

# 《自己动手写CPU(含CD光盘1张)》

## 图书基本信息

书名：《自己动手写CPU(含CD光盘1张)》

13位ISBN编号：9787121239507

出版时间：2014-9-1

作者：雷思磊

页数：556

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：[www.tushu111.com](http://www.tushu111.com)

# 《自己动手写CPU(含CD光盘1张)》

## 内容概要

《自己动手写CPU(含CD光盘1张)》使用Verilog HDL设计实现了一款兼容MIPS32指令集架构的处理器——OpenMIPS。OpenMIPS处理器具有两个版本，分别是教学版和实践版。教学版的主要设计思想是尽量简单，处理器的运行情况比较理想化，与教科书相似，便于使用其进行教学、学术研究和讨论，也有助于学生理解课堂上讲授的知识。实践版的设计目标是能完成特定功能，发挥实际作用。

《自己动手写CPU(含CD光盘1张)》分为三篇。第一篇是理论篇，介绍了指令集架构、Verilog HDL的相关知识。第二篇是基础篇，采用增量模型，实现了教学版OpenMIPS处理器。首先实现了仅能执行一条指令的处理器，从这个最简单的情况出发，通过依次添加，实现逻辑操作指令、移位操作指令、空指令、移动操作指令、算术操作指令、转移指令、加载存储指令、协处理器访问指令、异常相关指令，最终实现了教学版OpenMIPS处理器。第三篇是进阶篇，通过为教学版OpenMIPS添加Wishbone总线接口，从而实现了实践版OpenMIPS处理器，并与SDRAM控制器、GPIO模块、Flash控制器、UART控制器、Wishbone总线互联矩阵等模块组成一个小型SOPC，然后下载到FPGA芯片以验证实现效果，最后为实践版OpenMIPS处理器移植了嵌入式实时操作系统 $\mu$ C/OS-II。

《自己动手写CPU(含CD光盘1张)》适合计算机专业的学生、FPGA开发人员、处理器设计者、嵌入式系统应用开发工程师、MIPS平台开发人员以及对处理器内部的实现感兴趣的读者阅读，也可以作为高等院校计算机原理、计算机体系结构等课程的实践参考书。

# 《自己动手写CPU(含CD光盘1张)》

## 作者简介

雷思磊，理工男，好静，倡导低碳生活，常以环保人士自居，喜读书，自幼笃信“博观而约取厚积而薄发”，是故，所读书籍甚为驳杂，年近而立，尚不确定根本兴趣目标，一日，驻足书架之前，细览所读书籍，惊觉随岁月增长，关注点依次从应用编程、操作系统、驱动设计转移至处理器结构，此一脉络极其清晰，遂如醍醐灌顶，幡然醒悟，原来余根本兴趣目标在“底层”，在于从根本上理解世界之运行，遂耗数年时间钻研处理器工作原理，乃有些微收获，拙作当为数年辛苦之小结，然学无止境，科技发展亦日新月异，唯有持续钻研，方能大成，屈子曰：路漫漫其修远兮，吾将上下而求索。此言甚是，余定谨记而遵行之。

