

# DSP 技术原理与应用

# Techniques And Applications *Of* DSP Chips

● 关华 胡秀蒙 庄华伟 著



济南出版社

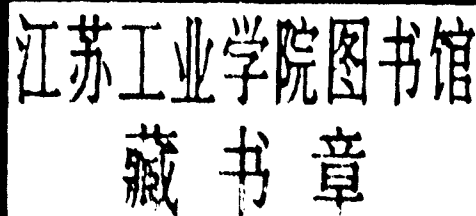
DSP 芯片的原理与应用

# Techniques And Applications Of DSP Chips

By

Guan Hua Hu XiuMeng Zhuang HuaWei

关 华 胡秀蒙 庄华伟 著



济南出版社

2006.10

#### **Important Legal Information: Warning and Disclaimer**

This book presents the fundamentals applications of DSP using examples from common science and engineering problems. While the authors believe that the concepts and data contained in this book are accurate and correct, they should not be used in any application without proper verification by the person making the application. Extensive and detailed testing is essential where incorrect functioning could result in personal injury or damage to property. The material in this book is intended solely as a reference and teaching aid, and is not represented to be an appropriate or safe solution to any particular problem. For this reason, the author, publisher, and distributors make no warranties, express or implied, that the concepts, examples, data, algorithms, techniques, or programs contained in this book are free from error, conform to any industry standard, or are suitable for any application. The author, publisher, and distributors disclaim all liability and responsibility to any person or entity with respect to any loss or damage caused, or alleged to be caused, directly or indirectly, by the information contained in this book. If you do not wish to be bound by the above, you may return this book to the publisher for a full refund.

#### **图书在版编目 (CIP) 数据**

DSP 芯片的原理与应用 / 关华, 胡秀蒙, 庄华伟 著。

济南: 济南出版社, 2006.10

ISBN 7-80710-337-X

I. D... II. ①关... ②胡... ③庄... III. 数字信号—信号处理 IV. TN911.72

中国版本图书馆 CIP 数据核字 (2006) 第 114810 号

责任编辑: 张静

封面设计: 苗壮

地址: 济南市经七路 251 号

印刷: 山东省实验中学印刷厂

版次: 2006 年 10 月第一版

印次: 2006 年 10 月第一次印刷

开本: 大 16 开

印张: 19

字数: 300K 字

定价: ¥38.00 元

# **Preface**

## **About This Book**

The term DSP stands for Digital Signal Processing, or the term DSPs stands for Digital Signal Processors.

Digital signal processing is one of the brightest spots in the semiconductor business today, and one of the few deserving the title "breakthrough." Like earlier advances in microprocessors and computer memories, digital signal processing is a foundation technology with the power to transform broad areas of the electronics industry. Its impact is being felt in applications as diverse as stereo systems, cars, personal computers, and cellular phones. In the next few years, digital signal processing will give rise to hundreds of new products and change what people expect from technology.

Digital Signal Processing is a technique that converts signals from real world sources (usually in analog form) into digital data that can then be analyzed.

A Digital Signal Processor is a special-purpose CPU (Central Processing Unit) that provides ultra-fast instruction sequences, such as shift and add, and multiply and add, which are commonly used in math-intensive signal processing applications.

The introduction of the microprocessor in the late 1970's and early 1980's made it possible for DSP techniques to be used in a much wider range of applications. DSP chips are used in sound cards, fax machines, modems, cellular phones, high-capacity hard disks and digital TVs. Among the numerous manufacturers, Texas Instruments (TI) is one of the major providers of DSPs.

The DSPs, the specifically designed microprocessors for handling Digital Signal Processing tasks, have some special features. This book is designed to introduce you to the common features, the architectures characteristics and applications of these devices, the DSPs. The TMS320C6000 platform from TI will be taken in this book as examples to do analysis and descriptions on the DSPs' applications.

In the first part of this book, which contains chapter 1 and 2, we give an overview on the Digital Signal Processing (DSP) and Digital Signal Processors (DSPs). The evolution and miscellaneous aspects on DSPs have also been described in details in this part. Part two of this book gives a detailed description on the DSPs' features and architectures. Taking the TMS320C6000 platform as an example, we make a thorough analysis and descriptions on every aspect of DSP devices in chapter 3 to 9, such as the CPU architecture, memory, peripherals, instruction set, pipeline operations and interrupts.

