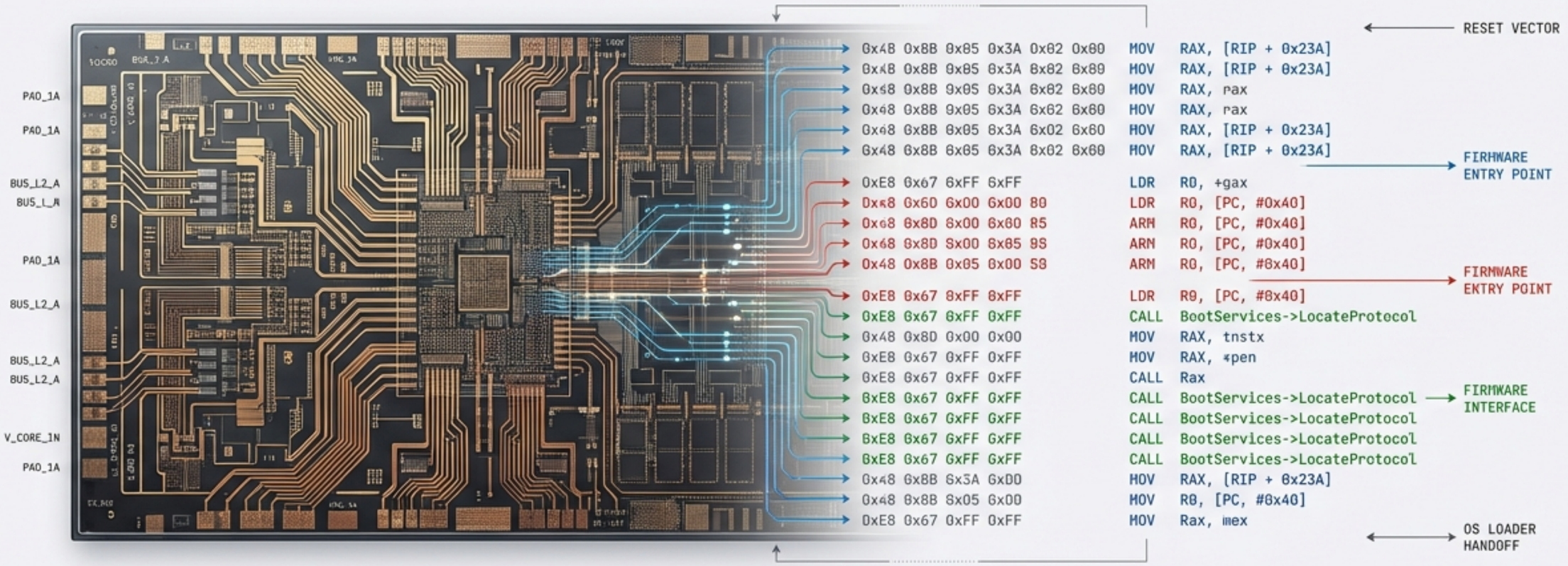
Modern Boot Architectures: Silicon to Software

A Technical Deep Dive into Intel Arrow Lake (SBL), ARM TF-A, and UEFI/PI



PLATFORM FOCUS	SCOPE	DOCUMENT VERSION
Intel Arrow Lake Core Ultra 200 Series	Hardware Root of Trust Execution Flows Data Handoffs	1.0 // Verified Jan 2026

The Architecture Landscape

1. Intel Slim Bootloader (SBL)

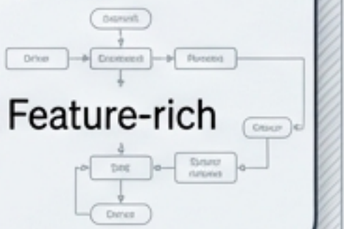
- Optimized for speed and static configuration.
- **Target:** IoT and Client.
- **Philosophy:** Deterministic & Lightweight.

2. ARM Trusted Firmware-A (TF-A)

- Built for security isolation (TrustZone).
- **Target:** Mobile/Embedded.
- **Philosophy:** Secure World vs. Normal World.

3. UEFI/PI (EDK II)

- Built for broad compatibility and extensibility.
- **Target:** Enterprise/General Purpose.
- **Philosophy:** Feature-rich & Dynamic.

