



手把手教你学系列丛书

DSP

手把手教你学 DSP

——基于TMS320C55x

陈泰红 任胜杰 魏 宇 编著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

手把手教你学系列丛书

手把手教你学 DSP ——基于 TMS320C55x

陈泰红 任胜杰 魏 宇 编著

北京航空航天大学出版社

内 容 简 介

本书以 TMS320C55x 系列高性能低功耗 DSP 为主,主要介绍了以数字信号处理器(DSP)为核心的实时数字信号处理器的硬件结构和片内外设,论述了 eXpress 算法标准软件尤其是 CCS 的使用,详细说明了 DSP 与外围接口电路的设计以及最小系统的设计,给出了 DSP 相关软件编程和开发调试,还介绍了 MATLAB 在数字信号处理中的应用和 DSP/BIOS 的基础知识。在介绍功能模块的基础上,列出了相应的实战项目开发实例,并讲述了 DSP+FPGA 复杂系统的设计。本书提供的所有电路全部可实现,所有程序在设计的实验板上均已调试通过。

本书附光盘,包括:书中程序源代码、开发板电路图源文件以及常用网站地址,敬请读者参考。本书书友会网址:<http://www.hellodsp.com/bbs/forum.php?mod=forumdisplay&fid=95>(红尘有约的专栏)或者 <http://www.study-bbs.com/forum-95-1.html>,读者可以在书友会中了解到更多与本书相关的信息。

本书可以作为本科生和研究生学习 DSP 的教材,也可以作为 DSP 开发人员、广大电子制作爱好者的参考书。

图书在版编目(CIP)数据

手把手教你学 DSP: 基于 TMS320C55x / 陈泰红, 任胜杰, 魏宇, 编著. — 北京: 北京航空航天大学出版社, 2011. 8

ISBN 978 - 7 - 5124 - 0505 - 9

I. ①手… II. ①陈… ②任… ③魏… III. ①数字信号处理②数字信号—微处理器—高等学校—教材 IV.

①TN911.72②TP332

中国版本图书馆 CIP 数据核字(2011)第 130361 号

版权所有,侵权必究。

手把手教你学 DSP——基于 TMS320C55x

陈泰红 任胜杰 魏宇 编著

责任编辑 刘星

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:emsbook@gmail.com 邮购电话:(010)82316936

北京时代华都印刷有限公司印装 各地书店经销

*

开本:787×1 092 1/16 印张:22.75 字数:582 千字

2011 年 8 月第 1 版 2011 年 8 月第 1 次印刷 印数:4 000 册

ISBN 978 - 7 - 5124 - 0505 - 9 定价:46.00 元(含光盘 1 张)

若本书有倒页、脱页、缺页等印装质量问题,请与本社发行部联系调换。联系电话:(010)82317024

序

送给勤奋的探索者

有幸接到作者的邀请,要我为他的新书写序,奈何这几个月来一直沉溺于凡尘俗务,竟不得一块完整时间能静下心来完成这篇序,实在令我自己汗颜。

人其实都是如此,有一种惯性和惰性存在。处于一种工作状态时间长了,对于这个状态就习惯了,了解得多,做起来也比较轻松了,所以就懒得再去改变,因为换一种工作状态又要去熟悉和努力,这就是惯性和惰性。但人处于一种状态久了,就容易固步自封,磨灭心志,而时间就这样在浑浑噩噩中虚度了。对于有志向的年轻人来讲,正确的工作方法就是过一段时间停下来看一下,把自己过去这一阶段的工作总结一下,看看自己的方向是否有错误、有偏差。方向性的错误要及时调整,小的偏差则进行微调,这样才能保证自己在走向成功的路上尽量少走弯路。而方向的确定,就需要我们做一个勤奋的探索者,不断地实践,不断地尝试,分析得失,分析自己的优点和缺点,从而找到一条最适合自己的路。

很高兴看到作者就是这样一个勤奋的探索者。作者从大学二年级开始做项目,经历研究生阶段,一直到现在工作,这些项目的开发让作者积累了丰富的实践经验,从而使得这本书更具有实战指导性,而不仅仅是空乏的理论和枯燥的文字。

