Politecnico di Torino

UVM verification for Multicore architecture RISC-V

MASTER'S DEGREE THESIS

SUPERVISOR:
PROF. ERNESTO SANCHEZ
PROF.RUOSPO ANNACHIARA

CANDIDATE:
ISMIANA QOSE

AGENDA

INTRODUCTION

METHODOLOGY

DESIGN AND IMPLEMENTATION

EXPERIMENTAL RESULTS

CONCLUSION

INTRODUCTION

▶ BACKGROUND

RISC-V

- RISC-V IS AN OPEN, ROYALTY-FREE INSTRUCTION SET ARCHITECTURE (ISA) BASED ON THE PRINCIPLES OF REDUCED INSTRUCTION SET COMPUTING (RISC).
- IT IS MODULAR AND EXTENSIBLE, SUPPORTING A WIDE RANGE OF APPLICATIONS FROM EMBEDDED SYSTEMS TO HIGH-PERFORMANCE COMPUTING.
- THE OPEN NATURE OF RISC-V ENCOURAGES INNOVATION, ACADEMIC RESEARCH, AND INDUSTRY ADOPTION.

INTRODUCTION

▶ BACKGROUND

CVA6

- 6-STAGE PIPELINE, 32- OR 64-BIT, IN-ORDER ISSUE, OUT-OF-ORDER EXECUTION, IN-ORDER COMMIT

- CONFIGURABLE SIZE, SEPARATE TLBS, A HARDWARE PTW AND BRANCH-PREDICTION (BTB AND BHT)

- DESIGNED FOR FLEXIBILITY, SUPPORTING FEATURES LIKE AXI4 MEMORY INTERFACES, RVFI (RISC-V FORMAL INTERFACE) FOR VERIFICATION, AND INTEGRATION WITH VARIOUS SOC PLATFORMS.

INTRODUCTION

▶ BACKGROUND

MULTI-CORE

NETWORK-ON-CHIP (OPENPITON)

PROS:

- EACH CORE/TILE CONNECTS TO A ROUTER IN A MESH OR TORUS NETWORK
- HIGHLY SCALABLE (10S–1000S OF CORES)
- SUPPORTS ADVANCED FEATURES (CACHE COHERENCE, FLEXIBLE COMMUNICATION)

CONS:

- HIGHER DESIGN AND VERIFICATION COMPLEXITY.
- MORE AREA AND POWER OVERHEAD.

INTRODUCTION

▶ MOTIVATION

DUAL-CVA6 SYSTEM

PROS:

- SIMPLE AND EFFICIENT FOR 2–4 CORES
- EASY INTEGRATION WITH STANDARD IP AND VERIFICATION TOOL
- SUPPORTS HIGH THROUGHPUT AND FLEXIBLE ARBITRATION

CONS:

- LIMITED SCALABILITY: COMPLEXITY AND RESOURCE USAGE GROW RAPIDLY .
- ARBITRATION OVERHEAD: MORE CORES LEAD TO INCREASED CONTENTION AND BOTTLENECKS
- NOT IDEAL FOR MANYCORE SYSTEMS

INTRODUCTION

▶ OBJECTIVES

- IMPLEMENT A DUAL-CORE CVA6 PLATFORM WITH SHARED MEMORY
- USE AN INTERCONNECT SCHEME FOR EFFICIENT ARBITRATION BETWEEN CORES
- INTEGRATE INVALIDATION LOGIC TO ENSURE SAFE AND CONSISTENT MEMORY ACCESS
- ENABLE ROBUST MULTICORE OPERATION AND SYSTEMATIC VERIFICATION FOR ADVANCED RISC-V RESEARCH

