

实验 6 报告

2016K8009909006

刘杰

一、实验任务（10%）

补全 PMON 源代码中串口初始化，TLB 初始化和 Cache 初始化的代码，编译完成在 CPU232 上正常启动并加载启动 Linux 内核。

要求刚下载完 CPU232 的 bit 文件后，PMON 打印信息出现“TLB init OK!!!”和“Cache init OK!!!”。

二、实验设计（30%）

PMON 源代码中需要补全串口初始化，TLB 初始化和 Cache 初始化。编译完成后在线调试，为便于调试，先调通串口后能打印出 TLB 和 Cache 初始化信息。

（一）初始化串口

串口初始化是对串口控制器里的寄存器进行赋值，使得串口可以正常工作。本次实验的串口初始化部分只要完成波特率的设置。

设置波特率相关的寄存器有 UART_TLL，UART_TLH 和 UART_LCR。当 UART_LCR[7]为 1 时，UART_TLL 和 UART_TLH 才组合为分频锁存器。在设置波特率时 UART_LCR 低 7 位写 0。

CPU 时钟频率记为 CPU_CLK，波特率记为 B，则 $\{UART_TLH, UART_TLL\} = CPU_CLK / (16 * B)$ 。波特率选择 B=38400bps，时钟频率 CPU_CLK=33MHZ，计算得 UART_TLH=0x00，UART_TLL=0x36。

由于上述三个寄存器位宽为 8，所以设置寄存器时用 SB 访存指令。

（二）初始化 TLB

TLB 初始化是用 TLBWI 指令填充 TLB 的 32 个表项。在使用 TLBWI 指令前需要先设置好 CP0_ENTRYHI，CP0_PAGEMASK，CP0_ENTRYLO0，CP0_ENTRYLO1，CP0_INDEX。

EntryHi 中，我们需要赋值 VPN2 和 ASID 域，其中 ASID 域赋任意值都可，VPN2 赋值为 0x50000。PageMask 中，赋值全 0 即可。因为 CPU232 实现为 4KB 页大小。EntryLo0 和 EntryLo1 的 PFN 域赋值为任意值均可，但 flags 域建议赋值为全 0。Index 用于索引 TLB，由于 TLB 有 32 项，故 Index 赋初值 0，之后从 0 加到 31 即可。

一次 TLBWI 指令只能依据 Index 初始化一项 TLB，所以需要循环 32 次。

（三）初始化 Cache

Cache 初始化，就是将 Cache 里面的有效位清 0，数据位写 0 或写 1，使得 Cache 进入确定的状态。一般初始化，就是依次向所有的 Cache 行写全 0。而向 Cache 行写的动作可以通过 Cache 指令 Index Store Tag 完成。

Index Store Tag 指令是将 offset(base)的地址[12:12]位作为路号去选择某一路 cache，并将[11:5]位作为索引去寻址该路的 Cache 行，并将 TagLo/TagHi 的值写入到 Cache 行的 Tag 域。

初始化 Cache 使用 Cache 指令，初始化指令 Cache 时 op 域 0x08，初始化数据 Cache 时 op 域 0x09。在初始化前需要先向 TagLo 写入 0。Cache 采用二路组相连，每路 128 个 Cache 块。所以 Cache 指令的 offset(base)地址的[12]作为路号从 0 到 1，offset(base)地址的[11:5]作为索引每路从 0 到 7'b1111111。

三、实验过程（60%）

（一）实验流水账

2018.12.22 13:30-14:00 编写代码，在线调试，PMON 没有打印任何信息，debug

2018.12.22 14:00-14:10 查代码，找出问题修改，初始化 Cache 失败

2018.12.22 14:10-14:30 查代码，测试通过

（二）错误记录

1、错误 1

（1）错误现象

PMON 打印不出信息。

（2）分析定位过程

根据讲义，推测串口初始化失败。重新计算串口分频，发现之前计算错误。

（3）错误原因

由于串口分频计算错误，所以 UART_TLL 和 UART_TLH 寄存器设置错误。

（4）修正效果

正确计算后重新设置 UART_TLL 和 UART_TLH 寄存器，运行后 PMON 能打印信息。

2、错误 2

（1）错误现象

PMON 打印信息显示 Cache 初始化失败

（2）分析定位过程

查看 Cache 初始化相应代码，发现 offset(base)地址设置有误。

(3) 错误原因

因为 `offset(base)` 地址中表示 Index 部分为 [11:5]，但是在设置时误认为是最低 7 位，所以累加时 `offset(base)` 地址只增 0x1。

(4) 修正效果

累加时 `offset(base)` 地址增加 0x20，修改后测试通过。

四、实验总结（可选）

最后一次实验也完成了。体系结构实验课基本上结束了，在这里感谢老师们的帮助。