Caliptra Onboarding for ODM Partners
Silicon Lifecycle & Debug

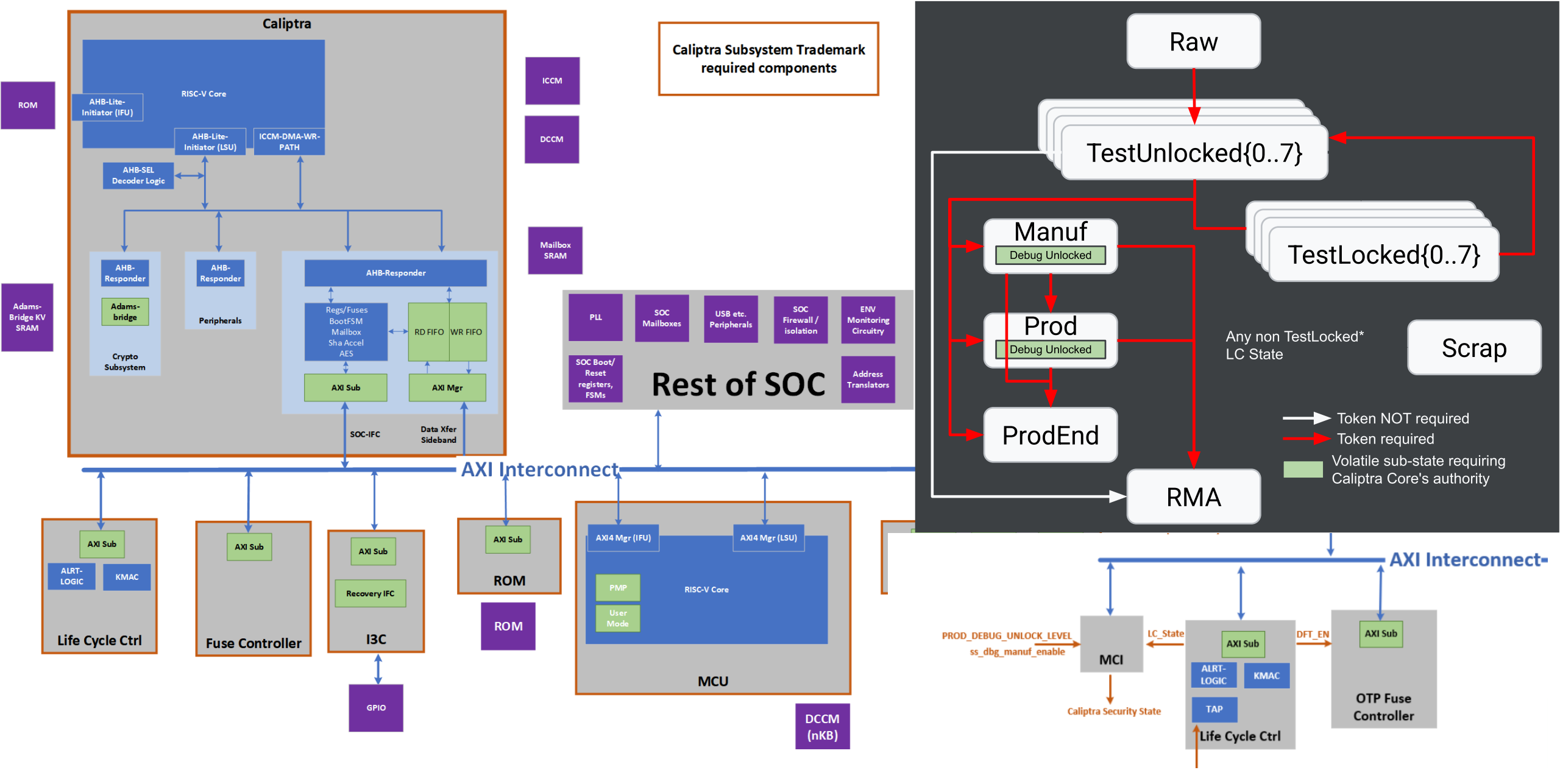
Device provisioning stages, fuses, and debug unlock

