OpenTitan Integrated

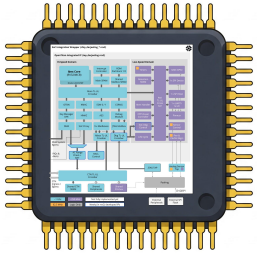
A RISC-V Open-Source Silicon Root-of-Trust for large SoCs

Robert Schilling
rschilling@rivosinc.com

Rivos

What is OpenTitan Integrated?

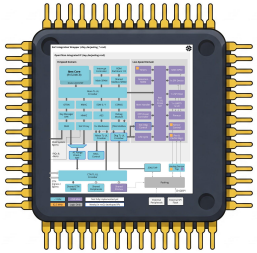
From a discrete chip…



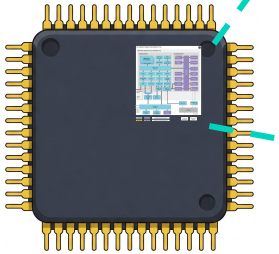
- **Open Source Silicon Root of Trust (RoT)**
- **Fully Open Design:** RTL, DV, firmware, and documentation under a **permissive** license: **https://opentitan.org**
- **Trustworthy & Verifiable Security:** Enhancing hardware security through an open and auditable foundation
- **Focus on Quality & Flexibility:** Emphasizes rigorous verification and adaptable design for diverse integrations

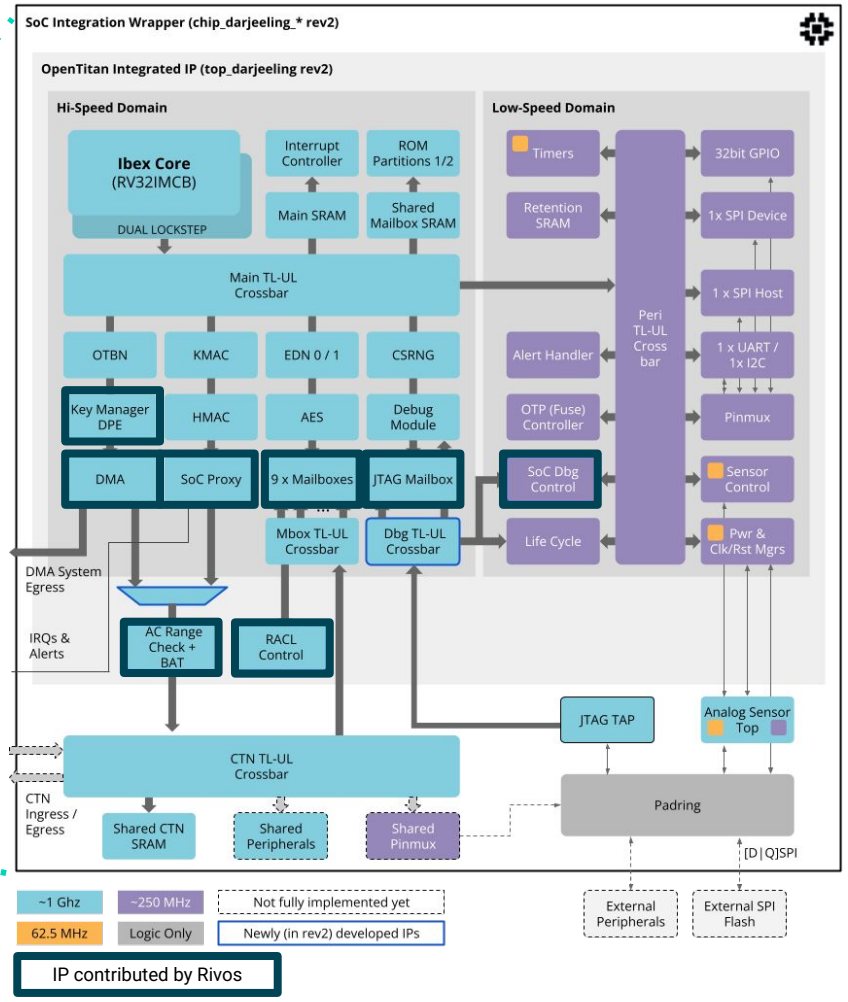
What is OpenTitan Integrated?