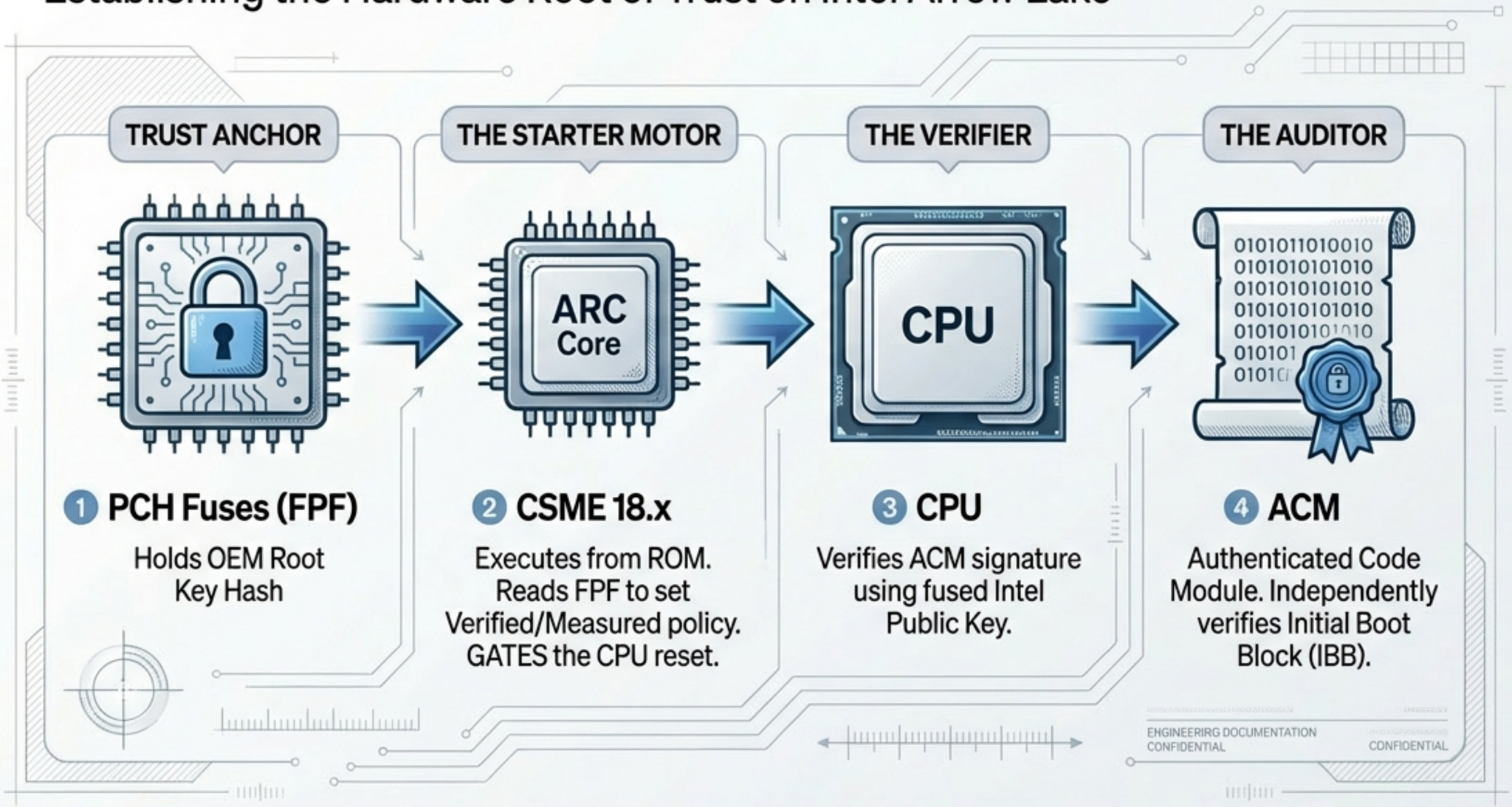


ARROW LAKE SPECIFICATIONS

- **CSME:** Version 18.x (ARC-based microcontroller, NOT i486)
- **Boot Guard:** Version 3 (CBnT - Converged Boot Guard and TXT)
- **Silicon:** Disaggregated Tile Architecture (Compute, SoC, I/O tiles)

The Foundation: Boot Guard v3 & CBnT Trust Model

Establishing the Hardware Root of Trust on Intel Arrow Lake



ARROW LAKE SPECIFICATIONS

- **CSME:** Version 18.x (ARC-based microcontroller, NOT i486)
- **Boot Guard:** Version 3 (CBnT - Converged Boot Guard and TXT)
- **Silicon:** Disaggregated Tile Architecture (Compute, SoC, I/O tiles)

Critical Correction: The Verification Chain

Addressing common misconceptions regarding CSME and ACM interaction

MYTH ✕	FACT ✓
Myth: CSME is the authoritative validator that verifies the ACM signature.	**Fact:** The CPU verifies the ACM signature using an Intel public key fused directly into the CPU silicon.
Myth: CSME and ACM jointly verify the Initial Boot Block.	**Fact:** The ACM independently reads FPF fuses to retrieve the OEM Key Hash and performs the authoritative 4-step validation (Key Manifest -> Boot Policy Manifest -> IBB).
Myth: CSME is an i486 core.	**Fact:** CSME 18.x is an ARC-based microcontroller.