# 《自己动手写CPU(含CD光盘1张)》

## 书籍目录

### 第一篇 理论篇

#### 第1章 处理器与MIPS

2

##### 1.1 计算机的简单模型

2

###### 1.1.1 计算机的简单组成模型

2

###### 1.1.2 计算机的简单使用模型

3

##### 1.2 架构与指令集

4

###### 1.2.1 CISC与RISC

4

###### 1.2.2 主要的几种ISA

5

##### 1.3 MIPS指令集架构的演变

6

##### 1.4 MIPS32指令集架构简介

9

###### 1.4.1 数据类型

9

###### 1.4.2 寄存器

9

###### 1.4.3 字节次序

10

###### 1.4.4 指令格式

11

###### 1.4.5 指令集

11

###### 1.4.6 寻址方式

13

###### 1.4.7 协处理器CP0

13

###### 1.4.8 异常

14

##### 1.5 本书的目标与组织方式

14

#### 第2章 可编程逻辑器件与Verilog HDL

16

##### 2.1 可编程逻辑器件概述

16

##### 2.2 基于PLD的数字系统设计流程

18

###### 2.2.1 设计输入

19

###### 2.2.2 综合

20

2.2.3 布局布线	20
2.2.4 下载	20
2.2.5 仿真	20
2.2.6 工具介绍	21
2.3 Verilog HDL简介	21
2.4 Verilog HDL中模块的结构	22
2.5 Verilog HDL基本要素	24
2.5.1 常量	24
2.5.2 变量声明与数据类型	24
2.5.3 向量	26
2.5.4 运算符	26
2.6 Verilog HDL行为语句	29
2.6.1 过程语句	29
2.6.2 赋值语句	31
2.6.3 条件语句	32
2.6.4 循环语句	34
2.6.5 编译指示语句	35
2.6.6 行为语句的可综合性	37
2.7 电路设计举例	38
2.8 仿真	41
2.8.1 系统函数	42
2.8.2 Test Bench	43
2.8.3 ModelSim仿真	45
2.9 本章小结	48
第二篇 基础篇	

## 第3章 教学版OpenMIPS处理器蓝图

50

### 3.1 系统设计目标

50

#### 3.1.1 设计目标

50

#### 3.1.2 五级流水线

50

#### 3.1.3 指令执行周期

52

### 3.2 教学版OpenMIPS处理器接口

53

### 3.3 文件说明

54

### 3.4 实现方法

55

## 第4章 第一条指令ori的实现

58

### 4.1 ori指令说明

58

### 4.2 流水线结构的建立

59

#### 4.2.1 流水线的简单模型

59

#### 4.2.2 原始的OpenMIPS五级流水线结构

60

#### 4.2.3 一些宏定义

62

#### 4.2.4 取指阶段的实现

63

#### 4.2.5 译码阶段的实现

65

#### 4.2.6 执行阶段的实现

74

#### 4.2.7 访存阶段的实现

78

#### 4.2.8 回写阶段的实现

81

#### 4.2.9 顶层模块OpenMIPS的实现

81

### 4.3 验证OpenMIPS实现效果

85

#### 4.3.1 指令存储器ROM的实现

85

#### 4.3.2 最小SOPC的实现

87

#### 4.3.3 编写测试程序

88

#### 4.3.4 建立Test Bench文件

89	
4.3.5	使用ModelSim检验OpenMIPS实现效果
90	
4.4	MIPS编译环境的建立
92	
4.4.1	VisualBox的安装与设置
93	
4.4.2	GNU工具链的安装
96	
4.4.3	使用GNU工具进行编译
97	
4.4.4	使用GNU工具进行链接
99	
4.4.5	得到ROM初始化文件
102	
4.4.6	编写Makefile文件
103	
4.5	第一条指令实现小结
105	
第5章	逻辑、移位操作与空指令的实现
107	
5.1	流水线数据相关问题
107	
5.2	OpenMIPS对数据相关问题的解决措施
111	
5.3	测试数据相关问题的解决效果
115	
5.4	逻辑、移位操作与空指令说明
115	
5.5	修改OpenMIPS以实现逻辑、移位操作与空指令
119	
5.5.1	修改译码阶段的ID模块
120	
5.5.2	修改执行阶段的EX模块
127	
5.6	测试程序1——测试逻辑操作实现效果
129	
5.7	测试程序2——测试移位操作与空指令实现效果
130	
5.8	小结
131	
第6章	移动操作指令的实现
132	
6.1	移动操作指令说明
132	
6.2	移动操作指令实现思路
133	
6.2.1	新的数据相关情况的解决
135	

