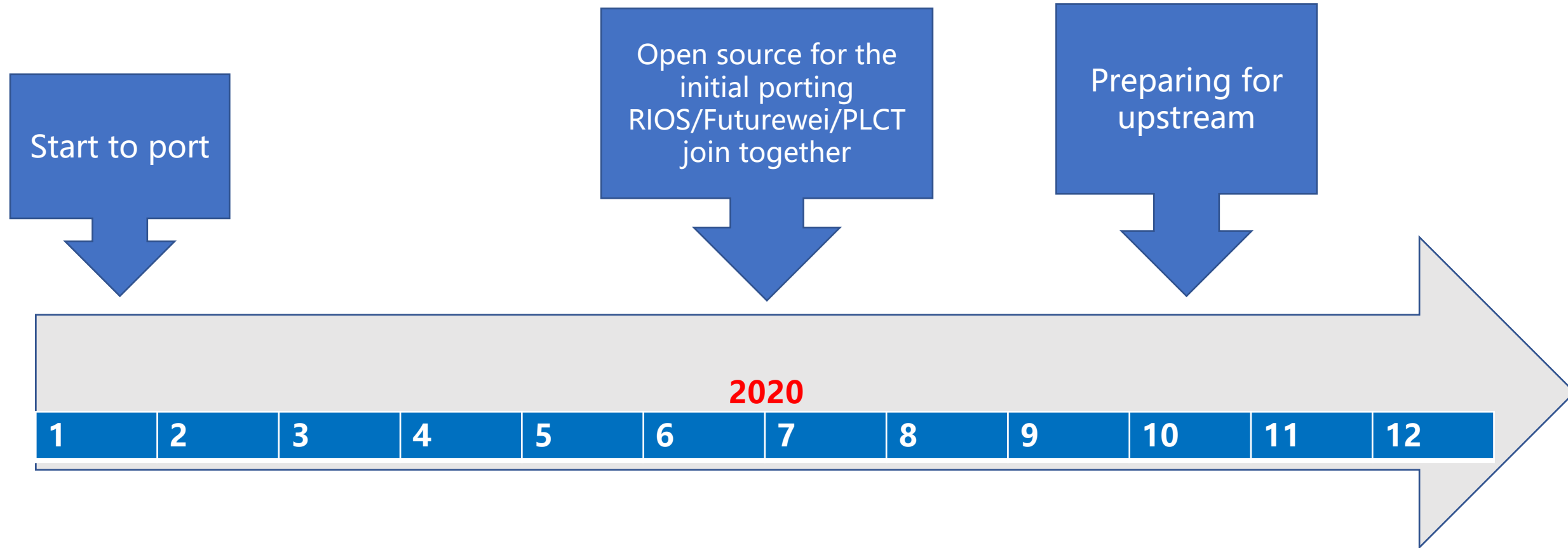


V8 for RISC-V: Working Toward a Tier-1 Platform



PLCT V8 Team
yahan@iscas.ac.cn

Retrospect: V8 for RISC-V in 2020



The porting process can be referred to:

1. Unlocking JavaScript V8 on RISC-V by Peng Wu, <https://www.youtube.com/watch?v=MdHL9yGQer8>
2. V8 for RISC-V, where to come and where to go by PLCT lab, <https://github.com/isrc-cas/PLCT-OpenDay-2020/blob/main/V8forRV-PLCTOD2020.pdf>

Retrospect: V8 for RISC-V in 2020

V8 RISC-V Core Team



The
1. Un
2. V8
<http://>

Contents for today's presentation

- upstreaming process
- daily maintain after upstream
- node.js status
- feedback from tuning V8 to the ISA
- future work

The upstream process

Contribute your code to v8

- Get source code:

1.git clone https://chromium.googlesource.com/chromium/tools/depot_tools.git

2.export PATH=/path/to/depot_tools:\$PATH

3. mkdir v8 & cd v8

3. fetch v8 & gclient sync

Configure Git

Configure Git with a cookie for this service by copying this script and pasting it into a shell.

Windows Users ↴

Authenticate as **luyahan@qq.com** to chromium.googlesource.com

```
eval 'set +o history' 2>/dev/null || setopt HIST_IGNORE_SPACE 2>/dev/null
touch ~/.gitcookies
chmod 0600 ~/.gitcookies

git config --global http.cookiefile ~/.gitcookies

tr , \\t <<\\__END__ >>~/.gitcookies
chromium.googlesource.com,F
chromium-review.googlesource.com,F
__END__
eval 'set -o history' 2>/dev/null || unsetopt HIST_IGNORE_SPACE 2>/dev/null
```

This script is designed to work in **bash** and **zsh**. It ensures that the password is not saved in your shell history.
If your home directory is on a network filesystem, consider using a location on local disk.