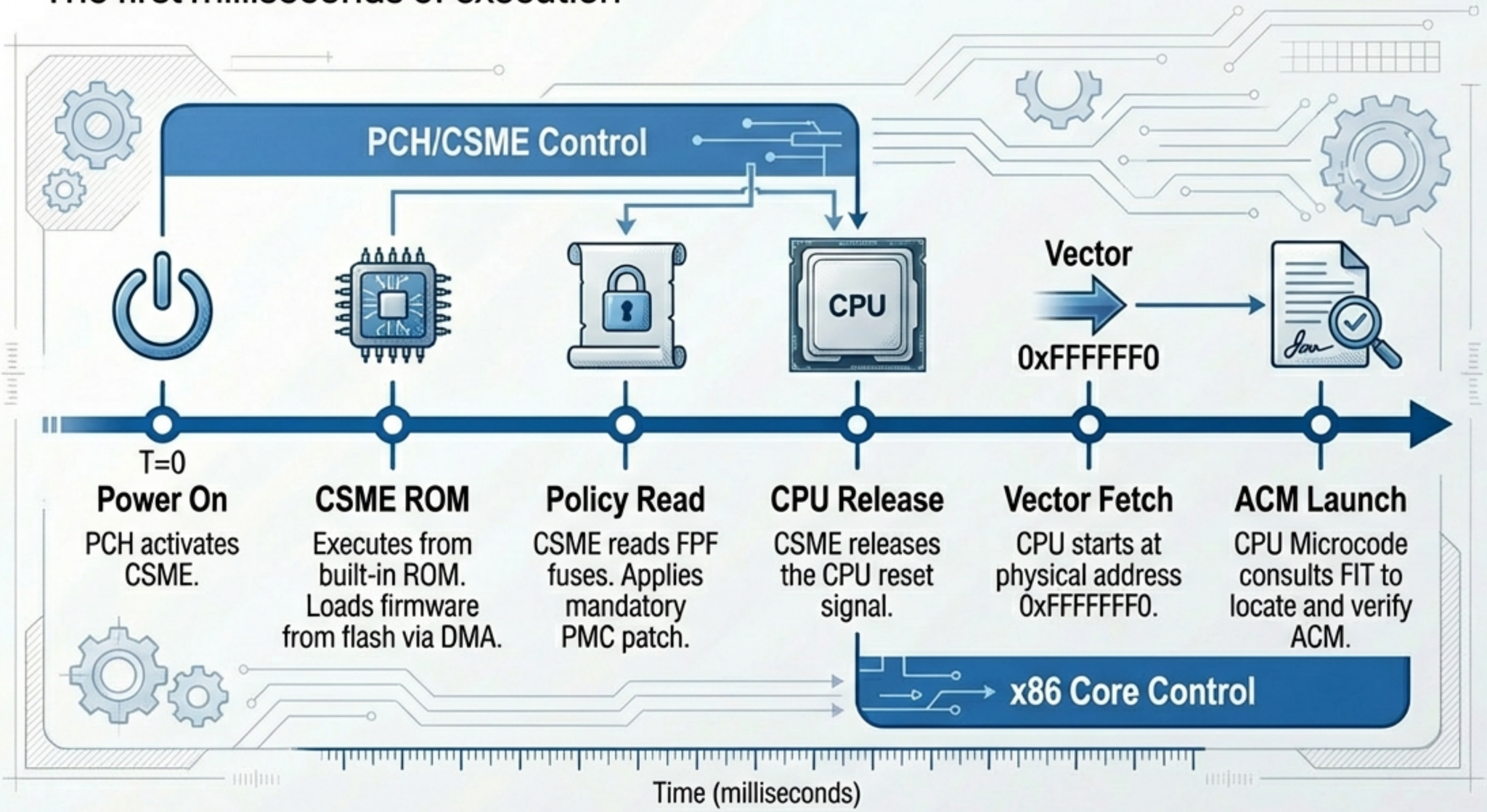
Source: Intel CSME Security White Paper v1.5 & Arrow Lake Technical Guide

ARROW LAKE SPECIFICATIONS

- **CSME:** Version 18.x (ARC-based microcontroller, NOT i486)
- **Boot Guard:** Version 3 (CBnT - Converged Boot Guard and TXT)
- **Silicon:** Disaggregated Tile Architecture (Compute, SoC, I/O tiles)

