



RISC-V Introduction Benchmark Call

Rick O'Connor

rickoco@riscv.org

<http://www.riscv.org>



Outline

- RISC-V ISA Overview
- RISC-V Foundation Overview & Growth
- RISC-V Use Case Examples
 - NVIDIA and Western Digital
- Summary



RISC-V Background

- In 2010, after many years and many projects using MIPS, SPARC, and x86 as basis of research, it was time for the Computer Science team at UC Berkeley to look at what ISAs to use for their next set of projects
- Obvious choices: x86 and ARM
 - x86 impossible – too complex, IP issues
 - ARM mostly impossible – complex, IP issues
- So UC Berkeley started “3-month project” during the summer of 2010 to develop their own clean-slate ISA



RISC-V Background (cont'd)

- Four years later, in May of 2014, UC Berkeley released frozen base user spec
 - many tapeouts and several research publications along the way
- The name RISC-V (pronounced *risk-five*), was chosen to represent the fifth major RISC ISA design effort at UC Berkeley
 - RISC-I, RISC-II, SOAR, and SPUR were the first four with the original RISC-I publications dating back to 1981
- In August 2015, articles of incorporation were filed to create a non-profit RISC-V Foundation to govern the ISA



Why Instruction Set Architectures matter

- Why are 99%+ of laptops/desktops/servers based on AMD x86-64 ISA (over 95%+ built by Intel)?
- Why are 99%+ of mobile phones + tablets based on ARM v7/v8 ISA?
- Why can't Intel sell mobile chips?
- Why can't ARM vendors sell servers?
- How can IBM still sell mainframes?
- ISA is most important interface in a computer system
 - Where software meets hardware

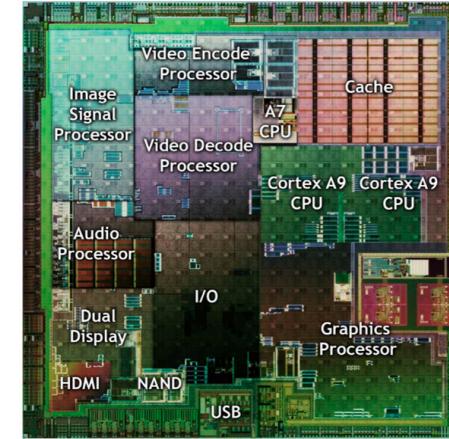
Open Software / Standards Work!

<i>Field</i>	<i>Standard</i>	<i>Free, Open Impl.</i>	<i>Proprietary Impl.</i>
Networking	Ethernet, TCP/IP	Many	Many
OS	Posix	Linux, FreeBSD	M/S Windows
Compilers	C	gcc, LLVM	Intel icc, ARMcc
Databases	SQL	MySQL, PostgresSQL	Oracle 12C, M/S DB2
Graphics	OpenGL	Mesa3D	M/S DirectX
ISA	???????	--	x86, ARM

- Why are there no successful free & open ISA standards and free & open implementations, like other fields?

Most CPU chips are SoCs with many ISAs

- Applications processor (usually ARM)
- Graphics processors
- Image processors
- Radio DSPs
- Audio DSPs
- Security processors
- Power-management processor
-



NVIDIA Tegra SoC

- Apps processor ISA too large for base accelerator ISA
- IP bought from different places, each proprietary ISA
- Home-grown ISA cores
- Over a dozen ISAs on some SoCs – each with unique software stack

Do we need all these different ISAs?

Must they be proprietary?

What if there was one free and open ISA everyone could use for everything?



What's Different about RISC-V?

- ***Simple***
 - Far smaller than other commercial ISAs
- ***Clean-slate design***
 - Clear separation between user and privileged ISA
 - Avoids μarchitecture or technology-dependent features
- A ***modular*** ISA
 - Small standard base ISA
 - Multiple standard extensions
- Designed for ***extensibility/specialization***
 - Variable-length instruction encoding
 - Vast opcode space available for instruction-set extensions
- ***Stable***
 - Base and standard extensions are frozen
 - Additions via optional extensions, not new versions