From a discrete chip…



… to a an integrated RoT





Controlled Communication Interface

RoT and SoC communication need proper isolation

Principle of **Least Privilege**

- SoC should not have unfettered access into RoT
- RoT should not have unfettered access into SoC

SoC may have **different** memory space partitions

- OpenTitan controlled
- SoC controlled
- …

SoC to OpenTitan - Mailbox

SoC has **no direct access** into OpenTitan space

All transactions managed **through a mailbox**

- External host deposits transactions, OpenTitan software reads
- OpenTitan software deposits transactions, external host reads

Many applications

  - Debug authorization request
  - Security services request

OpenTitan to SoC - DMA

DMA has **limited access** to OpenTitan private memory

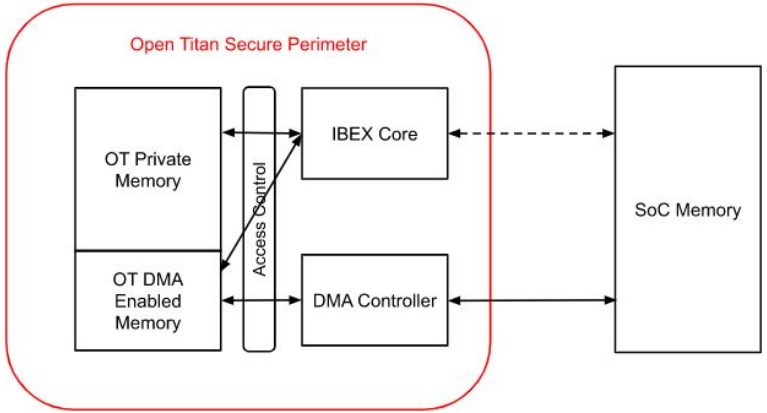
- Only operates on a **isolated** memory range

Support for inline hashing operation

- Compute SHA-2 digest while transferring data

Many applications

- Code loading and verification
- Data transfer to low-speed IOs

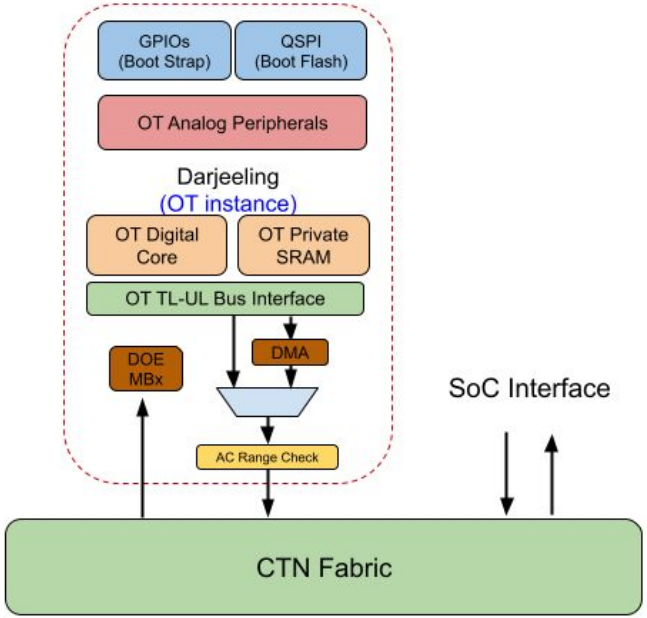


Access Control Range Check

New IP

- Configurable number of ranges
- **TOR** matching logic with static prioritization
- Permission checks for **R/W/X** and **RACL**

