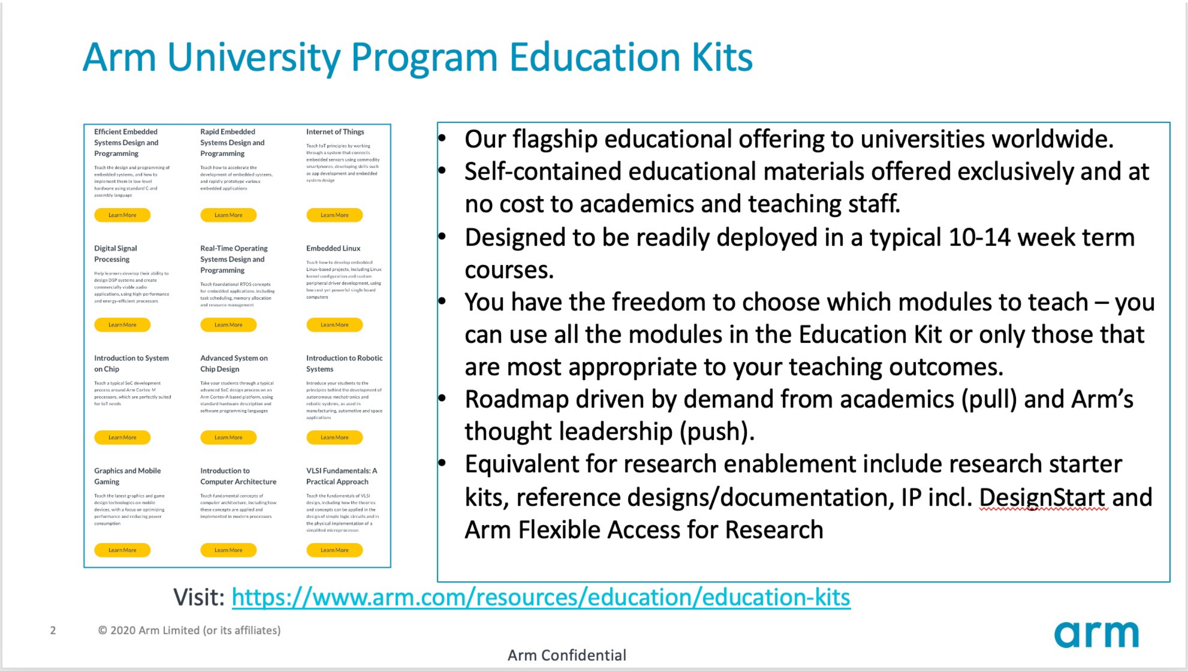
Extensible Secure SoC Platform

UKDF Workshop - Newcastle

John Goodenough
March 17th 2022



Arm University Program Education Kits

Efficient Embedded Systems Design and Programming
Teach the design and programming of embedded systems, and how to implement them in low-level hardware using standard C and assembly language
Learn More

Rapid Embedded Systems Design and Programming
Teach how to accelerate the development of embedded systems, and rapidly prototype various embedded applications
Learn More

Internet of Things
Teach IoT principles by working through a system that connects embedded sensors using commodity smartphones, developing skills such as app development and embedded system design
Learn More

Digital Signal Processing
Help learners develop their ability to design DSP systems and create commercially viable audio applications, using high-performance and energy-efficient processors
Learn More

Real-Time Operating Systems Design and Programming
Teach foundational RTOS concepts for embedded applications, including task scheduling, memory allocation and resource management
Learn More

Embedded Linux
Teach how to develop embedded Linux-based projects, including Linux kernel configuration and custom peripheral driver development, using low cost yet powerful single board computers
Learn More

Introduction to System on Chip
Teach a typical SoC development process around Arm Cortex-M processors, which are perfectly suited for IoT needs
Learn More

Advanced System on Chip Design
Take your students through a typical advanced SoC design process on an Arm Cortex-A based platform, using standard hardware description and software programming languages
Learn More

Introduction to Robotic Systems
Introduce your students to the principles behind the development of autonomous mechatronics and robotic systems, as used in manufacturing, automotive and space applications
Learn More

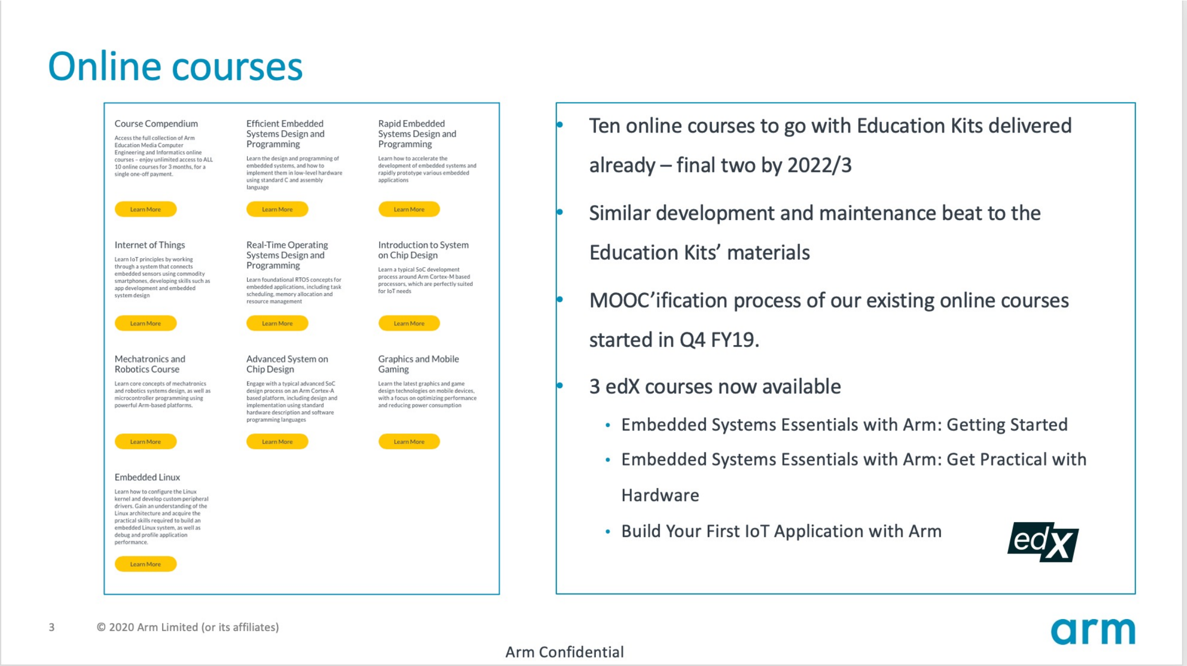
Graphics and Mobile Gaming
Teach the latest graphics and game design technologies on mobile devices, with a focus on optimizing performance and reducing power consumption
Learn More

