Maximizing Performance at Low Area Cost in RISC-V Processors Leveraging Fine-Grained Multithreading

Arbi Gunbardhi[1], Riccardo Tedeschi[1], Filippo Grillotti[3], Fabio De Ambroggi[3], Elio Guidetti[3], Luca Benini[1,2] , Davide Rossi[1]

[1] ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA
[2] ETH zürich
[3] STMicroelectronics

Motivation

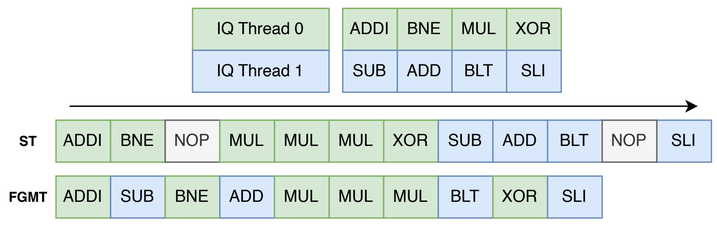
Designing processors requires balancing high execution throughput with strict silicon area constraints. However, traditional compact, in-order pipelines suffer from **performance degradation due to basic data dependencies and control hazards**. To overcome these bottlenecks **without relying on complex, area-heavy branch prediction**, in this work we propose:

➢ A 32-bit RISC-V core that implements the RV32ECM ISA supporting **Fine-Grained Multi-Threading (FGMT).**

➢ The **Thread Forwarding (TFW)** mechanism that dynamically allows instructions from different threads to coexist in the same pipeline stage, maximizing the utilization of available functional units.

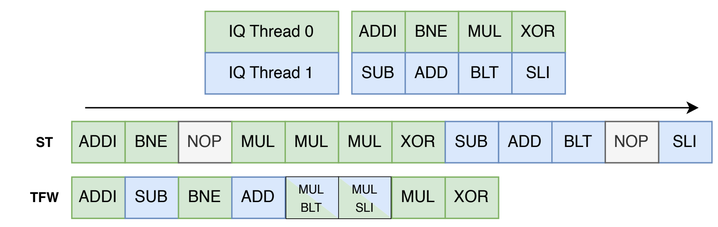
Fine Grained Multi-Threading (FGMT)

Fine-Grained Multi-Threading (**FGMT**) is a hardware multithreading paradigm that natively supports **multiple execution contexts within a single core**. The **interleaving** creates a latency between instruction of the same thread, which allows **mitigation of data and control hazards**.

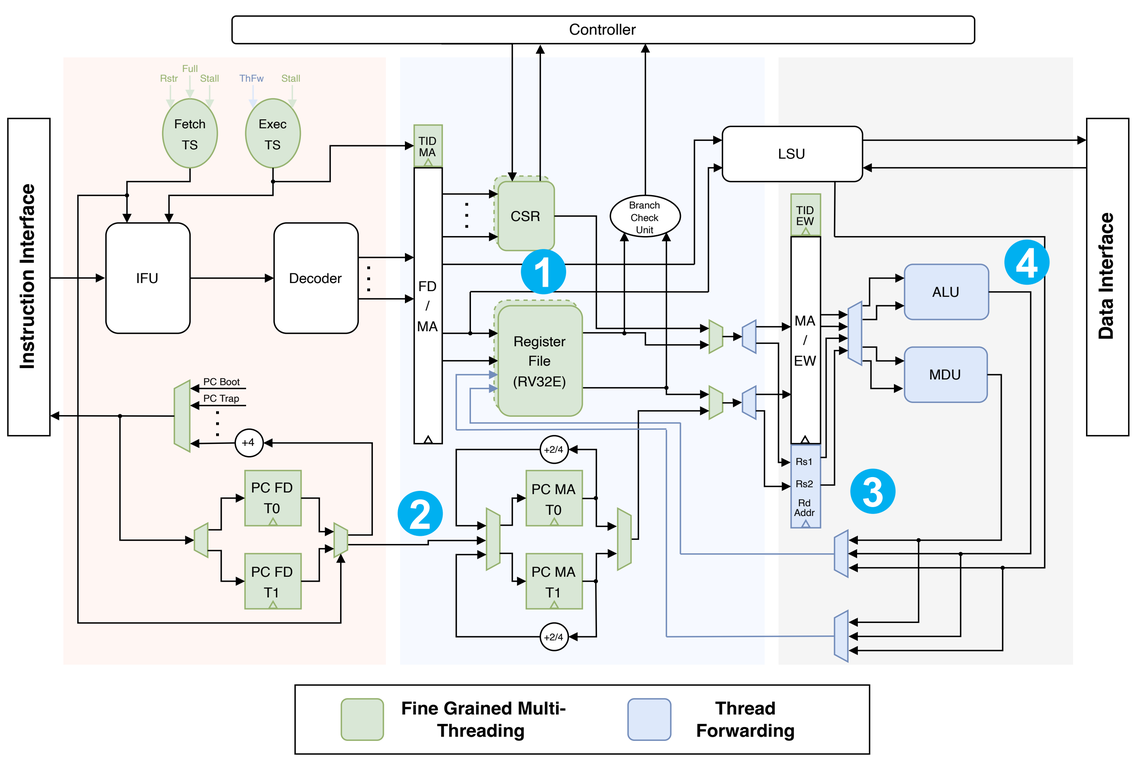


Thread Forwarding (TFW)

Thread Forwarding (**TFW**) is an optimized architectural enhancement that allows instructions from **different threads** to **coexist** within the **same pipeline stage**. The dynamic forwarding utilizes idle functional units during multi-cycle operations, which allows the **mitigation of structural hazards** and throughput degradation.