TMS320C55x DSP 是德州仪器(TI)C5000 DSP 系列里的新一代产品,C55x 对 C54x 有很好的继承性,与 C54x 源代码兼容,使用户代码有很好的可移植性,从而可以减少用户在软件上的重复投资。DSP 对于一般学习者而言,门槛显得稍微有点高,市面上关于 DSP 的书籍也不像单片机和 ARM 那么多,现在比较多的书籍还是基于早期的 TMS320C5402 DSP,而 DSP 的发展日新月异,新的工艺催生了更多功能更强大的 DSP 芯片的诞生,书籍配套显得有点跟不上发展。此书的出版可以填补这样一个缺失,使学习者多了一个实用的参考书。

作者曾主持了专门针对 TMS320VC5509A 的开发板 Easy5509 的开发工作,对于此系列芯片的开发和应用非常熟悉。本书的内容涵盖了硬件设计、软件设

计、CCS 开发环境、常用算法、DSP/BIOS、工程实施等内容,并且基于 Easy5509 开发板有详尽的实验指导,可以说非常实用,是一本不可多得的 DSP 入门书籍。

勤奋,是成功之路的必然态度;探索,是实践经验的积累手段。宝剑锋从磨砺出,梅花香自苦寒来。让我们做一个勤奋的探索者,在自我不断修正的过程中实现自我的价值,达到自我的成功。

开发板之家站长 涛行九天
21IC 知名版主

2011 年 6 月

前 言

笔者读本科的时候,幸运地加入了学院科协以及学校创新实验室,接触电子设计比同级的同学较早一些,关于本人本科时候的经历,已经发表在本人博客(ahong007.2lic.org)《我的大学四年》,本科大四下学期进入研究生实验室,开始做军用以及民用产品的项目,后来一直在公司研发嵌入式领域项目。虽然有本科时候的底子,但在学习 DSP 时也经历一些痛苦的过程。

书店里的书是琳琅满目,但翻译资料比较多、指导入门实践的相对较少,很多书籍不能深入到工程实践中去。即使一些名气比较大的开发板公司,也只是简单地把一些例程交给用户,而没有把学习的方法、学习中一些注意的问题明明白白地说清楚。回想自己学习电子的时候,幸运地得到高年级的学长热心的指点;回想刚开始学单片机的时候,关于单片机的一些书籍也比较丰富,我认为比较好的有平凡老师的《平凡单片机教程》,肖洪兵老师的《跟我学用单片机》等一些比较贴近实际的优秀的入门书籍。电子技术是一门实践性很强的科目,如果只是停留在理论上而没有深入实践,就没有深刻的印象,当然在做项目的时候也不会正确地应用。本人不才,根据自己在做项目中经常使用的一些资料制作了 Easy5509 开发板,并结合自己的研发经验编写了这本书。

本书以 TMS320C55x 系列高性能低功耗 DSP 为主,主要介绍了以数字信号处理器(DSP)为核心的实时数字信号处理器的硬件结构和片内外设,论述了 eXpress 算法标准软件尤其是 CCS 的使用,详细说明了 DSP 与外围接口电路的设计以及最小系统的设计,给出了 DSP 相关软件编程和开发调试,还介绍了 MATLAB 在数字信号处理中的应用和 DSP/BIOS 的基础知识。在介绍功能模块的基础上,列出了相应的实战项目开发实例,并讲述了 DSP+FPGA 复杂系统的设计。本书提供的所有电路全部可实现,所有程序在设计实验板上均已调试通过。

笔者希望通过这本书,能让更多的初学者比较快地入门,也希望能对研发的同行们提供一些借鉴和参考。笔者精选了一些例程和工程实践项目,由浅入深,希望通过这种循序渐进的方式让更多的人受益,让 DSP 开发不再是一件痛苦的事。