Introduction to Computer Architecture
Teach fundamental concepts of computer architecture, including how these concepts are applied and implemented in modern processors
Learn More

VLSI Fundamentals: A Practical Approach
Teach the fundamentals of VLSI design, including how the theories and concepts can be applied in the design of simple logic circuits and in the physical implementation of a simplified microprocessor.
Learn More

- Our flagship educational offering to universities worldwide.
- Self-contained educational materials offered exclusively and at no cost to academics and teaching staff.
- Designed to be readily deployed in a typical 10-14 week term courses.
- You have the freedom to choose which modules to teach – you can use all the modules in the Education Kit or only those that are most appropriate to your teaching outcomes.
- Roadmap driven by demand from academics (pull) and Arm's thought leadership (push).
- Equivalent for research enablement include research starter kits, reference designs/documentation, IP incl. DesignStart and Arm Flexible Access for Research

Visit: https://www.arm.com/resources/education/education-kits



Online courses

Course Compendium
Access the full collection of Arm Education Media Computer Engineering and Informatics online courses – enjoy unlimited access to ALL 10 online courses for 3 months, for a single one-off payment.
Learn More

Efficient Embedded Systems Design and Programming
Learn the design and programming of embedded systems, and how to implement them in low-level hardware using standard C and assembly language
Learn More

Rapid Embedded Systems Design and Programming
Learn how to accelerate the development of embedded systems and rapidly prototype various embedded applications
Learn More

Internet of Things
Learn IoT principles by working through a system that connects embedded sensors using commodity smartphones, developing skills such as app development and embedded system design
Learn More

Real-Time Operating Systems Design and Programming
Learn foundational RTOS concepts for embedded applications, including task scheduling, memory allocation and resource management
Learn More

Introduction to System on Chip Design
Learn a typical SoC development process around Arm Cortex-M based processors, which are perfectly suited for IoT needs
Learn More

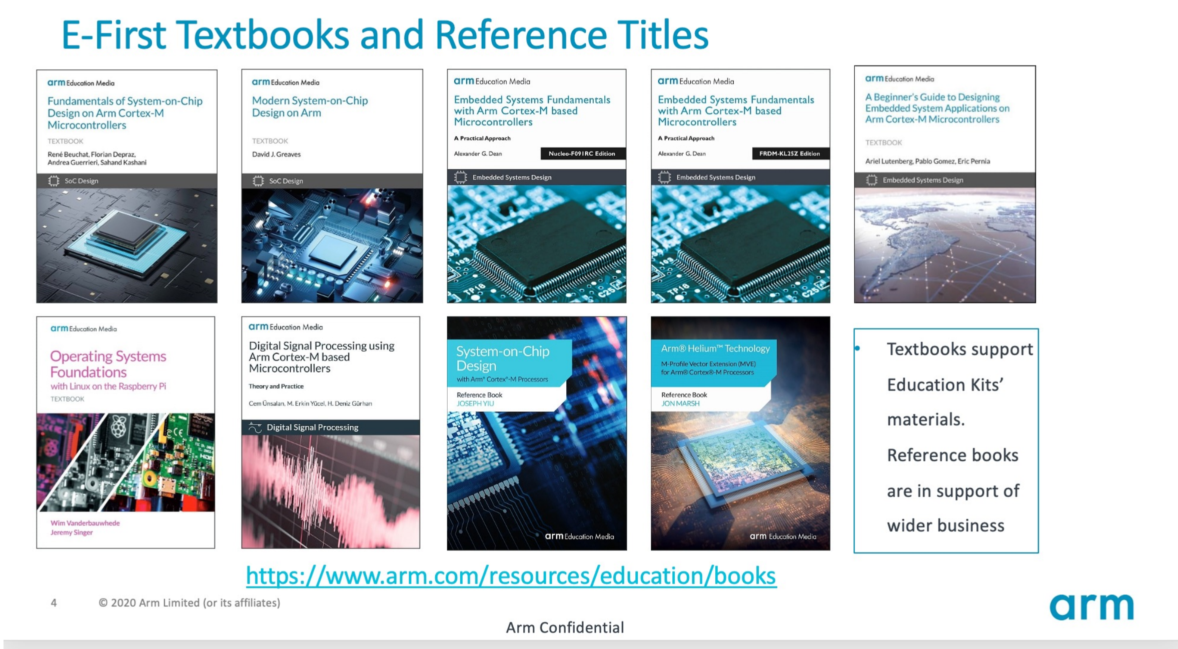
Mechatronics and Robotics Course
Learn core concepts of mechatronics and robotics systems design, as well as microcontroller programming using powerful Arm-based platforms.
Learn More

Advanced System on Chip Design
Engage with a typical advanced SoC design process on an Arm Cortex-A based platform, including design and implementation using standard hardware description and software programming languages
Learn More

Graphics and Mobile Gaming
Learn the latest graphics and game design technologies on mobile devices, with a focus on optimizing performance and reducing power consumption
Learn More

Embedded Linux
Learn how to configure the Linux kernel and develop custom peripheral drivers. Gain an understanding of the Linux architecture and acquire the practical skills required to build an embedded Linux system, as well as debug and profile application performance.
Learn More

- Ten online courses to go with Education Kits delivered already – final two by 2022/3
- Similar development and maintenance beat to the Education Kits' materials
- MOOC'ification process of our existing online courses started in Q4 FY19.
- 3 edX courses now available
  - Embedded Systems Essentials with Arm: Getting Started
  - Embedded Systems Essentials with Arm: Get Practical with Hardware
  - Build Your First IoT Application with Arm



E-First Textbooks and Reference Titles

arm Education Media
Fundamentals of System-on-Chip Design on Arm Cortex-M Microcontrollers
TEXTBOOK
René Beuchat, Florian Depraz, Andrea Guerrieri, Sahand Kashani
SoC Design

arm Education Media
Modern System-on-Chip Design on Arm
TEXTBOOK
David J. Greaves
SoC Design

arm Education Media
Embedded Systems Fundamentals with Arm Cortex-M based Microcontrollers
A Practical Approach
Alexander G. Dean
Nucleo-F091RC Edition
Embedded Systems Design

arm Education Media
Embedded Systems Fundamentals with Arm Cortex-M based Microcontrollers
A Practical Approach
Alexander G. Dean
FRDM-KL25Z Edition
Embedded Systems Design

arm Education Media
A Beginner's Guide to Designing Embedded System Applications on Arm Cortex-M Microcontrollers
TEXTBOOK
Ariel Lutenberg, Pablo Gomez, Eric Pernia
Embedded Systems Design

arm Education Media
Operating Systems Foundations
with Linux on the Raspberry Pi
TEXTBOOK
Wim Vanderbauwhede
Jeremy Singer

