevm_#_##/bios_5_##/packages/ti/bios/doc
	Application Notes	www.dspvillage.com
Code Generation Tools	*TMS320C6000 Optimizing C Compiler User's Guide* (SPRU187)	~/dvevm_#_##/cg6x_#_#
	TMS320C6000 Programmer's Guide (SPRU189)	
Framework Components	Release Notes	~/dvevm_#_##/framework_components_#_##
Digital Video Test Bench	README.txt	~/dvevm_#_##/dvtb/docs
	Section 2.4.1, *Using the Digital Video Test Bench (DVTB)*	this document
Codec Engine	*Codec Engine Application Developer User's Guide* (SPRUE67)	~/dvevm_#_##/codec_engine_#_##/docs
	Codec Engine Server Integrator User's Guide (SPRUED5)	~/dvevm_#_##/codec_engine_#_##/docs
	Codec Engine Algorithm Creator User's Guide (SPRUED6)	~/dvevm_#_##/codec_engine_#_##/docs
	Example Build and Run Instructions	~/dvevm_#_##/codec_engine_#_##/examples/build_instructions.html
	Codec Engine API Reference	~/dvevm_#_##/codec_engine_#_##/docs/html/index.html

Table 2-1. Documentation for DVSDK Components

Component	Title	Location
	Codec Engine SPI Reference Guide	~/dvevm_#_##/codec_engine_#_##/docs/spi/html/index.html
	Configuration Reference	~/dvevm_#_##/codec_engine_#_##/packages/xdoc/index.html
XDC Tools (used by Codec Engine)	Documentation Links	~/dvevm_#_##/xdctools_#_##/doc/index.html
xDAIS	*xDAIS-DM (Digital Media) User Guide* (SPRUEC8)	~/dvevm_#_##/xdais_5_00/packages/ti/xdais/dm/doc

2.4.1 Using the Digital Video Test Bench (DVTB)

To use the Digital Video Test Bench, follow these steps:

1) Move to the ~/dvevm_#_##/dvtb subdirectory.

2) Copy the pre-built dvtb ARM executable file from dvevm_#_##/dvtb to /opt/dvevm on the device's target filesystem (either hard disk or NFS) and run it there. It must be in the same directory as the DSP executables.

3) See the README.txt file on DVTB in the ~/dvevm_#_##/dvtb/docs directory. This file contains information about the features of DVTB, along with how to build, install, and use DVTB.

4) In addition, to the information provided in the docs subdirectory, make the following update:

   - As of this printing, DVTB does not clear or initialize the OSD transparency attributes, so the video window will not be visible. To fix this, after each reboot of the system, simply write zeros to the attribute buffer using the following Linux command:

     ```
     root# cat /dev/zero > /dev/fb/2
     root# dvtb
     ```

     You can ignore any errors that say "cat: write error: No space left on device".

Appendix A

Additional Procedures

This appendix describes optional procedures you may use depending on your setup and specific needs.

Topic **Page**

A.1 Installing Components Under Windows A–2

A.1 Installing Components Under Windows

A Linux host is the recommended development platform for DVEVM and DVSDK development. However, it is possible to use a Microsoft Windows host for such development. Code Composer Studio is supported only on Windows, so using a Windows host will allow you to use the Code Composer Studio IDE for DSP-side development and debugging. However, since the expected host is a Linux platform, less documentation is provided for Windows host build issues.

The DVEVM and DVSDK provide the following Microsoft Windows installers for customers who already have the Windows version of MontaVista Linux Tools:

- ❏ DVEVM: mvl_lsp_setupwin32_#_##_##_##.exe
- ❏ DVEVM: dvevm_setupwin32_#_##_##_##.exe
- ❏ DVSDK: dvsdk_setupwin32_#_##_##_##.exe
- ❏ DVSDK: dsp_bios_setupwin32_#_##_##_##.exe
- ❏ DVSDK: cg_setupwin32_#_##_##_##.exe

The DVEVM and DVSDK do not include the Windows version of the MontaVista Linux Tools. If you want to use a Windows host for DaVinci development, you will need to obtain the Windows version of the MontaVista Linux Tools from Texas Instruments. These are cross compilers that run on Windows to generate target-side code. You must also have the Windows version of the MVL Target (file system).

