

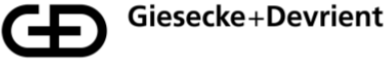
The Silicon Commons

Build Together, Build Well & Build Securely

Dr. Gavin Ferris, CEO @ lowRISC CIC

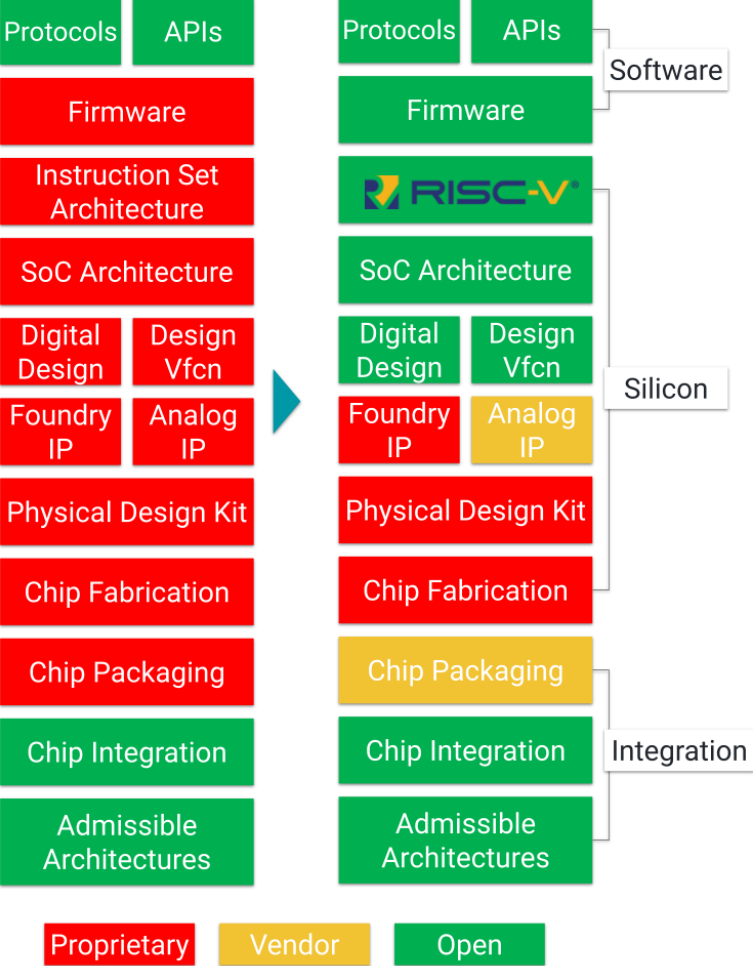


Open ISA to Open Source Silicon



Other Silicon

opentitan



Commercial-Grade Open Source Silicon Is Here

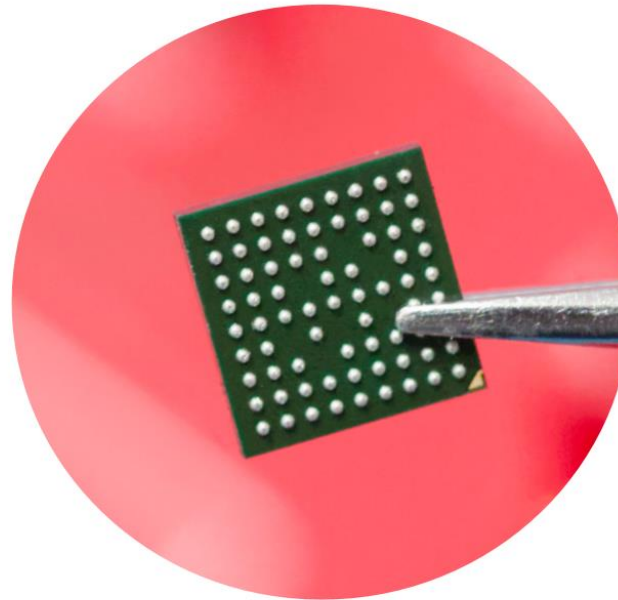
*“Nuvoton Technology Corporation [...] announced today that **Google’s ChromeOS plans to use the first commercial chip built on the OpenTitan** open source secure silicon design as an evolution of its security chip **for Chromebooks.**”*

Nuvoton, May 2024

nuvoTon

 **opentitan**

 **chromebook**



*“**Hardware security is something we don’t compromise on.** We are excited to partner with the dream team of Nuvoton, a valued, historic, strategic partner, and lowRISC, a leader in secure silicon, to maintain this high bar of quality.”*

Prajakta Gudadhe
Sr Director, ChromeOS Platform Engineering

<https://www.nuvoton.com/news/news/all/TSNuvotonNews-000514>

The Silicon Commons[®] Approach



Code review and approval process



Open development with clear IP provenance



Governance structure



Continuous Integration testing



Training for contributors



Accessible verification collateral



Extensive documentation



Permissive licensing to encourage re-use

World's Most Active Open Silicon Project

RTL · design verification collateral · documentation · low-level firmware · tests

25,000+
total commits
(Ibex + OpenTitan)