METHODOLOGY

1 ARCHITECTURE DEFINITION

DEFINE ARCHITECTURE WITH DUAL-CVA6 CORES AND SHARED MEMORY

2 INTERCONNECT

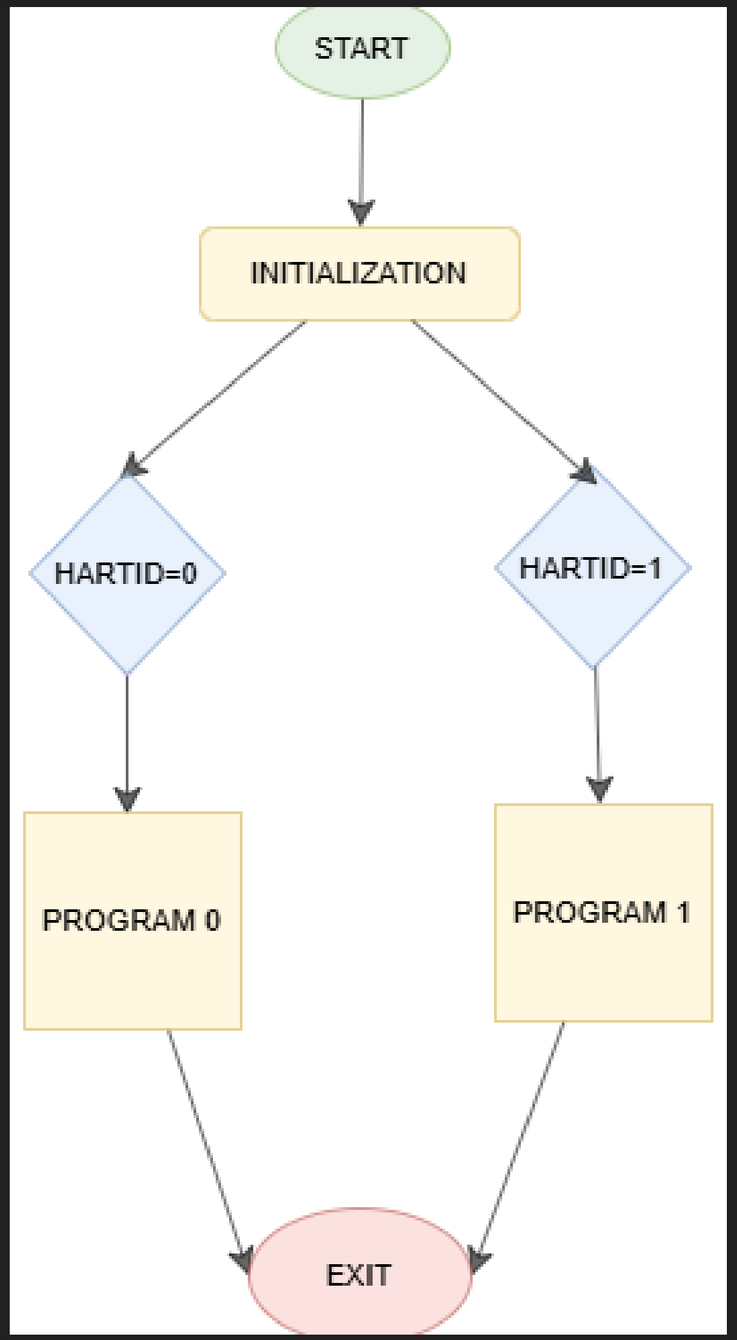
CONNECT BOTH CORES TO MEMORY VIA A MUX FOR ARBITRATION

3 AW-CHANNEL-BASED CACHE INVALIDATION

INTER-CORE CACHE COHERENCE BY MONITORING WRITE REQUESTS