在使用本书的时候,笔者建议有一块自己的开发板,在本书所附的光盘中,有

所有与本书相关的程序以及开发板电路图,希望能在开发板上都做一下试验,建立一个感性的认识,让学习 DSP 不再枯燥。笔者建议使用开发板之家(www.study-kit.com)提供的 Easy5509 开发板,本书中实例设计得以完成,此开发板功不可没。此外,TI 公司提供的参考文档是开发 DSP 必须要读的,这些文档在 CCS 安装目录里提供。面对繁杂的 TI 文档,笔者在书中论述了阅读方法和顺序,敬请读者浏览。

笔者在此推荐一个好的 DSP 论坛 HELLODSP(www.hellodsp.com)。这个论坛里有丰富的 DSP 开发资料,而且很多网友无偿义务在此解答,人气比较旺,是学习 DSP 的乐园。另外,联系笔者的 QQ:754944643,可以为读者做解答。

参加本书编写工作的有史厚兰、任忠阁、陈珍、吕会杰、张玉柱等,他们为本书提供了大量资料,进行了大量实验,编写验证了各个应用程序等,在此表示感谢。特别感谢亲爱的陈珍女士,为本书顺利完成提供了有益的帮助与支持。

感谢公司领导赵涛、李国平、李楠等给予的支持与帮助,并提出的一些宝贵意见和建议,同事张穗毅等也给予了很多帮助,其他同事不再一一列举。特别是师傅李国平部长,在工作中的认真负责、敬业与专注,让我铭记于心。再次表示歉意,因为个人的原因离开,希望师傅师母的生活甜甜密密,事业蒸蒸日上。另外感谢李旭峰、赵亚丽、田小飞、刘翔、祁志才、王贞、郭昕、杨海东、赵治强、韩松、陈志洋、麻少磊等参与本书的文稿编辑、图表修改等工作。

特别感谢涛行九天的热情推荐为我作序,很荣幸能与九天合作,敬佩他的为人和处事风格,也仰慕他所做出的一番事业,祝愿其事业蒸蒸日上,家庭生活幸福美满。

本书成书过程中还得到北京航空航天大学出版社编辑人员的大力支持,没有他们的帮助,完成本书是不可想象的工作;在这里还要感谢所有与出版此书相关的工作人员,他们参与了编写、校对和录入工作。感谢无名网友在网络上提供的资料,因诸多资料无法考证作者来源,如出版中涉及著作权,请联系笔者。

本书可以作为本科生和研究生学习 DSP 的教材,也可作为 DSP 开发人员、广大电子制作爱好者的参考书。本书附光盘,包括:书中程序源代码、开发板电路图源文件以及常用网站地址,敬请读者参考。本书书友会网址:<http://www.hellodsp.com/bbs/forum.php?mod=forumdisplay&fid=95>(红尘有约的专栏)或者 <http://www.study-bbs.com/forum-95-1.html>,读者可以在书友会中了解到更多与本书相关的信息。

由于笔者水平有限,难免会出现一些疏漏,敬请谅解,有兴趣的朋友可发送邮件到:ahong007@yeah.net,与本书笔者进行交流;也可发送邮件到:bhcbstlx@sina.com,与本书策划编辑进行交流。

陈泰红

2011年4月于北京

目 录

第 1 章 绪 论	1
1.1 数字信号处理器简介	1
1.2 数字信号处理器的发展	2
1.3 DSP 处理器的性能指标及选择	3
1.4 DSP 系统的开发	4
1.4.1 DSP 系统设计开发流程	4
1.4.2 DSP 系统软硬件开发工具	5
第 2 章 TMS320C55x 的硬件结构	6
2.1 C55x DSP 简介	6
2.2 C55x 的总体结构.....	6
2.2.1 CPU 内核	6
2.2.2 C55x 存储空间	8
2.2.3 C55x 片内外设	9
2.2.4 C55x 低功耗特性.....	10
2.3 C55x 的封装和引脚功能	10
2.3.1 引脚信号定义与描述	12
2.3.2 存储空间与引脚设置	15
2.4 中断和复位操作.....	15
2.4.1 中 断	15
2.4.2 中断向量与优先级	16
2.4.3 不可屏蔽中断	18
2.4.4 外部中断使用举例	18
第 3 章 eXpressDSP 算法标准软件	20
3.1 CCS 集成开发环境	20
3.2 实时操作系统内核 DSP/BIOS	21
3.3 CSL 概述	24
3.3.1 CSL 体系结构	24

