# Staking on Power Efficient and Low Cost Hardware

From ARM64 to RISC-V Boards

aavegotch1.eth, haurog

Independent community members

# Scope of the presentation - Our experience

Home stakers / node operators:

- Simplest entry point hardware
- Cheap / power efficient (ARM64)
- Promising future option (RISC-V boards)

Software Engineer view on RISC-V:

- Why RISC-V ISA
- Current state of Software / Hardware
- Benchmarking ARM64 / RISC-V hardware

# Inception and Collaborators

We are solo home stakers / node operators for 4+ years.
We run our nodes on commodity hardware and data-center grade servers.
This July, aavegotch1.eth made a call to bring nodes to RISC-V CPUs.
`Ethereum on ARM` team also investigated RISC-V as a possibility.

Involved people:

- Diego, Ethereum on ARM
- Fernando, Ethereum on ARM
- Leonardo Arias, works at flashbots
- aavegotch1.eth, independent
- Haurog, independent

# ARM64 boards

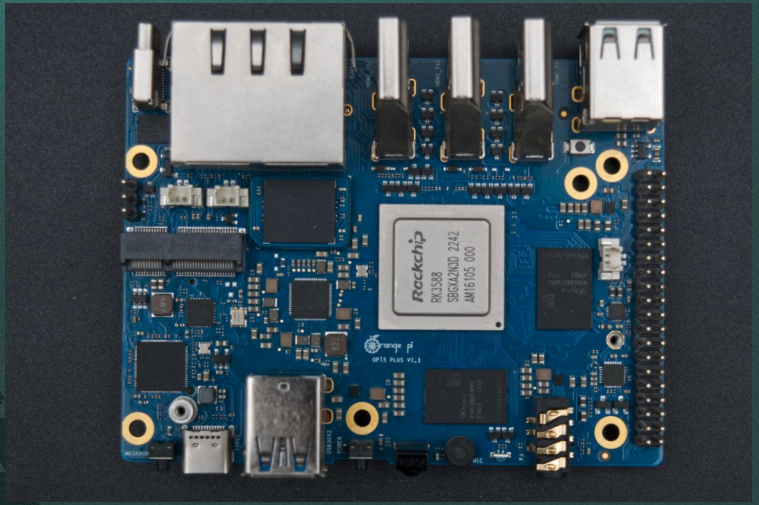
# ARM64 boards vs Intel NUCs

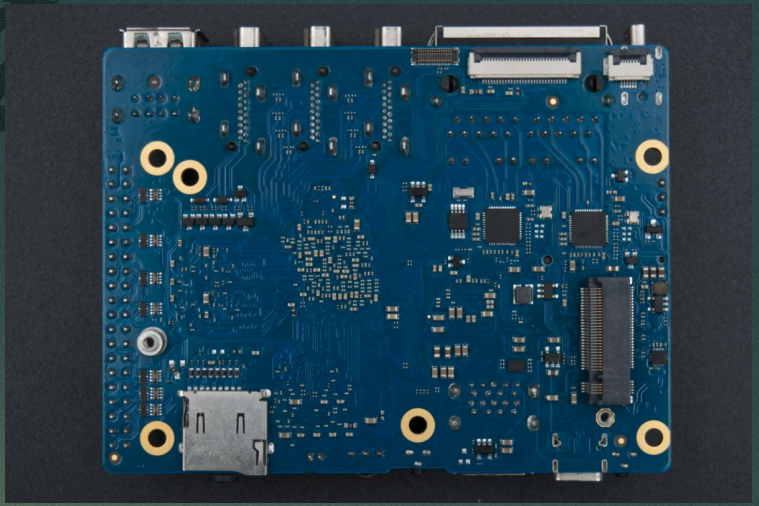
Advantages:

- All-in-one board
- Low power consumption
- Well balanced hardware

Disadvantages:

- Steeper learning curve
- Software support not on par
- Availability



# How about a Raspberry Pi 5?

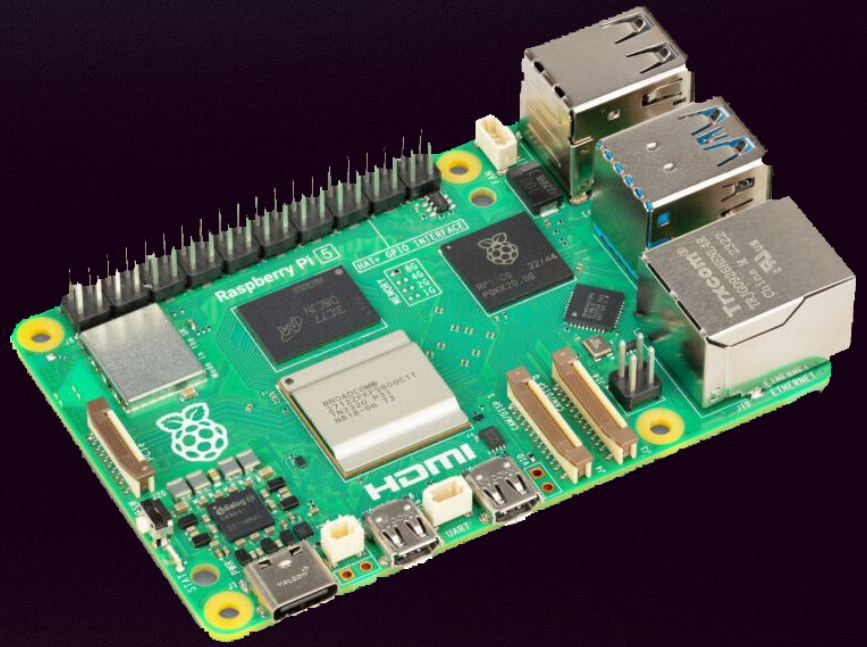
Pros:

- Widely available
- Cheap
- Good Linux support

Cons:

- Only 8 GB RAM
- CPU slower on multi-core
- Need additional board to attach NVMe ssd

-> Possible to run a node, but not the best option nowadays.



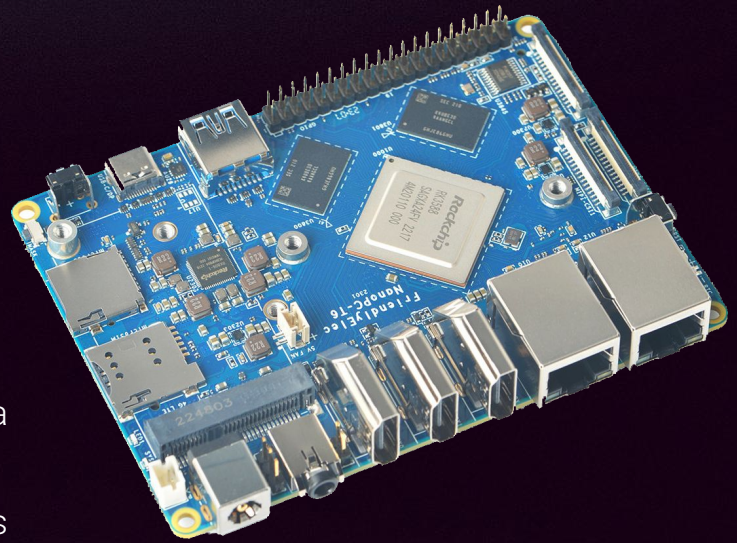
# Good ARM64 boards available

The following boards use the same RK3588 CPU:

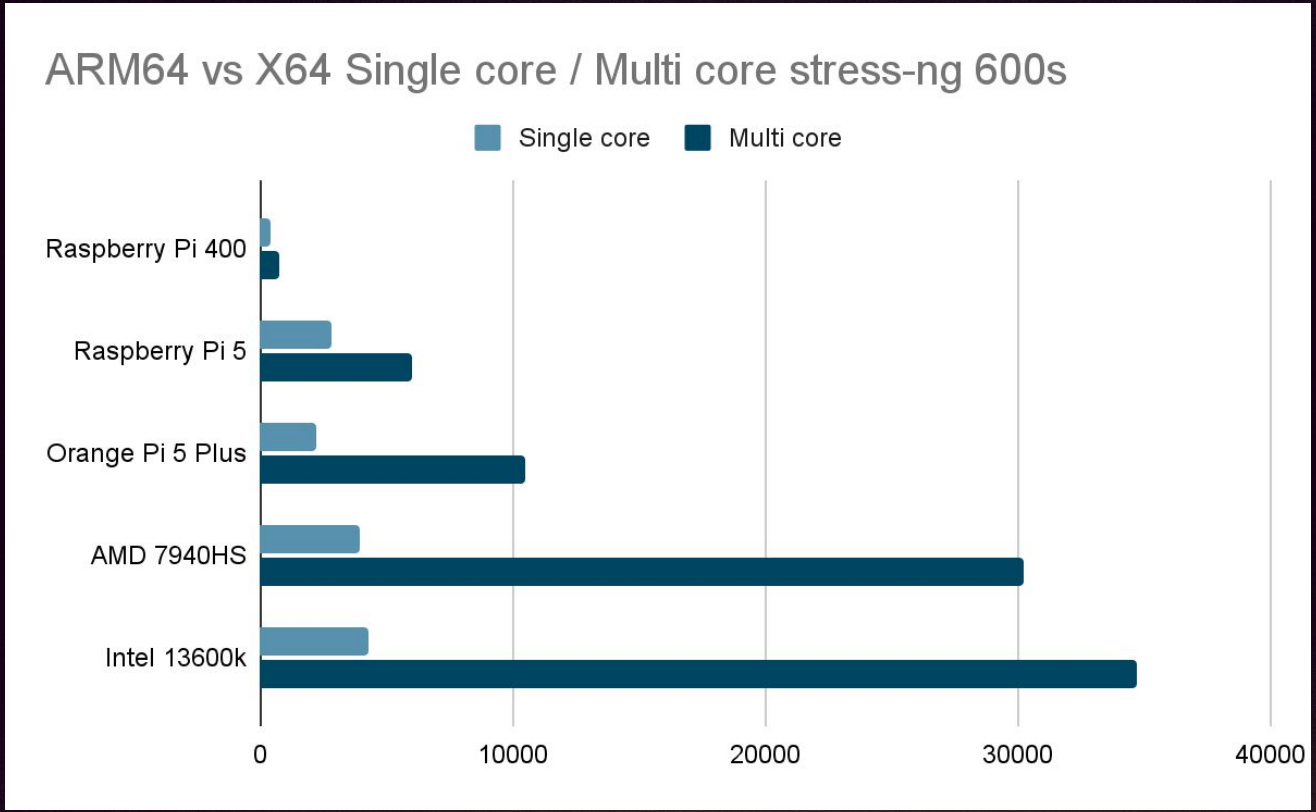
**Orange Pi 5 Plus**: 16 or 32 GB RAM, 120-150$
**Nano PC T-6**: 16 GB RAM only, 140$
**Radxa Rock 5B**: 16 or 32 GB RAM, 140-230$

**Always buy with the ‘official’ power supply.**
Will have to add an NVME SSD, a fan and maybe a case.
Use a Linux specifically built for these boards

- Joshua Rieks Ubuntu for Rockchip.
- Ethereum on ARM



# CPU Performance comparison



# Running a validator node

Used eth-docker to run mainnet on Nethermind/Nimbus.
Node only:

- Node used 5-8 Watt
- Cpu usage 10-25%
- RAM usage ~50%

As a validating Node:

- Power usage slightly higher, but similar
- CPU usage increased to 25-40%
- RAM usage the same

Same attestation efficiency as on more powerful AMD Node.
[Using a cheap 8 Core ARM minicomputer as a full node]

## The ARM boards can easily handle the load of an Ethereum node.

# Tips and tricks

Case required - dust or unwanted electrical short-circuits

Active cooling required - the CPU will consume less power if the temperature is nominal and also work at higher clock speeds

NVME:

- Use the smallest one needed for the job
- 4TB ones have double the chip surface area vs the 2TB, they need active cooling
- PCI Gen 4 or Gen 5 have higher clocks and need even better cooling
- The power consumption can get high (3 to even 15W of power draw), if the power budget is not enough, the board will randomly reboot / freeze with no logs

Overclock the CPU - it is worth it, as the staking is sometimes a spiky workload

## More technical things worth knowing.

# Day 2 operations

- Use a well maintained OS
- Kernel live patching
- Use a well maintained staking framework (Dappnode, eth-docker, eth on arm)
- Firewall-proof your system (ufw, endless-ssh, randomize SSH port)
- VPN for management (wireguard, openvpn, tailscale)
- Monitoring and alerts
- Invest in a small UPS (Ex: APC 750VA - 150$)

**More and more technical things worth knowing.**

# Summary of ARM64 boards

The discussed ARM boards are the right choice if you want to:

| | |
|---|---|
| run a solo home validator | 🤔 |
| have a fallback node | ✅ |
| run an Ethereum node | ✅ |
| run a testnet node | ✅ |
| play with hardware | ✅ |

## ARM64 is good enough


RISC-V boards

# What is RISC-V?

- Young alternative to x64 / ARM64
- Standards already in place
- Free Open Source - free to use and contribute
- Royalty free
- All the above should foster quicker iteration and innovation, leading to cheaper computational hardware

**The Linux of the CPU architectures!**

# RISC-V hardware / software timeline

## The inception

RISC-V has started as an university project in 2010. Small volume CPU / SOC designs have been created and proven to work. A foundation to foster the project has been created, and most of the ISA has been standardized.