RISC-V Base Plus Standard Extensions

- Four base integer ISAs
 - RV32C, RV32I, RV64I, RV128I
 - RV32C is 16-register subset of RV32I
 - Only <50 hardware instructions needed for base
- Standard extensions
 - M: Integer multiply/divide
 - A: Atomic memory operations (AMOs + LR/SC)
 - F: Single-precision floating-point
 - D: Double-precision floating-point
 - G = IMAFD, “General-purpose” ISA
 - Q: Quad-precision floating-point
- All the above are a fairly standard RISC encoding in a fixed 32-bit instruction format
- Above user-level ISA components frozen in 2014
 - Supported forever after

RV32I



①

Base Integer Instructions (32 64 128)					
Category	Name	Fmt	RV(32 64 128)I	Base	
Loads	Load Byte	I	LB	rd,rs1,imm	
	Load Halfword	I	LH	rd,rs1,imm	
	Load Word	I	L(W D Q)	rd,rs1,imm	
	Load Byte Unsigned	I	LBU	rd,rs1,imm	
	Load Half Unsigned	I	L(H W D)U	rd,rs1,imm	
Stores	Store Byte	S	SB	rs1,rs2,imm	
	Store Halfword	S	SH	rs1,rs2,imm	
	Store Word	S	S(W D Q)	rs1,rs2,imm	
Shifts	Shift Left	R	SLL({W D})	rd,rs1,rs2	
	Shift Left Immediate	I	SLLI({W D})	rd,rs1,shamt	
	Shift Right	R	SRL({W D})	rd,rs1,rs2	
	Shift Right Immediate	I	SRLI({W D})	rd,rs1,shamt	
	Shift Right Arithmetic	R	SRA({W D})	rd,rs1,rs2	
	Shift Right Arith Imm	I	SRAI({W D})	rd,rs1,shamt	
Arithmetic	ADD	R	ADD({W D})	rd,rs1,rs2	
	ADD Immediate	I	ADDI({W D})	rd,rs1,imm	
	SUBtract	R	SUB({W D})	rd,rs1,rs2	
	Load Upper Imm	U	LUI	rd,imm	
	Add Upper Imm to PC	U	AUIPC	rd,imm	
	XOR	R	XOR	rd,rs1,rs2	
Logical	XOR Immediate	I	XORI	rd,rs1,imm	
	OR	R	OR	rd,rs1,rs2	
	OR Immediate	I	ORI	rd,rs1,imm	
	AND	R	AND	rd,rs1,rs2	
Compare	AND Immediate	I	ANDI	rd,rs1,imm	
	Set <	R	SLT	rd,rs1,rs2	
	Set < Immediate	I	SLTI	rd,rs1,imm	
	Set < Unsigned	R	SLTU	rd,rs1,rs2	
Branches	Set < Imm Unsigned	I	SLTIU	rd,rs1,imm	
	Branch =	SB	BEQ	rs1,rs2,imm	
	Branch ≠	SB	BNE	rs1,rs2,imm	
	Branch <	SB	BLT	rs1,rs2,imm	
Jump & Link	Branch ≥	SB	BGE	rs1,rs2,imm	
	Branch < Unsigned	SB	BLTU	rs1,rs2,imm	
	Branch ≥ Unsigned	SB	BGEU	rs1,rs2,imm	
	J&L	UJ	JAL	rd,imm	
Sync	Jump & Link Register	I	JALR	rd,rs1,imm	
	Synch thread	I	FENCE		
System	Synch Instr & Data	I	FENCE.I		
	System CALL	I	SCALL		
System	System BREAK	I	SBREAK		
	Read CYCLE	I	RDCYCLE	rd	
Counters	Read CYCLE upper Half	I	RDCYCLES	rd	
	Read TIME	I	RDTIME	rd	
	Read TIME upper Half	I	RDTIMEH	rd	
	Read INSTR RETired	I	RDINSTRET	rd	
	Read INSTR upper Half	I	RDINSTRETH	rd	

②

+14
Privileged

③ RISC-V Reference Card ④

+ 8 for M

+ 34
for F, D, Q

+ 46 for C

+ 11 for A

32-bit Instruction Formats

R	31	30	25 24	21	20	19	15 14	12 11	8	7	6	0
I	funct7			rs2		rs1	funct3		rd		opcode	
				imm[11:0]					rd		opcode	
S	imm[11:5]				rs2	rs1	funct3	imm[4:0]		rd		opcode
SB	imm[12] imm[10:5]				rs2	rs1	funct3	imm[4:1] imm[11]		rd		opcode
U					imm[31:12]				rd		opcode	
UJ	imm[20] imm[10:1]			imm[11]		imm[19:12]			rd		opcode	