Microarchitecture



1 *Context Duplication*

The **register file** is doubled alongside the **CSRs**, to maintain independent **execution states**.

2 *PC Duplication*

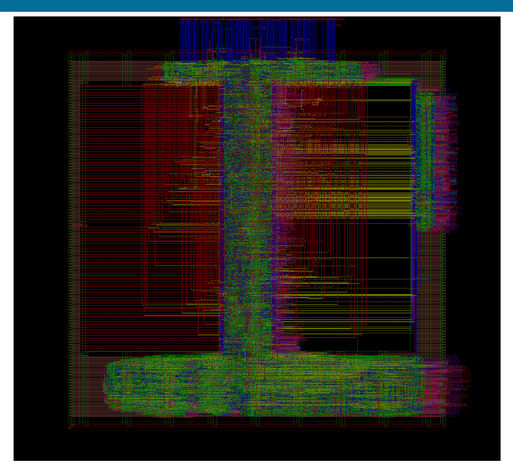
3 *New pipeline registers*

Additional **registers** in the **execute/writeback stage** preserve the operands and destination addresses of forwarded instructions, enabling **concurrent execution**.

4 *Steering Logic*

A dual-scheduler system and dynamic multiplexer network, driven by the **Thread ID**, **regulate access to shared pipeline resources** and redirect control signals.