## The big effort. We are here.

Achieve hardware performance and efficiency parity with ARM64 and X64. Software stack has become usable in most of the big frameworks and OSses, with work required to make the software more efficient and to use all the ISA capabilities.

## Mainstream usage

RISC-V hardware will become ubiquitous, used in multiple industries, as a 1:1 alternative to ARM64/X64, where the switch will be seamless. Software support will be on par with the other architectures.

# RISC-V Boards

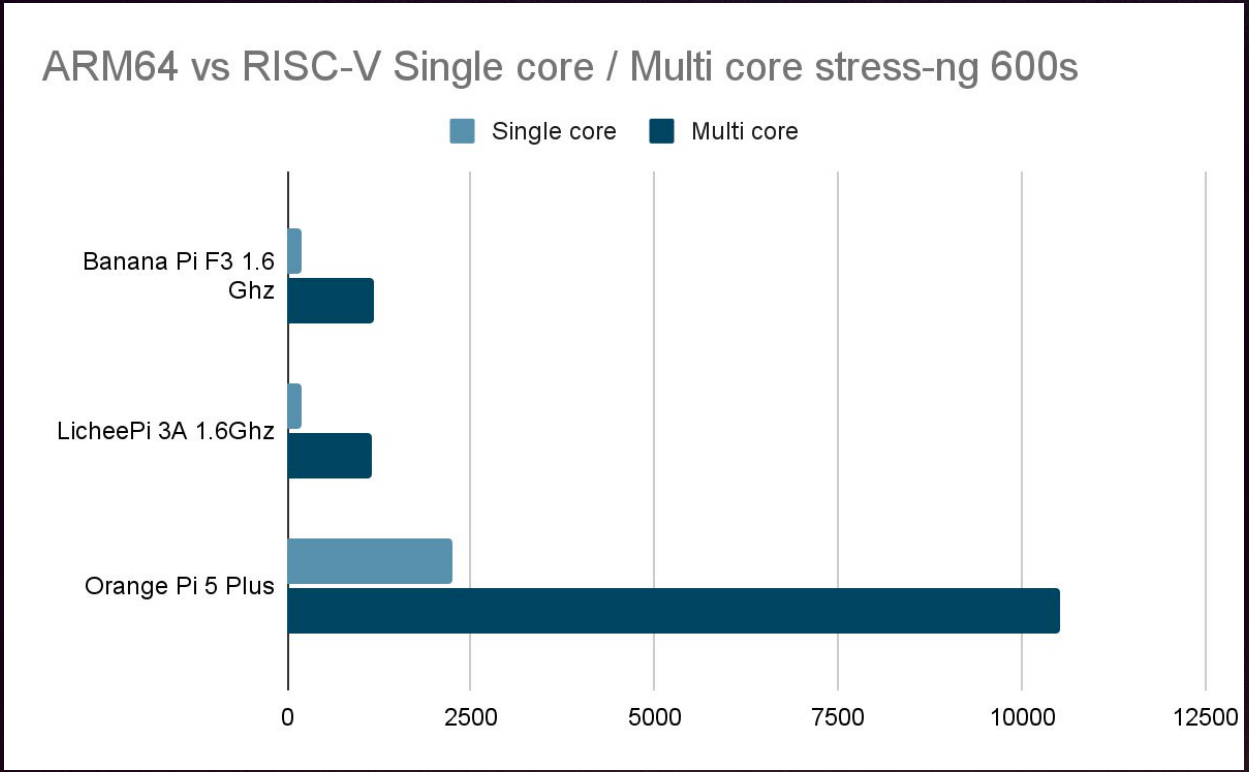
This year, a new iteration of RISC-V boards appeared on the market, promising big performance and efficiency improvements vs the previous iteration.