Part three of this book, chapter 10 to 14, may be served as a reference book for DSP design and applications. This part presents descriptions on the DSP developing tools and summarizes the DSPs' developing experiences undergone by the research team to which the authors belong. Specifically, this part complements the DSP knowledge by providing information to assist managers and hardware/software engineers in application development. It includes example code, hardware connections, design examples and schemes for various applications. It also defines the principles involved in many applications for instructional purposes.

This book was written for scientists and engineers in the field of DSP applications. The goal is to present practical techniques. Much of the information and data for TMS320 DSP was come from the TI's data books and data sheets. For more detailed information on these data, please refer to the original materials provided by Texas Instruments Incorporated.

## **Acknowledgements**

Special thanks go to the many reviewers who provided comments and suggestions on this book, and to the many technical staff members of TI and TI's distributor in Engineering and Marketing who provided invaluable materials and information during this project. Thanks are also due research team to which the authors belong for providing valuable materials and suggestions.

Authors of this book  
Oct. 2006

# CONTENTS

## Part One Overview

<b>Chapter 1 A Beginner's Guide to DSP .....</b>	<b>3</b>
<b>1.1 What is DSP? .....</b>	<b>4</b>
1.1.1 Digital Signal Processing	
1.1.2 Digital Signal Processor	
<b>1.2 An Introduction to Digital Signal Processing .....</b>	<b>5</b>
1.2.1 DSP Basics	
1.2.2 From Analog to Digital	
1.2.3 Signal Processing	
1.2.4 Development of DSP	
1.2.5 Blinding Speed	
<b>1.3 An Introduction to Digital Signal Processors .....</b>	<b>7</b>
1.3.1 Digital Signal Processors (DSPs)	
1.3.2 DSPs vs. Microprocessors	
1.3.3 Types of DSPs	
1.3.4 Different DSPs For Different Jobs	
1.3.5 DSPs Evolution	
1.3.6 Applications of DSP	
1.3.7 Miscellaneous	
<b>Chapter 2 DSP Hardware .....</b>	<b>17</b>
<b>2.1 Architecture of the Digital Signal Processor .....</b>	<b>18</b>
2.1.1 What's Inside a DSP?	
2.1.2 Characteristics of DSP Processors	
<b>2.2 Data Formats .....</b>	<b>30</b>
<b>2.3 Interfaces: Input / Output, Communication .....</b>	<b>34</b>

2.3.1 Synchronous Serial Ports	
2.3.2 Host Port and Host Communications	
2.3.3 Link Ports (Parallel Ports) for Interprocessor Communications	
2.3.4 JTAG Port for System Test	
<b>2.4 Example Processors</b>	<b>38</b>
<b>2.5 Introduction to TI's DSP</b>	<b>43</b>
2.5.1 History and Features of TMS320 DSPs	
2.5.2 Typical Applications for the TMS320 Family	
2.5.3 Platforms of TMS320 DSP	
<b>2.6 DSP System</b>	<b>47</b>
2.6.1 A Diagram of DSP Application System	
2.6.2 A Basic Mini DSP System	

## **Part Two TMS320C6000 DSPs**

<b>Chapter 3 Introduction to the TMS320C6000 Platform of DSPs</b>	<b>53</b>
3.1 Overview of the TMS320C6000 Platform of DSPs	54
3.2 Features and Options of the TMS320C6000 Devices	54
<b>Chapter 4 CPU Architecture</b>	<b>57</b>
4.1 TMS320C62x/C67x Block Diagram	58
4.2 Central Processing Unit (CPU)	59
4.3 CPU Data Paths	60
4.3.1 General-Purpose Register Files	
4.3.2 Functional Units	
4.3.3 TMS320C62x/C67x Control Register Files	
4.3.4 TMS320C67x Control Register File Extensions	
4.3.5 Register File Cross Paths	
4.3.6 Memory, Load and Store Paths	
4.3.7 Data-Address Paths	
4.4 Mapping Between Instructions and Functional Units	65
4.5 Addressing Modes	67
4.6 Interrupts	67
<b>Chapter 5 Memory</b>	<b>69</b>
5.1 Memory Maps	70
5.2 Internal Memory	71