RV32I / RV64I / RV128I + M, A, F, D, Q, C

① RISC-V Reference Card											
Base Integer Instructions (32 64 128)						RV Privileged Instructions (32 64 128)					
Category	Name	Fmt	RV(32 64 128) Base	Category	Name	Fmt	RV mnemonic	Category	Name	Fmt	RV{F D Q} (HP SP,DP,QP)
Loads	Load Byte	I	LB rd,rs1,imm	CSR Access	Atomic R/W	R	CSRWR rd,csr,rs1	Load	Load	I	FL(W,D,Q) rd,rs1,imm
	Load Halfword	I	LH rd,rs1,imm		Atomic Read & Set Bit	R	CSRRS rd,csr,rs1		Store	S	FS(W,D,Q) rs1,rs2,imm
	Load Word	I	L(W D Q) rd,rs1,imm		Atomic Read & Clear Bit	R	CSRRC rd,csr,rs1			R	FADD.(S D Q) rd,rs1,rs2
	Load Byte Unsigned	I	LBU rd,rs1,imm		Atomic R/W Imm	R	CSRRWI rd,csr,imm			R	FSUB.(S D Q) rd,rs1,rs2
	Load Half Unsigned	I	L(H W D U) rd,rs1,imm		Atomic Read & Set Bit Imm	R	CSRSRI rd,csr,imm			R	FMUL.(S D Q) rd,rs1,rs2
Stores	Store Byte	S	SB rs1,rs2,imm		Atomic Read & Clear Bit Imm	R	CSRRCI rd,csr,imm			R	FDIV.(S D Q) rd,rs1,rs2
	Store Halfword	S	SH rs1,rs2,imm							R	FSQRT.(S D Q) rd,rs1
	Store Word	S	S(W D Q) rs1,rs2,imm							R	FSQRT.(S D Q) rd,rs1,rs2,rs3
Shifts	Shift Left	R	SLL.(W D) rd,rs1,rs2							R	FMADD.(S D Q) rd,rs1,rs2,rs3
	Shift Left Immediate	I	SLLI.(W D) rd,rs1,shamt							R	FMSUB.(S D Q) rd,rs1,rs2,rs3
	Shift Right	R	SRRL.(W D) rd,rs1,rs2							R	FMNSUB.(S D Q) rd,rs1,rs2,rs3
	Shift Right Immediate	I	SRRLI.(W D) rd,rs1,shamt							R	FMADD.(S D Q) rd,rs1,rs2,rs3
	Shift Right Arithmetic	R	SRAI.(W D) rd,rs1,rs2							R	FMNSUB.(S D Q) rd,rs1,rs2,rs3
	Shift Right Arith Imm	I	SRALI.(W D) rd,rs1,shamt							R	FMADD.(S D Q) rd,rs1,rs2,rs3
Arithmetic	ADD	R	ADD.(W D) rd,rs1,rs2							R	FSGNJN.(S D Q) rd,rs1,rs2
	ADD Immediate	I	ADDI.(W D) rd,rs1,imm							R	FSGNJN.(S D Q) rd,rs1,rs2
	SUBtract	R	SUB.(W D) rd,rs1,rs2							R	FSGNJX.(S D Q) rd,rs1,rs2
	Load Upper Imm	R	LUI rd,imm							R	FSGNJX.(S D Q) rd,rs1,rs2
	Add Upper Imm to PC	R	AUIPC rd,imm							R	FMIN.(S D Q) rd,rs1,rs2