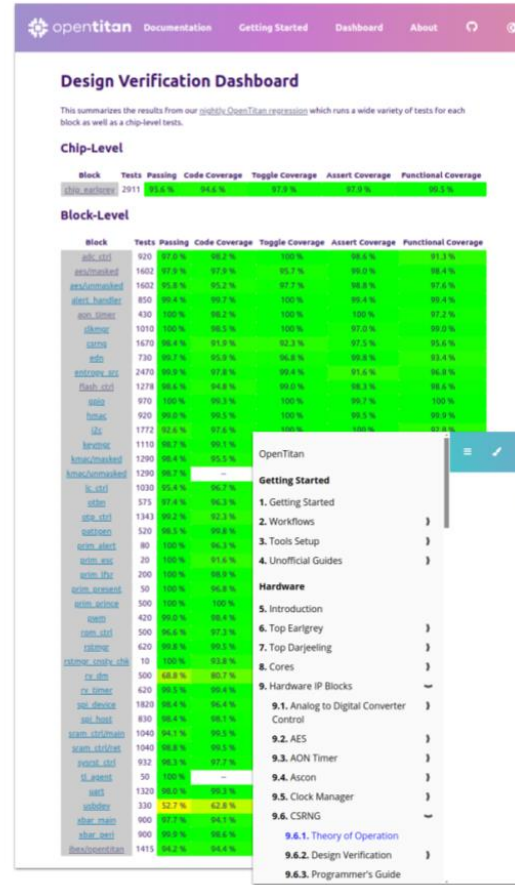
250+
contributors
(Ibex + OpenTitan)

7,200+
GitHub issues
(Ibex + OpenTitan)

3,700+
GitHub stars
(Ibex + OpenTitan)

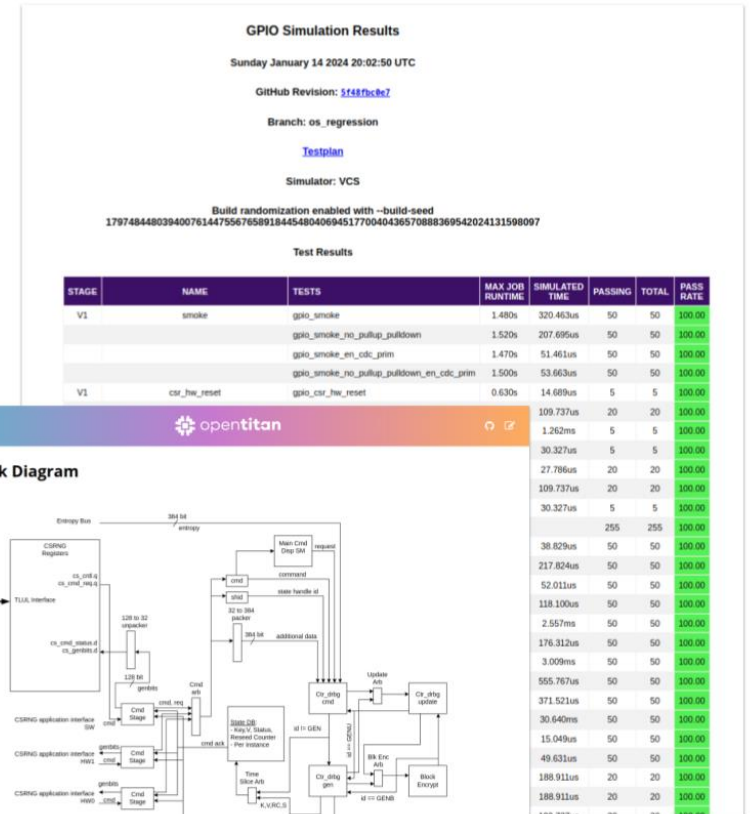
440,000+
lines of SystemVerilog
(Digital Design and Verification for
Ibex + OpenTitan)

40,000+
test runs in nightly
regressions
(run multiple times per
week)



The dashboard shows test results for various blocks. Key statistics include:

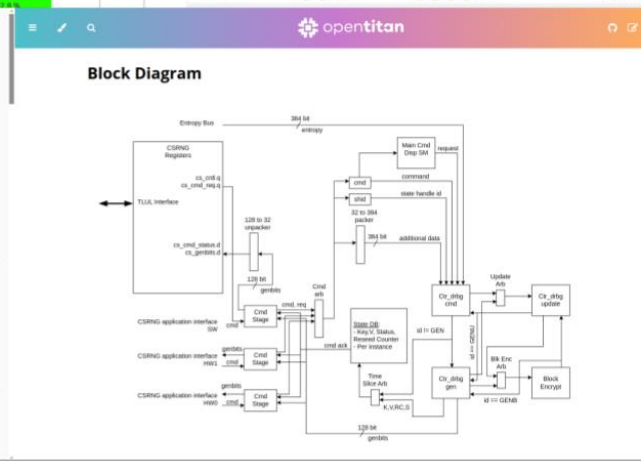
- Chip-Level:** 2911 tests, 91.8% passing, 94.6% code coverage, 97.8% toggle coverage, 97.8% assert coverage, 98.2% functional coverage.
- Block-Level:** A table listing 49 blocks with their respective test counts and coverage metrics. For example, 'ibex_csr' has 1330 tests, 92.7% passing, and 97.3% code coverage.



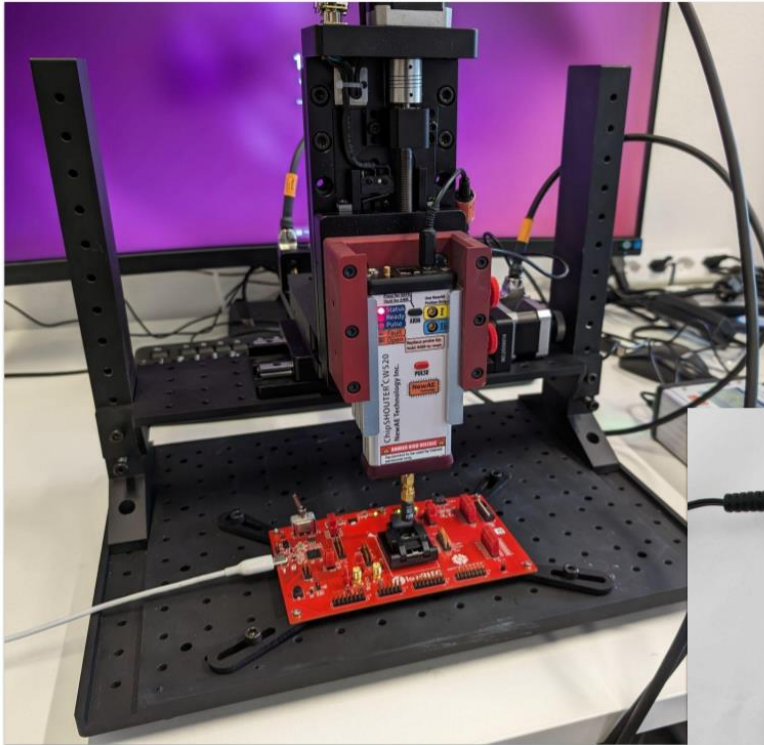
GPIO Simulation Results for Sunday January 14 2024 20:02:50 UTC. The test plan includes:

- Branch: os_regression
- Simulator: VCS
- Build randomization enabled with --build-seed 1797484480394007614475567658918445480406945177004043657088369542024131598097

STAGE	NAME	TESTS	MAX JOB RUNTIME	SIMULATED TIME	PASSING	TOTAL	PASS RATE
V1	smoke	gpi0_smoke	1.480s	320.463us	50	50	100.00
		gpi0_smoke_no_pulldup_pulldown	1.520s	207.695us	50	50	100.00
		gpi0_smoke_en_cdc_prim	1.470s	51.461us	50	50	100.00
V1	csr_hw_reset	gpi0_smoke_no_pulldup_pulldown_en_cdc_prim	1.500s	53.663us	50	50	100.00
		gpi0_smoke_no_pulldup_pulldown	0.630s	14.689us	5	5	100.00
		gpi0_csr_hw_reset	109.737us	20	20	100.00	



Build Securely



README Apache-2.0 license

ot-sca - Side-Channel Analysis & Fault Injection Setup for OpenTitan

About the repository

This repository contains infrastructure code useful for performing side-channel analysis (SCA) and fault injection (FI) attacks for [OpenTitan](#).

See [getting started](#) for instructions.

How to contribute

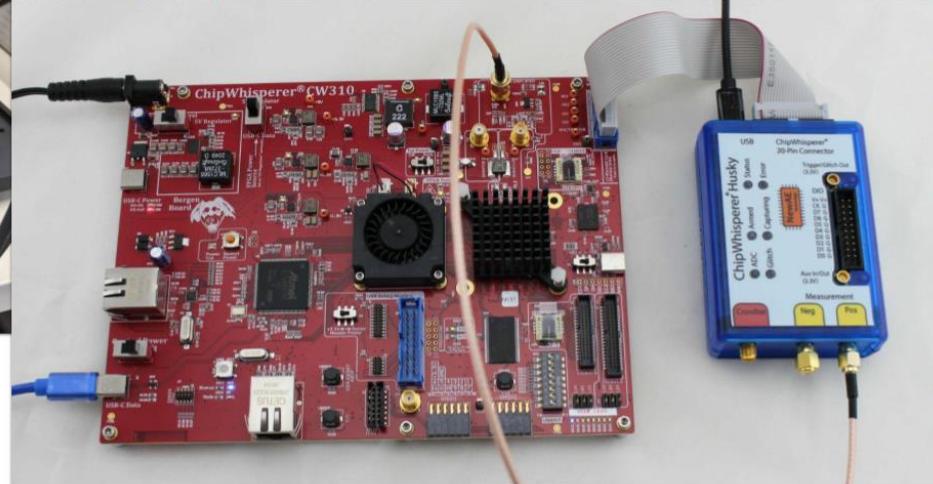
Have a look at [CONTRIBUTING](#) for guidelines on how to contribute code to this repository.

Licensing

Unless otherwise noted, everything in this repository is covered by the Apache License, Version 2.0. See [LICENSE](#) for full text).

