



结合协程调度的中断控制器设计与应用

清华大学计算机系 赵方亮

2024年1月12日

指导教师：向勇

设计背景

- 构建低时延服务成为重要研究领域
 - 协程
 - 中断

研究现状

调度策略

- D-FCFS
- D-FCFS + Work stealing
- C-FCFS
- C-FCFS + PS

中断 + 协程

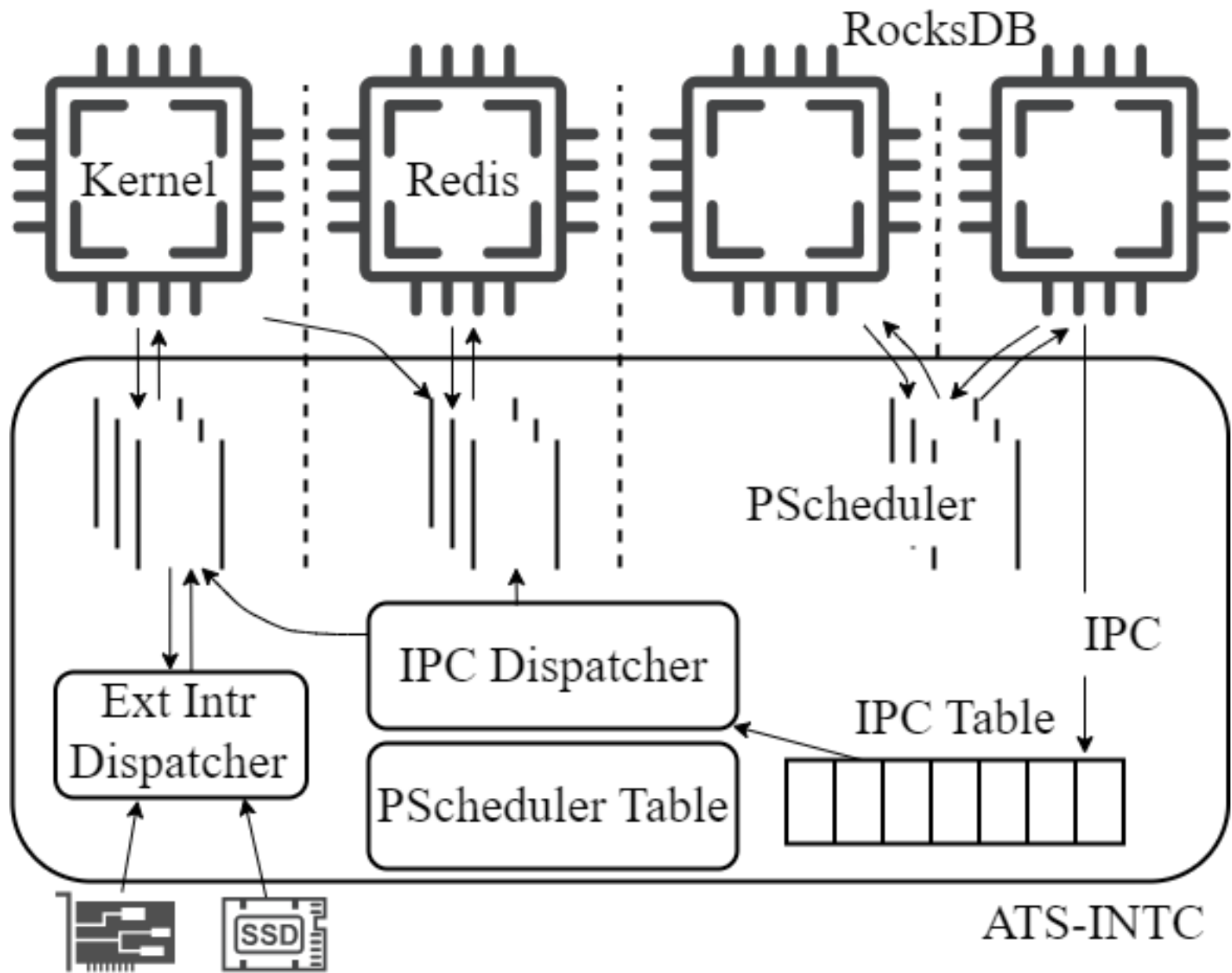
- 中断开销VS 协程调度开销

中断

- 用途
 - 资源控制
 - 任务抢占
 - 模拟中断抢占
- 中断处理
