CEC Continuing Education Center

DesignNews

Secure MCUs and RTOSs

DAY 4 : Secure Boot and Firmware Updates

Sponsored by

Digi-Key ELECTRONICS

BENINGO EMBEDDED GROUP

informa markets

Webinar Logistics

- Turn on your system sound to hear the streaming presentation.
- If you have technical problems, click “Help” or submit a question asking for assistance.
- Participate in ‘Group Chat’ by maximizing the chat widget in your dock.

THE SPEAKER

Jacob Beningo

Visit ‘Lecturer Profile’

Beningo Embedded Group - President

Focus: Embedded Software Consulting

An independent consultant who specializes in the design of real-time, microcontroller based embedded software.
He has published two books:

- Reusable Firmware Development
- MicroPython Projects
- Embedded Software Design

Writes a weekly blog for DesignNews.com focused on embedded system design techniques and challenges.

Visit www.beningo.com to learn more …

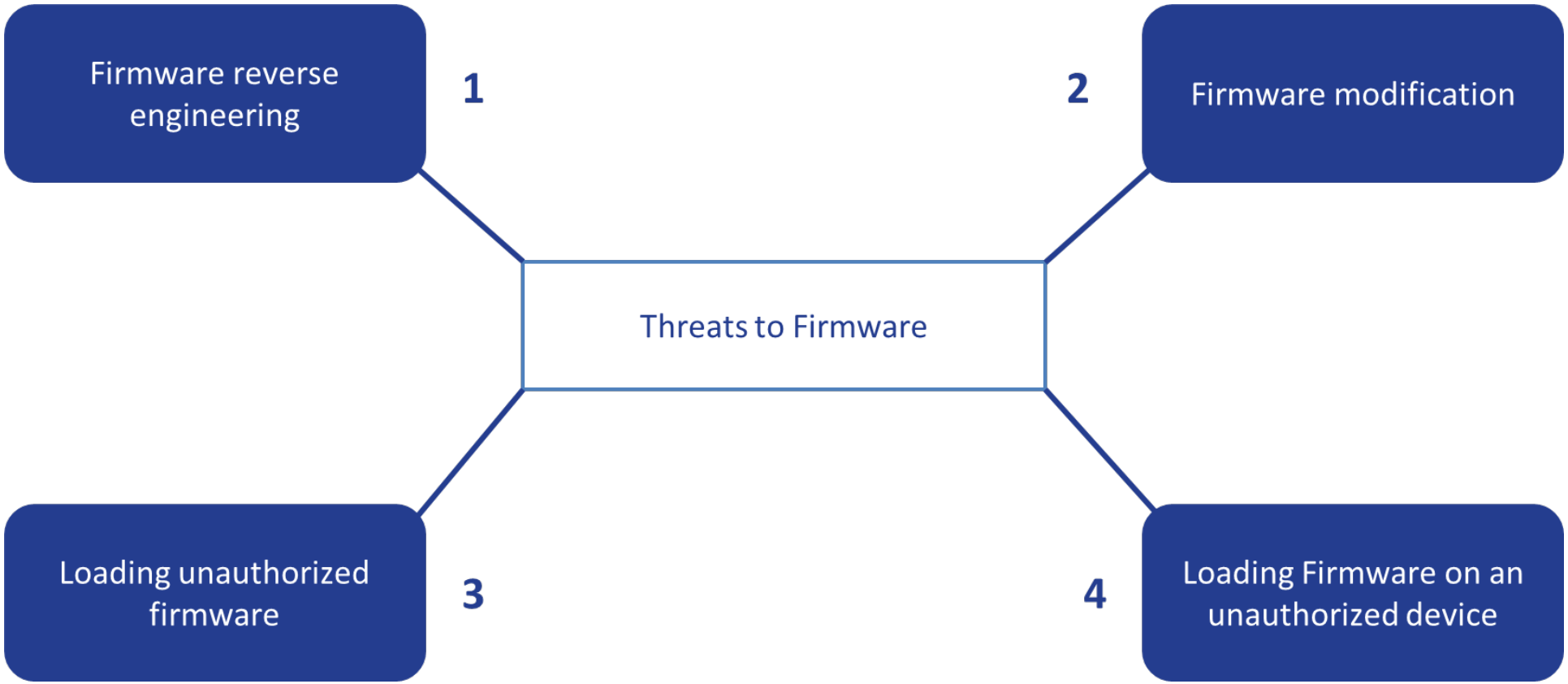
Visit ‘Lecturer Profile’ in your console for more details.