ATE / Factory

ODM Board Build

System Integration

Datacenter / RMA


Caliptra
RISC-V Core
AHB-Lite-Initiator (IFU)
AHB-Lite-Initiator (LSU)
ICCM-DMA-WR-PATH
AHB-SEL Decoder Logic
AHB-Responder
Adams-bridge
Crypto Subsystem
AHB-Responder
Peripherals
AHB-Responder
Regs/Fuses BootFSM Mailbox Sha Accel AES
RD FIFO
WR FIFO
AXI Sub
AXI Mgr
SOC-IFC
Data Xfer Sideband
ROM
ICCM
DCCM
Mailbox SRAM
Adams-Bridge KV SRAM
Caliptra Subsystem Trademark required components
PLL
SOC Mailboxes
USB etc. Peripherals
SOC Firewall / isolation
ENV Monitoring Circuitry
SOC Boot/ Reset registers, FSMs
Rest of SOC
Address Translators
AXI Interconnect
AXI Sub
ALRT-LOGIC
KMAC
Life Cycle Ctrl
AXI Sub
Fuse Controller
AXI Sub
Recovery IFC
I3C
GPIO
AXI Sub
ROM
ROM
AXI4 Mgr (IFU)
AXI4 Mgr (LSU)
PMP
User Mode
RISC-V Core
MCU
DCCM (nKB)
Raw
TestUnlocked{0..7}
Manuf
Debug Unlocked
Prod
Debug Unlocked
ProdEnd
RMA
TestLocked{0..7}
Scrap
Any non TestLocked* LC State
Token NOT required
Token required
Volatile sub-state requiring Caliptra Core's authority
AXI Interconnect
PROD_DEBUG_UNLOCK_LEVEL
ss_dbg_manuf_enable
MCI
Caliptra Security State
LC_State
AXI Sub
ALRT-LOGIC
KMAC
TAP
Life Cycle Ctrl
DFT_EN
AXI Sub
OTP Fuse Controller

Life-Cycle States Across the Provisioning Stages

Which LCC states each stage lives in, and where debug unlock, RMA and SCRAP become possible

01

ATE / Factory

Silicon vendor

- RAW
- TEST_UNLOCKED
- MANUF
- MANUF debug

02

ODM Board Build

ODM board + system build

- MANUF
- PROD
- MANUF debug
- PROD debug (level 0+)

03

System Integration

ODM system integration

- PROD
- PROD debug (level 0+)

04

Datacenter / RMA

CSP operations

- PROD
- PROD debug (level 0+)
- RMA
- SCRAP

From ODM board build onward: both MANUF and PROD parts can enter debug mode
End of life: RMA opens hardware debug; SCRAP disables the device.

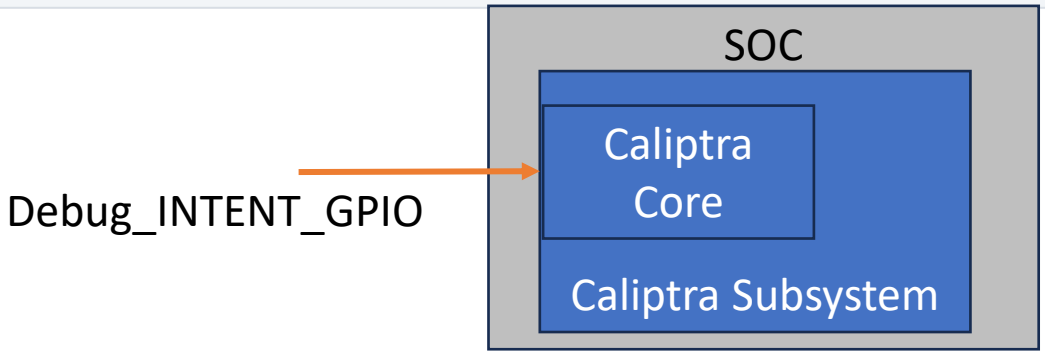
SoC Debug Capabilities and Expressing Debug Intent

What the Caliptra Subsystem offers the SoC at t=0, and the construction-specific paths for delivering the intent