5.2.1 Memory in TMS320C6201 (Revision 2)	
5.2.2 Memory in TMS320C6201B (Revision 3)	
5.2.3 Memory in TMS320C6701	
5.2.4 Memory in TMS320C6202	
5.2.5 Memory in TMS320C6211	
<b>5.3 Data Memory Access</b>	<b>74</b>
5.3.1 TMS320C6201, C6202, C6701	
5.3.2 TMS320C6211	
5.3.3 Access Data Memory via Peripheral Bus	
5.3.4 Access Data Memory via Expansion Bus	
<b>5.4 External Memory Interface (EMIF)</b>	<b>77</b>
<b>Chapter 6 Peripherals</b>	<b>79</b>
<b>6.1 Direct Memory Access (DMA) Controller</b>	<b>80</b>
<b>6.2 Enhanced Direct Memory Access (EDMA)</b>	<b>81</b>
<b>6.3 Host-Port Interface (HPI)</b>	<b>82</b>
<b>6.4 Expansion Bus (XB)</b>	<b>83</b>
<b>6.5 External Memory Interface (EMIF)</b>	<b>85</b>
6.5.1 SDRAM Interface	
6.5.2 SBSRAM Interface	
6.5.3 Asynchronous Interface	
<b>6.6 Boot Configuration Logic</b>	<b>89</b>
6.6.1 Device Reset	
6.6.2 Boot Configuration	
<b>6.7 Multichannel Buffered Serial Port (McBSP)</b>	<b>90</b>
<b>6.8 Timers</b>	<b>93</b>
<b>6.9 Interrupt Selector</b>	<b>93</b>
<b>6.10 Power-Down Logic</b>	<b>93</b>
<b>6.11 PCI Port</b>	<b>95</b>
6.11.1 PCI Architecture	
6.11.2 PCI Registers	
6.11.3 TMS320C6000 PCI Memory Map	
6.11.4 PCI Transfers to/from Program Memory (TMS320C62x/C67x)	
6.11.5 DSP Reset Source/Status Register (RSTSRC)	
6.11.6 PCI Reset of DSP	
6.11.7 PCI Boot	



<b>Chapter 7 TMS320C6000 Instruction Set .....</b>	<b>105</b>
<b>7.1 TMS320C62x/C64x/C67x Fixed-Point Instruction Set .....</b>	<b>106</b>
7.1.1 Delay Slots for Fixed-Point Instructions	
7.1.2 Parallel Operations	
7.1.3 Conditional Operations	
7.1.4 Resource Constraints	
7.1.5 Addressing Modes	
7.1.6 Instruction Descriptions	
<b>7.2 TMS320C67x Floating-Point Instruction Set .....</b>	<b>118</b>
7.2.1 Overview of IEEE Standard Single- and Double-Precision Formats	
7.2.2 Delay Slots for Floating-Point Instructions	
7.2.3 TMS320C67x Instruction Constraints	
<b>Chapter 8 TMS320C62x/C64x Pipeline .....</b>	<b>127</b>
<b>8.1 Pipeline Operation Overview .....</b>	<b>128</b>
8.1.1 Fetch	
8.1.2 Decode	
8.1.3 Execute	
8.1.4 Summary of Pipeline Operation	
<b>8.2 Pipeline Execution of Instruction Types .....</b>	<b>137</b>
8.2.1 Single-Cycle Instructions	
8.2.2 Two-Cycle Instructions and C64x Non-Multiply .M Unit Operations	
8.2.3 Store Instructions	
8.2.4 Extended Multiply Instructions	
8.2.5 Load Instructions	
8.2.6 Branch Instructions	
<b>8.3 Performance Considerations .....</b>	<b>144</b>
8.3.1 Pipeline Operation With Multiple Execute Packets in a Fetch Packet	
8.3.2 Multicycle NOPs	
8.3.3 Memory Considerations	
<b>Chapter 9 Interrupts .....</b>	<b>153</b>
<b>9.1 Overview of Interrupts .....</b>	<b>154</b>
9.1.1 Types of Interrupts and Signals Used	
9.1.2 Interrupt Service Table (IST)	
9.1.3 Summary of Interrupt Control Registers	
<b>9.2 Globally Enabling and Disabling Interrupts .....</b>	<b>160</b>