Logical	XOR	R	XOR rd,rs1,rs2							R	FMIN.(S D Q) rd,rs1,rs2
	XOR Immediate	I	XORI rd,rs1,imm							R	FMAX.(S D Q) rd,rs1,rs2
	OR	R	OR rd,rs1,rs2							R	FMAX.(S D Q) rd,rs1,rs2
	OR Immediate	I	ORI rd,rs1,imm							R	FEQ.(S D Q) rd,rs1,rs2
	AND	R	AND rd,rs1,rs2							R	FLT.(S D Q) rd,rs1,rs2
	AND Immediate	I	ANDI rd,rs1,imm							R	FLT.(S D Q) rd,rs1,rs2
Compare	Set <	R	SLT rd,rs1,rs2							R	FLPE.(S D Q) rd,rs1,rs2
	Set < Immediate	I	SLTI rd,rs1,imm							R	FLPE.(S D Q) rd,rs1,rs2
	Set < Unsigned	R	SLTU rd,rs1,rs2							R	FCCLASS.(S D Q) rd,rs1
	Set < Imm Unsigned	I	SLTIU rd,rs1,imm							R	FMV.S.X rd,rs1
Branches	Branch =	SB	BEQ rd,rs1,rs2,imm							R	FMV.S.X rd,rs1
	Branch ≠	SB	BNE rd,rs1,rs2,imm							R	FCVTF.(S D Q).W rd,rs1
	Branch <	SB	BLT rd,rs1,rs2,imm							R	FCVTF.(S D Q).WU rd,rs1
	Branch ≥	SB	BGE rd,rs1,rs2,imm							R	FCVTF.W.(S D Q) rd,rs1
	Branch < Unsigned	SB	BLTU rd,rs1,rs2,imm							R	FCVTF.W.(S D Q) rd,rs1
	Branch ≥ Unsigned	SB	BGEU rd,rs1,rs2,imm							R	FCVTF.WU.(S D Q) rd,rs1
Jump & Link	J&L	UJ	JAL rd,imm							R	FCVTF.WU.(S D Q) rd,rs1
	Jump & Link Register	I	JALR rd,rs1,imm							R	FSWAP rd,rs1,rs2
Synch	Synch thread	I	FENCE rd							R	FSWAP rd,rs1,rs2
	Synch Instr & Data	I	FENCE.I rd							R	FSWAP rd,rs1,rs2
System	System CALL	I	SCALL rd							R	FSWAP rd,rs1,rs2
	System BREAK	I	SBREAK rd							R	FSWAP rd,rs1,rs2
Counters	Read CYCLE	I	RD CYCLE rd							R	FSWAP rd,rs1,rs2
	Read CYCLE upper Half	I	RD CYCLEH rd							R	FSWAP rd,rs1,rs2
	Read TIME	I	RD TIME rd							R	FSWAP rd,rs1,rs2
	Read TIME upper Half	I	RD TIMEH rd							R	FSWAP rd,rs1,rs2
	Read INSTR RETired	I	RD INSTRETD rd							R	FSWAP rd,rs1,rs2
	Read INSTR upper Half	I	RD INSTRETH rd							R	FSWAP rd,rs1,rs2
16-bit (RVC) and 32-bit Instruction Formats											
② RV32I / RV64I / RV128I											
③ RISC-V Reference Card											
④ Optional Compressed Instructions: RVC											