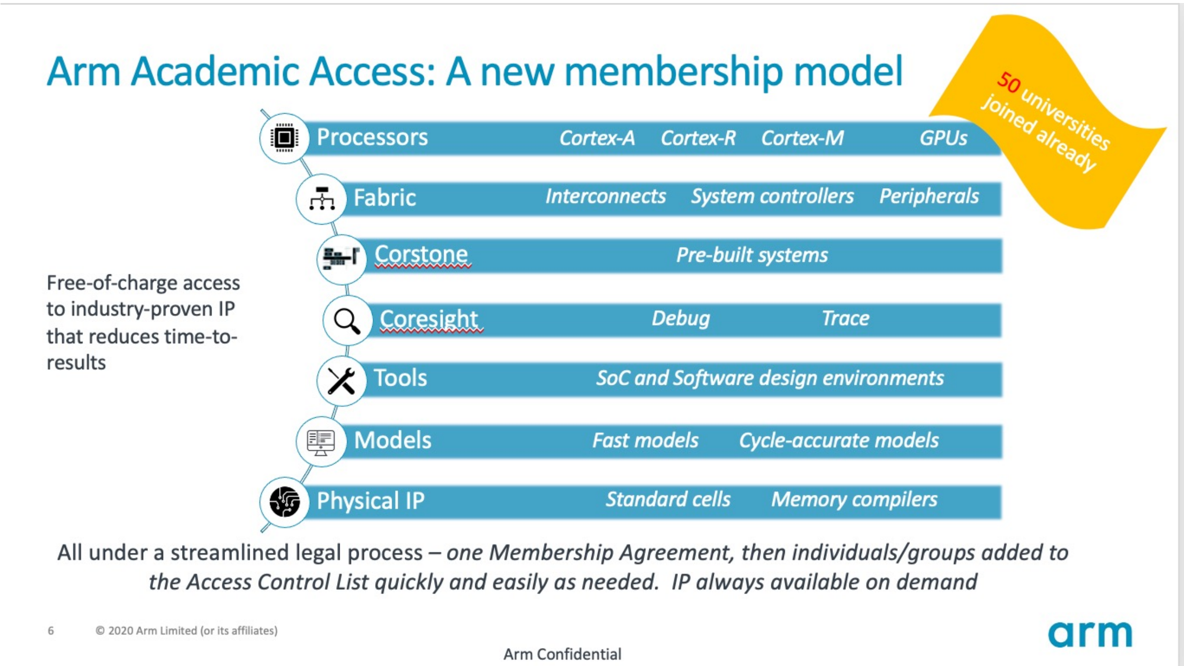
arm Education Media
Digital Signal Processing using Arm Cortex-M based Microcontrollers
Theory and Practice
Cem Ünsalan, M. Erkin Yücel, H. Deniz Gürhan
Digital Signal Processing

System-on-Chip Design
with Arm® Cortex®-M Processors
Reference Book
JOSEPH YIU
arm Education Media

Arm® Helium™ Technology
M-Profile Vector Extension (MVE) for Arm® Cortex®-M Processors
Reference Book
JON MARSH
arm Education Media

- Textbooks support Education Kits' materials. Reference books are in support of wider business

https://www.arm.com/resources/education/books



Arm Academic Access: A new membership model

Free-of-charge access to industry-proven IP that reduces time-to-results

All under a streamlined legal process – *one Membership Agreement, then individuals/groups added to the Access Control List quickly and easily as needed. IP always available on demand*

Infomerical: Find out more:

- **Arm Research**: http://www.arm.com/resources/research
  - Explore our research areas in more detail: www.arm.com/resources/research/impact-report
  - Find out more about how to collaborate with us: https://developer.arm.com/solutions/research/research-collaborations

- **Arm Academic Access:**
  - Increase your research impact with free access to the world's leading SoC design portfolio
  - https://www.arm.com/resources/research/rce/academic-access

- **Arm Education**:
  - Helping close education and skills gaps in Computer Engineering and STEM
  - https://www.arm.com/resources/education

Motivation

- Intelligent Edge Devices optimized for domain-specific operating constraints will be a more significant component of the future compute substrate
  - Post-Moore Domain Specific Accelerator integration, secure management of data and analysis
  - Edge service operations constraints mandate aggressive power operating states and threat resistance

- Device: Includes software stack exists within a Device Management context

- Innovation including academic research innovation is active at a number of TRL levels
  - Integrating a variety of novel compute methods with differing PPA potential, novel sensors and security capabilities.
  - Prototyping new semiconductor materials for RF or analog processing or explore new approaches to heterogenous integration.

- A common platform to enable innovators to focus enables more efficient innovation and a rapid pathway to deployment
- Target a minimum 5x reduction in deployment NRE through common SoC architecture & automation

- ***Access to IP Portfolio is necessary but not sufficient for complex modern SoC need Reference Design pattern to support research and innovation ecosystems***

AISS/CANASTA Overview – Challenges shared in academia

- Problem Statement
  - Current DoD performers & Innovation community have high NRE costs and limited expertise to design, integrate and **deploy** differentiated (SWaP optimized) Systems & SoCs which are secured with novel or emerging capabilities against a variety of Threat Vectors.
  - Access to cutting edge commercial technologies or novel security IP is difficult and difficult to integrate
  - It is difficult to estimate the cost of different security measures and options and trade off against system performance power and area

- DARPA's Project Goal
  - 'Automated' Integration for a configurable 'secure' SoC.
    - **Design Methodology for Secure SoC Integration** – Front End Design & SW integration Focused (*stops at Gates*)
    - **Secure SoC Architectural Framework (in AMI* context)**, IP Library, PoC SoC Platform PoC AMI PoC
    - **Secure Cloud Design Infrastructure (in AMI context)** – Enterprise Architecture for SoC Design with Federated Attested AMI
    - **Vulnerability testing** of SoC in AMI system context – define metrics/certification and test AMI/Device integration in secure factory floor emulation

    - The delivered Methodology, Architecture Framework and Example Security IP will be a commercial supported. Design flow, IP/SW Catalog, SoC Integration and Implementation tools