- Used at the boundary of RoT
- Also comes with **block-level DV**



Debug and DFT Governance

OpenTitan maintains life cycle scheme

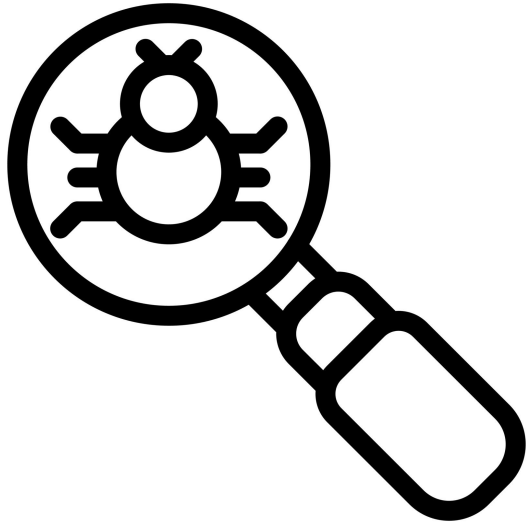
- Local debug and test gated directly by the life cycle

OpenTitan authorizes SoC debug and test

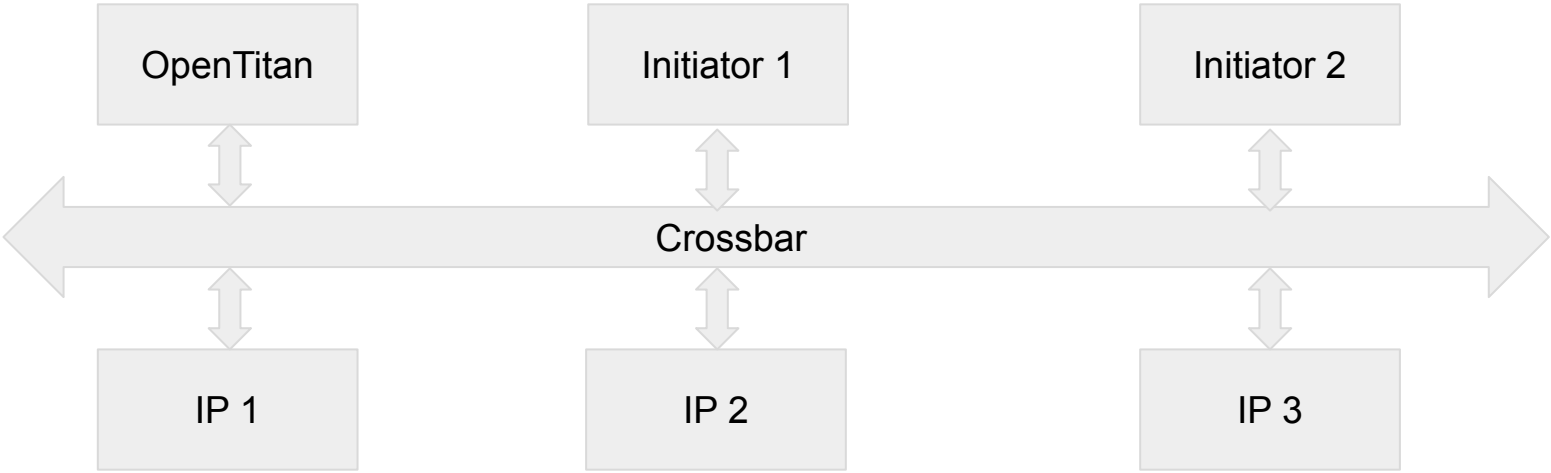
- SoC can either rely solely on OpenTitan authorization or combine with SoC scheme