+ 6 for
64{F|D|Q}/
128{F|D|Q}



Outline

- RISC-V ISA Overview
- RISC-V Foundation Overview & Growth
- RISC-V Use Case Examples
 - NVIDIA and Western Digital
- Summary



RISC-V Foundation Overview

- Incorporated August, 2015 as a 501c6 non-profit Foundation
- Membership Agreement & Bylaws ratified December 2016
- The RISC-V ISA and related standards shall remain open and license-free to all parties
 - RISC-V ISA specifications shall always be publicly available as an online download
- The compatibility test suites shall always be publicly available as a source code download
- To protect the standard, only members (with commercial RISC-V products) of the Foundation in good standing can use “RISC-V” and associated trademarks, and only for devices that pass the tests in the open-source compatibility suites maintained by the Foundation

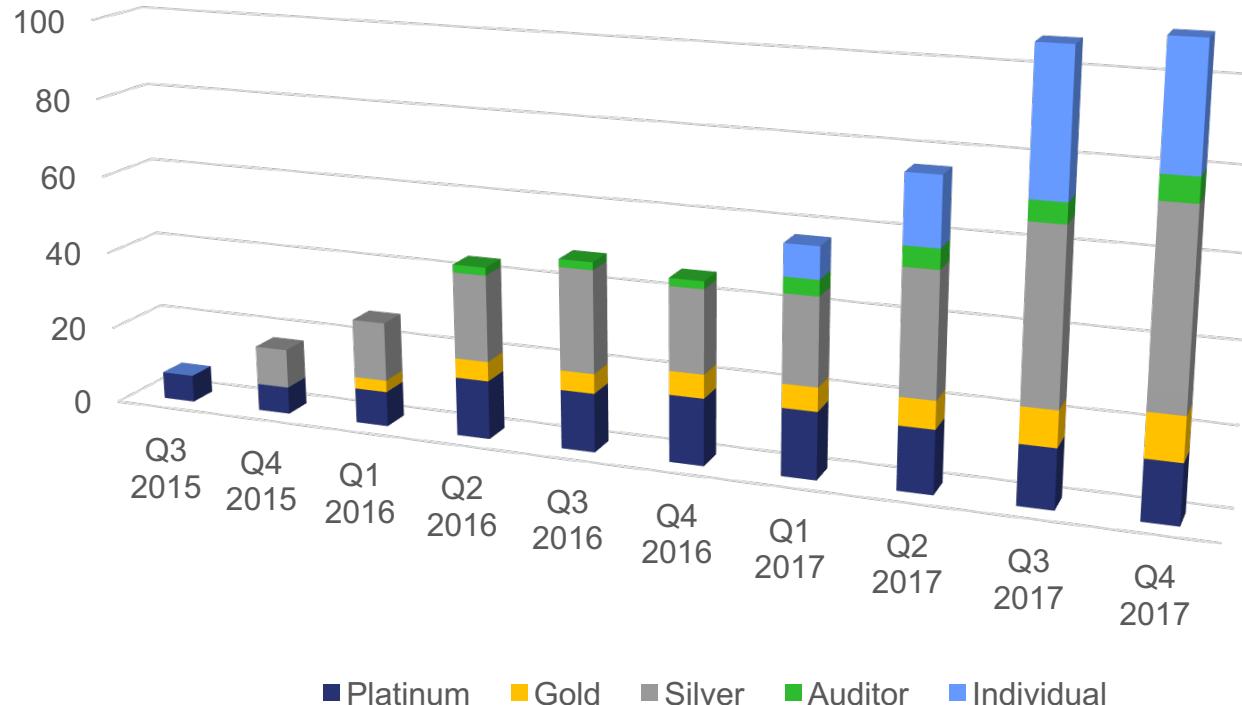


Foundation Organization

- The Board of Directors consists of seven+ members, whose replacements are elected by the membership
- The Board can amend the By-Laws of the RISC-V foundation via a two-thirds affirmative vote
- The Board appoints chairs of ad-hoc committees to address issues concerning RISC-V, and has the final vote of approval of the recommendation of the ad-hoc committees.
 - Technical Committee Chair – Yunsup Lee, SiFive
 - Marketing Committee Chair – Ted Marena, Microsemi
- All members of committees must be members of the RISC-V Foundation

RISC-V Foundation Growth History

August 2015 to November 2017





Berkeley
Architecture
Research

D R A P E R

bluespec

C-SKY Mentor®
A Siemens Business

LATTICE
SEMICONDUCTOR

Mellanox®
TECHNOLOGIES

NXP

cortus

Rambus



SiFive

Micron®

SAMSUNG

Google

Cryptography Research



QUALCOMM®

IBM

Western Digital®

Microsemi

nVIDIA

HUAWEI

RISC-V

Foundation: 100+ Members

ESPRESSIF

MEDIATEK

ANDES
TECHNOLOGY

runtime.io

DOVER
MICROSYSTEMS

BAE SYSTEMS

Esperanto
Technologies

VeriSilicon

inside
secure
DRIVING TRUST

GERRY
RESEARCH

antmicro
EMBEDDED SYSTEMS

IDT

ISTUARY
INNOVATION GROUP

SEGGER
It simply works!

