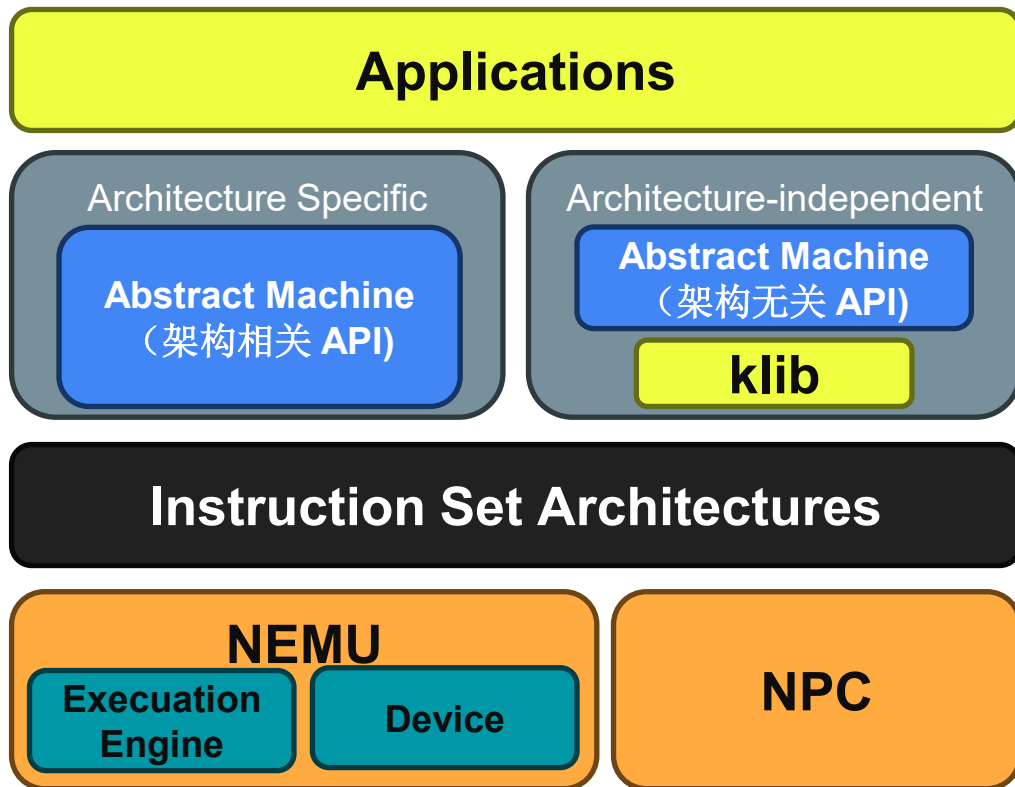


# **Golden Apple Corelet:** a compact, in-order RISC-V microarchitecture optimized for embedded systems.

Shuao Jia

Beijing University of Posts and Telecommunications

# Project Architecture Overview



# Functional Simulator (NEMU)

- **The "Differential Testing" Mechanism.**

- Instead of writing test cases, we compare processor's architectural state every cycle.
- **DUT (Design Under Test):** My RISC-V Core (Verilator simulation).
- **REF (Reference):** NEMU (C++ functional simulator).