- Sign the [Google Individual Contributor License Agreement](#)
- Init git setup for chromium
visit <https://chromium.googlesource.com>

The upstream process

Contribute your code to v8

- Checkout your branch

git checkout -b mychange +origin/main

- Upload your cl(ch

git cl upload

- Review your code

The screenshot displays the Chromium Gerrit web interface. At the top, the navigation bar includes 'Chromium Gerrit', 'CHANGES', 'YOUR', 'DOCUMENTATION', and 'BROWSE'. The main header shows the change is 'Active', has 2952865 stars, and is titled '[riscv664] Port the changest patch'. The 'Change Info' section on the left lists the owner 'Lu Yahan', reviewers 'Brice Dobry' and 'Ji Qiu', and CCs 'v8-reviews@g...', 'leszeks+watc...', 'verwaest+wat...', and 'V8 LUCI CQ'. It also shows the repository 'v8/v8' and branch 'main', with the topic 'riscv664'. The 'Submit requirements' section shows 'Code-Review' with 'No votes'. The 'Other labels' section shows 'Auto-Submit', 'Bot-Commit', 'Commit-Queue', and 'Owners-Override', all with 'No votes'. On the right, the 'REPLY' button is visible, followed by the change description: '[riscv664] Port the changest patch', 'Port 2944844: [sparkplug][arm][arm64[ia32] Callee-saved registers for RecordWrite', 'Delete kTestReg due to 2945538', and 'Change-Id: I33d12df7cbee6842a05c1dbe3e6158be1ff4cbc7'. At the bottom, it shows '4 unresolved' and '1 resolved' comments.

The upstream process

Review status

- WIP(work in process)
- Active
 - If you are a commiter, you can add flag CQ(Commit-Queue)+1 to trigger test.
 - OWNERS review code.
If your code is good for owner, owner will reply LGTM and CQ+2 and trigger test again.
If all test all green, your code will be merge.
- Merged
The cl had been merged into upstream
- Abanded
The cl had been rejected.

The upstream process

- at first: downstream maintain on GitHub
 - <https://github.com/riscv/v8>
- launch upstream at 2020 October
- many rounds of rebase and bug fix
- skipped several test cases for the regression routine
- finished at 2021 February 10

🔍 v8 / v8 Mirror

mirrored from <https://chromium.googlesource.com/v8/v8.git>

<> Code

🔗 Pull requests

🎮 Actions

📖 Wiki

🛡 Security

📈 Insights

Add RISC-V backend

This very large changeset adds support for RISC-V.

Bug: v8:10991

Change-Id: Ic997c94cc12bba6881bc208e66526f423dd0679c

Reviewed-on: <https://chromium-review.googlesource.com/c/v8/v8/+/2571344>

Commit-Queue: Brice Dobry <brice.dobry@futurewei.com>

Commit-Queue: Georg Neis <neis@chromium.org>

Reviewed-by: Georg Neis <neis@chromium.org>

Reviewed-by: Hannes Payer <hpayer@chromium.org>

Reviewed-by: Michael Achenbach <machenbach@chromium.org>

Reviewed-by: Michael Stanton <mvstanton@chromium.org>

Cr-Commit-Position: refs/heads/master@{#72598}

🔗 master

📦 9.3.38 ... 9.0.150



fw-brice authored and **Commit Bot** committed on 10 Feb



Showing **104 changed files** with **41,269 additions** and **135 deletions**.

<https://github.com/v8/v8/commit/ffd9e82dd5dd314bad48ab602517bd378b6d2184>

Daily maintain after upstream

- latest changes port
 - for passing build
 - for new features
- ISA and ABI related configurations and fix
- bug fix
- arch-related optimization

Daily maintain after upstream

- latest changes port
 - for passing build
 - for new features

1. Port the latest changes to pass build
 1. 2690182[*]
 2. 2914246
 3. 2894036
2. Port the latest changes to introduce new feature
 1. OSR shadow stack (2831171)
 2. add static interface descriptors (2834632)
 3. The newly introduced compiler Sparkplug port (2763963)
 4. use EnterFrame and LeaveFrame with StackFrame (2848735)
 5. wasm atomic implementation (2881494)

[*] This is the change number in the Chromium' s Gerrit system.
e.g., <https://chromium-review.googlesource.com/c/v8/v8/+2690182>

Daily maintain after upstream

- latest changes port: (1) for passing build (2) for new feature
- **ISA and ABI related configurations and fix**
- bug fix
- arch-1
 1. add toolchain configurations for RISC-V (2725484, 2754588)
 2. make t6 as a call target register (2814726)
 3. rename riscv flags (2878150)
 4. C-ext configuration (2876854)
 5. fix disassemble of illegal instruction (2874399)