Sequence Phase I: Power-On to Reset Vector

The first milliseconds of execution



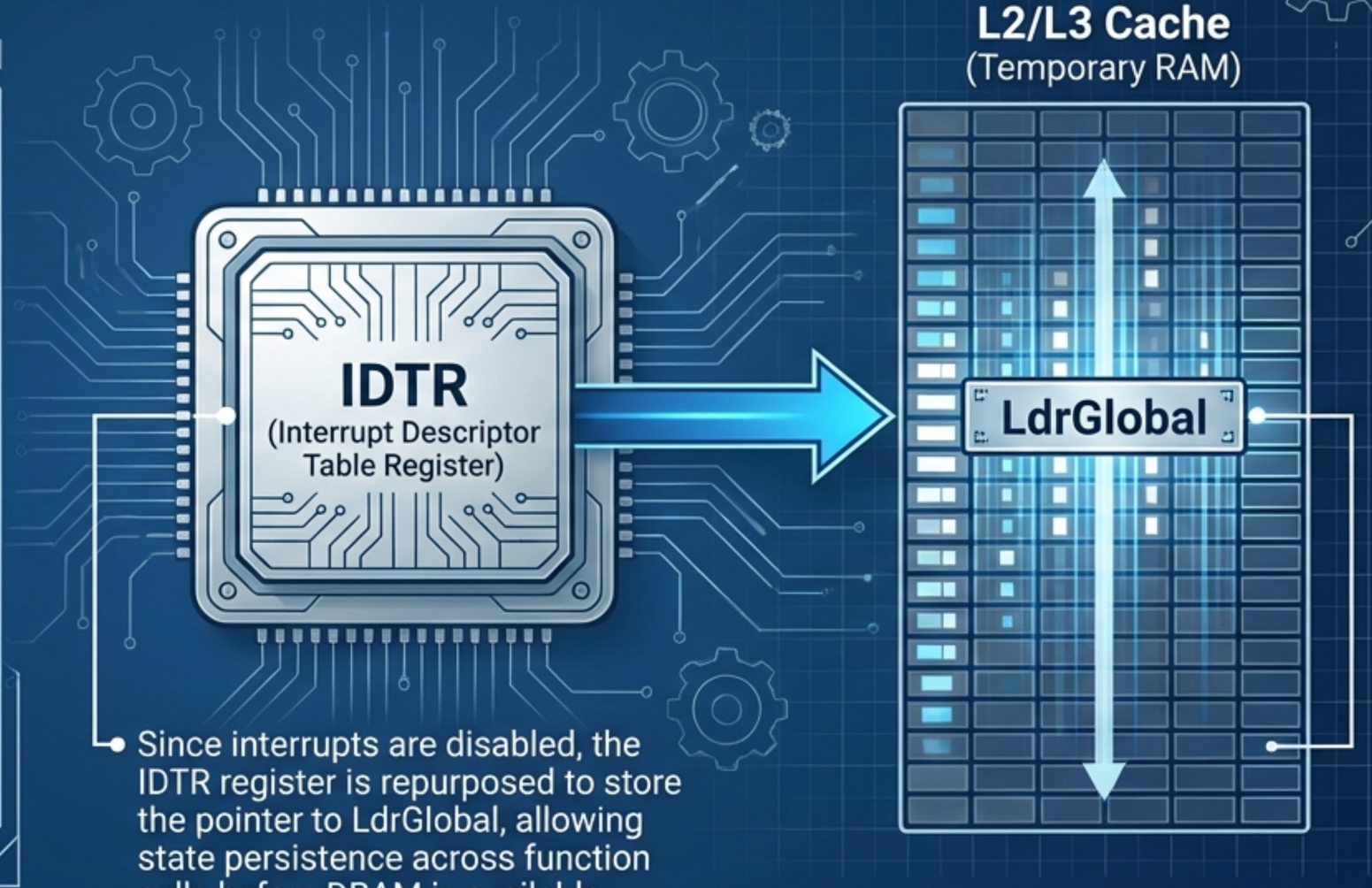
ARROW LAKE SPECIFICATIONS

- **CSME:** Version 18.x (ARC-based microcontroller, NOT i486)
- **Boot Guard:** Version 3 (CBnT - Converged Boot Guard and TXT)
- **Silicon:** Disaggregated Tile Architecture (Compute, SoC, I/O tiles)

Stage 1A: The Real-to-Protected Transition

Executing code without RAM: The IDTR Trick

1. Reset Vector (0xFFFFFFF0)
2. Switch from 16-bit Real Mode to 32-bit Protected Mode
3. Load Global Descriptor Table (GDT)
4. Initialize Cache-as-RAM (CAR) via FSP-T



Since interrupts are disabled, the IDTR register is repurposed to store the pointer to LdrGlobal, allowing state persistence across function calls before DRAM is available.

ENGINEERING DOCUMENTATION
CONFIDENTIAL

Stage 1B: Memory Initialization & Training

The migration from Cache to DRAM

1

Parse Config

Parse CfgData (YAML). Reads DIMM slots, frequencies, and GPIOs.

2

Execute FSP-M

Calls FspMemoryInit. Performs DDR Training (Write/Read Leveling).

3

HOB Creation

FSP-M builds Hand-Off Blocks (HOBs) describing the discovered memory map.