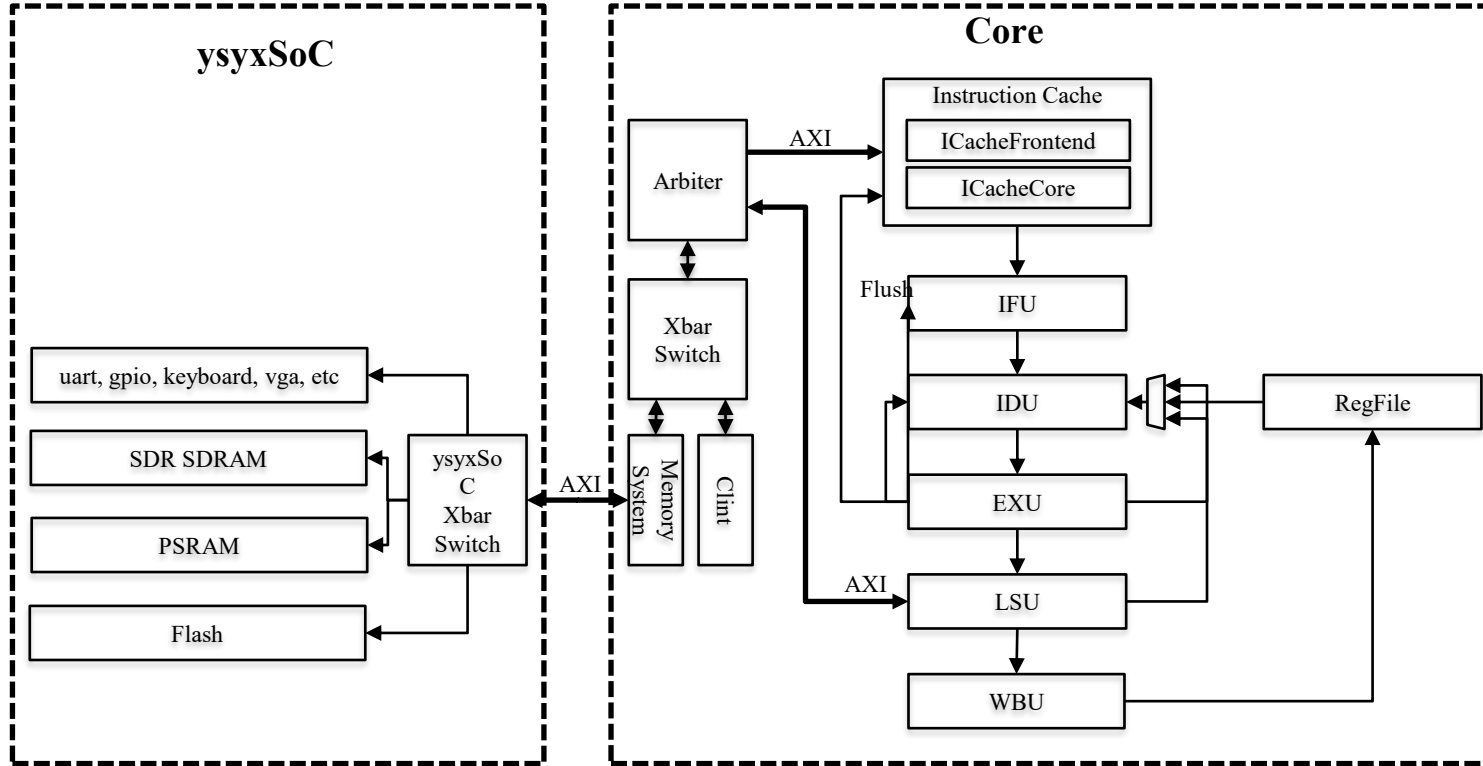
- **Workflow:**

1. Execute one instruction on both DUT and REF.
2. Check for consistency in GPRs and PC.
3. Abort immediately upon mismatch.

- **Result:**

- Successfully identified logic errors.

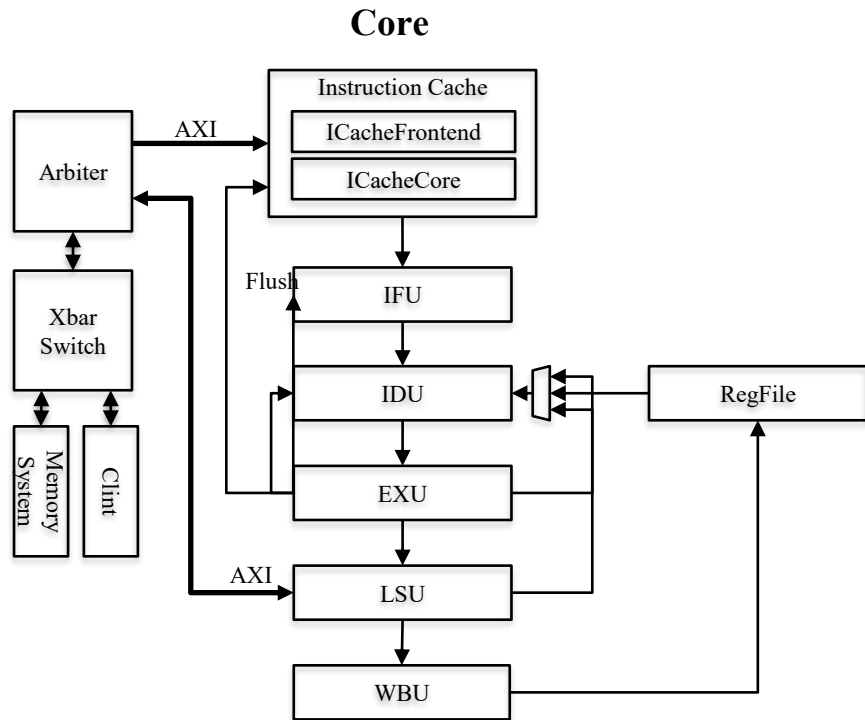
# Microarchitecture & System Integration



# Core Microarchitecture: RV32E

## 2+5 Stage Pipeline Design

- **Instruction Fetch (2 Stages):** Decoupled fetch unit with dual-stage ICache for high throughput.
- **Execution (5 Stages):** Classic RISC pipeline with full bypassing and hazard detection.
- **Prediction:** Static branch prediction to reduce control hazards.



