

The 5th China Functional Programming Meetup (Shanghai 2021)

Scala-based FOSS EDA on ARM

Feng Li (李枫)

hkli2013@126.com

Jul 10, 2021



Agenda

I. Overview

- Open Source EDA
 - Evolution of HDL
 - RISC-V
 - Testbed
-

II. Chisel/SpinalHDL on ARM

- Introduction
- Scala on ARM
- Chisel/SpinalHDL on RPi4

III. Speed up Chisel/SpinalHDL

- GraalVM
- Acceleration of Chisel/SpinalHDL

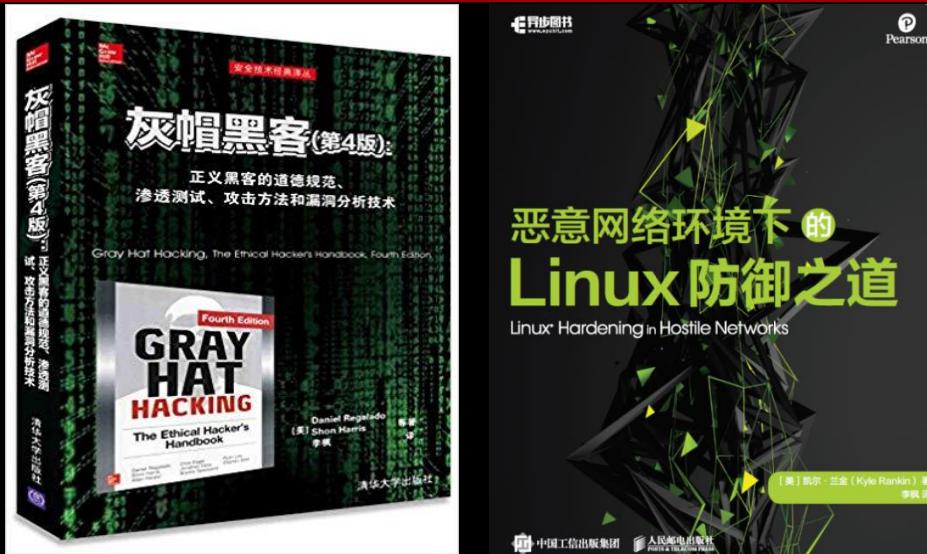
IV. Scala 3

- Scala 3
- Dotty with GraalVM

V. Wrap-up

Who Am I

- The main translator of the book «Gray Hat Hacking The Ethical Hacker's Handbook, Fourth Edition» (ISBN: 9787302428671) & «Linux Hardening in Hostile Networks, First Edition» (ISBN: 9787115544384)



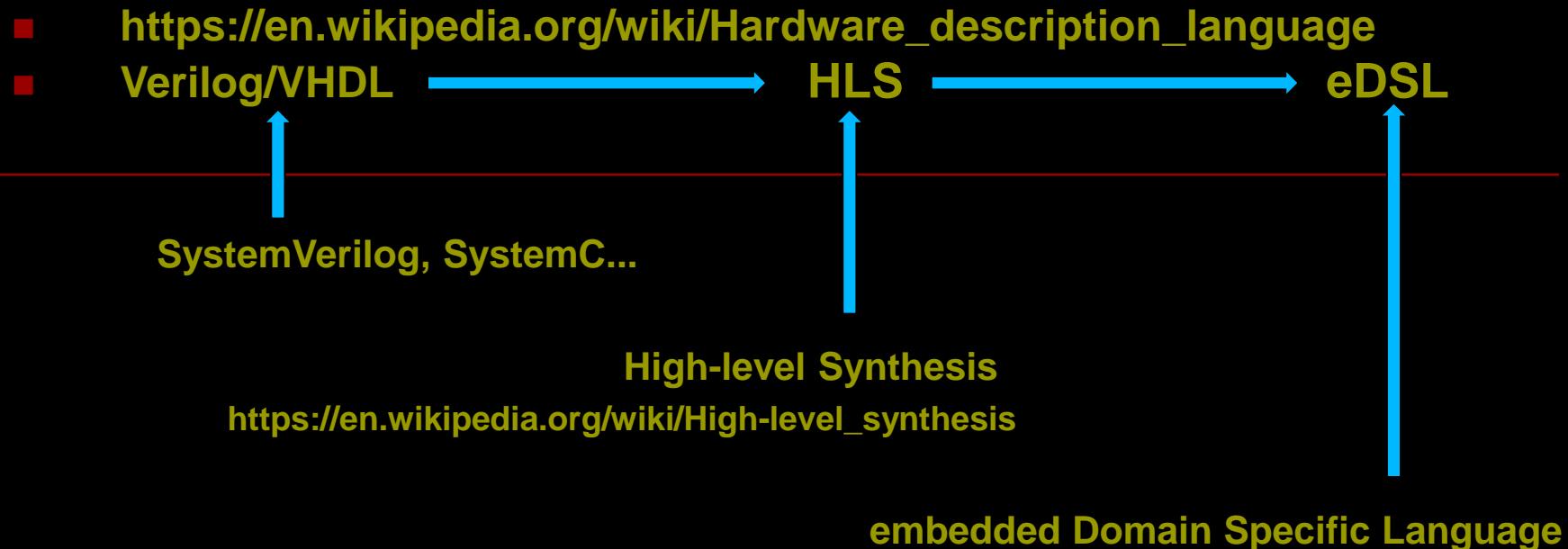
- Pure software development for ~15 years
- Actively participate in various activities of the open source community
 - <https://github.com/XianBeiTuoBaFeng2015/MySlides/tree/master/Conf>
 - <https://github.com/XianBeiTuoBaFeng2015/MySlides/tree/master/LTS>
- Recently, focus on infrastructure of Cloud/Edge Computing, AI, Virtualization, Program Runtimes, Network, 5G, RISC-V, EDA...

I. Overview

1) Open Source EDA

- https://en.wikipedia.org/wiki/Comparison_of_EDA_software
- <https://sem wiki.com/wikis/industry-wikis/eda-open-source-tools-wiki/>
- <https://ieeexplore.ieee.org/document/9398963>
- <https://ieeexplore.ieee.org/document/9398960>
- <https://ieeexplore.ieee.org/document/9336682>
- <https://ieeexplore.ieee.org/document/9105619>
- ...

2) Evolution of HDL



Typical eDSLs

- **Haskell as host**
Bluespec...
- **Scala as host**
Chisel, SpinalHDL...
- **Python as host**
FHDL...
- ...

3) RISC-V

Overview

- <https://en.wikipedia.org/wiki/RISC-V>
- <https://riscv.org/>

Designer	University of California, Berkeley
Bits	32 • 64 • 128
Introduced	2010
Version	unprivileged ISA 20191213, ^[1] privileged ISA 20190608 ^[2]
Design	RISC
Type	Load-store
Encoding	Variable
Branching	Compare-and-branch
Endianness	Little ^{[1][3]}
Page size	4 KiB
Extensions	M: Multiplication A: Atomics — LR/SC & fetch-and-op F: Floating point (32-bit) D: FP Double (64-bit) Q: FP Quad (128-bit) Zicsr: Control and status register support Zifencei: Load/store fence C: Compressed instructions(16-bit)
Open	Yes, and royalty free
Registers	16 32 (including one always-zero register)
Floating point	32(F extension)/64 (D extension)/128 (Q extension) (optional)

ISA base and extensions				
Name	Description	Version	Status ^[a]	Instruction Count
Base				
RVWMO	Weak Memory Ordering	2.0	Ratified	
RV32I	Base Integer Instruction Set, 32-bit	2.1	Ratified	49
RV32E	Base Integer Instruction Set (embedded), 32-bit, 16 registers	1.9	Open	49
RV64I	Base Integer Instruction Set, 64-bit	2.1	Ratified	14
RV128I	Base Integer Instruction Set, 128-bit	1.7	Open	14
Extension				
M	Standard Extension for Integer Multiplication and Division	2.0	Ratified	8
A	Standard Extension for Atomic Instructions	2.1	Ratified	11
F	Standard Extension for Single-Precision Floating-Point	2.2	Ratified	25
D	Standard Extension for Double-Precision Floating-Point	2.2	Ratified	25
Zicsr	Control and Status Register (CSR)	2.0	Ratified	
Zifencei	Instruction-Fetch Fence	2.0	Ratified	
G	Shorthand for the IMAFDZicsr Zifencei base and extensions, intended to represent a standard general-purpose ISA	N/A	N/A	
Q	Standard Extension for Quad-Precision Floating-Point	2.2	Ratified	27
L	Standard Extension for Decimal Floating-Point	0.0	Open	
C	Standard Extension for Compressed Instructions	2.0	Ratified	36
B	Standard Extension for Bit Manipulation	0.93	Open	42
J	Standard Extension for Dynamically Translated Languages	0.0	Open	
T	Standard Extension for Transactional Memory	0.0	Open	
P	Standard Extension for Packed-SIMD Instructions	0.2	Open	
V	Standard Extension for Vector Operations	1.0RC	Open	186
N	Standard Extension for User-Level Interrupts	1.1	Open	3
H	Standard Extension for Hypervisor	0.4	Open	2
S	Standard Extension for Supervisor-level Instructions ^[26]	1.12	Open	7
Zam	Misaligned Atomics	0.1	Open	
Ztso	Total Store Ordering	0.1	Frozen	

■ Assembly

<https://github.com/riscv/riscv-asm-manual/blob/master/riscv-asm.md>

Register name	Symbolic name	Description	Saved by
32 integer registers			
x0	Zero	Always zero	
x1	ra	Return address	Caller
x2	sp	Stack pointer	Callee
x3	gp	Global pointer	
x4	tp	Thread pointer	
x5	t0	Temporary / alternate return address	Caller
x6–7	t1–2	Temporary	Caller
x8	s0/fp	Saved register / frame pointer	Callee
x9	s1	Saved register	Callee
x10–11	a0–1	Function argument / return value	Caller
x12–17	a2–7	Function argument	Caller
x18–27	s2–11	Saved register	Callee
x28–31	t3–6	Temporary	Caller
32 floating-point extension registers			
f0–7	ft0–7	Floating-point temporaries	Caller
f8–9	fs0–1	Floating-point saved registers	Callee
f10–11	fa0–1	Floating-point arguments/return values	Caller
f12–17	fa2–7	Floating-point arguments	Caller
f18–27	fs2–11	Floating-point saved registers	Callee
f28–31	ft8–11	Floating-point temporaries	Caller

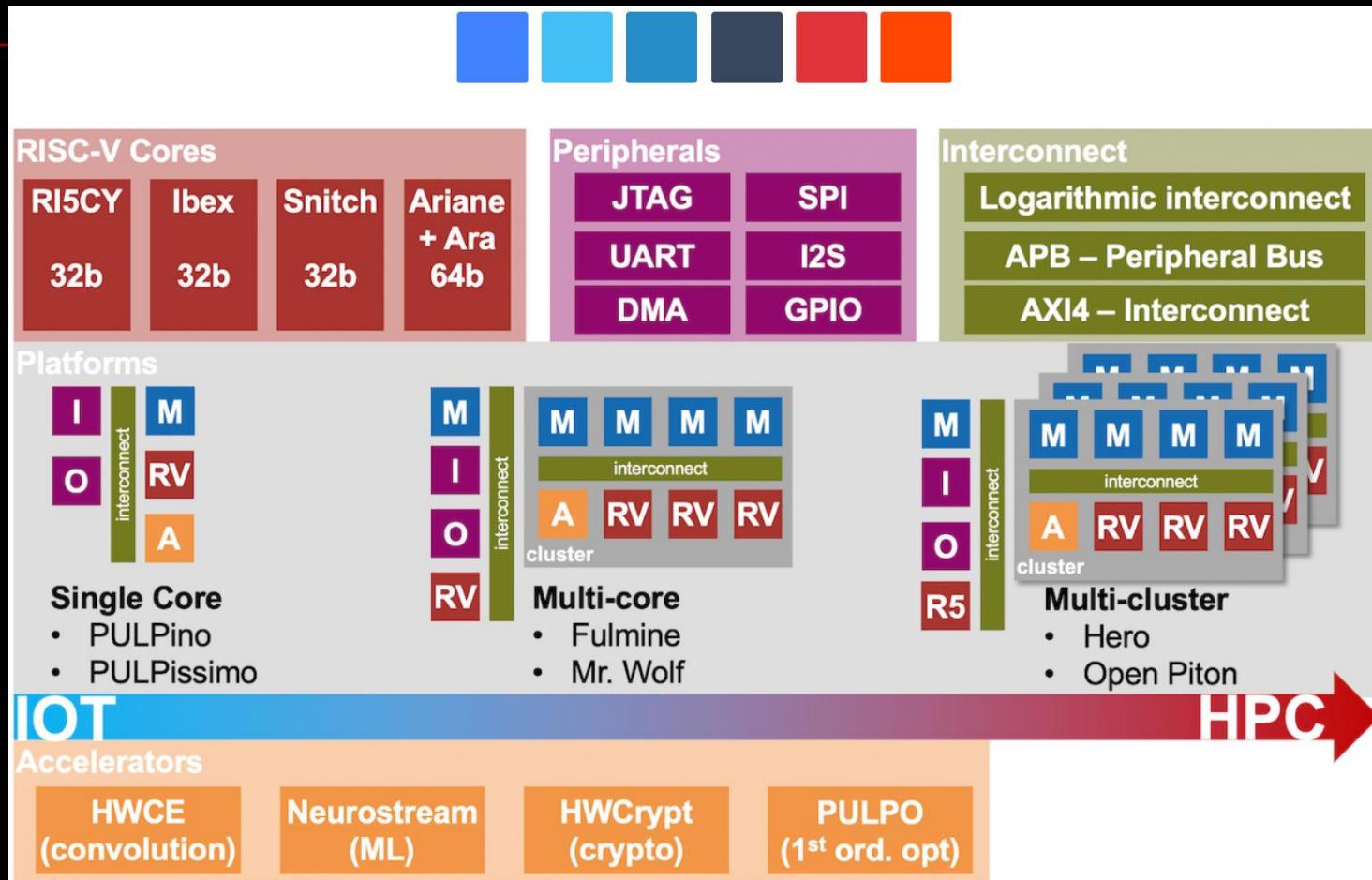
- **Documents**

<https://riscv.org/technical/specifications/>

<https://wiki.riscv.org/display/TECH>

3.1 Cores & SoC

- <https://github.com/riscv/riscv-cores-list>
- <https://riscv.org/exchange/cores-socs/>



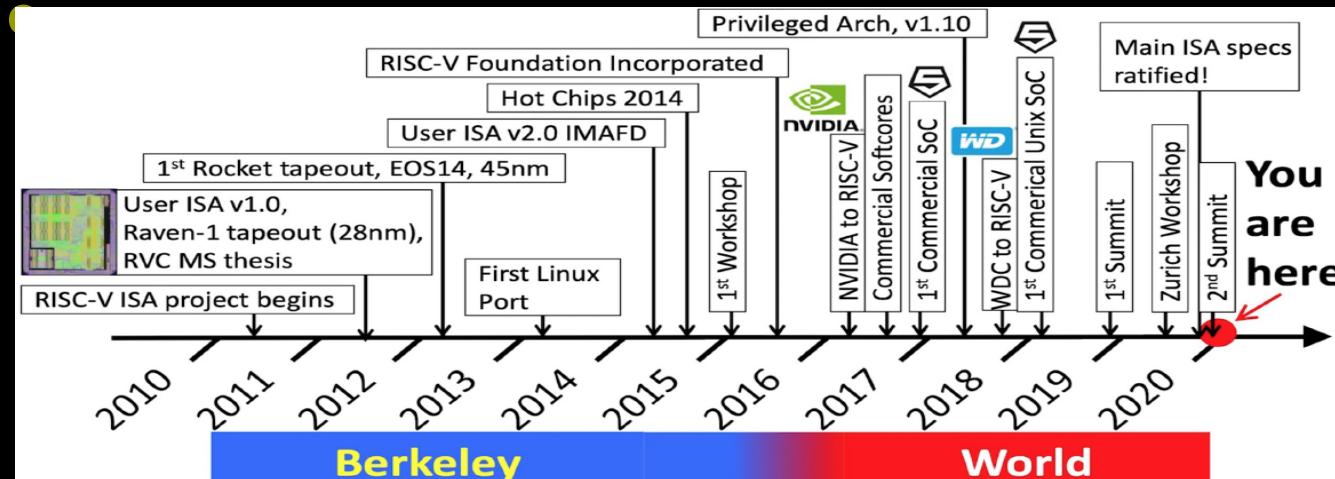
Source: <https://fuentitech.com/what-is-the-risc-v-ecosystem/19227/>

3.2 Development

- <https://riscv.org/exchange/>
 - <https://github.com/riscv/riscv-isa-manual>
-

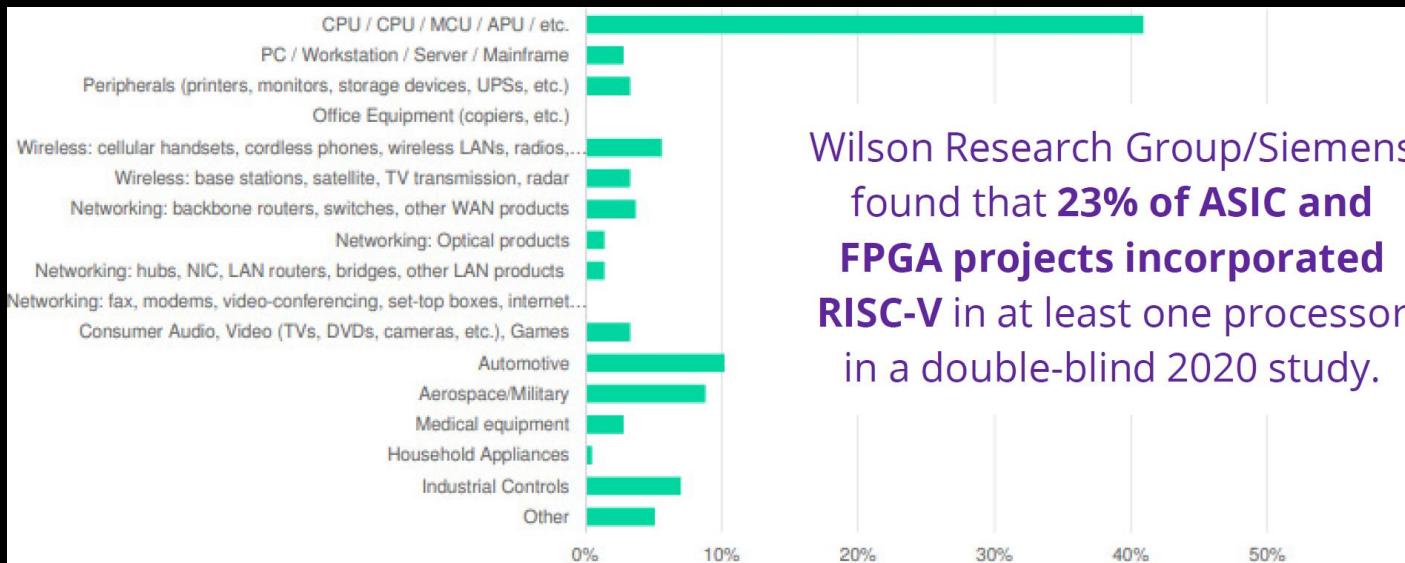
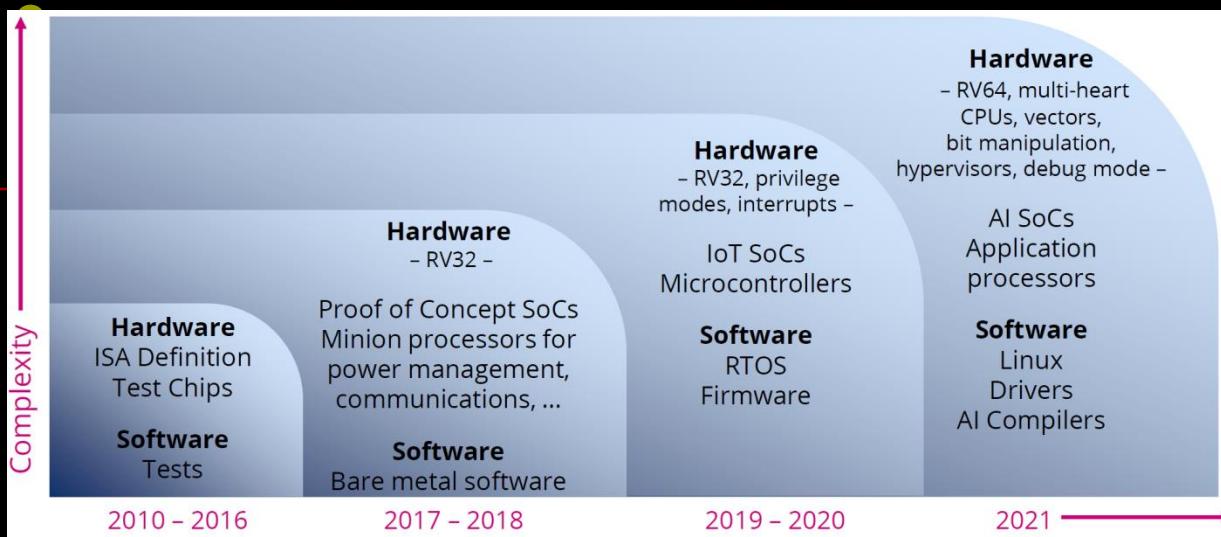
3.3 Ecosystem

- <https://riscv.org/exchange/>
- <https://community.riscv.org/events/details/risc-v-foundation-bay-area-risc-v-group-presents-2021-risc-v-ecosystem-updates/>
- <https://fuentitech.com/what-is-the-risc-v-ecosystem/19227/>
- <https://www.programmersought.com/article/14557389405/>
- ...
-



Source: "Linux on RISC-V", D. Fustimi, ELC2020

Industry innovation on RISC-V

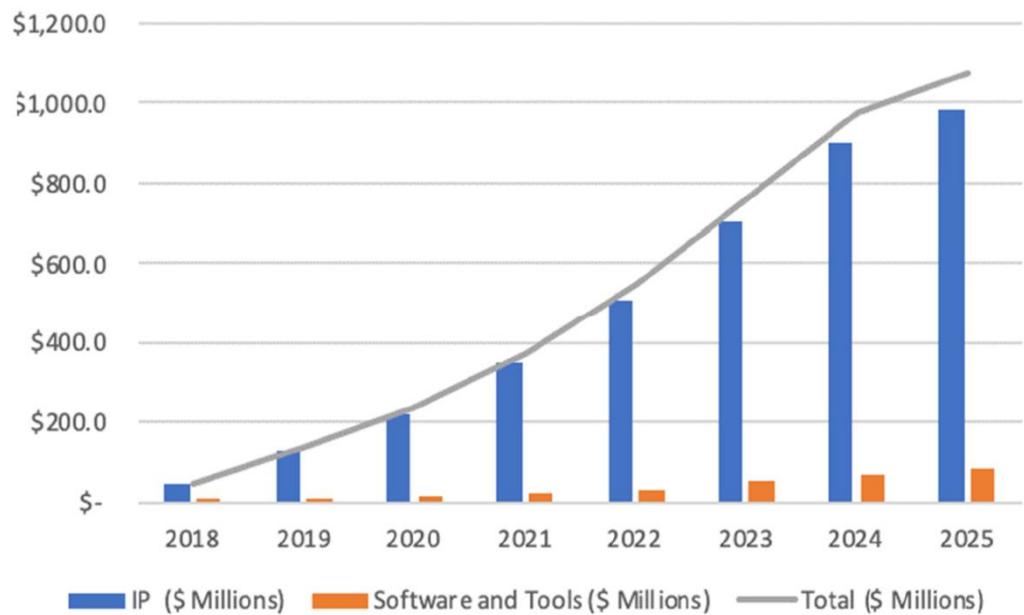


Wilson Research Group/Siemens found that **23% of ASIC and FPGA projects incorporated RISC-V** in at least one processor in a double-blind 2020 study.

