



中国科学技术大学

University of Science and Technology of China

011174.01: Operating System 操作系统原理与设计

Project 5: TaskManager & FCFS

陈香兰(xlanchen@ustc.edu.cn)

高效智能计算实验室, CS, USTC @ 合肥

嵌入式系统实验室, CS, USTC @ 苏州

2021/6/7

实验5基础



中国科学技术大学
University of Science and Technology of China

- 本实验在实验4的基础上进行
- 在实验4提交的截止时间过后，同学们可以就实验4的内容互通有无
- 实验5可以在其他同学实验4的基础上进行
 - 无论你使用哪一个（包括自己的），请在实验报告中标注，实验5的基础来自哪个同学（可以是自己）
 - 给你使用的实验4打分



- **【必须】任务管理器**
 - **【必须】任务数据结构**
 - **【必须】任务池** **【二选一】数据结构的静态 or 动态管理**
 - **【必须】任务创建/销毁，任务启动和终止**
 - **【必须】上下文切换**
 - **【必须】idle任务、init任务**
 - **【必须】调度器FCFS**
 - **【必须】初始化完成后，转入多任务状态运行**
- **【必须】自测**
 - Shell封装成task来运行
 - 自编测试用例
 - **【可选】shell命令封装成task来运行**

提纲



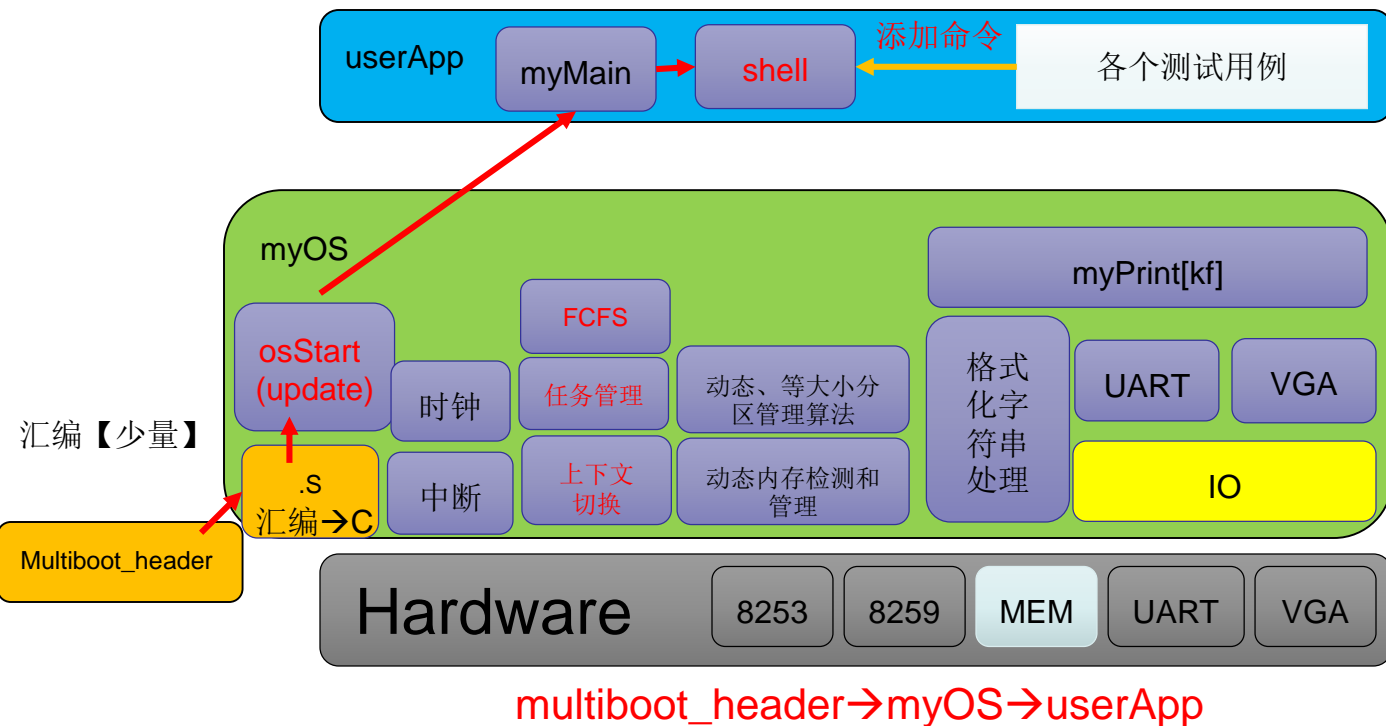
中国科学技术大学
University of Science and Technology of China

1. 软件架构和功能说明
2. 主要功能模块说明
3. 其他模块变更
4. 验收标准

1 软件架构和功能



中国科学技术大学
University of Science and Technology of China



流程:

Multiboot_header

为进入C程序准备好上下文

初始化操作系统各个模块

调用userApp入口
myMain (自测) +shell

主要功能模块【新】:

内核: 上下文切换、任务管理和调度
用户: 新功能测试

测试:

被测功能: 任务创建、FCFS
自测: userApp

主要源码文件（参考）



中国科学技术大学
University of Science and Technology of China

- （演示）
- myOS/i386/CTX_SW.S
- myOS/include/task.h
- myOS/kernel/task.c

2.1 任务管理



- 设计任务数据结构，命名为myTCB
 - myTCB中的具体内容，根据需要自定义
 - 一个任务应当包含下列内容（至少）
 - 栈
 - 状态（就绪、等待、...）
 - 任务ID
 - 调度相关参数
- 提供一个任务池
 - 【二选一】静态：任务数组，任务的个数根据需要配置
或者 动态：使用**kmalloc**，按需分配
 - 创建一个任务时，从任务池中分配一个任务数据结构
 - 销毁一个任务时，将任务数据结构释放回任务池

2.2 任务的创建和销毁



- 实现任务创建原语（接口命名为createTsk()）
 - `int createTsk(void (*tskBody)(void))`
 - 给定任务的入口函数
 - 返回任务ID
- 实现任务销毁原语（接口命名为destroyTsk()）
 - `void destroyTsk(int tskIndex)`
 - 以任务ID为输入参数

2.3 任务的启动和终止



- 实现任务启动原语，接口命名为**tskStart()**

```
void tskStart(myTCB *tsk)
```

- 将指定任务的状态置为就绪，加入就绪队列

- 实现任务终止原语，接口命名为**tskEnd()**

- 当前任务从就绪队列出列，并销毁该任务

2.4 上下文切换



- 实现任务上下文切换原语，CTX_SW()

```
void CTX_SW(void*prev_stkTop, void*next_stkTop);
```

```
.text
```

```
.code32
```

```
.global CTX_SW
```

```
CTX_SW:
```

不需要跟这个一样，仅供参考

```
    pushf
```

```
    pusha
```

```
    movl prevTSK_StackPtr,%eax
```

```
    movl %esp, (%eax)
```

```
    movl nextTSK_StackPtr, %esp
```

```
    popa
```

```
    popf
```

```
    ret
```

2.5 调度相关接口



- 内部接口参考
 - 初始化就绪队列
 - 判断就绪队列是否为空
 - 返回就绪队列上的下一个任务（即将被调度的任务）
 - 入列
 - 出列
 - 调度入口
- 本实验仅仅实现FCFS

2.6 Idle任务和init任务



- 创建idle任务，idle任务的主体是一个死循环，在循环体中，调用schedule
- 创建init任务，init任务的主体（接口命名为initTskBody()）由测试用例提供
 - 我们使用myMain对接initTskBody

2.7 进入多任务运行



- 在任务管理器的各个子模块都初始化完毕后，整个系统转入多任务状态运行
 - 从启动流程转入多任务运行
 - 提供一个启动流程的上下文
 - 找到调度器中的**next**任务
 - 使用上下文切换原语，从启动上下文切换到**next**任务（**next**任务，一般就是**init**任务）

```
unsigned long BspContextBase[STACK_SIZE];
unsigned long *BspContext;

//start multitasking
void startMultitask(void){
    BspContext = BspContextBase + STACK_SIZE -1;
    prevTSK_StackPtr = &BspContext;
    currentTsk = nextFCFSTsk();
    nextTSK_StackPtr = currentTsk->stkTop;
    CTX_SW(prevTSK_StackPtr,nextTSK_StackPtr);
}
```

2.8 任务管理器的初始化



- 修改**osStart**原语，增加任务管理器的初始化
- 任务管理器的初始化
 - 任务池初始化
 - 创建**Idle**任务
 - 初始化**FCFS**调度队列
 - 创建**init**任务
 - 进入多任务状态（上下文切换到**myMain**）

3.1 Shell的变化



中国科学技术大学
University of Science and Technology of China

- **【必须】** 将shell作为一个task
- **【可选】** 将shell中的命令作为一个task

3.2 使用头文件



中国科学技术大学
University of Science and Technology of China

- 随时增加头文件

4 验收标准



中国科学技术大学
University of Science and Technology of China

- 提交：源代码打包 + 实验报告；验收标准如下：
 - 完成源代码编写和调试，能够编译正确运行
 - 实现主流程，提供规定接口
 - 实现主要功能，提供规定接口
 - 将源代码进行合理的组织、提供相应的Makefile，能够生成myOS
 - 提供编译和运行脚本
 - 提交实验报告，实验报告中包括
 - 给出软件的框图，并加以概述
 - 详细说明主流程及其实现，画出流程图
 - 详细说明主要功能模块及其实现，画出流程图
 - 源代码说明（目录组织、Makefile组织）
 - 代码布局说明（地址空间）
 - 编译过程说明
 - 运行和运行结果说明
 - 遇到的问题和解决方案说明

演示



中国科学技术大学
University of Science and Technology of China

Q & A