GLORYSPACE
WWW.GLORYSPACE.CN

GOWIN
HALFBRIDGE FOR FUTURE

GRIDOG
"Eyes on the Grid, Eyes on the Buck"

LAUTERBACH
DEVELOPMENT TOOLS

INTRINSIX

ETH zürich

INSTITUTE OF TECHNOLOGY
MADRAS
Sri Sankaracharya University of Technology

imperas

NUS
National University
of Singapore

ASHLING
THE DEVELOPMENT SYSTEMS COMPANY

codasip

Roa Logic

GREENWAVES
TECHNOLOGIES

SHC
SH CONSULTING

MINIMA
PROCESSOR

Rumble
Development

TRINAMIC

XtremeEDA

VectorBlox
embedded supercomputing

UBILITE
Hello IoT World

SEAGATE

EMDALO TECHNOLOGIES
Embedded Software Solutions

EMBECOSM®

SECURERF
Securing the Internet of Things™

BERKELEY LAB

Syntacore™
Custom cores and tools

lowRISC

TechanaLyse

SIEMENS

优矽科技
UC TECH IP

Technolution

ultraSOC

csem

PRINCETON
UNIVERSITY

Blockstream

dxcorr

GLOBALFOUNDRIES®

expresslogic



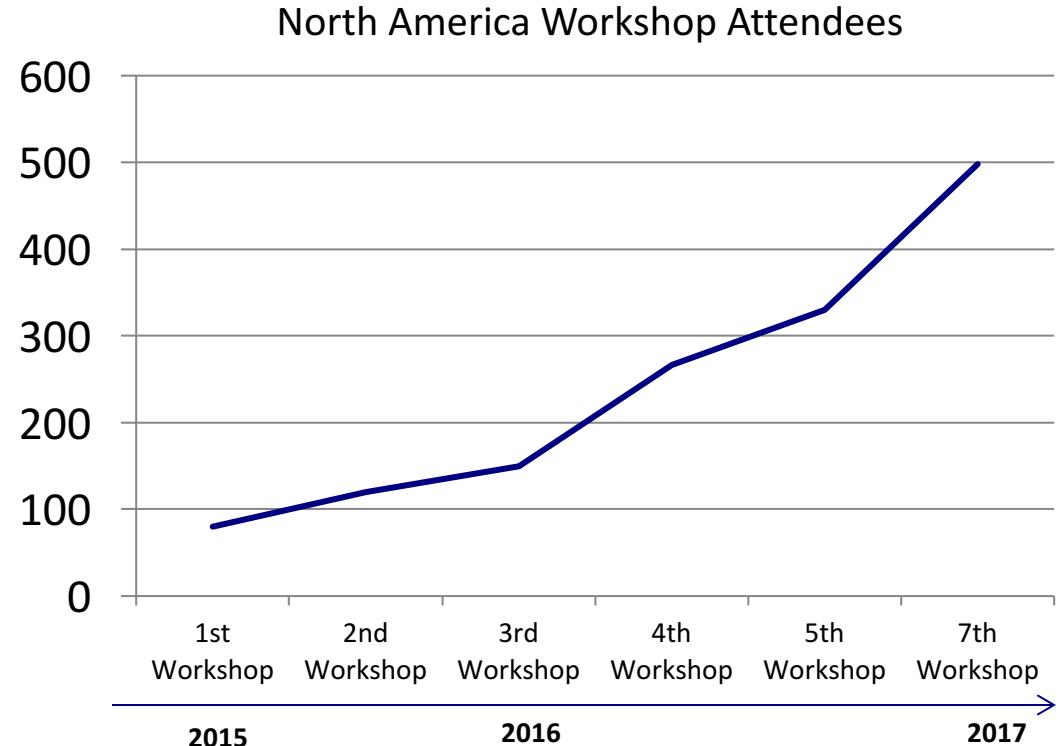
Some Workshop Stats...

- 1st RISC-V workshop Jan 14-15, 2015 in Monterey, CA
 - Sold out: 144 (33 companies & 14 universities) [Slides & videos can be found here](#)
- 2nd RISC-V workshop Jun 29-30, 2015 at UC Berkeley, CA
 - Sold out: 120 (30 companies & 20 universities) [Slides & videos can be found here](#)
- 3rd RISC-V workshop Jan 5-6, 2016 at Oracle Redwood City, CA
 - Sold out: 157 (42 companies & 26 universities) [Slides & videos can be found here](#)
- 4th RISC-V workshop Jul 12-13, 2016 at MIT Cambridge, MA
 - Sold out: 252 (63 companies & 42 universities) [Slides & videos can be found here](#)
- 5th RISC-V workshop Nov 29-30, 2016 at Google Mountain View, CA
 - Sold out: 350 (107 companies & 29 universities) [Slides & videos can be found here](#)
- 6th RISC-V workshop May 8-11, 2017 at NVIDIA / SJTU Shanghai, China
 - Sold out: 287 (52 companies & 27 universities) [Slides & videos can be found here](#)
- 7th RISC-V workshop Nov 28-30, 2017 at Western Digital Milpitas, CA
 - Sold out: 498 (138 companies & 35 universities) [Slides & videos can be found here](#)