What the subsystem provides

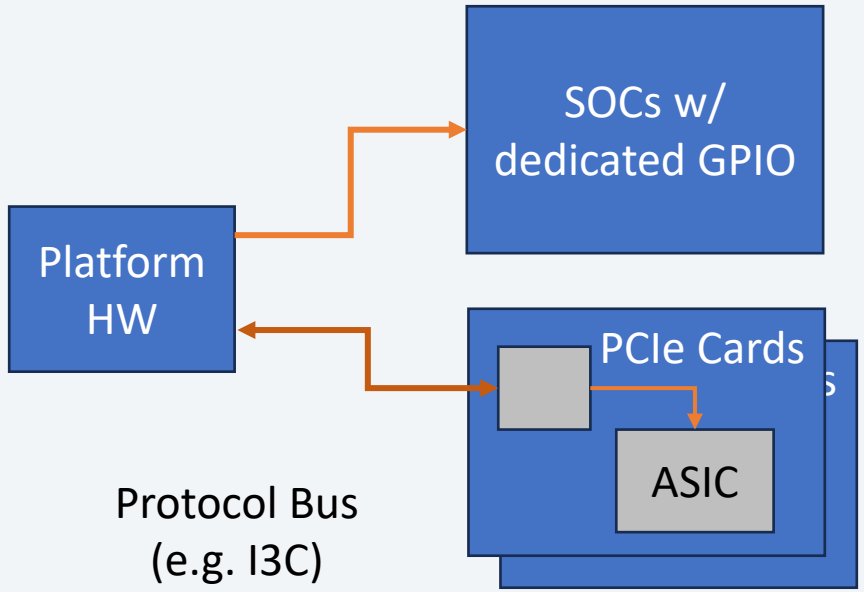
1. **Time-zero HW and ROM debug** — the debug intent is sampled only at t=0; once authorized, the Caliptra and MCU TAP interfaces open.
2. **Secrets first** — UDS and Field Entropy are wiped or never latched, so no identity keys can be derived in debug mode.
3. **Delivery is SoC construction specific** — how DEBUG_INTENT reaches the subsystem is the integrator's choice; authorization is unchanged.

Where the intent lands: the Caliptra Subsystem inside the SoC



Delivering the intent: Platform integration view

A dedicated GPIO, or a protocol bus such as I3C — the Caliptra-side contract is identical



SOC Hardware Debug over JTAG (t=0) Unlock Flow

What the debug manager, the SoC and the Caliptra Subsystem each do, from t=0 through runtime

1
Assert `SoC DEBUG_INTENT GPIO`
Debug Manager expresses the intent to debug before power good — it is only sampled at t=0.

2
Set `Caliptra BootFSMBrk`
Boot FSM is halted if we want to debug Subsystem with ROM.

3
SoC powers up
Power good reaches the subsystem;.

4
Caliptra samples the debug intent
At t=0 only. UDS and Field Entropy are wiped or never latched, so no keys can be derived.

5
ROM or RT FW evaluates the credential
Note: Don’t forget to assert BootGO if you set BootFSMBrk

6
Caliptra debug ports open

7
SoC debug ports open
MCI mask registers OR in DFT_EN, SOC_DFT_EN and SOC_HW_DEBUG_EN for the granted level.

8
ROM or RT FW runs with `debug ports open`

SOC Hardware Debug over JTAG Unlock Flow at RT

Platform wants to unlock SOC at a future time

Assert `SoC DEBUG_INTENT GPIO`
Debug Manager expresses the intent to debug before power good — it is only sampled at t=0.

2
Set `Caliptra BootFSMBrk`
Boot FSM is halted if we want to debug Subsystem with ROM.

SoC powers up
Power good reaches the subsystem;.

Caliptra samples the debug intent
At t=0 only. UDS and Field Entropy are wiped or never latched, so no keys can be derived.

SOC FW Loads (Caliptra, MCU etc. FW)

Debug Unlock Command Flow through Caliptra Core Mailbox (whenever unlock is required)

SoC (JTAG) debug ports open

ROM or RT FW runs with `debug ports open`

SOC Firmware Debug

No need to unlock JTAG
SOC Firmware = “Any” FW except Caliptra core
Pre-step: Set Debug bit in SOC manifest

1
Assert `SoC DEBUG_INTENT GPIO`
Debug Manager expresses the intent to debug before power good — it is only sampled at t=0.

2
SoC powers up
Power good reaches the subsystem;.

3
Caliptra samples the debug intent
At t=0 only. UDS and Field Entropy are wiped or never latched, so no keys can be derived.

4
Caliptra samples the debug intent
At t=0 only. UDS and Field Entropy are wiped or never latched, so no keys can be derived.