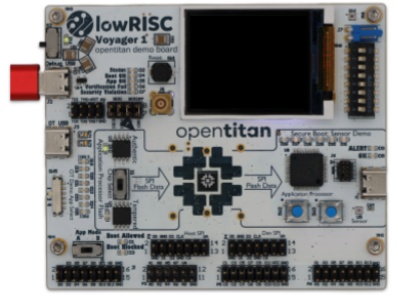
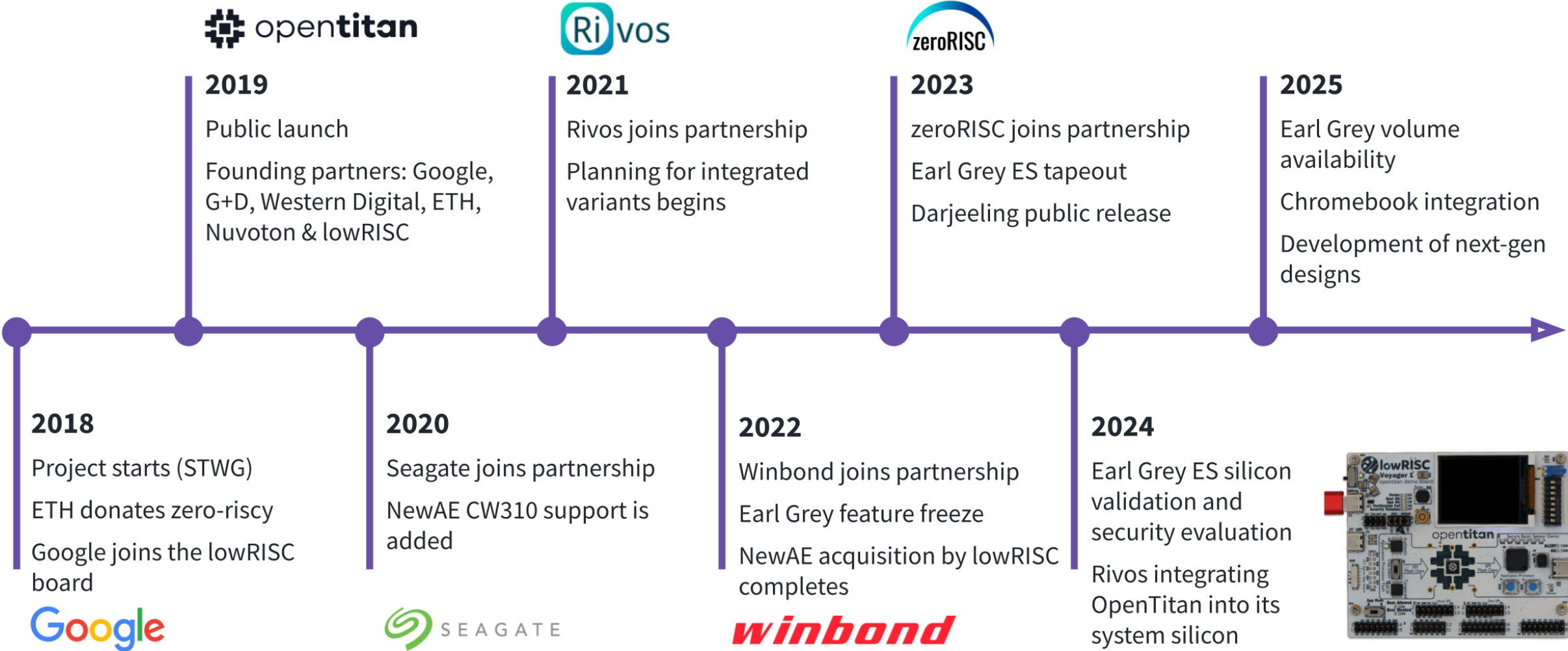
References

```
C verify.c x
opentitan > sw > device > silicon_creator > lib > sigverify > sphincsplus > C verify.c > spx_verify(const uint32_t*, const uint8_t*, size_t, const uint8_t*)
1 // Copyright lowRISC contributors (OpenTitan project).
2 // Licensed under the Apache License, Version 2.0, see LICENSE for details.
3 // SPDX-License-Identifier: Apache-2.0
4 //
5 // Derived from code in the SPHINCS+ reference implementation (CC0 license):
6 // https://github.com/sphincs/sphincsplus/blob/ed15dd78658f63288c7492c00260d86154b84637/ref/sign.c
7
8 > #include "sw/device/silicon_creator/lib/sigverify/sphincsplus/verify.h"
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23 static_assert(kSpxVerifySigWords * sizeof(uint32_t) == kSpxVerifySigBytes,
24              "kSpxVerifySigWords and kSpxVerifySigBytes do not match.");
25 static_assert(kSpxVerifyPkWords * sizeof(uint32_t) == kSpxVerifyPkBytes,
26              "kSpxVerifyPkWords and kSpxVerifyPkBytes do not match.");
27 static_assert(kSpxD <= UINT8_MAX, "kSpxD must fit into a uint8_t.");
28 rom_error_t spx_verify(const uint32_t *sig, const uint8_t *msg_prefix_1,
29                       size_t msg_prefix_1_len, const uint8_t *msg_prefix_2,
30                       size_t msg_prefix_2_len, const uint8_t *msg,
31                       size_t msg_len, const uint32_t *pk, uint32_t *root) {
32     spx_ctx_t ctx;
33     memcpy(&ctx.pub_seed, pk, kSpxN);
34
35     // This hook allows the hash function instantiation to do whatever
36     // it needs, based on the public seed.
37     ix_hash_initialize(&ctx);
38
39     ldr = {0};
40     ldr = {0};
41     .addr = {0};
42     lr, kSpxAddrTypeWots;
43     lr, kSpxAddrTypeHashFree;
44     .addr, kSpxAddrTypeWotsPk;
45
46     t and leaf index from R || PK || H.
47     a result of the hash domain separator.
48     res);
49
50     ix_hash_message(
51     msg_prefix_1_len, msg_prefix_2, msg_prefix_2_len,
52     tree, &idx_leaf);
53
54     to 0, so no need to set_layer_addr.
55     lr, tree);
56     .addr, idx_leaf);
57
58     , &ctx, &wots_addr, root);
```