Workshop Growth

7th Workshop By the Numbers

- **498** registered attendees, more than **5X** the attendance at the 1st Workshop
- ~80 abstracts submitted
- **47** sessions squeezed into 12 & 24 minute increments
- **26** poster / demo sessions





Outline

- RISC-V ISA Overview
- RISC-V Foundation Overview & Growth
- RISC-V Use Case Examples
 - NVIDIA and Western Digital
- Summary



6th RISC-V Workshop - May 8th – 11th, 2017 Shanghai China

Following slides excerpt from Frans Sijstermans, VP Engineering,
NVIDIA Keynote Address - Full [presentation is here](#)



NVIDIA®



Falcon's history

- Embedded in 15+ designs
- Taped out in ~50 chips
- Shipped ~3 billion times
- No stop-ship bugs

Falcons shipped estimate	
dGPU Volume /year	50M*
Years Falcon shipping	10
Avg. #Falcons / GPU	10
Avg. NVIDIA market share	60%
Total shipped	3 billion

<http://www.anandtech.com/show/10864/discrete-desktop-gpu-market-trends-q3-2016>

Selecting the next architecture

Technical criteria

- >2x performance of Falcon
- <2x area cost of Falcon
- Support for caches as well tightly coupled memories
- 64-bit addresses
- Suitable for modern OS

Considered architectures

- ARM
- Imagination Technologies MIPS
- Synopsys ARC
- Cadence Tensilica
- RISC-V

Why RISC-V for Falcon Next

RISC-V is the only architecture that meets all our criteria

https://riscv.org/wp-content/uploads/2016/07/Tue1100_Nvidia_RISCV_Story_V2.pdf

Item	Requirement	ARM A53	ARM A9	ARM R5	RISC-V Rocket	NV RISC-V
Core perf	>2x falcon	Yes	Yes	Yes	Yes	Yes
Area (16ff)	<0.1mm^2	No	No	Yes	Yes	Yes
Security	Yes	TZ	TZ	No	Yes	Yes
TCM	Yes	Yes	No	Yes	No	Yes
L1 I/D \$	Yes	Yes	Yes	Yes	Yes	Yes
Addressing	64bit	Yes	No	No	Yes	Yes
Extensible ISA	Yes	No	No	No	Yes	Yes
Safety (ECC/Parity)	Yes	Yes	Yes	Yes	Yes	Yes
Functional Simulation model	Yes	Yes	No	No	No	Yes



7th RISC-V Workshop - November 28th –
30th, 2017 Milpitas, California

Western Digital®

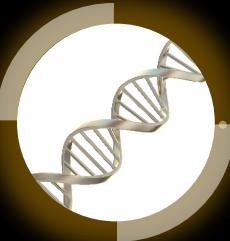


Following slides excerpt from
Martin Fink, CTO, Western Digital
Keynote Address
Full [presentation is here](#)