3.3.2	CSL 命名规则	26
3.3.3	通用 CSL 函数	26
3.3.4	CSL 宏	28
3.3.5	CSL 调用	29
3.4	XDAIS 算法标准	29
3.5	eXpress DSP 参考框架	30
3.5.1	RF3 简述	31
3.5.2	RF5 简述	32
3.6	TI 官方文档资源介绍	35
第 4 章	CCS 集成开发环境	37
4.1	开发与开发步骤	37
4.1.1	代码的开发方法	37
4.1.2	开发工具	38
4.1.3	开发步骤	38
4.2	CCS 简介	38
4.2.1	CCS 版本支持	39
4.2.2	CCS 基本功能	39
4.3	CCS3.3 软件的安装与 USB 仿真驱动设置	39
4.3.1	CCS 文件的安装	39
4.3.2	CCS 文件的简单说明	40
4.3.3	目标板与驱动的安装设置	40
4.4	CCS 集成开发环境	43
4.4.1	CCS 集成开发环境概述	43
4.4.2	DSP 程序的仿真模式	43
4.4.3	CCS 菜单详解	44
4.5	建立 DSP 工程文件	46
4.5.1	创建、打开和关闭工程	46
4.5.2	编辑、编译和执行程序	48
4.5.3	调试工具	51
4.6	CCS 开发中的一些问题	55
4.6.1	DSP 型号和 CCS 版本之间的关系	55
4.6.2	run 和 animate 的区别	56
4.6.3	Probe Point 和 Break Points 的区别和联系	56
4.6.4	CCS 文件数据的格式	56
4.6.5	CCS 调试中的一些小技巧	58
4.7	第一个试验：驱动一个 LED	60
第 5 章	TMS320C55x 的片内外设、接口及应用	65
5.1	时钟发生器	65
5.1.1	时钟发生器概况	65

5.1.2	时钟工作模式	66
5.1.3	CLKOUT 输出	67
5.1.4	使用方法与举例	67
5.2	通用定时器	68
5.2.1	定时器概述	68
5.2.2	工作原理	70
5.2.3	定时器应用实例	70
5.2.4	通用定时器的调试	71
5.3	通用 I/O 口	72
5.3.1	GPIO 概述	72
5.3.2	GPIO 使用举例	72
5.4	外部存储器接口	73
5.4.1	EMIF 存储器概述	73
5.4.2	对存储器的考虑	75
5.4.3	存储器接口设计	75
5.4.4	EMIF 中的控制寄存器	79
5.4.5	使用举例	79
5.5	多通道缓冲串口	82
5.5.1	McBSP 概述	82
5.5.2	McBSP 组成框图	82
5.5.3	采样率发生器	83
5.5.4	多通道模式选择	85
5.5.5	异常处理	85
5.5.6	McBSP 寄存器	86
5.5.7	McBSP 使用举例	90
5.6	模/数转换器	92
5.6.1	ADC 的结构和时序	92
5.6.2	ADC 的寄存器	93
5.6.3	ADC 实例	94
5.7	看门狗定时器	95
5.7.1	看门狗定时器概述	95
5.7.2	看门狗定时器的配置	97
5.7.3	看门狗定时器的寄存器	97
5.7.4	看门狗应用举例	98
5.8	I ² C 模块	99
5.8.1	I ² C 模块工作原理	100
5.8.2	I ² C 寄存器	103
5.8.3	I ² C 程序使用举例	103
5.9	USB 模块	105