Overall scheme

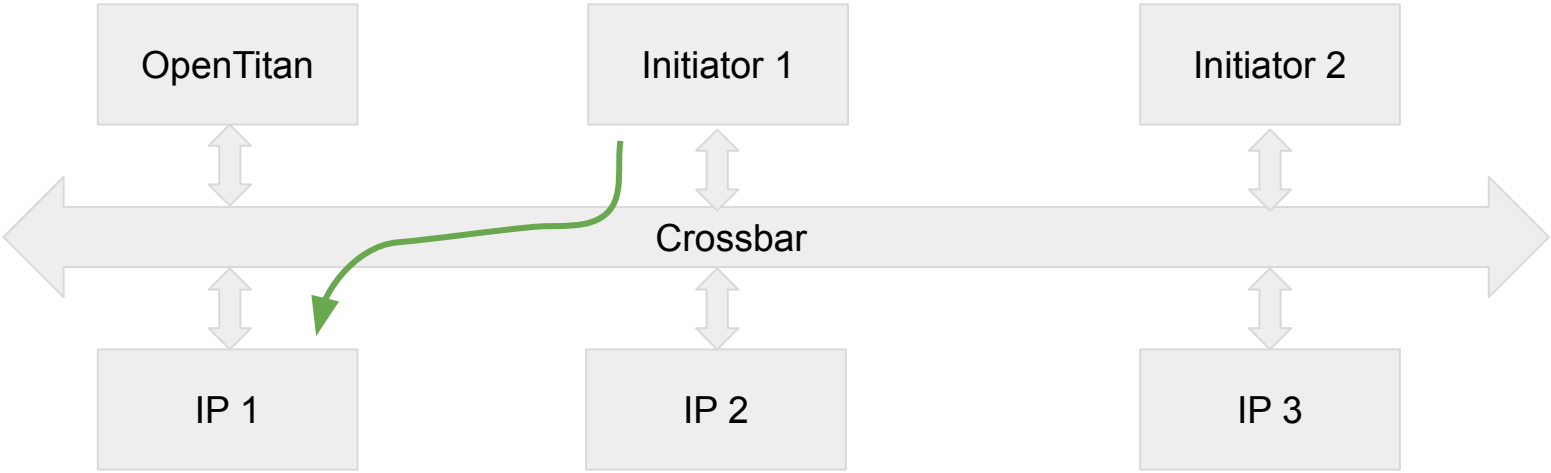
- SoC requests debug through mailbox
- OpenTitan software initiates a challenge / response protocol
- If SoC provides valid response, debug and test is **unlocked** via a distributed **debug policy bus**



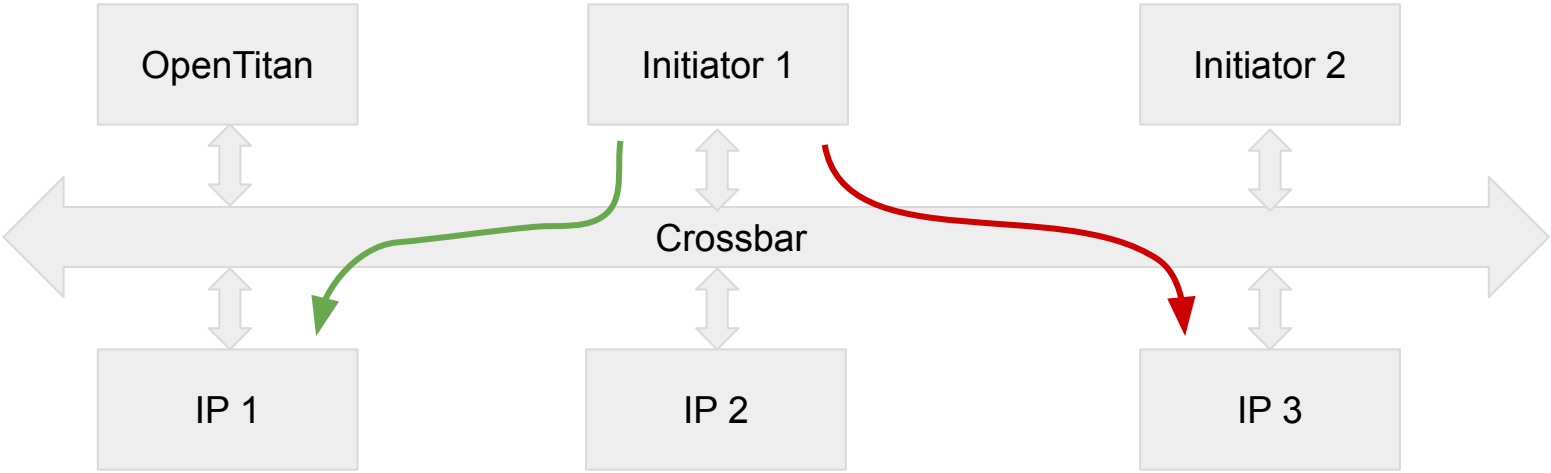
Restrict Register Access in a Shared Environment



