

TAIGA: A CONFIGURABLE RISC-V SOFT-PROCESSOR FRAMEWORK FOR HETEROGENEOUS COMPUTING SYSTEMS RESEARCH

Eric Matthews and Lesley Shannon

Simon Fraser University, Burnaby, BC, Canada {ematthew, lshannon}@sfu.ca

MOTIVATION

- Increasing interest in the integration of hardware accelerators and combining reconfigurable fabric with processors
- Wide range of accelerator types from tightly-integrated to loosely-integrated
- Many additional complexities arise as additional accelerators are integrated within a system
- Why RISC V:
 - People like to be able to repeat your experiments
 - You don't have to build everything else

MOTIVATION

- Increasing interest in the integration of hardware accelerators and combining reconfigurable fabric with processors
- Wide range of accelerator types from tightly-integrated to loosely-integrated
- Many additional complexities arise as additional accelerators are integrated within a system
- Why RISC V:
 - People like to be able to repeat your experiments
 - You don't have to build **everything** else

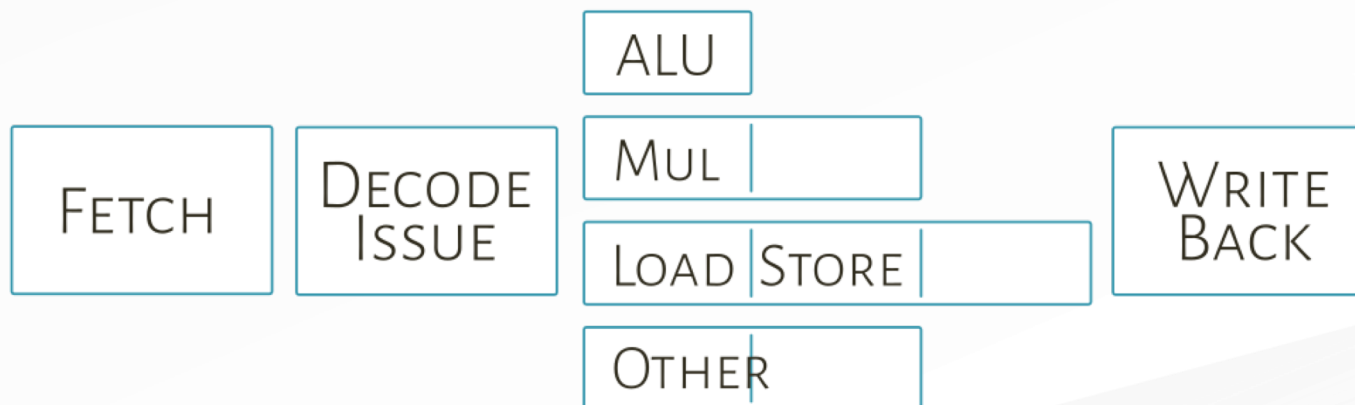
MOTIVATION

Today's soft-processors feature fixed-pipeline designs limiting integration options and performance gains