Boards we tested:

**LicheePi 3A**: 16GB RAM, 160$
**Banana PI F3**: 16GB RAM, 130$

## Can I run a staking node on a RISC-V board?

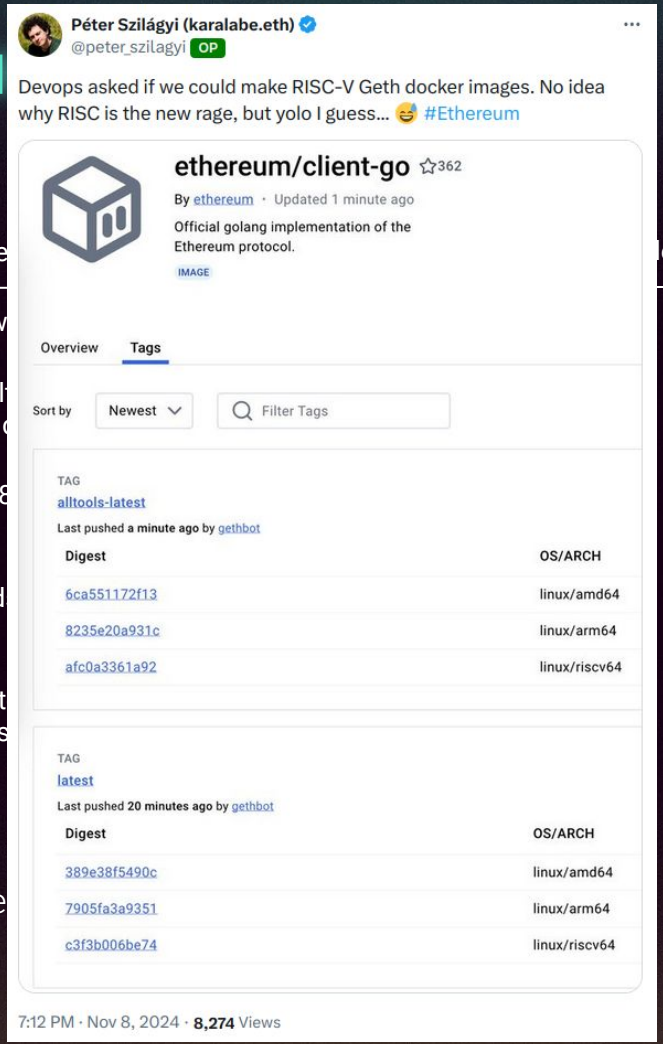
# CPU Performance comparison



# Clients and Softwa

| Client | Compile | Run | Comme |
|---|---|---|---|
| Geth | ✅ | ✅ | It just w |
| Besu | ✅ | ❌ | Segfaul<br>a Besu |
| Nethermind | ❌ | ❌ | dotnet 8<br>RISC-V |
| Erigon | ❌ | ❌ | Depend<br>code |
| Reth | ✅ | ❌ | Needs t<br>address |

-> Many clients still ne
run on RISC-V




| Run | Comment |
|---|---|
| ✅ | Add compile target, now in main release |
| ✅ | Only from forked repo, not yet upstreamed |
| ❌ | Build environment not working |
| ❌ | Missing libraries |
| ❌ | Issues with dependencies |
| ❌ | From forked repo, Unspecified error |
| ? | Thanks to flashbots team |

# Staking Ethereum on RISC-V

Both clients on RISC-V board

| Clients | Ethereum | Holesky | Sepolia | Chiado |
|---|---|---|---|---|
| Nimbus + geth | ❌ | ❌ | ✅ | ? |
| Lighthouse + geth | ? | ❌ | 0 Peers | ✅ |

Split setup (one client on RISC-V)

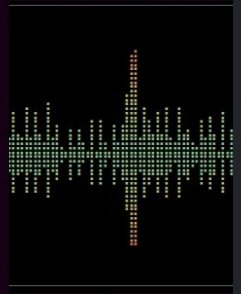
| Clients | Ethereum | Holesky | Sepolia | Chiado |
|---|---|---|---|---|
| Nimbus | ❌ | ❌ | ✅ | ? |
| Lighthouse | ❌ | ❌ | 0 Peers | ✅ |
| geth | ✅ | | | |

-> Node works with small networks (Validator count)

I can run a staking node on a RISC-V board, but...