 - 软件处理中断
 - 硬件处理中断

研究内容

系统架构

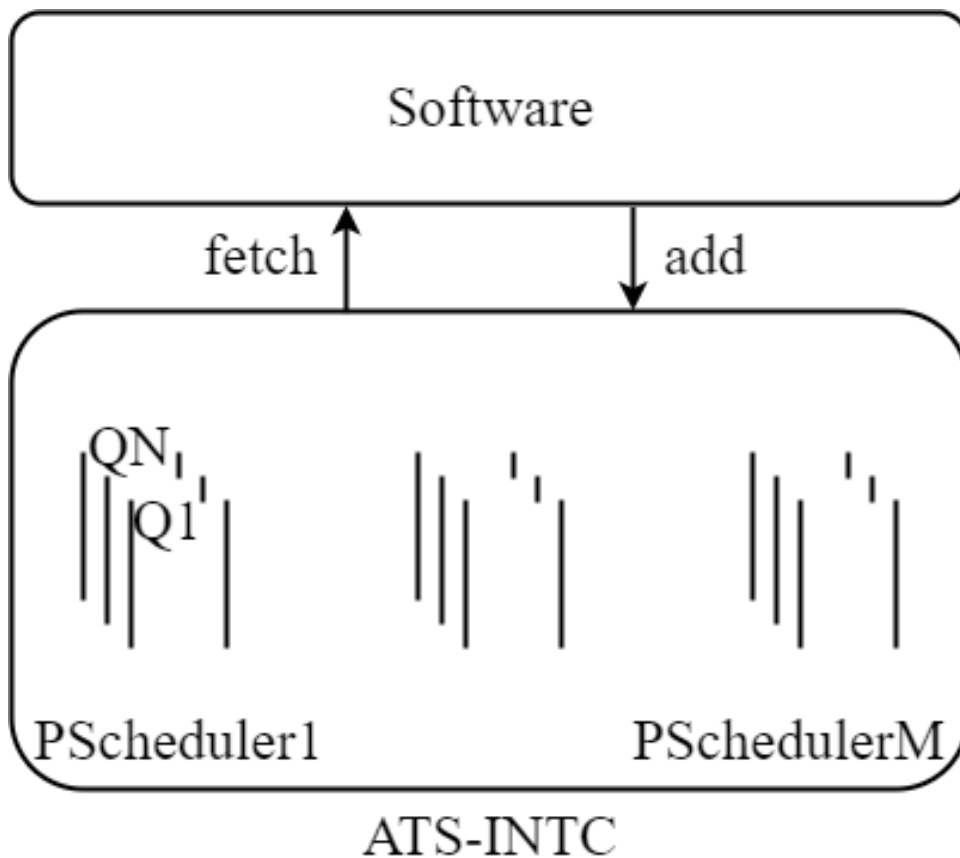


研究内容

```
#[repr(C)]
pub struct Task {
    pub cid: u32,
    pub priority: AtomicU32,
    pub ctype: TaskType,
    pub fut: AtomicCell<Pin<Box<dyn Future<Output=usize> + 'static + Send + Sync>>>,
}
```