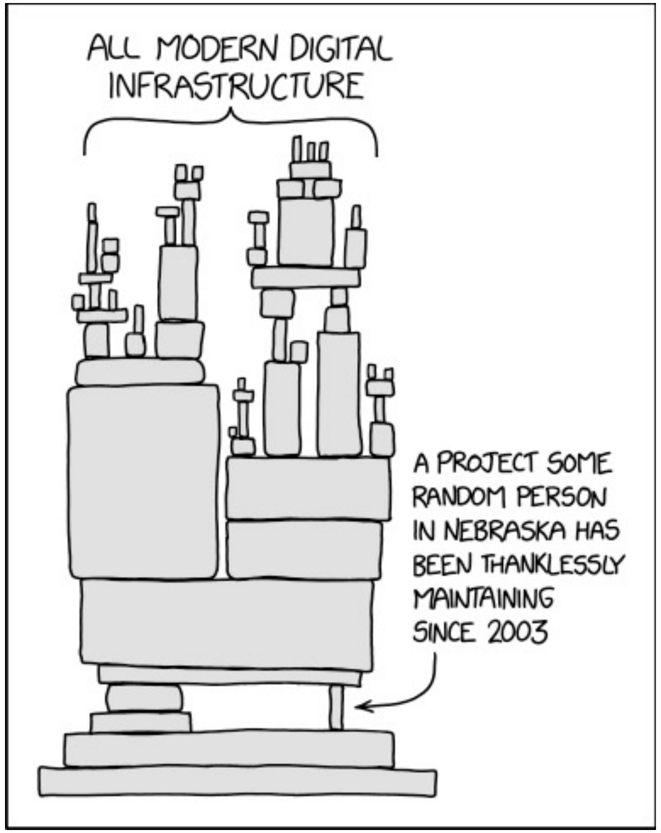
** AMI Asset Management Infrastructure*

Secure SoC - Ontology

Closed source and secure room strategies have acted as barriers

Security, Classic SoC and Fleet Management historic silos

Dependency Management


ALL MODERN DIGITAL INFRASTRUCTURE
A PROJECT SOME RANDOM PERSON IN NEBRASKA HAS BEEN THANKLESSLY MAINTAINING SINCE 2003

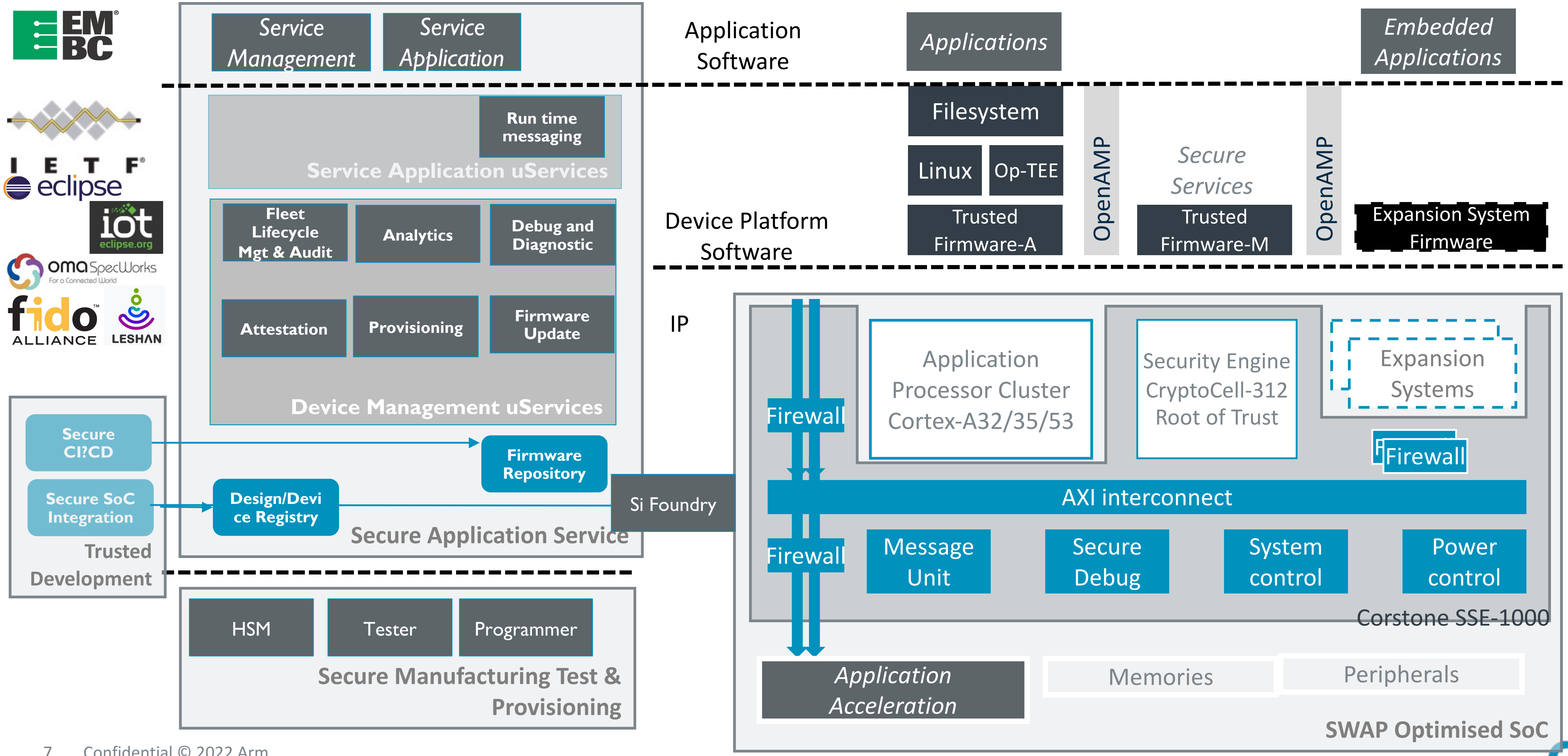
Secure SoC Composition

- Hardware APIs
- Software APIs
- Service APIs
- Protocols
- IP packaging for reuse in tools – HW & SW
- Floorplan/BOM Integration (eg space qual)

- Research Question: Do different strategies and design patterns result in better performance easier integration and performance?

Extensible & Configurable SoC & System Reference Architecture

Based on Extant Open Access/Open-source Design IP & Software and Industry standards