Source: “RISC-V: The Open Era of Computing”, Calista Redmond, The Linux Foundation Spring Member Meeting 2021

RISC-V IP, SW, and Tools build momentum

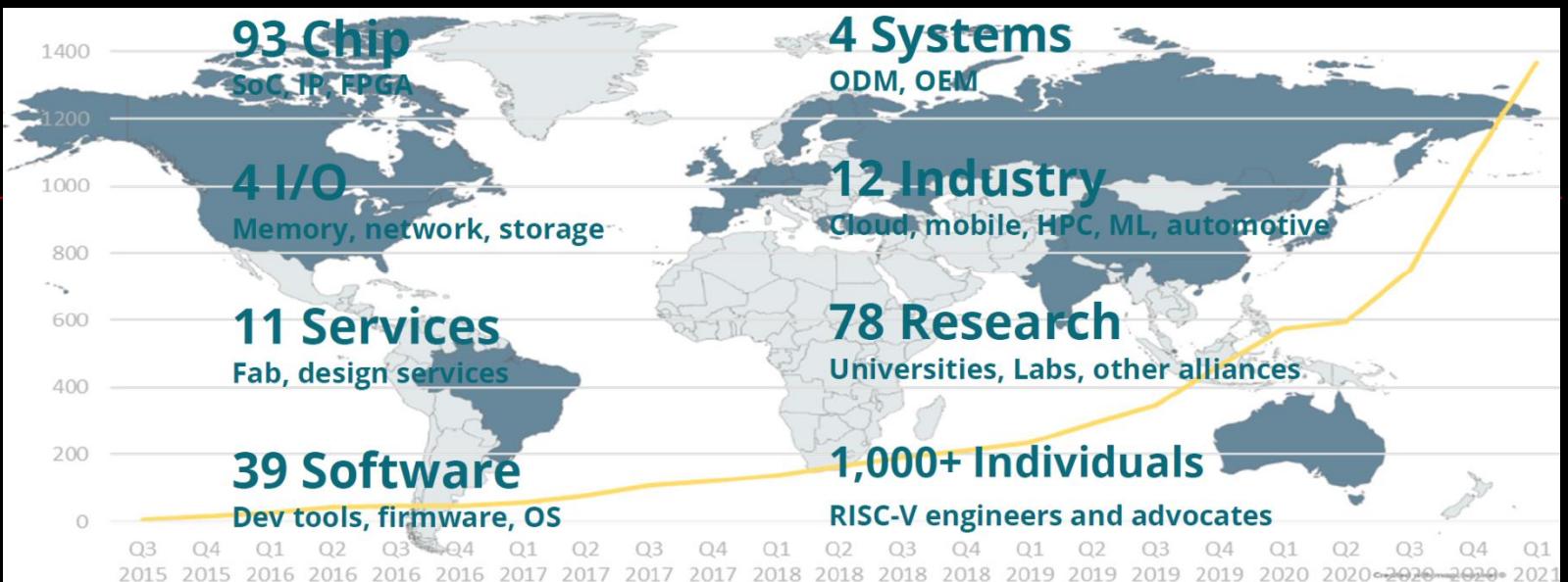
The total market for RISC-V IP and Software is expected to grow from to \$1.07 billion by 2025 at a CAGR of 54.1%



Source: "RISC-V: The Open Era of Computing", Calista Redmond, The Linux Foundation Spring Member Meeting 2021

Source: Tractica

Nearly 1,400 RISC-V Members(across 70 Countries)



In 2020, RISC-V membership grew 133%

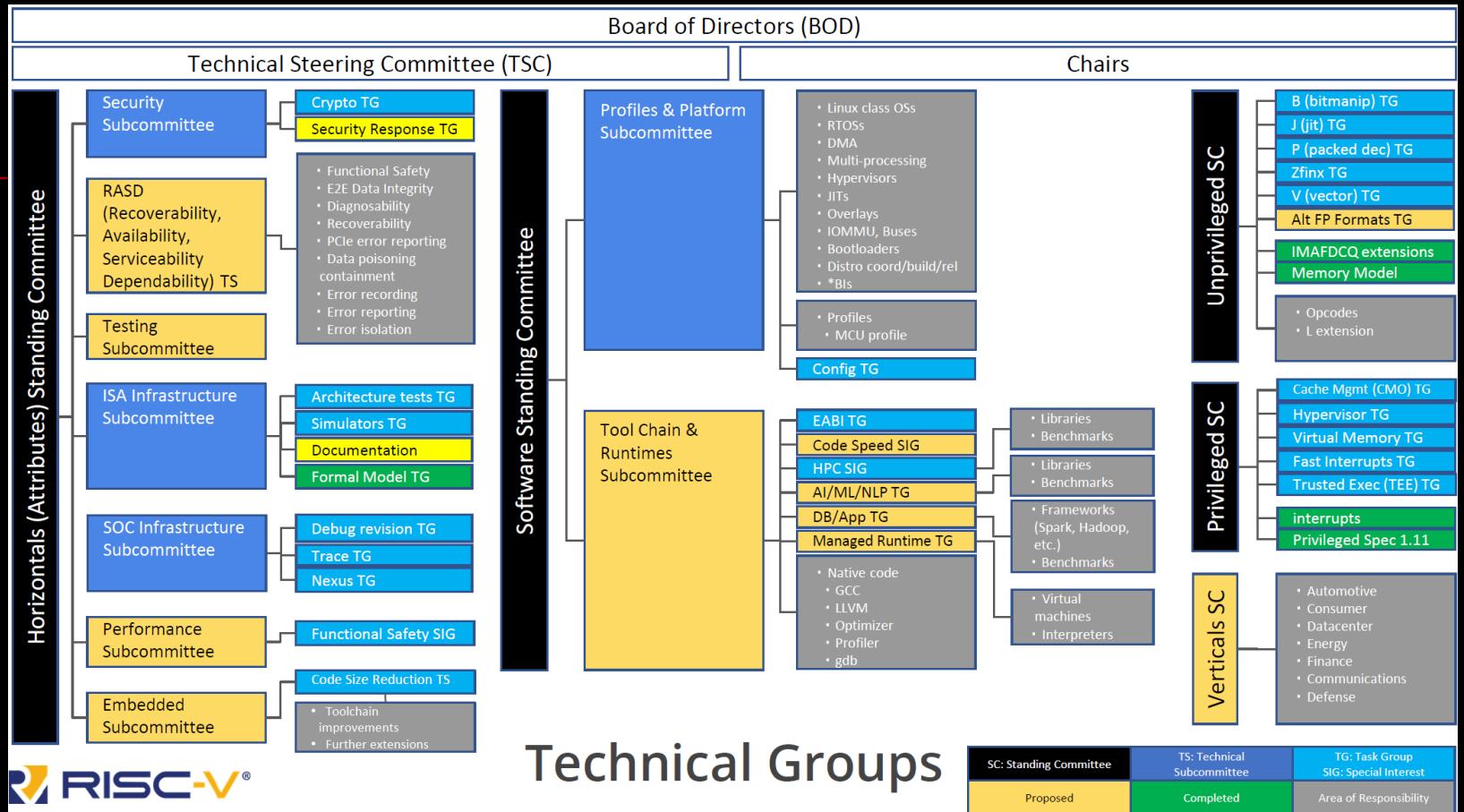
Source: “RISC-V: The Open Era of Computing”, Calista Redmond, The Linux Foundation Spring Member Meeting 2021

Incredible industry progress

- The European Processor Initiative finalized the first version of its **RISC-V accelerator architecture** and will deliver test chip in 2021.
- The RIOS Lab announced PicoRio, an affordable **RISC-V open source small-board computer** available in 2021.
- Imperas announced first **RISC-V verification reference model with UVM encapsulation**.
- Seagate announced **hard disk drive controller** with high-performance RISC-V CPU.
- GreenWaves **ultra-low power GAP9 hearables platform** enabling scene-aware and neural network-based noise reduction.
- Alibaba unveiled RV64GCV core in its Xuantie 910 processor for **cloud and edge servers**.
- Microchip released the first **SoC FPGA development kit** based on the RISC-V ISA.
- Andes released **superscalar multicore and L2 cache controller** processors.
- StarFive released the world's first **RISC-V AI visual processing platform**
- SiFive unveiled world's fastest development board for **RISC-V Personal Computers**.
- Micro Magic announced an incredibly **fast 64-bit RISC-V core** achieving 5GHz and 13,000 CoreMarks at 1.1V.

Source: “RISC-V: The Open Era of Computing”, Calista Redmond, The Linux Foundation Spring Member Meeting 2021

Organizational Structures



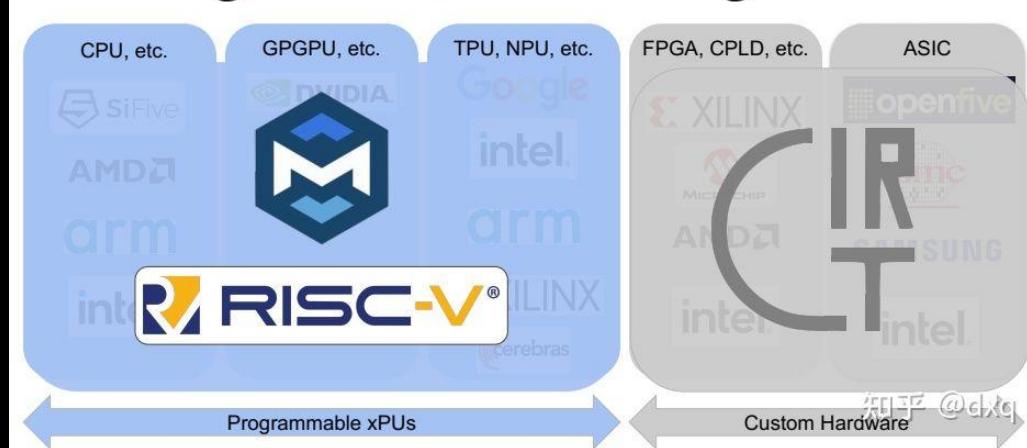
Source: “RISC-V: The Open Era of Computing”, Calista Redmond, The Linux Foundation Spring Member Meeting 2021

The Golden Age of Compiler Design in an Era of HW/SW Co-design

-



Co-design of HW and SW design



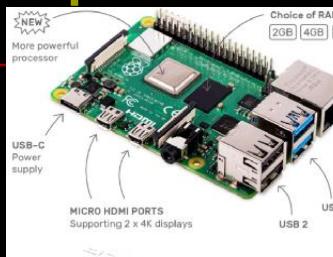
-

<https://zhuanlan.zhihu.com/p/367035973> //看Chris Lattner在ASPLOS演讲有感

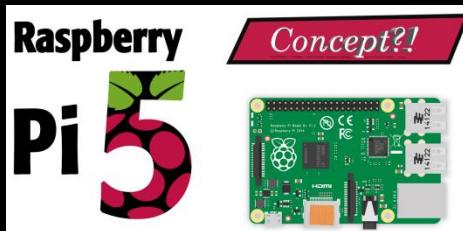
4) Testbed

4.1 Raspberry Pi

- <https://www.raspberrypi.org/products/raspberry-pi-4-model-b/>



Features/Specs	Raspberry Pi 4B	Raspberry Pi 3 B+
Release date	24th June 2019	14th March 2018
SoC	Broadcom BCM2711 quad-core Cortex-A72 @ 1.5 GHz	Broadcom BCM2837B0 quad-core Cortex-A53 @ 1.4 GHz
GPU	VideoCore VI with OpenGL ES 11, 2.0, 3.0	VideoCore IV with OpenGL ES 11, 2.0
Video Decode	H.265 4Kp60, H.264 1080p60	H.264 & MPEG-4 1080p30
Video Encode	H.264 1080p30	
Memory	1GB, 2GB, or 4GB LPDDR4	1GB LPDDR2
Storage	microSD card	
Video & Audio Output	2x micro HDMI ports up to 4Kp60 3.5mm AV port (composite + audio) MIPI DSI connector	1x HDMI 1.4 port up to 1080p60 3.5mm AV port (composite + audio) MIPI DSI connector
Camera	MIPI CSI connector	
Ethernet	Native Gigabit Ethernet	Gigabit Ethernet (300 Mbps max.)
WiFi	Dual band 802.11 b/g/n/ac	
Bluetooth	Bluetooth 5.0 + BLE	Bluetooth 4.2 + BLE
USB	2x USB 3.0 + 2x USB 2.0	4x USB 2.0
Expansion	40-pin GPIO header	
Power Supply	5V via USB type-C up to 3A 5V via GPIO header up to 3A Power over Ethernet via PoE HAT	5V via micro USB up to 2.5A 5V via GPIO header up to 3A Power over Ethernet via PoE HAT
Dimensions	85x56 mm	
Default OS	Raspbian (after June 24, 2019)	Raspbian (after March 2018)
Price	\$35 (1GB RAM), \$45 (2GB RAM), \$55 (4GB RAM)	\$35 (1GB RAM)



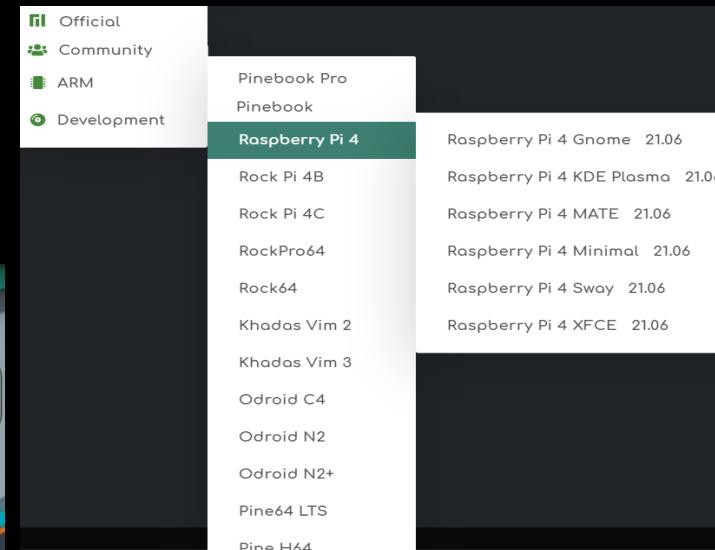
Fedora

- A Linux distribution developed by the community-supported Fedora Project which is sponsored primarily by Red Hat
- [https://en.wikipedia.org/wiki/Fedora_\(operating_system\)](https://en.wikipedia.org/wiki/Fedora_(operating_system))
- <https://getfedora.org/>
- <https://alt.fedoraproject.org/alt/>
- <https://spins.fedoraproject.org/>
- <https://fedoraproject.org/wiki/Architectures/ARM>
- <https://fedoramagazine.org/>
- <https://silverblue.fedoraproject.org/>
- Developer friendly!

Manjaro

- an open-source Linux distribution based on the Arch Linux operating system
- https://en.wikipedia.org/wiki/Manjaro_Linux
- <https://distrowatch.com/>

Page Hit Ranking		
Rank	Distribution	HPD*
1	MX Linux	3369▼
2	Manjaro	2490▼
3	Mint	2172▼
4	Pop!_OS	1967▼
5	EndeavourOS	1643▲
6	Ubuntu	1378▼
7	Debian	1281▬
8	elementary	1149▼
9	Fedora	1018▼
10	openSUSE	890▲

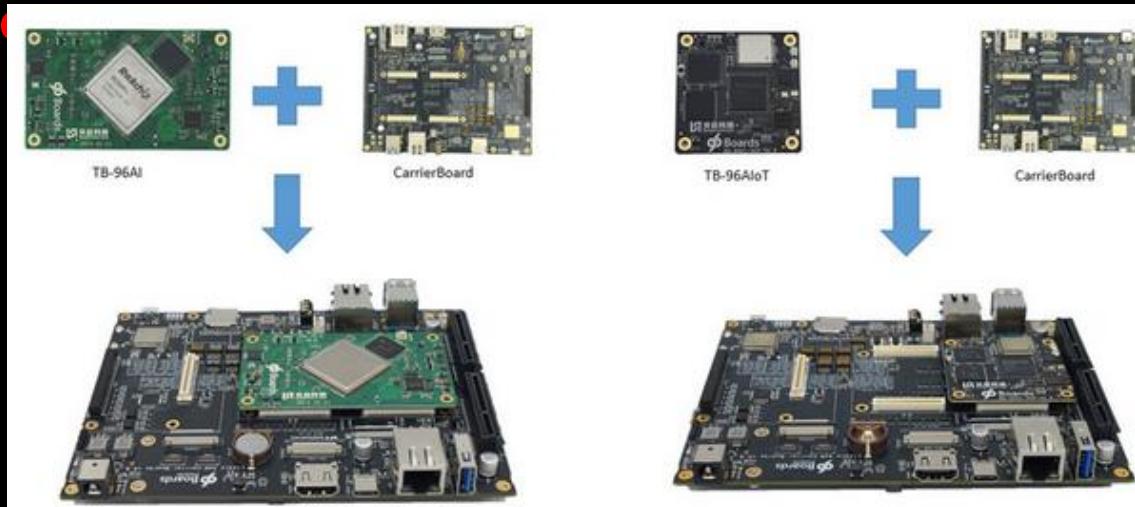


- <https://manjaro.org>
-  Manjaro 21.0 Ornara | XFCE
- More and more developer friendly!

4.2 Modularization

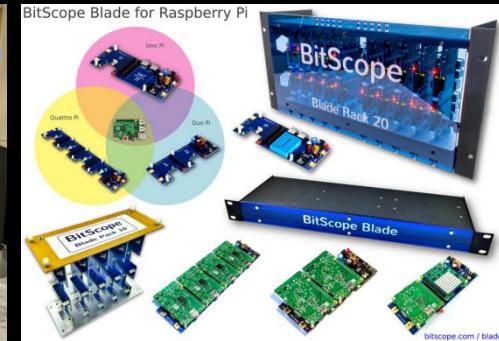
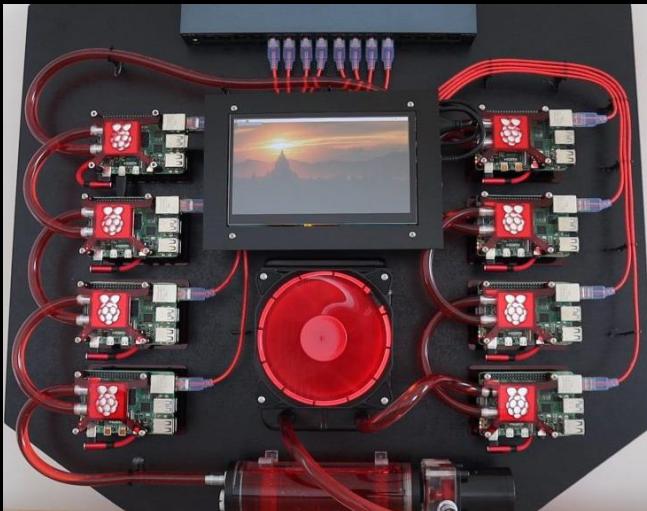
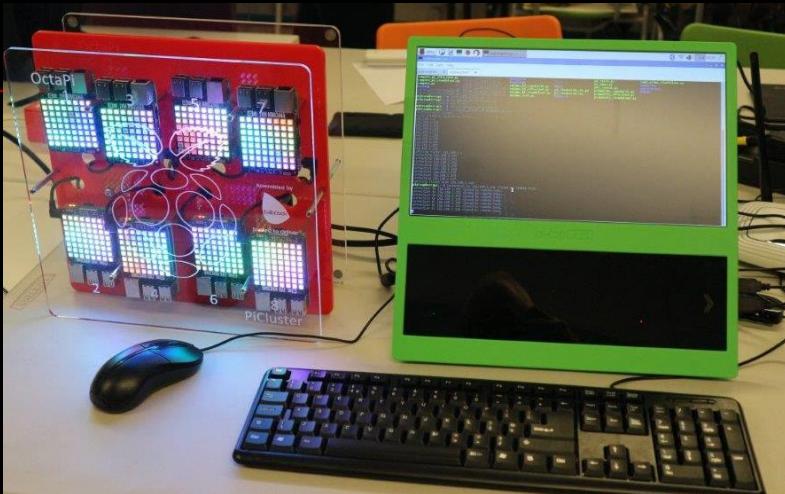
SOM/COM

- <https://www.linaro.org/news/linaro-announces-launch-of-96boards-system-on-module-som-specification/>
- <http://linuxgizmos.com/linaro-launches-two-96boards-som-specifications/>
- <http://static.linaro.org/assets/specifications/96BoardsComputeSoMSpecificationV1.0.pdf>
- <https://static.linaro.org/assets/specifications/96BoardsWirelessSoMSpecificationV1.0.pdf>
-

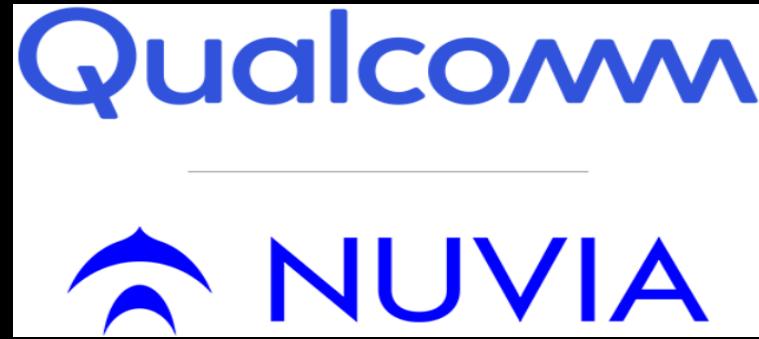
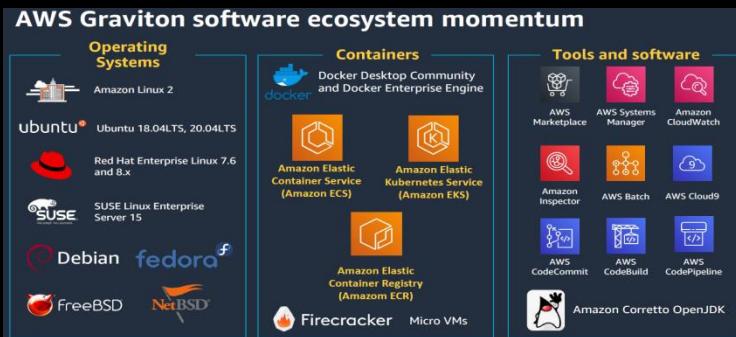
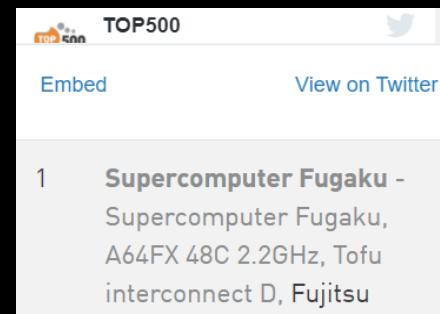
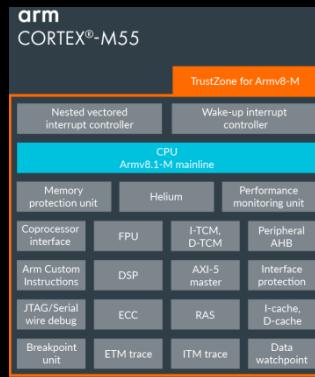
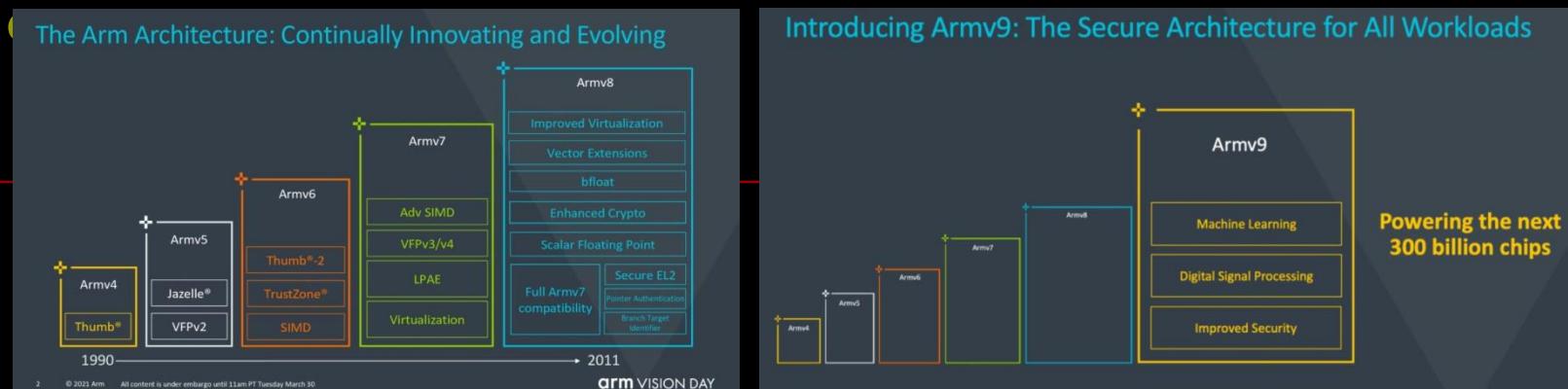


Clustering

- <https://www.raspberrypi.org/blog/supercomputing-with-raspberry-pi-hackspace-41/>



4.3 ARM Ecosystem in 2021



II. Chisel/SpinalHDL on ARM

1) Introduction

1.1 Chisel

- **<https://www.chisel-lang.org/>**

Chisel is a hardware design language that facilitates advanced circuit generation and design reuse for both ASIC and FPGA digital logic designs. Chisel adds hardware construction primitives to the Scala programming language, providing designers with the power of a modern programming language to write complex, parameterizable circuit generators that produce synthesizable Verilog. This generator methodology enables the creation of re-usable components and libraries, such as the FIFO queue and arbiters in the Chisel Standard Library, raising the level of abstraction in design while retaining fine-grained control.

