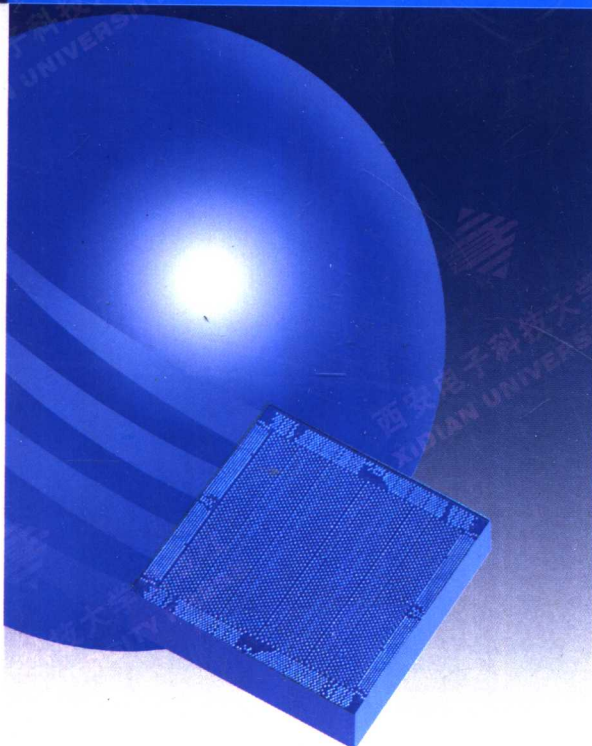


ALTERA®

Nios II Software Architecture Analysis

Nios II 软件架构解析

蔡伟纲 编著



西安电子科技大学出版社
<http://www.xduph.com>

Nios II 软件架构解析

蔡伟纲 编著

西安电子科技大学出版社

2007

内 容 简 介

本书以 Altera 公司的技术资料为蓝本,结合作者多年的 Nios 软件开发经验及长期在 EDACN 专业论坛 Nios II 版块担任版主积累的心得,首先从复位、中断、编程环境、设备驱动以及应用程序二进制接口等五个方面对 Nios II 软件架构进行横向揭密,接着又从汇编语法、makefile、链接脚本、调试手段以及开发技巧等多个角度对 Nios II 软件架构进行纵向解析,纵横交错层层揭开了 Nios II 软件架构的真面目,因而本书既可作为 Nios II 嵌入式软件开发的中高阶指导书,也可作为 Nios II 软件开发手册。

本书书名虽为“Nios II 软件架构解析”,但内容涉及嵌入式软件开发的方方面面,主要面向学习或从事嵌入式软件开发的学生和工程技术人员,也可作为相关专业研究生和高年级本科生的教材。

图书在版编目(CIP)数据

Nios II 软件架构解析 / 蔡伟纲编著. —西安: 西安电子科技大学出版社, 2007.11

ISBN 978-7-5606-1922-4

I. N… II. 蔡… III. 微处理器—系统设计 IV. TP332

中国版本图书馆 CIP 数据核字(2007)第 142316 号

策 划 戚文艳

责任编辑 戚文艳

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

<http://www.xduph.com> E-mail: xdupfb@pub.xaonline.com

经 销 新华书店

印刷单位 西安文化彩印厂

版 次 2007 年 11 月第 1 版 2007 年 11 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 15.625

字 数 350 千字

印 数 1~4000 册

定 价 24.00 元

ISBN 978-7-5606-1922-4/TP·0995

XDUP 2214001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜,谨防盗版。

前 言

随着大规模集成电路技术的不断发展,嵌入式计算机系统开始了从 MCU 逐步过渡到 SOC 的新阶段。SOPC(System On a Programmable Chip, 片上可编程系统)是 Altera 公司提出来的一种灵活、高效的 SOC 解决方案。它将处理器、存储器、IO 接口、各种外围设备等系统设计需要的部件,集成到一个可编程逻辑器件上,构建成一个可编程的片上系统,具有灵活的设计方式,可裁减、可扩充、可升级,并具备软、硬件在系统可编程的功能。SOPC 技术大量采用 IP 复用、软硬件协同设计、自顶向下和自底向上混合设计的方法,使得整个设计在规模、可靠性、功能、性能指标、上市周期、开发成本、产品维护及硬件升级等多方面实现了最优化。