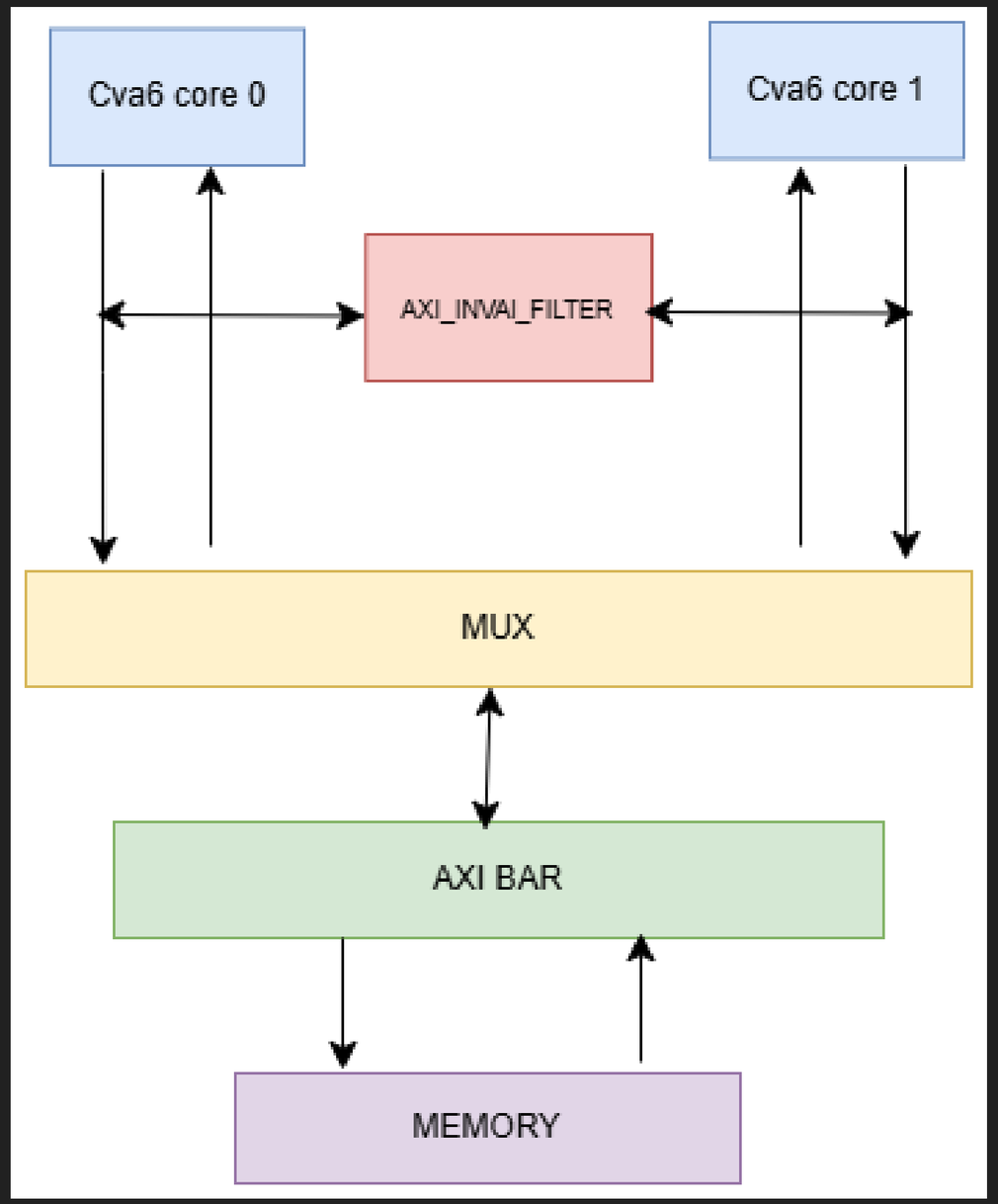
4 UVM-BASED VERIFICATION

UVM PROVIDES A ROBUST AND REUSABLE VERIFICATION ENVIRONMENT ENABLING SYSTEMATIC BUG DETECTION



DESIGN AND IMPLEMETATION

- DUAL-CORE CVA6 ARCHITECTURE AND SHARED MEMORY ACCESS
- AW-CHANNEL-BASED CACHE INVALIDATION FOR INTER-CORE COHERENCE
- VERIFICATION FRAMEWORK