Service Management

Service Application

Run time messaging

Service Application uServices

Fleet Lifecycle Mgt & Audit

Analytics

Debug and Diagnostic

Attestation

Provisioning

Firmware Update

Device Management uServices

Secure CI?CD

Secure SoC Integration

Trusted Development

Firmware Repository

Design/Device Registry

Secure Application Service

Si Foundry

HSM

Tester

Programmer

Secure Manufacturing Test & Provisioning

Application Software

Applications

Embedded Applications

Device Platform Software

Filesystem

Linux

Op-TEE

Trusted Firmware-A

OpenAMP

Secure Services

Trusted Firmware-M

OpenAMP

Expansion System Firmware

IP

Firewall

Application Processor Cluster Cortex-A32/35/53

Security Engine CryptoCell-312 Root of Trust

Expansion Systems

Firewall

AXI interconnect

Firewall

Message Unit

Secure Debug

System control

Power control

Corstone SSE-1000

Application Acceleration

Memories

Peripherals

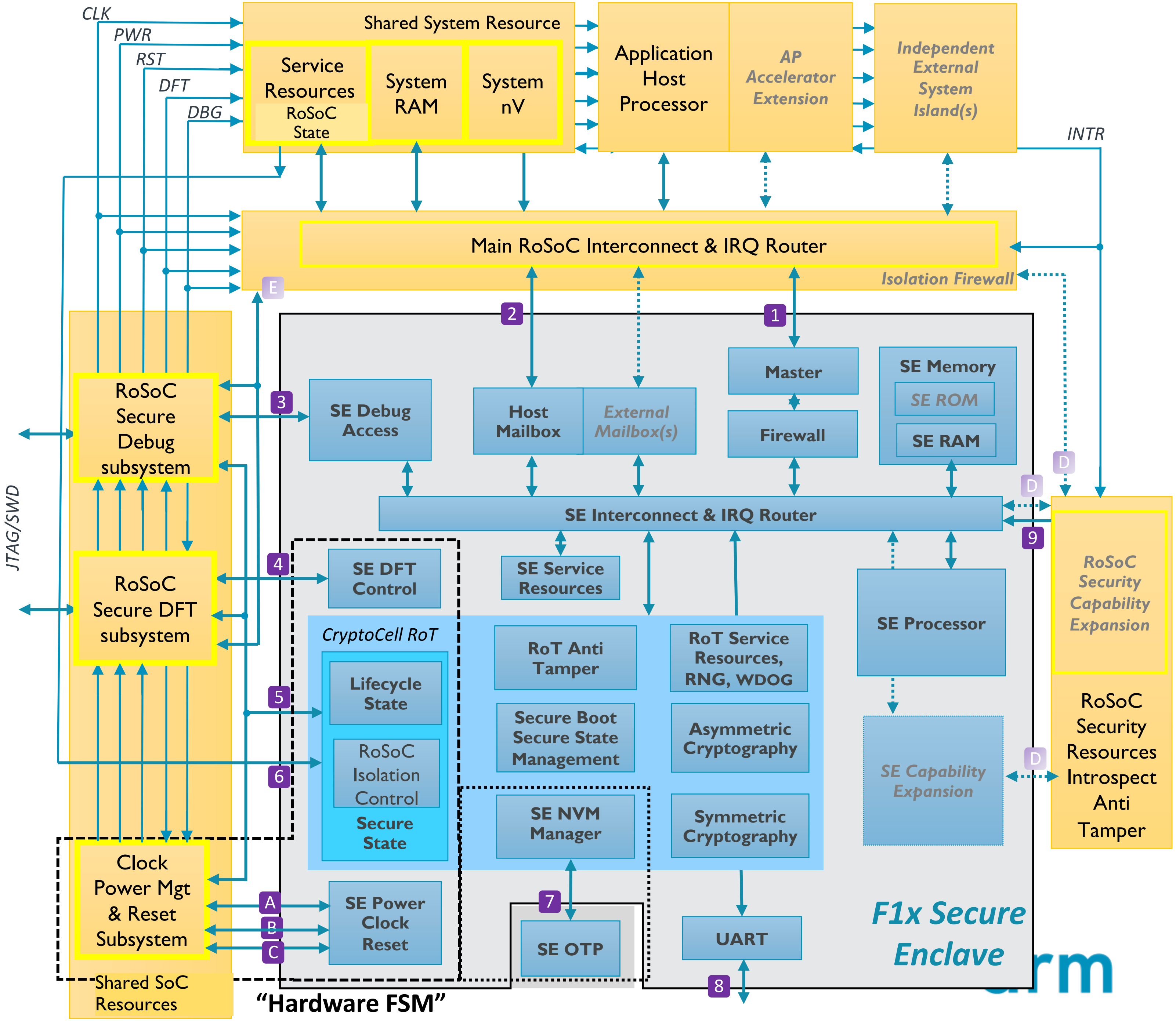
SWAP Optimised SoC

Software Available Open Source Hardware Available Open Access through AAA

AISS Secure SoC SE Integration

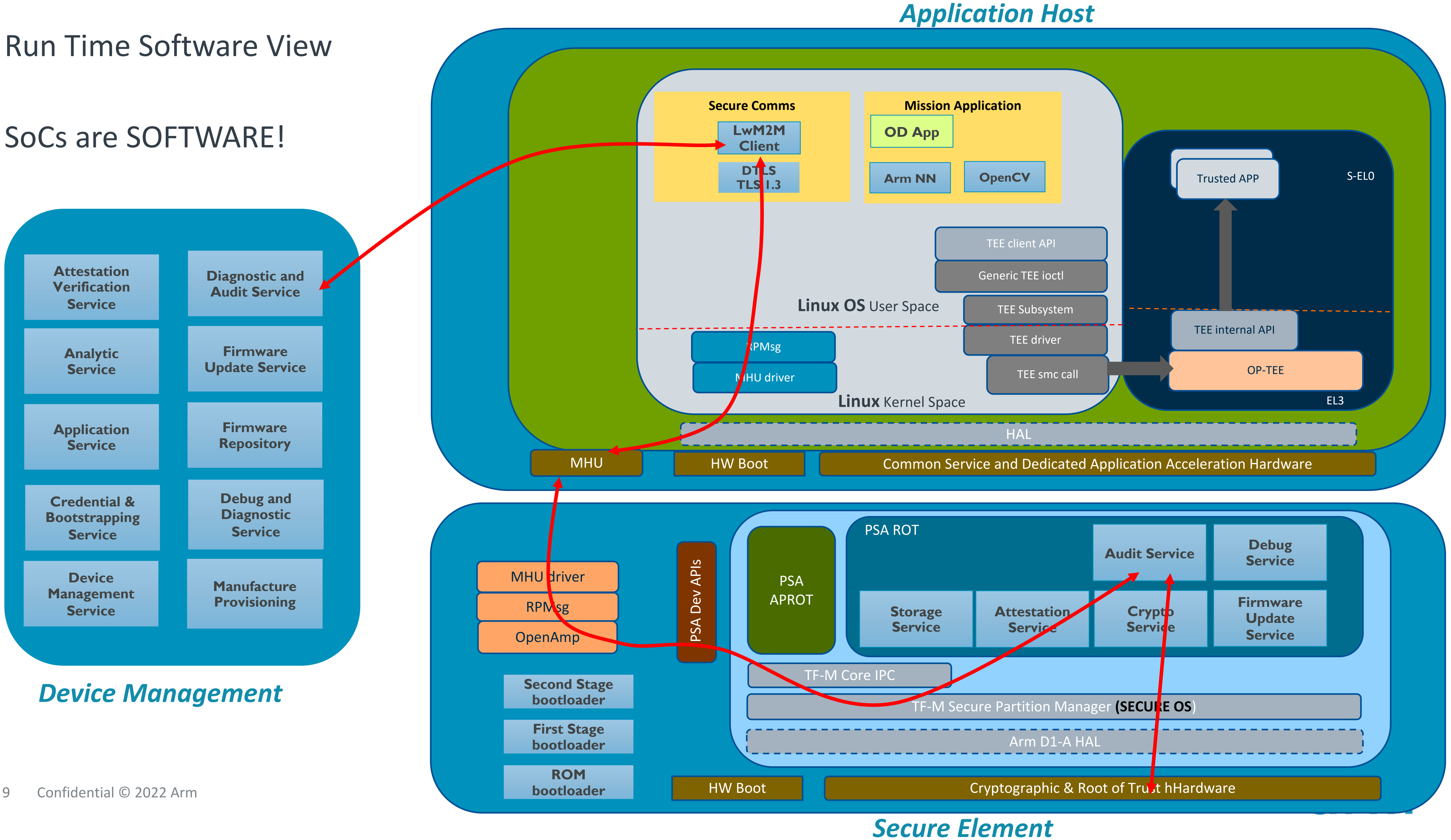
- Enclave provides secure system services
- Secure Boot and Configuration
- Cryptographic

