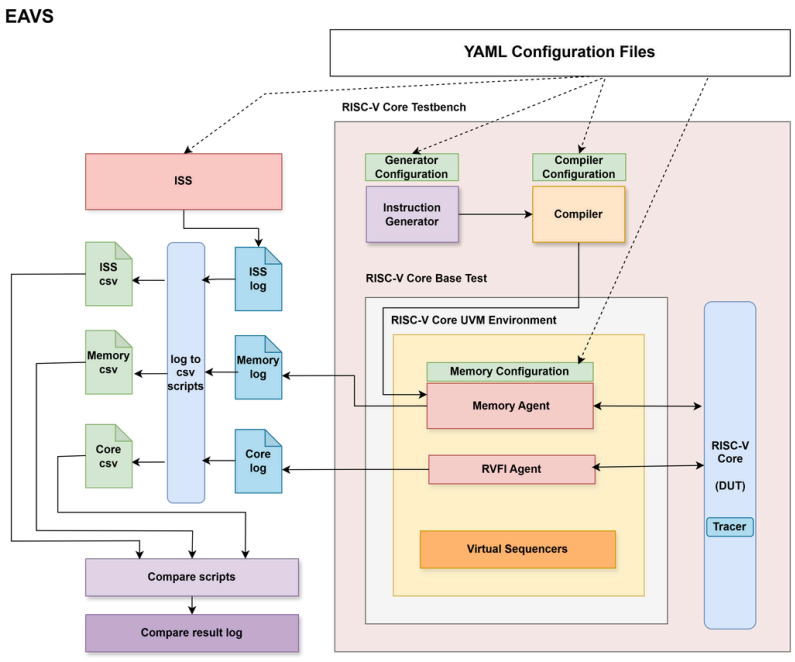
EAVS

Electra IC Advanced Verification Suite for RISC-V Cores

Merve Eyüboğlu, Murat Tökez, Ibrahim Mouamar Ali Ahmed,
Melike Atay Karabalkan, Berna Ors

WHAT IS EAVS?

Electra IC Advanced Verification Suite (EAVS), has been developed to facilitate the verification of any RISC-V core. EAVS comprises an Instruction Set Simulator (ISS), YAML configuration files, and a RISC-V Core UVM Testbench.The RISC-V Core UVM Testbench contains a RISC-V Core, referred to as the Design Under Test (DUT), along with an Instruction Generator, Compiler, and a RISC-V Core Base Test.



COMPONENTS

The complex log format is obtained from Spike is subsequently converted into a .csv file. Spike is currently executed externally rather than being integrated into the UVM environment. YAML configuration files are used to manage test parameters (e.g., boot address, RAM width, peripheral addresses) and verification scenarios (e.g., floating-point, loop, and CSR tests), ensuring consistent memory mapping. The memory map is adjusted to avoid conflicts with Spike's embedded memory layout.

EAVS-DV: INSTRUCTION GENERATOR

In this study, COREV-DV layered on top of Google riscv-dv has been used. We propose EAVS-DV as an enhancement to COREV-DV. The improvements are:

- All fixed address spaces in COREV-DV have been parameterized in EAVS-DV to enable compatibility with any DUT that has memory address limitations.
- Spike has specific fixed memory expectations, so virtual peripheral register addresses have been updated to prevent collisions with Spike's memory space.
- Spike requires detecting specific values at the ``tohost'' and ``fromhost'' addresses to identify program termination. EAVS-DV adds terminating instructions at the end of the generated instruction list to prevent an infinite loop in Spike.

UVM ENVIRONMENT

The environment is developed in accordance with the UVM testbench architecture, enabling detailed monitoring of the core's program counter (PC), instruction and data memory, and register file behavior. Key components, including the Memory Agent, RVFI Agent, Scoreboard, Virtual Sequencers, and Coverage, collaborate to ensure comprehensive verification. The Memory Agent manages both instruction and data memory, driving instructions and data to the DUT while monitoring the program counter and data addresses. The RVFI Agent functions as a passive monitor, tracking RVFI signals across the pipeline stages.

LOG RESULTS

```
TIME        | RVFI |    CYCLE|  ORDER |    PC    |   INSTR   | M | RS1 | RS1_DATA | RS2 | RS2_DATA | RD  | RD_DATA
146961.000 ns | RVFI |    48981 |  11472 | 8000986c | 80400bb7 | M | x0  | 00000000 | x4  | 00000000 | x23 | 80400000
146970.000 ns | RVFI |    48984 |  11473 | 80009870 | 00001497 | M | x0  | 00000000 | x0  | 00000000 | x9  | 8000a870
146973.000 ns | RVFI |    48985 |  11474 | 80009874 | 8104a483 | M | x9  | 8000a870 | x16 | ffffffff | x9  | 075bcd15
146979.000 ns | RVFI |    48987 |  11475 | 80009878 | 009ba023 | M | x23 | 80400000 | x9  | 075bcd15 | x0  | 8000a870
146982.000 ns | RVFI |    48988 |  11476 | 8000987c | 30405073 | M | x0  | 00000000 | x4  | 00000000 | x0  | 00000000
```

RVFI Agent's Log File

```
core   0: 0x8000986c (0x80400bb7) lui     s7, 0x80400
core   0: 3 0x8000986c (0x80400bb7) x23 0x80400000
core   0: 0x80009870 (0x00001497) auipc   s1, 0x1
core   0: 3 0x80009870 (0x00001497) x9  0x8000a870
core   0: 0x80009874 (0x8104a483) lw      s1, -2032(s1)
core   0: 3 0x80009874 (0x8104a483) x9  0x075bcd15 mem 0x8000a080
core   0: 0x80009878 (0x009ba023) sw      s1, 0(s7)
core   0: 3 0x80009878 (0x009ba023) mem 0x80400000 0x075bcd15
core   0: 0x8000987c (0x30405073) csrwi   mie, 0
core   0: 3 0x8000987c (0x30405073) c772_mie 0x00000000
```

Spike's Log File

pc	instr	gpr	mem	csr	binary	mode	instr	operand
0x8000986c	lui	s7:0x80400000			0x80400bb7	3	"lui s7 0x80400"	"s7 0x80400"
0x80009870	auipc	s1:0x8000a870			0x00001497	3	"auipc s1 0x1"	"s1 0x1"
0x80009874	lw	s1:0x075bcd15	mem[0x8000a080]		0x8104a483	3	"lw s1 -2032(s1)"	"s1 1 -2032"
0x80009878	sw		mem[0x80400000]:0x075bcd15		0x009ba023		"sw s1 0(s7)"	"s1 s7 0"
0x8000987c	csrrwi			mie:0x00000000	0x30405073		"csrwi mie 0"	"zero mie 0"

Spike and Core Random Arithmetic Test Results

COMPARISON & CONCLUSION

Log files are converted to .csv format. Because the Spike ISS logs memory contents only for load instructions, both data and addresses are compared for loads, while only addresses are verified for stores. The mode field represents the privilege level, aligning with the M column in the RVFI log and lines labeled 3 in the Spike ISS log. A comparison of the CSV outputs confirms that all random tests generated by EAVS-DV are successfully executed on the cv32e40p RISC-V core. The primary advantage of this design is the introduction of EAVS-DV as an enhancement over COREV-DV. All fixed address spaces in COREV-DV are parameterized in EAVS-DV to enable compatibility with any DUT and Spike that has memory address limitations.