4

The Migration

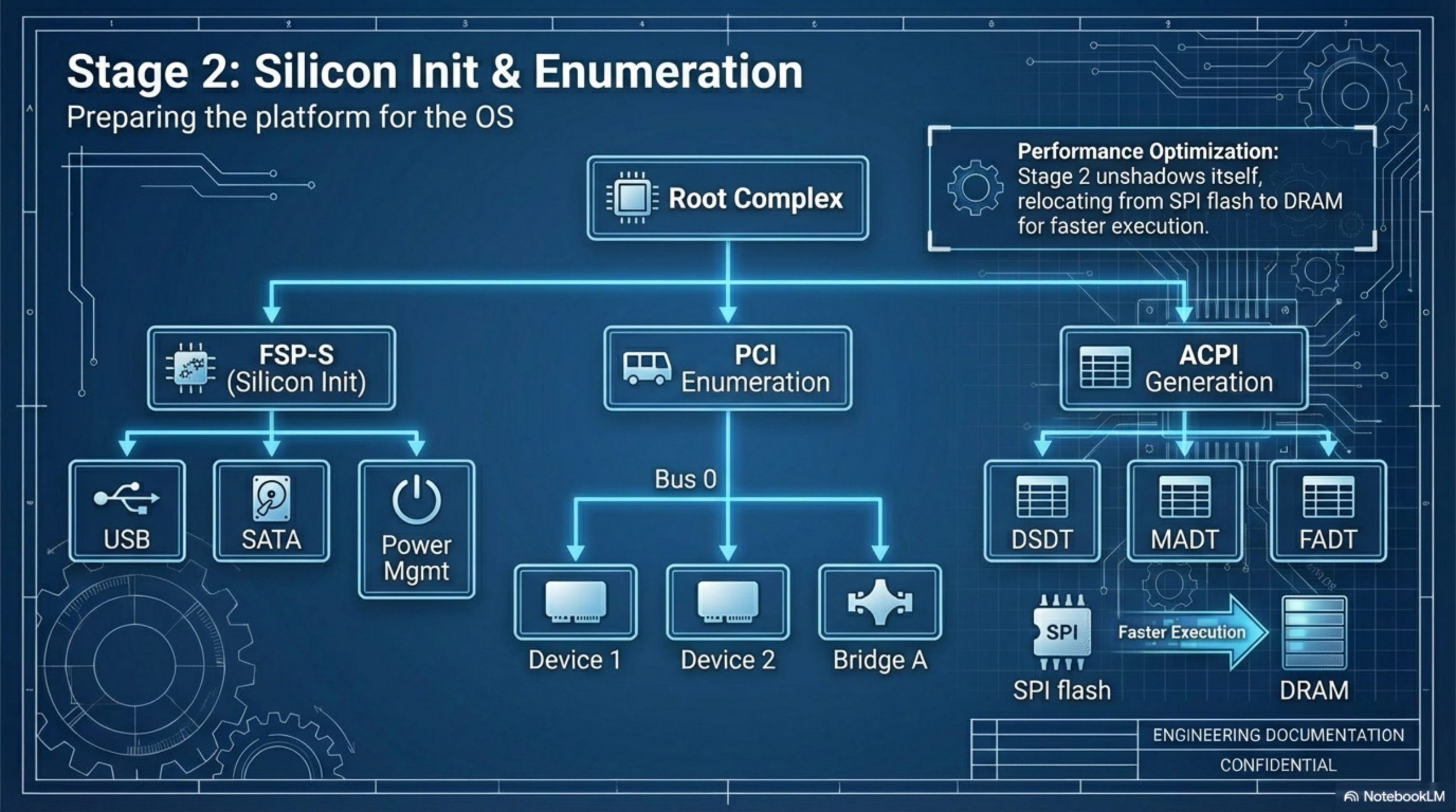
LdrGlobal and Stack are copied from Cache-as-RAM to permanent DRAM. TempRamExit is called to close CAR.