- Common microcontroller platform
- Common integration to RoSoC
- Differentiate Security Capabilities – Entropy, Crypto, Introspection



Run Time Software View

SoCs are SOFTWARE!



Application Host

Secure Comms

LwM2M Client

DTLS TLS 1.3

Mission Application

OD App

Arm NN

OpenCV

Trusted APP

S-EL0

TEE client API

Generic TEE ioctl

Linux OS User Space

TEE Subsystem

TEE driver

TEE internal API

RPMsg

MHU driver

TEE smc call

OP-TEE

Linux Kernel Space

EL3

HAL

MHU

HW Boot

Common Service and Dedicated Application Acceleration Hardware

Attestation Verification Service

Diagnostic and Audit Service

Analytic Service

Firmware Update Service

Application Service

Firmware Repository

Credential & Bootstrapping Service

Debug and Diagnostic Service

Device Management Service

Manufacture Provisioning

Device Management

MHU driver

RPMsg

OpenAmp

PSA Dev APIs

PSA APROT

PSA ROT

Audit Service

Debug Service

Storage Service

Attestation Service

Crypto Service

Firmware Update Service

TF-M Core IPC

TF-M Secure Partition Manager **(SECURE OS)**

Arm D1-A HAL

Second Stage bootloader

First Stage bootloader

ROM bootloader

HW Boot

Cryptographic & Root of Trust hHardware

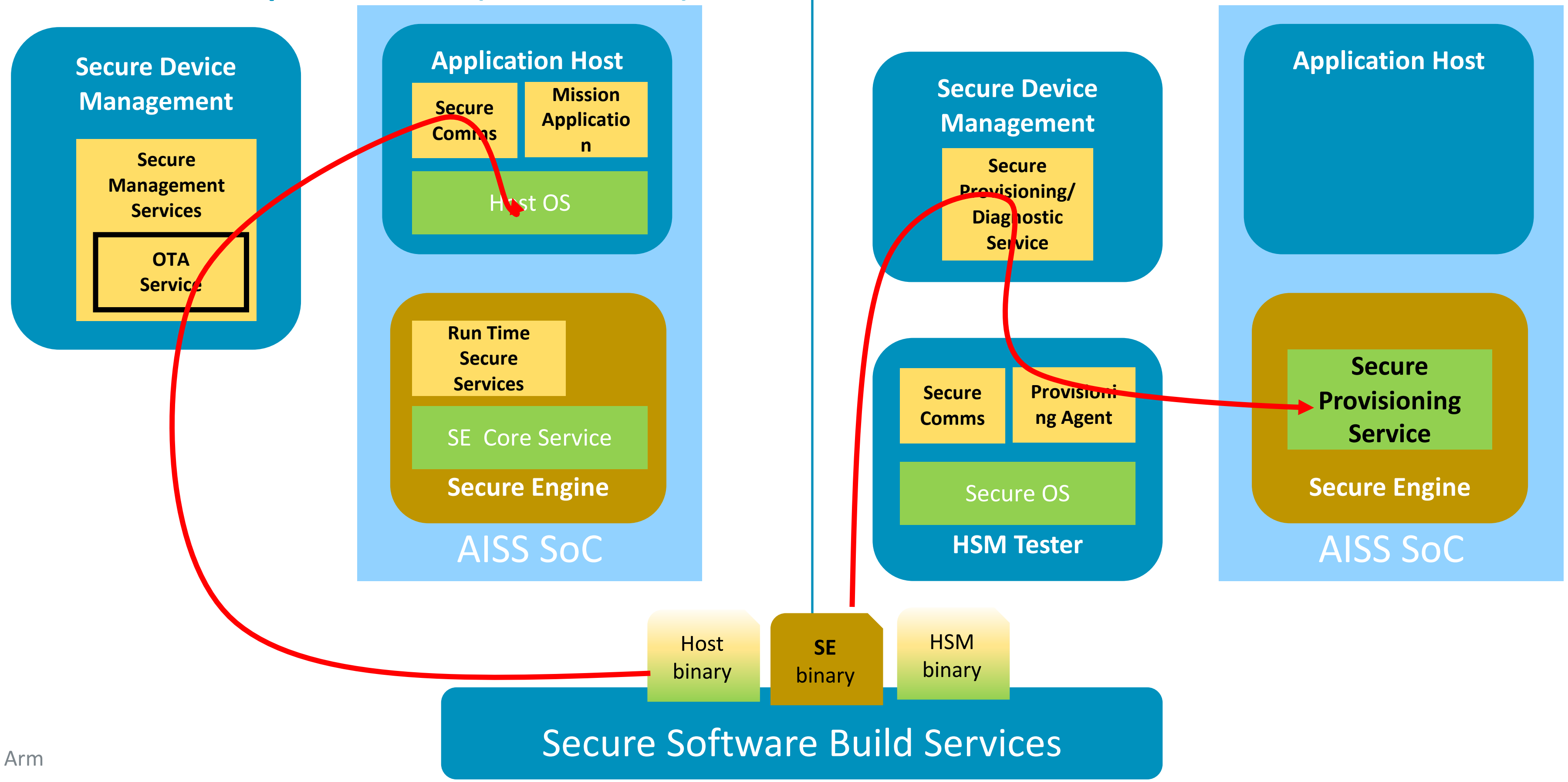
Secure Element

Software/Services for the AISS SoC Device

Different Software for the Secure Engine required in Different Lifecycle States

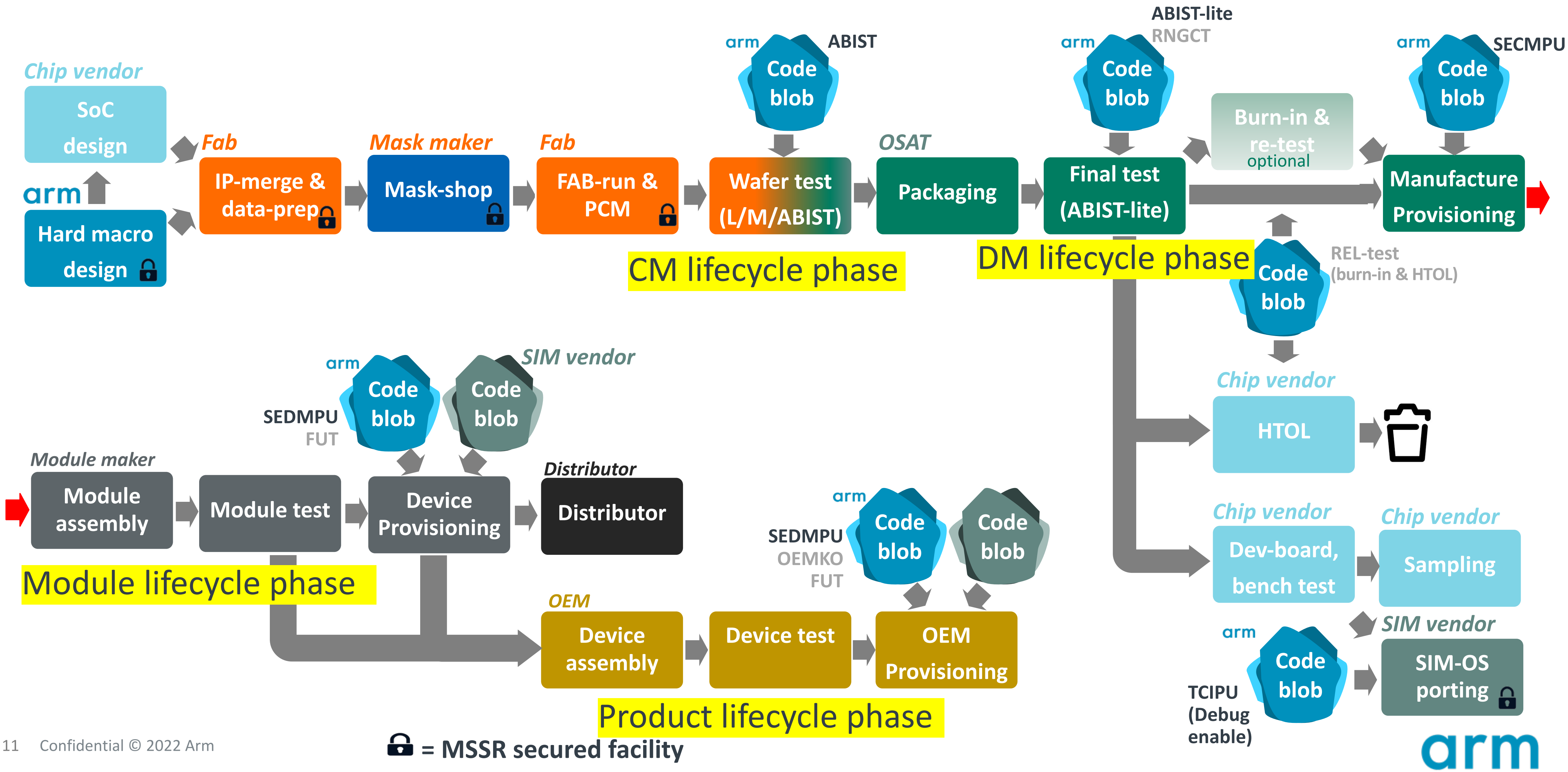
Demonstrating PPAS
Application Service Operation (Runtime)

Demonstrating Zero-Trust Supply chain Provisioning & RMA



- Kernel space
- User space
- REE/TEE
- Immutable SE

Enter the Fractal: Manufacturing flow & provisioning



Chip vendor
SoC design

arm
Hard macro design 🔒

Fab
IP-merge & data-prep 🔒

Mask maker
Mask-shop 🔒

Fab
FAB-run & PCM 🔒

arm
Code blob
ABIST

Wafer test (L/M/ABIST)

CM lifecycle phase

OSAT
Packaging

arm
Code blob
ABIST-lite
RNGCT

Final test (ABIST-lite)

DM lifecycle phase

Burn-in & re-test
optional

arm
Code blob
SECMPU

Manufacture Provisioning

Code blob
REL-test
(burn-in & HTOL)

Chip vendor
HTOL

Chip vendor
Dev-board, bench test

Chip vendor
Sampling

SIM vendor
SIM-OS porting 🔒

arm
Code blob
TCIPU
(Debug enable)

