

循序渐进，学习开发一个 RISC-V 上的操作系统



第 13 章 抢占式多任务

汪辰

- 抢占式多任务
- 抢占式多任务的设计
- 兼容协作式多任务

- **【参考 1】** : The RISC-V Instruction Set Manual , Volume I: Unprivileged ISA , Document Version 20191213
- **【参考 2】** : The RISC-V Instruction Set Manual , Volume II: Privileged Architecture , Document Version 20190608-Priv-MSU-Ratified
- **【参考 3】** : SiFive FU540-C000 Manual , v1p0

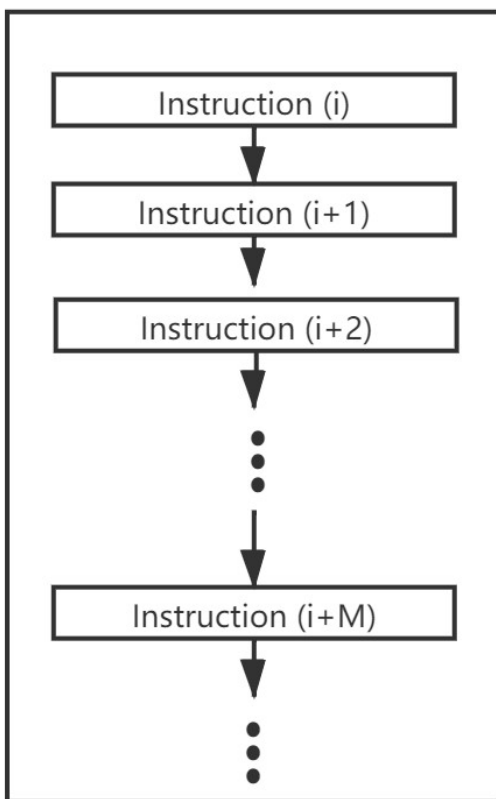
- 抢占式多任务
- 抢占式多任务的设计
- 兼容协作式多任务

- **协作式多任务 (Cooperative Multitasking)：**协作式环境下，下一个进程被调度的前提是当前进程主动放弃时间片。
- **抢占式多任务 (Preemptive Multitasking)：**抢占式环境下，操作系统完全决定进程调度方案，操作系统可以剥夺耗时长的进程的时间片，提供给其它进程。