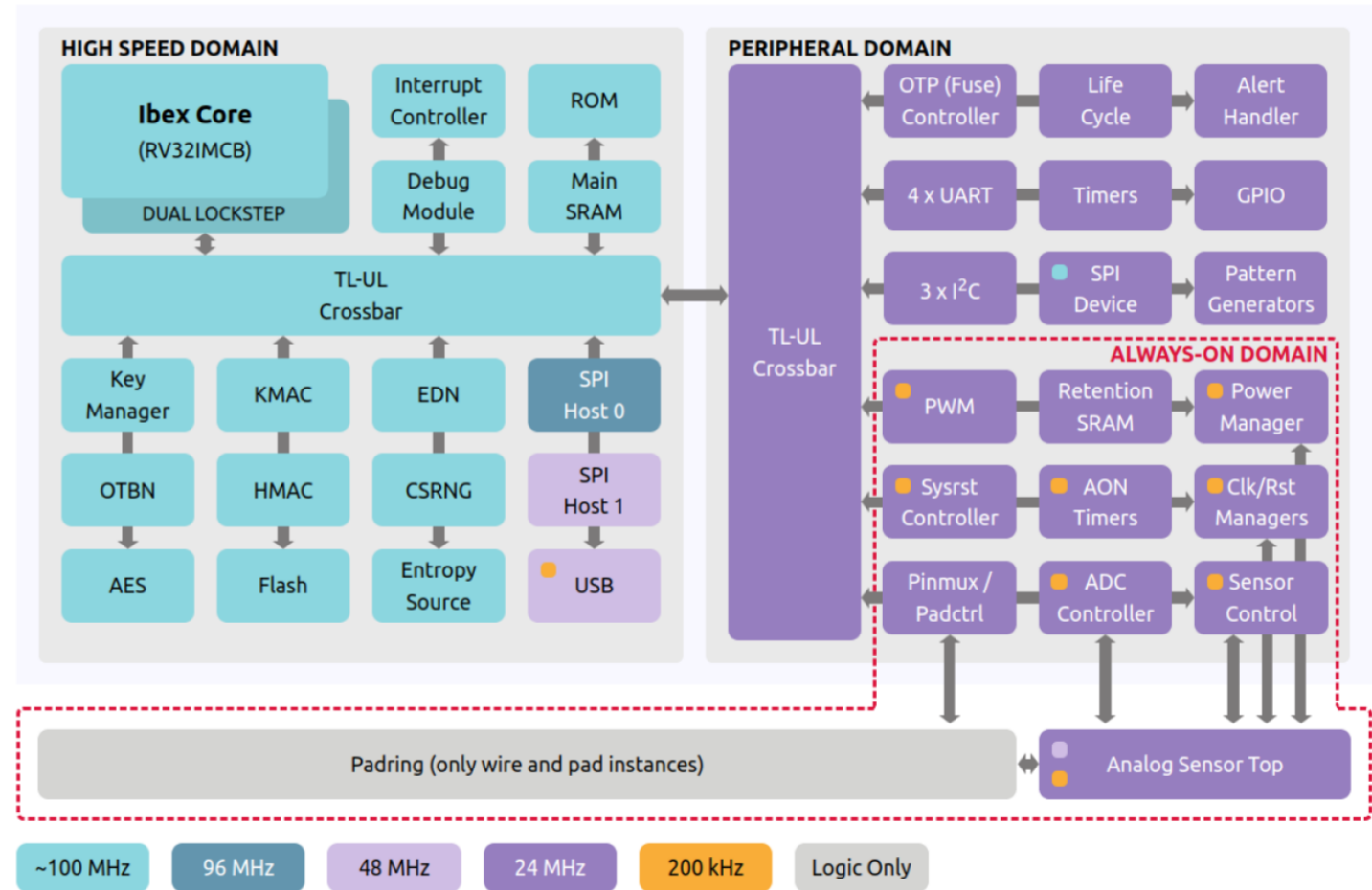
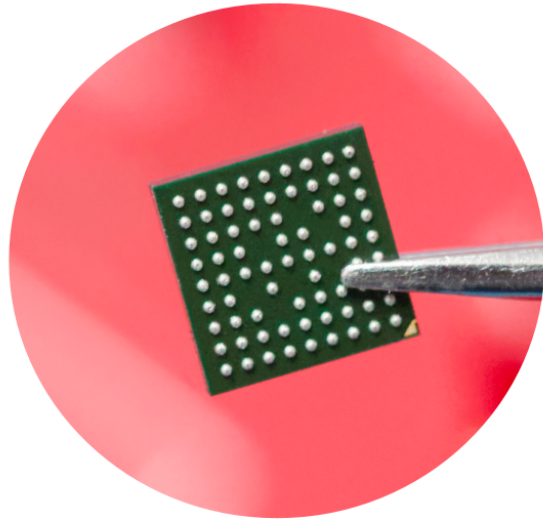


PQC secure boot
(SPHINCS+)

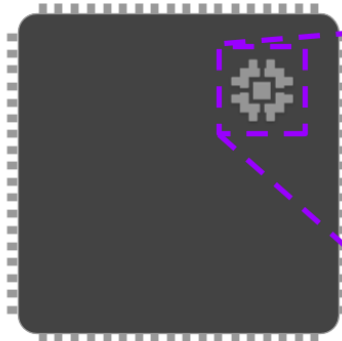
Timeline



The First OpenTitan[®] Chip Design: Earl Grey



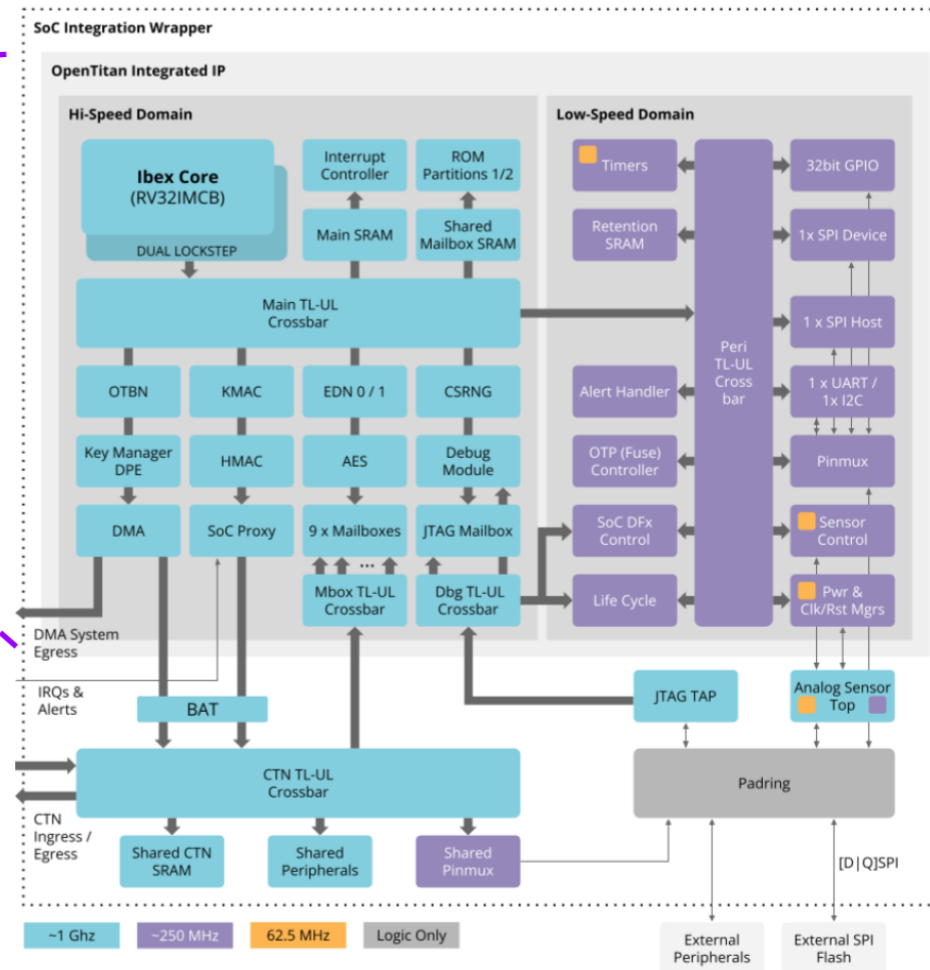
Integrating OpenTitan[®]: Darjeeling and Beyond