5.9.1	USB 协议简介	105
5.9.2	VC5509A USB 的硬件资源	106
5.9.3	USB 时钟发生器	108
5.9.4	VC5509A USB 的总线连接与中断	108
5.9.5	USB CSL 模块简介	109
5.9.6	USB 模块的编程实现	111
第 6 章	DSP 系统的硬件设计	113
6.1	DSP 电源的选择与设计	113
6.1.1	电源加电顺序	116
6.1.2	电源检测与复位	116
6.1.3	复位电路设计	117
6.2	JTAG 接口设计	118
6.3	时钟电路的设计	119
6.3.1	时钟电路的分类	120
6.3.2	时钟电路选择原则	121
6.3.3	C55x 时钟电路的调试	121
6.4	总线隔离与驱动的器件选择	121
6.5	Flash 自举引导设计	122
6.5.1	自举引导模式的配置	122
6.5.2	引导表	123
6.5.3	EMIF 模式引导	124
6.5.4	I ² C 模式引导	124
6.6	DSP 多机通信接口选择与设计	124
6.6.1	通过双口 RAM(或双向 FIFO)桥接	125
6.6.2	通过主机接口实现通信	125
6.6.3	通过 GPIO 实现通信	126
6.6.4	通过 I ² C 实现通信	126
6.6.5	通信方式的优缺点	126
6.7	应用系统的低功耗设计	127
6.7.1	合理选择 DSP 器件	127
6.7.2	让 DSP 以适当的速度运行	127
6.7.3	在软件设计中降低功耗	128
6.7.4	存储器类型对功耗的影响	128
6.7.5	正确处理外围电路	128
第 7 章	DSP 软件程序设计	129
7.1	DSP 软件开发流程	129
7.1.1	软件开发流程	129
7.1.2	DSP 程序的基本组成	131

7.1.3 VC5509A 的工作流程	131
7.1.4 DSP C 语言简介	132
7.1.5 DSP C 语言关键字	133
7.1.6 动态分配内存	134
7.2 汇编伪指令	134
7.2.1 汇编伪指令概述	135
7.2.2 C 程序在 DSP 中的定位	137
7.3 CMD 文件的编写	139
7.4 混合编程	143
7.5 GEL 文件	146
7.5.1 GEL 语法概述	146
7.5.2 存储器映射	148
7.5.3 详解 C5509.gel 文件	148
7.6 归档器的使用	150
7.7 反汇编的实现	151
第 8 章 软件开发进阶	154
8.1 Big Endian 和 Little Endian	154
8.2 程序的优化	155
8.2.1 获得最佳性能的代码开发流程	156
8.2.2 工程层的优化	157
8.2.3 算法的改变	159
8.2.4 数据内存的优化	159
8.2.5 提高流水线的效率	159
8.3 程序的编程素养	160
8.3.1 程序注释	160
8.3.2 函 数	163
8.3.3 变 量	164
8.3.4 其他编程规范	165
8.4 数字信号处理库	165
8.4.1 DSPLIB 的调用	166
8.4.2 DSPLIB 函数	167
8.5 图像/视频算法库	168
8.5.1 图像/视频算法库概述	168
8.5.2 图像/视频处理库的安装与使用	168
8.5.3 图像处理算 API 接口	169
第 9 章 DSP 最小系统电路详解	172
9.1 供电电路	172
9.2 时钟振荡电路	173
9.3 峰鸣器控制电路	173

9.4	外扩存储 SDRAM 电路	174
9.5	DSP bootloader 模式电路	175
9.6	SD 卡接口电路	176
9.7	音频控制电路	177
9.8	USB 控制电路	178
第 10 章	硬件电路的设计与调试	179
10.1	完整的硬件开发流程	179
10.1.1	原理图设计	181
10.1.2	PCB 设计注意事项	182
10.1.3	总线等效交换	182
10.1.4	硬件调试前电路板的常规检查	182
10.1.5	调试中遇到问题的解决步骤	183
10.1.6	JTAG 连接错误常用解决办法	183
10.2	碰到问题的常用解决办法	184
10.3	CCS 调试中常见错误信息	184
10.4	电路的抗干扰设计	187
10.4.1	干扰的来源与结果	187
10.4.2	系统电源干扰设计	187
10.4.3	硬件抗干扰设计	188
第 11 章	软件实验详解——扩展篇	189
11.1	SPI bootloader 实验	189
11.1.1	EEPROM 的读/写	190
11.1.2	DSP bootLoader 烧写步骤	193
11.2	USB 自举实验	194
11.3	音频 Codec 实验	198
11.4	SD 卡读/写实验	202
11.4.1	SD 卡简介	202
11.4.2	SD 卡读/写的实现	203
11.5	SDRAM 读/写实验	206
11.5.1	SDRAM 简述	206
11.5.2	配置 EMIF 访问 SDRAM	207
11.5.3	SDRAM 配置与初始化	207
11.5.4	SDRAM 的刷新	208
11.5.5	SDRAM 的读/写操作	209
11.6	12864 图形液晶显示实验	211
11.6.1	简介与型号选型	211
11.6.2	电路接口	212
11.6.3	底层驱动函数	213