Nios II 处理器是 Altera 公司推出的 32 位 RISC 嵌入式处理器,它是 SOPC 的核心。具有超过 200 DMIPS 的性能,在 FPGA 中的实现成本最低只有 35 美分。处理器以软核形式实现,具有高度的灵活性和可配置性。结合 Altera 提供的配套开发工具 Quartus II、SOPC Builder 以及 Nios II IDE,可以在几分钟内实现一个完善的系统。

本书结合作者多年应用 Nios 的经验及长期在 EDACN 专业论坛 Nios II 版块担任版主职务积累的心得,从 Nios 嵌入式软件开发的各个角度层层揭开了 Nios II 软件架构的真面目。

本书适合于电子工程、计算机、自动控制专业,以及对此类专业有兴趣的高年级本科生和研究生学习或自学 Nios II 嵌入式软件开发。读者只要具备单片机硬件基础及 C 语言软件基础,并结合本书勤于实践,即可在短时间内成为 Nios II 嵌入式软件开发的高手。

本书得到 Altera 上海代表处的大力支持,高级应用工程师陈燕女士在书稿编写过程中一直给予热心的帮助并提出很多宝贵的意见和建议,资深应用工程师何伟欣无私地贡献了自己的研究成果为本书添加精彩,大学计划的徐平波经理在整个出版过程中给予了热心的帮助和督促,在此一并致敬和表示谢意。另外,还要感谢好友匡晓霞、李琦、杨喆、张烨以及广大网友对我的支持和鼓励!

由于水平有限,文中难免存在不妥之处,敬请读者批评指正。

蔡伟纲

2007 年 5 月 31 日

导 读

版本

本书基于 Nios II IDE (Version: 6.0, Build: 93) 编写。

章节框架

本书内容基于“程序”框架编写，既有主程序也有函数库。前 5 章包括中断揭密篇、复位揭密篇、HAL 揭密篇、外设驱动篇以及应用程序二进制接口篇，是本书论述的重点，它们一起构成了主程序部分。而后续的各章节则构成了函数库，供主程序调用，它们的存在是为了让读者能更好地消化和吸收前 5 个章节的内容。

源文件阅读须知

本书在解读 HAL 源文件时大量引用了 Nios II 软件架构中的预处理器、特殊宏定义、汇编伪指令以及链接脚本中定义的符号，为了避免在源文件中大量地重复介绍这些知识点，也为了读者更好地阅读和理解源文件，笔者把这些知识点分门别类，归纳为独立的四个章节：软件架构中的预处理器、软件架构中的特殊宏定义、汇编语言相关语法以及链接脚本 (Linker Script)。希望读者在阅读本书源文件之前能对这些章节有所了解，并结合这几个章节解读源文件。

最佳阅读流程

本书大部分知识具有很强的应用背景，读者很难一下子理解文中所要传达的全部内容，因此本书的最佳阅读流程是：首先通读全文，不求深入理解和掌握文中的每个知识点，但求对各章节有个感性认识，以便在实际项目开发或自学过程中碰见问题后，能判定问题属性并结合目录快速定位相关知识，此时再对知识点进行仔细咀嚼、消化和吸收。



最好的学习方法：理论结合实践，带着问题学习！

笔者在读大学的时候，为了自学单片机，曾经反反复复、一遍又一遍地阅读关于 51 单片机的书籍，可是一年过后，却还是觉得两手空空，什么也不会。后来有幸参加全国电子设计竞赛，仅半个月时间，对 51 单片机就能够运用自如了。所以，不管学习什么东西，能够实际动手并带着问题学习将起到事半功倍的效果。

学好 Nios II 的前提

- 单片机硬件基础。
- 较强的 C 语言编程能力。
- 较强的调试能力(调试能力远比编程能力重要)。
- 初级汇编语言基础。
- 初步了解 GNU 编译器工具链的工作原理。
- 了解 Quartus II 的开发流程。
- 良好的英文阅读能力，因为所有 Nios 的权威资料都来自 Altera 的官方网站：
<http://www.altera.com.cn/literature/lit-nio2.jsp>
- 一个 Nios II 参考开发板，最好是出自于 Altera 公司。
- 一个要求用 Nios II 处理器开发的项目，没有项目也不要紧，两者的区别在于：
 - 前者的学习过程是：碰见问题→解决问题→提高自己；
 - 后者的学习过程是：构思问题→解决问题→提高自己。