	ENGINEERING DOCUMENTATION
	CONFIDENTIAL

Stage 2: Silicon Init & Enumeration

Preparing the platform for the OS

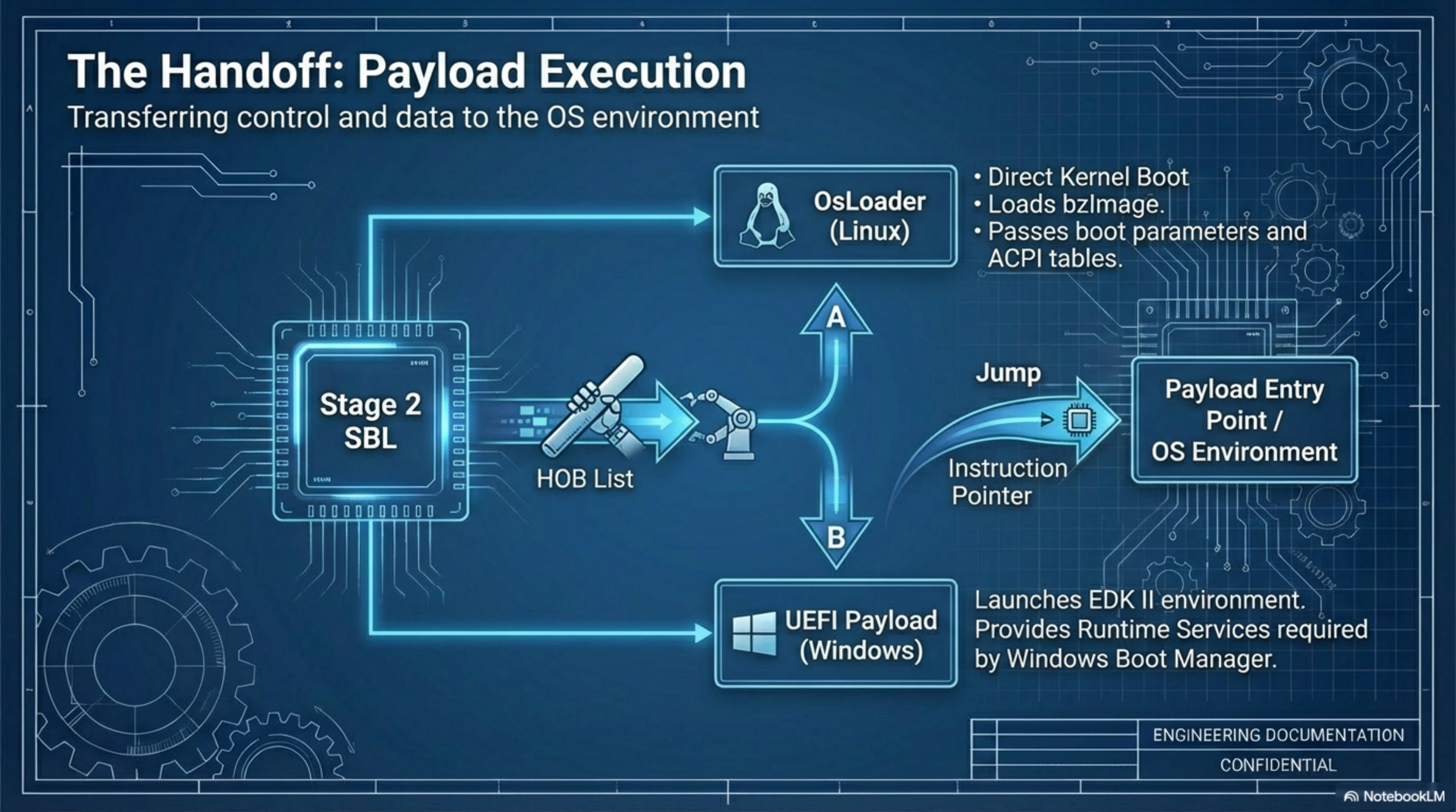
Performance Optimization:
Stage 2 unshadows itself, relocating from SPI flash to DRAM for faster execution.



