

Introduction

The creation and consumption of cool gadgets has become a way of life for many designers, engineers, and billions of consumers. These gadgets, whether for music, video, Web, navigation, games, or communications, keep getting ever more sophisticated, *and* cleverer and smaller.

Buried inside these electronic gadgets are high-tech embedded digital processors. Depending on the device, the processor may be responsible for decompressing and displaying video, storing music, running and displaying a Web browser, taking inputs from a touch-screen, or playing amusing ringtones. The tasks handled by these embedded processors usually fall into one of two categories:

- ✓ General purpose processing tasks such as data transfer or running graphical user interfaces
- ✓ Digital signal-processing tasks such as compressing and decompressing video, audio, and speech streams

Traditionally, different physical devices would be employed to perform these somewhat diverse functions. Today, however, all of this functionality can be packed into a single device.

Texas Instruments has been a pioneer in the area of processor integration with its family of System on Chip (SOC) devices branded *OMAP* and *DaVinci*. These devices make clever use of a combination of general-purpose processors, specialized digital-signal processors, and dedicated hardware accelerators to make digital gadgets literally sing and dance. Even better, devices based on OMAP and DaVinci are really stingy on power consumption, so smaller, lighter batteries can last a long time.

But — you just knew there had to be a but — to make any OMAP or DaVinci device perform the way you want requires a *lot* of software. Fortunately, there's good news, and *really* good news. The first piece of good news is that software *is* soft. In other words, the functionality of a gadget is only limited by the creativity of its programmers.

The really good news is that Texas Instruments provides much of the embedded software and tools required to get OMAP and DaVinci developers up and running fast. Developers can focus on cool applications and not lose valuable time on tasks that don't add real value.

Who Should Read This Book

Any designer or manager about to embark on a project that might involve an OMAP or a DaVinci processor should read this book. If you're a software programmer, you'll discover the various embedded software components and tools that are offered by Texas Instruments, enabling you to get to the good stuff sooner rather than later. If you're worried about being forced into the world of grungy low-level software, you'll appreciate TI's solid software infrastructure that's described in this book.

How to Use This Book

OMAP and DaVinci Software For Dummies is organized so that you can either read it front to back, jump to interesting or relevant sections, or skip straight to the hands-on stuff. If you have little or no experience with software on Texas Instruments devices, we recommend that you become familiar with all the concepts described in Part I before embarking on your development journey in Part II. You wouldn't want to drive off in a rental car at night without knowing how to turn on the headlights, and you wouldn't want to start developing software on a new OMAP or DaVinci device before understanding how TI's system works.

The embedded software and tooling available from Texas Instruments is represented in Figure 1. This book introduces you to the components in the figure and shows you how to use these components to get real-world applications up and running.

How This Book Is Organized

This book is broken into three parts, each offering a different view of the embedded software and tools that run on OMAP and DaVinci devices.

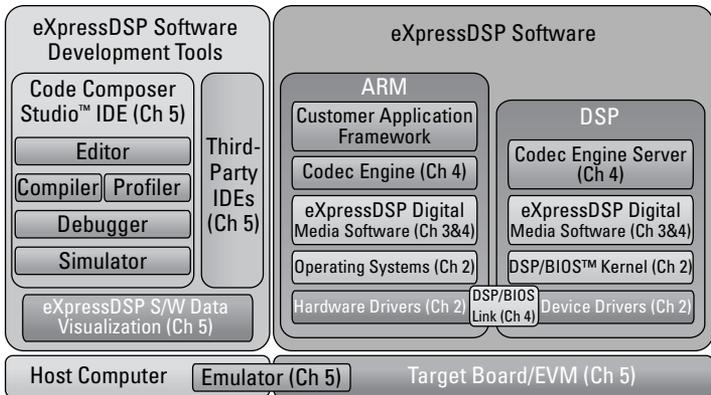


Figure 1: TI offers all these components to speed up your development.

Part I: Understanding the Embedded Software and Tools for OMAP and DaVinci Processors

In Chapter 1, we show you why heterogeneous multi-core devices like OMAP and DaVinci processors have risen to such prominence. Chapter 2 covers the operating systems (Linux, for example) and kernels (DSP/BIOS) required to make OMAP and DaVinci processors run efficiently. Chapter 3 digs into standards (XDAIS, XDM, RTSC) that TI has developed to make it easier for developers to quickly integrate multiple signal processing codecs into the final system. Chapter 4 looks at the framework software, Codec Engine, which forms a convenient housing for XDAIS-compliant algorithms. We also discuss the inter-processor communication protocol (DSP/BIOS Link) used to communicate between the two cores on the SOC. Finally in this part, Chapter 5 explores the development tool choices for both ARM and DSP.

Part II: Building Something Real — Now!

In Part II you work hands-on with one of TI's hardware evaluation platforms (OMAP35xx EVM). You build a demo

application where you can literally see and hear an OMAP processor in action. You actually get a chance to run an MPEG4 video decoder and an AAC audio decoder.

Chapter 6 helps you build a “Hello World” program on the OMAP35xx EVM hardware. Chapter 7 explores the TI MPEG4 video decoder that is used in later chapters. Chapter 8 leads you through packaging a codec and preserving its performance when implemented in a framework. Chapter 9 is where you create DSP server executables. In Chapter 10, you get real codecs running in a test bench. You get to see video and hear audio being processed on the OMAP processor. There’s nothing to it!

Part III: The Part of Tens

A standard feature of all *For Dummies* books, each Part of Tens chapter provides ten useful bits of information. In Chapter 11, we cover ten top codec recommendations that Texas Instruments publishes for codec developers to follow. In Chapter 12, we give you ten great resources for helping you out during your OMAP and DaVinci development.

Icons Used in This Book

We use icons throughout this book to call attention to material worth noting in a special way. Here’s a list of the icons you’ll see and a description of what each icon means.



Some points bear repeating, and others bear remembering. When you see this icon, take special note of what you’re about to read.



This icon indicates technical information that is probably most interesting to programmers, but you never know when you may need to talk to one.



If you see a Tip icon, pay attention — this is handy, real-world advice.

Where to Go from Here

Simply turn the page.