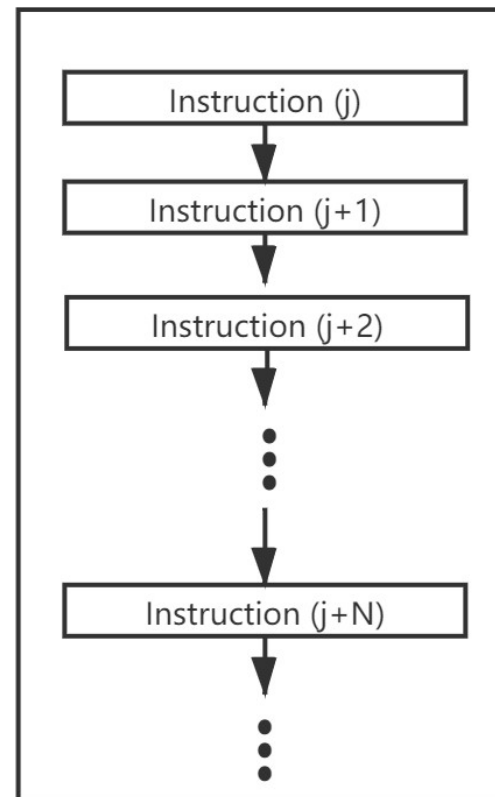
- 抢占式多任务
- 抢占式多任务的设计
- 兼容协作式多任务

抢占式多任务的设计

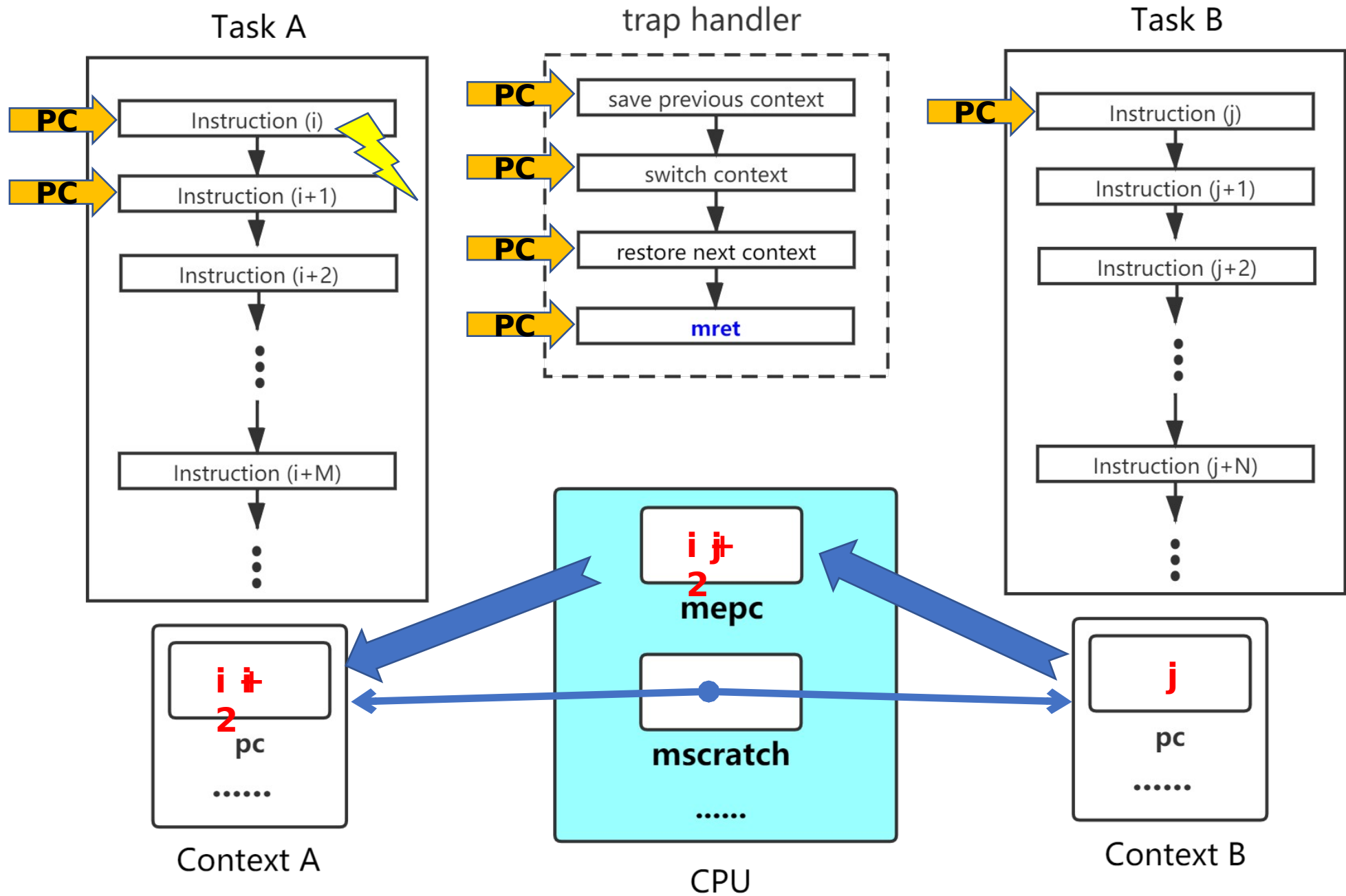
Task A



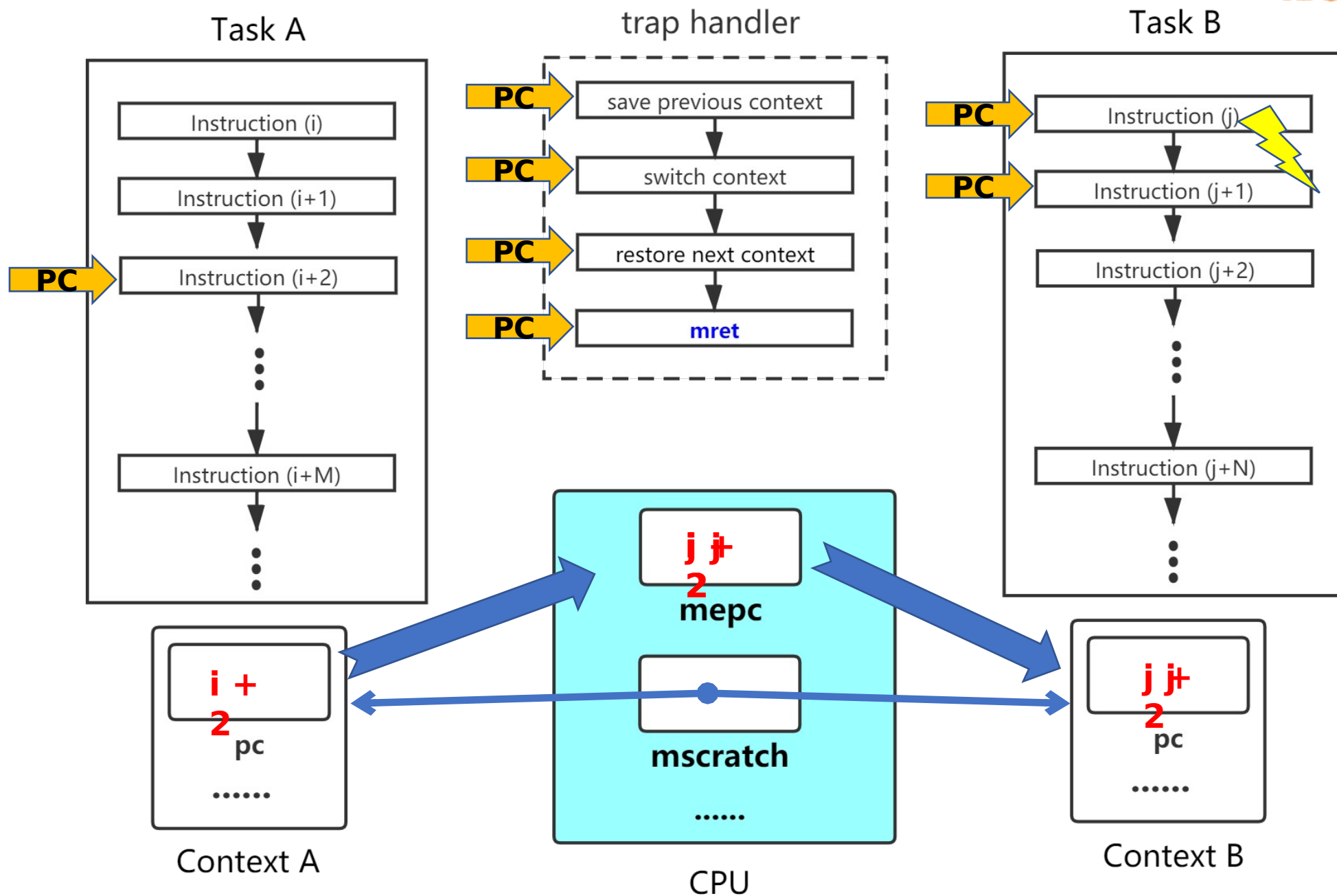
Task B



抢占式多任务的设计



抢占式多任务的设计



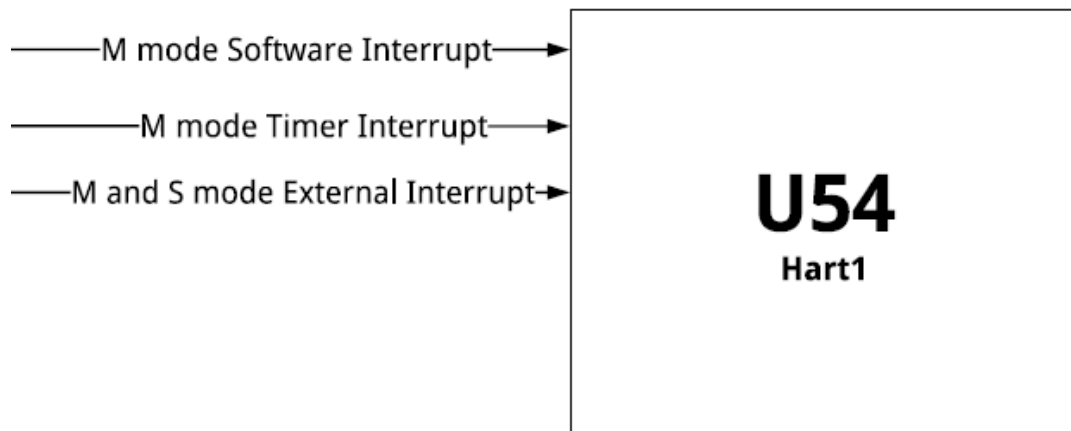
- 抢占式多任务
- 抢占式多任务的设计
- 兼容协作式多任务

兼容 task_yield()

```
/*  
 * DESCRIPTION  
 *      task_yield() causes the calling task to relinquish the CPU and a new  
 *      task gets to run.  
 */  
void task_yield()
```

RISC-V 中断（Interrupt）的分类

- 本地（Local）中断
 - software interrupt
 - timer interrupt
- 全局（Global）中断
 - external interrupt



Interrupt	Exception Code	Description
1	0	User software interrupt
1	1	Supervisor software interrupt
1	2	<i>Reserved for future standard use</i>
1	3	Machine software interrupt
1	4	User timer interrupt
1	5	Supervisor timer interrupt
1	6	<i>Reserved for future standard use</i>
1	7	Machine timer interrupt
1	8	User external interrupt
1	9	Supervisor external interrupt
1	10	<i>Reserved for future standard use</i>
1	11	Machine external interrupt
1	12–15	<i>Reserved for future standard use</i>
1	≥16	<i>Reserved for platform use</i>

【参考 2】Table 3.6: Machine cause register (mcause) values after trap.