For more information on the benefits of Chisel see: "What benefits does Chisel offer over classic Hardware Description Languages?"

Chisel is powered by FIRRTL (Flexible Intermediate Representation for RTL), a hardware compiler framework that performs optimizations of Chisel-generated circuits and supports custom user-defined circuit transformations.

- **<https://github.com/chipsalliance/chisel3>**

Languages

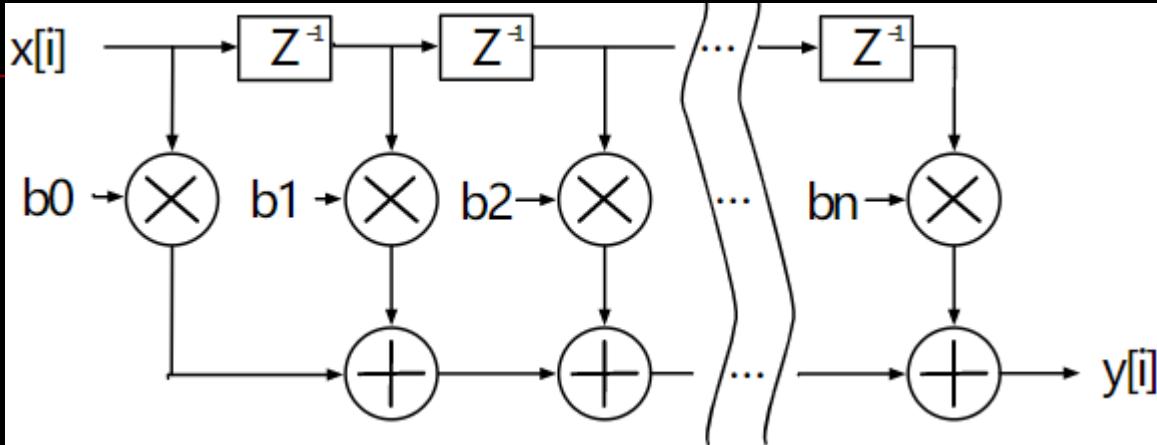


- **<https://github.com/topics/chisel>**

- **<http://www.imm.dtu.dk/~masca/chisel-book.html>**

Examples

- Consider an FIR filter that implements a convolution operation, as depicted in this block diagram:



While Chisel provides similar base primitives as synthesizable Verilog, and *could* be used as such:

```
// 3-point moving average implemented in the style of a FIR filter
class MovingAverage3(bitWidth: Int) extends Module {
    val io = IO(new Bundle {
        val in = Input(UInt(bitWidth.W))
        val out = Output(UInt(bitWidth.W))
    })

    val z1 = RegNext(io.in)
    val z2 = RegNext(z1)

    io.out := (io.in * 1.U) + (z1 * 1.U) + (z2 * 1.U)
}
```

the power of Chisel comes from the ability to create generators, such as n FIR filter that is defined by the list of coefficients:

```
// Generalized FIR filter parameterized by the convolution coefficients
class FirFilter(bitWidth: Int, coeffs: Seq[UInt]) extends Module {
    val io = IO(new Bundle {
        val in = Input(UInt(bitWidth.W))
        val out = Output(UInt(bitWidth.W))
    })
    // Create the serial-in, parallel-out shift register
    val zs = Reg(Vec(coeffs.length, UInt(bitWidth.W)))
    zs(0) := io.in
    for (i <- 1 until coeffs.length) {
        zs(i) := zs(i-1)
    }

    // Do the multiplies
    val products = VecInit.tabulate(coeffs.length)(i => zs(i) * coeffs(i))

    // Sum up the products
    io.out := products.reduce(_ + _)
}
```

and use and re-use them across designs:

```
val movingAverage3Filter = Module(new FirFilter(8, Seq(1.U, 1.U, 1.U))) // same 3-point moving average filter
val delayFilter = Module(new FirFilter(8, Seq(0.U, 1.U))) // 1-cycle delay as a FIR filter
val triangleFilter = Module(new FirFilter(8, Seq(1.U, 2.U, 3.U, 2.U, 1.U))) // 5-point FIR filter with a tria
```

The above can be converted to Verilog using `ChiselStage`:

```
import chisel3.stage.{ChiselStage, ChiselGeneratorAnnotation}

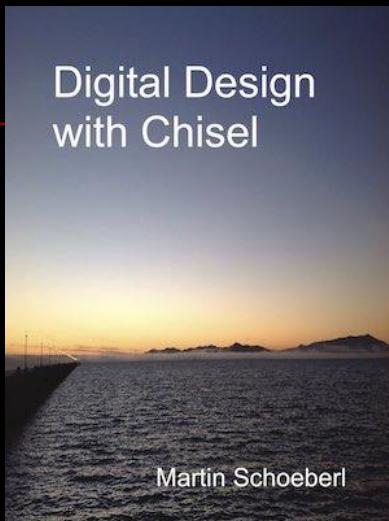
(new chisel3.stage.ChiselStage).execute(
    Array("-X", "verilog"),
    Seq(ChiselGeneratorAnnotation(() => new FirFilter(8, Seq(1.U, 1.U, 1.U)))))
```

Alternatively, you may generate some Verilog directly for inspection:

```
val verilogString = (new chisel3.stage.ChiselStage).emitVerilog(new FirFilter(8, Seq(0.U, 1.U)))
println(verilogString)
```

Chisel Books

- <https://github.com/schoeberl/chisel-book>



1.2 SpinalHDL

■ <https://github.com/SpinalHDL/SpinalHDL>

- A language to describe digital hardware
- Compatible with EDA tools, as it generates VHDL/Verilog files
- Much more powerful than VHDL, Verilog, and SystemVerilog in its syntax and features
- Much less verbose than VHDL, Verilog, and SystemVerilog
- Not an HLS, nor based on the event-driven paradigm
- Only generates what you asked it in a one-to-one way (no black-magic, no black box)
- Not introducing area/performance overheads in your design (versus a hand-written VHDL/Verilog design)
- Based on the RTL description paradigm, but can go much further
- Allowing you to use Object-Oriented Programming and Functional Programming to elaborate your hardware and verify it
- Free and can be used in the industry without any license

Languages



Scala 71.0%	Verilog 14.4%
Python 5.3%	VHDL 3.3%
SystemVerilog 2.8%	C++ 1.4%
Other 1.8%	

■ <https://spinalhdl.github.io/SpinalDoc-RTD/>



2) Scala on ARM

2.1 Scala

- [https://en.wikipedia.org/wiki/Scala_\(programming_language\)](https://en.wikipedia.org/wiki/Scala_(programming_language))

Scala (/skəˈla:/ SKAH-lah)^[8] is a strong statically typed general-purpose programming language which supports both object-oriented programming and functional programming. Designed to be concise,^[9] many of Scala's design decisions are aimed to address criticisms of Java.^[7]

Scala source code can be compiled to Java bytecode and run on a Java virtual machine (JVM). Scala provides language interoperability with Java so that libraries written in either language may be referenced directly in Scala or Java code.^[10] Like Java, Scala is object-oriented, and uses a syntax termed *curly-brace* which is similar to the language C. Since Scala 3, there is also an option to use the *off-side rule* (indenting) to structure blocks, and its use is advised. Martin Odersky has said that this turned out to be the most productive change introduced in Scala 3.^[11]

Unlike Java, Scala has many features of functional programming languages like Scheme, Standard ML, and Haskell, including currying, immutability, lazy evaluation, and pattern matching. It also has an advanced type system supporting algebraic data types, covariance and contravariance, higher-order types (but not higher-rank types), and anonymous types. Other features of Scala not present in Java include operator overloading, optional parameters, named parameters, and raw strings. Conversely, a feature of Java not in Scala is checked exceptions, which has proved controversial.^[12]

The name Scala is a blend of *scalable* and *language*, signifying that it is designed to grow with the demands of its users.^[13]

- <https://www.scala-lang.org>
- <https://www.lightbend.com/>
- <http://scalacookbook.com/>
- <https://docs.scala-lang.org/getting-started/index.html>
- <https://ammonite.io/>
- ...

JDK

- <https://docs.scala-lang.org/overviews/jdk-compatibility/overview.html>
Version compatibility table

JDK version	Minimum Scala versions	Recommended Scala versions
17	2.13.6, 2.12.15 (forthcoming)	2.13.6, 2.12.15 (forthcoming)
16	2.13.5, 2.12.14	2.13.6, 2.12.14
13, 14, 15	2.13.2, 2.12.11	2.13.6, 2.12.14
12	2.13.1, 2.12.9	2.13.6, 2.12.14
11	2.13.0, 2.12.4, 2.11.12	2.13.6, 2.12.14, 2.11.12
8	2.13.0, 2.12.0, 2.11.0, 2.10.2	2.13.6, 2.12.14, 2.11.12, 2.10.7
6, 7	2.11.0, 2.10.0	2.11.12, 2.10.7

GraalVM Native Image compatibility notes

There are several records of successfully using Scala with GraalVM Native Image(i.e.: ahead of time compiler) to produce directly executable binaries. Beware that, even using solely the Scala standard library, Native Image compilation have some heavy requirements in terms of [reflective access](#), and it very likely require additional configuration steps to be performed.

A few sbt plugins are offering support for GraalVM Native Image compilation:

- [sbt-native-packager](#)
- [sbt-native-image](#)

Scala 3

The Scala 3.x series supports JDK 8, as well as 11 and beyond.

As Scala and the JVM continue to evolve, some eventual Scala version may drop support for JDK 8, in order to better take advantage of new JVM features. It isn't clear yet what the new minimum supported version might become.

Installation

- <https://www.scala-lang.org/download/scala2.html>

Archive	System	Size
scala-2.13.6.tgz	Mac OS X, Unix, Cygwin	22.32M
scala-2.13.6.msi	Windows (msi installer)	131.46M
scala-2.13.6.zip	Windows	22.36M
scala-2.13.6.deb	Debian	646.94M
scala-2.13.6.rpm	RPM package	131.73M
scala-docs-2.13.6.txz	API docs	58.10M
scala-docs-2.13.6.zip	API docs	112.56M
scala-sources-2.13.6.tar.gz	Sources	

- install on RPi4

```
[mydev@fedora v2.13.6]$ sudo rpm -ivh scala-2.13.6.rpm
[sudo] password for mydev:
Verifying... ################################################ [100%]
Preparing... ################################################ [100%]
Updating / installing...
 1:scala-2.13.6-1 ################################################ [100%]
[mydev@fedora v2.13.6]$
[mydev@fedora v2.13.6]$
[mydev@fedora v2.13.6]$ which scala
/usr/bin/scala
[mydev@fedora v2.13.6]$ which scalac
/usr/bin/scalac
[mydev@fedora v2.13.6]$ 
```

2.2 Scala Native

- <https://scala-native.readthedocs.io/en/latest/>
An optimizing ahead-of-time compiler and lightweight managed runtime designed specifically for Scala.
- <https://github.com/scala-native/>
- **Features:**

- Low-level primitives.

```
type Vec = CStruct3[Double, Double, Double]

val vec = stackalloc[Vec] // allocate c struct on stack
vec._1 = 10.0           // initialize fields
vec._2 = 20.0
vec._3 = 30.0
length(vec)             // pass by reference
```

Pointers, structs, you name it. Low-level primitives let you hand-tune your application to make it work exactly as you want it to. You're in control.

- Seamless interop with native code.

```
@extern object stdlib {
  def malloc(size: CSize): Ptr[Byte] = extern
}

val ptr = stdlib.malloc(32)
```

Calling C code has never been easier. With the help of extern objects you can seamlessly call native code without any runtime overhead.

- Instant startup time.

```
> time hello-native
hello, native!

real    0m0.005s
user    0m0.002s
sys     0m0.002s
```

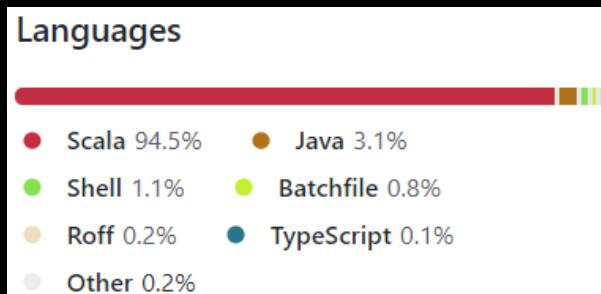
Scala Native is compiled ahead-of-time via [LLVM](#). This means that there is no sluggish warm-up phase that's common for just-in-time compilers. Your code is immediately fast and ready for action.

- <https://contributors.scala-lang.org/t/scala-native-next-steps/4216>

2.3 Build System

SBT

- <https://www.scala-sbt.org>
- **Define your tasks in Scala. Run them in parallel from SBT's interactive shell.**
- <https://github.com/sbt/sbt>



- install on RPi4

<https://www.scala-sbt.org/1.x/docs/Installing-sbt-on-Linux.html>

On Fedora (31 and above), use `sbt-rpm.repo` :

```
# remove old Bintray repo file
sudo rm -f /etc/yum.repos.d/bintray-rpm.repo
curl -L https://www.scala-sbt.org/sbt-rpm.repo > sbt-rpm.repo
sudo mv sbt-rpm.repo /etc/yum.repos.d/
sudo dnf install sbt
```

- **Tetralogy**
 - sbt compile**
 - sbt test**
 - sbt assembly**
 - sbt publishLocal**
-

Mill

- https://com-lihaoyi.github.io/mill/mill/Intro_to_Mill.html

[Mill](#) is your shiny new Java/Scala build tool! Scared of SBT? Melancholy over Maven?
Grumbling about Gradle? Baffled by Bazel? Give Mill a try!

-
- <https://github.com/com-lihaoyi/mill/>
 - <https://www.handsonscala.com/>

3) Chisel/SpinalHDL on RPi4

3.1 Env Setup

Fedora

```
[mydev@fedora ~]$ uname -a
Linux fedora 5.12.13-300.fc34.aarch64 #1 SMP Wed Jun 23 16:03:11 UTC 2021 aarch64 aarch64 aarch64 GNU/Linux

[mydev@fedora ~]$ java --version
openjdk 11.0.11 2021-04-20
OpenJDK Runtime Environment 18.9 (build 11.0.11+9)
OpenJDK 64-Bit Server VM 18.9 (build 11.0.11+9, mixed mode, sharing)
[mydev@fedora ~]$
[mydev@fedora ~]$ scala --version
Scala code runner version 2.13.6 -- Copyright 2002-2021, LAMP/EPFL and Lightbend, Inc.
[mydev@fedora ~]$
[mydev@fedora ~]$ sbt --version
sbt version in this project: 1.5.4
sbt script version: 1.5.4
[mydev@fedora ~]$
[mydev@fedora ~]$ gcc -v
Using built-in specs.
COLLECT_GCC=/usr/bin/gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/aarch64-redhat-linux/11/lto-wrapper
Target: aarch64-redhat-linux
Configured with: ../configure --enable-bootstrap --enable-languages=c,c++,fortran,objc,obj-c++,ada,go,lto --prefix=/usr --mandir=/usr/share/man --infodir=/usr/share/info --with-bugurl=http://bugzilla.redhat.com/bugzilla --enable-shared --enable-threads=posix --enable-checking=release --enable-multilib --with-system-zlib --enable-cxa_atexit --disable-libumwind-exceptions --enable-gnu-unique-object --enable-linker-build-id --with-gcc-major-version-only --with-linker-hash-style=gnu --enable-plugin --enable-initfini-array --with-isl=/builddir/build/BUILD/gcc-11.1.1-20210531/obj-aarch64-redhat-linux/isl-install --enable-gnu-indirect-function --build=aarch64-redhat-linux
Thread model: posix
Supported LTO compression algorithms: zlib zstd
gcc version 11.1.1 20210531 (Red Hat 11.1.1-3) (GCC)
[mydev@fedora ~]$ ■
```

RPi4 8GB RAM:

```
[mydev@fedora ~]$ free -m
              total        used        free      shared  buff/cache   available
Mem:       7836        6919         132          3        784        766
Swap:      7835         865       6970
[mydev@fedora ~]$ ■
```

```
[mydev@fedora /]$ swapon -s
Filename                                Type  Size  Used  Priority
/dev/zram0                               partition 8024060 0 100
[mydev@fedora ~]$ ■
```

- **sudo dnf groupinstall "Electronic Lab"**
- **sudo dnf install boost boost-devel verilator yosys z3 z3-devel z3-libs java-z3 python-z3 cocotb**
- ...

Toolchain

- <https://github.com/riscv/riscv-gnu-toolchain>
for bare metal(make -j\$(nproc)):

```
[mydev@fedora ~]$ tree -L 2 -C /opt/MyWorkSpace/DevSW/Toolchain/RISC-V/GCC/Official/ELF/  
/opt/MyWorkSpace/DevSW/Toolchain/RISC-V/GCC/Official/ELF/  
+-- bin  
|   |-- riscv64-unknown-elf-addr2line  
|   |-- riscv64-unknown-elf-ar  
|   |-- riscv64-unknown-elf-as  
|   |-- riscv64-unknown-elf-c++  
|   |-- riscv64-unknown-elf-c++filt  
|   |-- riscv64-unknown-elf-cpp  
|   |-- riscv64-unknown-elf-elfedit  
|   |-- riscv64-unknown-elf-g++  
|   |-- riscv64-unknown-elf-gcc  
|   |-- riscv64-unknown-elf-gcc-10.2.0  
|   |-- riscv64-unknown-elf-gcc-ar  
|   |-- riscv64-unknown-elf-gcc-nm  
|   |-- riscv64-unknown-elf-gcc-ranlib  
|   |-- riscv64-unknown-elf-gcov  
|   |-- riscv64-unknown-elf-gcov-dump  
|   |-- riscv64-unknown-elf-gcov-tool  
|   |-- riscv64-unknown-elf-gdb  
|   |-- riscv64-unknown-elf-gdb-add-index  
|   |-- riscv64-unknown-elf-gprof  
|   |-- riscv64-unknown-elf-ld  
|   |-- riscv64-unknown-elf-ld.bfd  
|   |-- riscv64-unknown-elf-lto-dump  
|   |-- riscv64-unknown-elf-nm  
|   |-- riscv64-unknown-elf-objcopy  
|   |-- riscv64-unknown-elf-objdump  
|   |-- riscv64-unknown-elf-ranlib  
|   |-- riscv64-unknown-elf-readelf  
|   |-- riscv64-unknown-elf-run  
|   |-- riscv64-unknown-elf-size  
|   |-- riscv64-unknown-elf-strings  
|   |-- riscv64-unknown-elf-strip  
+-- include  
    |-- gdb  
+-- lib  
    |-- bfd-plugins  
    |-- gcc  
    |-- libcc1.a  
    |-- libcc1.so -> libcc1.so.0.0.0  
    |-- libcc1.so.0 -> libcc1.so.0.0.0  
    |-- libcc1.so.0.0.0  
    |-- libriscv64-unknown-elf-sim.a  
+-- libexec  
    |-- gcc  
+-- riscv64-unknown-elf  
    |-- bin  
    |-- include  
    |-- lib  
+-- share  
    |-- gcc-10.2.0  
    |-- gdb  
    |-- info  
    |-- locale  
    |-- man
```

directly built on RPi4 against “riscv64-unknown-elf-” for ~2.5h(8G RAM +)



for Linux(make linux -j\$(nproc)):

```
[mydev@fedora ~]$ tree -L 2 -C /opt/MyWorkSpace/DevSW/Toolchain/RISC-V/GCC/Official/Linux
/opt/MyWorkSpace/DevSW/Toolchain/RISC-V/GCC/Official/Linux
├── bin
│   ├── riscv64-unknown-linux-gnu-addr2line
│   ├── riscv64-unknown-linux-gnu-ar
│   ├── riscv64-unknown-linux-gnu-as
│   ├── riscv64-unknown-linux-gnu-c++
│   ├── riscv64-unknown-linux-gnu-c++filt
│   ├── riscv64-unknown-linux-gnu-cpp
│   ├── riscv64-unknown-linux-gnu-elfedit
│   ├── riscv64-unknown-linux-gnu-g++
│   ├── riscv64-unknown-linux-gnu-gcc
│   ├── riscv64-unknown-linux-gnu-gcc-10.2.0
│   ├── riscv64-unknown-linux-gnu-gcc-ar
│   ├── riscv64-unknown-linux-gnu-gcc-nm
│   ├── riscv64-unknown-linux-gnu-gcc-ranlib
│   ├── riscv64-unknown-linux-gnu-gcov
│   ├── riscv64-unknown-linux-gnu-gcov-dump
│   ├── riscv64-unknown-linux-gnu-gcov-tool
│   ├── riscv64-unknown-linux-gnu-gdb
│   ├── riscv64-unknown-linux-gnu-gdb-add-index
│   ├── riscv64-unknown-linux-gnu-gfortran
│   ├── riscv64-unknown-linux-gnu-gprof
│   ├── riscv64-unknown-linux-gnu-ld
│   ├── riscv64-unknown-linux-gnu-ld.bfd
│   ├── riscv64-unknown-linux-gnu-lto-dump
│   ├── riscv64-unknown-linux-gnu-nm
│   ├── riscv64-unknown-linux-gnu-objcopy
│   ├── riscv64-unknown-linux-gnu-objdump
│   ├── riscv64-unknown-linux-gnu-ranlib
│   ├── riscv64-unknown-linux-gnu-readelf
│   ├── riscv64-unknown-linux-gnu-run
│   ├── riscv64-unknown-linux-gnu-size
│   ├── riscv64-unknown-linux-gnu-strings
│   └── riscv64-unknown-linux-gnu-strip
├── include
└── lib
    ├── bfd-plugins
    │   └── gcc
    │       └── libriscv64-unknown-linux-gnu-sim.a
    ├── libexec
    │   └── gcc
    ├── riscv64-unknown-linux-gnu
    │   ├── bin
    │   ├── include
    │   ├── lib
    │   ├── lib32
    │   └── lib64
    ├── share
    │   ├── gcc-10.2.0
    │   ├── gdb
    │   ├── info
    │   └── man
    └── sysroot
        ├── etc
        ├── lib
        ├── lib32
        ├── lib64
        ├── sbin
        └── usr
            └── var
```

directly built on RPi4 against “riscv64-unknown-elf-” for ~5h(8G RAM +)