6.2.2 系统结构的修改	136
6.3 修改OpenMIPS以实现移动操作指令	137
6.3.1 HI、LO寄存器的实现	137
6.3.2 修改译码阶段的ID模块	138
6.3.3 修改执行阶段	141
6.3.4 修改访存阶段	146
6.3.5 修改回写阶段	149
6.3.6 修改OpenMIPS顶层模块	149
6.4 测试程序	150
第7章 算术操作指令的实现	152
7.1 简单算术操作指令说明	153
7.2 简单算术操作指令实现思路	156
7.3 修改OpenMIPS以实现简单算术操作指令	157
7.3.1 修改译码阶段的ID模块	157
7.3.2 修改执行阶段的EX模块	164
7.4 测试简单算术操作指令实现效果	171
7.5 流水线暂停机制的设计与实现	174
7.5.1 流水线暂停机制的设计	174
7.5.2 流水线暂停机制的实现	175
7.6 乘累加、乘累减指令说明	180
7.7 乘累加、乘累减指令实现思路	181
7.8 修改OpenMIPS以实现乘累加、乘累减指令	182
7.8.1 修改译码阶段的ID模块	182
7.8.2 修改执行阶段的EX模块	184
7.8.3 修改EX/MEM模块	



189
7.8.4 修改OpenMIPS模块
190
7.9 测试乘累加、乘累减指令实现效果
190
7.10 除法指令说明
191
7.11 除法指令实现思路
192
7.11.1 试商法
192
7.11.2 实现思路
193
7.11.3 系统结构的修改
193
7.12 修改OpenMIPS以实现除法指令
194
7.12.1 增加DIV模块
194
7.12.2 修改译码阶段的ID模块
199
7.12.3 修改执行阶段的EX模块
200
7.12.4 修改OpenMIPS模块
204
7.13 测试除法指令实现效果
204
7.14 数据流图的修改
205
第8章 转移指令的实现
206
8.1 延迟槽
206
8.2 转移指令说明
207
8.3 转移指令实现思路
210
8.3.1 实现思路
210
8.3.2 数据流图的修改
210
8.3.3 系统结构的修改
211
8.4 修改OpenMIPS以实现转移指令
212
8.4.1 修改取指阶段的PC模块
212
8.4.2 修改译码阶段
213

8.4.3 修改执行阶段的EX模块	223
8.4.4 修改OpenMIPS模块	225
8.5 测试转移指令的实现效果	225
8.5.1 测试跳转指令	225
8.5.2 测试分支指令	227
第9章 加载存储指令的实现	230
9.1 加载存储指令说明	230
9.1.1 加载指令lb、lbu、lh、lhu、lwi说明	230
9.1.2 存储指令sb、sh、swi说明	231
9.1.3 加载存储指令用法示例	232
9.1.4 加载指令lwl、lwrl说明	233
9.1.5 存储指令swl、swrl说明	235
9.2 加载存储指令实现思路	238
9.2.1 数据流图的修改	239
9.2.2 系统结构的修改	240
9.3 修改OpenMIPS以实现加载存储指令	240
9.3.1 修改译码阶段	240
9.3.2 修改执行阶段	247
9.3.3 修改访存阶段	249
9.3.4 修改OpenMIPS顶层模块	260
9.4 修改最小SOPC	261
9.4.1 添加数据存储器RAM	262
9.4.2 修改最小SOPC	264
9.5 测试程序	265
9.6 链接加载指令ll、条件存储指令sc说明	

267	
9.7 II、sc指令实现思路	
269	
9.7.1 II、sc指令实现思路	
269	
9.7.2 数据流图的修改	
270	
9.7.3 系统结构的修改	
271	
9.8 修改OpenMIPS以实现II、sc指令	
271	
9.8.1 LLbit寄存器的实现	
271	
9.8.2 修改译码阶段的ID模块	
273	
9.8.3 修改访存阶段	
275	
9.8.4 修改OpenMIPS模块	
279	
9.9 测试II、sc指令实现效果	
279	
9.10 load相关问题	
281	
9.10.1 load相关问题介绍	
281	
9.10.2 解决方法	
281	
9.11 修改OpenMIPS以解决load相关问题	
283	
9.11.1 修改译码阶段的ID模块	
283	
9.11.2 修改OpenMIPS模块	
284	
9.12 测试load相关问题解决效果	
285	
9.13 小结	
286	
第10章 协处理器访问指令的实现	
287	
10.1 协处理器介绍	
287	
10.2 协处理器CP0中的寄存器	
288	
10.3 协处理器CP0的实现	
295	
10.4 协处理器访问指令说明	
300	
10.5 协处理器访问指令实现思路	
300	