获取 Nios II 的相关帮助

求助方式

- Altera 官方网站有一个 mysupport 网页，用户可以提交问题，目前已支持中英文版本
<https://mysupport.altera.com/eservice/>
- Nios 的官方论坛 www.niosforum.com，是个英文网站，里面都是 Nios 的高手，其中有很多版主都是 Nios 的鼻祖。
 - 向 Altera 的 FAE 求助。
 - 如果你们的产品量很大，或者是个大公司，就可以向 Altera 或者代理商直接求助：
<http://www.altera.com.cn/corporate/contact/con-index.html>
 - 国内也有几个不错的 Nios 论坛：
<http://www.edacn.net/bbs/forum-540-1.html>
<http://bbs.eccn.com/ecbbs/index.asp?boardid=9>

如果希望尽快得到别人的回复，问题的描述是非常重要的。为了得到最快的响应，最好在提问时说明下面的详细信息：

- Quartus II 版本。
- Nios II IDE 版本。

- Quartus II 或 Nios II IDE 的详细出错信息。
- 问题的详细描述。
- 尝试过的解决办法。
- 很多时候还需要提供项目的源文件。

自助方式

➤ Nios II IDE 集成的帮助系统，主要提供与 IDE 相关的帮助。打开 Nios II IDE，选择菜单：Nios II IDE->Help->Help Contents，此时会弹出一个 Help 网页，该网页支持搜索功能，可以输入要查找信息的关键字，得到相关信息。

➤ Altera 的官方提供了一个 Find Answer 的网站，这个网站汇总了与 Altera 相关的问题及其解答；并且该网站除了简单的关键字搜索外还采用了 Google 技术的高级搜索功能：

[http: //www.altera.com/support/spt-answers.html](http://www.altera.com/support/spt-answers.html)

➤ 在安装完 Nios II IDE 之后，系统会在<Nios II EDS install path>目录下生成一个名为 documents 的文件夹，该文件夹蕴涵了非常丰富的内容，特别是该目录下面的 index.htm 文件，它把 Nios II 软件开发的相关文档分门别类，并提供了超链接。

网络资源

➤ Altera 官方的 Nios II 相关文档下载网址：

[http: //www.altera.com.cn/literature/lit-nio2.jsp](http://www.altera.com.cn/literature/lit-nio2.jsp)

➤ Altera 官方的 ftp 资源，提供 Quartus II，Nios II IDE 等软件的各种版本下载：

[ftp: //ftp.altera.com/outgoing/release/](ftp://ftp.altera.com/outgoing/release/)

➤ Altera 官方网站上提供的 Nios II 相关的视频 demo，对初学者非常有帮助：

[http: //www.altera.com/education/demonstrations/online/embedded-processor/onl-nios2.html](http://www.altera.com/education/demonstrations/online/embedded-processor/onl-nios2.html)

➤ Altera 提供的技术培训课程很多是免费的：

[http: //mysupport.altera.com/etraining/etraining.asp](http://mysupport.altera.com/etraining/etraining.asp)

➤ 基于 niosforum 的 nioswiki，虽然目前内容不多，但是相信随着时间的推移，一定会成为 Nios 资源的集中营：

[http: //nioswiki.jot.com](http://nioswiki.jot.com)

本书的维护和升级

本书内容基于“程序”框架编写，目录相当于程序框架，文中的超链接相当于子程序调用，而前面“内容简介”以及“导读”等章节则相当于程序的注释。同样，为了再版的需要，本书也需要制定维护和升级的策略。