Daily maintain after upstream

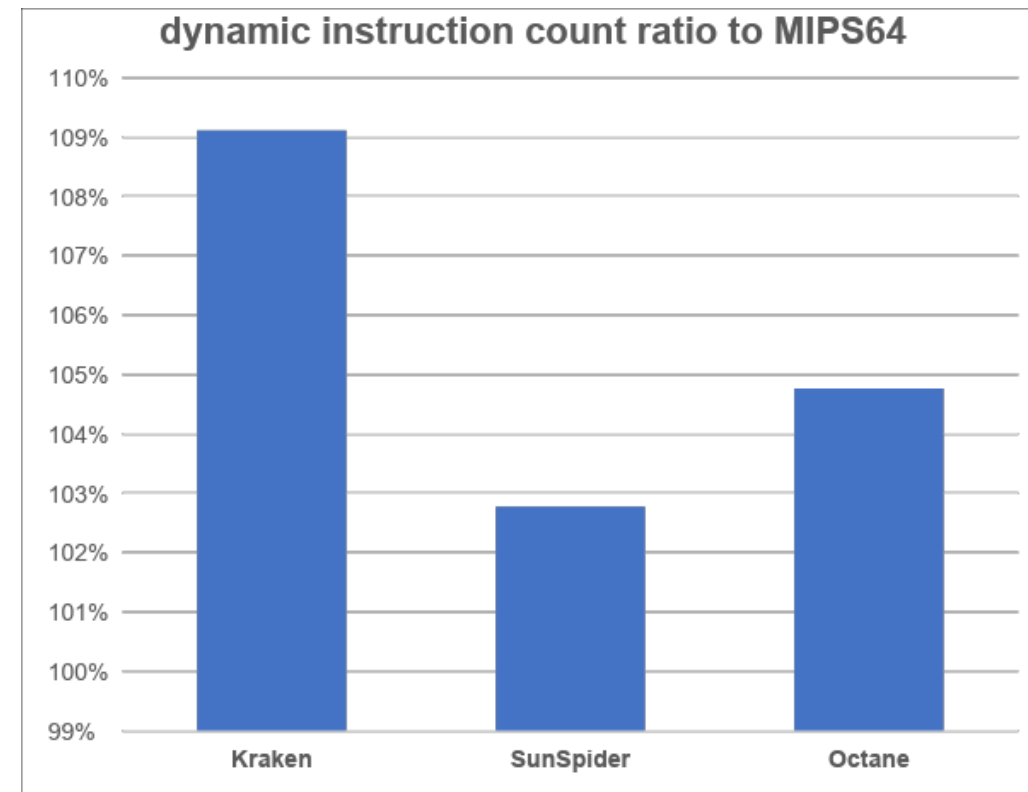
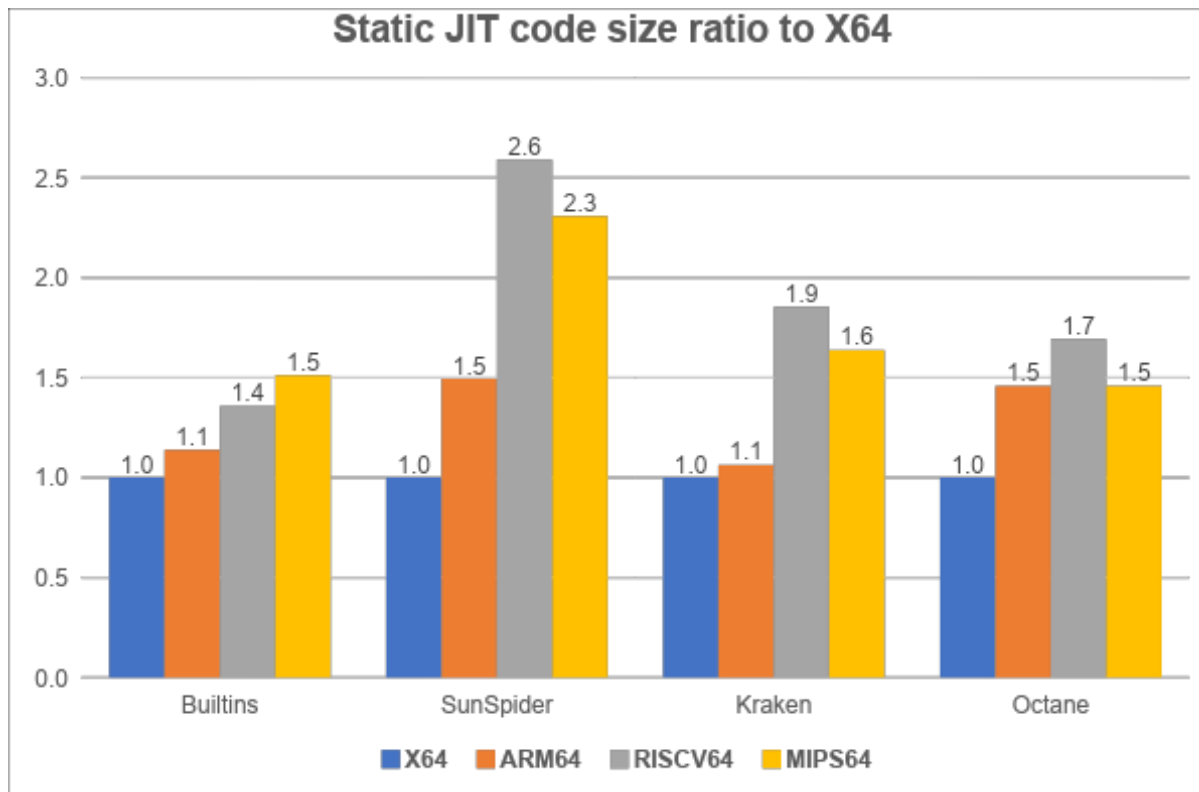
- latest changes port: (1) for passing build (2) for new feature
- ISA and ABI related configurations
- bug fix
 1. NaN processing issue (2814725)
 2. Record correct offset in StoreTaggedPointer (2814723)
 3. CPU profiler related bug (also help MIPS change the bug) (2902341)
 4. typo fix (2917597)
 5. unaligned mem access (2887023)
 6. fix constant pool (2839546)
 7. skip atomic test (2853282)
 8. skip cctests (2881505)
 9. skip inspector test (2894025)
- arch-related