Course Sessions

- Threat Model Security Analysis (TMSA)
- Secure Microcontroller Solutions
- Arm TrustZone
- **Secure Boot and Firmware Updates**
- Secure RTOSes

1 Root of Trust

Root of Trust – Use Cases



Root of Trust

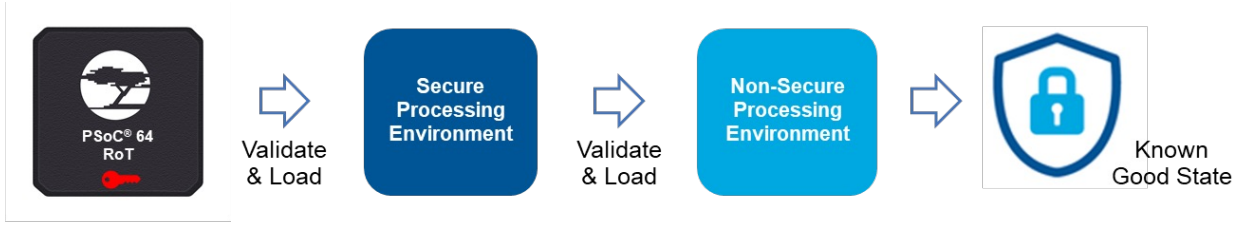
Root-of-Trust (RoT) – This is an immutable process or identity which is used as the first entity in a trust chain. No ancestor entity can provide a trustable attestation (in Digest or other form) for the initial code and data state of the Root of Trust.

Example:

The initial boot code stored in ROM which cannot be changed by users or Cypress provides the RoT for PSoC 64 Secure MCU's.

Root of Trust – PSoC® 64 Boot-time Security

Secure Boot and Secure Firmware Updates



Secure Boot

- Boot sequence validates image
  - Integrity: image has not been tampered with
  - Authenticity: image is from an authorized source
- Device boots to a known good state

Secure Firmware Updates

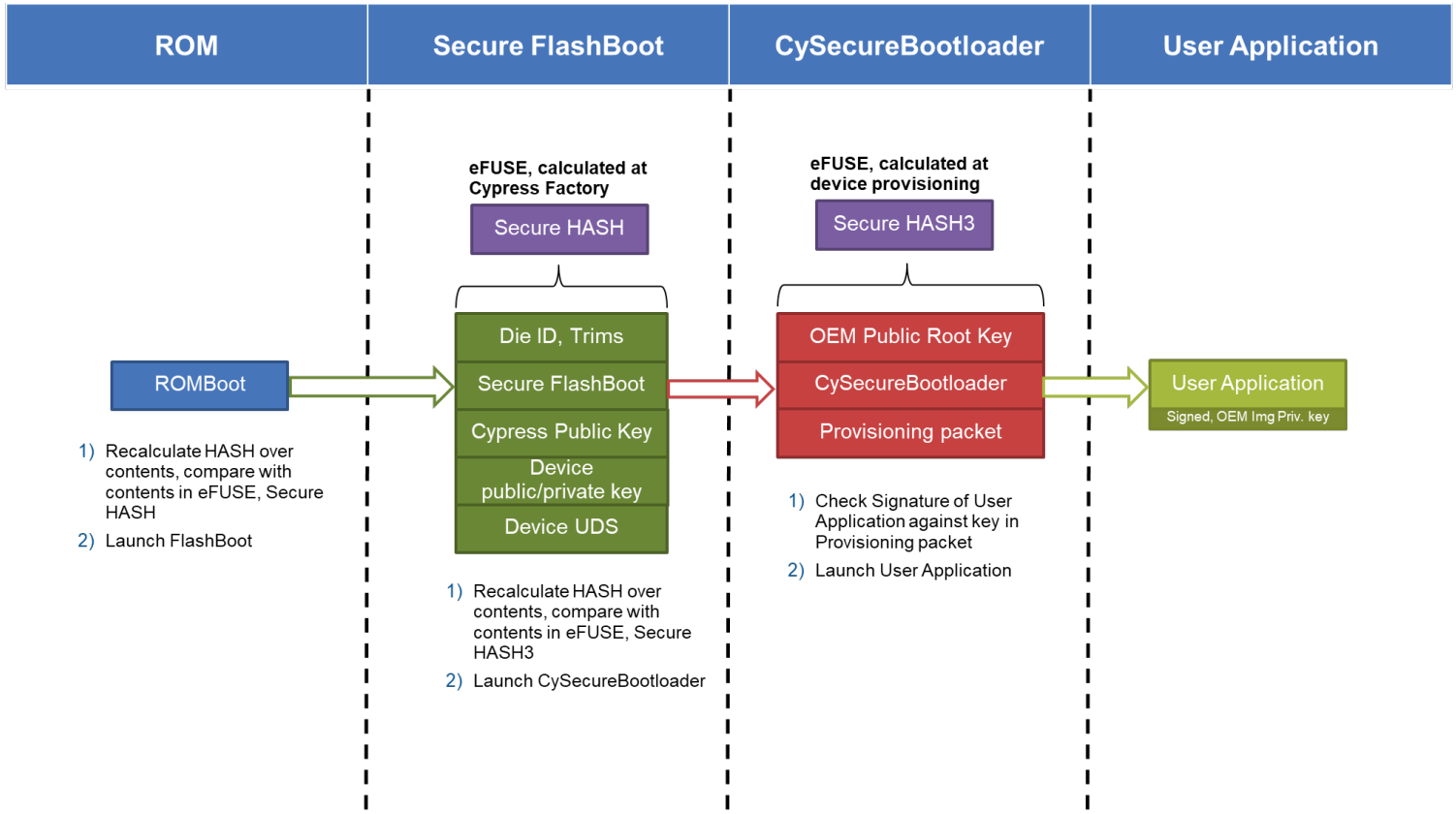
- Updated image can be stored and encrypted internally or with off-chip Quad SPI Flash
- Rollback protection prevents older firmware from being loaded with known vulnerabilities