DUAL-CORE CVA6 ARCHITECTURE

- **ROUND-ROBIN ARBITRATIION**

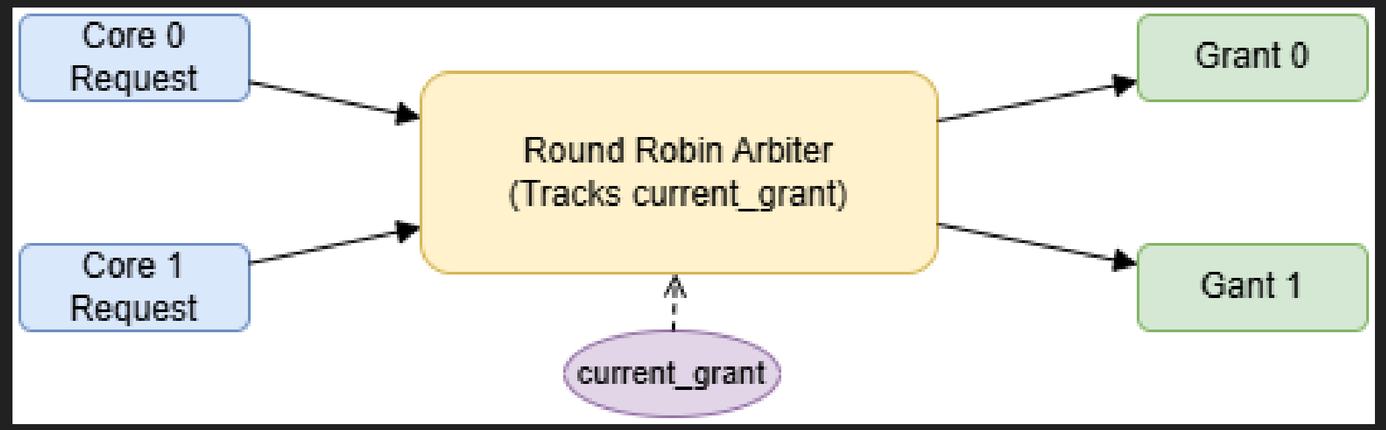
MUX GRANTS MEMORY ACCESS TO ONE CORE AT A TIME

- **ID PREPENDING**

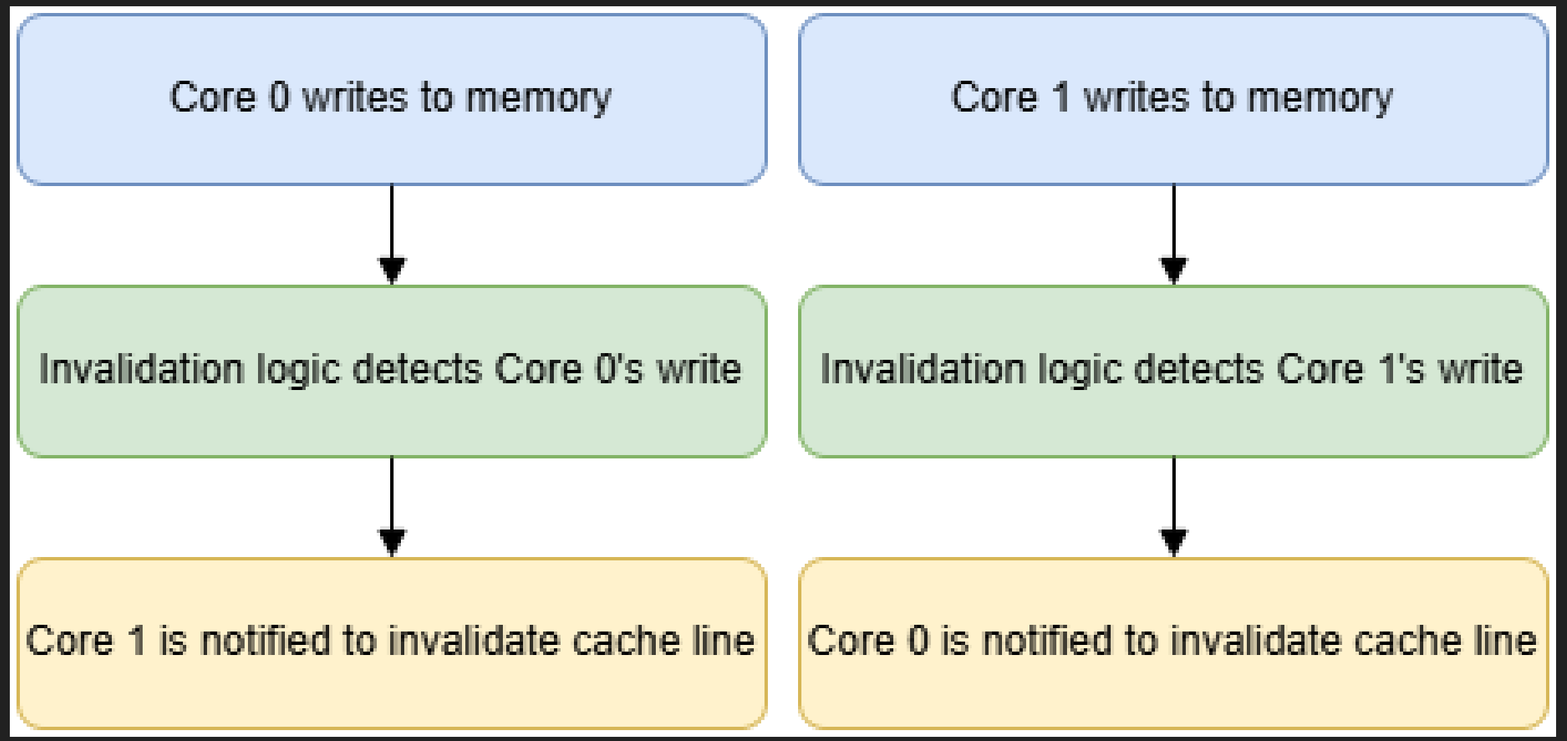
CORE ID IS ADDED TO AXI ID TO TRACK TRANSACTION OWNERSHIP

- **RESPONSE ROUTING**

RESPONSES ARE ROUTED BACK TO THE CORRECT CORE BY DECODING PREPENDED ID