# Resource usage for validator on Chiado

Normal ope



CPU: ~10-3(

RAM: 4 GB

-> RISC-V

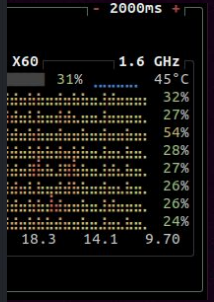
needs to d

## Validator 7481: Slots

Home / Validators / 7481 / Slots

Show 50 entries

| Epoch | Slot | Status | Time | Proposer | Attestations | Deposits / Exits | Slashings P / A | Tx Count | Sync Agg % | Graffiti |
|---|---|---|---|---|---|---|---|---|---|---|
| 820414 | 13126634 | Proposed | 12 hr. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 0 | 98.44% | RISC-V for the win |
| 819713 | 13115423 | Proposed | 1 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 1 | 98.63% | RISC-V for the win |
| 818885 | 13102162 | Proposed | 1 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 2 | 98.24% | RISC-V for the win |
| 818802 | 13100838 | Proposed | 2 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 6 | 98.24% | RISC-V for the win |
| 817239 | 13075839 | Proposed | 3 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 3 | 97.27% | RISC-V for the win |
| 817224 | 13075597 | Proposed | 3 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 0 | 97.27% | RISC-V for the win |
| 816962 | 13071398 | Proposed | 3 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 3 | 97.46% | RISC-V for the win |
| 816919 | 13070718 | Proposed | 3 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 0 | 97.46% | RISC-V for the win |
| 816198 | 13059181 | Proposed | 4 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 3 | 99.02% | RISC-V for the win |
| 816002 | 13056044 | Proposed | 4 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 6 | 98.05% | RISC-V for the win |
| 815975 | 13055606 | Proposed | 4 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 2 | 98.05% | RISC-V for the win |
| 815912 | 13054605 | Proposed | 4 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 1 | 98.05% | RISC-V for the win |
| 815293 | 13044693 | Proposed | 5 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 0 | 97.46% | RISC-V for the win |
| 815082 | 13041319 | Proposed | 5 day. ago | 7481 | 2 | 0 / 0 | 0 / 0 | 0 | 97.85% | RISC-V for the win |



# RISC-V required improvements

- Faster Hardware
  - CPU
  - NVMe Interface
- Clients
- Node setup tools (Eth-docker, Sedge etc.)
- Cryptography libraries need assembly implementation for RISC-V

**Low-level crypto primitives optimization needed.**

# RISC-V in the Ethereum EVM

The beacon chain to be implemented using the RISC-V instruction standard.

In theory, this could lead to having direct instruction execution on RISC-V boards, leading to almost native performance in execution.

# Current status of RISC-V boards

Are current RISC-V boards the right choice if you want to:

| | |
|---|---|
| run an Ethereum node | ❌ |
| run a testnet node | 🤔 |
| play with hardware | ✅ |
| improve clients | ✅ |
| write code for RISC-V | ✅ |
| have a glimpse into the future | ✅ |

**RISC-V boards are usable, but they are not yet in their prime just yet.**

# Ethereum future - home staker

- 4TB requirement soon to become a reality
- Stateless nodes - Verkle - might require more CPU power
- Lower the staking requirement to 1 ETH - a perfect fit for small boards?
- Maximum Ethereum Balance of the Validator to increase from 32 (less performance required as you need less validators)

## The future looks bright...?

# Conclusion

There is a wider range of power efficient / low cost platforms available to run your nodes on.
Some of them might fit a home staker need today.
Others might do so in the coming years.

Contributions welcome for RISC-V!

- Execution / consensus clients support (dependency libs)
- ASM implementation of crypto primitives
- Verify more RISC-V boards

**The Hardware space is much larger than NUCs and has an even brighter future.**

# Useful links

Main issue thread:
https://github.com/eth-educators/eth-docker/issues/1873

Hacks for RISC-V:
https://github.com/lazyprogrammerio/eth-docker-docs/blob/main/website/docs/Usage/OtherArches.md

RISC-V as native ETH implementation:
https://vitalik.eth.limo/general/2024/10/26/futures5.html
https://verified-zkevm.org/
https://github.com/risc0/risc0

# More RISC-V Boards available now

Current Generation 16GB RAM:
Milk-V Jupiter: 115$ (probably best option)

Previous Generation (not good enough):
Sipeed Lichee Pi 4a (no NVMe ssd)
VisionFive 2 (only 8GB RAM)
HiFive Unmatched (overall very slow)

High end last gen:
Milk-V Pioneer: 64 cores and ~10 times more expensive than the other options.

Boards available now are close to the min spec for running nodes and they are only getting faster. Perfect time to get software ready now before hardware is ready in 1-3 years.

# Boards coming soon:

Near term (this year):
HiFive Premier P550: 599$, dev boards available
Milk-V Megrez: ?$
Lichee Pi 5a: ?$.
StarPro64: ?$
-> all the same CPU. Hopefully price comes down for other boards. Faster single core speed compared to current gen boards, but only 4 cores. Do not support vector instruction standard

Next year (maybe):
Milk-V Oasis: 140$,
-> 16 cores which some people expect to reach levels of the ARM boards used for node running.