FIXED PIPELINE

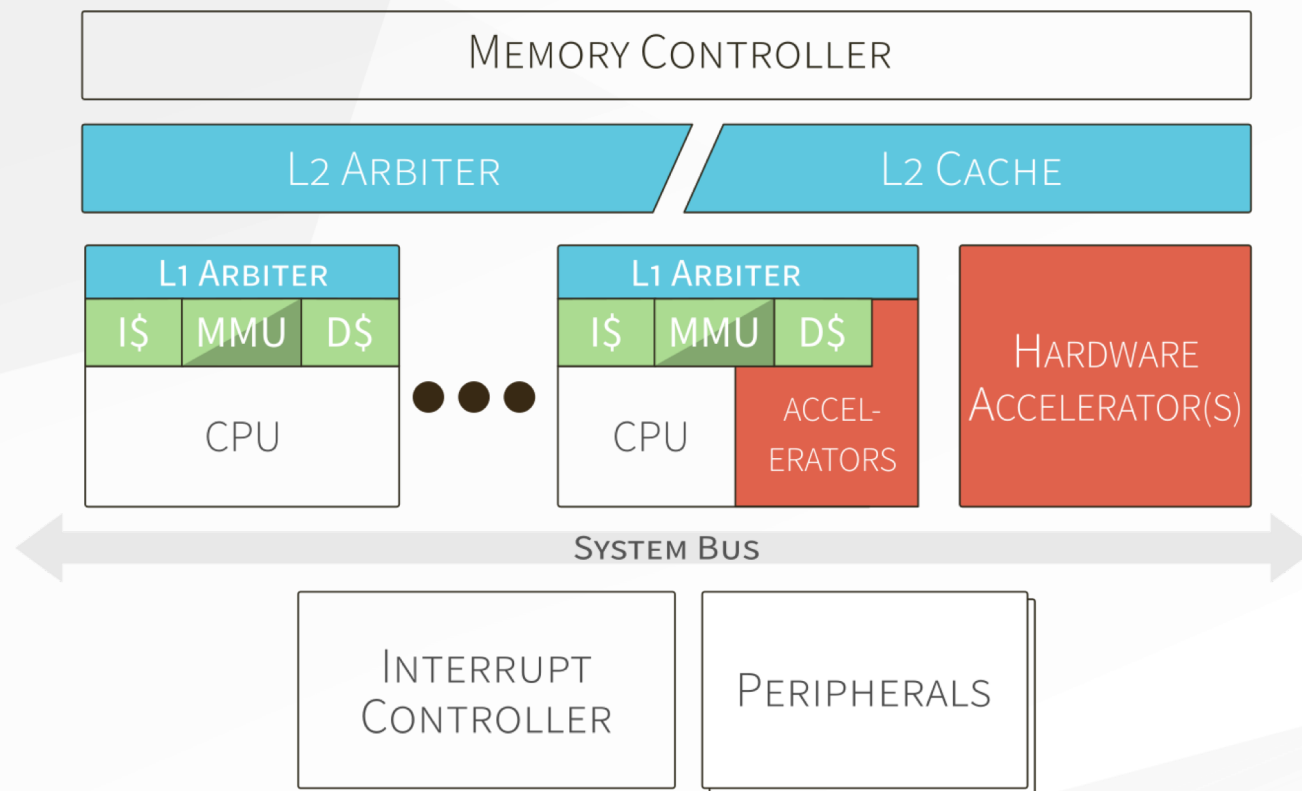


PARALLEL EXECUTION UNIT PIPELINE



OBJECTIVE

To create a framework for enabling research into heterogeneous computer systems research and reconfigurable computing systems.



TAIGA OVERVIEW

- Open-source processor
- Optimized for FPGAs
- RV32IMA
- Written in SystemVerilog
- Xilinx and Intel support



AVAILABLE AT:

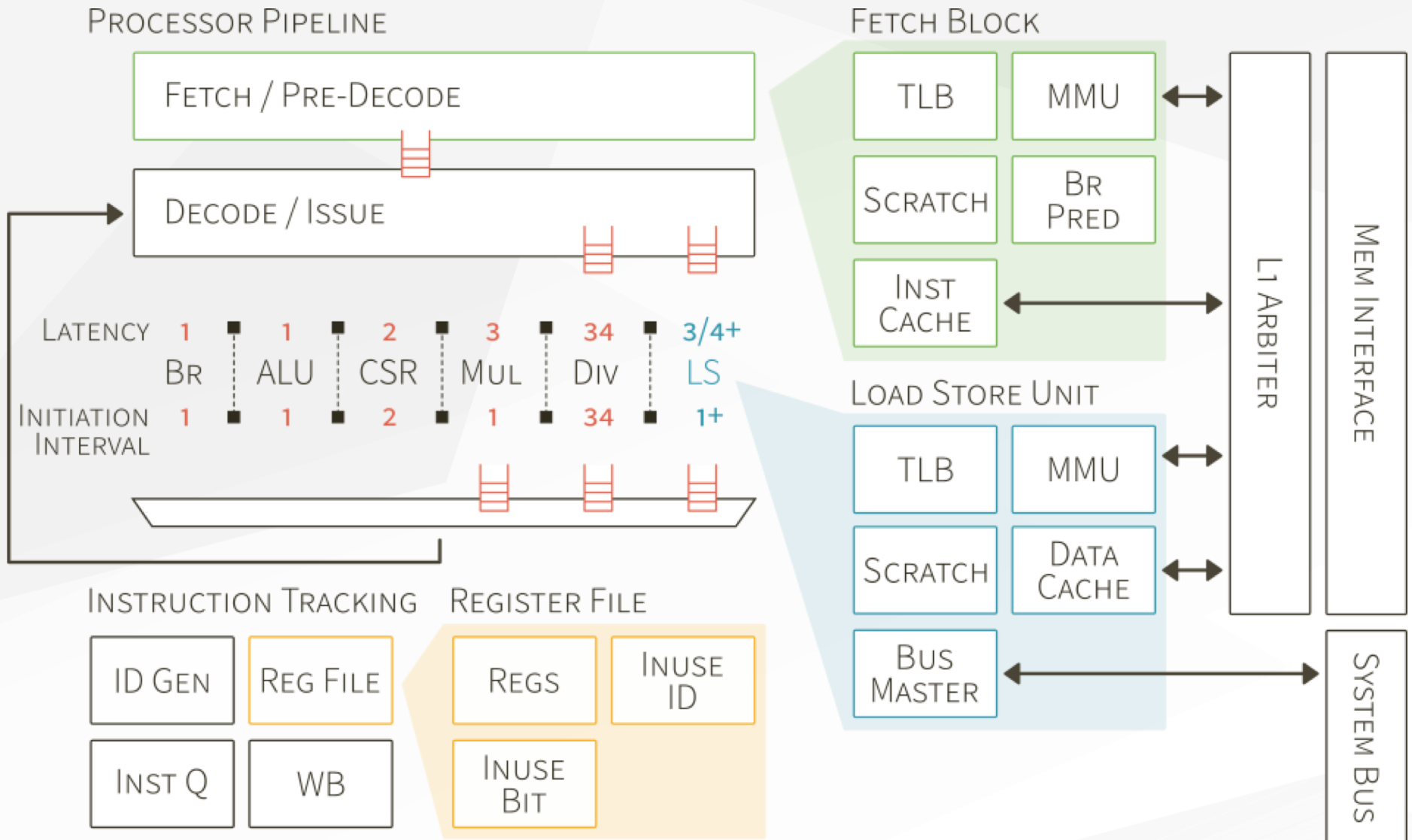
<https://gitlab.com/sfu-rcl/Taiga>

TAIGA OVERVIEW (2)

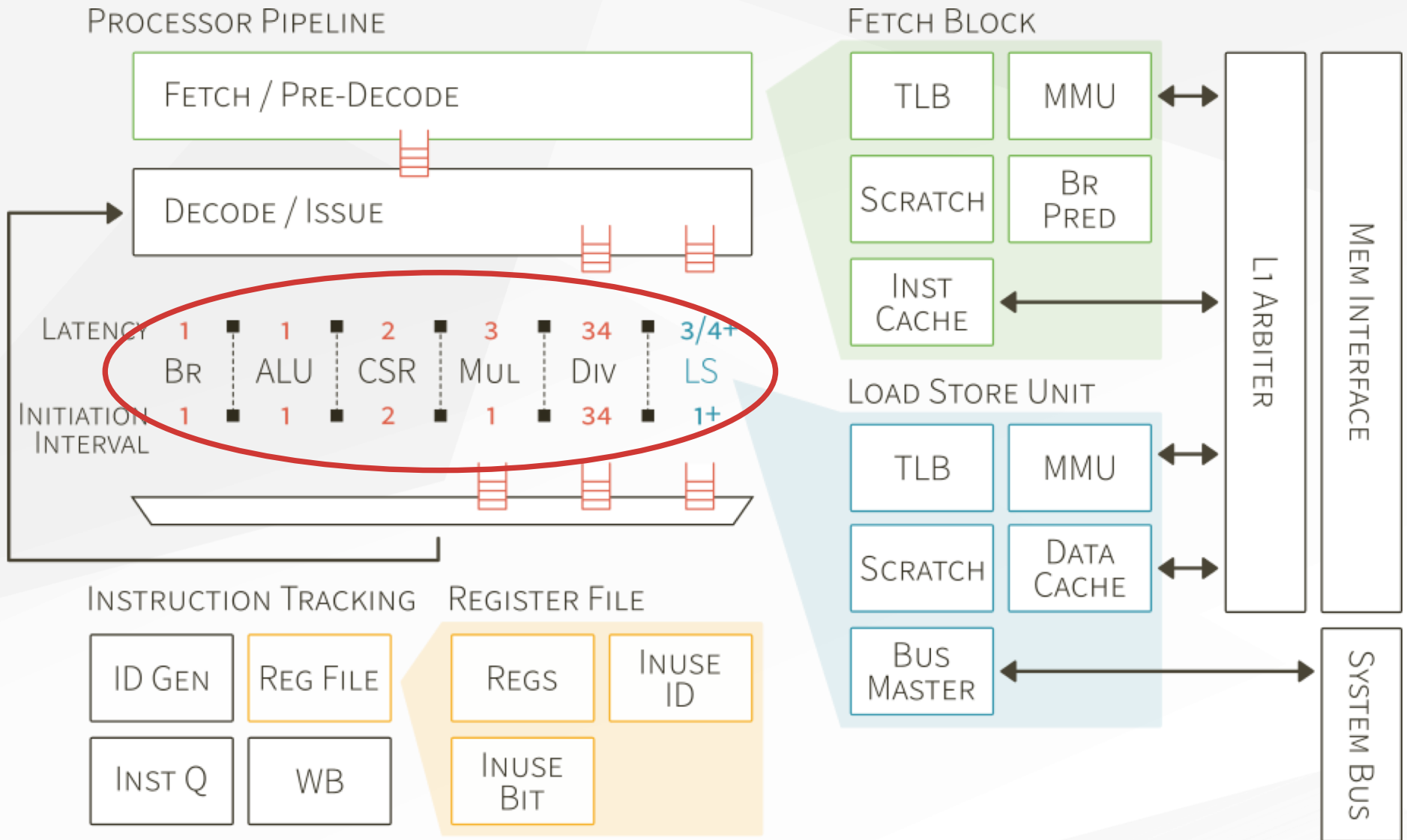
- Single core, single-issue, in-order-issue processor
- Highly decoupled and flexible design
- Supports independent variable-latency execution units
- Supports large range of configuration options