Rocket Tools

<https://github.com/chipsalliance/rocket-tools>

My current patch(just an ugly workaround, repreparing a better one):

```
[mydev@MyRPi4-Fedora-1 rocket-tools-master]$ git diff
diff --git a/build-rv32ima.sh b/build-rv32ima.sh
index 1f9a363..21398b2 100755
--- a/build-rv32ima.sh
+++ b/build-rv32ima.sh
@@ -10,6 +10,6 @@ echo "Starting RISC-V Toolchain build process"
 build_project riscv-isa-sim --prefix=$RISCV --with-isa=rv32ima
 build_project riscv-gnu-toolchain --prefix=$RISCV --with-arch=rv32ima --with-abi=ilp32
 CC= CXX= build_project riscv-pk --prefix=$RISCV --host=riscv32-unknown-elf
-build_project riscv-openocd --prefix=$RISCV --enable-remote-bitbang --disable-werror
+#build_project riscv-openocd --prefix=$RISCV --enable-remote-bitbang --disable-werror

 echo -e "\nRISC-V Toolchain installation completed!"
diff --git a/build.sh b/build.sh
index b9e9306..8d7e14a 100755
--- a/build.sh
+++ b/build.sh
@@ -16,7 +16,7 @@ check_version() {

 check_version automake 1.14 "OpenOCD build"
 check_version autoconf 2.64 "OpenOCD build"
-build_project riscv-openocd --prefix=$RISCV --enable-remote-bitbang --enable-jtag_vpi --disable-werror
+#build_project riscv-openocd --prefix=$RISCV --enable-remote-bitbang --enable-jtag_vpi --disable-werror
 build_project riscv-isa-sim --prefix=$RISCV
```

```
[mydev@MyRPi4-Fedora-1 rocket-tools-master]$ cd fsf-binutils-gdb
[mydev@MyRPi4-Fedora-1 fsf-binutils-gdb]$ git diff
diff --git a/bfd/development.sh b/bfd/development.sh
index bcce82f5779..47547ffcc05 100644
--- a/bfd/development.sh
+++ b/bfd/development.sh
@@ -16,7 +16,7 @@

 # along with this program. If not, see <http://www.gnu.org/licenses/>.

 # Controls whether to enable development-mode features by default.
-development=true
+development=false
```

```
[mydev@MyRPi4-Fedora-1 rocket-tools-master]$ cd riscv-gnu-toolchain/riscv-glibc
[mydev@MyRPi4-Fedora-1 riscv-glibc]$ git diff
diff --git a/scripts/config.guess b/scripts/config.guess
index bbd48b50e8..eabb39ff97 100755
--- a/scripts/config.guess
+++ b/scripts/config.guess
@@ -156,7 +156,7 @@ esac
# Note: order is significant - the case branches are not exclusive.

case "${UNAME_MACHINE}:${UNAME_SYSTEM}:${UNAME_RELEASE}:${UNAME_VERSION}" in
- *:NetBSD:*:*)
+ *:Linux:*:*)
        # NetBSD (nbsd) targets should (where applicable) match one or
        # more of the tuples: **-netbsdelf*, **-netbsdaut*, ...
        # **-netbsdecoff* and **-netbsd*. For targets that recently
```

```
[mydev@MyRPi4-Fedora-1 riscv-isa-sim]$ git diff
diff --git a/fesvr/dtm.cc b/fesvr/dtm.cc
index 993011d..96a67ba 100644
--- a/fesvr/dtm.cc
+++ b/fesvr/dtm.cc
@@ -6,6 +6,7 @@
 #include <string.h>
 #include <assert.h>
 #include <pthread.h>
+#include <stdexcept>

#define RV_X(x, s, n) \
(((x) >> (s)) & ((1 << (n)) - 1))
diff --git a/riscv/devices.h b/riscv/devices.h
index 1bc9618..3dd6c66 100644
--- a/riscv/devices.h
+++ b/riscv/devices.h
@@ -7,6 +7,7 @@
 #include <string>
 #include <map>
 #include <vector>
+#include <stdexcept>

class processor_t;

diff --git a/scripts/config.guess b/scripts/config.guess
index f32079a..00be375 100755
--- a/scripts/config.guess
+++ b/scripts/config.guess
@@ -142,7 +142,7 @@ UNAME_VERSION=`(uname -v) 2>/dev/null` || UNAME_VERSION=unknown
# Note: order is significant - the case branches are not exclusive.

case "${UNAME_MACHINE}:${UNAME_SYSTEM}:${UNAME_RELEASE}:${UNAME_VERSION}" in
-  *:NetBSD:*:*)
+  *:Linux:*:*)
      # NetBSD (nbsd) targets should (where applicable) match one or
      # more of the tuples: **-netbsdelf*, **-netbsdaout*,
      # **-netbsdecoff* and **-netbsd*. For targets that recently
[mydev@MyRPi4-Fedora-1 riscv-isa-sim]$ 
```

```
[mydev@MyRPi4-Fedora-1 riscv-pk]$ git diff
diff --git a/scripts/config.guess b/scripts/config.guess
index f32079a..00be375 100755
--- a/scripts/config.guess
+++ b/scripts/config.guess
@@ -142,7 +142,7 @@ UNAME_VERSION=`(uname -v) 2>/dev/null` || UNAME_VERSION=unknown
# Note: order is significant - the case branches are not exclusive.

case "${UNAME_MACHINE}:${UNAME_SYSTEM}:${UNAME_RELEASE}:${UNAME_VERSION}" in
-  *:NetBSD:*:*)
+  *:Linux:*:*)
      # NetBSD (nbsd) targets should (where applicable) match one or
      # more of the tuples: **-netbsdelf*, **-netbsdaout*,
      # **-netbsdecoff* and **-netbsd*. For targets that recently
[mydev@MyRPi4-Fedora-1 riscv-pk]$ 
```

```
[mydev@fedora ~]$ tree -L 2 -C /opt/MyWorkSpace/DevSW/Toolchain/RISC-V/Rocket/
/opt/MyWorkSpace/DevSW/Toolchain/RISC-V/Rocket/
+-- bin
|   |-- elf2hex
|   |-- riscv64-unknown-elf-addr2line
|   |-- riscv64-unknown-elf-ar
|   |-- riscv64-unknown-elf-as
|   |-- riscv64-unknown-elf-c++
|   |-- riscv64-unknown-elf-c++filt
|   |-- riscv64-unknown-elf-cpp
|   |-- riscv64-unknown-elf-elfedit
|   |-- riscv64-unknown-elf-g++
|   |-- riscv64-unknown-elf-occ
|   +-- riscv64-unknown-elf-occ-7.2.0 █
|       |-- riscv64-unknown-elf-gcc-ar
|       |-- riscv64-unknown-elf-gcc-nm
|       |-- riscv64-unknown-elf-gcc-ranlib
|       |-- riscv64-unknown-elf-gcov
|       |-- riscv64-unknown-elf-gcov-dump
|       |-- riscv64-unknown-elf-gcov-tool
|       |-- riscv64-unknown-elf-gdb
|       |-- riscv64-unknown-elf-gdb-add-index
|       |-- riscv64-unknown-elf-gprof
|       |-- riscv64-unknown-elf-ld
|       |-- riscv64-unknown-elf-ld.bfd
|       |-- riscv64-unknown-elf-nm
|       |-- riscv64-unknown-elf-objcopy
|       |-- riscv64-unknown-elf-objdump
|       |-- riscv64-unknown-elf-ranlib
|       |-- riscv64-unknown-elf-readelf
|       |-- riscv64-unknown-elf-run
|       |-- riscv64-unknown-elf-size
|       |-- riscv64-unknown-elf-strings
|       |-- riscv64-unknown-elf-strip
|       |-- spike
|       |-- spike-dasm
|       |-- spike-log-parser
|       |-- termios-xspike
|       |-- xspike
+-- include
    |-- fesvr
    |-- gdb
    |-- riscv
+-- lib
    |-- gcc
    |   |-- libfesvr.a
    |   |-- libriscv64-unknown-elf-sim.a
    |   |-- libsoftfloat.so
    |   |-- pkgconfig
    |-- libexec
        |-- gcc
    |-- riscv64-unknown-elf
        |-- bin
        |-- include
        |-- lib
        |-- share
    |-- share
        |-- gcc-7.2.0
        |-- gdb
        |-- info
        |-- locale
        |-- man
        |-- target
            |-- share
```

- Failed to build the latest GCC 11.x toolchain for RISC-V on RPi4(Fedora 34 with GCC 11.x)
-

3.2 Chisel

Firrtl

- <https://github.com/chipsalliance/firrtl>
an Intermediate Representation (IR) for digital circuits designed as a platform for writing circuit-level transformations.
- <https://github.com/chipsalliance/firrtl/blob/master/spec/spec.pdf>
- <https://github.com/chipsalliance/firrtl/wiki>
- **Tetralogy:**

```
2021-07-09 00:12:43 [warn] 72 warnings found
2021-07-09 00:12:43 [debug] Scala compilation took 399.173801018 s
2021-07-09 00:12:44 [debug] Attempting to call com.sun.tools.javac.api.JavacTool@3a5a091 directly...
2021-07-09 00:13:25 [info] /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/Firrtl-master_openjdk11/target/scala-2.12/src_managed/main/compiled_protobuf/firrtl/FirrtlProtos.java: FirrtlProtos.java uses or overrides a deprecated API.
2021-07-09 00:13:25 [info] /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/Firrtl-master_openjdk11/target/scala-2.12/src_managed/main/compiled_protobuf/firrtl/FirrtlProtos.java: Recompile with -Xlint:deprecation for details.
2021-07-09 00:13:25 [debug] Java compilation took 41.460717024 s
2021-07-09 00:13:34 [debug] Java analysis took 8.538800262 s
2021-07-09 00:13:34 [debug] Java compilation + analysis took 50.164186708 s
2021-07-09 00:13:34 [info] done compiling

2021-07-09 00:20:20 [debug] Done packaging.
2021-07-09 00:20:20 [info] :: delivering :: edu.berkeley.cs#firrtl_2.12;1.5-SNAPSHOT :: 1.5-SNAPSHOT :: integration :: Fri Jul 09 00:20:20 PDT 2021
2021-07-09 00:20:20 [debug]      options = status=integration pubdate=Fri Jul 09 00:20:20 PDT 2021 validate=true resolveDynamicRevisions=true merge=true resolveId=null pubBranch=null
2021-07-09 00:20:20 [info]      delivering ivy file to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/Firrtl-master_openjdk11/target/scala-2.12/ivy-1.5-SNAPSHOT.xml
2021-07-09 00:20:20 [debug]      deliver done (100ms)
2021-07-09 00:20:20 [debug] local do not support transaction. ivy pattern does not use revision as a directory
2021-07-09 00:20:20 [info]      published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/poms/firrtl_2.12.pom
2021-07-09 00:20:20 [info]      published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/jars/firrtl_2.12.jar
2021-07-09 00:20:20 [info]      published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/srcs/firrtl_2.12-sources.jar
2021-07-09 00:20:21 [info]      published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/docs/firrtl_2.12-javadoc.jar
2021-07-09 00:20:21 [info]      published ivy to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/ivys/ivy.xml
```

■ blocking test case with Z3

```
[mydev@fedora Materials]$ top
top - 11:03:36 up 3:14, 3 users, load average: 1.07, 1.15, 1.68
Tasks: 239 total, 2 running, 237 sleeping, 0 stopped, 0 zombie
%Cpu(s): 26.7 us, 5.3 sy, 0.0 ni, 68.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 7836.2 total, 105.6 free, 6858.0 used, 872.6 buff/cache
MiB Swap: 7836.0 total, 7171.7 free, 664.2 used, 866.2 avail Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
5543	mydev	20	0	8396300	3.8g	10724	R	100.0	49.4	24:49.52	z3
7800	mydev	20	0	249768	6100	5032	R	11.8	0.1	0:00.04	top
1928	mydev	20	0	429848	13688	12392	S	5.9	0.2	0:19.81	panel-8-pulseau
1932	mydev	20	0	349692	14412	12472	S	5.9	0.2	0:00.48	panel-9-power-m

```
[mydev@fedora Firrtl-master_openjdk11]$ ps aux |grep -i z3
mydev      5543 95.4 48.3 8068620 3882420 pts/0 R+    10:37   23:46 z3 test_run_dir/FPU-smt/2021070910372082852824856
36183091/FPU.smt2
```

Finally, out of memory!

Considering some remedies:

1) use jemalloc(<http://jemalloc.net/>)

```
[mydev@fedora Firrtl-master_openjdk11]$ cat /proc/5543/maps |grep -i jemalloc
ffff9491e000-ffff9499e000 r-xp 00000000 00:21 5856686          /usr/local/jemalloc/lib/libjemalloc.so.2
ffff9499e000-ffff949b8000 ---p 00080000 00:21 5856686          /usr/local/jemalloc/lib/libjemalloc.so.2
ffff949b8000-ffff949be000 r--p 0008a000 00:21 5856686          /usr/local/jemalloc/lib/libjemalloc.so.2
ffff949be000-ffff949bf000 rw-p 00090000 00:21 5856686          /usr/local/jemalloc/lib/libjemalloc.so.2
[mydev@fedora Firrtl-master_openjdk11]$
```

did not work.

2) replace ZRAM with SWAP file

<https://askubuntu.com/questions/1206157/can-i-have-a-swapfile-on-btrfs>

failed to enable a SWAP file on BTRFS

...

Treadle

- <https://github.com/chipsalliance/treadle>
- **an experimental circuit simulator that executes low Firrtl IR.**
- **Tetralogy:**

```
2021-07-06 01:14:00 [debug] Scala compilation took 193.945763528 s
2021-07-06 01:14:00 [info] Done compiling.

2021-07-06 01:19:26 [info] Run completed in 2 minutes, 12 seconds.
2021-07-06 01:19:26 [info] Total number of tests run: 247
2021-07-06 01:19:26 [info] Suites: completed 79, aborted 0
2021-07-06 01:19:26 [info] Tests: succeeded 247, failed 0, canceled 0, ignored 1, pending 0
2021-07-06 01:19:26 [info] All tests passed.

2021-07-06 01:21:08 [info] Wrote /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/Treadle-master_openjdk11/target/scala-2.12/treadle_2.12-1.5-SNAPSHOT.pom
2021-07-06 01:21:08 [debug] Done packaging.
2021-07-06 01:21:08 [info] Main Scala API documentation to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/Treadle-master_openjdk11/target/scala-2.12/api...

2021-07-06 01:23:13 [debug]      deliver done (83ms)
2021-07-06 01:23:13 [debug] local do not support transaction. ivy pattern does not use revision as a directory
2021-07-06 01:23:13 [info]      published treadle_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.12/1.5-SNAPSHOT/po
ms/treadle_2.12.pom
2021-07-06 01:23:13 [info]      published treadle_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.12/1.5-SNAPSHOT/ja
rs/treadle_2.12.jar
2021-07-06 01:23:13 [info]      published treadle_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.12/1.5-SNAPSHOT/sr
cs/treadle_2.12-sources.jar
2021-07-06 01:23:13 [info]      published treadle_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.12/1.5-SNAPSHOT/do
cs/treadle_2.12-javadoc.jar
2021-07-06 01:23:13 [info]      published ivy to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.12/1.5-SNAPSHOT/ivys/ivy.xml
```

Chisel3

- After **Firrtl** and **Treadle** settle down:
meet the same **Z3** related issue while doing test
-

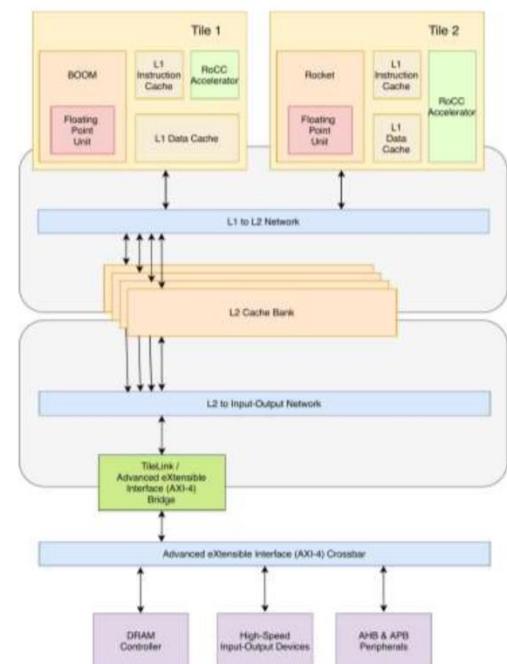
Rocket Chip Generator

- <https://github.com/chipsalliance/rocket-chip>

Rocket Chip is an open-source System-on-Chip design generator that emits synthesizable RTL. It leverages the Chisel hardware construction language to compose a library of sophisticated generators for cores, caches, and interconnects into an integrated SoC. Rocket Chip generates general-purpose processor cores that use the open RISC-V ISA, and provides both an in-order core generator (Rocket) and an out-of-order core generator (BOOM). For SoC designers interested in utilizing heterogeneous specialization for added efficiency gains, Rocket Chip supports the integration of custom accelerators in the form of instruction set extensions, coprocessors, or fully independent novel cores. Rocket Chip has been taped out (manufactured) eleven times, and yielded functional silicon prototypes capable of booting Linux.

- - Collection of SoC building blocks
 - Parameterized
 - Standard Interfaces
 - Chisel: open source hardware construction language embedded in scala
 - FIRRTL is used to emit Verilog from the Chisel Model
 - Composes RISC-V ISA based platforms

Cores	in-order scalar , out-of-order
Core Tiles	FPU, L1 Caches, Custom Accelerators, FPU
Tile Link	On chip fabric , Coherent Caches
Peripheral	Limited Support



Source: “Reverse Engineering of Rocket-Chip”, Dr. Roomi Naqvi, MERL Researchers & Students, RISC-V Summit 2020.

3.3 SpinalHDL

My patch to SpinalHDL for RPi4 (merged)

- <https://github.com/SpinalHDL/SpinalHDL/pull/412>

The screenshot displays three code editor windows side-by-side, showing patches for different files:

- Top Left:** sim/src/main/resources/SharedStruct.hpp. This window shows changes to the header file. It includes #include directives for boost::interprocess and std::exception, and defines ShmemAllocator and SharedVector using boost::interprocess::allocator and managed_shared_memory::segment_manager.
- Top Right:** sim/src/main/resources/sharedMemIface.cpp. This window shows a patch for the SharedMemIface class. It includes logic for checking if spin_count is less than SPINLOCK_MAX_ACQUIRE_SPINS, and if so, it performs _mm_pause() or _spin_pause() based on the NO_SPINLOCK_YIELD_OPTIMIZATION define.
- Bottom:** sim/src/main/scalable/sim/verilatorBackend.scala. This window shows a patch for the spinal/scalable/sim/verilatorBackend.scala file. It includes logic for generating C code for JNIEXPORT void API JNIMCALL \${jniPrefix}disableNative_1S(uniqueId) and handling flags and arch properties.

New Issue with GCC 11

```
2021-07-04 10:30:13 [info] SpinalSimLibTester:  
2021-07-04 10:30:13 [Warning] 8 signals were pruned. You can call printPruned on the backend report to get more informations.  
2021-07-04 10:30:13 [Done] at 6.105  
2021-07-04 10:30:13 [info] Axi4SharedSdramCtrlTesterCocotbBoot:  
2021-07-04 10:30:13 [info] - genVerilog  
2021-07-04 10:30:13  
2021-07-04 10:30:14 [Progress] at 7.258 : Checks and transforms  
2021-07-04 10:30:19 [Progress] at 11.569 : Generate Verilog  
2021-07-04 10:30:24 [Warning] 81 signals were pruned. You can call printPruned on the backend report to get more informations.  
2021-07-04 10:30:24 [Done] at 16.542  
2021-07-04 10:30:24 [Progress] Simulation workspace in /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/SpinalHDL/SpinalHDL-dev_openjdk11./simWorkspace/Usb0hciTbTop  
2021-07-04 10:30:24 [Progress] Verilator compilation started  
2021-07-04 10:30:25 /usr/share/verilator/include/verilated.cpp: In function 'IData VL_FGETS_NI(std::string&, IData)':  
2021-07-04 10:30:25 /usr/share/verilator/include/verilated.cpp:1318:36: error: 'numeric_limits' is not a member of 'std'  
2021-07-04 10:30:25 1318 |     return getLine(dest, fpi, std::numeric_limits<size_t>::max());  
2021-07-04 10:30:25 |     ~~~~~~  
2021-07-04 10:30:25 /usr/share/verilator/include/verilated.cpp:1318:57: error: expected primary-expression before '>' token  
2021-07-04 10:30:25 1318 |     return getLine(dest, fpi, std::numeric_limits<size_t>::max());  
2021-07-04 10:30:25 |     ~~~~~~  
2021-07-04 10:30:25 /usr/share/verilator/include/verilated.cpp:1318:60: error: '::max' has not been declared; did you mean 'std::max'?  
2021-07-04 10:30:25 1318 |     return getLine(dest, fpi, std::numeric_limits<size_t>::max());  
2021-07-04 10:30:25 |     ^~~  
2021-07-04 10:30:25 std::max  
2021-07-04 10:30:25 In file included from /usr/include/c++/11/algorithm:62,  
2021-07-04 10:30:25         from /usr/share/verilator/include/verilated_heavy.h:29,  
2021-07-04 10:30:25         from /usr/share/verilator/include/verilated_imp.h:29,  
2021-07-04 10:30:25         from /usr/share/verilator/include/verilated.cpp:25:  
2021-07-04 10:30:25 /usr/include/c++/11/bits/stl_algo.h:3467:5: note: 'std::max' declared here  
2021-07-04 10:30:25 3467 |     max(initializer_list<_Tp> __l, _Compare __comp)  
2021-07-04 10:30:25 |     ^~~  
2021-07-04 10:30:25 make: *** [/usr/share/verilator/include/verilated.mk:241: verilated.o] Error 1  
2021-07-04 10:30:25 [info] - CountOnes0 *** FAILED ***  
2021-07-04 10:30:25 [info] java.lang.AssertionError: assertion failed: Verilator C++ model compilation failed  
2021-07-04 10:30:25 [info] at scala.Predef$.assert(Predef.scala:170)  
2021-07-04 10:30:25 [info] at spinal.sim.VerilatorBackend.compileVerilator(VerilatorBackend.scala:491)  
2021-07-04 10:30:25 [info] at spinal.sim.VerilatorBackend.<init>(VerilatorBackend.scala:553)  
2021-07-04 10:30:25 [info] at spinal.core.sim.SpinalVerilatorBackend$.apply(SimBootstraps.scala:135)  
2021-07-04 10:30:25 [info] at spinal.core.sim.SpinalSimConfig.compile(SimBootstraps.scala:637)  
2021-07-04 10:30:25 [info] at spinal.core.sim.SpinalSimConfig.compile(SimBootstraps.scala:600)  
2021-07-04 10:30:25 [info] at SpinalSimLibTester$$anonfun$1$anonfun$apply$mcVI$sp$1.apply$mcV$sp(SpinalSimLibTester.scala:15)  
2021-07-04 10:30:25 [info] at SpinalSimLibTester$$anonfun$1$anonfun$apply$mcVI$sp$1.apply(SpinalSimLibTester.scala:18)  
2021-07-04 10:30:25 [info] at SpinalSimLibTester$$anonfun$1$anonfun$apply$mcVI$sp$1.apply(SpinalSimLibTester.scala:18)  
2021-07-04 10:30:25 [Runtime] SpinalHDL v1.5.1 git head : 99d50e8ab5f72cada1103d3a011a06059621bf25  
2021-07-04 10:30:25 [Runtime] JVM max memory : 1960.0MiB
```

Result:

```
2021-07-04 11:08:18 make[1]: Leaving directory '/opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/SpinalHDL/SpinalHDL-dev_openjdk11/tester/src/test/python/spinal/Axi4SharedOnChipRamTester'
2021-07-04 11:08:18
2021-07-04 11:08:18 /usr/lib64/python3.9/site-packages/cocotb/handle.py:166: UserWarning: Use of attribute 'log' is deprecated, use '_log' instead
2021-07-04 11:08:18   warnings.warn("Use of attribute %r is deprecated, use %r instead" % (name, self._compat_mapping[name]))
2021-07-04 11:08:18 sys::1: ResourceWarning: unclosed file <_io.TextIOWrapper name=2 mode='w' encoding='UTF-8'>
2021-07-04 11:08:18 [info] - cocotbVerilog
2021-07-04 11:08:18 [debug]     Produced 0 nested tasks and 2 events.
2021-07-04 11:08:19 [info] Run completed in 38 minutes, 15 seconds.
2021-07-04 11:08:19 [info] Total number of tests run: 546
2021-07-04 11:08:19 [info] Suites: completed 96, aborted 0
2021-07-04 11:08:19 [info] Tests: succeeded 345, failed 201, canceled 0, ignored 0, pending 0
2021-07-04 11:08:19 [info] *** 201 TESTS FAILED ***
```

GCC 11:

https://www.gnu.org/software/gcc/gcc-11/porting_to.html

Header dependency changes

Some C++ Standard Library headers have been changed to no longer include other headers that they do need to depend on. As such, C++ programs that used standard library components without including the right headers will no longer compile.

The following headers are used less widely in libstdc++ and may need to be included explicitly when compiled with GCC 11:

- <limits> (for std::numeric_limits)
- <memory> (for std::unique_ptr, std::shared_ptr etc.)
- <utility> (for std::pair, std::tuple_size, std::index_sequence etc.)
- <thread> (for members of namespace std::this_thread.)

Solution:

```
[mydev@fedora include]$ pwd
/usr/share/verilator/include
[mydev@fedora include]$
[mydev@fedora include]$ colordiff verilated.cpp.orgi verilated.cpp
35a36
> #include <limits>
[mydev@fedora include]$
```

Current result:

```
2021-07-04 23:10:08 make[1]: Leaving directory '/opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/SpinalHDL/SpinalHDL-dev_openjdk11/tester/src/test/python/spinal/Axi4SharedOnChipRamTester'
2021-07-04 23:10:08
2021-07-04 23:10:08 /usr/lib64/python3.9/site-packages/cocotb/handle.py:166: UserWarning: Use of attribute 'log' is deprecated, use '_log' instead
2021-07-04 23:10:08   warnings.warn("Use of attribute %r is deprecated, use %r instead" % (name, self._compat_mapping[name]))
2021-07-04 23:10:08 sys::1: ResourceWarning: unclosed file <_io.TextIOWrapper name=2 mode='w' encoding='UTF-8'>
2021-07-04 23:10:08 [info] - cocotbVerilog
2021-07-04 23:10:08 [debug]     Produced 0 nested tasks and 2 events.
2021-07-04 23:10:08 - /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/SpinalHDL/SpinalHDL-dev_openjdk11/tmp/job_64/unamed.v:22: Verilog $finish
2021-07-04 23:10:08 [info] Run completed in 48 minutes, 43 seconds.
2021-07-04 23:10:08 [info] Total number of tests run: 546
2021-07-04 23:10:08 [info] Suites: completed 96, aborted 0
2021-07-04 23:10:08 [info] Tests: succeeded 543, failed 3, canceled 0, ignored 0, pending 0
2021-07-04 23:10:08 [info] *** 3 TESTS FAILED ***

2021-07-04 23:10:08 [error] Failed tests:
2021-07-04 23:10:08 [error]    spinal.tester.scalatest.SpinalSimVerilatorIoTest
2021-07-04 23:10:11 [error] (tester / Test / test) sbt.TestsFailedException: Tests unsuccessful
```

There are still 3 failed tests.

VexRiscv

■ <https://github.com/SpinalHDL/VexRiscv>

A FPGA friendly 32 bit RISC-V CPU implementation which is written in SpinalHDL

■ **Specs:**

- RV32I[M][A][F[D]][C] instruction set
- Pipelined from 2 to 5+ stages ([Fetch*X], Decode, Execute, [Memory], [WriteBack])
- 1.44 DMIPS/Mhz --no-inline when nearly all features are enabled (1.57 DMIPS/Mhz when the divider lookup table is enabled)
- Optimized for FPGA, does not use any vendor specific IP block / primitive
- AXI4, Avalon, wishbone ready
- Optional MUL/DIV extensions
- Optional F32/F64 FPU (require data cache for now)
- Optional instruction and data caches
- Optional hardware refilled MMU
- Optional debug extension allowing Eclipse debugging via a GDB >> openOCD >> JTAG connection
- Optional interrupts and exception handling with Machine, [Supervisor] and [User] modes as defined in the [RISC-V Privileged ISA Specification v1.10](#).
- Two implementations of shift instructions: single cycle (full barrel shifter) and shiftNumber cycles
- Each stage can have optional bypass or interlock hazard logic
- Linux compatible (SoC : <https://github.com/enjoy-digital/linux-on-litex-vexriscv>)
- Zephyr compatible
- FreeRTOS port

■ <https://riscv.org/announcements/2018/10/risc-v-contest/>

Tetralogy:

```
2021-07-05 01:24:53 [warn] there were 36 deprecation warnings; re-run with -deprecation for details
2021-07-05 01:24:53 [warn] there were two feature warnings; re-run with -feature for details
2021-07-05 01:24:53 [warn] four warnings found
2021-07-05 01:24:53 [debug] Scala compilation took 273.337719866 s
2021-07-05 01:24:53 [info] Done compiling.

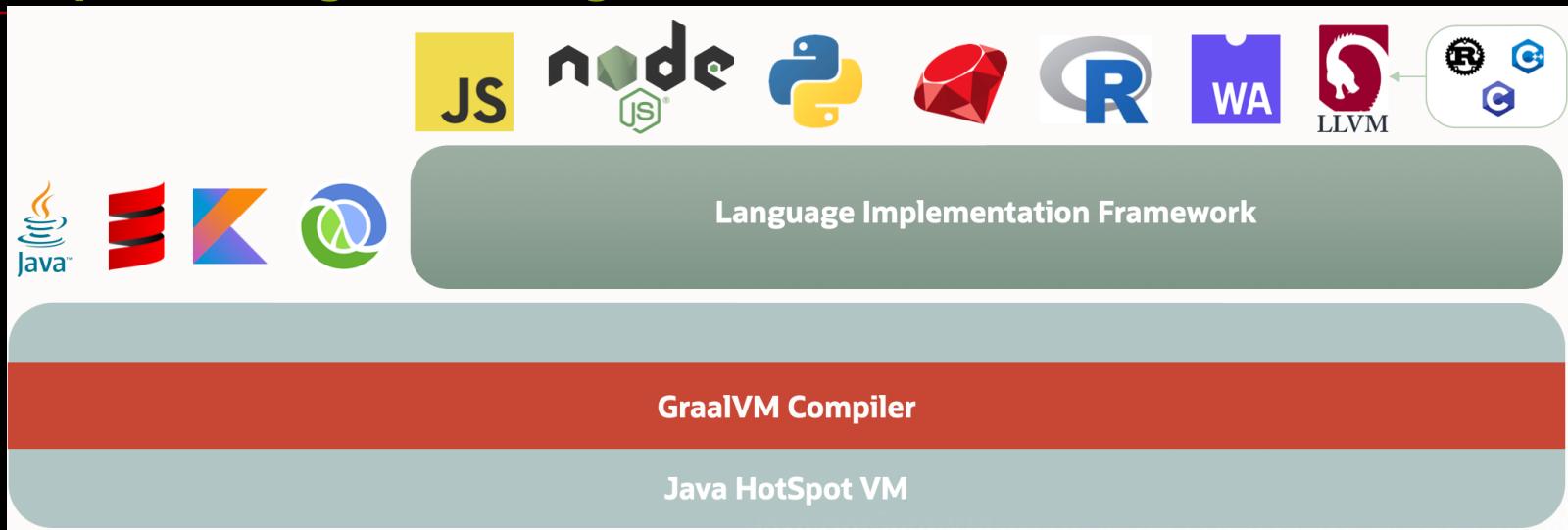
2021-07-05 06:22:06 [Progress] Start FpuCore test simulation with seed 42
2021-07-05 06:22:06 [Error] Simulation failed at time=0
2021-07-05 06:22:06 [info] - f32 *** FAILED ***
2021-07-05 06:22:06 [info] java.io.IOException: Cannot run program "testfloat_gen": error=2, No such file or directory
2021-07-05 06:22:06 [info] at java.base/java.lang.ProcessBuilder.start(ProcessBuilder.java:1128)
2021-07-05 06:22:06 [info] at java.base/java.lang.ProcessBuilder.startStart(ProcessorBuilder.java:1071)
2021-07-05 06:22:06 [info] at scala.sys.process.ProcessBuilderImpl$Simple.run(ProcessBuilderImpl.scala:69)
2021-07-05 06:22:06 [info] at scala.sys.process.ProcessBuilderImpl$AbstractBuilder.run(ProcessBuilderImpl.scala:100)
2021-07-05 06:22:06 [info] at scala.sys.process.ProcessBuilderImpl$AbstractBuilder.run(ProcessBuilderImpl.scala:99)
2021-07-05 06:22:06 [info] at vexriscv.ip.fpu.ProcessStream.<init>(FpuTest.scala:1647)
2021-07-05 06:22:06 [info] at vexriscv.ip.fpu.FpuTest$$anonfun$testP$2$TestCase$1$.build(FpuTest.scala:72)
2021-07-05 06:22:06 [info] at vexriscv.ip.fpu.FpuTest$$anonfun$testP$2$TestCase$1$.build(FpuTest.scala:145)
2021-07-05 06:22:06 [info] at vexriscv.ip.fpu.FpuTest$$anonfun$testP$2$TestCase$1$.RAW(FpuTest.scala:145)
2021-07-05 06:22:06 [info] ...
2021-07-05 06:22:06 [info] Cause: java.io.IOException: error=2, No such file or directory
2021-07-05 06:22:06 [info] at java.base/java.lang.ProcessImpl.forkAndExec(Native Method)
2021-07-05 06:22:06 [info] at java.base/java.lang.ProcessImpl.<init>(ProcessImpl.java:340)
2021-07-05 06:22:06 [info] at java.base/java.lang.ProcessImpl.start(ProcessImpl.java:271)
2021-07-05 06:22:06 [info] at java.base/java.lang.ProcessBuilder.start(ProcessBuilder.java:1107)
2021-07-05 06:22:06 [info] at java.base/java.lang.ProcessBuilder.startStart(ProcessorBuilder.java:1071)
2021-07-05 06:22:06 [info] at scala.sys.process.ProcessBuilderImpl$Simple.run(ProcessBuilderImpl.scala:69)
2021-07-05 06:22:06 [info] at scala.sys.process.ProcessBuilderImpl$AbstractBuilder.run(ProcessBuilderImpl.scala:100)
2021-07-05 06:22:06 [info] at scala.sys.process.ProcessBuilderImpl$AbstractBuilder.run(ProcessBuilderImpl.scala:99)
2021-07-05 06:22:06 [info] at vexriscv.ip.fpu.ProcessStream.<init>(FpuTest.scala:1647)
2021-07-05 06:22:06 [info] at vexriscv.ip.fpu.FpuTest$$anonfun$testP$2$TestCase$1$.build(FpuTest.scala:72)
2021-07-05 06:22:06 [info] ...
2021-07-05 06:22:06 [debug] Produced 0 nested tasks and 2 events.
2021-07-05 06:22:08 [info] Run completed in 4 hours, 55 minutes, 7 seconds.
2021-07-05 06:22:08 [info] Total number of tests run: 222
2021-07-05 06:22:08 [info] Suites: completed 4, aborted 0
2021-07-05 06:22:08 [info] Tests: succeeded 220, failed 2, canceled 0, ignored 0, pending 0
2021-07-05 06:22:08 [info] *** 2 TESTS FAILED ***
2021-07-05 06:22:08 [debug] Passed tests:
2021-07-05 06:22:08 [debug] vexriscv.FunTestPara
2021-07-05 06:22:08 [debug] vexriscv.DhrystoneBench
2021-07-05 06:22:08 [debug] vexriscv.TestIndividualFeatures
2021-07-05 06:22:08 [error] Failed tests:
2021-07-05 06:22:08 [error] vexriscv.ip.fpu.FpuTest
2021-07-05 06:22:09 [error] (Test / test) sbt.TestsFailedException: Tests unsuccessful