PSoC 64 Root-of-Trust serves as the trust anchor for secure chain-of-trust

Do you currently use a RoT in your products?
- No
- Yes

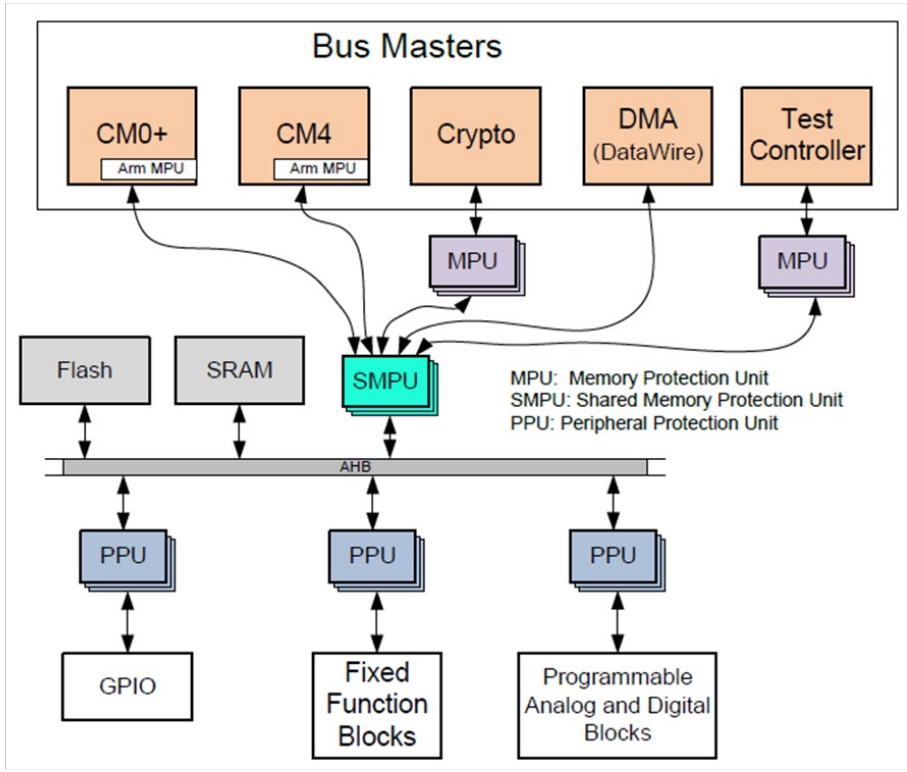
2 Secure Boot

Secure Boot – Boot Sequence

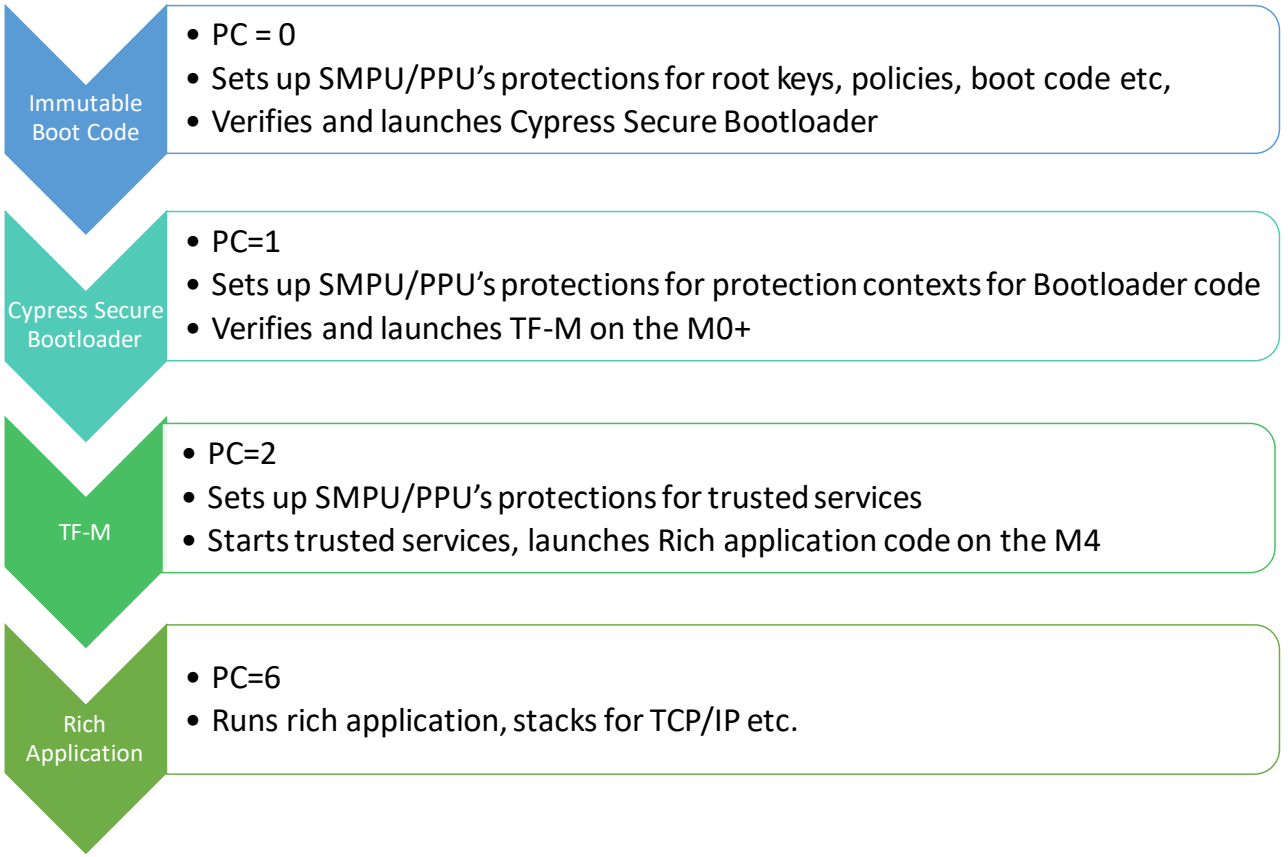


Secure Boot - Isolation

- 5 Bus Masters that can access the AHB and access Flash, SRAM and Peripherals
- Memory Protection Unit (MPU)
  - Provide high-level memory protection
  - Distinguish user and privileged access
- Shared Memory Protection Unit (SMPU)
  - Distinguishes between different protection contexts (PC)
  - Distinguishes secure from non-secure accesses
- Peripheral Protection Unit (PPU)
  - Manages access to individual peripheral blocks in different PC's, access and security states