信息反馈途径

书籍邮箱: CaiWG.Nios II @Gmail.com

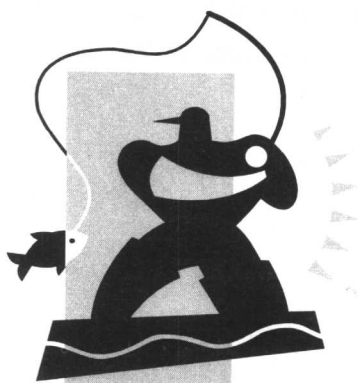
书籍博客: [http: //Nios II .cublog.cn](http://Nios II .cublog.cn)

奖励方式

凡反馈了重大 bug，并最终被收录到下一版书籍的读者，都将被列入下一版书籍的感谢者名单并免费获赠新版书籍 1 本，所以反馈 bug 信息的读者不要忘了留下详细的通信地址和邮编以及真实姓名。

感言

一本不错的书可以让你得到很多“鱼”，而一本真正的好书不但可以让你得到“鱼”，更重要的是教会了你“渔”。



从提笔开始写这本书到最终的完成已经整整一年半了，写书的过程就是自学的过程：从刚开始被.section 等汇编伪指令弄得晕头转向，到中间为了弄懂 makefile、linker script 而在网上苦苦搜索好几个月，再到最终整本书的完成，经历了无数次无知的烦恼与得解的欢愉，最贴切而简单的心情描述应该是“痛并快乐着”吧！

联系作者

本书有一个博客网站 <http://NiosII.cublog.cn>，上面包含了本书的样章、源码以及本书的更新和勘误。除此之外，作者的所有原创文章、范例、实验以及傻瓜式教学资料都将在博客发布，欢迎广大读者访问博客。

您可以通过 GaiWG.NiosII@Gmail.com 给作者发送 E-mail。作者会阅读所有的来信，但可能因收信量过大而无法及时回复。

目 录

第一部分 横向揭密

第 1 章 中断揭密篇.....	3
1.1 异常的概念.....	3
1.1.1 中断的引入.....	3
1.1.2 异常的引入.....	3
1.2 与异常相关的术语.....	3
1. application context(应用背景).....	3
2. exception context(异常背景).....	4
3. context switch(背景切换).....	4
4. exception handler(异常 handler).....	4
5. exception overhead(异常开销).....	4
6. hardware interrupt(硬件中断).....	4
7. interrupt service routine(ISR, 中断服务函数).....	4
8. software exception(软件异常).....	4
9. implementation-dependent instruction(依赖于处理器具体实现的指令).....	4
10. unimplemented instruction(未完成指令).....	4
11. invalid instruction(无效指令).....	4
1.3 Nios II 异常处理概述.....	4
1.4 Nios II 处理器的异常地址.....	5
1.5 Nios II 处理器支持的异常类型.....	5
1.6 Nios II 处理器的异常触发机理.....	6
1. 硬件中断的触发机理.....	6
2. 软件异常的触发机理.....	6
1.7 Nios II 处理器对异常的响应.....	7
1. 异常的硬件响应.....	7
2. 异常的软件响应.....	7
1.8 异常 handler 架构.....	7
1. 顶层异常 handler.....	8
2. 硬件中断 handler.....	8
3. 软件异常 handler.....	9
4. 外设中断的 ISR.....	10

1.9 中断向量表	12
1.10 中断控制器初始化	13
1.11 中断的调试	13
1.12 中断响应性能分析	14
1. 中断性能指标参数	14
2. 影响中断性能的因素	14
3. 中断性能测试实例	15
1.13 Nios II 异常处理小结	15
1.14 Nios II 异常处理注意事项	16
1. 软件异常与硬件中断返回地址的区别	16
2. 异常 handler 代码存储布局	16
3. 异常 handler 代码编译控制	17
1.15 源文件	19
1. alt_exception_entry.S	19
2. alt_irq_entry.S	26
3. alt_exception_trap.S	28
4. alt_exception_muldiv.S	29
5. alt_irq_register.c	30
6. alt_irq_handler.c	31
7. alt_irq_entry.h	33
8. alt_irq_table.h	33
9. alt_irq.h	34
10. alt_irq_vars.c	38
11. nios2.h	38
第 2 章 复位揭密篇	41
2.1 复位的概念	41
2.2 Nios II 处理器的复位地址	41
2.3 Nios II 处理器的复位响应	42
1. 复位的硬件响应	42
2. 复位的软件响应	42
2.4 Nios II 处理器的复位状态	43
2.5 Nios II 的启动代码	44
1. 代码段入口_start	44
2. 硬件描述层的 C 入口函数 alt_main	45
2.6 Nios II 处理器的 Bootloader	46
1. Bootloader 承担的任务	46
2. Altera 提供的 Bootloader	46
3. Bootloader 的模式	48