# Agile Development



## Chisel3

Constructed hardware  
using Scala-based  
generator language for  
parameterized and  
reusable modules.



## Verilator

High-speed cycle-  
accurate simulation  
for rapid iteration and  
regression testing.



## Yosys + iSTA

Utilized open-source  
synthesis and timing  
analysis flows to  
ensure physical  
realizability.

# System Bring-up: RT-Thread

- **Booting a real-time operating system is the ultimate stress test for architectural correctness.**
  - Implemented CSR (Control and Status Registers) for exception handling.
  - Handled timer interrupts and context switching.
  - Successfully entered the interactive shell of RT-Thread.

```
wsl: 检测到 localhost 代理配置, 但未镜像到 WSL. NAT 模式下的 WSL 不支持 localhost 代理.
(base) [root@Cal ~] $ cd rt-thread-am/bsp/abstract-machine
(base) [root@Cal abstract-machine] $ make ARCH=riscv32e-nemu run
# Building rtthread-run [riscv32e-nemu]
# Building am-archive [riscv32e-nemu]
# Building klib-archive [riscv32e-nemu]
+ OBJCOPY -> build/rtthread-riscv32e-nemu.bin
mainargs=
make -C /root/ysyx-workbench/nemu ISA=riscv32e run ARGS="-l /root/rt-thread-am/bsp/abstract-machine/build/rtthread-riscv32e-nemu.bin"
make[1]: Entering directory '/root/ysyx-workbench/nemu'
```

Boot

```
msh />date
[W/time] Cannot find a RTC device!
local time: Thu Jan 1 08:00:00 1970
timestamps: 0
timezone: UTC+0
msh />version

 \ | /
- RT -   Thread Operating System
 / | \   5.0.1 build May 12 2025 17:34:33
2006 - 2022 Copyright by RT-Thread team
msh />free
total      : 124936088
used       : 33583256
maximum    : 33583256
available  : 91352832
msh />ps
thread
-----
pri  status      sp      stack size max used left tick error
-----
tshell      20  running 0x00000060 0x00001000 15% 0x0000000a OK
sys workq   23  ready  0x00000060 0x00002000 01% 0x0000000a OK
tidle0      31  ready  0x00000060 0x00004000 00% 0x00000020 OK
timer       4   suspend 0x0000008c 0x00004000 01% 0x0000000a OK
main        10  close  0x00000080 0x00000800 55% 0x00000014 OK
msh />pwd
/
```

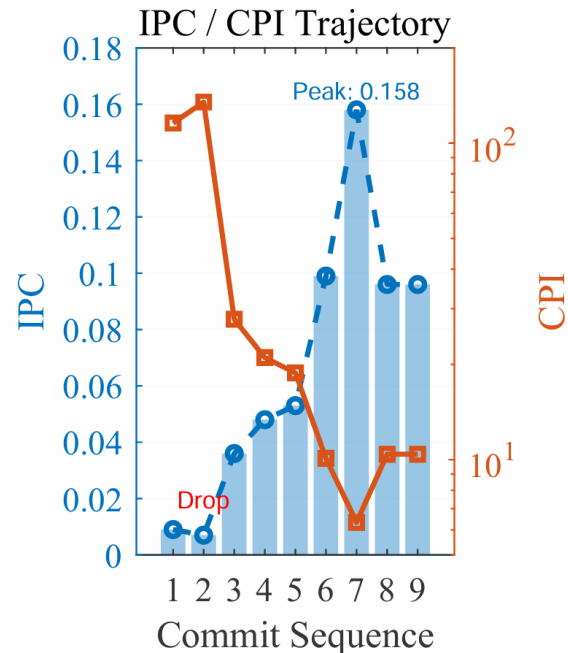
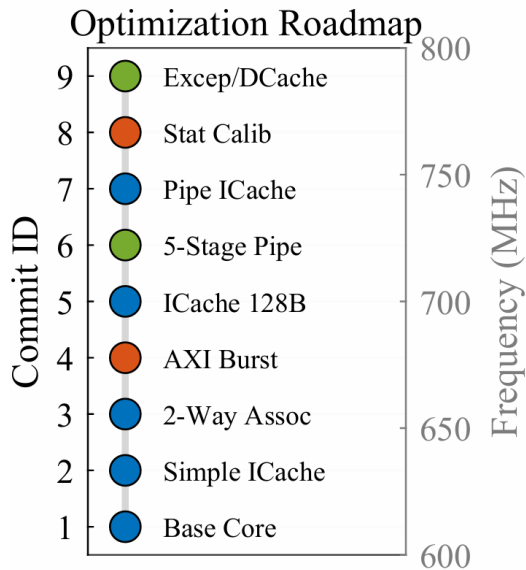
# Performance Evaluation: Microarchitectural Evolution