2021-07-05 12:34:31 [info] Done packaging.
2021-07-05 12:34:31 [info] :: delivering :: com.github.spinalhdl#vexriscv_2.11;2.0.0 :: 2.0.0 :: release :: Mon Jul 05 12:34:31 PDT 2021
2021-07-05 12:34:31 [debug] options = status=release pubdate=Mon Jul 05 12:34:31 PDT 2021 validate=true resolveDynamicRevisions=true merge=true res
olveId=null pubBranch=null
2021-07-05 12:34:31 [info] delivering ivy file to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/SpinalHDL/VexRiscv-master_openjdk11/target/scala-2.11/ivy
-2.0.0.xml
2021-07-05 12:34:31 [debug] deliver done (106ms)
2021-07-05 12:34:31 [debug] local do not support transaction. ivy pattern does not use revision as a directory
2021-07-05 12:34:31 [info] published vexriscv_2.11 to /home/mydev/.ivy2/local/com.github.spinalhdl/vexriscv_2.11/2.0.0/poms/vexriscv_2.11.pom
2021-07-05 12:34:31 [info] published vexriscv_2.11 to /home/mydev/.ivy2/local/com.github.spinalhdl/vexriscv_2.11/2.0.0/jars/vexriscv_2.11.jar
2021-07-05 12:34:31 [info] published vexriscv_2.11 to /home/mydev/.ivy2/local/com.github.spinalhdl/vexriscv_2.11/2.0.0/srcs/vexriscv_2.11-sources.ja
r
2021-07-05 12:34:31 [info] published vexriscv_2.11 to /home/mydev/.ivy2/local/com.github.spinalhdl/vexriscv_2.11/2.0.0/docs/vexriscv_2.11-javadoc.ja
r
2021-07-05 12:34:31 [info] published ivy to /home/mydev/.ivy2/local/com.github.spinalhdl/vexriscv_2.11/2.0.0/ivys/ivy.xml
```

III. Speed up Chisel/SpinalHDL

1) GraalVM

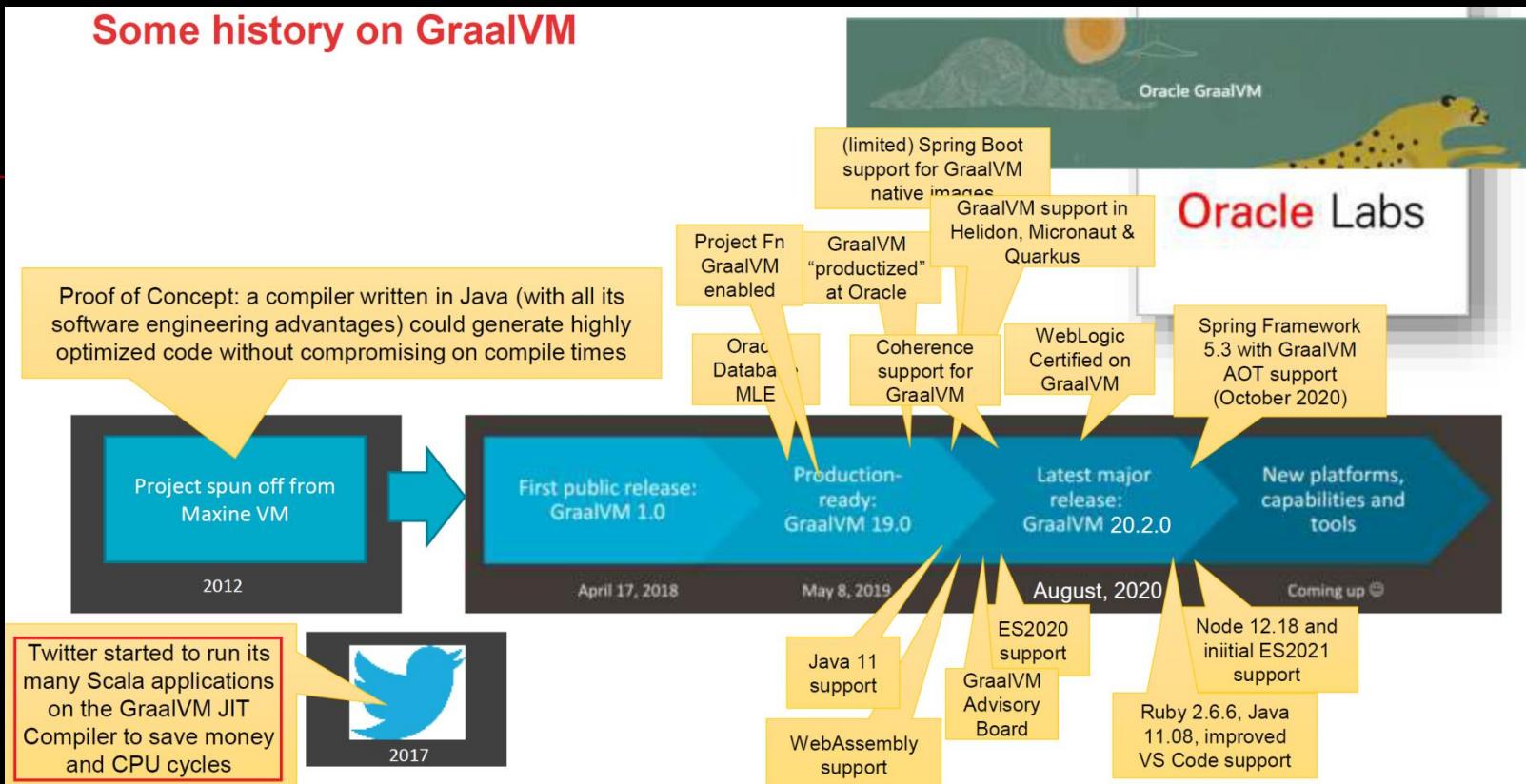
- <https://www.graalvm.org/>



- **A Universal High-Performance Polyglot VM**
- **A meta-runtime for Language-Level Virtualization**
- **Currently base an Oracle Labs JDK 8, 11, 16 with JVMCI support**
- <https://www.graalvm.org/docs/introduction/>
- <https://github.com/oracle/graal/blob/master/docs/>

History

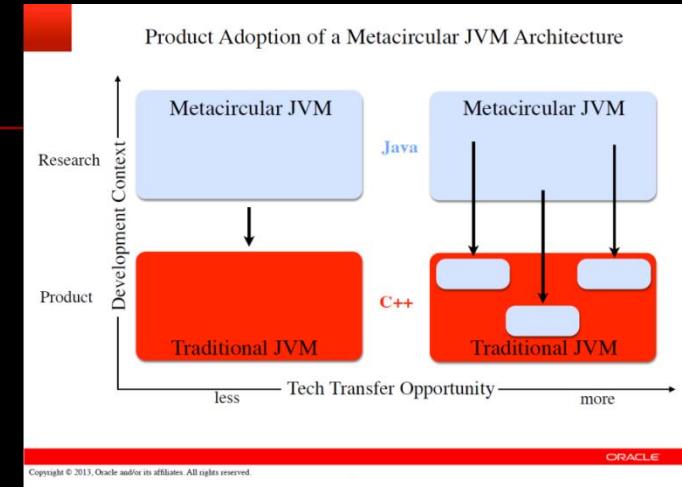
■ Some history on GraalVM



Source: “How and why GraalVM is quickly becoming relevant for you”, Lucas Jellema, DOAG 2020.

Meta-Circular

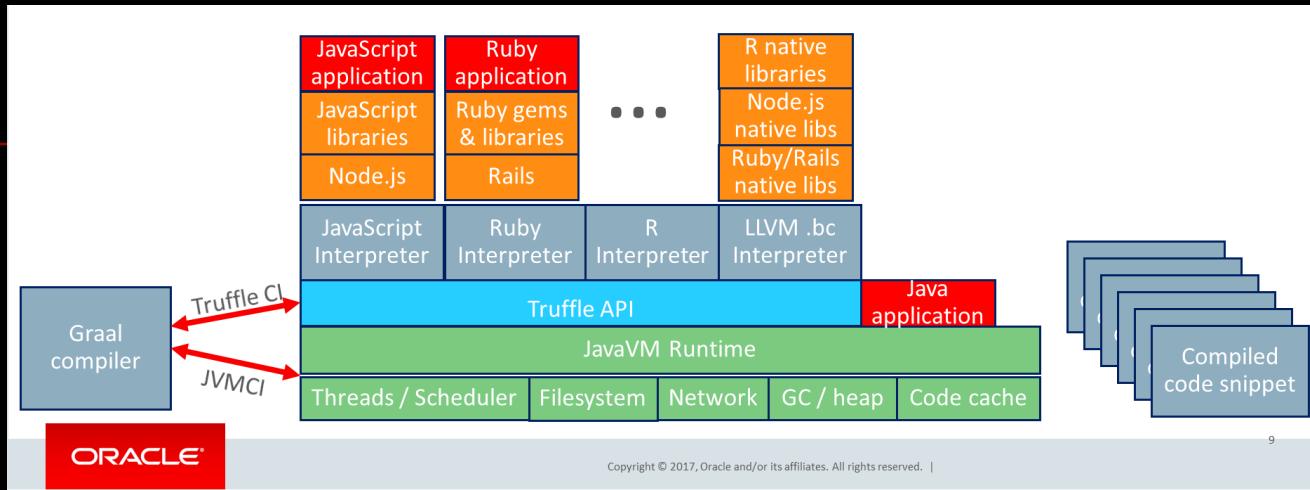
- Maxine → GraalVM



Source: <https://chrisseaton.com/truffleruby/jokerconf17/>

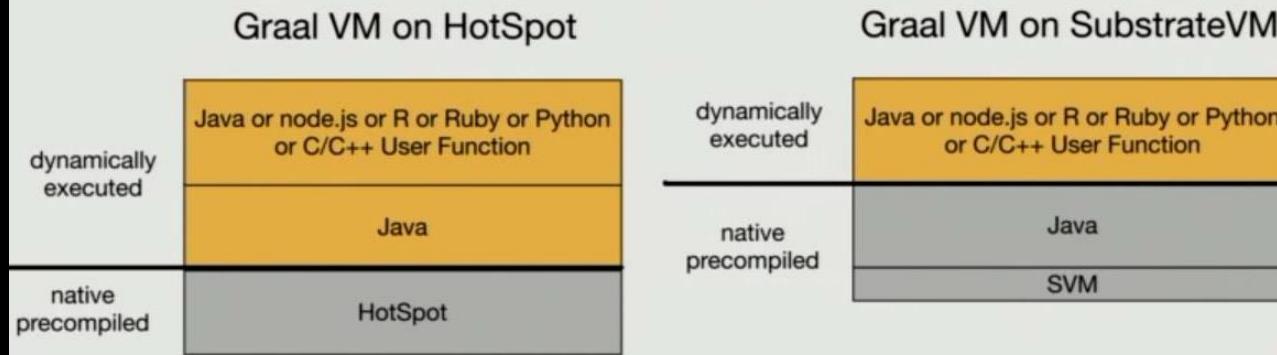
Arch

■ A hybrid of static & dynamic runtimes



Source: <https://ics.psu.edu/wp-content/uploads/2017/02/GraalVM-PSU.pptx>

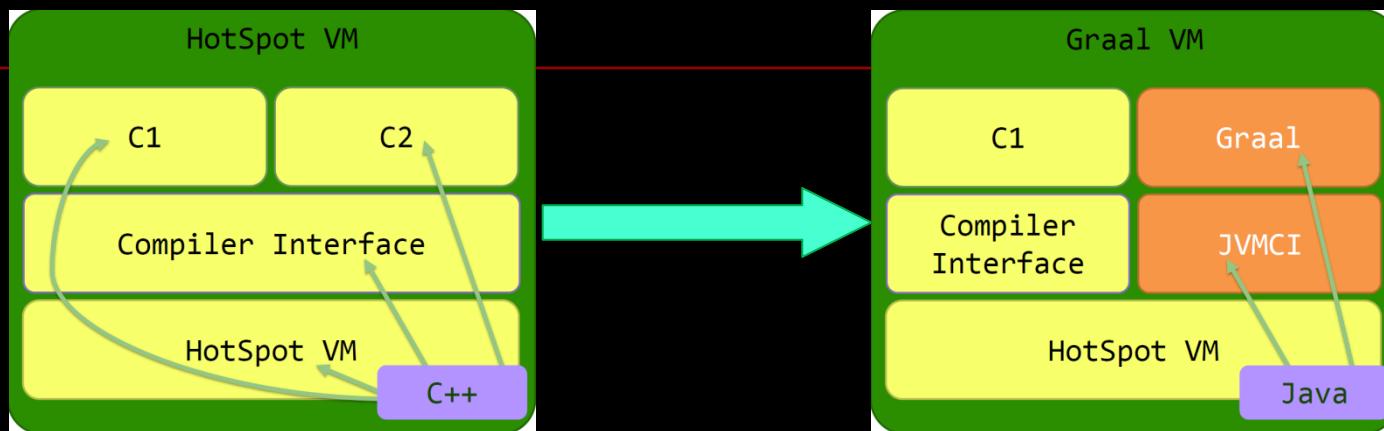
- Precompile core parts of application, but still allow extensibility!



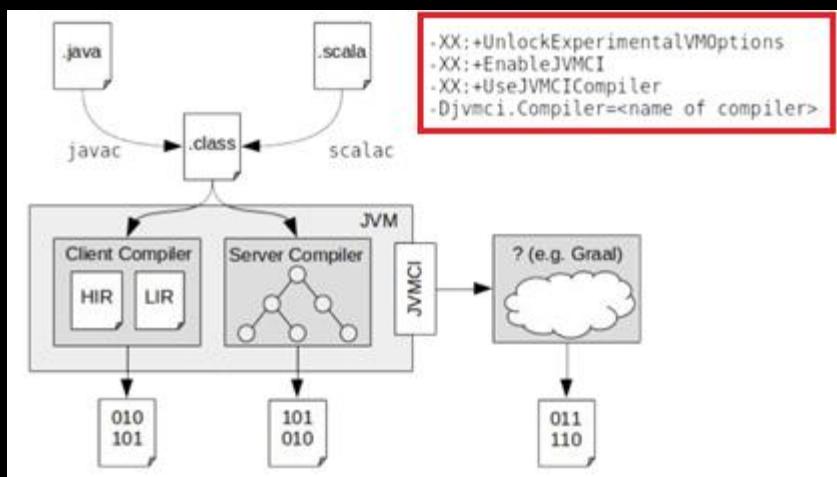
Source: “Adopting Java for the Serverless world”, Vadym Kazulkin, JUG London 2020.

JVMCI

- Java-Level JVM Compiler Interface
- [http://openjdk.java.net/jeps/243: experimental in JDK 9](http://openjdk.java.net/jeps/243)



Source: <https://www.slideshare.net/jyukutyo/jvmgraalopenj9>



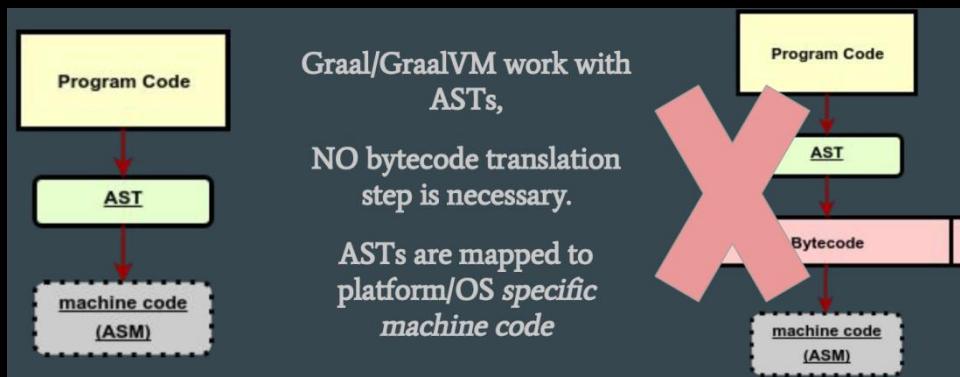
Source: <https://www.dynatrace.com/news/blog/new-ways-introducing-compiled-code-java-9/>

Truffle & Graal

- <https://github.com/neomatrix369/awesome-graal>
- AST

https://en.wikipedia.org/wiki/Abstract_syntax_tree

~~Graal/GraalVM: ASTs as first class citizen~~



Source: http://crest.cs.ucl.ac.uk/cow/59/slides/cow59_Sarkar.pdf

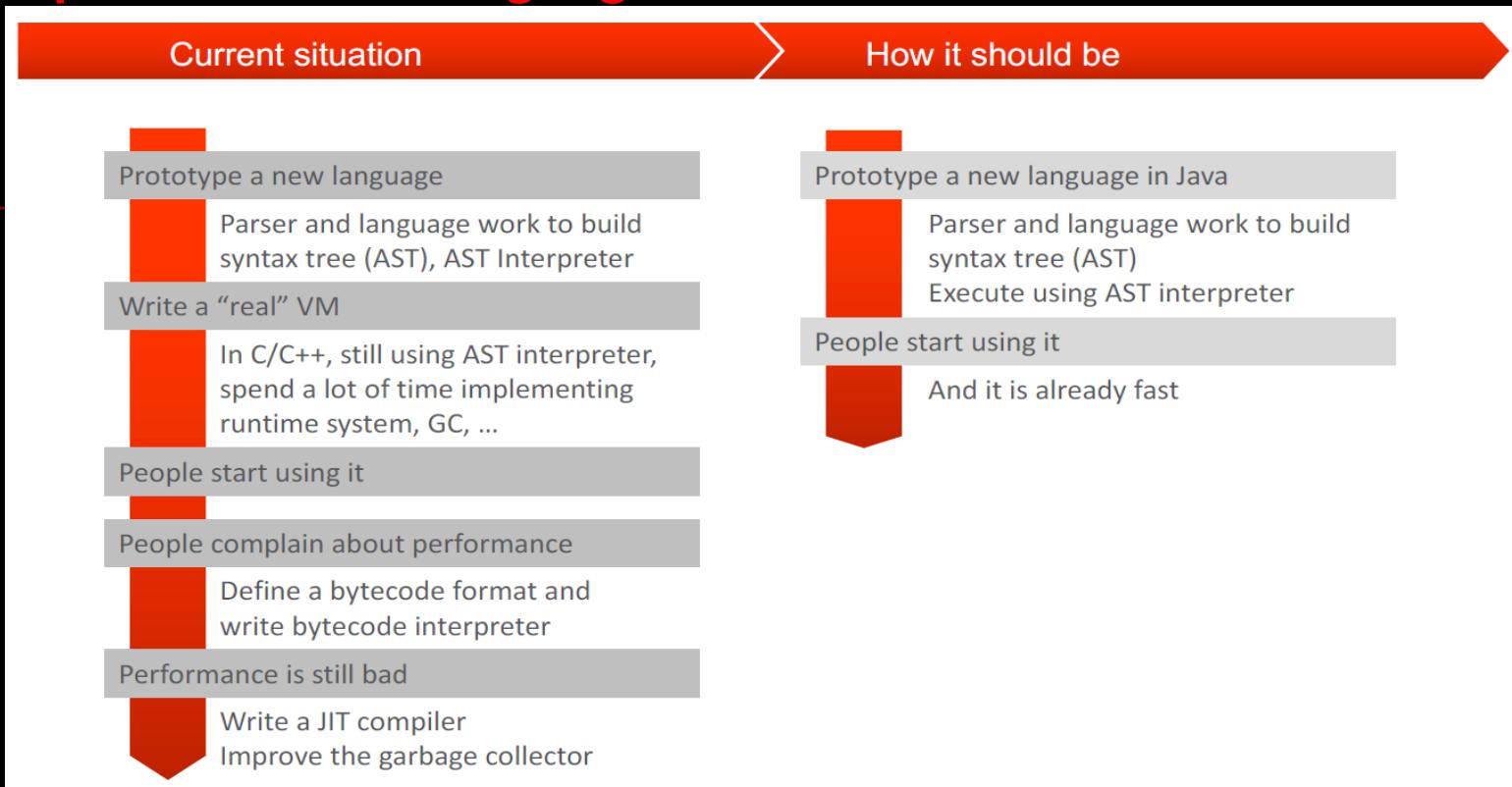
-

IR Graph-Based IR
 Platform Independent Graph-Based IR
 Platform Dependent



Source: http://ssw.jku.at/Research/Papers/Stadler14PhD/Thesis_Stadler_14.pdf

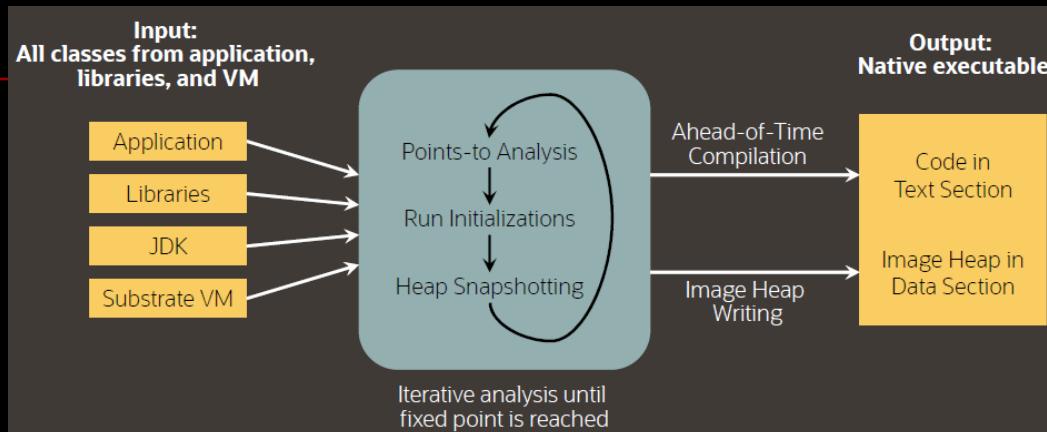
■ Implement a new language runtime



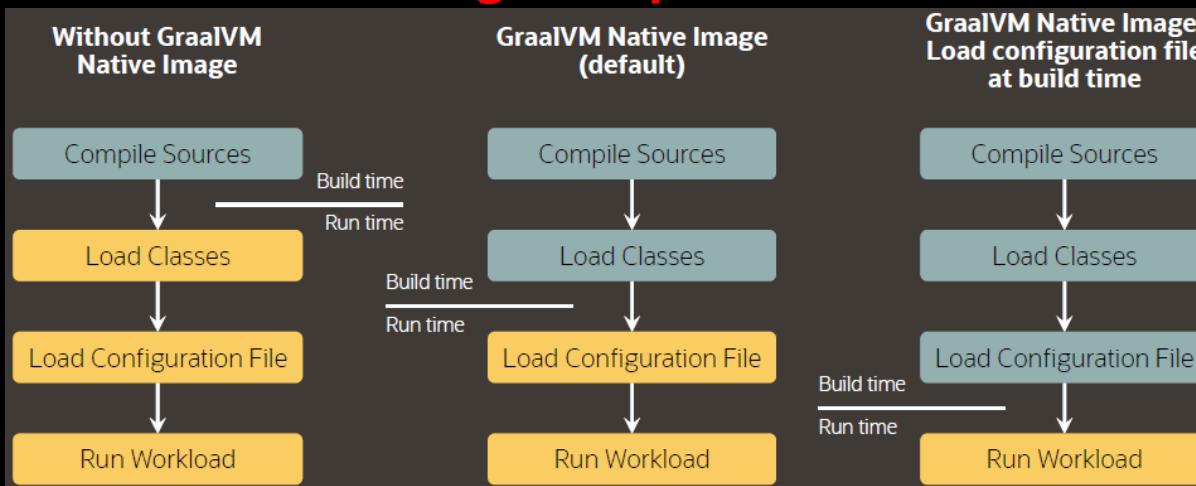
Source: “Turning the JVM into a Polyglot VM with Graal”, Chris Seaton, Oracle Labs

Native Image

- <https://www.graalvm.org/reference-manual/native-image/>
- <https://www.graalvm.org/examples/native-image-examples/>



■ Benefits of the Image Heap



Source: https://www.jug-gr.de/downloads/JDK_14_und_GraalVM_im_Java-%C3%96kosystem_WW.pdf

Src

- **<https://github.com/oracle/graal>**

Repository Structure

The GraalVM main source repository includes the components listed below. The documentation for each component includes developer instructions for the component.

- [GraalVM SDK](#) contains long term supported APIs of GraalVM.
- [GraalVM compiler](#) written in Java that supports both dynamic and static compilation and can integrate with the Java HotSpot VM or run standalone.
- [Truffle](#) language implementation framework for creating languages and instrumentations for GraalVM.
- [Tools](#) contains a set of tools for GraalVM languages implemented with the instrumentation framework.
- [Substrate VM](#) framework that allows ahead-of-time (AOT) compilation of Java applications under closed-world assumption into executable images or shared objects.
- [Sulong](#) is an engine for running LLVM bitcode on GraalVM.
- [GraalWasm](#) is an engine for running WebAssembly programs on GraalVM.
- [TRegex](#) is an implementation of regular expressions which leverages GraalVM for efficient compilation of automata.
- [VM](#) includes the components to build a modular GraalVM image.
- [VS Code](#) provides extensions to Visual Studio Code that support development of polyglot applications using GraalVM.

- **<https://github.com/graalvm>**
- **<https://github.com/oracle/graalpython>**
- ...

Installation

■ <https://www.graalvm.org/docs/getting-started/>

■ **Assets:**

e.g., [https://github.com/graalvm/graalvm-ce-dev-builds/releases
/tag/21.3.0-dev-20210701_2305](https://github.com/graalvm/graalvm-ce-dev-builds/releases/tag/21.3.0-dev-20210701_2305)

⋮ espresso-installable-svm-java11-darwin-amd64-dev.jar
⋮ espresso-installable-svm-java11-linux-aarch64-dev.jar
⋮ espresso-installable-svm-java11-linux-amd64-dev.jar
⋮ espresso-installable-svm-java11-windows-amd64-dev.jar
⋮ espresso-installable-svm-java8-linux-amd64-dev.jar
⋮ espresso-installable-svm-java8-windows-amd64-dev.jar
⋮ graalpython-dev-linux-amd64.tar.gz
⋮ graalpython-dev-macos-amd64.tar.gz
⋮ graalvm-ce-java11-darwin-amd64-dev.tar.gz
⋮ graalvm-ce-java11-linux-aarch64-dev.tar.gz
⋮ graalvm-ce-java11-linux-amd64-dev.tar.gz
⋮ graalvm-ce-java11-windows-amd64-dev.zip
⋮ graalvm-ce-java16-darwin-amd64-dev.tar.gz
⋮ graalvm-ce-java16-linux-aarch64-dev.tar.gz
⋮ graalvm-ce-java16-linux-amd64-dev.tar.gz
⋮ graalvm-ce-java16-windows-amd64-dev.zip
⋮ graalvm-ce-java8-linux-amd64-dev.tar.gz
⋮ graalvm-ce-java8-windows-amd64-dev.zip
⋮ llvm-toolchain-installable-java11-darwin-amd64-dev.jar
⋮ llvm-toolchain-installable-java11-linux-aarch64-devjar
⋮ llvm-toolchain-installable-java11-linux-amd64-devjar
⋮ llvm-toolchain-installable-java16-darwin-amd64-devjar
⋮ llvm-toolchain-installable-java16-linux-aarch64-devjar
⋮ llvm-toolchain-installable-java16-linux-amd64-devjar
⋮ llvm-toolchain-installable-java8-darwin-amd64-dev.jar
⋮ llvm-toolchain-installable-java8-linux-amd64-dev.jar
⋮ native-image-installable-svm-java11-darwin-amd64-dev.jar
⋮ native-image-installable-svm-java11-linux-aarch64-dev.jar
⋮ native-image-installable-svm-java11-linux-amd64-dev.jar
⋮ native-image-installable-svm-java11-windows-amd64-dev.jar
⋮ native-image-installable-svm-java16-darwin-amd64-devjar
⋮ native-image-installable-svm-java16-linux-aarch64-devjar
⋮ native-image-installable-svm-java16-linux-amd64-devjar
⋮ native-image-installable-svm-java16-windows-amd64-devjar
⋮ native-image-installable-svm-java8-linux-amd64-devjar
⋮ native-image-installable-svm-java8-windows-amd64-devjar

⋮ nodejs-installable-svm-java11-darwin-amd64-dev.jar
⋮ nodejs-installable-svm-java11-linux-aarch64-devjar
⋮ nodejs-installable-svm-java11-linux-amd64-devjar
⋮ nodejs-installable-svm-java11-windows-amd64-dev.jar
⋮ nodejs-installable-svm-java16-darwin-amd64-dev.jar
⋮ nodejs-installable-svm-java16-linux-aarch64-devjar
⋮ nodejs-installable-svm-java16-linux-amd64-devjar
⋮ nodejs-installable-svm-java16-windows-amd64-dev.jar
⋮ nodejs-installable-svm-java8-linux-amd64-dev.jar
⋮ nodejs-installable-svm-java8-windows-amd64-dev.jar
⋮ python-installable-svm-java11-darwin-amd64-dev.jar
⋮ python-installable-svm-java11-linux-amd64-devjar
⋮ python-installable-svm-java16-darwin-amd64-dev.jar
⋮ python-installable-svm-java16-linux-amd64-devjar
⋮ python-installable-svm-java8-linux-amd64-devjar
⋮ r-installable-java11-darwin-amd64-dev.jar
⋮ r-installable-java11-linux-amd64-devjar
⋮ r-installable-java16-darwin-amd64-devjar
⋮ r-installable-java16-linux-amd64-devjar
⋮ r-installable-java8-darwin-amd64-devjar
⋮ r-installable-java8-linux-amd64-devjar
⋮ ruby-installable-svm-java11-darwin-amd64-dev.jar
⋮ ruby-installable-svm-java11-linux-aarch64-devjar
⋮ ruby-installable-svm-java11-linux-amd64-devjar
⋮ ruby-installable-svm-java16-darwin-amd64-devjar
⋮ ruby-installable-svm-java16-linux-aarch64-devjar
⋮ ruby-installable-svm-java16-linux-amd64-devjar
⋮ ruby-installable-svm-java8-linux-amd64-devjar
⋮ truffleuby-dev-linux-aarch64.tar.gz
⋮ truffleuby-dev-linux-amd64.tar.gz
⋮ truffleuby-dev-macos-amd64.tar.gz

⋮ wasm-installable-svm-java11-darwin-amd64-dev.jar
⋮ wasm-installable-svm-java11-linux-aarch64-devjar
⋮ wasm-installable-svm-java11-linux-amd64-devjar
⋮ wasm-installable-svm-java11-windows-amd64-dev.jar