Restrict Register Access in a Shared Environment



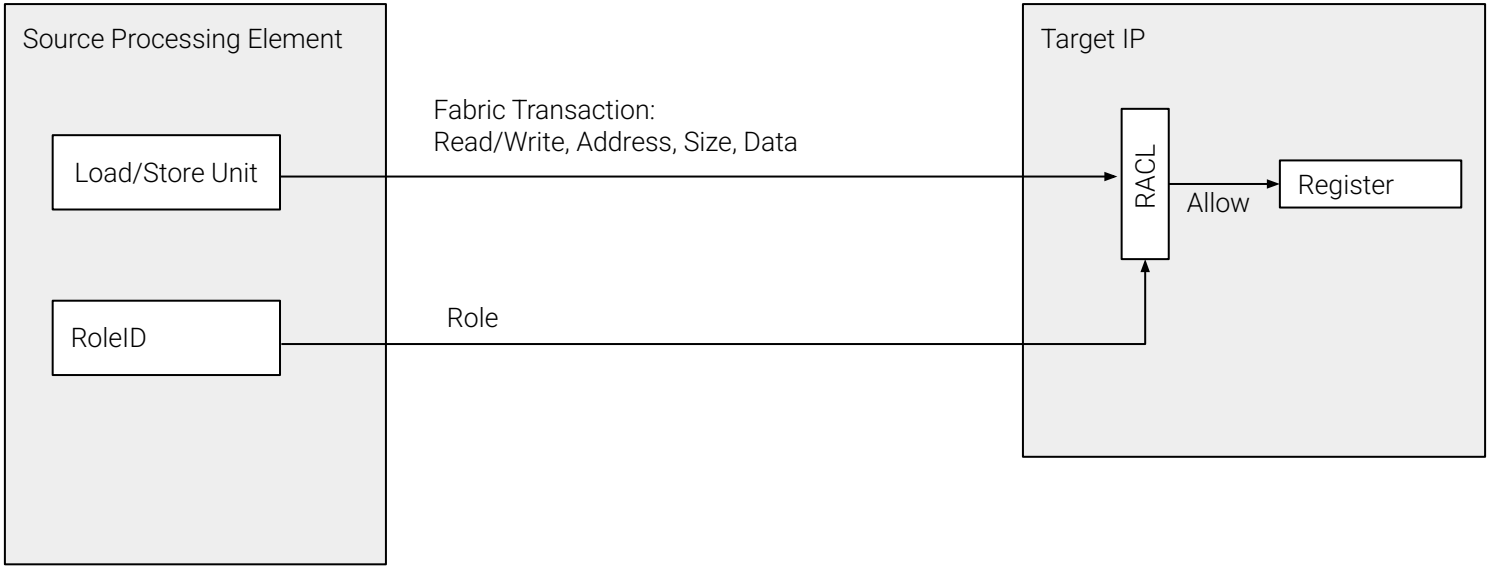
Restrict Register Access in a Shared Environment