ENGINEERING DOCUMENTATION
CONFIDENTIAL

The Handoff: Payload Execution

Transferring control and data to the OS environment

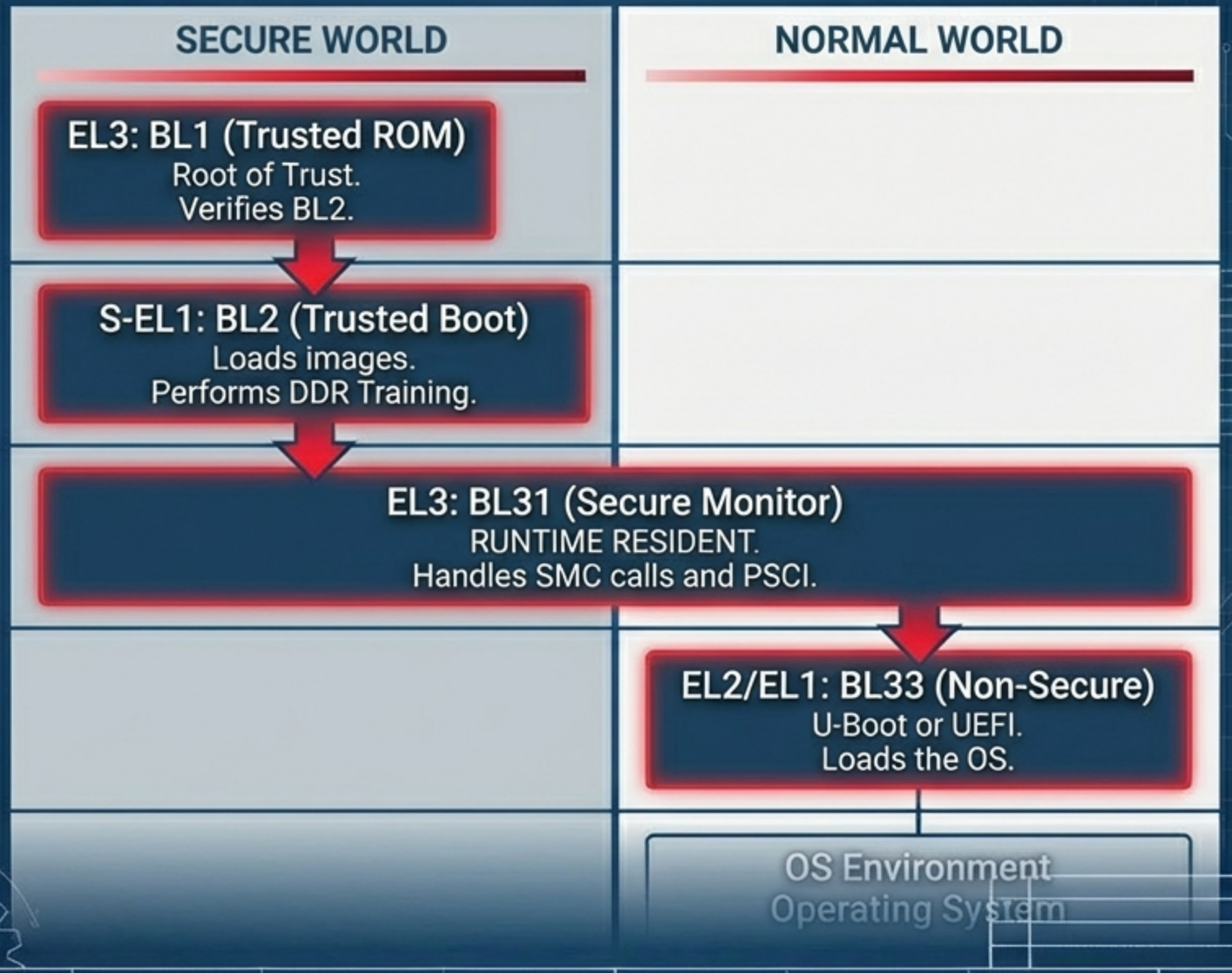


ENGINEERING DOCUMENTATION

CONFIDENTIAL

The ARM Approach: Trusted Firmware-A (TF-A)