⋮ wasm-installable-svm-java16-darwin-amd64-devjar
⋮ wasm-installable-svm-java16-linux-aarch64-devjar
⋮ wasm-installable-svm-java16-linux-amd64-devjar
⋮ wasm-installable-svm-java16-windows-amd64-dev.jar
⋮ wasm-installable-svm-java8-linux-amd64-devjar
⋮ wasm-installable-svm-java8-windows-amd64-dev.jar
⋮ Source code (zip)
⋮ Source code (tar.gz)

Better Performance: Scala

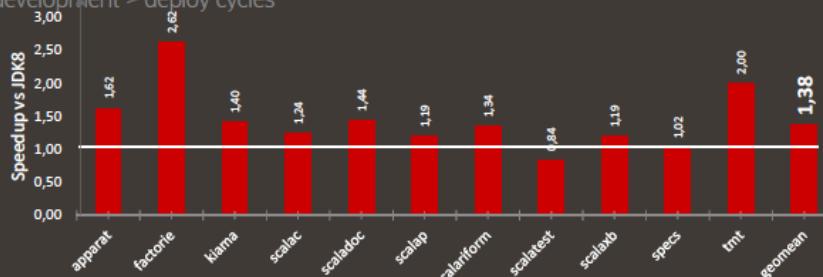
- Implementing GraalVM Enterprise for Scala, customers enjoy even higher performance improvement (average of 38%).
- A 38% performance improvement translates into:

Shorter application response time > better customer experience

Less memory/CPU usage > Less IT spending

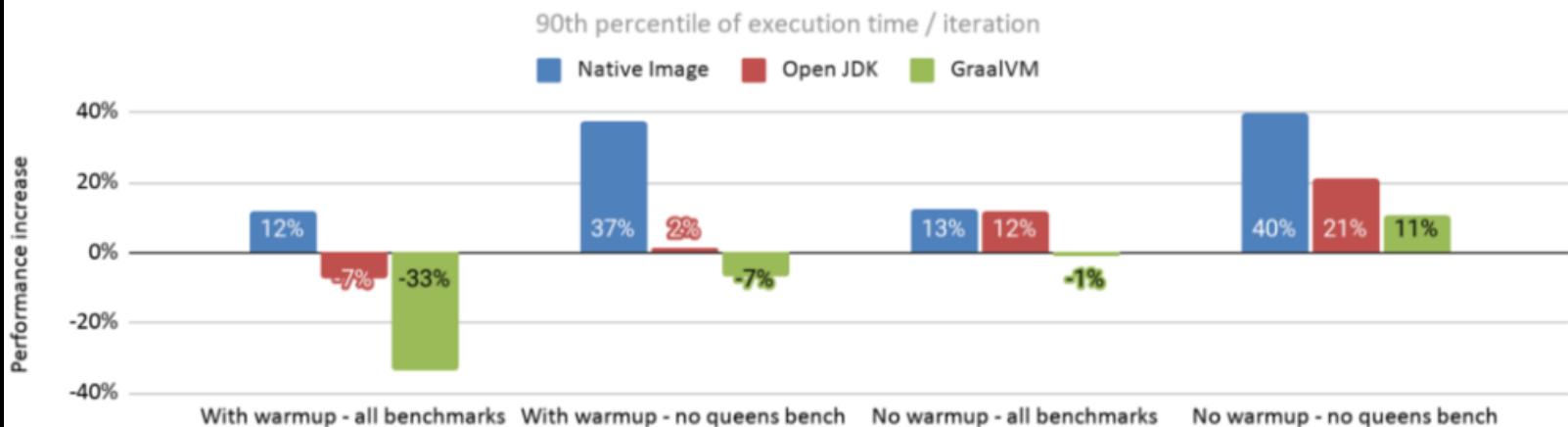
Faster build time - development > deploy cycles

Scalabench



Source: https://www.jug-gr.de/downloads/JDK_14_und_GraalVM_im_Java-%C3%96kosystem_WW.pdf

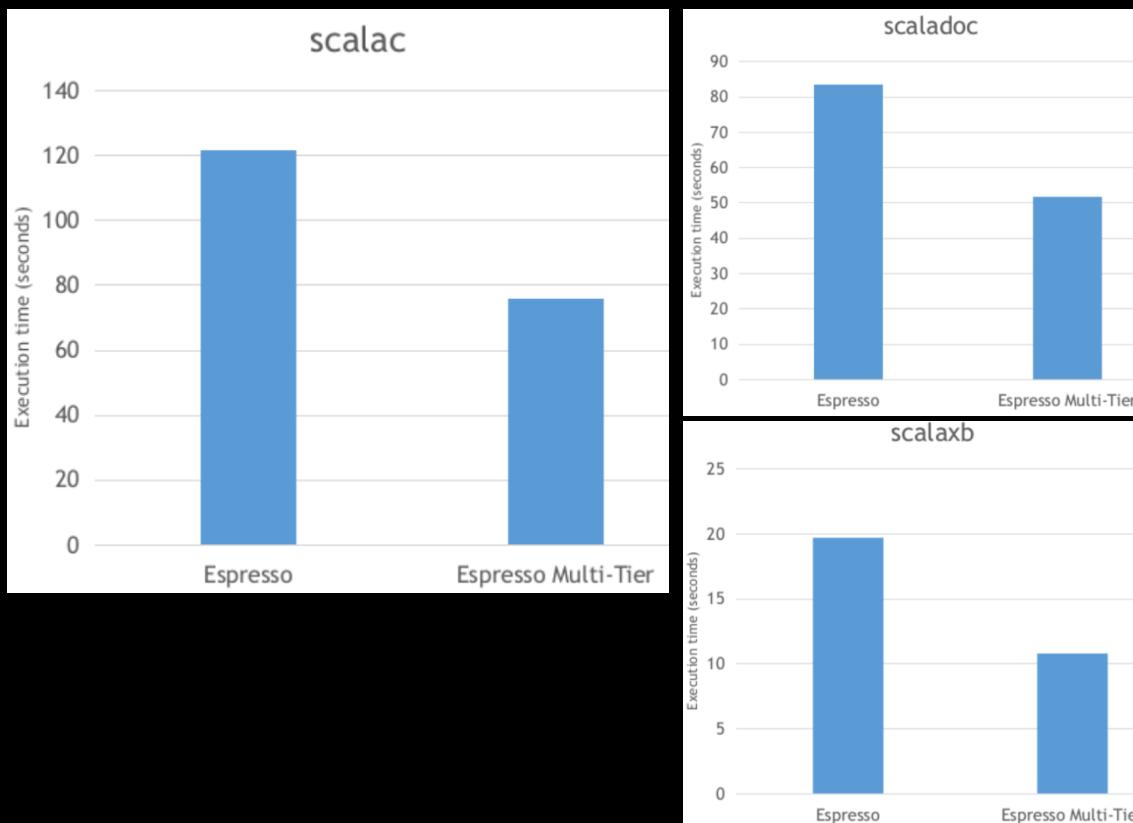
Average Scala Native relative performance difference compared with other technologies



Source: <https://medium.com/virtuslab/revisiting-scala-native-performance-67029089f241>

- <https://medium.com/graalvm/multi-tier-compilation-in-graalvm-5fbc65f92402>

Next, we examine the startup of several benchmarks from the Scalabench suite, on the Espresso engine — the implementation of Java using GraalVM’s Truffle framework. The *scalac* benchmark is particularly large in terms of code-size, and multi-tier compilation reduces the “one-shot” time by ~41%.



2) Acceleration of Chisel/SpinalHDL

2.1 GraalVM

- ○
-

Native Image

■ GraalVM vs Scala Native

- Applies to the whole JVM ecosystem
 - Java, Scala, Kotlin, Clojure, ...
 - Bigger app binaries size
- Bigger community, bigger sponsor: Oracle
- Framework support, related projects
 - Spring, Quarkus, sbt-native-packager
- native-image is early adopter technology
 - Hard to configure it right
- Polyglot features
- Only applies to Scala
 - Possibly more optimised?
- Smaller community, smaller sponsor: Scala Center
- Smaller libraries support
 - sbt-crossproject
- Not updated for a while, now restarted
 - Only supports Scala 2.11
- Interoperability with C

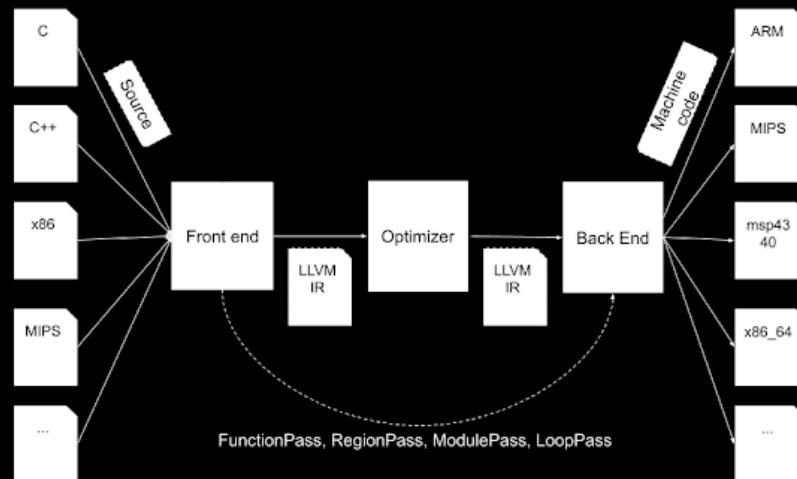
Source: <https://www.slideshare.net/ManfrediGiordano/graalvm-239334355>

2.2 LLVM

- <https://en.wikipedia.org/wiki/LLVM>

LLVM is a set of compiler and toolchain technologies,^[5] which can be used to develop a front end for any programming language and a back end for any instruction set architecture. LLVM is designed around a language-independent intermediate representation (IR) that serves as a portable, high-level assembly language that can be optimized with a variety of transformations over multiple passes.^[6]

- <https://llvm.org>
- <http://clang.llvm.org/>
-



Source: <http://blog.k3170makan.com/2020/04/learning-llvm-i-introduction-to-llvm.html>

- <https://llvm.org/OpenProjects.html>
- <https://developer.arm.com/tools-and-software/open-source-software/developer-tools/llvm-toolchain/architecture-support>

GCC vs LLVM

- a



GPL v3	UIUC, MIT
Front-end: CC1 / CPP	Front-end: Clang
ld.bfd / ld.gold	lld / mclinker
gdb	lldb
as / objdump	MC layer
glibc	llvm-libc?
libstdc++	libc++
libsupc++	libc++abi
libgcc	libcompiler-rt
libgccjit	libLLVMMCJIT
....	ORC JIT, Coroutines, Clangd, libclc, Falcon...

CIRCT

- <https://circt.llvm.org/>

The EDA industry has well-known and widely used proprietary and open source tools. However, these tools are inconsistent, have usability concerns, and were not designed together into a common platform. Furthermore these tools are generally built with Verilog (also VHDL) as the IRs that they interchange. Verilog has well known design issues, and limitations, e.g. suffering from poor location tracking support.

The CIRCT project is an (experimental!) effort looking to apply MLIR and the LLVM development methodology to the domain of hardware design tools. Many of us dream of having reusable infrastructure that is modular, uses library-based design techniques, is more consistent, and builds on the best practices in compiler infrastructure and compiler design techniques.

By working together, we hope that we can build a new center of gravity to draw contributions from the small (but enthusiastic!) community of people who work on open hardware tooling. In turn we hope this will propel open tools forward, enables new higher-level abstractions for hardware design, and perhaps some pieces may even be adopted by proprietary tools in time.

<https://github.com/llvm/circ/blob/main/docs/Charter.md>

<https://github.com/llvm/circ>

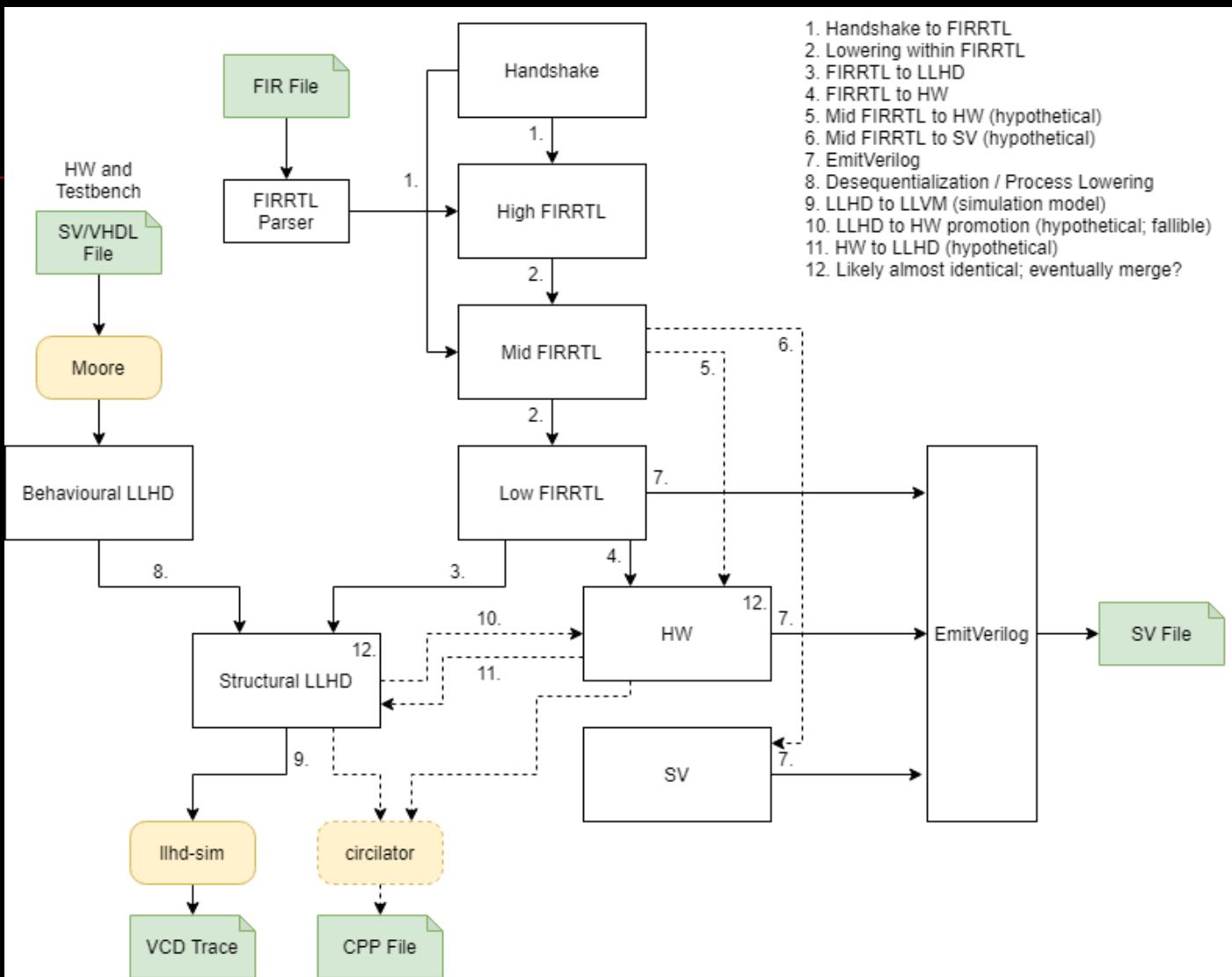


<https://mlir.llvm.org/>

<https://circt.org/perf/>

- <https://github.com/sifive/chisel-circ>
- ...

■ Design



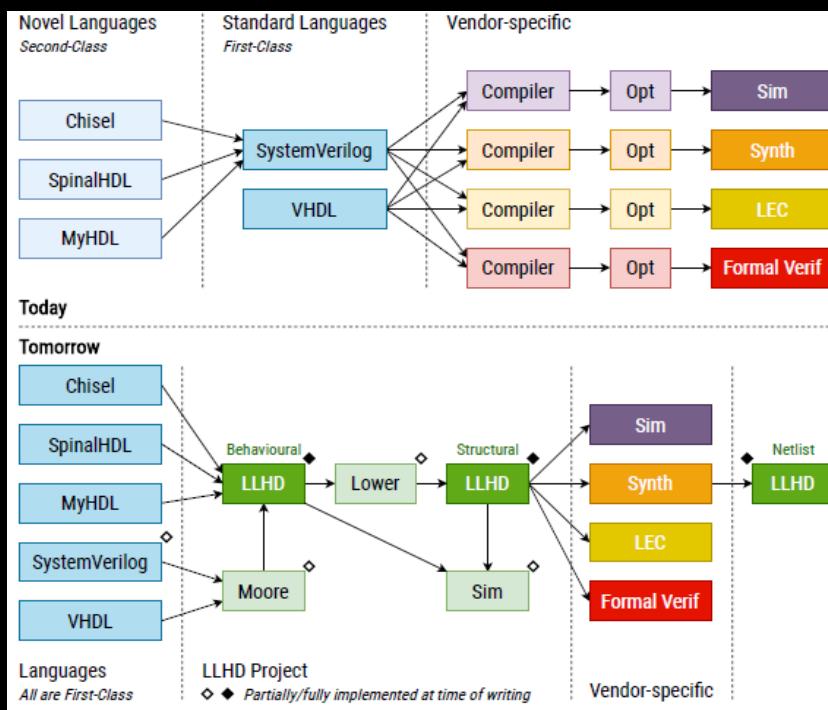
■ LLHD

<http://www.llhd.io>

<https://github.com/fabianschuiki/llhd>

The *Low Level Hardware Description language* is an intermediate representation for digital circuit descriptions, together with an accompanying simulator and SystemVerilog/VHDL compiler.

LLHD separates input languages from EDA tools such as simulators, synthesizers, and placers/routers. This makes writing such tools easier, allows for more rich and complex HDLs, and does not require vendors to agree upon the implementation of a language.



Source: “LLHD: A Multi-level Intermediate Representation for Hardware Description Languages”, Fabian Schuiki, Andreas Kurth, Tobias Grosser, and Luca Benini, PLDI 2020.

NVC

- <https://github.com/nickg/nvc>
A GPLv3 VHDL compiler and simulator aiming for IEEE 1076-2002 compliance.
 - **Trying to replace GHDL with NVC in the future**
-

2.3 Summary

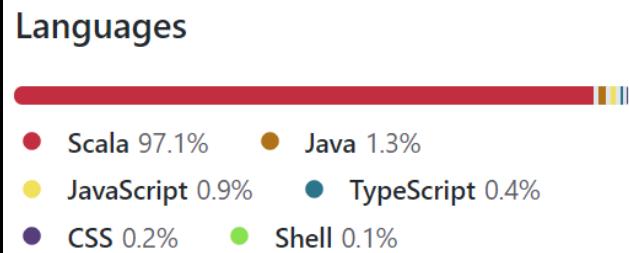
- Migrating from traditional GCC + Monolithic Runtime to LLVM + Polyglot Runtime(GraalVM, .Net, WASM...)
- Gradually adopting modern DevOps such like Container and Container-orchestration system;
e.g.,  
- For EDA in the Cloud(<https://www.arm.com/company/news/2020/12/arm-moves-production-level-electronic-design-automation-to-the-cloud-with-the-help-of-aws>), leverage Quarkus(using a downstream distribution of the GraalVM CE) to further save resource;
- Embracing Scala3 in the future.
- ...

IV. Scala 3

1) Scala 3

1.1 Overview

- <https://dotty.epfl.ch/>
- <https://dotty.epfl.ch/docs/reference/overview.html>
- <https://docs.scala-lang.org/scala3/getting-started.html>
- <https://scala-lang.org/blog/2021/04/08/scala-3-in-sbt.html>
- <https://www.scala-lang.org/blog/2021/05/14/scala3-is-here.html>
- <https://www.infoq.com/news/2021/06/scala-3-overhaul/>
- ...
- <https://github.com/lampepfl/dotty>



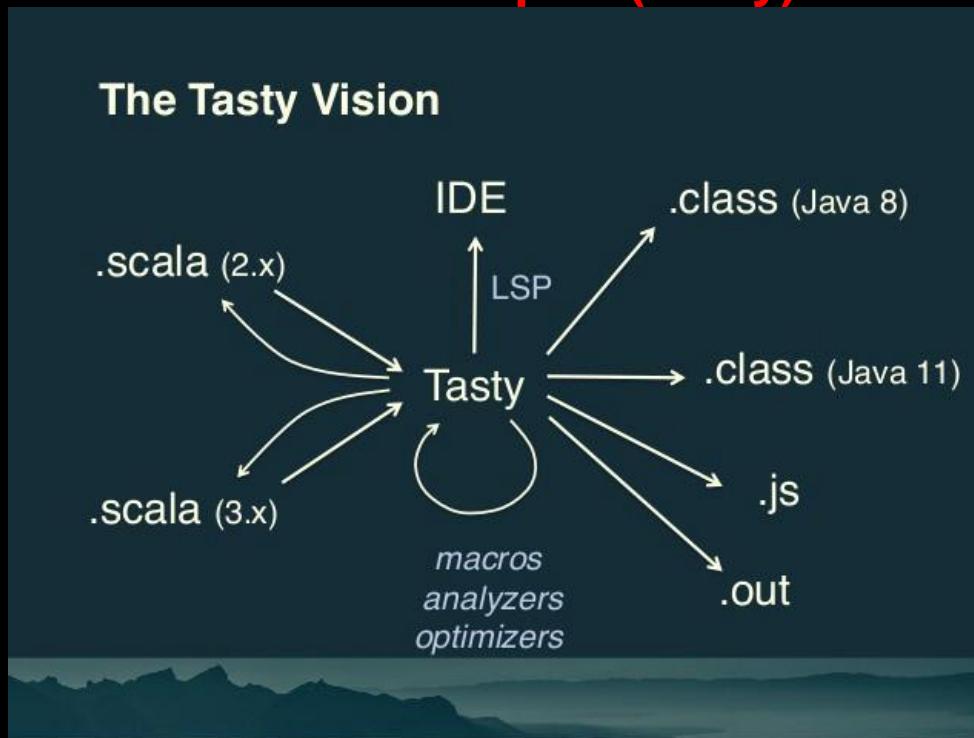
■ Features

<https://docs.scala-lang.org/scala3/book/scala-features.html>

<https://docs.scala-lang.org/scala3/new-in-scala3.html>

<https://betterprogramming.pub/whats-new-in-scala-3-586d69879253?gi=d4f09328b62d>

The New Scala Compiler(Dotty):



TASTy (short for Typed Abstract Syntax Trees):

<https://docs.scala-lang.org/scala3/guides/tasty-overview.html>

1.2 Migrate to Scala3

- <https://docs.scala-lang.org/scala3/guides/>
 - <https://docs.scala-lang.org/scala3/guides/migration/tooling-tour.html>
-

The Scala Compilers

- The Scala 2.13 Compiler
- The Scala 3 Compiler

Build tools

- sbt
- Mill
- Maven

Code editors and IDEs

- Metals
- IntelliJ IDEA

Formatting Tools

- Scalafmt

Migration Tools

- Scalafix
- The Scala 3 Migrate Plugin

Scaladex

- ...

2) Dotty with GraalVM

- o
-

Native Image

- <https://github.com/tabdulradi/hello-scala3-native-image>
 - <https://github.com/scalameta/sbt-native-image>
-

V. Wrap-up

- **A New Golden Age for Computer Architecture**

<https://cacm.acm.org/magazines/2019/2/234352-a-new-golden-age-for-computer-architecture/fulltext>



- Next generation Core/SoC frameworks that base on **eDSL** bring higher productivity to hardware development;
- Open source **EDA** tools are blooming, and corresponding ecosystem is becoming mature;
- Migrating traditional cross-platform development work from X86 hosts to that of ARM meets the trend that **ARM's HW/SW ecosystem** is entering a new outbreak period;
- **GraalVM & Scala 3** are the key points to future of Chisel/SpinalHDL.

Q & A

Thanks!



Reference

Slides/materials from many and varied sources:

- <http://en.wikipedia.org/wiki/>
- <http://www.slideshare.net/>
- https://en.wikipedia.org/wiki/Source-to-source_compiler
- [https://en.wikipedia.org/wiki/Runtime_\(program_lifecycle_phase\)](https://en.wikipedia.org/wiki/Runtime_(program_lifecycle_phase))
- https://en.wikipedia.org/wiki/Just-in-time_compilation
- https://en.wikipedia.org/wiki/Ahead-of-time_compilation
- <https://llvm.org/Users.html>
- <https://github.com/ohenley/awesome-ada>
- <https://www.infoq.com/news/2020/05/java-leyden/>
- <https://www.synopsys.com/designware-ip/processor-solutions/asips-tools/asip-newsletters/asip-eupdate-april-2020.html>
- ...

Backup Slides

(Chisel/SpinalHDL on RPi4 with GCC 10.x)

Chisel

■ Firrtl