RACL - Register Access Control List

Provides differentiated security on access to registers

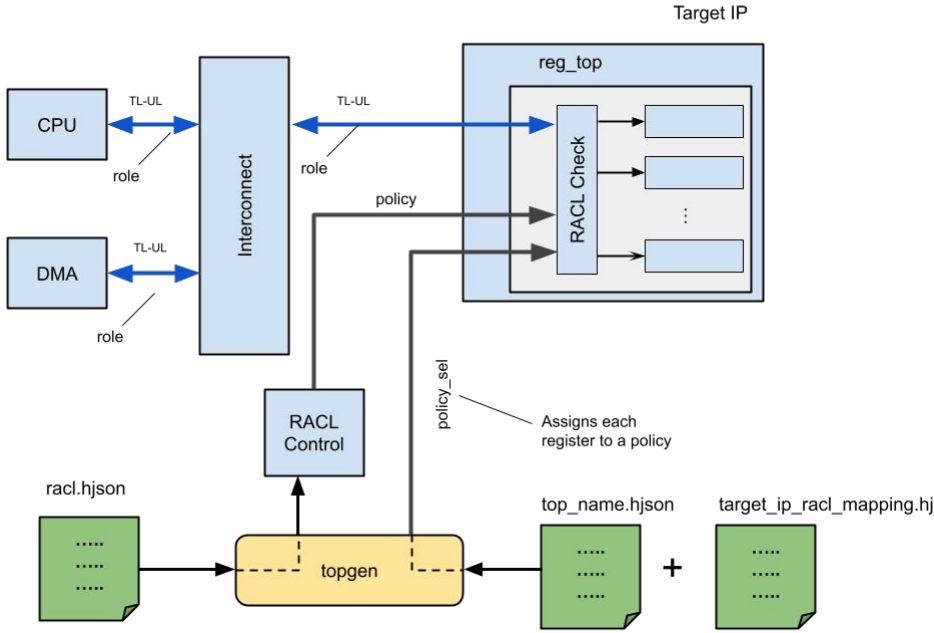
- Each processing element is assigned a role
- Each register is assigned with a policy defining what role can read or write



RACL Integration in OpenTitan

- **Machine readable specification**

  for RTL, DV, and documentation
  - Defines roles and policies
  - Provides register mapping
- Automated **code generation**
- Central new IP: **racl_ctrl**
  - Defines the policies for the subsystem
  - Collects error information
  - Error log arbitration
- Native RACL support in IPs



SoC Generator for other Designs

OpenTitan designs are defined in HJSON configuration files

- From that **single source of truth**,
  a SoC generator creates:
  - **The top-level RTL level**
  - **DV, software, documentation**

Support for **arbitrary designs**

- Minimal design with just a CPU, Memory, XBARs, UART
  - Use as a **companion** core in your SoC
- Discrete Root of Trust SoC
- etc

A common software stack and build system targets **all** designs

```
{ name: "dma",
  type: "dma",
  clock_srcs: {clk_i: "main"}
  clock_group: "infra",
  reset_connections: {rst_ni: "lc"},
  base_addr: {hart: "0x22010000"},
},
{ name: "mbx0",
  type: "mbx",
  clock_srcs: {clk_i: "main"}
  clock_group: "infra",
  reset_connections: {rst_ni: "lc"},
  base_addrs: {
    core: {hart: "0x22000000"},
    soc:  {soc_mbx: "0x01465000"},
  },
  racl_mappings: {
    soc: 'racl/all_rd_wr_mapping.hjson'
  }
},
```

OpenTitan is ready to be placed in a large SoC

- It's got the **right communication interfaces**
  - Secure mailbox and DMA interfaces
- Debug governance via a flexible debug policy bus
- RACL provides a **fine-granular** and customizable **access protection** for registers
- Generic SoC generator supports custom designs
  - Support for external alerts, interrupts
- … and much more: **Post-Quantum Computing, DICE,** …

- **Available at: https://opentitan.org**