

## 思考题1

mpidr\_el1存储了核ID，通过ID=0时跳转至primary进行初始化，其它核进入wait\_for\_bss\_clear以及wait\_until\_smp\_enabled等待核0执行

## 思考题4

- C函数中的局部变量会保存在栈上
- 调用函数时返回地址会保存在栈上
- 被调用函数会将某些寄存器保存在栈上

不设置栈可能会导致局部变量无法分配以及函数调用链断开

## 思考题5

clear\_bss将未初始化的全局变量和静态变量设置为0

不清理.bss段可能会导致使用一些全局变量做条件判断或者跳转时出错

## 思考题8

多级页表相较于单级页表的优劣：

1. 优：节约内存，只为实际映射的虚拟地址分配页表
2. 劣：地址翻译开销大，需要多次访问内存，TLB性能下降

映射 0 ~ 4GB 地址范围所需的物理内存大小（或页表页数量）

1. 4KB粒度：

- 需要  $4G / 4K = 2^{20}$  个页
- 需要L3级页表页  $2^{20} / 512 = 2048$  个
- 需要L2级页表页  $2048 / 512 = 4$  个
- 需要L1级页表页 1 个
- 需要L0级页表页 1 个

共2054个页表页

2. 2MB粒度：

- 需要  $4G / 2M = 2048$  个页
- 需要L2级页表页  $2048 / 512 = 4$  个
- 需要L1级页表页 1 个
- 需要L0级页表页 1 个

共6个页表页

## 思考题9

1. 0x00000000 到 0x80000000 以 2MB 块粒度映射，共  $8 * 16^7 = 2GB$  物理内存
2. 需要  $2GB / 2MB = 1024$  个二级页表项， $1024 / 512 = 2$  个二级页表页

3. 需要 1 个一级页表项，1 个一级页表页
4. 需要 1 个零级页表项，1 个零级页表页
5. 共 4 个页表页，占用  $4 * 4KB = 16KB$  内存

## 思考题11

在启用MMU后，PC仍然指向地位地址，所以需要配置低地址页表

## 思考题12

1. 在 main 函数执行完时钟、调度器、锁的初始化之后设置secondary\_boot\_flag，使得其他核可以跳出等待循环，进入内核的主流程
2. 避免多核同时初始化导致的竞争条件和资源冲突