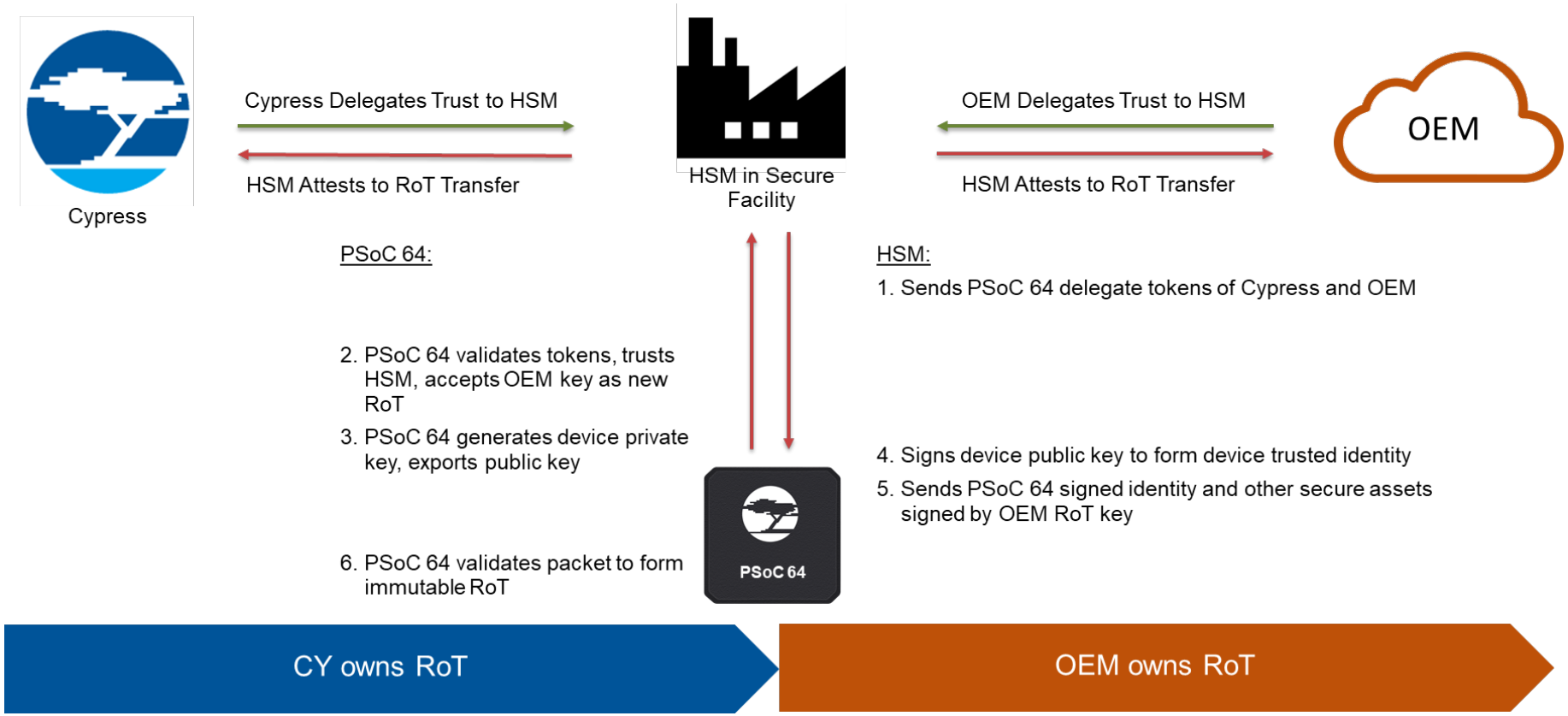
Secure Boot – Protection Contexts



What helps to improve the security of an embedded system?
- Using a RoT
- Implementing Secure Boot
- Leveraging Secure Updates
- Secure Peripherals and MPUs
- All the above
- None of the above

3 Secure Firmware Updates

Secure Firmware Updates – RoT Ownership Transfer



Secure Firmware Updates - Provisioning

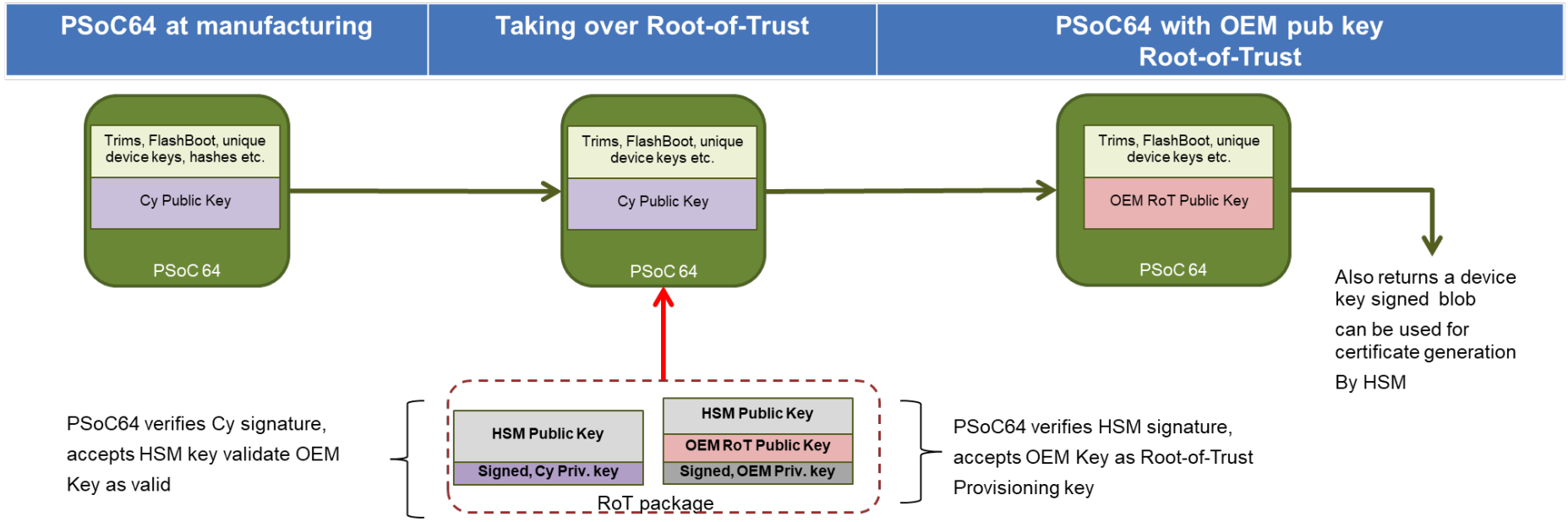
Provisioning - is a process where secure assets like keys and security policies are injected into the device. This step typically occurs in a secure manufacturing environment that has a Hardware Security Module (HSM). This process is irreversible. For the PSoC 64 Secure MCU, provisioning involves the following steps:

- Transferring the RoT from Cypress to the development user (called OEM in this course).
- Injecting user assets such as image-signing keys, device security policies, and certificates into the device.
- eFuses are blown (irreversible).