4. Elf2flash	48
2.7 Nios II 系统的 Boot 过程分析	49
1. 从 EPCS Flash 中 Boot	50
2. 从 CFI Flash 中 Boot	51
3. Bootloader 源文件	53
2.8 Nios II 处理器复位小结	54
2.9 源文件	55
1. Crt0.S	55
2. alt_main.c	60
3. walt_sys_init.c	63
4. alt_set_args.h	64
5. alt_load.c	65
6. alt_load.h	67
第 3 章 HAL 揭密篇	68
3.1 HAL 系统库概览	68
1. 主要功能	69
2. 通用设备模型	69
3. C 标准库——Newlib	70
3.2 基于 HAL 的项目架构	70
3.3 HAL 源码的组织架构	71
1. HAL 元件	71
2. HAL 元件的目录结构	72
3. HAL 头文件	72
4. HAL 源文件	72
5. HAL 外设头文件	72
3.4 HAL 编程环境	73
1. 系统描述文件 System.h	73
2. HAL 定义的数据类型	76
3. UNIX 风格接口函数	76
4. HAL 文件系统	77
5. 应用程序入口	78
6. 存储空间布局	78
7. 堆和栈空间	80
8. 全局变量指针寄存器	81
9. 缓存(Cache)	81
10. 紧耦合存储器(Tightly-Coupled Memory)	84
第 4 章 外设驱动篇	86
4.1 使用 HAL 外设驱动	86

1. 使用 HAL 外设总体思路.....	86
2. HAL 支持的外设.....	86
3. 使用字符模式设备.....	87
4. 使用文件子系统.....	88
5. 使用定时器设备.....	88
6. 使用 Flash 设备.....	93
7. 使用 DMA 设备.....	100
8. PIO 驱动详解.....	104
9. ioctl 函数.....	110
4.2 HAL 外设自动初始化原理.....	110
1. 系统初始化文件 alt_sys_init.c.....	110
2. alt_sys_init.c 的生成.....	111
3. 宏定义*_INSTANCE 和*_INIT.....	112
4.3 开发 HAL 外设驱动.....	112
1. 开发符合通用设备模型的 HAL 外设驱动.....	112
2. 开发自定义 HAL 外设驱动.....	120
4.4 集成 HAL 外设驱动.....	121
4.5 小封装驱动.....	121
4.6 命名空间.....	121
4.7 重载设备驱动.....	122
第 5 章 应用程序二进制接口篇	125
5.1 概述.....	125
5.2 数据类型约定.....	125
5.3 内存对齐约定.....	126
5.4 寄存器使用约定.....	126
1. 寄存器保护.....	127
2. Caller-Saved 寄存器.....	127
3. Callee-Saved 寄存器.....	127
5.5 Endianess.....	128
5.6 栈(stack)与帧(frame)空间.....	129
1. 帧结构.....	129
2. 消除帧指针.....	131
3. 帧空间的开辟与释放.....	132
5.7 函数参数与返回值接口.....	132
1. 函数参数.....	132
2. 函数返回值.....	133
5.8 栈空间代码示例.....	134
1. 源码.....	134