10.5.1 实现思路	300
10.5.2 数据流图的修改	301
10.5.3 系统结构的修改	301
10.6 修改OpenMIPS以实现协处理器访问指令	303
10.6.1 修改译码阶段	303
10.6.2 修改执行阶段	305
10.6.3 修改访存阶段	311
10.6.4 修改OpenMIPS模块	314
10.7 测试程序	314
第11章 异常相关指令的实现	316
11.1 MIPS32架构中定义的异常类型	316
11.2 精确异常	318
11.3 异常处理过程	319
11.4 异常相关指令介绍	321
11.4.1 自陷指令	321
11.4.2 系统调用指令syscall	324
11.4.3 异常返回指令eret	325
11.5 异常处理实现思路	325
11.5.1 实现思路	325
11.5.2 修改数据流图	326
11.5.3 修改系统结构	326
11.6 修改OpenMIPS以实现异常处理	328
11.6.1 修改取指阶段	328
11.6.2 修改译码阶段	330
11.6.3 修改执行阶段	

339	
11.6.4	修改访存阶段
346	
11.6.5	修改协处理器CP0
352	
11.6.6	修改控制模块CTRL
355	
11.6.7	修改OpenMIPS
357	
11.7	再次修改最小SOPC
358	
11.8	测试程序
359	
11.8.1	测试程序1——测试系统调用异常
359	
11.8.2	测试程序2——测试自陷异常
361	
11.8.3	测试程序3——测试时钟中断
364	
11.9	教学版OpenMIPS处理器实现小结
366	
	第三篇 进阶篇
	第12章 实践版OpenMIPS处理器设计与实现
368	
12.1	实践版OpenMIPS处理器的设计目标
368	
12.2	Wishbone总线介绍
370	
12.2.1	Wishbone总线接口说明
370	
12.2.2	Wishbone总线单次读操作的过程
372	
12.2.3	Wishbone总线单次写操作的过程
373	
12.2.4	SEL_O/SEL_I信号说明
374	
12.3	实践版OpenMIPS处理器接口
375	
12.4	实践版OpenMIPS处理器的实现思路
376	
12.5	从教学版OpenMIPS到实践版OpenMIPS
378	
12.5.1	Wishbone总线接口模块的实现
378	
12.5.2	修改CTRL模块
384	
12.5.3	修改OpenMIPS顶层模块
386	
12.6	实践版OpenMIPS处理器实现小结

386	
第13章 基于实践版OpenMIPS的小型SOPC	
387	
13.1 小型SOPC的结构	
387	
13.2 Wishbone总线互联矩阵WB_CONMAX	
388	
13.3 GPIO	
390	
13.4 UART控制器	
392	
13.4.1 UART简介	
392	
13.4.2 UART16550 IP核介绍	
394	
13.5 Flash控制器	
398	
13.5.1 Flash简介	
398	
13.5.2 Flash控制器的设计	
399	
13.5.3 Flash控制器的实现	
400	
13.6 SDRAM控制器	
403	
13.6.1 SDRAM简介	
403	
13.6.2 SDRAM CONTROLLER IP核	
409	
13.7 实现基于实践版OpenMIPS的小型SOPC	
412	
13.8 本章小结	
423	
第14章 验证实践版OpenMIPS处理器	
424	
14.1 DE2平台简介	
424	
14.2 测试需要的硬件连接	
425	
14.3 QuartusII工程建立	
426	
14.4 测试步骤说明	
430	
14.5 测试一——GPIO实验	
431	
14.5.1 测试内容	
431	
14.5.2 测试程序	
431	