第 12 章 数字信号处理算法与实践	214
12.1 基于 MATLAB 的 DSP 调试方法	215
12.1.1 定点数的定标	215
12.1.2 误差问题	216
12.2 CCSLink	217
12.2.1 CCSLink 简介	217
12.2.2 对象的建立	219
12.3 FDATool	219
12.3.1 FDATool 的设置	220
12.3.2 CCS 中滤波器的设计	221
12.4 FIR 滤波器的设计	221
12.4.1 使用 FDATool	221
12.4.2 利用 MATLAB 产生噪声信号用于滤波器测试	223
12.4.3 在 CCS 中编写 FIR 滤波器程序	224
12.4.4 滤波器仿真测试	225
12.5 快速傅里叶变换(FFT)的 DSP 实现	226
12.6 数字滤波器的 DSP 实现	228
12.6.1 FIR 滤波器与 IIR 滤波器的比较	229
12.6.2 FIR 滤波器的设计方法	230
12.6.3 IIR 滤波器的设计方法	232
12.7 FIRLMS 滤波器	234
12.8 数字图像处理	236
12.8.1 图像与 CCS 数据的转换	237
12.8.2 CCS 读取 bmp 文件	237
第 13 章 DSP/BIOS 实践与应用	239
13.1 操作系统与 DSP/BIOS 基础	239
13.1.1 操作系统简介	239
13.1.2 DSP/BIOS 简介	240
13.1.3 DSP/BIOS 组成	240
13.1.4 DSP/BIOS 内核	243
13.1.5 DSP/BIOS 启动过程	244
13.2 DSP/BIOS 的配置	245
13.2.1 建立 DSP/BIOS 配置文件	246
13.2.2 全局属性设置	248
13.2.3 MEM 设置	250
13.2.4 CLK 设置	250
13.2.5 Synchronization 设置	251
13.2.6 Input/Output 设置	252
13.3 HWI 模块的使用	252

13.4	SWI 模块的使用	254
13.4.1	SWI 概述	254
13.4.2	SWI 设置	255
13.4.3	API 函数接口说明	256
13.4.4	SWI 举例说明	256
13.5	TSK 模块的使用	257
13.5.1	TSK 模块概述	257
13.5.2	TSK 模块的设置	257
13.5.3	TSK 模块的接口函数	258
13.5.4	TSK 使用举例	261
13.5.5	阻塞和中断的区别和联系	263
13.6	SEM 模块的使用	264
13.6.1	SEM 模块概述	264
13.6.2	SEM 的接口函数	264
13.6.3	SEM 举例说明	264
13.7	MBX 模块的使用	269
13.7.1	Mailbox 的接口函数说明	269
13.7.2	使用举例	270
13.8	其他常用模块的使用	274
13.8.1	LOG 模块	274
13.8.2	LCK 模块	277
13.8.3	PRD 模块	277
13.8.4	QUE 模块	279
13.9	DSP/BIOS 实时检测与软件优化	280
13.9.1	DSP/BIOS 实时检测	280
13.9.2	DSP/BIOS 软件优化	282
第 14 章	工程项目实践与应用	284
14.1	基于 BIOS 的实时数据采集和处理	284
14.1.1	任务的划分	284
14.1.2	软件实现	285
14.2	DSP 与网络数据传输	288
14.2.1	常用网络芯片简介	288
14.2.2	开源网络协议的移植	291
14.2.3	TCP/IP 协议代码实现	293
14.3	Telnet 协议的实现	293
14.3.1	Telnet 协议简介	293
14.3.2	Telnet 的实现	295
14.3.3	Telnet 协议代码实现	295
14.4	TFTP 协议的实现	296