- **IPC Trajectory:**

- Significant improvement from Base Core (0.02) to Peak (0.158).

- **Bottleneck Shift:**

- *Phase 1:* Memory Bound (84.6%)
- *Phase 2:* Fetch Bound (64.8%)
- *Final:* IDU Stall Bound





# Performance Evaluation: Microarchitectural Evolution

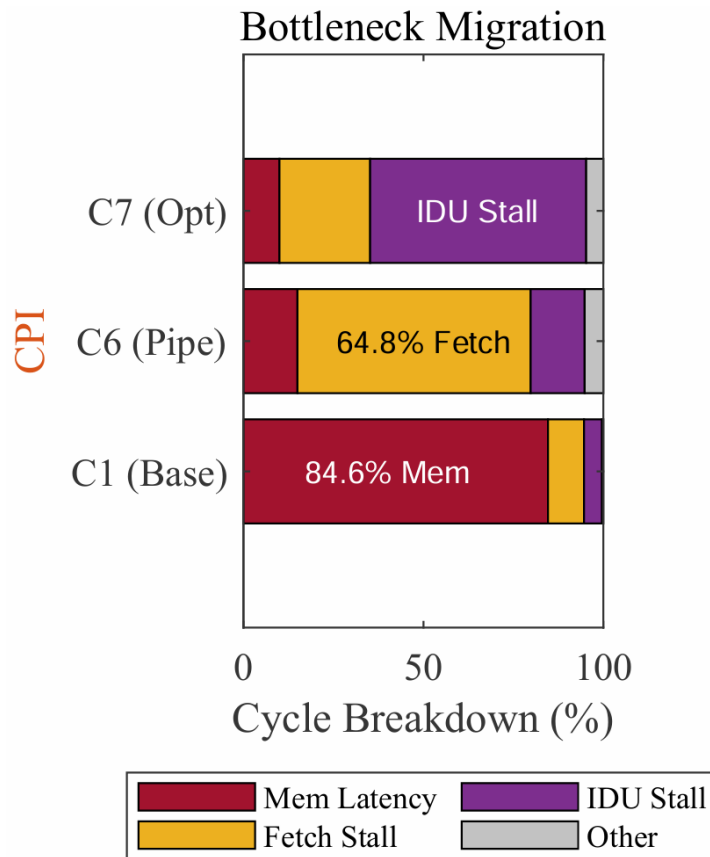
Iterative optimization process tracking IPC growth and bottleneck migration.

- **IPC Trajectory:**

- Significant improvement from Base Core (0.02) to Peak (0.158).