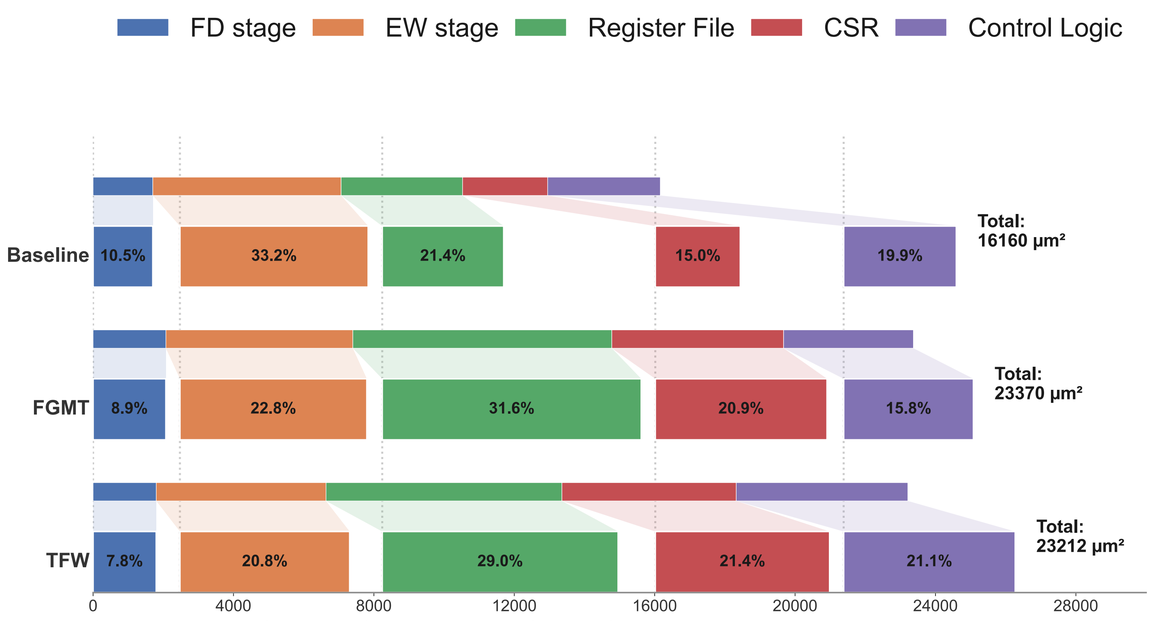
Physical Implementation



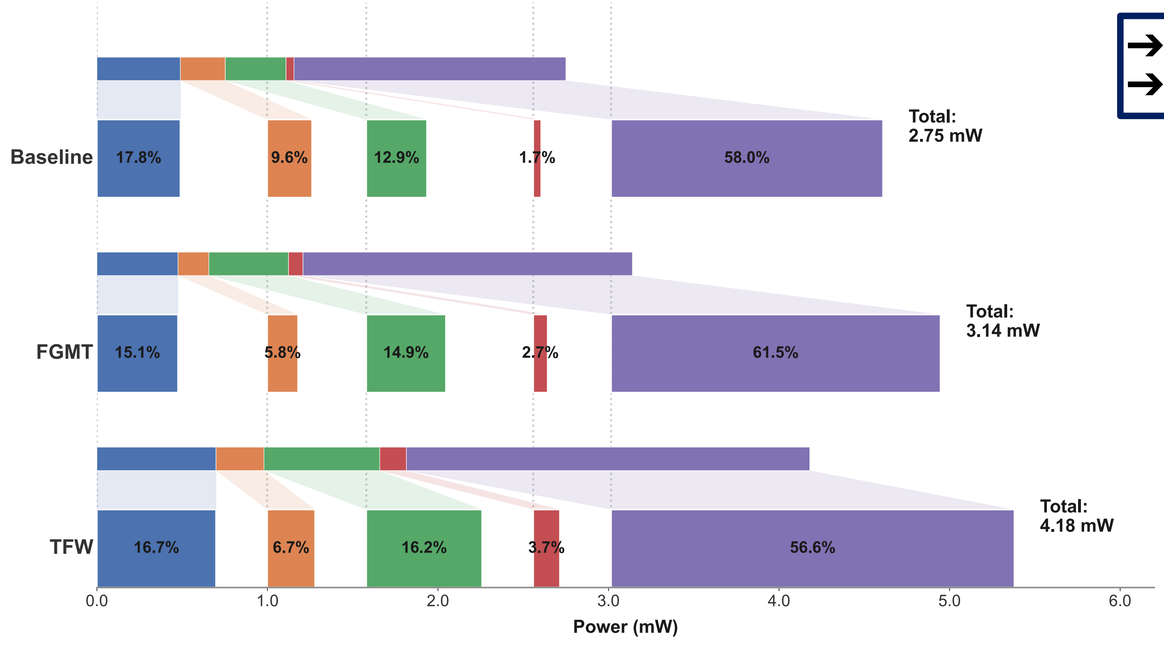
The design is physically implemented in a **40nm** (C40) technology node, targeting a maximum operating frequency of **300 MHz**. The core's place-and-route was conducted using Cadence Innovus, with post-layout **area** evaluations extracted under the **worst-case corner** (1.05V, SS, -40°C). **Power consumption** and switching activity were analyzed using Synopsys PrimeTime under the **typical corner** (1.10V, TT, 25°C, RC typical).

PPA Analysis

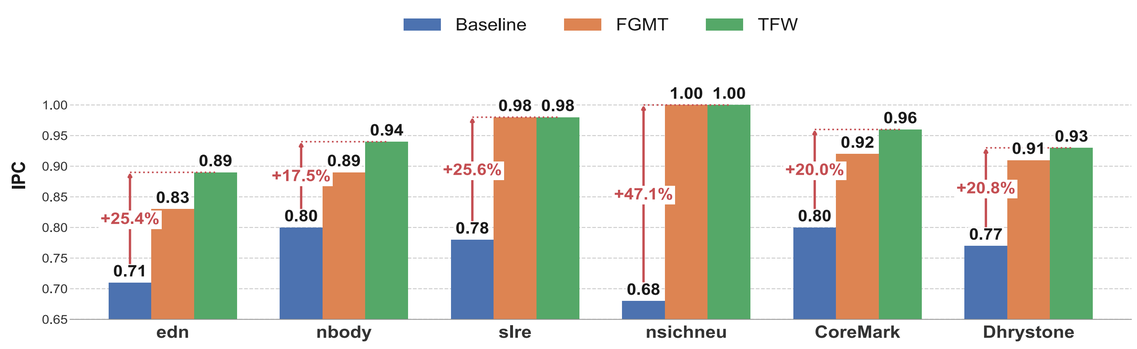
Area



Power



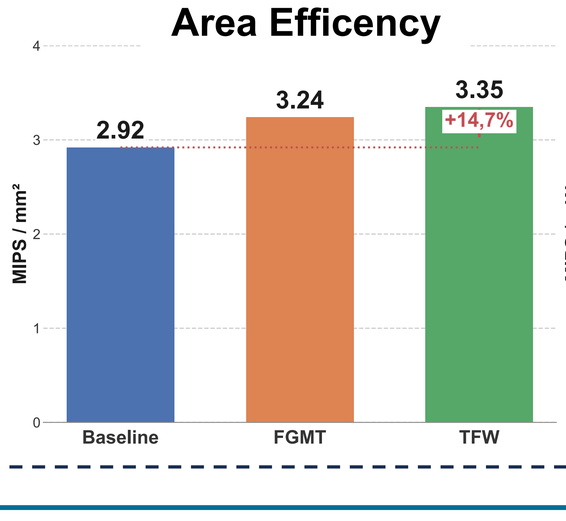
Performance



➔Most improvement in **control heavy** workloads
➔Robust improvement in more **comprehensive workloads**

➔**Most of the overhead** in power and area due to **CSR's and Register File duplication**
➔**FGMT and TFW** configuration are really **close in terms of** power and area **overhead**

Area Efficency



Energy Efficency