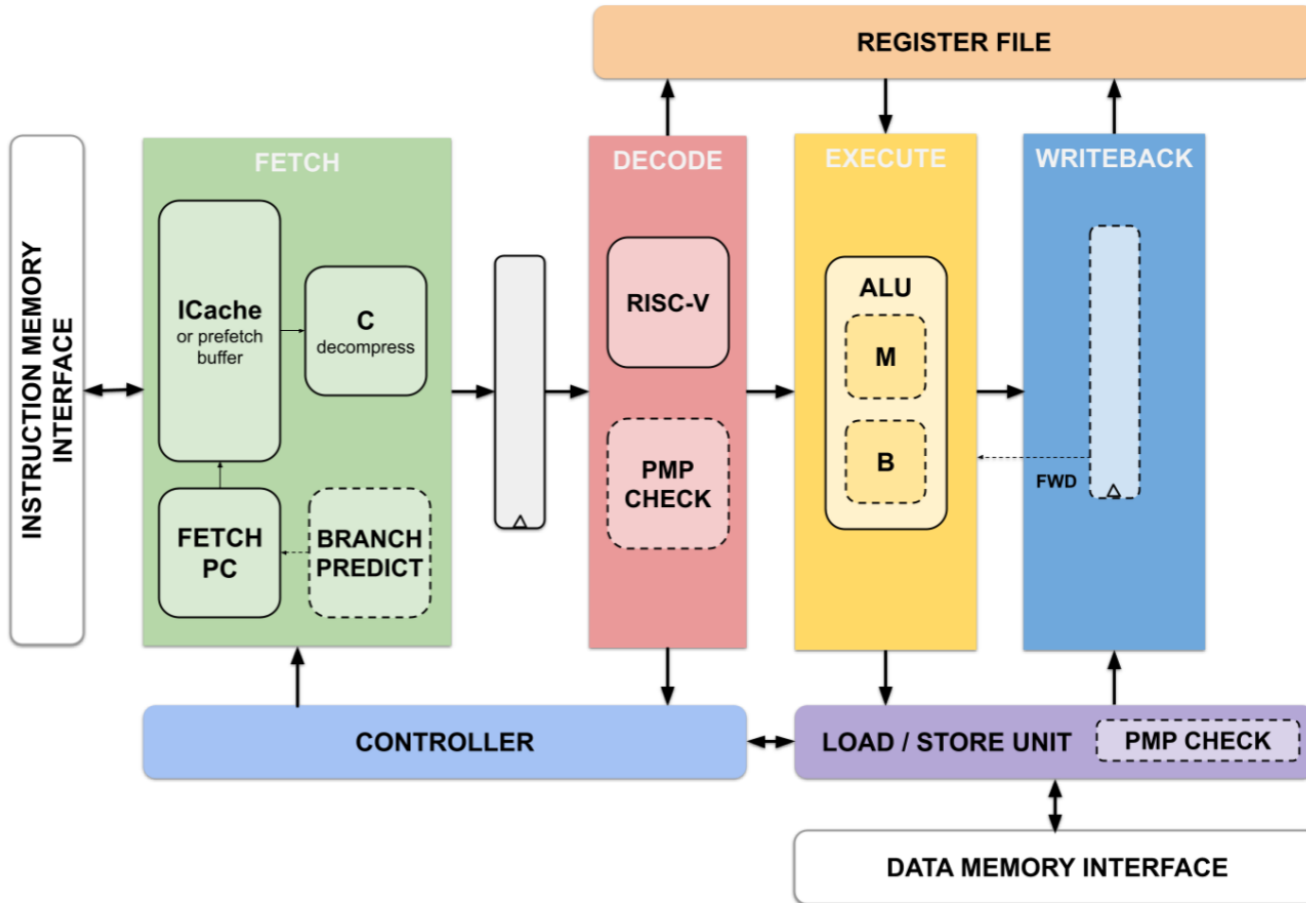
Third-party SoC or chiplet

*“With security as a foundational pillar, Rivos is integrating a RISC-V based root of trust solution, **OpenTitan**, directly into its system silicon.”*

<https://rivos.com/technology>



RISC-V at the Core: Ibex[®]