To set up a Windows host for DaVinci development, follow these steps:

1) Run all the Windows installers in the sequence listed above. The DVEVM and DVSDK software installers expect the MontaVista Linux tools to have already been installed.
2) When running the installers (even if you are in a MVL bash shell), specify a full Windows-style path with backslashes. For example, C:\mvcyg4.0\home\ada.
3) Use a text editor to edit the Rules.make file. Make the following changes:
   - MVTOOL_PREFIX: Specify the prefix to be added before the GNU compiler tools. Use UNIX format (forward slashes) with a preceding C:. For example, you might use:

```
C:/mvcyg4.0/opt/mv_pro_4.0/montavista/pro/devkit/arm/v5t_le/bin/arm_v5t_le-
```

   - EXEC_DIR: Specify the location to which the resulting executables and data should be copied using UNIX format (forward slashes). For example, C:/mvcg4.0/opt/dvevm.

Index

A

account 2-3
analyzer 1-2

B

base directory 2-2
BIOS_INSTALL_DIR environment variable 2-5

C

CDs 1-2
cg_setuplinux_#.bin installer 2-3
cg_setupwin32_#.exe installer A-2
Code Composer Studio 1-3
  in emulator kit 1-3
Codec Engine 1-3, 1-4
  documentation 2-6
  remote server 1-5
Codec Server 1-4
Codegen Tools 1-3
  documentation 2-6
  installation location 2-2
command prompts 2-2
contents of kit 1-2
cross compiler A-2

D

DaVinci technology 1-2
DevRocket 1-2
Digital Video Test Bench (DVTB) 1-3
  building 2-6, 2-7
  documentation 2-6, 2-7
directories 2-2
  base 2-2
  installation 2-4
DISPLAY environment variable 2-3
DM644x SoC Analyzer 1-2
  installing 1-5
DMA resources 1-3
documentation
  Linux aliases 2-4
DSA 1-2
  installing 1-5
DSP/BIOS 1-3, 1-5
  Configuration Tool 1-3
  documentation 2-6
  installation location 2-2
DSP/BIOS Link 1-5
dsp_bios_setuplinux_#.bin installer 2-3
dsp_bios_setupwin32_#.exe installer A-2
DSP-side development 1-2
DVEVM 1-2
dvevm_setupwin32_#.exe installer A-2
DVSDK 1-2
dvsdk_setuplinux_#.bin installer 2-3
dvsdk_setupwin32_#.exe installer A-2

E

emulator 1-3
environment variables
  BIOS_INSTALL_DIR 2-5
  FRAMEWORK_COMPONENTS_INSTALL_DIR 2-5
EVM # prompt 2-2
EXEC_DIR environment variable
  environment variables
    EXEC_DIR A-2

F

Framework Components 1-3
FRAMEWORK_COMPONENTS_INSTALL_DIR environment variable 2-5

H

host $ prompt 2-2
host platform 2-3
  Windows A-2

I

installation location 2-4
installing
  on Linux host 2-3
  on Windows host A-2

J

JTAG 1-3

K

kit contents 1-2

L

Linux 1-2
Linux host 2-3

M

Microsoft Windows 2-3, A-2
MontaVista Linux Pro 1-2
  installing 1-5, 2-3
MontaVista Linux Tools for Windows A-2
mvl_lsp_setupwin32_#.exe installer A-2
MVTOOL_PREFIX environment variable
  environment variables
    MVTOOL_PREFIX A-2

O

OSD transparency 2-7

P

profiling 1-2
prompts 2-2

Q

Quick Start Guide 1-2

R

Real-Time Analysis (RTA) tools 1-3
Rules.make
  editing on Linux host 2-5

S

server, codec 1-4
SoC Analyzer 1-2
  installing 1-5
SPRUE66 2-2
subdirectories 2-2

T

target $ prompt 2-2

U

user account 2-3

W

Windows 2-3, A-2
  MontaVista Linux Tools A-2

X

X graphical display 2-3
xDAIS 1-3, 1-5
XDS560 emulator 1-3