AW-CHANNEL-BASED CACHE INVALIDATION FOR INTER-CORE COHERENCE

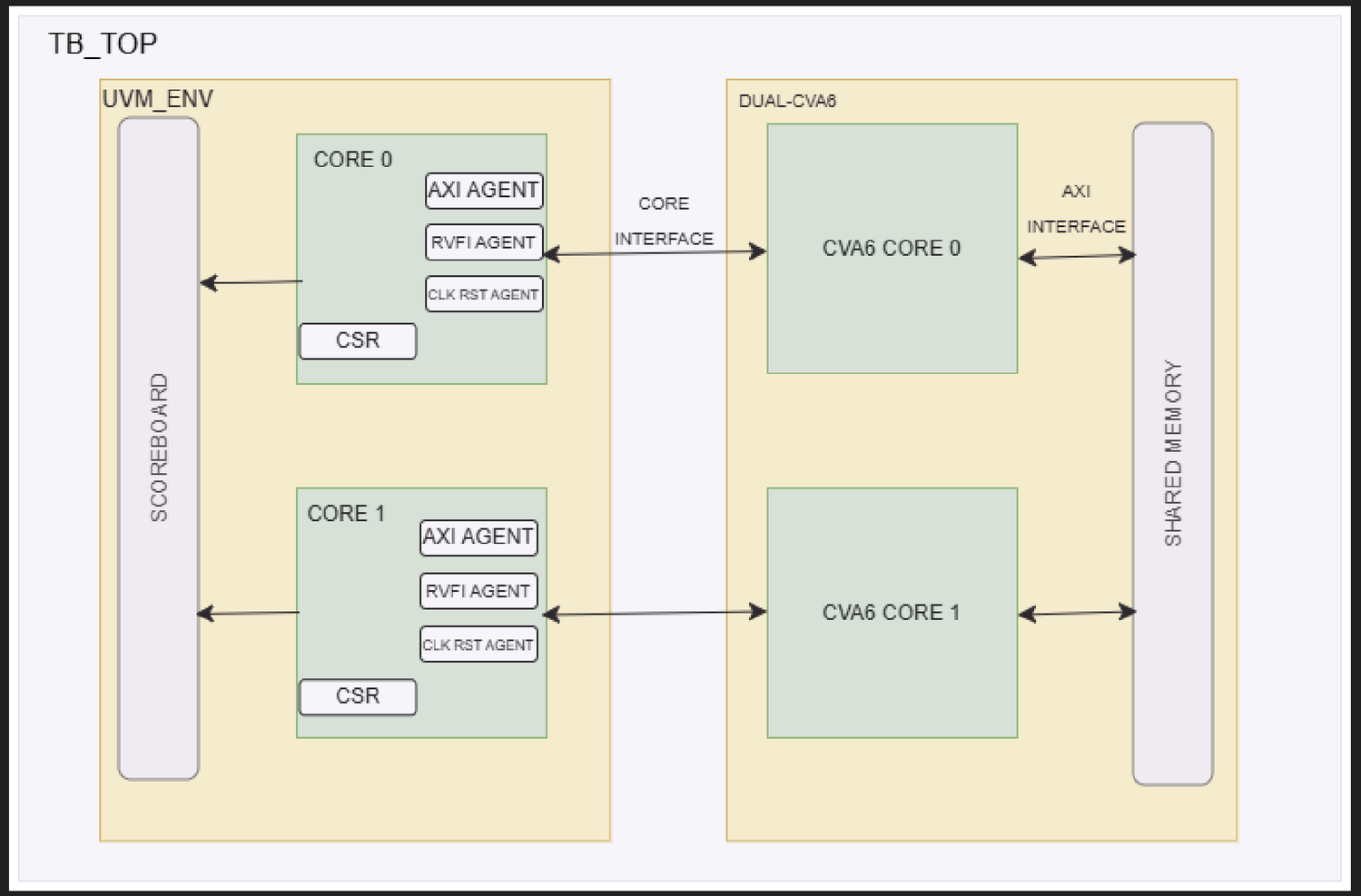


PURPOSE

- MONITORS WRITE REQUESTS ON THE AXI WRITE ADDRESS CHANNEL FROM EACH CORE
- FORWARDS ADDRESS AND VALID SIGNAL TO THE OTHER CORE WHEN A MEMORY ACCESS OCCURS
- ENSURES BOTH CORES ARE PROMPTLY NOTIFIED OF PEER MEMORY WRITES, PREVENTING STALE CACHE DATA