TAIGA OVERVIEW (3)

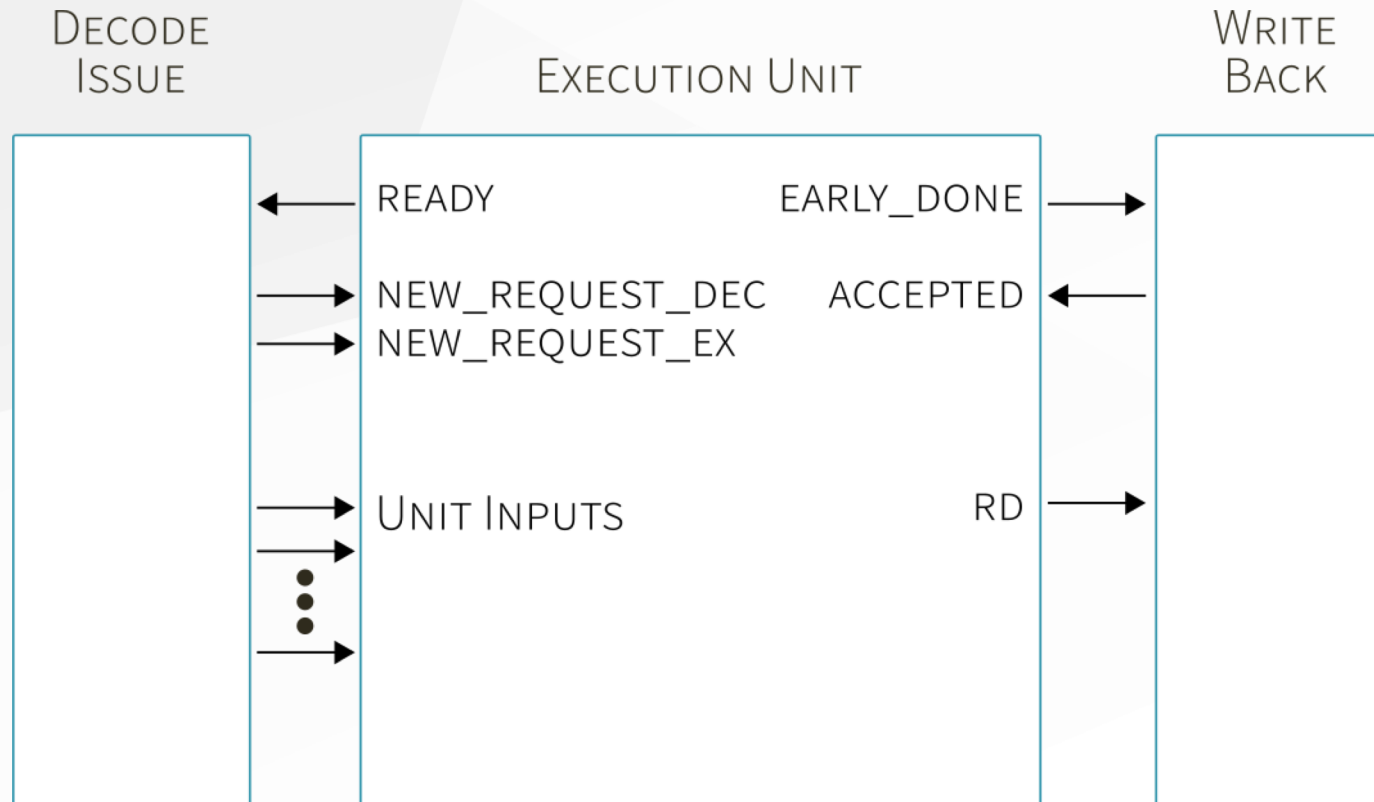


TAIGA OVERVIEW (3)