Daily maintain after upstream

- latest changes port: (1) for passing build (2) for new feature
- ISA and ABI related configurations
- bug fix
- **arch-related optimization**

1. Port PC-relative builtin-to-builtin calls (2814722)
2. enable constant pool (2814724)
3. remove of unnecessary not (2847673)
4. add/sub with immediate (2848732)
5. add RISCV CmpZero to gen bnez/beqz with zero_reg (2814562)
6. fold imm to the load/store offset (2814563)

Current status-performance



dynamic instruction count: collected from the V8 simulation run

Current status-stability and the CI system

The screenshot shows the Jenkins web interface for the V8 project. The browser address bar indicates the URL is `ci.rvperf.org/view/V8/`. The Jenkins dashboard includes a sidebar with navigation links: People, Build History, Project Relationship, Check File Fingerprint, Open Blue Ocean, and Credentials. The main content area displays a table of build statuses for various targets. The 'V8' tab is selected, showing two builds: 'v8-github-riscv64' and 'v8-upstream-master'. The table columns are: S (Status), W (Weather icon), Name, Last Success, Last Failure, and Last Duration. The 'v8-github-riscv64' build is shown as a success (blue circle) with a weather icon of a cloud and rain, indicating a failure. The 'v8-upstream-master' build is shown as a success (blue circle) with a weather icon of a sun, indicating a success. Below the table, there is a legend and links for Atom feeds.

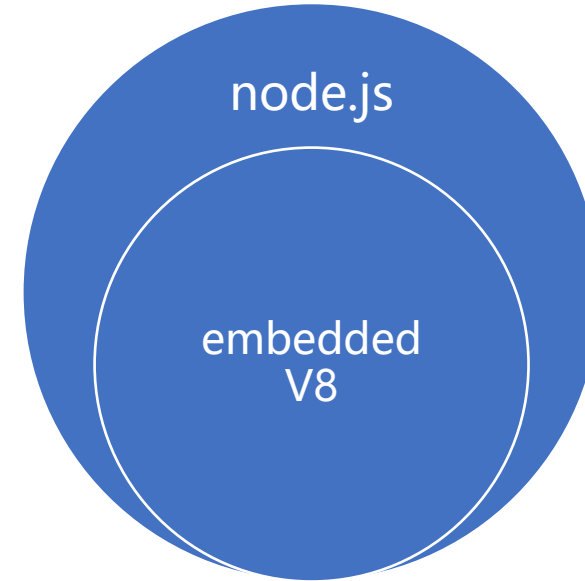
S	W	Name ↓	Last Success	Last Failure	Last Duration
		v8-github-riscv64	14 hr - #149	23 hr - #148	1 hr 57 min
		v8-upstream-master	5 hr 36 min - #188	2 days 5 hr - #182	51 min

Icon: S M L

Legend: [Atom feed for all](#) [Atom feed for failures](#) [Atom feed for just latest builds](#)