RISC-V Meets the Needs of Big Data and Fast Data

**Big
Data**

Predictive
Analytics



Genomics

Machine
Learning



RISC-V

Autonomous
Machines



**Fast
Data**

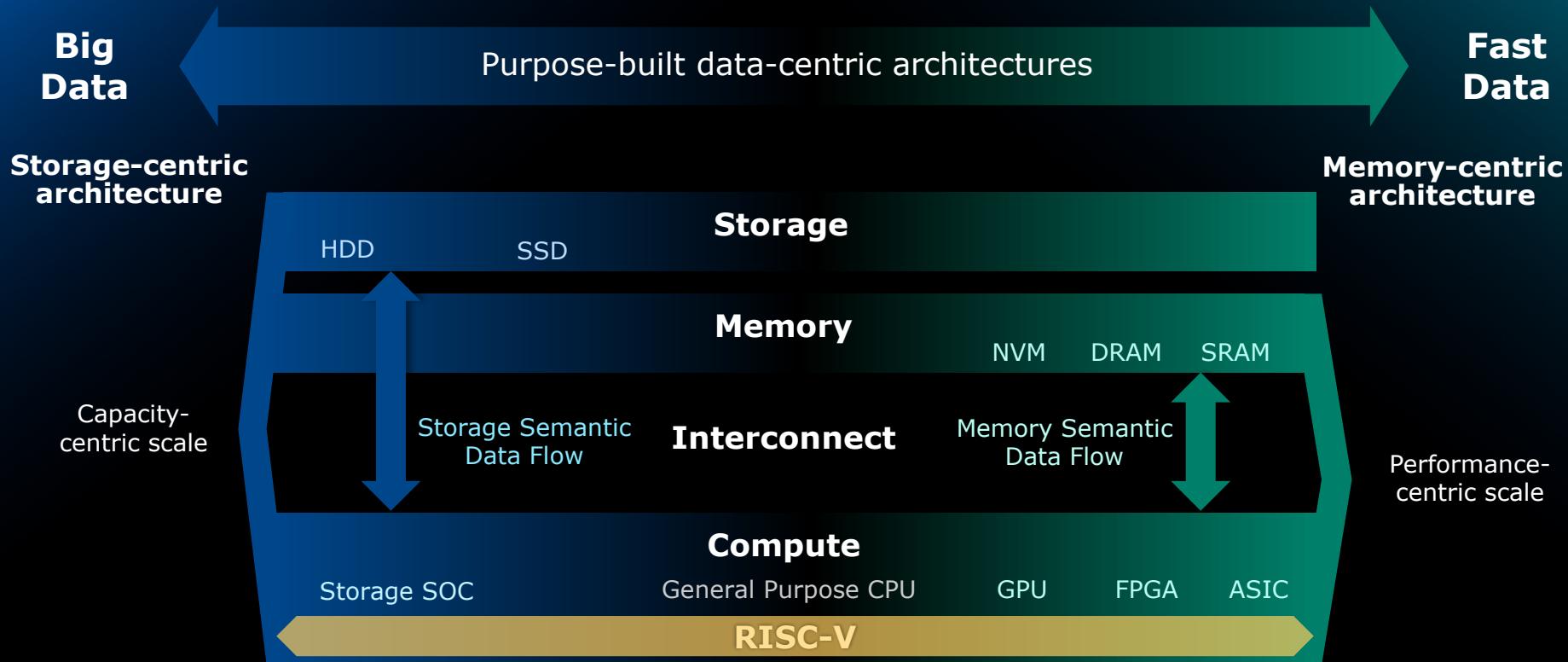
Safety
& Security



Private
Exchange



RISC-V Enables Purpose-Built Environments for Big Data and Fast Data Applications



Driving Momentum

Western Digital ships in excess of
1 Billion cores per year
...and we expect to **double that.**

Accelerating the RISC-V Ecosystem

Western Digital to contribute one billion cores annually to fuel RISC-V

- 1 Support development of open source IP building blocks for the community
- 2 Actively partner and invest in the ecosystem
- 3 Accelerate development of purpose-built processors for a broad range of Big Data and Fast Data environments
- 4 Multi-year transition of Western Digital devices, platforms and systems to RISC-V purpose-built architectures



Save the Date – 8th RISC-V Workshop

- Co-Hosted by Barcelona Supercomputing Center (BSC) and Universitat Politècnica de Catalunya (UPC)
- May 7th – 10th, 2018



UNIVERSITAT POLITÈCNICA
DE CATALUNYA
BARCELONATECH



**Barcelona
Supercomputing
Center**
Centro Nacional de Supercomputación





RISC-V ISA & Foundation Summary

- The free and open RISC-V ISA is enabling a new innovation frontier across all computing devices
- Strong Industry Support
 - 100+ members; Broad commercial and academic interest (sold out 7 straight workshops)
- Chosen Best Technology of 2016 by The Linley Group
- RISC-V Twitter [@risc_v](http://twitter.com/risc_v)
- RISC-V LinkedIn Page
 - <http://www.linkedin.com/company/risc-v-foundation>