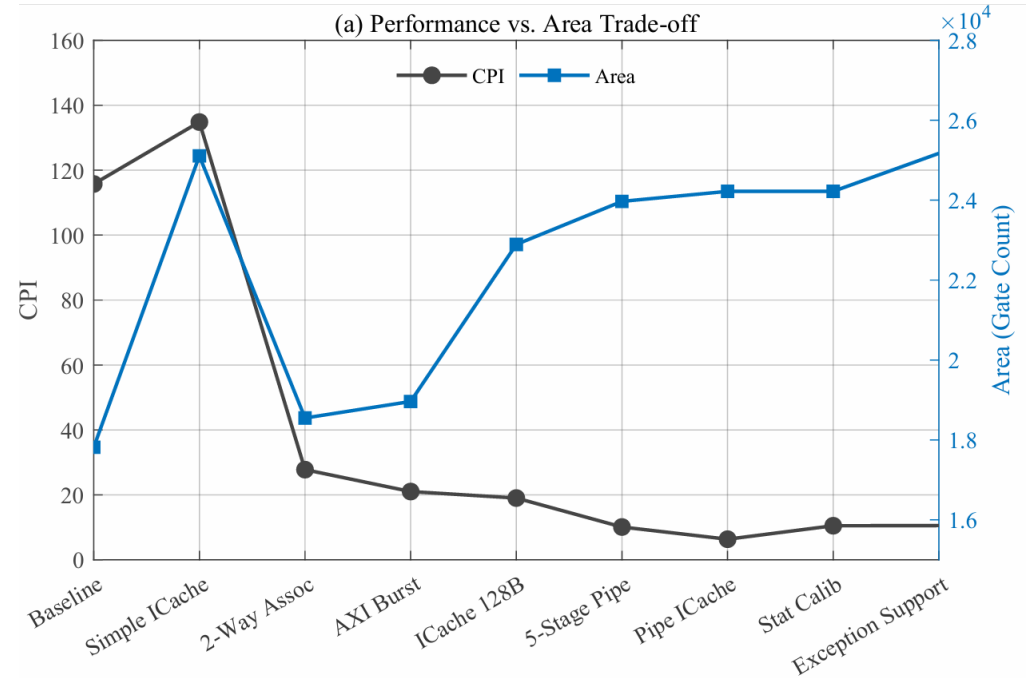
- **Bottleneck Shift:**

- *Phase 1:* Memory Bound (84.6%)
- *Phase 2:* Fetch Bound (64.8%)
- *Final:* IDU Stall Bound



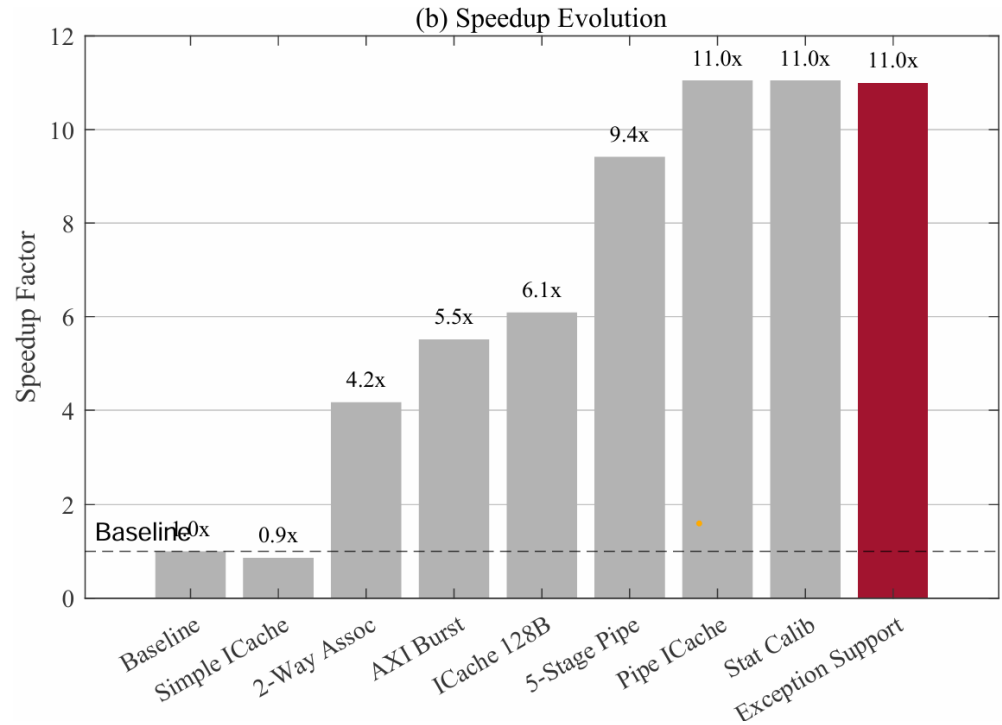
# Performance Evaluation: Optimization Results

- **Performance vs. Area:** Gate count increased linearly while CPI dropped exponentially, validating the efficiency of complex structures.
- **Final Speedup:** Achieved an 11.0x total speedup over the baseline implementation.



# Performance Evaluation: Optimization Results

- **Performance vs. Area:** Gate count increased linearly while CPI dropped exponentially, validating the efficiency of complex structures.
- **Final Speedup:** Achieved an 11.0x total speedup over the baseline implementation.



# Summary & Future Work

## Achievements

- ✓ Designed 2+5-stage RISC-V Core
- ✓ Integrated SoC & Booted OS
- ✓ Passed Stage C Review

## Tape-out Target

Advance from Stage C to Stage B to complete full chip tape-out verification and silicon validation.

## Advanced Architecture

Participate in the NSCSCC (全国计算机系统能力大赛) to explore Out-of-Order (OoO) execution and dynamic scheduling.