

Testing Infrastructure for RISC-V Toolchains & Runtimes

Wei Wu


lazyparser@gmail.com




PLCT Lab

<https://ci.rvperf.org>


- Jenkins + 5 buildbots
 - Ubuntu 18.04 x 3
 - Ubuntu 20.04 x 2
 - Docker enabled on all buildbots
 - 16c-128g-4t or 32c-192g-2t
- CI for github.com/riscv/riscv-gnu-toolchain had been established
 - Default master branch
 - rvv-branch
 - fsf-trunk-daily-build
- Also has build jobs for V8, OpenJDK, and QEMU


Adding new jobs now. Suggestions & Issues are welcome!


 **Jenkins**


  PLCT Lab  log out


Dashboard


 New Item


 People


 Build History


 Project Relationship


 Check File Fingerprint

 Check File Fingerprint

 My Views

 Open Blue Ocean


 Lockable Resources

 New View


Build Queue

No builds in the queue.


Build Executor Status

 plct-build-7

1 idle

 pts-01

1 idle

 pts-02

1 idle


CI Infrastructure for RISC-V Ecosystem. Maintained by PLCT Lab. Open issues on <https://github.com/plctlab/riscv-ci> [edit description](#)

All GNU Krypto-Scalar OpenJDK P-extension QEMU V8 Zfinx upstream +

S	W	Name ↓	Last Success	Last Failure	Last Duration	Fav
		fsf-gnu-toolchain-master-all-ubuntu1804	7 hr 54 min - #5	16 hr - #2	43 min	 
		fsf-gnu-toolchain-master-all-ubuntu2004	13 hr - #4	N/A	2 hr 46 min	 
		linsinan-riscv-gnu-toolchain-ubuntu1804	3 hr 0 min - #11	12 hr - #9	6 min 30 sec	 
		linsinan-riscv-gnu-toolchain-ubuntu2004	3 hr 1 min - #10	4 days 12 hr - #3	9 min 25 sec	 
		plct-gnu-zfinx-rv32E	6 days 1 hr - #9	16 hr - #15	26 min	 
		plct-gnu-zfinx-rv32gc	15 hr - #16	6 days 2 hr - #9	17 min	 
		plct-gnu-zfinx-rv64gc	1 hr 50 min - #27	6 days 2 hr - #19	17 min	 
		plct-llvm-k-extension-ubuntu2004	22 hr - #8	N/A	44 min	 
		plct-llvm-p-extension-ubuntu2004	18 hr - #18	10 days - #1	30 min	 
		plct-llvm-rvv-ubuntu2004	21 hr - #26	10 days - #8	42 min	 
		plct-llvm-zfinx-ubuntu2004	N/A	3 days 8 hr - #16	29 min	 
		plct-qemu-k-ext-dev	21 hr - #7	N/A	10 min	 
		plct-qemu-nuclei	5 days 11 hr - #1	5 days 0 hr - #3	7 min 27 sec	 
		plct-qemu-rtthread	5 days 11 hr - #1	5 days 1 hr - #3	10 min	 

Several failures observed

<https://ci.rvperf.org/job/fsf-gnu-toolchain-master-all-ubuntu1804/>

 **Jenkins**

search

PLCT Lab log out

Dashboard fsf-gnu-toolchain-master-all-ubuntu1804

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Build Now](#)

[Configure](#)

[Delete Pipeline](#)

[Full Stage View](#)

[Open Blue Ocean](#)

[Rename](#)

[Pipeline Syntax](#)

[Git Polling Log](#)

Build History trend ^

find

#5
Mar 11
03:08
No Changes

#4
Mar 10
18:44
1 commit

#3
Mar 10
18:27
1 commit

Pipeline fsf-gnu-toolchain-master-all-ubuntu1804

[Recent Changes](#)

[add description](#)

Disable Project

Stage View

	Declarative: Checkout SCM	Get the source	make newlib	make linux	make linux multilib	build newlib-rv32i-ilp32-medlow on ubuntu1804	report binutils newlib-rv32i-ilp32-medlow on ubuntu1804	report gcc newlib-rv32i-ilp32-medlow on ubuntu1804	build newlib-rv32i-ilp32-medlow on ubuntu1804	report binutils newlib-rv32i-ilp32-medlow on ubuntu1804	report gcc newlib-rv32i-ilp32-medlow on ubuntu1804	build newlib-rv32i-ilp32-medlow on ubuntu
Average stage times: (Average full run time: ~5h 38min)	3s	4min 9s	12min 10s	19min 39s	415ms	3min 20s	4min 3s	1min 50s	3min 20s	4min 7s	1min 49s	3min
#5 Mar 11 03:08 No Changes	707ms	6s	16min 49s failed	25min 15s failed	519ms failed	478ms	501ms	758ms failed	494ms	477ms	751ms failed	476ms
#4 Mar 10 18:44 1 commit	900ms	10min 14s	17min 33s failed	25min 54s failed	504ms failed	8min 23s	9min 54s	523ms failed	8min 25s	9min 59s	502ms failed	8min
#3 Mar 10 18:27 1 commit	588ms	4s	9min 44s failed	22min 1s failed	504ms failed	724ms	45s	9min 8s failed	488ms	45s	9min 6s failed	513ms

<https://ci.rvperf.org/blue/organizations/jenkins/fsf-gnu-toolchain-master-all-ubuntu1804/detail/fsf-gnu-toolchain-master-all-ubuntu1804/5/pipeline/32>

Stage Logs (make linux)