2. 反汇编代码.....	135
---------------	-----

第二部分 纵向解析

第 6 章 IDE 属性配置	141
6.1 系统库属性配置	141
1. 菜单操作	141
2. 属性配置示例	141
3. 属性介绍	142
6.2 C/C++ Build 属性配置	144
1. 菜单操作	144
2. 属性配置示例	144
3. 属性介绍	144
6.3 Preferences 属性配置	145
1. 菜单操作	145
2. 属性配置示例	146
3. 属性介绍	146
第 7 章 软件调试技巧	148
7.1 观察寄存器	148
7.2 观察变量	148
7.3 观察内存	149
7.4 观察表达式	150
7.5 设置断点	150
7.6 Watchpoint	151
7.7 在 HAL 源文件中设置断点	152
7.8 指令单步调试模式	153
7.9 观察复杂的结构变量	154
7.10 观察 .objdump 文件	154
7.11 Trace	155
7.12 Profiling	156
第 8 章 软件开发技巧	157
8.1 优化代码尺寸	157
1. 使能编译优化选项	157
2. 使用小封装设备驱动	157
3. 减少文件描述器的数目	158
4. 使用/dev/null 设备名	158
5. 使用 UNIX 风格的 IO 函数取代标准 C 函数	158

6. 使用精简版本 NewLib C 标准库	159
7. 消除未用的设备驱动	159
8. 使用 _exit 替代 exit 函数	160
9. 禁止指令仿真功能	160
8.2 开发和使用库文件	161
1. 库文件的优点	161
2. Nios II IDE 支持的库项目类型	161
3. 创建库项目	161
4. 如何在 Nios II IDE 中使用库文件	161
8.3 使用自定义的 Linker Script	162
8.4 重载 HAL 源码	163
8.5 自定义代码和数据属性	164
8.6 使用 asm/C/C++ 混合编程	165
1. C 与 C++ 混合编程范例	166
2. C 与 ASM 混合编程范例	167
3. 内联汇编范例	168
第 9 章 软件架构中的预处理器	171
9.1 如何在 Nios II IDE 中控制全局预处理器	171
9.2 Nios II 软件架构中的预处理器	172
1. ALT_NO_BOOTLOADER	172
2. ALT_USE_SMALL_DRIVERS	173
3. ALTERA_AVALON_JTAG_UART_SMALL	173
4. ALT_EXCEPTION_STACK	173
5. ALT_STACK_CHECK	174
6. ALT_NO_INSTRUCTION_EMULATION	174
7. ALT_TRAP_HANDLER	174
8. ALT_NO_CLEAN_EXIT	175
9. NIOS2_ICACHE_SIZE	175
10. ALT_ASM_SRC	175
11. ALT_SIM_OPTIMIZE	176
12. NIOS2_HAS_DEBUG_STUB	176
第 10 章 软件架构中的特殊宏定义	177
10.1 语言符号粘贴操作符(##)	177
10.2 __ALT_LOAD_SECTIONS, _ALT_LOAD_SECTIONS, ALT_LOAD_SECTIONS	177
10.3 ALT_LOAD_SECTION_BY_NAME	178
10.4 <COMPONENT_NAME>_INSTANCE	179
10.5 <COMPONENT_NAME>_INIT	180

第 11 章 汇编语言相关语法	181
11.1 Nios II 汇编器的特殊宏	181
1. %lo	181
2. %hi	181
3. %hiadj.....	181
4. %gprel	181
11.2 Nios II 汇编器特殊宏的应用示范.....	181
1. 把一个 32 位的立即数 IMMED32 赋予寄存器.....	181
2. 在汇编指令中寻址全局变量.....	182
11.3 Nios II 指令集.....	182
11.4 GCC 汇编语言语法知识.....	182
第 12 章 链接脚本(Linker Script).....	184
12.1 链接脚本基本概念	184
12.2 链接脚本的相关语法	185
1. ENTRY	185
2. INCULDE	185
3. INPUT	185
4. OUTPUT_FORMAT	185
5. OUTPUT_ARCH.....	185
6. KEEP	185
7. PROVIDE.....	186
8. MEMORY	186
9. (.)	187
10. LOADADDR	187
11. SECTIONS.....	187
12.3 Nios II 链接脚本中定义的符号.....	188
1. __alt_stack_pointer	188
2. __alt_stack_limit.....	188
3. __alt_exception_stack_pointer.....	188
4. __alt_exception_stack_limit	188
5. __alt_heap_start	188
6. __alt_heap_end	189
7. __bss_start.....	189
8. __bss_end.....	189
9. _gp.....	189
12.4 在源文件中使用链接脚本定义的符号.....	189
12.5 Linker Script 输入节点	190
1. 默认节点属性.....	190