

# DIGITAL SIGNAL PROCESSING IMPLEMENTATION

using the TMS320C6000™ DSP PLATFORM

Suranaree University of Technology



31051000609228

NAIM DAHNOUN

DSP

TMS320C6000

Prentice  
Hall



# Contents

---

<b>Preface</b>	xi
<b>Acknowledgements</b>	xiv
<b>Chapter 1 Introduction</b>	<b>1</b>
1.1 Introduction	1
1.2 Why do we need DSP processors?	2
1.3 How do we define real-time?	2
1.4 What are the typical DSP algorithms?	2
1.5 Who are the main general-purpose DSP processor manufacturers?	3
1.6 Parameters to be considered when choosing a DSP processor	5
1.7 General-purpose DSP vs DSP in ASIC	6
1.8 DSP market	7
1.9 The TMS320 family evolution	7
<b>Chapter 2 The TMS320C6000 architecture</b>	<b>8</b>
2.1 Overview	8
2.2 The Central Processing Unit (CPU)	9
2.2.1 Program control unit	10
2.2.2 CPU data paths	11
2.2.3 Program execute units	15
2.2.4 Control registers	18
2.2.5 Register files	18
2.3 Memory	20
2.3.1 Data memory access	20
2.3.2 Internal memory organisation for the 'C6201 Rev.2	23
2.3.3 Direct Memory Access (DMA) controller	24
2.4 Serial ports	25
2.5 Host Port Interface	25
2.6 Boot function	26
2.7 Internal timers	26
2.8 'C62xx/'C67xx interrupts	27
2.8.1 Reset ( $\overline{\text{RESET}}$ )	28
2.8.2 NonMaskable Interrupt (NMI)	28
2.8.3 Maskable interrupts (INT4–INT15)	29
2.9 Instruction set	37
2.9.1 Writing Assembly code	38

<b>Chapter 3 Software development tools and TMS320C6201 EVM overview</b>	40
<b>3.1 Introduction</b>	40
<b>3.2 Software development tools</b>	40
3.2.1 Compiler	41
3.2.2 Assembler	42
3.2.3 Linker	43
3.2.4 Compile, assemble and link	45
<b>3.3 TMS320C6201 Evaluation Module, EVM</b>	45
3.3.1 EVM features	45
3.3.2 Using interrupts	46
3.3.3 Stereo codec overview	48
3.3.4 Configuring and using the internal timers	51
3.3.5 Testing the EVM	52
<b>Chapter 4 Software optimisation</b>	57
<b>4.1 Introduction</b>	57
4.1.1 Code optimisation procedure	57
4.1.2 The C compiler options	59
<b>4.2 Assembly optimisation</b>	60
4.2.1 Parallel instructions	61
4.2.2 Removing the NOPs	62
4.2.3 Loop unrolling	63
4.2.4 Word access	63
4.2.5 Optimisation summary	64
<b>4.3 Software pipelining</b>	64
4.3.1 Software pipelining procedure	65
<b>4.4 Linear assembly</b>	73
4.4.1 Trip count	75
<b>Chapter 5 Finite Impulse Response (FIR) filter implementation</b>	79
<b>5.1 Introduction</b>	79
<b>5.2 Properties of an FIR filter</b>	79
5.2.1 Filter coefficients	79
5.2.2 Frequency response of an FIR filter	80
5.2.3 Phase linearity of an FIR filter	81
<b>5.3 Design procedure</b>	82
5.3.1 Specifications	82
5.3.2 Coefficients calculation	83
5.3.3 Realisation structure	89
5.3.4 Filter implementation	93

<b>Chapter 6 Infinite Impulse Response (IIR) filter implementation</b>	114
6.1 Introduction	114
6.2 Design procedure	115
6.3 Coefficients calculation	115
6.3.1 Pole-zero placement approach	115
6.3.2 Analogue to digital filter design	116
6.3.3 Bilinear transform (BZT) method	116
6.3.4 Impulse invariant method	126
6.4 IIR filter implementation	131
6.5 Testing the designed IIR filter	137
<b>Chapter 7 Adaptive filter implementation</b>	138
7.1 Introduction	138
7.2 Mean square error (MSE)	139
7.3 Least mean square (LMS)	140
7.4 Implementation of an adaptive filter using the LMS algorithm	141
<b>Chapter 8 Goertzel algorithm implementation</b>	152
8.1 Introduction	152
8.2 Modified Goertzel algorithm	153
8.3 Implementing the modified Goertzel algorithm	156
8.4 Algorithm optimisation	160
8.4.1 Hand optimisation of the Goertzel algorithm	160
8.4.2 Optimisation by using linear assembly	164
8.5 Direct Memory Access (DMA)	166
8.5.1 DMA features	167
8.6 Practical example for programming the DMA	169
<b>Chapter 9 Implementation of the Discrete Cosine Transform</b>	173
9.1 Introduction	173
9.2 Optimisation of DCT and IDCT for DSP implementation	176
9.2.1 Two-dimensional DCT using a one-dimensional DCT pair	176
9.3 Block-based DCT and IDCT in C	178
9.4 Simple DSP implementation of DCT and IDCT on an image	180
9.4.1 McGovern Fast 1-D DCT and IDCT	186
9.5 TMS320C6201 EVM-PC host communication	194
9.5.1 Communication through the mailbox	195
9.5.2 Communication through the FIFOs	196
9.5.3 Demonstration program	197

## **Contents**

<b>Appendix A</b>	<b>Optimisation of 1-D DCT and IDCT</b>	213
<b>A.1</b>	<b>1-D DCT (DCT)</b>	213
<b>A.2</b>	<b>1-D Inverse DCT (IDCT)</b>	222
<b>A.3</b>	<b>Application in 2-D DCT</b>	225
<b>Appendix B</b>	<b>Block memory display</b>	226
<b>References</b>		229
<b>Index</b>		231