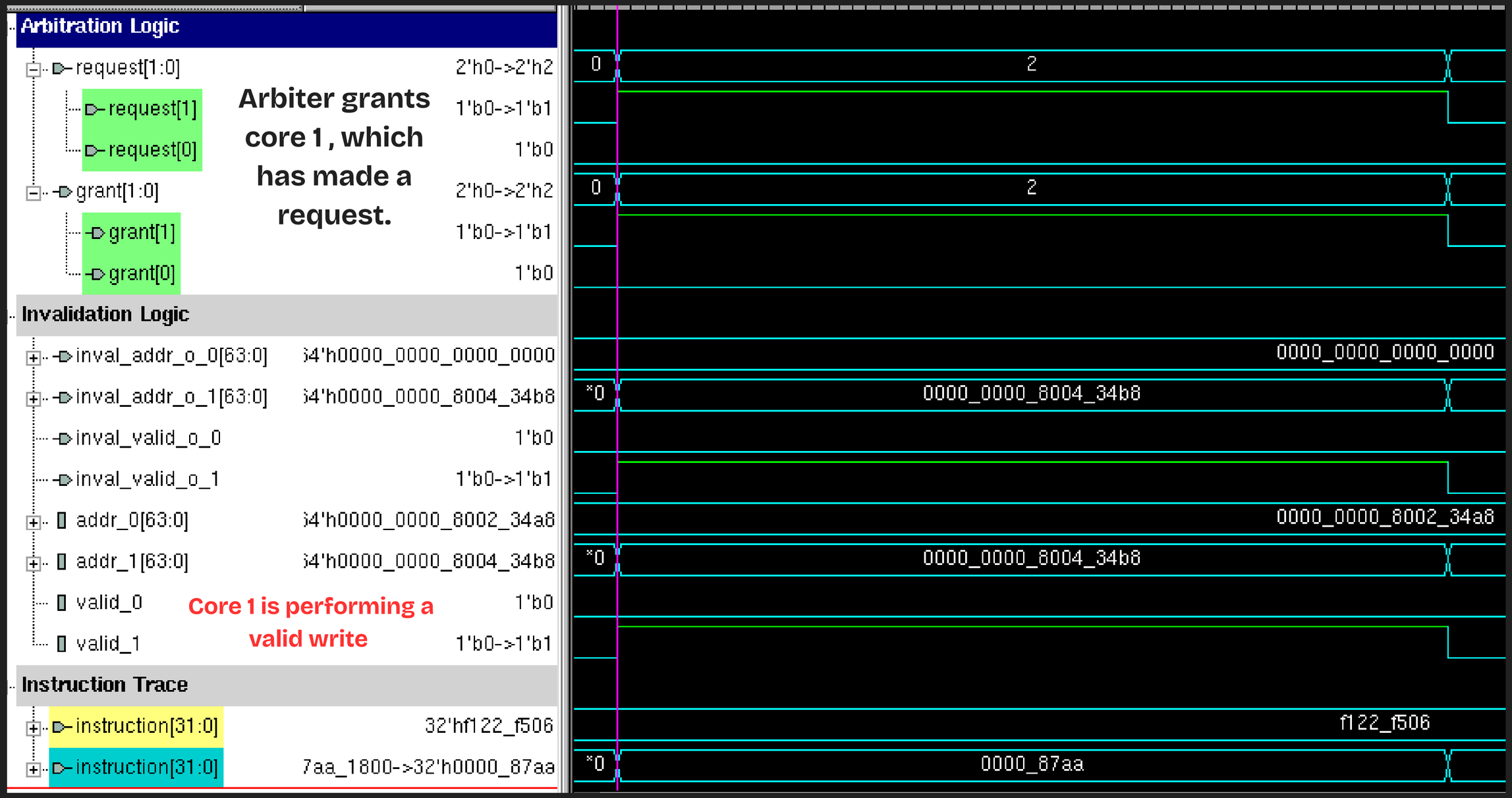
VERIFICATION FRAMEWORK

UVM ENABLES SYSTEMATIC VERIFICATION BY STRUCTURING THE TESTBENCH INTO REUSABLE AGENTS AND SCOREBOARDS FOR EACH CORE



- AUTOMATICALLY GENERATES AND CHECKS TRANSACTIONS (AXI, RVFI, CSR) FOR BOTH CORES
- COLLECTS RESULTS IN A SCOREBOARD FOR AUTOMATED CHECKING
- ENABLED DETECTION OF UNCOVERED LOGIC AND IMPROVES TESTBENCH QUALITY