Provisioning the device can be done through the Cypress Secure Boot SDK.

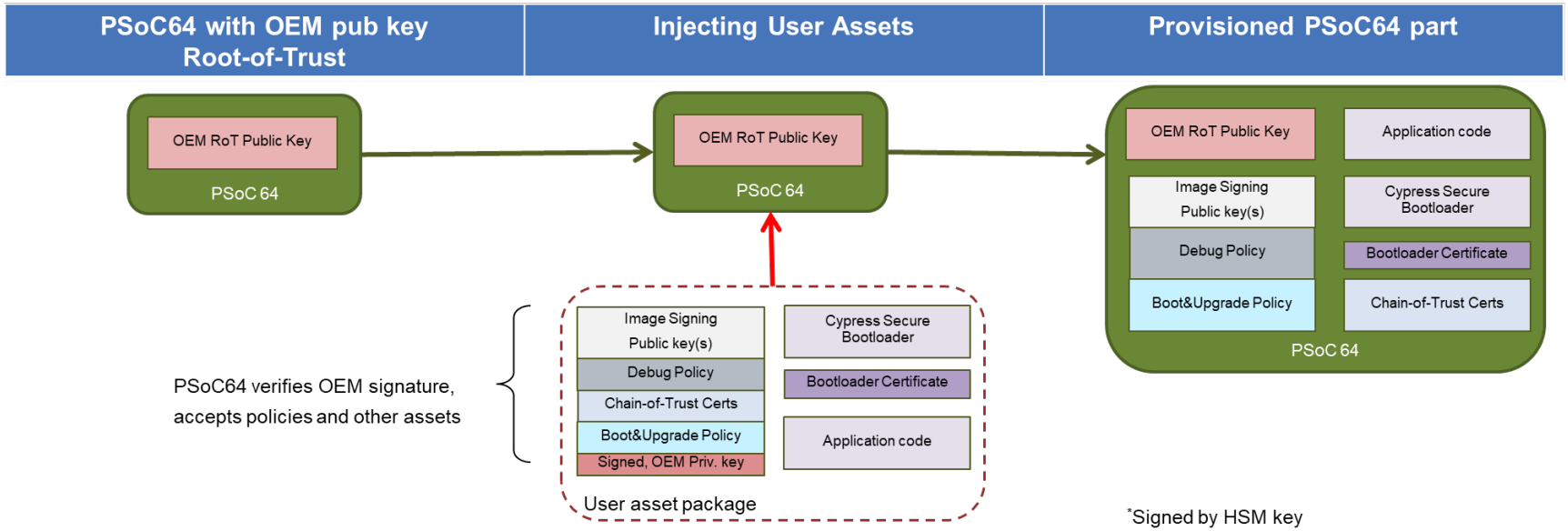
Secure Firmware Updates – Provisioning RoT

- Every PSoC 64 has a Cypress Public key
  - Secure FlashBoot enables provisioning process
  - Provisioning requires Cypress to authorize an HSM to inject OEM key
  - HSM signs OEM public key to allow Root-of-Trust to be transferred to OEM key



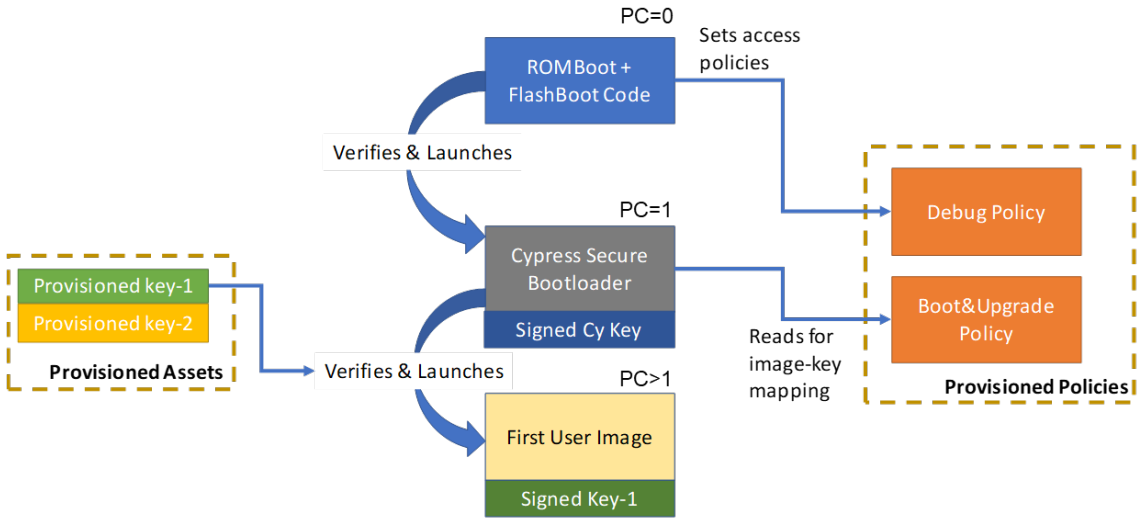
Secure Firmware Updates – Provisioning User Assets