Security through isolation: The Secure World

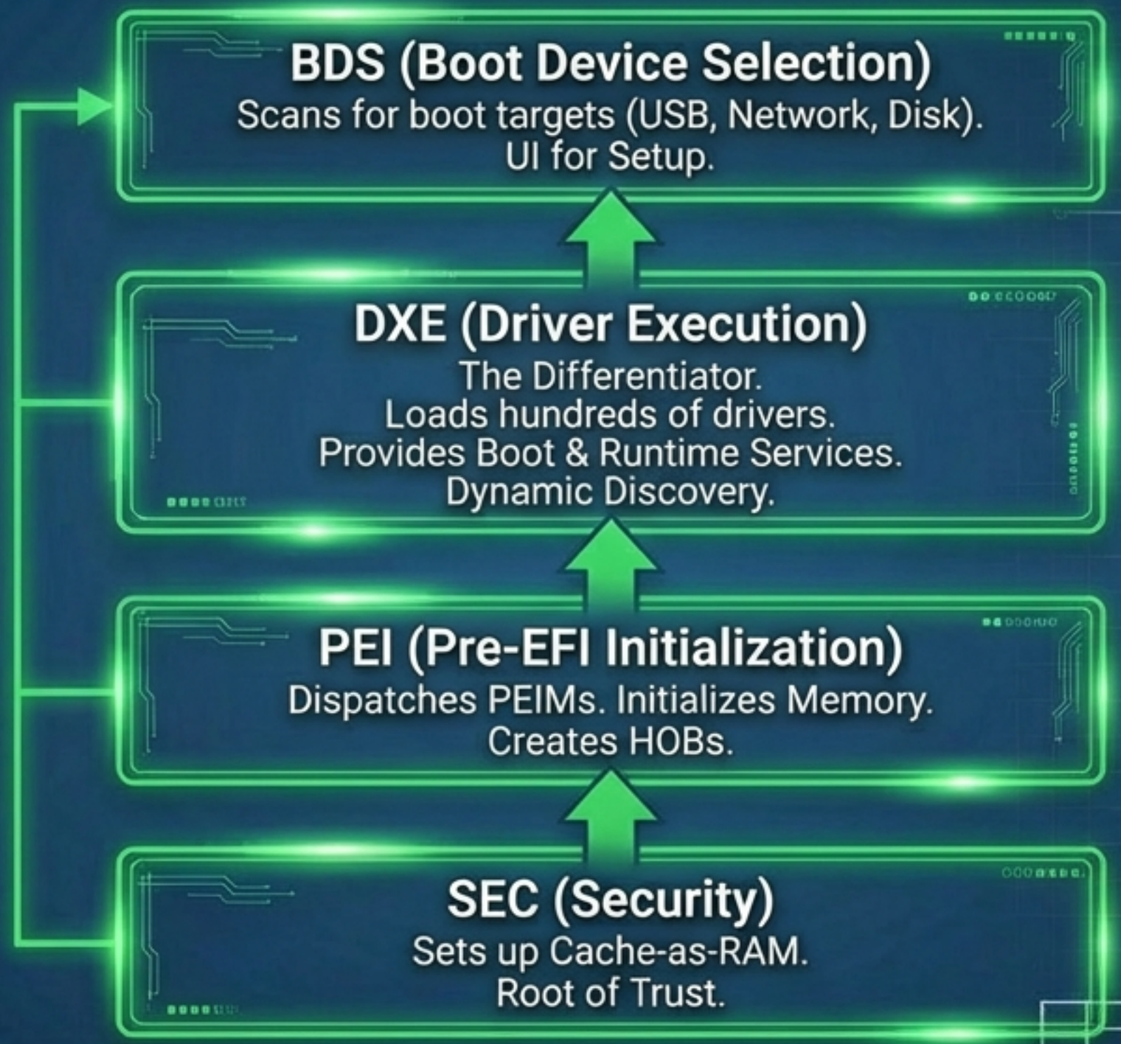


ENGINEERING DOCUMENTATION

CONFIDENTIAL

The Traditional Stack: UEFI/PI (EDK II)

Maximizing compatibility and extensibility.



ENGINEERING DOCUMENTATION

CONFIDENTIAL

Architectural Comparison Matrix

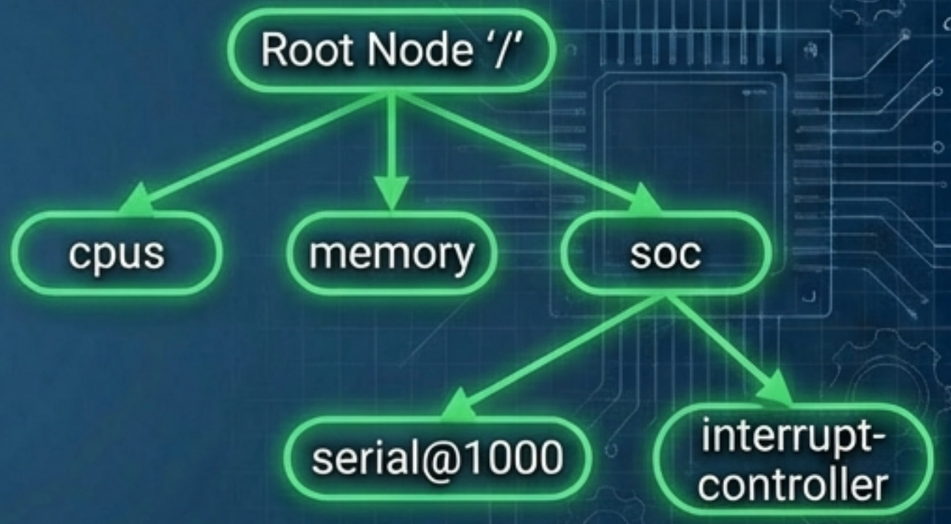
Feature	Intel SBL	ARM TF-A	UEFI/PI
Root of Trust	CPU Key + FPF Fuses	eFuse OEM Key	Platform Dependent
Secure Co-processor	CSME 18.x (Policy)	TrustZone (Isolation)	CSME or TEE
Early Memory	CAR (via FSP-T)	Secure SRAM	CAR (SEC Phase)
Runtime Services	None (Payload)	BL31 (PSCI/SMC)	UEFI Runtime Services
Handoff Data	HOB List + LdrGlobal	Device Tree (DTB)	HOBs -> Protocols

ENGINEERING DOCUMENTATION
CONFIDENTIAL

Data Structures: The Language of Handoffs

Intel/UEFI HOB List

ARM Device Tree Blob (DTB)



Binary structure passed via Register X0.
Describes hardware topology.

ENGINEERING DOCUMENTATION
CONFIDENTIAL

Architectural Philosophies & Trade-offs

Intel SBL

SPEED

Deterministic execution.
Static YAML configuration.
Minimal runtime residency.

Best for: IoT, Embedded, Fixed Function.

ARM TF-A

SECURITY

Hardware enforced isolation.
Runtime resident monitor (BL31).
TrustZone boundaries.

Best for: Mobile, Secure Edge.

UEFI/PI

COMPATIBILITY

Dynamic discovery.
Abstraction via Protocols.
Massive driver ecosystem.

Best for: Enterprise, General Purpose PC.

Developer Takeaways & References

Arrow Lake Development

- **Manifests**: KM/BPM are signed structures in flash. FIT only points to ACM.
- **Fuses**: Trust anchors are burned into PCH FPFs.
- **Payloads**: Use OsLoader for speed, UEFI Payload for Windows compatibility.

Verified References

1. Intel CSME Security White Paper v1.5
2. Arrow Lake Technical Guide (Boot Flow-4)
3. ARM Trusted Firmware-A Technical Reference Manual