WAVEFORM RESULTS



COVERAGE RESULTS

	Score	Toggle	Assert	Line	Condition
cva6_tb_wrapper_i	40.09%	10.54%		61.79%	47.95
cva6_axi_bus	17.11%	17.11%			
i_axi_inval_filter	71.77%	15.31%		100.00%	100.00
i_axi_master_connect_cva6_to...	29.65%	9.30%			50.00
i_cva6	40.26%	11.28%		61.21%	48.27
i_cva6_1	39.85%	10.89%		61.12%	47.53
i_cva6_axi2mem	46.75%	23.47%		72.32%	44.44
i_cva6_rvfi	55.87%	10.01%		98.88%	58.71
i_round_robin_arbiter	100.00%	100.00%		100.00%	
i_rvfi_tracer	29.36%	3.50%		53.33%	31.25
i_sram	74.69%	52.09%		97.30%	

Each core has slightly different coverage because of different program running on them

Arbitration

Line | Toggle | FSM | Condition | Branch | Assert

Variable	Type	Coverage	Display
clk	port	100.00%	
current_grant	signal	100.00%	
grant[1:0]	port	100.00%	
request[1:0]	port	100.00%	
reset	port	100.00%	

Every bit of each signal has toggled between logic 0 and 1 during simulation

Line | Toggle | FSM | Condition | Branch | Assert

Category	Coverage
Block	100.00%
Statement	100.00%

Every block and line in the arbiter code has been executed during simulation.

INVALIDATION

Line | Toggle | FSM | Condition | Branch | Assert

Only a small subset of all possible address values are being used for invalidation

Variable	Type	Coverage	Display
addr_0[63:0]	signal	11.72%	
addr_1[63:0]	signal	14.06%	
clk_i	port	100.00%	
inval_addr_o_0[63:0]	port	15.62%	
inval_addr_o_1[63:0]	port	14.06%	
inval_valid_o_0	port	100.00%	
inval_valid_o_1	port	100.00%	
rst_ni	port	100.00%	
slv_req_i_0_aw_addr...	port	11.72%	
slv_req_i_0_aw_valid	port	100.00%	
slv_req_i_1_aw_addr...	port	14.06%	
slv_req_i_1_aw_valid	port	100.00%	
valid_0	signal	100.00%	
valid_1	signal	100.00%	

Only writes (AW channel) trigger invalidation

INVALIDATION

Line | Toggle | FSM | Condition | Branch | Assert

CORE 0

*

Expression	Coverage	Line No.
(valid_0 && addr_0)	100.00%	42
(valid_1 && addr_1)	100.00%	51

```
(valid_0 && addr_0)
 ---1---    --2---
```

Coverage	1	2
✓	0	1
✓	1	0
✓	1	1

All possible combinations of the two inputs have been exercised during simulation.

Line | Toggle | FSM | Condition | Branch | Assert

CORE 1

*

Expression	Coverage	Line No.
(valid_0 && addr_0)	100.00%	42
(valid_1 && addr_1)	100.00%	51

```
(valid_1 && addr_1)
 ---1---    --2---
```

Coverage	1	2
✓	0	1
✓	1	0
✓	1	1

Line | Toggle | FSM | Condition | Branch | Assert

Category	Coverage
Block	100.00%
Statement	100.00%

Every procedural line and every code block has been executed for both cores

AREA REPORT

METRIC	2-CVA6	ARBITRATION	INVALIDATION
Num of ports	4902	6	262
Num of nets	484708	13	303
Num of cells	450816	8	173
Num of combinational cells	401610	7	173
Num of sequiental cells	48212	1	0
Num of buf/inv	67688	3	1
Num of references	164	5	6
AREA µm²	**2-CVA6**	**ARBITRATION**	**INVALIDATION**
Combinational Area	466.018,702	55,86000	196,840003
Buf/inv Area	42.028,5318	15,96000	7,98000
Noncombinational Area	256.190,46	53,22300	-
Absolute total Area	722.209,162	125,043	204,82