- PSoC64 securely provisions user assets like,
  - Debug policies like, CM0+/CM4/SysAP DAP access ports
  - Image signing keys(typically are different from Root-of-Trust key)
  - Boot & Upgrade policies which specify key map to images, Slot sizes and addresses
  - Any certificates needed

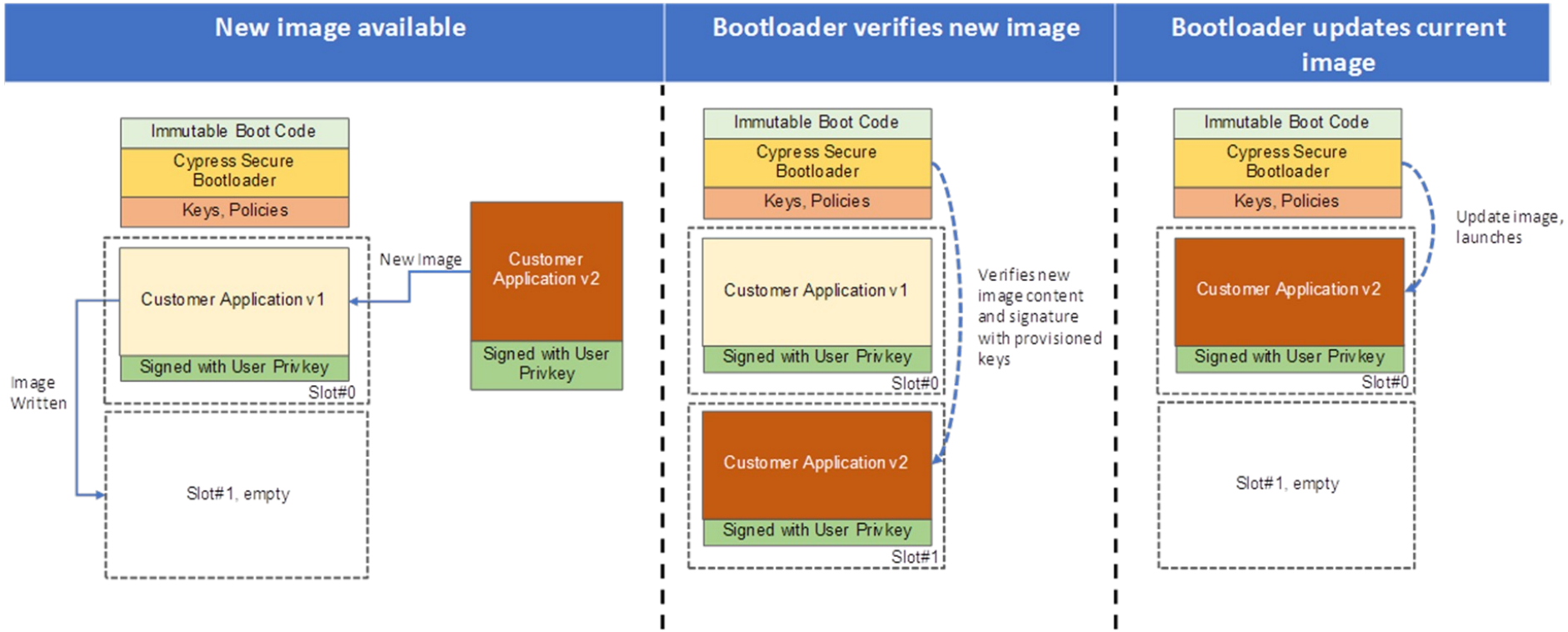


Secure Firmware Updates – PSoC 64 Secure Bootloader

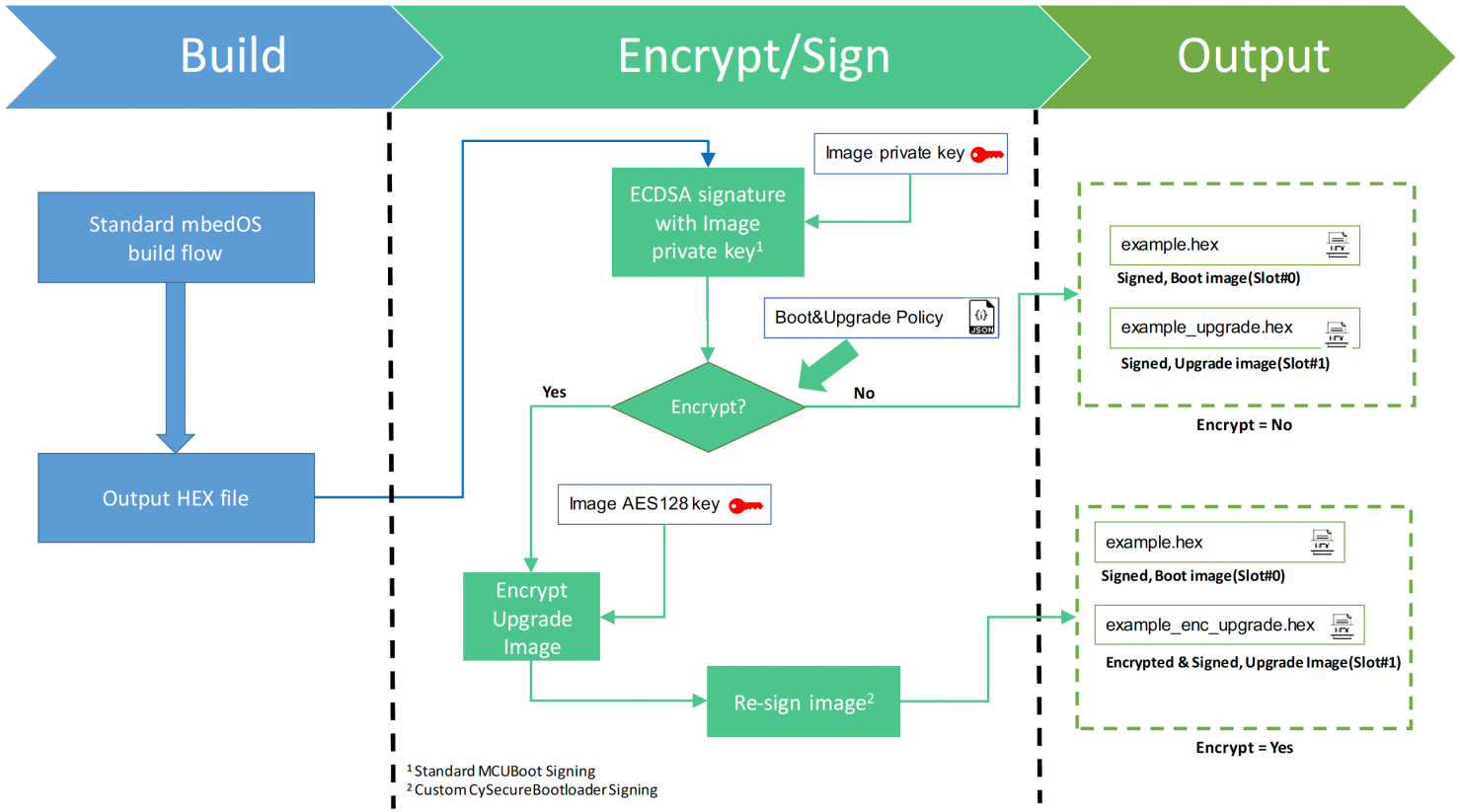
- The Cypress Secure Bootloader is a Cypress developed piece of firmware which
  - Implements MCUBoot library
  - Has knowledge of keys & policies to setup/launch first image
  - Can be considered an extension of FlashBoot; made immutable once provisioned



Secure Firmware Updates – PSoC 64 Secure Bootloader



Secure Firmware Updates – Upgrading the Firmware



What does a secure bootloader need to update firmware?
- Encryption Keys
- Hash algorithms
- Authentication mechanism
- None of the above
- All of the above

4 Going Further

Security and RTOS Resources

- Jacob's RTOS Blogs
- Jacob's RTOS courses
- Jacob's Security Blogs
- TrustZone for Cortex-M

- Embedded Bytes Newsletter
  - http://bit.ly/1BAHYXm

BENINGO EMBEDDED GROUP

www.beningo.com

CEC Continuing Education Center

DesignNews

Thank You

Sponsored by

Digi-Key ELECTRONICS

BENINGO EMBEDDED GROUP

informa markets