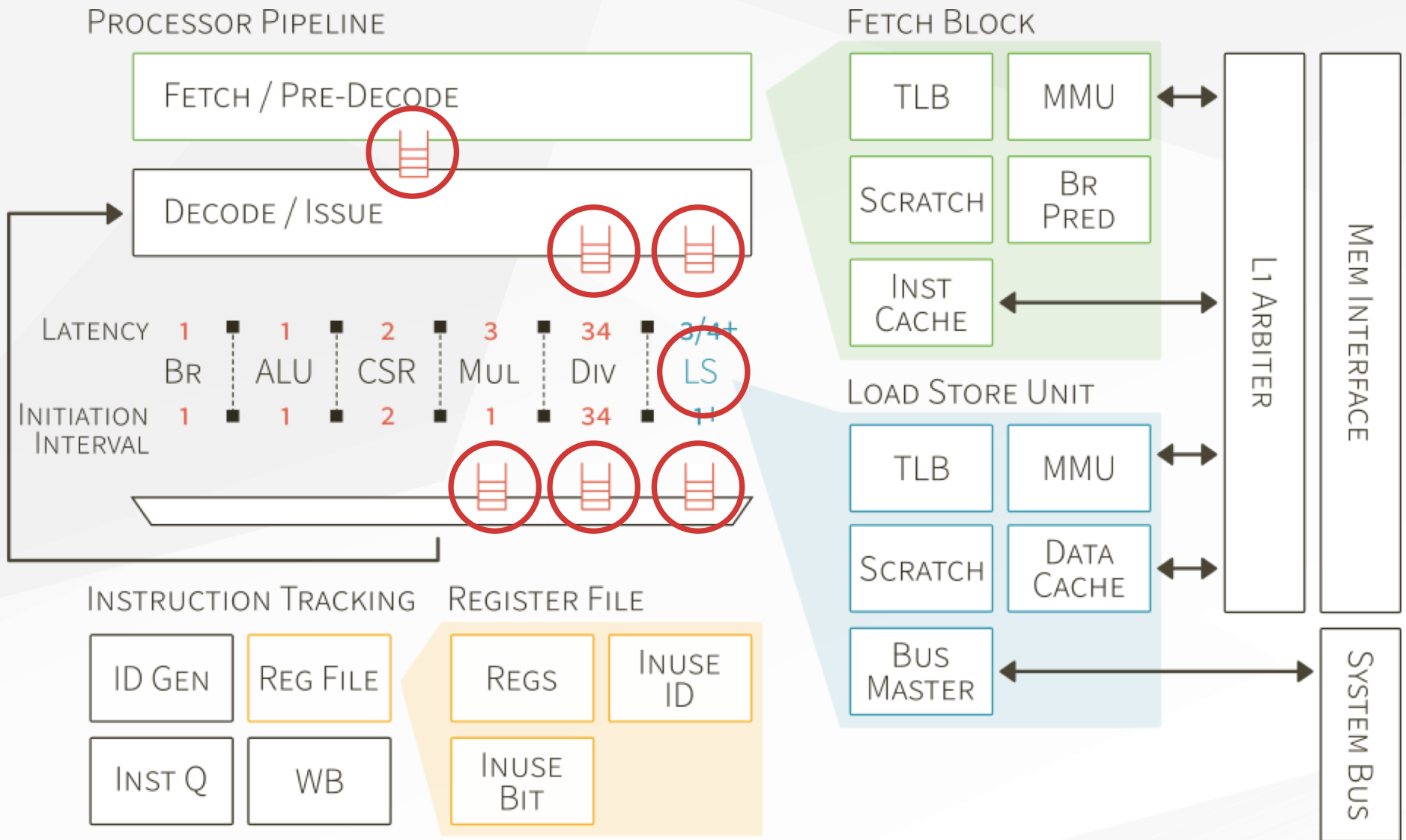


EXECUTION UNITS

- Standardized interface for control logic
- Interfaces can be decoupled with FIFOs
- Supports variable latency operation



TAIGA FIFOs (1)



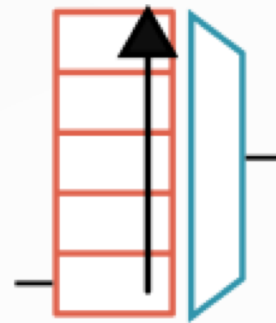
TAIGA FIFOs (2)

- Cross-vendor exploration of low-depth (2-4 entry) FIFOs
- For Intel Type A always smallest, Xilinx Type A smallest for >2 entries, Type B, register-based for 2 entries.