Load/Stream `Caliptra Core FW`

5
`Caliptra Core FW Runs`

6
SOC RT FW Manifest with debug bit = 1 loaded/streamed

7
Caliptra core checks if debug intent is set

8
SOC `debug FW images can run (if authentications pass)`

Caliptra Firmware Debug

No need to unlock JTAG
Pre-step: Set Debug bit in SOC manifest

1
Assert `SoC DEBUG_INTENT GPIO`
Debug Manager expresses the intent to debug before power good — it is only sampled at t=0.

2
SoC powers up
Power good reaches the subsystem;.

3
Caliptra samples the debug intent
At t=0 only. UDS and Field Entropy are wiped or never latched, so no keys can be derived.

4
Caliptra samples the debug intent
At t=0 only. UDS and Field Entropy are wiped or never latched, so no keys can be derived.

5
`Caliptra Core FW Loaded/Streamed`

6
Caliptra core ROM checks if debug intent is set

7
Caliptra Debug Image can run if passes authentication

Fuses, Lifecycle, and Debug Unlock Fuses

Fuse map v2.1 — what gets burned, and how a locked production part is opened for debug

Key fuse partitions

`SECRET_MANUF_PARTITION`

CPTRA_CORE_UDS_SEED — root of the identity chain

`SECRET_PROD_PARTITION_0..3`

CPTRA_CORE_FIELD_ENTROPY_0..3 — four one-shot slots

`SW_MANUF_PARTITION`

IDEVID cert attributes, HSM identifier, SoC stepping ID

`VENDOR_HASHES_MANUF/PROD`

Vendor PK hashes, owner PK hash and valid flag

`SW_TEST_UNLOCK_PARTITION`

CPTRA_SS_MANUF_DEBUG_UNLOCK_TOKEN

`SECRET_LC_TRANSITION_PARTITION`

Test unlock, TEST-to-MANUF, MANUF-to-PROD, RMA tokens

Debug unlock

Manufacturing debug unlock

Used while the part is still in MANUF lifecycle state. Needs to be provisioned in TEST_UNLOCKED state (unprovisioned security state).
The requester presents the token and Caliptra first applies sha512 and then compares it against fuses before releasing enabling debug.

Production debug unlock

For parts already in PROD. No shared secret is burned in.
Authorized by signature: eight CPTRA_SS_PROD_DEBUG_UNLOCK_PKS entries hold the public key hashes permitted to authenticate an unlock request, so debug authority can be delegated per level and revoked.
Signing authority is challenged with a nonce

Caliptra Onboarding for ODM Partners

Device provisioning & Certificate Extraction

What Caliptra Gives the Device

A fuse-rooted identity chain, built once in silicon and endorsed through manufacturing

IDEVID

UDS seed in fuses. CSR extracted at ATE

LDEVID

UDS CDI is combined with Field-entropy and then generates this certificate

ALIASFMC

FMC bound attestation

Roots that must exist before any of this works

UDS seed and vendor public key hashes are burned at ATE, so the part reaches your line already in PROD lifecycle state, secure-boot enforced, running production-signed firmware only.
Field entropy is programmed last — after it, LDEVID and everything below it change. Order matters.

Provisioning Stages and Who Does What

Each stage hands off to the next one a device plus a data set — certificates, CSRs, and serial numbers

01 ATE / Factory

Silicon vendor

- Vendor SoC specific fuses (e.g. PK hash, LCC tokens, UEID etc.)
- Manage lifecycle Transitions
- Provision root seeds (e.g. UDS-seed) and CSR extraction

02 ODM Board Build

ODM board + system build

- Device has a fully functional iRoT
- CSP specific flows happen here (e.g. FE programming, LDEVID flow, Owner PK hash)
- I3C available to BMC OOB communication
- Prod Debug Unlock available for HW debug
- Debug FWs can be loaded securely

03 System Integration

ODM system integration

- Some ODM board build functions may run here (CSP specific flows)
- All ODM board build capabilities continue to be available (e.g. Endorse ALIASFMC certificate)
- DOT Flow for system (typically happens here)

04 Datacenter / RMA

Microsoft operations

- Extract manufacturing, infra-owner, tenant cert chains
- Attestation using signed firmware measurements

Provisioning Demo