2 CVA6 cores have 466018 µm² area, Arbitration has 55,86 µm²; integration increases area by 0.018%

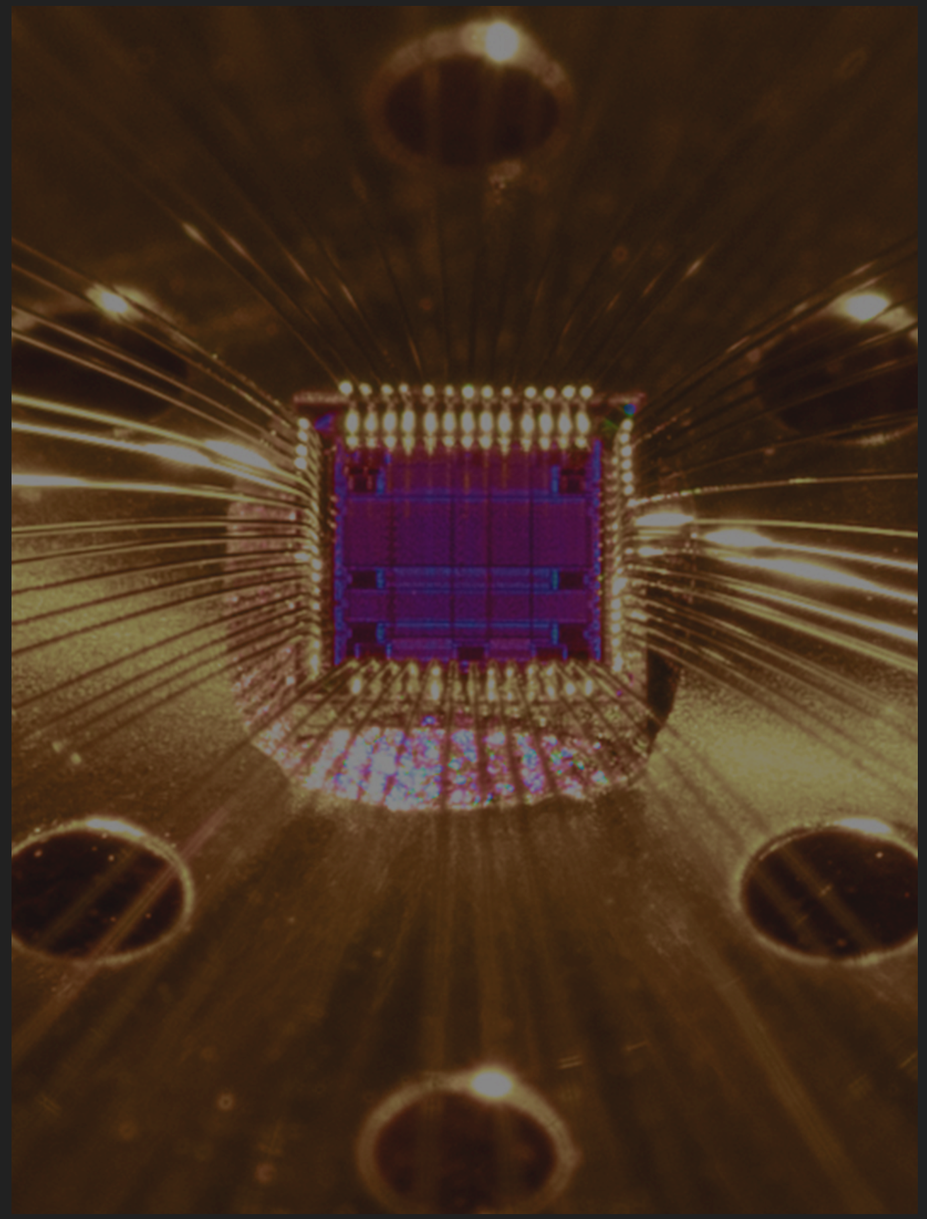
Invalidation has 196,84 µm²; integration increases area by 0.03%

CONCLUSION

- DUAL-CORE CVA6 SYSTEM WITH INTERCONNECT MECHANISM AND INVALIDATION LOGIC.
- ENSURES CACHE COHERENCE ON AW CHANNEL BY NOTIFYING EACH CORE OF WRITE OPERATIONS
- GUARANTEES FAIR AND STARVATION-FREE MEMORY ACCESS

FUTURE WORK

EXPAND THE DESIGN TO MORE CORES AND ADD CACHE COHERENCE PROTOCOLS



THANK YOU