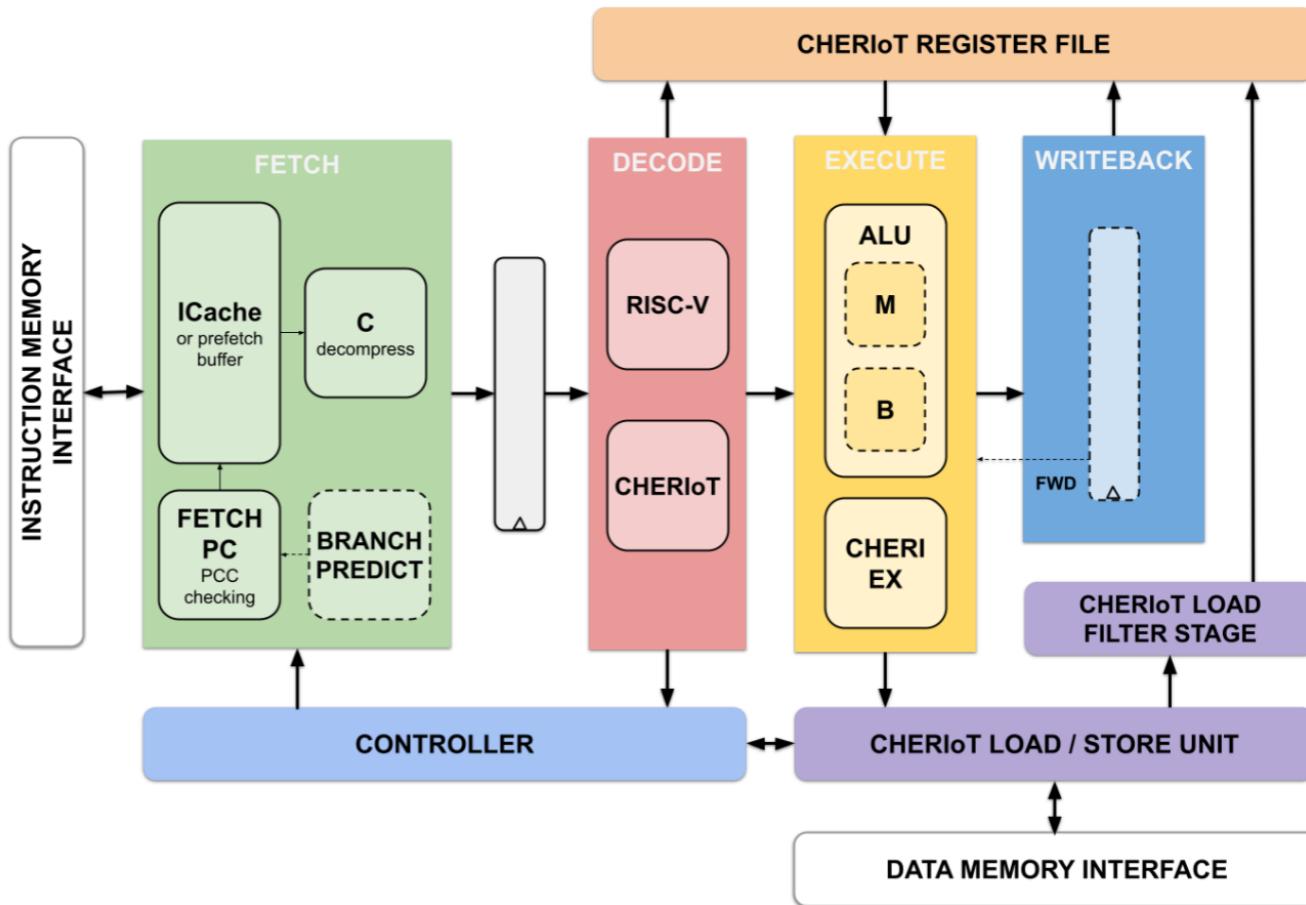


<https://github.com/lowRISC/ibex>

Highly configurable **open source** RV32IMCB core with security features that include:

- Instruction cache memory scrambling
- Dual-core lockstep
- Data independent timing
- Dummy instruction insertion
- Bus and register file integrity
- Hardened PC