基于优先级的协程调度机制

- 硬件维护就绪优先级队列
 - 缩小调度开销
- 协作式调度 + 优先级
 - 拟合抢占式调度
- 内核通过硬件接口感知用户态协程



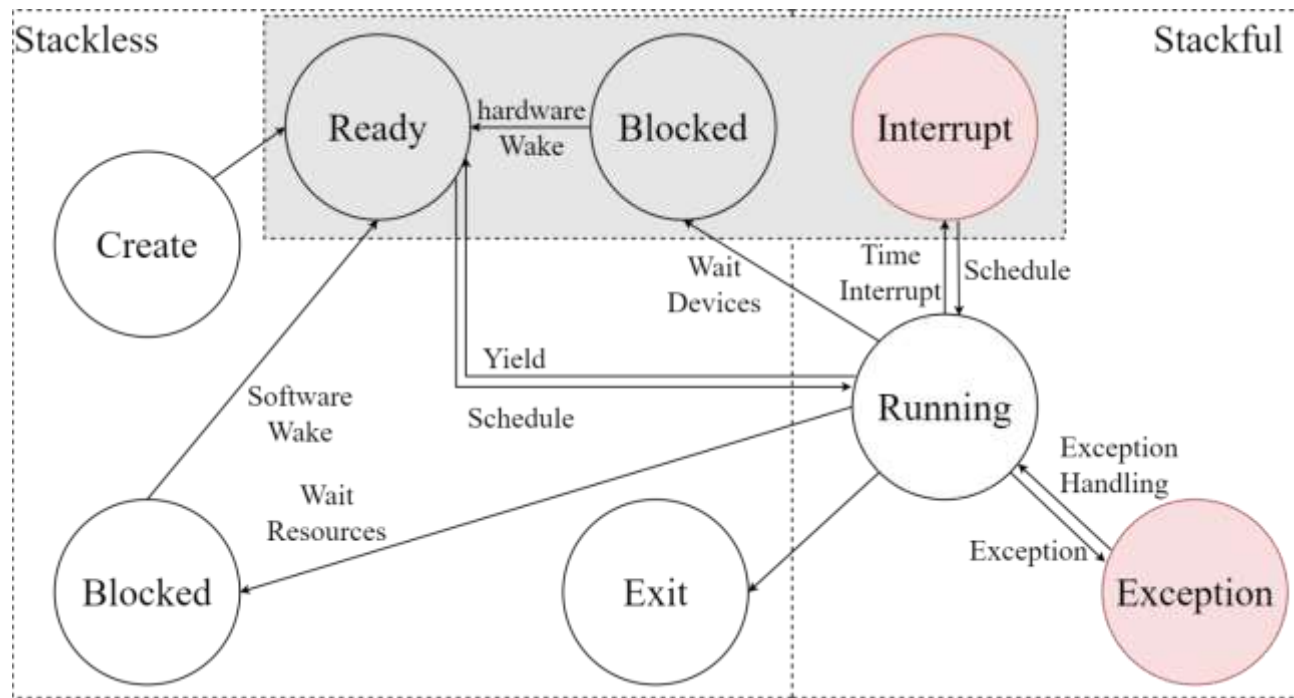
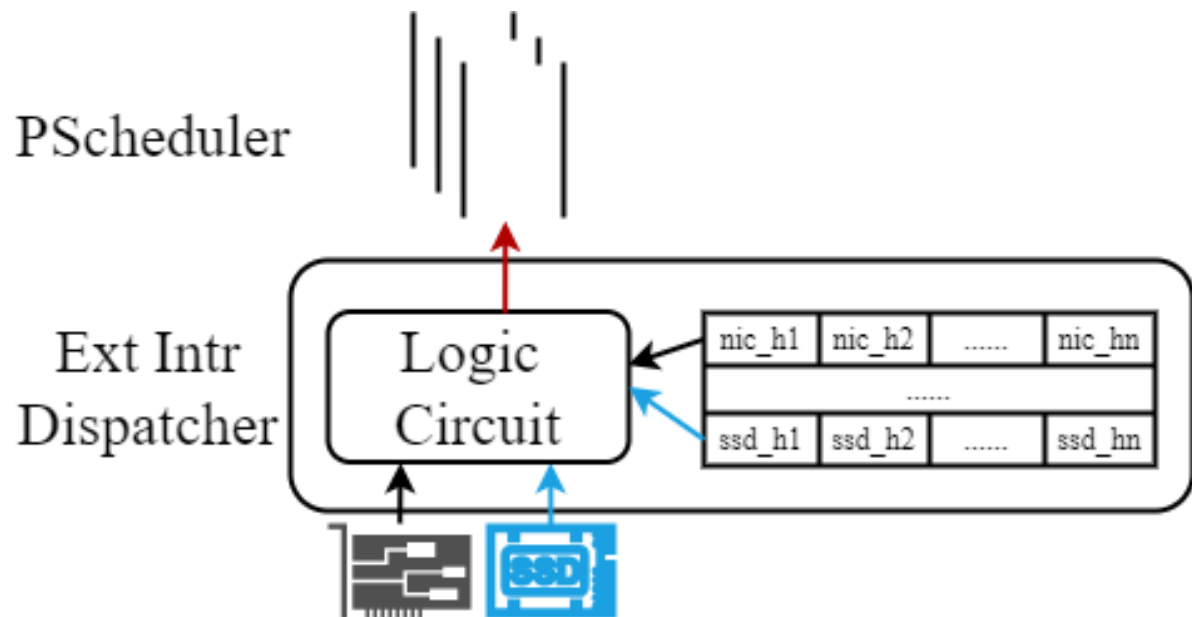
研究内容

