**组号： 2**

****



计算机系统综合课程设计

方案及认证报告

东南大学计算机科学与工程学院

20 22 年 10 月

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 组长姓名 | | XXX | | 手机号 |  |
| Email地址 | |  | | | |
| 本组成员情况（含组长） | | | | | |
| 姓 名 | 学 号 | | 教学班 | 承担的任务（预分配） | |
| XXX | 09019xxx | | 2 | OS代码理解、Verilog语言学习、硬件时序模块、驱动模块、BootLoader模块。 | |
| XXX | 09019xxx | | 2 | OS代码理解、汇编语言学习、进程管理模块、视情况开展拓展任务的探索（如进程调度等）。 | |
| XXX | 09019xxx | | 2 | OS代码理解、内存管理模块、视情况开展拓展任务的探索（如内存管理、文件管理等）。 | |
| XXX | 09019xxx | | 2 | OS代码理解、汇编语言学习、MIPS原装文档理解、异常处理模块。 | |
| XXX | 09019xxx | | 2 | OS代码理解、汇编语言学习、内存管理模块。 | |

本组要做的课题（在相应的项上标记🗹）

□ 普通课题

🗹 操作系统特别课题

□ 开放架构C编译特别课题

□ RISC-V处理器及编译特别课题

□ HW特别课题

□ 普通课题提高或挑战

注意：本表所列为本组预计的目标设计方案，具体在实施的时候原则上目标不能删减，如要删减，需经过教师同意。技术路线可以根据实际情况修改，技术路线的修改无需教师认可。

本报告页数不够可以增加页数。

|  |
| --- |
| 本组设计的预计目标描述（含所有预计模块的功能） |
| （主要是CPU做到什么程度、软件包括哪几项，预计可做较高要求或更高要求的哪些部分。特别课题的小组要达到的目标是什么，准备做哪些可选项目）  底层硬件时序：   * 完成键盘、数码管等硬件设备的IP核构建。 * 保证硬件—驱动—BootLoader—OS内核—Shell的逻辑通畅运行。   驱动：   * 补全驱动代码，保证各个硬件设备正常运行。   BootLoader：   * 完成一系列初始化操作。 * 完善OS自举启动的逻辑。   异常处理：   * 完成OS的异常处理逻辑。   内存管理：   * 完善页表、TLB等数据结构。 * 完善OS的页式内存管理体系。 * 完善TLB缓存页表的逻辑。   进程管理：   * 完善进程控制块等数据结构。 * 完善进程上下文切换等逻辑。 * 完善进程调度算法。   拓展任务：   * 视情况开展探索，可能探索的课题包括但不限于：VGA显示屏、文件系统、内存管理、网络系统。 |
| 本组设计的技术路线 |
| （主要是上述设计的各个模块的可行性分析，以及自己准备采用什么样的思路去完成该模块，包括准备采用什么工具、语言等）  底层硬件时序：   * 工具链：vivado。 * 语言：Verilog硬件语言。 * 帮助材料：慕课。   驱动：   * 工具链：vscode、makefile。 * 语言：C语言。 * 帮助材料：指导手册、框架代码。   BootLoader：   * 工具链：vscode、makefile。 * 语言：MIPS32汇编语言、C语言。 * 帮助材料：指导手册、框架代码、MIPS32原装文档。   异常处理：   * 工具链：vscode、makefile。 * 语言：MIPS32汇编语言、C语言。 * 帮助材料：指导手册、框架代码、MIPS32原装文档。   内存管理：   * 工具链：vscode、makefile。 * 语言：MIPS32汇编语言、C语言。 * 帮助材料：指导手册、框架代码、MIPS32原装文档、OS教材。   进程管理：   * 工具链：vscode、makefile。 * 语言：MIPS32汇编语言、C语言。 * 帮助材料：指导手册、框架代码、OS教材。   打通整个逻辑：   * 工具链：两块硬件板、vivado、OpenGCD、PuTTY、makefile、gdb。 * 帮助材料：指导手册。 |

|  |
| --- |
| 预计可能会遇到的问题及解决方案 |
| （人员上或者技术上可能会遇到困难的或需要解决的问题及解决的方案）   * 困难：人手不足/时间精力不够。   + 解决方案：①在任务开始阶段，进行合理的统筹协调，减少不合理的任务分工、可避免的工作量浪费、内耗；②尽力在完成度/拓展度与工作量上做出权衡，避免死磕解决不了的问题。 * 困难：框架代码难以理解。   + 解决方案：多跟老师、助教学长、其他小组交流，多参考网络上的文档、论坛等。 * 困难：自我感觉代码没问题，但就是跑不起来。   + 解决方案：①多跟老师、助教学长、其他小组交流，多参考网络上的文档、论坛等；②尽力在完成度与debug时间上做出权衡，避免死磕解决不了的问题。 * 困难：慕课太难，慕课工作量太大。   + 解决方案：①多跟老师、助教学长、其他小组交流，多参考网络上的文档、论坛等；②在有限的时间精力约束下，尽力在主要任务完成度与慕课完成度上做出权衡，以OS完成度为重点。 * 困难：Verilog、汇编太难写。   + 解决方案：多跟老师、助教学长、其他小组交流，多参考网络上的文档、论坛等。 |
| 大致的进度安排 |
| （本组大概的进度计划）   |  |  | | --- | --- | | 时间段 | 工作 | | 6-10周 | 理解框架代码、理解指导手册、理解MIPS32原装文档、学习Verilog、汇编等。 | | 10-11周 | 完成硬件时序部分，基于已有的Linux OS，打通整个工具链。 | | 11-15周 | 完成OS代码，视情况开展拓展任务的探索。 | | 15-16周 | 完善报告，准备验收、答辩。 | |  |  | |  |  | |