【参考 3】 Figure 3: FU540-C000 Interrupt Architecture Block Diagram.

Core Local INTerruptor

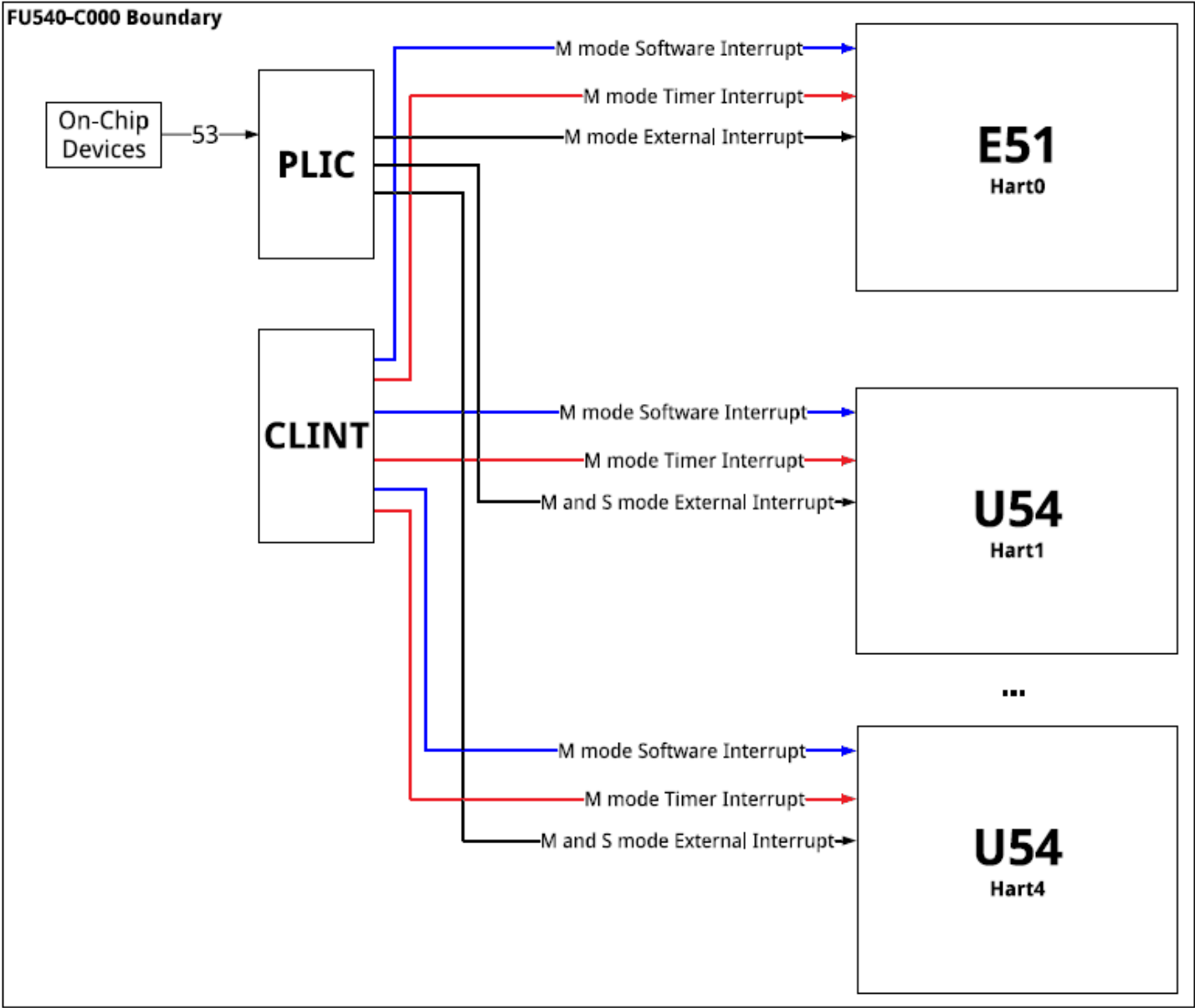


Figure 3: FU540-C000 Interrupt Architecture Block Diagram.

可编程寄存器	功能描述	内存映射地址
MSIP	32-bit 字长。高 31 位不可用，最低位和 CSR mip.MSIP 对应。写入 1 时触发 software interrupt，写入 0 表示对该中断进行应答。	BASE + 4 * (hart)

- RISC-V 规范规定，Machine 模式下的 mip.MSIP 对应到一个 memory-mapped 的控制寄存器。QEMU-virt 映射到 CLINT 的 MSIP。
- 具体寄存器编址采用 base + offset 的格式，且 base 由各个特定 platform 自己定义。针对 QEMU-virt，其 CLINT 的设计参考了 SFIVE，base 为 0x2000000。

谢谢

欢迎交流合作