14.5.3 编译测试程序	432
14.5.4 将测试程序写入Flash芯片	433
14.5.5 下载小型SOPC到DE2	435
14.5.6 测试效果	435
14.6 测试二——UART实验	435
14.6.1 测试内容	435
14.6.2 测试程序	436
14.6.3 测试效果	438
14.7 测试三——模拟操作系统的加载过程	439
14.7.1 测试内容	439
14.7.2 测试程序BootLoader	439
14.7.3 测试程序SimpleOS	443
14.7.4 将测试程序写入Flash	446
14.7.5 测试效果	448
14.8 本章小结	449
第15章 为OpenMIPS处理器移植 $\mu$ C/OS-II	450
15.1 为什么需要操作系统	450
15.2 嵌入式实时操作系统介绍	451
15.3 $\mu$ C/OS-II简介	452
15.4 $\mu$ C/OS-II特点	452
15.5 $\mu$ C/OS-II的几个概念	454
15.5.1 任务	454
15.5.2 任务调度	456
15.5.3 任务切换	456
15.5.4 $\mu$ C/OS-II的中断处理	

# 《自己动手写CPU(含CD光盘1张)》

457	
15.5.5	时钟节拍
457	
15.5.6	$\mu$ C/OS-II的初始化
458	
15.5.7	$\mu$ C/OS-II的启动
458	
15.6	$\mu$ C/OS-II的基本功能
458	
15.6.1	任务间的通信与同步
459	
15.6.2	任务管理
459	
15.6.3	时间管理
459	
15.6.4	内存管理
460	
15.7	$\mu$ C/OS-II的文件体系
460	
15.8	$\mu$ C/OS-II的移植条件
461	
15.9	C语言中使用汇编代码
463	
15.10	MIPS函数调用规范
465	
15.10.1	寄存器使用规范
465	
15.10.2	参数传递
466	
15.10.3	函数返回值
466	
15.10.4	堆栈布局
467	
15.10.5	示例
468	
15.11	$\mu$ C/OS-II在OpenMIPS处理器上的移植
470	
15.11.1	文件目录的建立
470	
15.11.2	修改os_cpu.h文件
473	
15.11.3	修改os_cpu_a.S文件
474	
15.11.4	修改os_cpu_c.c文件
496	
15.12	测试程序
500	
15.12.1	创建openmips.h文件
500	



15.12.2 创建openmips.c文件	502
15.13 编译指示文件的建立	508
15.14 OpenMIPS处理器运行移植后的 $\mu$ C/OS-II	515
15.15 本章小结	516
附录A 教学版OpenMIPS各个模块的接口说明	517
A.1 PC模块接口说明	517
A.2 IF/ID模块接口说明	518
A.3 ID模块接口说明	518
A.4 Regfile模块接口说明	520
A.5 ID/EX模块接口说明	520
A.6 EX模块接口说明	521
A.7 DIV模块接口说明	524
A.8 EX/MEM模块接口说明	525
A.9 MEM模块接口说明	527
A.10 MEM/WB模块接口说明	529
A.11 CP0模块接口说明	530
A.12 LLbit模块接口说明	532
A.13 HILO模块接口说明	532
A.14 CTRL模块接口说明	533
附录B OpenMIPS实现的所有指令及对应的机器码	534
B.1 逻辑操作指令	534
B.2 移位操作指令	534
B.3 移动操作指令	535
B.4 算术操作指令	535
B.5 转移指令	

536

B.6 加载存储指令

536

B.7 协处理器访问指令

537

B.8 异常相关指令

537

B.9 空指令及其他指令

537

参考文献

538

# 《自己动手写CPU(含CD光盘1张)》

## 精彩短评

- 1、边看边练，很好
- 2、结构很好。直接从实现最简单的流水线开始，层层递进。代码清楚。

# 《自己动手写CPU(含CD光盘1张)》

## 精彩书评

1、很喜欢这本书，讲解十分详细，比照着代码理解感觉事半功倍，非常感谢作者给我们带来的知识盛宴，希望能彻底理解CPU的内部原理，努力ing.....撕掉处理器“高大上”的标签有助于深入理解计算机组成原理、计算机体系结构从无到有、从小到大，介绍一款处理器的成长过程

# 《自己动手写CPU(含CD光盘1张)》

## 版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:[www.tushu111.com](http://www.tushu111.com)