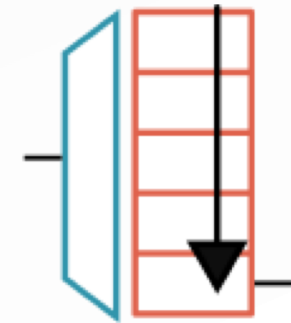
(A)

VARIABLE INPUT LOCATION
VARIABLE OUTPUT LOCATION
(LUT-RAM BASED)



(B)

FIXED INPUT LOCATION
VARIABLE OUTPUT LOCATION
(XILINX: SRL / INTEL: REGISTERS)



(C)

VARIABLE INPUT LOCATION
FIXED OUTPUT LOCATION
(REGISTERS)

RESOURCE USAGE COMPARISON

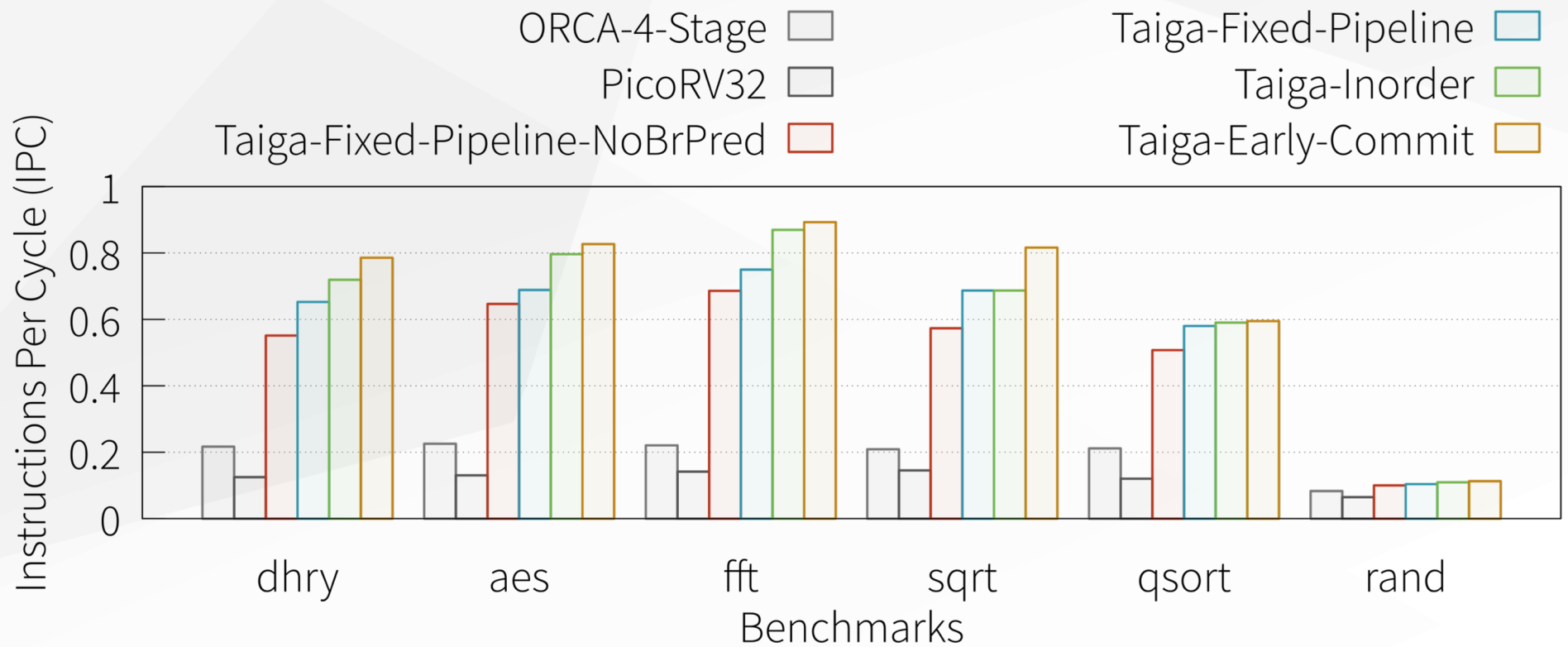
Resource usage and operating frequency on Zynq X7CZ020 (Zedboard)