<b>9.3 Individual Interrupt Control .....</b>	<b>162</b>
9.3.1 Enabling and Disabling Interrupts	
9.3.2 Status of Setting and Clearing Interrupts	
9.3.3 Returning From Interrupt Servicing	
<b>9.4 Interrupt Detection and Processing .....</b>	<b>167</b>
9.4.1 Setting the Nonreset Interrupt Flag	
9.4.2 Conditions for Processing a Nonreset Interrupt	
9.4.3 Actions Taken During Nonreset Interrupt Processing	
9.4.4 Setting the RESET Interrupt Flag for the TMS320C6000	
9.4.5 Actions Taken During RESET Interrupt Processing	
<b>9.5 Performance Considerations .....</b>	<b>171</b>
9.5.1 General Performance	
9.5.2 Pipeline Interaction	
<b>9.6 Programming Considerations .....</b>	<b>172</b>
9.6.1 Single Assignment Programming	
9.6.2 Nested Interrupts	
9.6.3 Manual Interrupt Processing	
9.6.4 Traps	

### **Part Three DSP Applications**

<b>Chapter 10 Tips on Hardware Design Techniques for DSP .....</b>	<b>179</b>
<b>10.1 Overview on the TMS320C6205 .....</b>	<b>180</b>
10.1.1 Device Characteristics	
10.1.2 Functional and CPU (DSP Core) Block Diagram	
10.1.3 CPU (DSP Core) Description	
10.1.4 Memory Map Summary	
10.1.5 Signal Groups Description	
10.1.6 Signal Descriptions	
10.1.7 Device Nomenclature	
10.1.8 Power-Supply and Ratings	
<b>10.2 System Clock Circuit .....</b>	<b>197</b>
10.2.1 Clock PLL	
10.2.2 Clock Synthesizer Circuit	
<b>10.3 Reset Timing and Reset Circuit .....</b>	<b>200</b>

10.3.1 Reset Process	
10.3.2 Reset Timing	
10.3.3 Reset Signal and Circuit	
10.3.4 How to Initialize the Processor	
<b>10.4 Boot Modes and Configuration</b>	<b>206</b>
10.4.1 Overview	
10.4.2 Device Reset	
10.4.3 Memory Map	
10.4.4 Boot Configuration	
10.4.5 Boot Processes	
<b>Chapter 11 Tips on Programming The DSP</b>	<b>213</b>
11.1 Writing C/C++ Code	214
11.1.1 Tips on Data Types	
11.1.2 Analyzing C Code Performance	
11.2 Structure of Assembly Code	215
11.2.1 Labels	
11.2.2 Parallel Bars	
11.2.3 Conditions	
11.2.4 Instructions	
11.2.5 Functional Units	
11.2.6 Operands	
11.2.7 Comments	
11.3 Interrupts Routines	221
11.3.1 Single Assignment vs. Multiple Assignment	
11.3.2 Interruptible Loops	
11.3.3 Interruptible Code Generation	
11.3.4 Interrupt Subroutines	
<b>Chapter 12 Designing for JTAG Emulation</b>	<b>233</b>
12.1 Designing Your Target System's Emulator Connector	234
12.2 Bus Protocol	235
12.3 IEEE 1149.1 Standard	235
12.4 JTAG Emulator Cable Pod Logic	235
12.5 JTAG Emulator Cable Pod Signal Timing	236
12.6 Emulation Timing Calculations	237
12.7 Connections Between the Emulator and the Target System	239

---

12.7.1 Buffering Signals	
12.7.2 Using a Target-System Clock	
12.7.3 Configuring Multiple Processors	
<b>12.8 Mechanical Dimensions for the 14-Pin Emulator Connector</b>	<b>242</b>
<b>12.9 Emulation Design Considerations</b>	<b>244</b>
12.9.1 Using Scan Path Linkers	
12.9.2 Emulation Timing Calculations for SPL	
12.9.3 Using Emulation Pins	
12.9.4 Performing Diagnostic Applications	
<b>Chapter 13 Overview on Code Composer Studio</b>	<b>253</b>
<b>13.1 Code Composer Studio Development</b>	<b>254</b>
<b>13.2 Code Generation Tools</b>	<b>254</b>
<b>13.3 Code Composer Studio Integrated Development Environment</b>	<b>257</b>
13.3.1 Program Code Editing Features	
13.3.2 Application Building Features	
13.3.3 Application Debugging Features	
<b>13.4 DSP/BIOS Plug-ins</b>	<b>259</b>
13.4.1 DSP/BIOS Configuration	
13.4.2 DSP/BIOS API Modules	
<b>13.5 Hardware Emulation and Real-Time Data Exchange</b>	<b>263</b>
<b>13.6 Third-Party Plug-ins</b>	<b>266</b>
<b>13.7 Code Composer Studio Files and Variables</b>	<b>266</b>
13.7.1 Installation Folders	
13.7.2 File Extensions	
13.7.3 Environment Variables	
13.7.4 Increasing DOS Environment Space	
<b>13.8 Developing a Simple Program</b>	<b>268</b>
13.8.1 Creating a New Project	
13.8.2 Adding Files to a Project	
13.8.3 Reviewing the Code	
13.8.4 Building and Running the Program	
13.8.5 Changing Program Options and Fixing Syntax Errors	
13.8.6 Using Breakpoints and the Watch Window	
13.8.7 Using the Watch Window with Structures	
13.8.8 Profiling Code Execution Time	