```
[info] - should support xorreduce_high_low_one
CompilerAnnotation is deprecated since FIRRTL 1.4.0. Please use 'RunFirrtlTransformAnnotation(new firrtl.MinimumVerilogEmitter)' instead.
input = 0, expected = 0, expr = 0, equal = 1
[info] - should support xorreduce_zero
[info] ScalaTest
[info] Run completed in 7 minutes, 36 seconds.
[info] Total number of tests run: 1796
[info] Suites: completed 212, aborted 0
[info] Tests: succeeded 1796, failed 0, canceled 0, ignored 12, pending 0
[info] All tests passed.
[info] Passed: Total 1796, Failed 0, Errors 0, Passed 1796, Ignored 12
[success] Total time: 466 s (07:46), completed Apr 28, 2021, 4:24:45 PM
```

```
[mydev@MyRPi4-Fedora-1 firrtl-master]$ sbt assembly
[info] Loading settings for project firrtl-master-build from plugins.sbt ...
[info] Loading project definition from /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/firrtl-master/project
[info] Loading settings for project firrtl from build.sbt ...
[info] Set current project to firrtl (in build file:/opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/firrtl-master/)
[warn] Multiple main classes detected. Run 'show discoveredMainClasses' to see the list
[info] Strategy 'discard' was applied to 17 files (Run the task at debug level to see details)
[info] Strategy 'rename' was applied to 4 files (Run the task at debug level to see details)
[success] Total time: 46 s, completed Apr 28, 2021, 4:29:22 PM

https://repo1.maven.org/maven2/org/ow2/asm/asm-commons/6.2.1/asm-commons-6.2.1.jar
100.0% [#####] 77.1 kB (147.8 kB / s)
https://repo1.maven.org/maven2/ognl/ognl/3.1.12/ognl-3.1.12.jar
100.0% [#####] 230.1 kB (466.8 kB / s)
https://repo1.maven.org/maven2/org/javassist/javassist/3.20.0-GA/javassist-3.20.0-GA.jar
100.0% [#####] 733.0 kB (778.9 kB / s)
https://repo1.maven.org/maven2/info/picocli/picocli/4.0.4/picocli-4.0.4.jar
100.0% [#####] 337.5 kB (169.9 kB / s)
[info] Fetched artifacts of
[warn] There may be incompatibilities among your library dependencies; run 'evicted' to see detailed eviction warnings.
[warn] There may be incompatibilities among your library dependencies; run 'evicted' to see detailed eviction warnings.
[info] Main Scala API documentation to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/firrtl-master/target/scala-2.12/unidoc...
model contains 1186 documentable templates
[info] Main Scala API documentation successful.
[info] :: delivering :: edu.berkeley.cs#firrtl_2.12;1.5-SNAPSHOT :: 1.5-SNAPSHOT :: integration :: Wed Apr 28 16:41:57 PDT 2021
[info] delivering ivy file to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/firrtl-master/target/scala-2.12/ivy-1.5-SNAPSHOT.xml
[info] published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/poms/firrtl_2.12.pom
[info] published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/jars/firrtl_2.12.jar
[info] published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/srcs/firrtl_2.12-sources.jar
[info] published firrtl_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/docs/firrtl_2.12-javadoc.jar
[info] published ivy to /home/mydev/.ivy2/local/edu.berkeley.cs/firrtl_2.12/1.5-SNAPSHOT/ivys/ivy.xml
[success] Total time: 473 s (07:53), completed Apr 28, 2021, 4:41:58 PM
[mydev@MyRPi4-Fedora-1 firrtl-master]$
```

Treadle

```
[warn] /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/src/main/scala/treadle/vcd/diff/VcdComparator.scala:129:44: method toIterator  
in trait IterableOnceOps is deprecated (since 2.13.0): Use .iterator instead of .toIterator  
[warn]     val i2 = wireList2.keys.toSeq.sorted.toIterator  
[warn]                                         ^  
[warn] /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/src/main/scala/treadle/TreadleRepl.scala:360:25: match may not be exhaustive.  
[warn] It would fail on the following input: (None, Some(_))  
[warn]     getTwoArgs(  
[warn]     ^  
[warn] /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/src/main/scala/treadle/repl/ReplVcdController.scala:97:9: match may not be ex-  
haustive.  
[warn] It would fail on the following inputs: List((x: String forSome x not in "step")), Nil  
[warn]     tail match {  
[warn]     ^  
[warn] 11 warnings found  
[success] Total time: 687 s (11:27), completed Apr 28, 2021, 5:11:48 PM  
[mydev@MyRPi4-Fedora-1 treadle-master]$
```

```
[info] MemoryUsageSpec:  
[info] - chirrtl mems should parse and run ok  
[info] - memory primitives should run this circuit  
[info] - read-write memory should work with this simple example  
[info] - this is a more complex circuit  
[info] - basic memory latency read 0 write 1  
[info] - basic memory with varying latencies  
[info] - memory can be initialized at startup  
[info] - write port: en and masks should be respected  
[info] - read port: enable should be pipelined correctly  
[info] - SyncReadMem write collision behaviors should work  
[info] Run completed in 2 minutes, 13 seconds.  
[info] Total number of tests run: 235  
[info] Suites: completed 79, aborted 0  
[info] Tests: succeeded 235, failed 0, canceled 0, ignored 2, pending 0  
[info] All tests passed.  
[success] Total time: 280 s (04:40), completed Apr 28, 2021, 5:31:37 PM  
[mydev@MyRPi4-Fedora-1 treadle-master]$
```

```
[mydev@MyRPi4-Fedora-1 treadle-master]$ tree utils  
utils  
└── bin  
    └── treadle  
        └── treadle.jar
```

```
1 directory, 2 files  
[mydev@MyRPi4-Fedora-1 treadle-master]$  
[mydev@MyRPi4-Fedora-1 treadle-master]$ sbt publishLocal  
[info] Loading settings for project treadle-master-build from plugins.sbt ...  
[info] Loading project definition from /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/project  
[info] Loading settings for project treadle from build.sbt ...  
[info] Set current project to treadle (in build file:/opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/)  
[info] Wrote /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/target/scaladoc-2.13/treadle_2.13-1.5-SNAPSHOT.pom  
[info] Main Scala API documentation to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/target/scaladoc-2.13/api...  
[warn] Multiple main classes detected. Run 'show discoveredMainClasses' to see the list  
[info] Main Scala API documentation successful.  
[info] :: delivering :: edu.berkeley.cs#treadle 2.13;1.5-SNAPSHOT :: integration :: Wed Apr 28 17:43:59 PDT 2021  
[info] delivering ivy file to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/target/scaladoc-2.13/ivy-1.5-SNAPSHOT.xml  
[info] published treadle_2.13 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.13/1.5-SNAPSHOT/poms/treadle_2.13.pom  
[info] published treadle_2.13 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.13/1.5-SNAPSHOT/jars/treadle_2.13.jar  
[info] published treadle_2.13 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.13/1.5-SNAPSHOT/srcs/treadle_2.13-sources.jar  
[info] published treadle_2.13 to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.13/1.5-SNAPSHOT/docs/treadle_2.13-javadoc.jar  
[info] published ivy to /home/mydev/.ivy2/local/edu.berkeley.cs/treadle_2.13/1.5-SNAPSHOT/ivys/ivy.xml  
[success] Total time: 115 s (01:55), completed Apr 28, 2021, 5:43:59 PM  
[mydev@MyRPi4-Fedora-1 treadle-master]$
```

```
[mydev@MyRPi4-Fedora-1 treadle-master]$ sbt assembly  
[info] Loading settings for project treadle-master-build from plugins.sbt ...  
[info] Loading project definition from /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/project  
[info] Loading settings for project treadle from build.sbt ...  
[info] Set current project to treadle (in build file:/opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/treadle-master/)  
[warn] Multiple main classes detected. Run 'show discoveredMainClasses' to see the list  
[warn] Negative time  
[warn] Negative time
```

Chisel3

```
[warn] /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/chisel3-master/src/main/scala/chisel3/util/experimental/LoadMemoryTransform.scala:208:33: object LowForm in package firrtl is deprecated (since FIRRTL 1.3): Mix-in the DependencyAPIMigration trait into your Transform and specify its Dependency API dependencies. See: https://bit.ly/2Voppre
[warn]   def outputForm: CircuitForm = LowForm
[warn]
[warn] 33 warnings found
[success] Total time: 230 s (03:50), completed Apr 28, 2021, 5:51:22 PM
[mydev@MyRPi4-Fedora-1 chisel3-master]$
```

```
[info] - It should be possible to bulk connect a Vec and a Seq
Elaborating design...
[error] Vec.scala:263: Vec and Seq being bulk connected have different lengths! in class chiselTests.VecSpec$$anon$10
[error] There were 1 error(s) during hardware elaboration.
[info] - Bulk connecting a Vec and Seq of different sizes should report a ChiselException
Elaborating design...
Done elaborating.
[info] - It should be possible to initialize a Vec with DontCare
Elaborating design...
[info] - Indexing a Chisel type Vec by a hardware type should give a sane error message
[info] ScalaTest
[info] Run completed in 45 minutes, 25 seconds.
[info] Total number of tests run: 779
[info] Suites: completed 139, aborted 0
[info] Tests: succeeded 779, failed 0, canceled 0, ignored 21, pending 0
[info] All tests passed.
[info] Passed: Total 779, Failed 0, Errors 0, Passed 779, Ignored 21
[success] Total time: 3043 s (50:43), completed Apr 28, 2021, 6:49:18 PM
[mydev@MyRPi4-Fedora-1 chisel3-master]$
```

```
[warn] 8 warnings found
[info] Main Scala API documentation successful.
[warn] 16 warnings found
[info] Main Scala API documentation successful.
[info] :: delivering :: edu.berkeley.cs#chisel3_2.12;3.5-SNAPSHOT :: 3.5-SNAPSHOT :: integration :: Wed Apr 28 19:08:53 PDT 2021
[info] delivering ivy file to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/chisel3-master/target/scala-2.12/ivy-3.5-SNAPSHOT.xml
[info] published chisel3_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3_2.12/3.5-SNAPSHOT/poms/chisel3_2.12.pom
[info] published chisel3_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3_2.12/3.5-SNAPSHOT/jars/chisel3_2.12.jar
[info] published chisel3_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3_2.12/3.5-SNAPSHOT/srcs/chisel3_2.12-sources.jar
[info] published chisel3_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3_2.12/3.5-SNAPSHOT/docs/chisel3_2.12-javadoc.jar
[info] published ivy to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3_2.12/3.5-SNAPSHOT/ivys/ivy.xml
[info] :: delivering :: edu.berkeley.cs#chisel3-core_2.12;3.5-SNAPSHOT :: 3.5-SNAPSHOT :: integration :: Wed Apr 28 19:08:54 PDT 2021
[info] delivering ivy file to /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/Chisel/chisel3-master/core/target/scala-2.12/ivy-3.5-SNAPSHOT.xml
[info] published chisel3-core_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3-core_2.12/3.5-SNAPSHOT/poms/chisel3-core_2.12.pom
[info] published chisel3-core_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3-core_2.12/3.5-SNAPSHOT/jars/chisel3-core_2.12.jar
[info] published chisel3-core_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3-core_2.12/3.5-SNAPSHOT/srcs/chisel3-core_2.12-sources.jar
[info] published chisel3-core_2.12 to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3-core_2.12/3.5-SNAPSHOT/docs/chisel3-core_2.12-javadoc.jar
[info] published ivy to /home/mydev/.ivy2/local/edu.berkeley.cs/chisel3-core_2.12/3.5-SNAPSHOT/ivys/ivy.xml
[success] Total time: 138 s (02:18), completed Apr 28, 2021, 7:08:54 PM
[mydev@MyRPi4-Fedora-1 chisel3-master]$
```

Rocket HelloWorld for Rocket emulator <https://www.programmersought.com/article/7539888504/>

//Add HelloWorld to riscv-tests

<https://github.com/riscv/riscv-tests>

[https://github.com/riscv/riscv-tests/pull/335 \(not merged\)](https://github.com/riscv/riscv-tests/pull/335 (not merged))

```
[mydev@fedora riscv-tests-master]$ git status
On branch master
Your branch is up to date with 'origin/master'.

Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   benchmarks/Makefile

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    benchmarks/helloworld/
```

```
[mydev@fedora riscv-tests-master]$ git diff
diff --git a/benchmarks/Makefile b/benchmarks/Makefile
index c9469e2..85005fc 100644
--- a/benchmarks/Makefile
+++ b/benchmarks/Makefile
@@ -30,6 +30,7 @@ bmarks = \
        mt-vadd \
        mt-matmul \
        pmp \
+       helloworld \
```

```
[mydev@MyRPi4-Fedora-1 emulator]$ ./emulator-freechips.rocketchip.system-freechips.rocketchip.system.DefaultConfig-debug pk helloworld
This emulator compiled with JTAG Remote Bitbang client. To enable, use +jtag_rbb_enable=1.
```

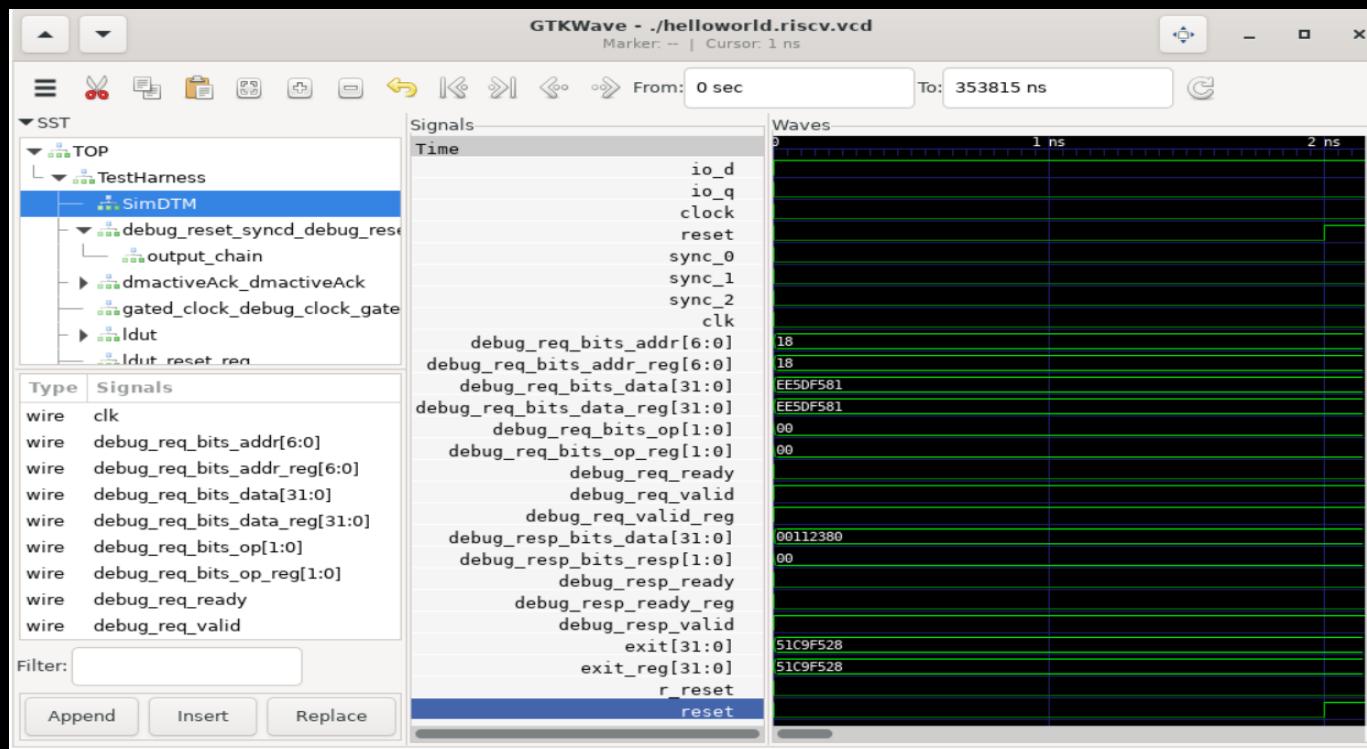
```
Listening on port 35659
Hello world!
```

```
[mydev@MyRPi4-Fedora-1 emulator]$ ./emulator-freechips.rocketchip.system-freechips.rocketchip.system.DefaultConfig-debug helloworld.riscv
This emulator compiled with JTAG Remote Bitbang client. To enable, use +jtag_rbb_enable=1.
Listening on port 44507
Hello world!
```

```
[mydev@MyRPi4-Fedora-1 emulator]$ ./emulator-freechips.rocketchip.system-freechips.rocketchip.system.DefaultConfig-debug +max-cycles=100000 +verbose helloworld.riscv 2> helloworld.riscv.out
Hello world!
```

```
Using random seed 1620530695
This emulator compiled with JTAG Remote Bitbang client. To enable, use +jtag_rbb_enable=1.
Listening on port 41253
C0:      19 [1] pc=[0000000000010040] W[r 0=0000000000000208][1] R[r 0=0000000000000000] R[r 0=0000000000000000] inst=[7c105073] DASM
(7c105073)
C0:      24 [1] pc=[0000000000010044] W[r10=0000000000000000][1] R[r 0=0000000000000000] R[r 0=0000000000000000] inst=[f1402573] DASM
(f1402573)
C0:      25 [1] pc=[0000000000010048] W[r11=0000000000010048][1] R[r 0=0000000000000000] R[r 0=0000000000000000] inst=[00000597] DASM
(00000597)
C0:      26 [1] pc=[000000000001004c] W[r11=0000000000010080][1] R[r11=0000000000010048] R[r 0=0000000000000000] inst=[03858593] DASM
(03858593)
C0:      27 [1] pc=[0000000000010050] W[r 0=ede659d0adc3e9fd][1] R[r 0=0000000000000000] R[r 0=0000000000000000] inst=[30405073] DASM
```

```
[mydev@MyRPi4-Fedora-1 emulator]$ ./emulator-freechips.rocketchip.system-freechips.rocketchip.system.DefaultConfig-debug +max-cycles=100
0000 --vcd=helloworld.riscv.vcd helloworld.riscv
This emulator compiled with JTAG Remote Bitbang client. To enable, use +jtag_rbb_enable=1.
Listening on port 38365
Hello world!
[mydev@MyRPi4-Fedora-1 emulator]$ ll
total 2200488
drwxr-xr-x. 1 mydev mydev      466 May  5 11:46 .
drwxr-xr-x. 1 mydev mydev     888 May  1 23:22 ..
-rwxr-xr-x. 1 mydev mydev 13142576 May  4 14:12 emulator-freechips.rocketchip.system-freechips.rocketchip.system.DefaultConfig-debug*
drwxr-xr-x. 1 mydev mydev    1892 May  4 13:42 generated-src/
drwxr-xr-x. 1 mydev mydev     82 May  4 14:02 generated-src-debug/
-rw-r--r--. 1 mydev mydev     99 May  1 22:39 .gitignore
-rwxr-xr-x. 1 mydev mydev 18968 May  5 10:07 helloworld*
-rw-r--r--. 1 mydev mydev 17224 May  5 11:09 helloworld.riscv
-rw-r--r--. 1 mydev mydev 7806183 May  5 11:27 helloworld.riscv.out
-rw-r--r--. 1 mydev mydev 2232331462 May  5 11:57 helloworld.riscv.vcd
-rw-r--r--. 1 mydev mydev   2183 May  1 22:39 Makefile
-rw-r--r--. 1 mydev mydev    4204 May  1 22:39 Makefrag-verilator
drwxr-xr-x. 1 mydev mydev     6136 May  4 14:12 output/
drwxr-xr-x. 1 mydev mydev      64 May  4 14:02 verilator/
[mydev@MyRPi4-Fedora-1 emulator]$
```



SpinalHDL

■ SpinalHDL

```
[mydev@MyRPi4-Fedora-1 SpinalHDL-dev]$ sbt compile
[info] Loading settings for project spinalhdl-dev-build from plugin.sbt ...
[info] Loading project definition from /opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/SpinalHDL/SpinalHDL-dev/project
[info] Loading settings for project all from build.sbt ...
[info] Set current project to SpinalHDL-all (in build file:/opt/MyWorkSpace/MyProjs/HW-EDA/RISC-V/SpinalHDL/SpinalHDL-dev/)
[info] Executing in batch mode. For better performance use sbt's shell
[success] Total time: 66 s (01:06), completed Apr 29, 2021, 8:28:38 PM
[mydev@MyRPi4-Fedora-1 SpinalHDL-dev]$
```

two blocked test cases:

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
155936	mydev	20	0	274328	47280	10444	R	97.4	0.6	298:57.48	axi4crossbarter
124350	mydev	20	0	251172	24092	8868	R	96.1	0.3	321:31.65	axi4crossbarter
247864	mydev	20	0	2271760	2.0g	7936	R	96.1	25.5	274:19.95	streamtester2
122481	mydev	20	0	6098972	985408	28480	S	44.6	12.3	94:12.38	java
2655	root	20	0	1162120	94524	45096	S	8.9	1.2	94:50.84	Xorg
374018	mydev	30	10	14304	5116	4592	S	1.3	0.1	0:06.16	bouboule
376649	mydev	20	0	235140	4300	3584	R	1.0	0.1	0:00.19	top

```
[mydev@MyRPi4-Fedora-1 Materials]$ ps aux |grep -i axi4crossbarter
mydev 110682 0.0 0.0 11368 4712 pts/4 S+ May02 0:00 /usr/bin/ghdl -r --workdir=sim_build --work=work axi4crossbarter
-vpi=/usr/lib64/python3.9/site-packages/cocotb/libs/libcocotbvp_ghdl.so --ieee-asserts=disable
mydev 110683 73.2 0.3 251460 25364 pts/4 R+ May02 119:47 ./axi4crossbarter --vpi=/usr/lib64/python3.9/site-packages/cocotb
libs/libcocotbvp_ghdl.so --ieee-asserts=disable
mydev 133323 0.0 0.0 11368 4744 pts/4 S+ 00:11 0:00 /usr/bin/ghdl -r --workdir=sim_build --work=work axi4crossbarter2
-vpi=/usr/lib64/python3.9/site-packages/cocotb/libs/libcocotbvp_ghdl.so --ieee-asserts=disable
mydev 133324 57.8 0.5 274328 48116 pts/4 R+ 00:11 66:59 ./axi4crossbarter2 --vpi=/usr/lib64/python3.9/site-packages/cocotb
/libs/libcocotbvp_ghdl.so --ieee-asserts=disable
mydev 164274 0.0 0.0 221368 2288 pts/2 R+ 02:07 0:00 grep --color=auto -i axi4crossbarter
[mydev@MyRPi4-Fedora-1 Materials]$ ps aux |grep -i streamtester2
mydev 153218 0.0 0.0 11368 4744 pts/4 S+ 00:52 0:00 /usr/bin/ghdl -r --workdir=sim_build --work=work streamtester2 --vpi
/usr/lib64/python3.9/site-packages/cocotb/libs/libcocotbvp_ghdl.so --ieee-asserts=disable
mydev 153219 49.2 4.7 604300 378116 pts/4 R+ 00:52 37:19 ./streamtester2 --vpi=/usr/lib64/python3.9/site-packages/cocotb/libs
libcocotbvp_ghdl.so --ieee-asserts=disable
mydev 164381 0.0 0.0 221364 804 pts/2 R+ 02:08 0:00 grep --color=auto -i streamtester2
```

<https://github.com/SpinalHDL/SpinalHDL/issues/414>

jiegec commented on 4 May • edited

Contributor

and retested it on my Raspberry Pi, but it does not work, Axi4CrossbarTester and StreamTester seems run without endless...

I have reproduced this problem.

EDIT: This problem exists even on x86_64 platform(ghdl v1.0.0 on Linux). It's probably something wrong with ghdl integration.

EDIT: ghdl v0.36 on Linux (debian buster) works.

VexRiscv