Evolving Ibex[®]: Memory Safety with CHERIoT

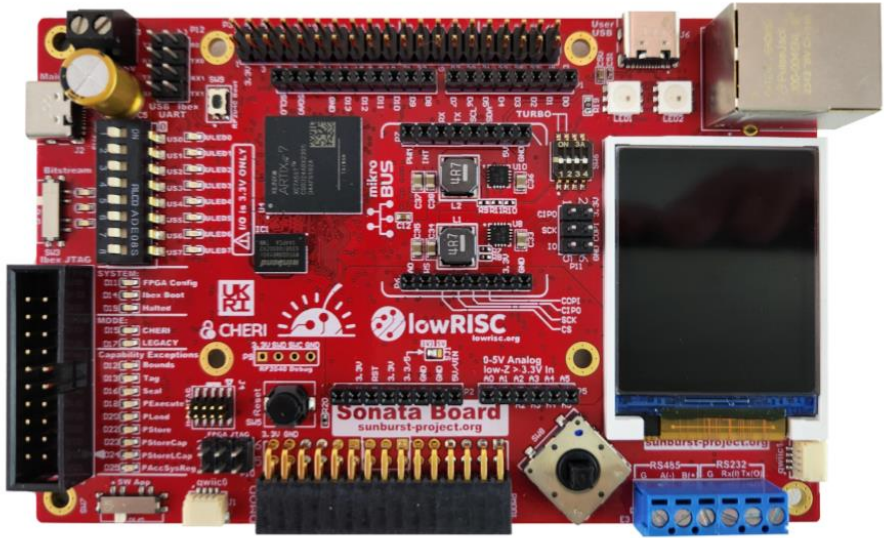
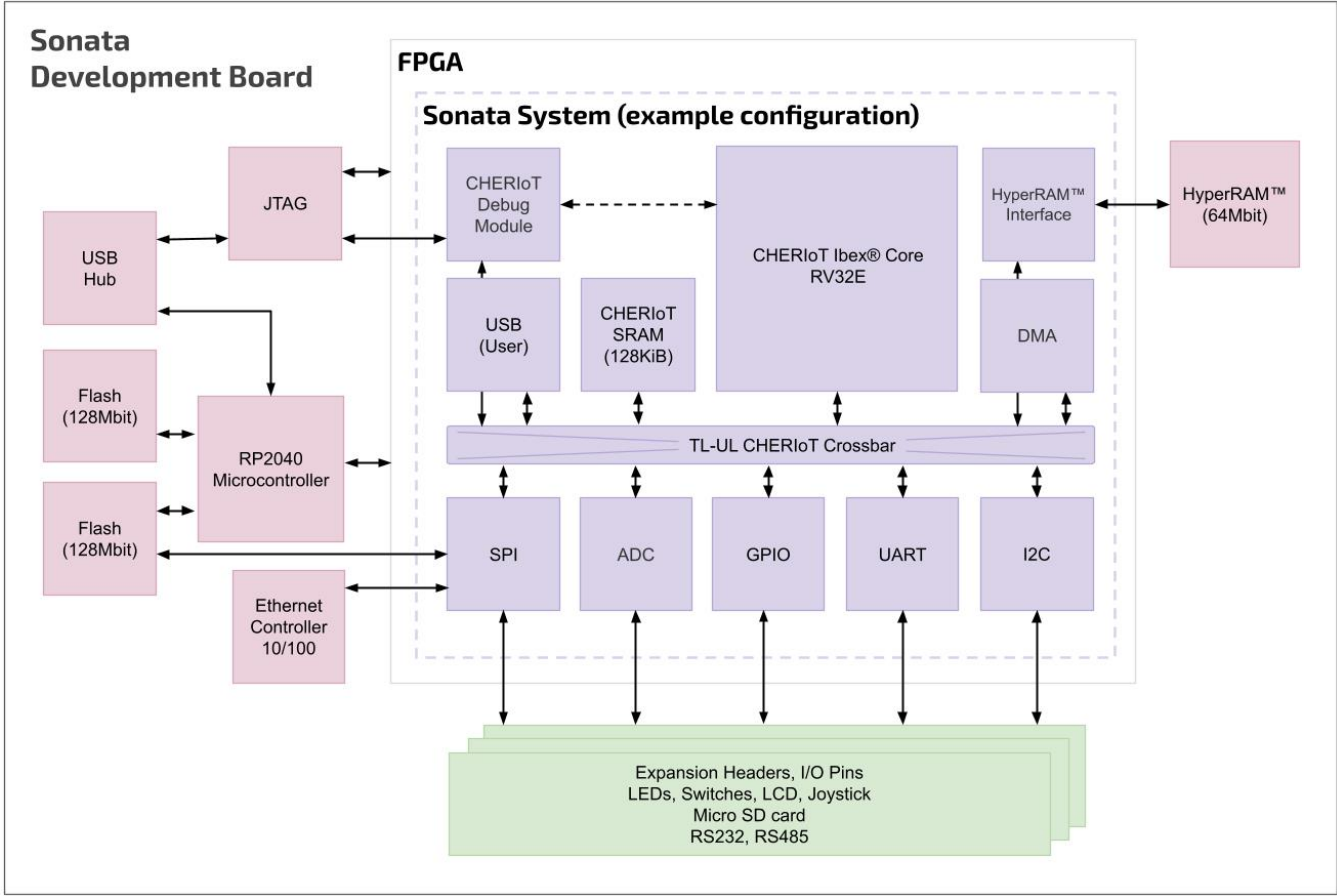


“This is truly important foundational work, as it will help make CHERIoT-Ibex the world’s first production grade, open-source CHERI-enabled microcontroller core. We’re looking forward to seeing it broadly leveraged in commercial designs, bringing much-needed hardware security — in an efficient manner — to a broad swathe of critical applications.”

Tony Chen
Partner Security Architect, Microsoft

<https://github.com/microsoft/CherIoT-ibex>

Securing Operational Technology: Sonata



Delivered by
Innovate UK,
EPSRC and ESRC

Digital Security
by Design

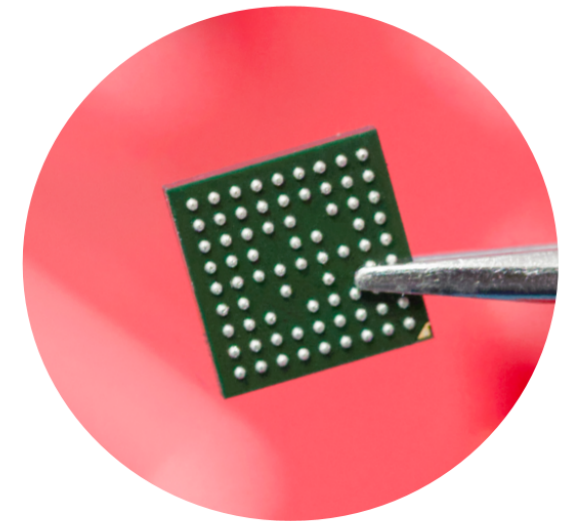
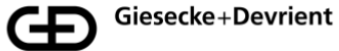
sunburst-project.org

DSbD/UKRI Project Grant Number: 107540

Build Together, Build Well & Build Securely



Silicon
Commons



*lowRISC's full-stack engineering team can help develop and integrate open source silicon IP in **your** products!*

Contact us at info@lowrisc.org to find out more