结合中断处理的协程唤醒机制

- 硬件与 CPU 并行处理中断，唤醒阻塞协程

协程状态模型

- 就绪
- 阻塞：硬件、软件
- 运行
- 挂起：时钟中断、异常

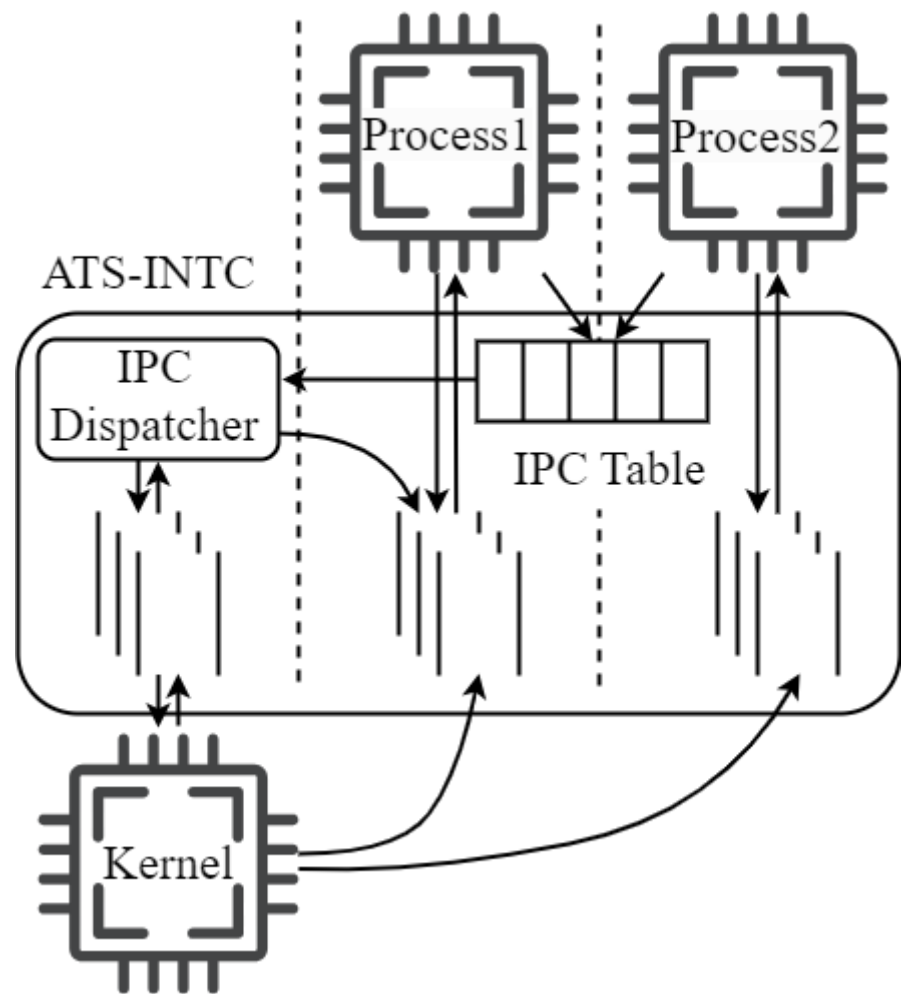


研究内容

基于协程调度与中断处理的 IPC 加速机制

- 系统调用加速
- 基于 IPI 实现的 IPC 加速

high					low
+-----+-----+-----+-----+-----+					
process info pointer	related user task_ref	ipc type	arguments	ipc res	
+-----+-----+-----+-----+-----+					



实施方案

QEMU 模拟环境

- 在 QEMU 模拟器中，构建任务调度器模块，搭建原型系统，进行 MicroBench，验证方案可行性

FPGA 实现

- 在 FPGA 中实现任务调度器模块，构建原型系统。修改上层应用，进行 MacroBench，构建出低时延、高吞吐量、隔离性强的服务

计划安排

1. 2023 年 11 月至 2024 年 1 月，查找相关资料，了解国内外的发展动态，细化设计方案，理清实施思路，完成开题工作。
2. 2024 年 1 月至 2024 年 2 月，在 QEMU 模拟器中实现基于优先级的协程调度器。
3. 2024 年 2 月至 2024 年 3 月，在 QEMU 模拟器中实现结合了协程调度的中断控制器（只考虑网卡中断处理）。
4. 2024 年 3 月至 2024 年 5 月，在 FPGA 中实现上一阶段的结合协程调度的中断控制器。在 Arceos unikernel 环境下，测试网卡的性能，并改造 redis 等上层应用，完成相关的性能对比测试，争取发表一篇论文。
5. 2024 年 5 月至 2024 年 6 月，在第 3 阶段基础上，将系统调用进行并行异步批处理优化。
6. 2024 年 6 月至 2024 年 8 月，将上一阶段成果移植至 FPGA 中，并基于 Arceos 宏内核 Starry，对上层应用进行改造，完成系统调用性能对比测试，争取发表一篇论文。
7. 2024 年 8 月至 2024 年 9 月，在第 5 阶段基础上，将基于 IPI 的 IPC 进行优化。
8. 2024 年 9 月至 2024 年 11 月，将上一阶段成果移植至 FPGA 中，并基于 Arceos 宏内核 Starry，对上层应用进行改造，完成 IPC 性能对比测试，争取发表一篇论文。
9. 2024 年 11 月至 2025 年 1 月，完成学位论文写作。
10. 2025 年 6 月，完成论文答辩。

参考文献

低时延服务

- [IX](#)
- [ZygOS](#)
- [Shenango](#)
- [Shinjuku](#)
- [Concord](#)
- [Demikernel](#)

系统调用优化

- [FlexSC](#)
- [Userspace Bypass](#)
- [Privbox](#)
- [Cassyopia](#)

IPC 优化

- [x86 User Interrupt](#)



谢谢！