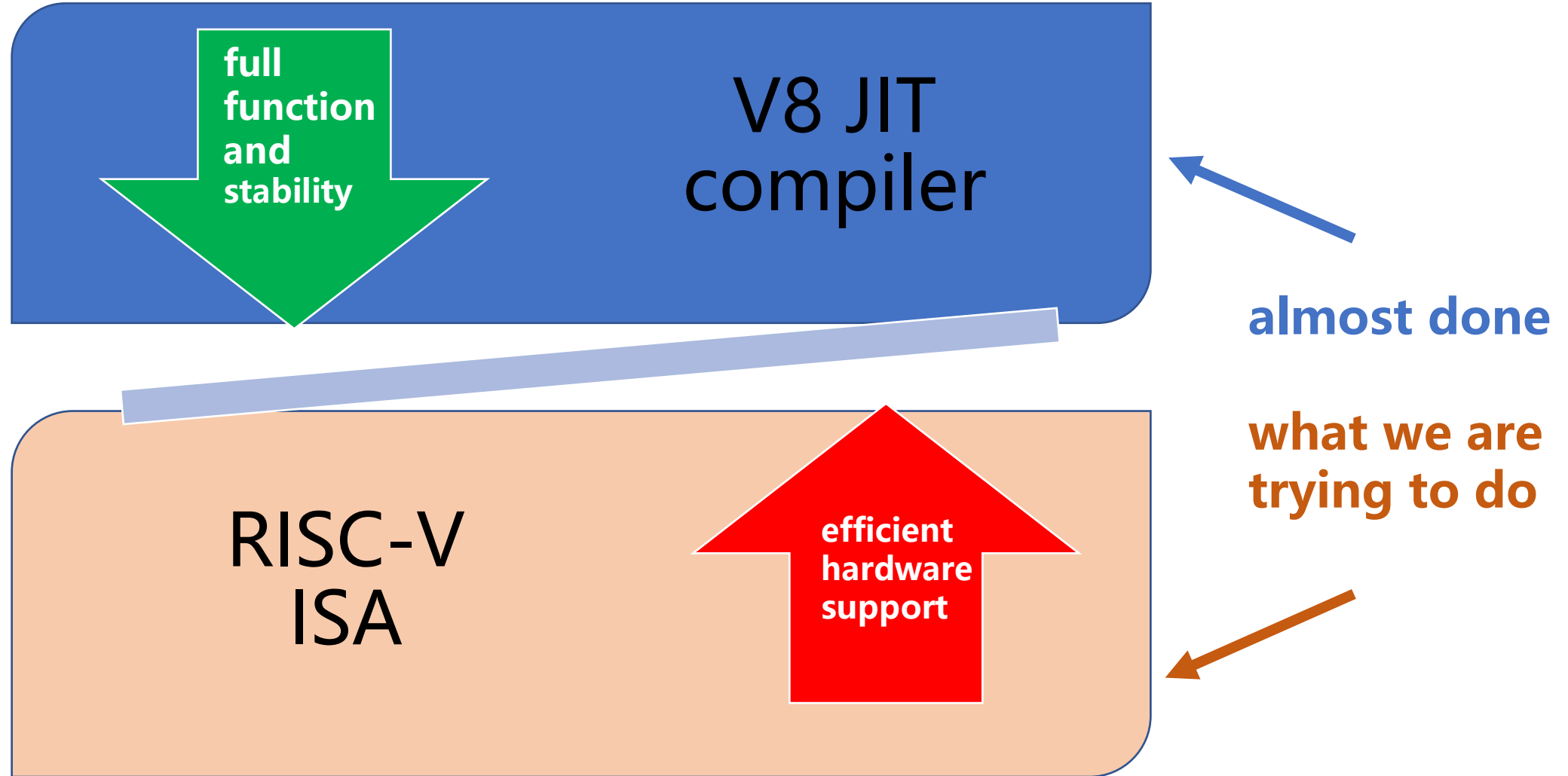
- no full CI on upstream
 - new commits may break the RISC-V target
- PLCT lab had set up a website for the RISC-V software
 - master branch: always keep up with upstream master
 - riscv64 branch: the latest commit that is good for RISC-V (pass all the regression test suit)

node.js support



- a downstream version was ported:
 - <https://github.com/v8-riscv/node/tree/riscv64>
- committed the RISC-V 64G ISA build configurations
 - <https://github.com/nodejs/node/commit/1d21a8d140f78d4408c5375caedc917a83902a80>
- node.js merged v8-9.1, node.js for riscv64 has been build pass

