

CS334 lab4 report

1. 请自己总结执行 `ebreak` 后，我们的操作系统是如何进行断点中断处理的

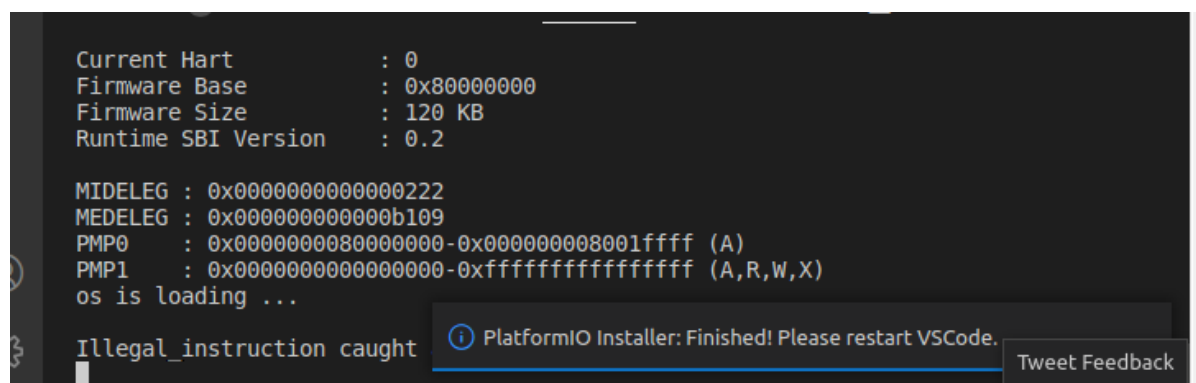
操作系统先是初始化相关寄存器，之后自动进入 `stvec` 寄存器当中存储的函数地址，执行对应的函数。注意在执行对应函数之前需要先保留对应的寄存器状态，包括当前 CPU 的状态。在执行完对应的函数之后，需要对之前保留的寄存器状态以及 CPU 状态进行恢复。

2. 请阅读手册，描述 `epc` 寄存器的作用

当异常发生时，保留发生异常的位置。即发生例外的指令的 PC 会存储进入 `epc`

3. 编程题：触发一条非法指令异常 (ILLEGAL_INSTRUCTION)，在 `kern/trap/trap.c` 的异常处理函数中捕获，并对其进行处理，简单输出异常类型和指令即可。

提示：可以在 S 态执行 `mret` 汇编指令进行触发



```
Current Hart      : 0
Firmware Base    : 0x80000000
Firmware Size    : 120 KB
Runtime SBI Version : 0.2

MIDELEG : 0x0000000000000222
MEDELEG : 0x000000000000b109
PMP0    : 0x0000000080000000-0x000000008001ffff (A)
PMP1    : 0x0000000000000000-0xffffffffffff (A,R,W,X)
os is loading ...

Illegal_instruction caught
```

PlatformIO Installer: Finished! Please restart VSCode. [Tweet Feedback](#)