Shell Script -- ./configure --prefix=\$PWD/build-rv64-linux & make -j \$(nproc) report-linux (self time 25min 15s)

```
FAIL: gfortran.dg/pr96711.f90 -00 (internal compiler error)
FAIL: gfortran.dg/pr96711.f90 -00 (test for excess errors)
FAIL: gfortran.dg/pr96711.f90 -01 (internal compiler error)
FAIL: gfortran.dg/pr96711.f90 -01 (test for excess errors)
FAIL: gfortran.dg/pr96711.f90 -02 (internal compiler error)
FAIL: gfortran.dg/pr96711.f90 -02 (test for excess errors)
FAIL: gfortran.dg/pr96711.f90 -03 -fomit-frame-pointer -funroll-loops -fpeel-loops -ftracer -finline-functions (internal compiler error)
FAIL: gfortran.dg/pr96711.f90 -03 -fomit-frame-pointer -funroll-loops -fpeel-loops -ftracer -finline-functions (test for excess errors)
FAIL: gfortran.dg/pr96711.f90 -03 -g (internal compiler error)
FAIL: gfortran.dg/pr96711.f90 -03 -g (test for excess errors)
FAIL: gfortran.dg/pr96711.f90 -0s (internal compiler error)
FAIL: gfortran.dg/pr96711.f90 -0s (test for excess errors)
```

===== Summary of gcc testsuite =====

	# of unexpected case		# of unique unexpected case	
	gcc		g++	
	gcc		g++	
rv64imafdc/ lp64d/ medlow	4	2	28	7
			12	1

Makefile:920: recipe for target 'report-gcc-linux' failed

make: *** [report-gcc-linux] Error 1

Build scripts are open source

- <https://github.com/plctlab/riscv-ci/blob/main/riscv-gnu-toolchain-master-ubuntu1804.jenkinsfile>
 - In a separate repo
- <https://github.com/plctlab/riscv-gnu-toolchain/blob/fsf-trunk-daily-build/jenkinsfile.ubuntu1804>
 - Directly in the repo
- <https://github.com/plctlab/PLCT-Toolbox/tree/master/ci.rvperf.org/job-configs>
 - Bash scripts. Obsolete

Help to add more CI for GNU Toolchain

- Physical RISC-V machines/devboards will be available in late April
- More x86 VMs would be added
- One QA staff checks the results and sync to developers
- Need to add more testsuites, benchmarks, and build configs
-
- <https://github.com/plctlab/riscv-ci/issues/new>
-

RISC-V All-In-One Project

Wei Wu

lazyparser@gmail.com

PLCT Lab

All-in-one Project in one page

- RISC-V has several ISA extension; extension has different status
- Draft, stable, frozen, ratified.
- Only ratified extensions are accepted
- Extensions in other status have a develop branch

Problem: If you want to use V, P, K, and B extensions, you have to merge them yourself. Very hard work.

Solution: Maintain a series of downstream repos/branches publicly

Roadmap

priv	Linux Kernel 2021H2	Hypervisor TBD.	
un-priv	Toolchain: GNU, LLVM 2021H1	Simulator: Spike, QEMU 2021H1	FPGA SoC: TBD. TBD.



Option 2: Create Two Vendor Branch on Upstream [github/plctlab/](https://github.com/plctlab/)

- Master branch keep current scheme, not accept unratified extension.
- One branch for all **non-draft** unratified extension.
 - e.g. v 0.9, b 0.92 and zfh 0.1 in this branch
- **One branch for all draft unratified extension.**
 - e.g. v 1.0-draft, b 0.93-draft in this branch.

The PLCT Lab will maintain a branch for this purpose.

- **Pros:**
 - Easier to use unratified ext.
- **Cons:**
 - Need extra effort when certain extension become ratified.
 - e.g. We have V, B and Zfh in branch, but V is ratified now, we need to extract the V part from the branch and merge into master.



Policy in Other Open Source Projects

- **Linux kernel**

- Same policy as GNU toolchain, **NO** unratified extension and custom extension could upstream.
 - RFC patches are welcome for review, but won't merge until ratified.
 - Must be maintained in downstream.
 - NO vendor/custom ext. allowed.

- **Qemu**

Plan: new RISC-V Machine that can enable all ratified/stable/draft extensions.
fork a new machine/cpu in 2021Q1, enable V,P,B,K in 2021Q2

- Allow unratified extension upstream **master** branch, but must be **non-draft** version.
 - e.g. V-ext v0.9 is OK, V-ext 1.0-draft is not OK
 - Disable by default must enable by option -cpu rv64, **x-v=true, vext_spec=v0.9**
 - NO vendor/custom ext. allowed.

- **LLVM**

- Allow unratified extension upstream, but must be non-draft version
- Must enable with an extra option: -menable-experimental-extension
- Must explicitly specify the extension version: -march=rv64gc **v0p9**
- No vendor/custom ext. there, but RISC-V LLVM maintainers are open mind for that.

Open source

- <https://github.com/riscv/riscv-gcc/tree/riscv-gcc-experiment>
- <https://github.com/riscv/riscv-binutils-gdb/tree/riscv-binutils-experiment>
-
- <https://github.com/plctlab/riscv-gnu-toolchain/>
- <https://github.com/plctlab/riscv-gcc>
- <https://github.com/plctlab/riscv-binutils-gdb>
- <https://github.com/plctlab/riscv-glibc>
- <https://github.com/plctlab/riscv-newlib>
- <https://github.com/plctlab/plct-qemu>
- <https://github.com/plctlab/llvm-project>
- (No kernel developer yet.)

Thanks!

- Issues and Comments are welcome!
- PRs are welcome!
- The downstream branches are never go upstream.
 - So, no copy assignments needed. (IMO)
-