feedback from tuning V8 to the ISA



Feedback1 : li_ptr macro generates more instruction

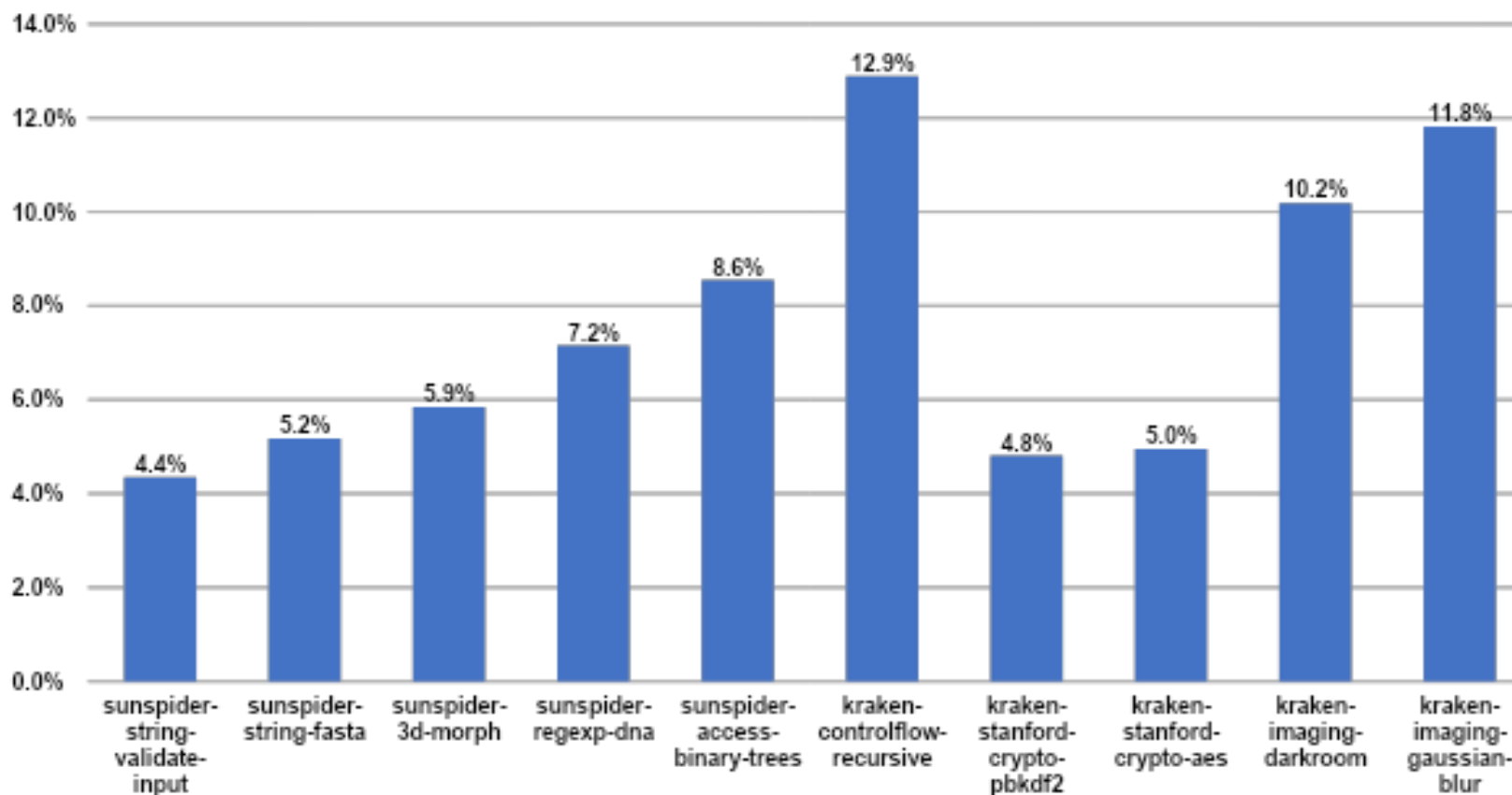
ISA	Instruction sequence	counts
RISCV64	<pre>int64_t a6 = imm & 0x3f; // bits 0:5. 6 bits int64_t b11 = (imm >> 6) & 0x7ff; // bits 6:11. 11 bits int64_t high_31 = (imm >> 17) & 0x7fffffff; // 31 bits int64_t high_20 = ((high_31 + 0x800) >> 12); // 19 bits int64_t low_12 = high_31 & 0xfff; // 12 bits lui(rd, (int32_t)high_20); addi(rd, rd, low_12); // 31 bits in rd. slli(rd, rd, 11); // Space for next 11 bis ori(rd, rd, b11); // 11 bits are put in. 42 bit in rd slli(rd, rd, 6); // Space for next 6 bits ori(rd, rd, a6); // 6 bits are put in. 48 bis in rd</pre>	6
MIPS64	<pre>lui(rd, (j.immediate() >> 32) & klmm16Mask); ori(rd, rd, (j.immediate() >> 16) & klmm16Mask); dsll(rd, rd, 16); ori(rd, rd, j.immediate() & klmm16Mask);</pre>	4