	LUTs	FFs	Slices	BRAMs	DSPs	Freq (MHz)
Taiga-Fixed-Pipe-noBrPred	1434	948	495	0	4	120
ORCA 4-stage	1512 (+5%)	800 (-16%)	505 (+2%)	1 ¹	4	73 (-39%)
ORCA 5-stage	1625 (+13%)	934 (-1%)	527 (+6%)	1 ¹	4	75 (-38%)
PicoRV32	1545 (+8%)	830 (-12%)	481 (-3%)	0	4	172 (+43%)
Taiga-Fixed-Pipe	1496 (+4%)	1018 (+7%)	523 (+6%)	1 ²	4	116 (-3%)
Taiga-Early-Commit	1553 (+8%)	1038 (+9%)	536 (+8%)	1 ²	4	115 (-4%)

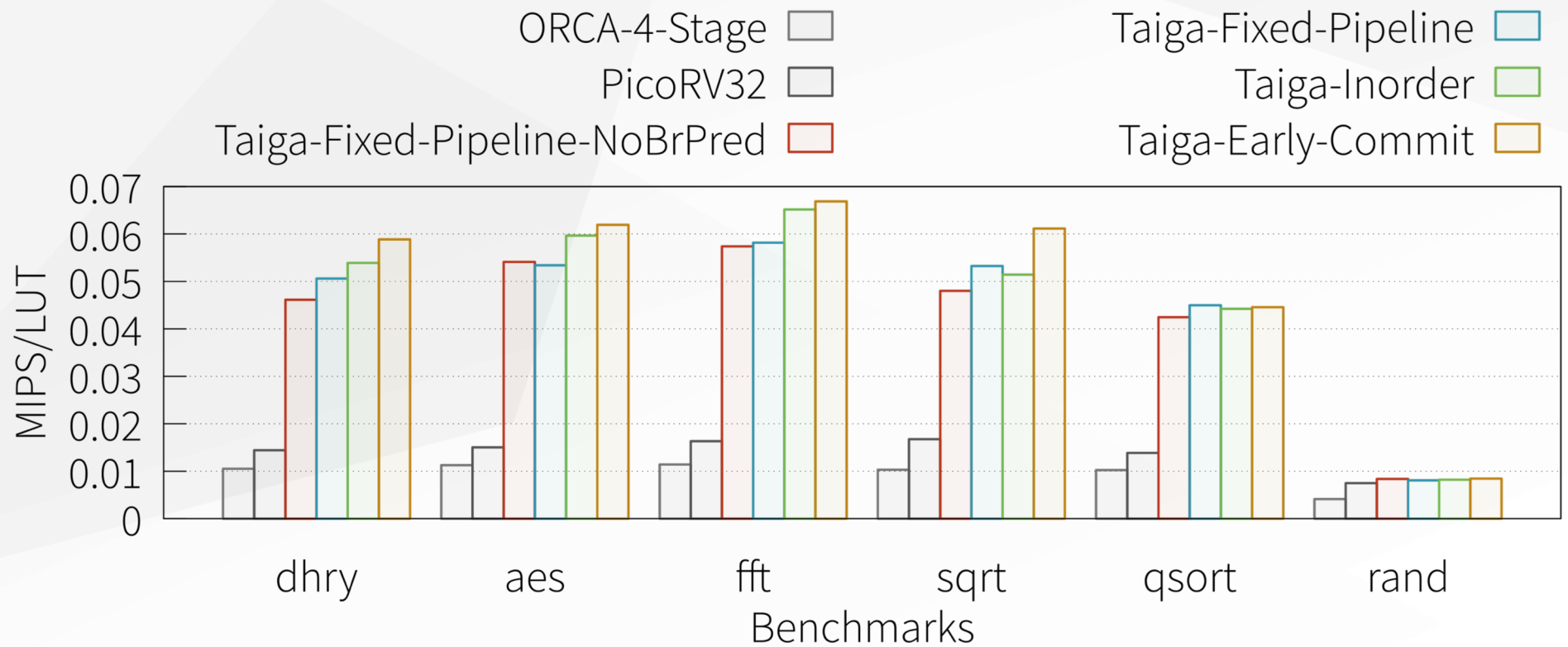
¹ BRAM is used for register file

² BRAM is used for dynamic branch predictor

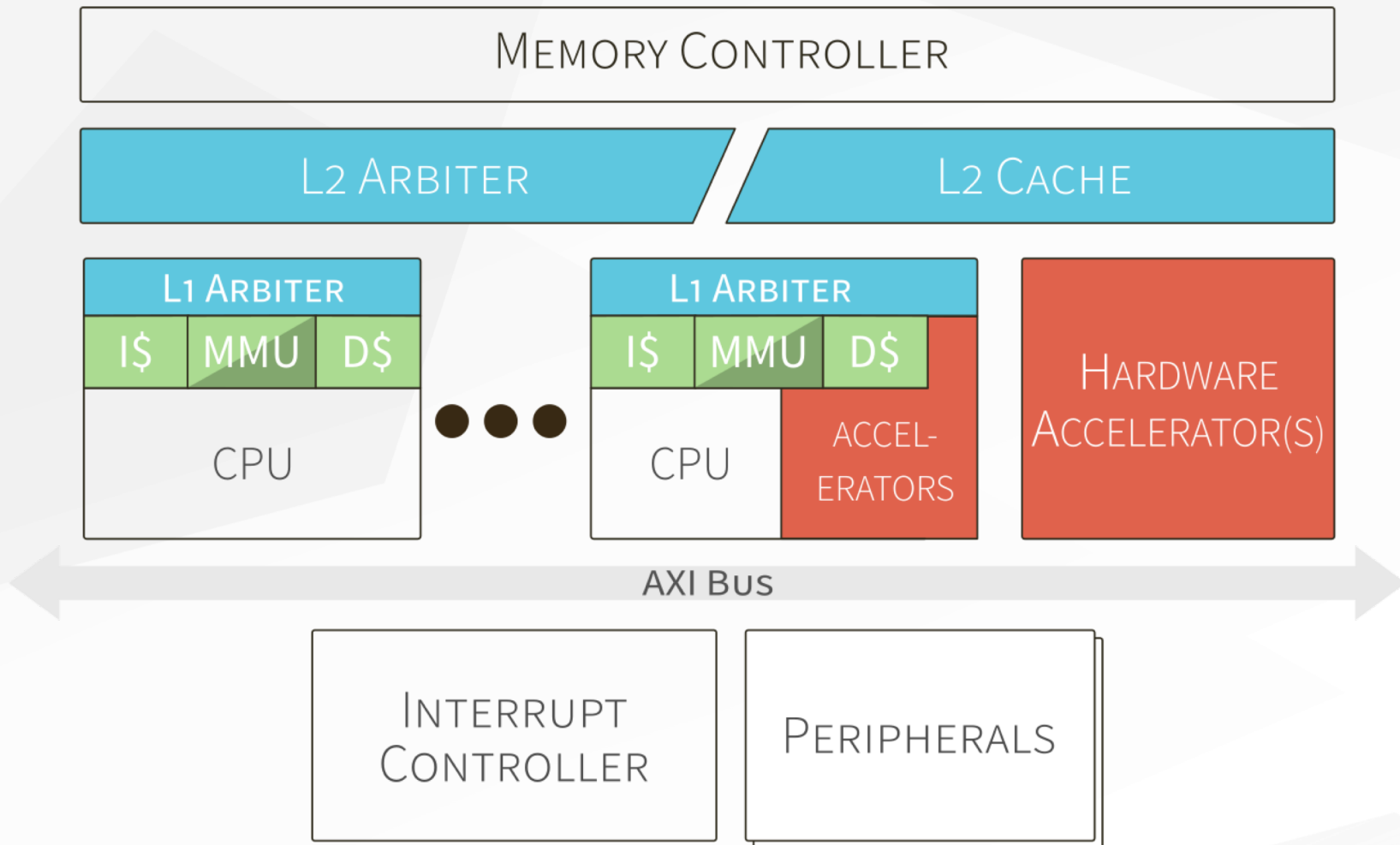
PERFORMANCE COMPARISON (1)



PERFORMANCE COMPARISON (2)



MULTICORE FUTURE POSSIBILITIES (1)



MULTICORE FUTURE POSSIBILITIES (2)

- Asymmetric configurations
- Cache configurations
- Heterogeneous configurations
 - Accelerators
 - Custom Instructions
 - Loosely integrated
 - ISA subsets (FPU/no FPU, Divider/no Divider)

CONCLUSIONS / FUTURE WORK

Taiga provides a high level of flexibility in its structure and configuration options for exploration of heterogeneous accelerator systems

Future Work:

- Linux support
 - (Privileged Instruction set support)
- 32-bit/64-bit FPU
- Multicore support

AVAILABLE AT:

<https://gitlab.com/sfu-rcl/Taiga>

CONFIGURATION OPTIONS

- Multiplier Unit
- Division Unit
- Branch Predictor
- TLBs/MMUs
- Scratchpad
- Caches
- Queue Sizes

● Entries (2+)

● Instruction/Data

● Size (fully configurable)

● Associativity (1+)

● Line Size, in words (4+)

● Lines per way (1+)