14.4.1	TFTP 协议简介	296
14.4.2	TFTP 的实现	298
14.5	SD 卡与文件系统	302
14.5.1	解读 FAT32 文件系统	302
14.5.2	文件系统的移植	310
第 15 章	DSP+FPGA 复杂系统设计	315
15.1	FPGA 与 DSP 结构特点	315
15.1.1	DSP 的结构特点	315
15.1.2	FPGA 的结构特点	316
15.1.3	DSP 和 FPGA 的性能比较	316
15.1.4	DSP+FPGA 系统设计	318
15.2	FPGA 系统的设计	318
15.2.1	FPGA 系统设计流程	318
15.2.2	FPGA 最小系统的设计	321
15.2.3	VHDL 语言概述	322
15.2.4	FPGA 系统的调试	323
15.2.5	MAX II 系列芯片与 Quartus II	324
15.2.6	FPGA 常用思想与技巧	325
15.3	FPGA 与 DSP 数据交互	326
15.3.1	EMIF 接口方式	326
15.3.2	HPI 接口方式	327
15.3.3	跨时钟域的设计	329
15.3.4	DSP 与 FPGA 的数据交互	330
15.4	DSP 系统的扩展设计	330
15.4.1	EPM240T100C5 电路设计	331
15.4.2	数码管与 LED 显示实验	332
15.4.3	按键实验	333
15.4.4	串口扩展实验	335
15.4.5	LCD 液晶模块显示实验	337
15.4.6	EPM240 的程序实现	342
附 录	下载 DSP 资料的一些常用网站	345
后 记	346
参考文献	348

第 1 章

绪 论

1.1 数字信号处理器简介

DSP(Digital Signal Processor)芯片是指一类专用于数字信号处理的高速器件,广泛应用于实时快速进行信号处理的场合。DSP 芯片的特定应用目标使它具有的结构与通用处理器有明显不同。体系结构上的不断创新,是 DSP 芯片拥有惊人高性能的一个重要原因。

DSP 芯片一般具有如下主要特点:

① 在一个指令周期内可完成一次乘法和一次加法。通常,DSP 芯片都拥有专用的高精度并行高速硬件乘法器。当两个数相乘时,全精度的乘积位数应等于两个操作数的位数之和。一般的处理器只保留一定的有效位数,而大多数 DSP 芯片结构允许用户保留全精度乘积。DSP 芯片均配置有专为数字信号处理设计的 DSP 指令集,且指令系统具有高度并行性、灵活性和强有力的间接寻址功能。

② 以 PC 机广泛使用的 x86 为代表的通用微处理器,其程序代码和数据共用一个公共的存储空间,这样的结构称为冯·诺依曼结构(Von Neumann Architecture),它只具有有限的存储带宽。DSP 芯片则采用了与之完全不同的存储结构——修正的哈佛结构(Modified Harvard Architecture),即程序和数据空间分开、可以同时访问指令和数据、允许使用寄存器到寄存器操作指令,这使得 DSP 芯片具有非常高的存储带宽。

③ 在内部设计上采用快速存储器设计,大量使用多端口存储器,设置多个存储区域,采用流水线操作,加快指令执行速度,这些都极大地提高了 DSP 器件的性能。

④ 具有低开销或无开销循环及跳转的硬件支持。

⑤ 快速的中断处理和硬件 I/O 支持。

⑥ 具有在单周期内操作的多个硬件地址产生器。

⑦ 可以并行执行多个操作。

⑧ 支持流水线(pipeline)操作,使取指、译码和执指等操作可以重叠进行。

随着 3G 技术和 Internet 的发展,要求处理器的速度越来越高、体积越来越小,DSP 的发展正好能满足这一发展的要求,因为传统的其他处理器都有不同的缺陷,如 MCU 的速度较慢,CPU 体积较大、功耗较高,嵌入 CPU 的成本较高。DSP 的发展使得在许多速度要求较高、算法较复杂的场合,DSP 取代 MCU 或其他处理器,而其成本也有可能更低。