li_ptr: load a 48bit immediate into a register in a fix length sequence

RISC-V: I-Type has only 12bit imm filed

MIPS64: I-Type has 16 bit imm field

li_ptr macro generates more instruction-2

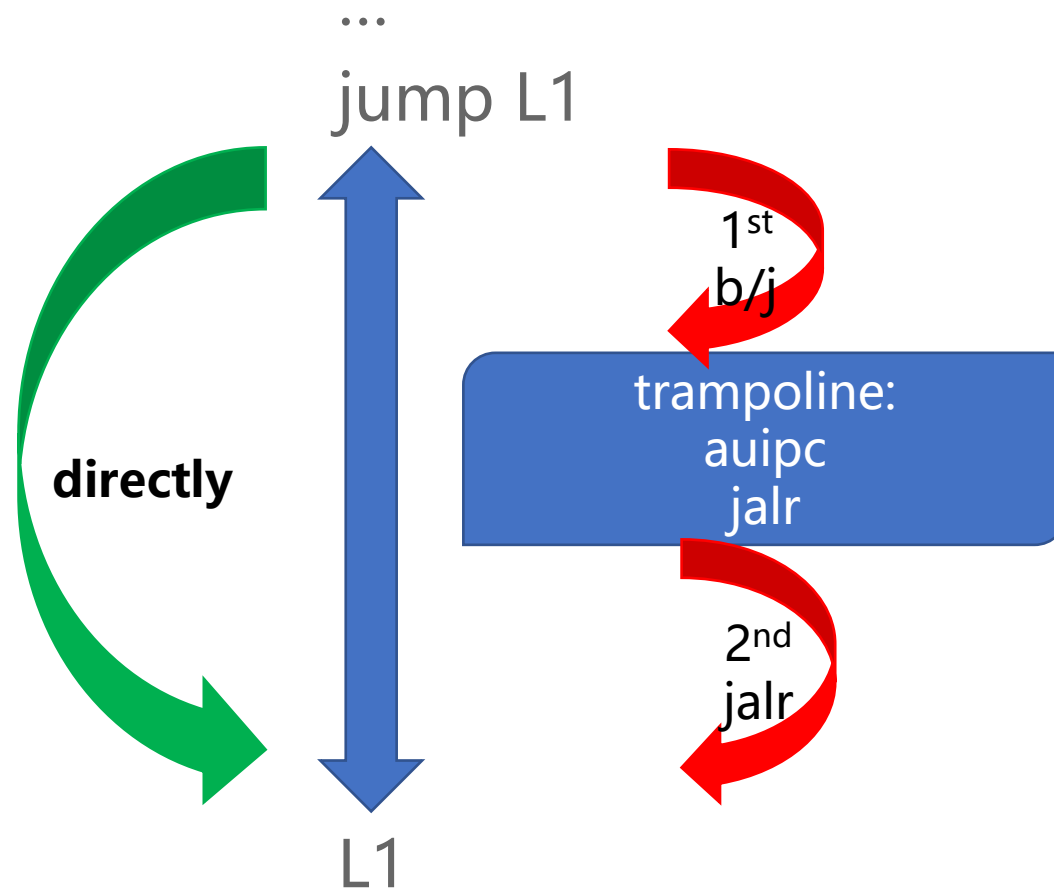


Feedback2: short offsets of the direct branch and jump

Trampoline must be emitted

When the distance is within the offset range of jal and beq.

Else, trampoline is emitted.

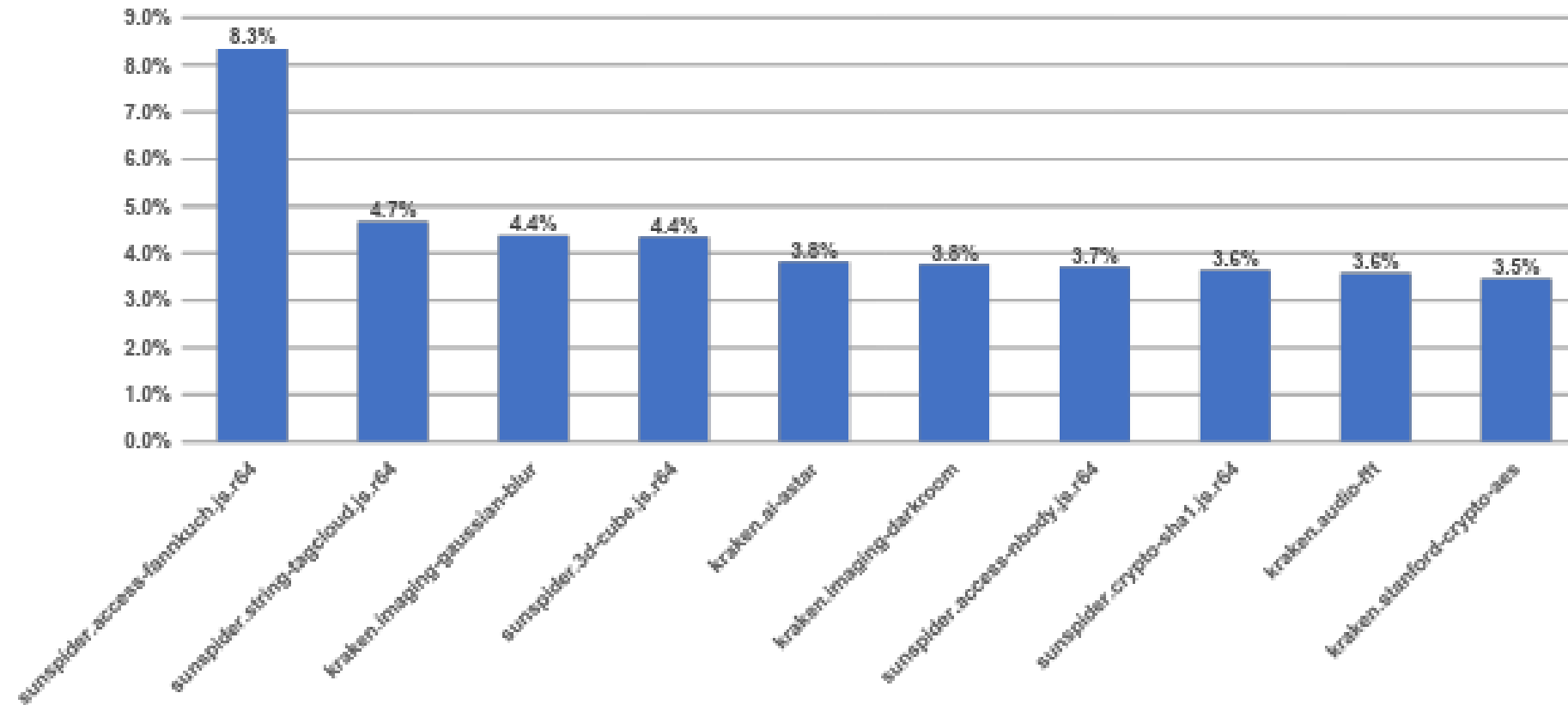


penalty:

1. codesize+
+
2. branch
(prediction miss)++

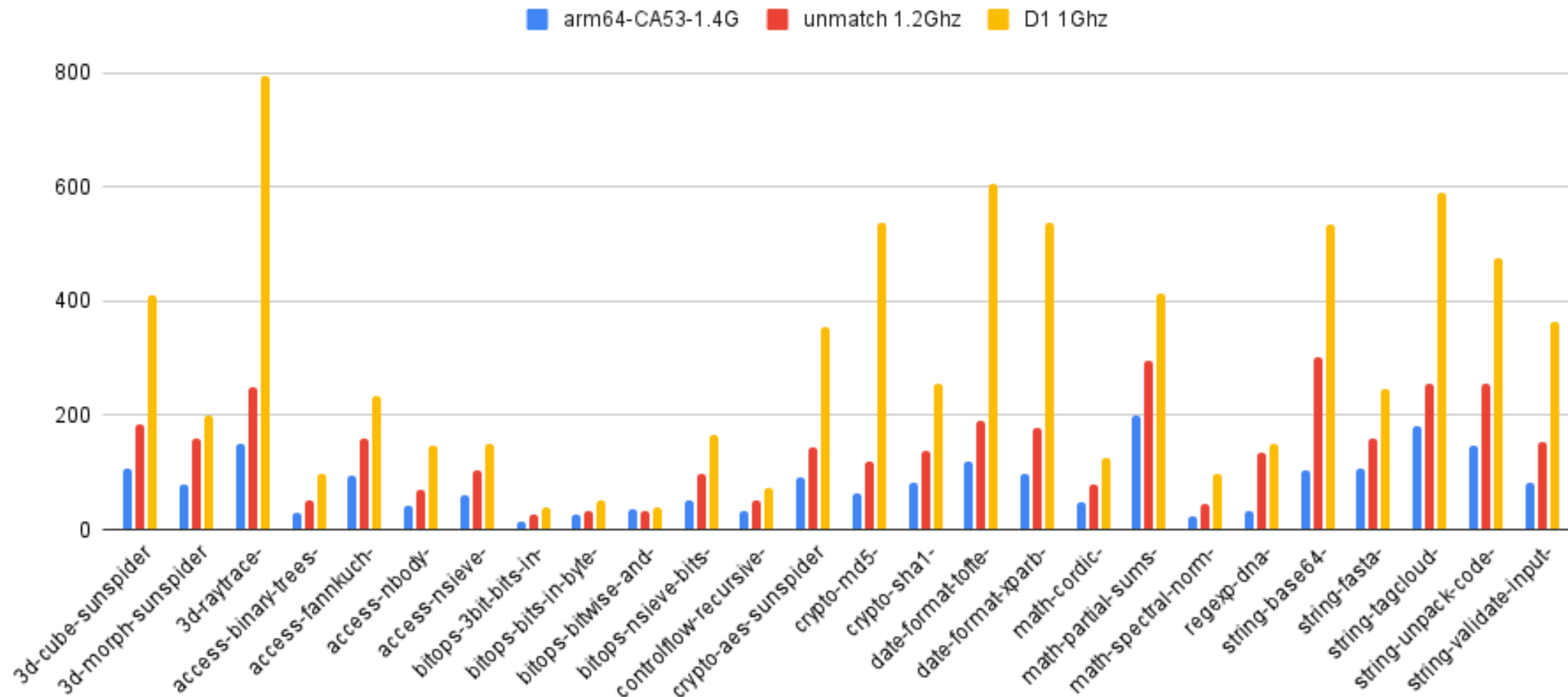
trampolines result in larger code size

static code size ratio of trampoline
on RISC-V














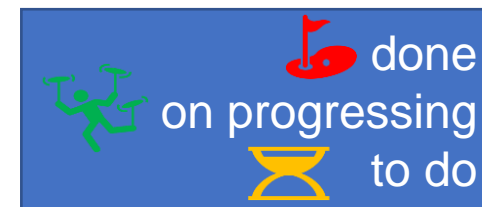
V8 Benchmark score

V8 Benchmark score



V8 for RISC-V Roadmap-Items lists with priority from high to low

- Upstreaming(now porting and upstreaming, keeping pace with master to ensure successfully build) 
- Demo and application launch
 - Demo tensorflow.js on RISC-V (a very simple cmd line demo is done by the end of Q1) 
 - Embrace node.js community and support all the node.js ecosystem running on RV64GC (upstreaming is launching by the end of Q1)  <https://github.com/v8-riscv/node/>
 - Chromium running on PicoRio with V8(on going by RIOS lab) 
- Performance (on going by PLCT)
 - Set up the performance tracking mechanism. (<https://ci.rvperf.org/view/V8/> is on progressing) 
 - Speed up (blind say 5x) on RV64GC compared with the code we first upstreaming 
- ISA-extension
 - C (on going by RIOS lab) 
 - V (on going by PLCT) 
 - B/P(todo) 
- Ecosystem adaption
 - build v8 with LLVM(todo) 
- Community develop and enlarge (on going by PLCT)
 - resurrecting more than 10 new interns/contributors/graduates into our project. 



Q&A

Thanks to All the Contributors and Community Members!



THANK YOU

Join us:

<https://github.com/riscv/v8/wiki>

<https://github.com/plctlab/weloveinterns>