13.8.9 Things to Try	
13.8.10 Learning More	
<b>Chapter 14 Design Examples</b>	<b>283</b>
<b>14.1 Developing Board DB6205</b>	<b>284</b>
14.1.1 DB6205 Hardware Function Overview	
14.1.2 DB6205 Block Diagram	
<b>14.2 Information Security Application</b>	<b>285</b>
<b>14.3 TMS320C6201/6701 EVM Board</b>	<b>286</b>
<b>14.4 Related Application Schemes</b>	<b>288</b>
14.4.1 Motor Control and Driving	
14.4.2 Power Line Supervision Application	
14.4.3 Information Appliances Application	
<b>References</b>	<b>291</b>

# **Part One**

## **Overview**

Digital signal processing is one of the brightest spots in the semiconductor business today, and one of the few deserving the title "breakthrough." Like earlier advances in microprocessors and computer memories, digital signal processing is a foundation technology with the power to transform broad areas of the electronics industry. Its impact is being felt in applications as diverse as stereo systems, cars, personal computers, and cellular phones. In the next few years, digital signal processing will give rise to hundreds of new products and change what people expect from technology.



# **Chapter 1**

## **A Beginner's Guide to DSP**

Have you noticed how everything is becoming digital these days – DSP filters, DSP modems, DSP transceivers and more – and for good reason. Digital representation of signals has many advantages over their analog counterparts:

- Fewer components;
- Stable, deterministic performance;
- Wide range of applications;
- No filter adjustments;
- Filters with much closer tolerances;
- Adaptive filters easily implemented.

Digital Signal Processing is so pervasive in everything, an amateur radio operator does and it is becoming an important tool. The part one of this book is designed to introduce you to DSP.



## **1.1 What is DSP?**

The term DSP stands for Digital Signal Processing, or the term DSPs stands for Digital Signal Processors.

### **1.1.1 Digital Signal Processing**

Digital Signal Processing is a technique that converts signals from real world sources (usually in analog form) into digital data that can then be analyzed. Analysis is performed in digital form because once a signal has been reduced to numbers, its components can be isolated, analyzed and rearranged more easily than in analog form.

Eventually, when the DSP has finished its work, the digital data can be turned back into an analog signal, with improved quality. For example, a DSP can filter noise from a signal, remove interference, amplify frequencies and suppress others, encrypt information, or analyze a complex wave form into its spectral components.

This process must be handled in real-time – which is often very quickly. For instance, stereo equipment handles sound signals of up to 20 kilohertz (20,000 cycles per second), requiring a DSP to perform hundreds of millions of operations per second.

### **1.1.2 Digital Signal Processor**

A Digital Signal Processor is a special-purpose CPU (Central Processing Unit) that provides ultra-fast instruction sequences, such as shift and add, and multiply and add, which are commonly used in math-intensive signal processing applications.

DSPs are not the same as typical microprocessors though. Microprocessors are typically general purpose devices that run large blocks of software. They are not often called upon for real-time computation and they work at a slower pace, choosing a course of action, then waiting to finish the present job before responding to the next user command. A DSP, on the other hand, is often used as a type of embedded controller or processor that is built into another piece of equipment and is dedicated to a single group of tasks. In this environment, the DSP assists the general purpose host microprocessor.

Digital signal processing takes real-time, high-speed information, such as radio, sound or video signals, and manipulates it for a variety of purposes. Digital signal processing can restore vintage jazz recordings to their original clarity, erase the static from long-distance phone lines and enable satellites to pick out terrestrial objects as small as a golf ball. In cars, Digital Signal Processors (DSPs) create digital audio "surround sound" and are responsible for active suspension systems that adjust automatically to road conditions. In cellular phones, digital signal processing helps