

版本历史

文档更新记录		文档名:	Lab05_CPU 例外与中断支持	
		版本号	V0.1	
		创建人:	计算机体系结构研讨课教学组	
		创建日期:	2017-11-11	
更新历史				
序号	更新日期	更新人	版本号	更新内容
1	2017/11/11	邢金璋	V0.1	初版。

文档信息反馈: xingjinzhang@loongson.cn

1 实验五 CPU 例外与中断支持

在学习并尝试本章节前，你需要具有以下环境和能力：

- (1) 较为熟练使用 Vivado 工具。
- (2) 一定的 CPU 设计与实现能力。

通过本章节的学习，你将获得：

- (1) MIPS 架构的例外与中断分类。
- (2) CPU 例外相关的系统控制寄存器（CP0 寄存器）的知识。
- (3) MIPS 例外与中断产生、标记和处理的知识。

在本章节的学习过程中，你可能需要查阅：

- (1) 文档“A05_“体系结构研讨课”MIPS 指令系统规范_V0.2”。
- (2) MIPS32 官方文档的卷 II 和卷 III。
- (3) 课本上例外与中断知识。

在开展本次实验前，请确认自己知道以下知识：

- (1) MTC0/MFC0、ERET、SYSCALL 指令的定义和作用。
- (2) 理解 CP0 寄存器 STATUS、CAUSE、EPC、COUNT、COMPARE 的定义和作用。请认真阅读文档“A05_“体系结构研讨课”MIPS 指令系统规范_V0.2”。
- (3) 理解系统调用、断点、地址错、整型溢出、保留指令例外。
- (4) 理解时钟中断、硬件中断、软件中断。
- (5) 理解例外或中断产生时 CPU 的动作，特别是例外发生处为延迟槽指令时的处理方式。

1.1 实验目的

1. 理解 MIPS 架构的例外与中断的处理机制。
2. 初步理解软硬件协同。

1.2 实验设备

1. 装有 Xilinx Vivado、MIPS 交叉编译环境的计算机一台。
2. 龙芯体系结构教学实验箱（Artix-7）一套。

1.3 实验任务

为 myCPU 增加例外与中断支持，完成功能测试，并支持运行一定的应用程序。

本次实验分两周完成，第一周（2017 年 11 月 14 日检查），需要完成：

- (1) CPU 增加 MTC0、MFC0、ERET 指令。
- (2) CPU 增加 CP0 寄存器 STATUS、CAUSE、EPC。
- (3) CPU 增加 SYSCALL 指令，也就是增加 syscall 例外支持。
- (4) 运行功能测试通过。功能测试程序为 lab5_func_1，是在 lab3_func_4 的基础上增加第 69 个功能点测试，也就是 SYSCALL 例外测试。

第二周（2017 年 11 月 21 日检查），需要完成：

-
- (1) CPU 增加 BREAK 指令，也就是增加 break 例外支持，。
 - (2) CPU 增加地址错、整数溢出、保留指令例外支持。
 - (3) CPU 增加 CP0 寄存器 COUNT、COMPARE。
 - (4) CPU 增加时钟中断支持，时钟中断要求固定绑定在硬件中断 5 号上，也就是 CAUSE 对应的 IP7 上。
 - (5) CPU 增加 6 个硬件中断支持，编号为 0~5，对应 CAUSE 的 IP7~IP2。
 - (6) CPU 增加 2 个软件中断支持，对应 CAUSE 的 IP1~IP0。
 - (7) 完成 lab5_fun_2 功能测试。
 - (8) 在 myCPU 上运行 lab4 的电子表程序，要求能实现相同功能。
 - (9) 在 myCPU 上运行记忆游戏程序，要求能正确运行。

1.4 实验环境

本次实验实验环境与 lab3 基本一致，目前是 ucas_CDE_v0.3。

ucas_CDE_v0.3 中 func 放置在 soft 目录下，整个实验过程中陆续发布 lab5_func_1、lab5_func_2。

1.5 实验检查

本次实验在 2017 年 11 月 14 日、2017 年 11 月 21 日分别进行检查。现场分仿真检查和上板检查，类似 lab3 的检查步骤。

本次实验作品提交只要求提交一次，截止日期是 2017 年 11 月 21 日 18:00。提交内容包含实验报告和 myCPU 的 RTL 源码，源码目录需有简单介绍的 readme。

1.6 实验说明

1.6.1 快速功能测试

本次实验的功能测试前 68 项与 lab3 一致，lab5 新增功能测试点依次向后编号。为加快验证本次新增功能点，可以修改 start.S，注释掉前 68 项功能点测试。等新增功能运行无误，再开启前 68 项功能点测试，以便确保新增 RTL 未影响原先的功能。

另外，还可以在仿真时，调整 SoC_lite 里的 pll_clk 的 CPU clk，跳到最高，也可加快仿真。原理和 lab3 性能测试时一致。

1.6.2 例外与中断实现说明

本次实验共需实现系统调用、断点、地址错、整型溢出、保留指令例外，以及时钟中断、硬件中断、软件中断。

关于这些例外与中断的详细描述，以及 CPU 处理机制、CP0 寄存器等请参见文档“A05_“体系结构研讨课”MIPS 指令系统规范_V0.2”。该文档 V0.2 版本已经补充完整系统控制寄存器的描述，关于系统控制寄存器的描述，属于 MIPS 架构的子集，是针对本课程实验整理得到的简化设计描述。请认真阅读该文档，避免运行功能测试时 trace 比对出错。