arm
Code blob
SEDMPU
FUT

SIM vendor
Code blob

Module maker
Module assembly

Module test

Device Provisioning

Distributor
Distributor

Module lifecycle phase

arm
Code blob
SEDMPU
OEMKO
FUT

Code blob

OEM
Device assembly

Device test

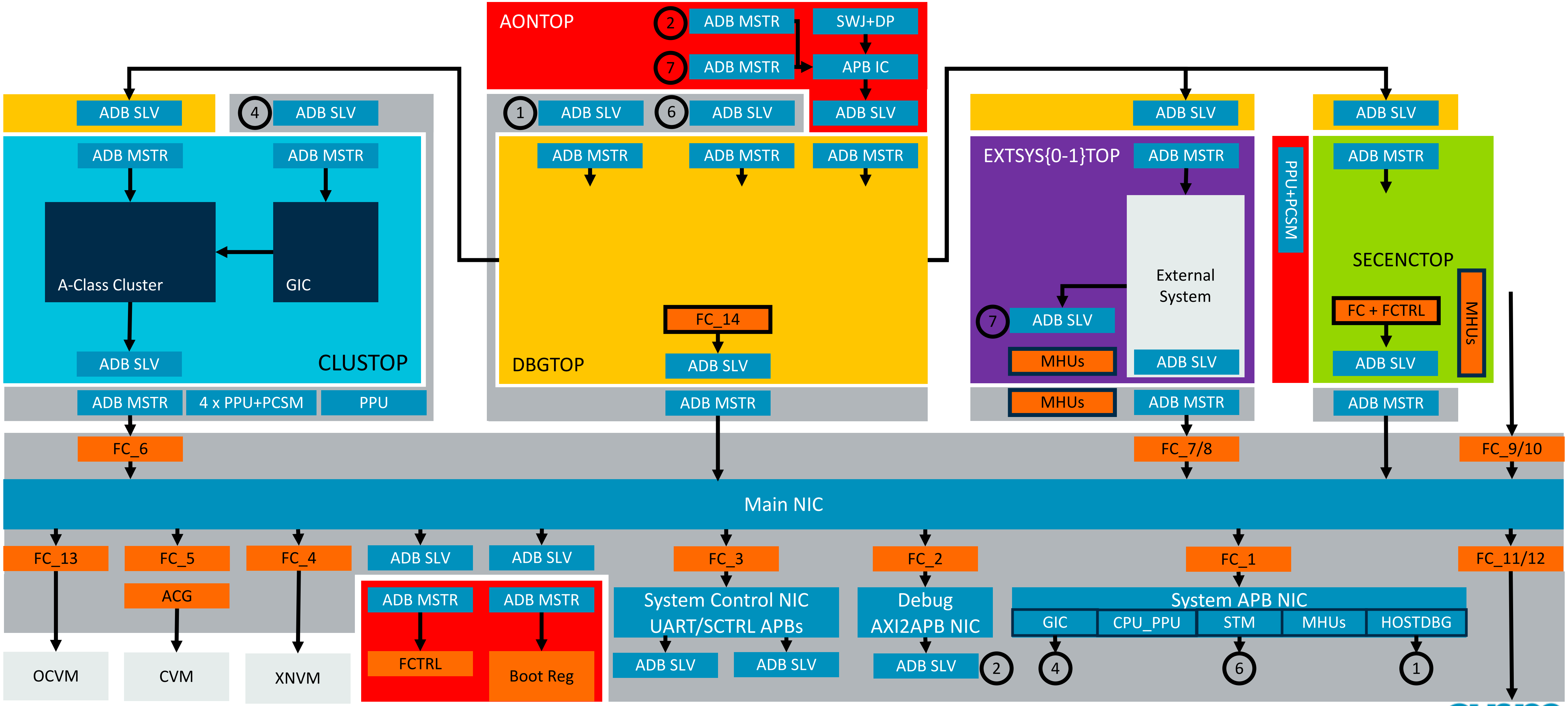
OEM Provisioning

Product lifecycle phase

🔒 = MSSR secured facility

SoC Design Pattern for Aggressive Power Mangement

Power domain topology

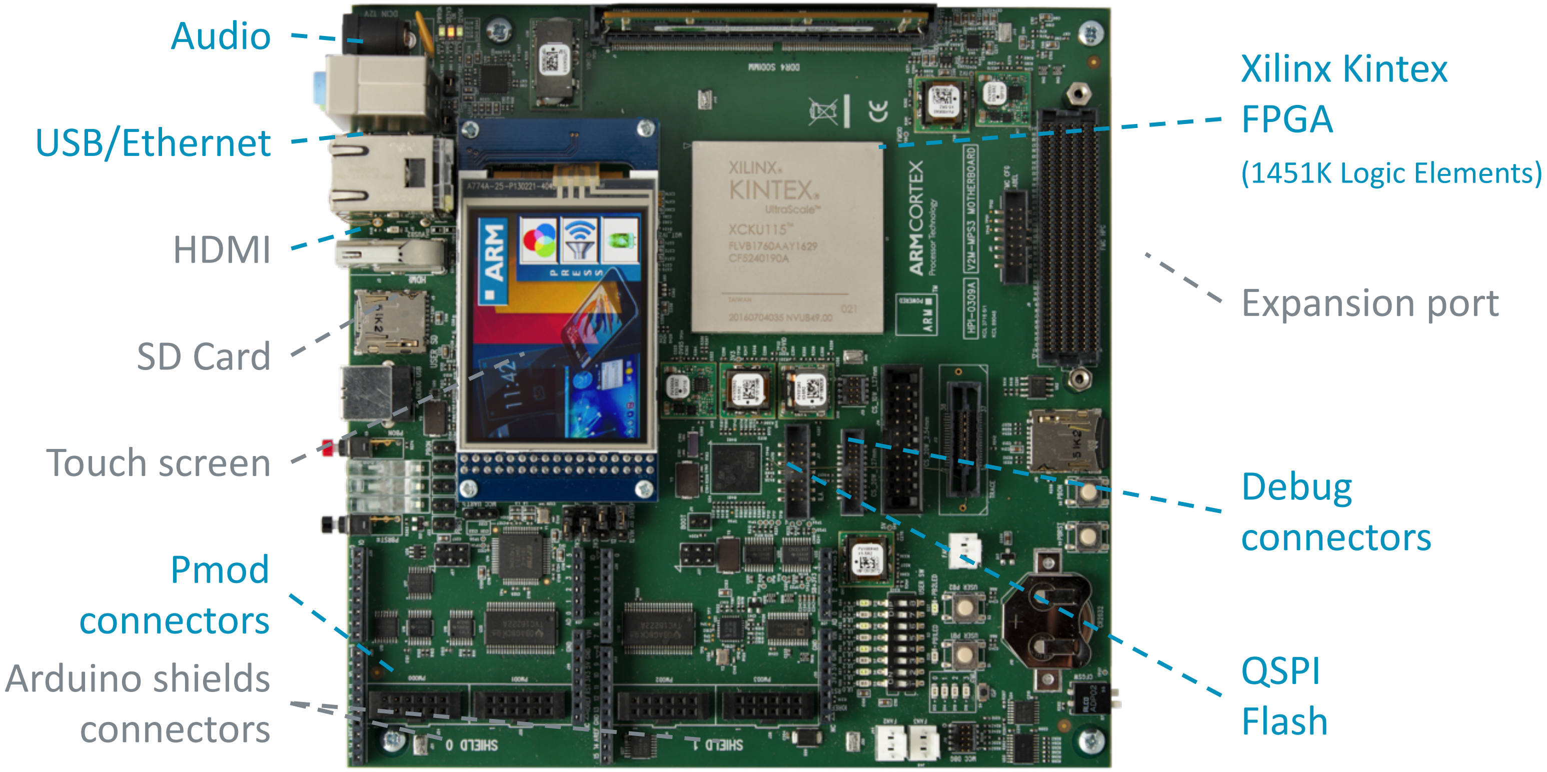


Status – IP Available through AAA

Full Secure Engine TF-M & Application Linux stacks Booting on MPS3 FPGA Dev Board

Platform also avaiallbe with accelerator integrated on HAPS FPGA system

- Full System compatible with current AISS SoC Architecture for AP Host <> Secure Element integration
- Application Host system: Cortex-A35 MP1
- F1A Secure enclave
  - Cortex-M0+, Toot of Trust
  - 32KB ROM (implemented as RAM)
  - 128KB RAM
- System memories
  - DDR4 – 2GB QSPI flash – 8MB SRAM – 4MB
- 20MHz System Frequency
- *Compatible Fixed Virtual Prototype Model making progress*
- *Also boots in Simulation for early stage boot verification*



Audio

USB/Ethernet

HDMI

SD Card

Touch screen

Pmod connectors

Arduino shields connectors

Xilinx Kintex FPGA

(1451K Logic Elements)

Expansion port

Debug connectors

QSPI Flash

Legend: Supported peripheral - - - - - Not supported peripheral - - - - -

Current Software Status #2

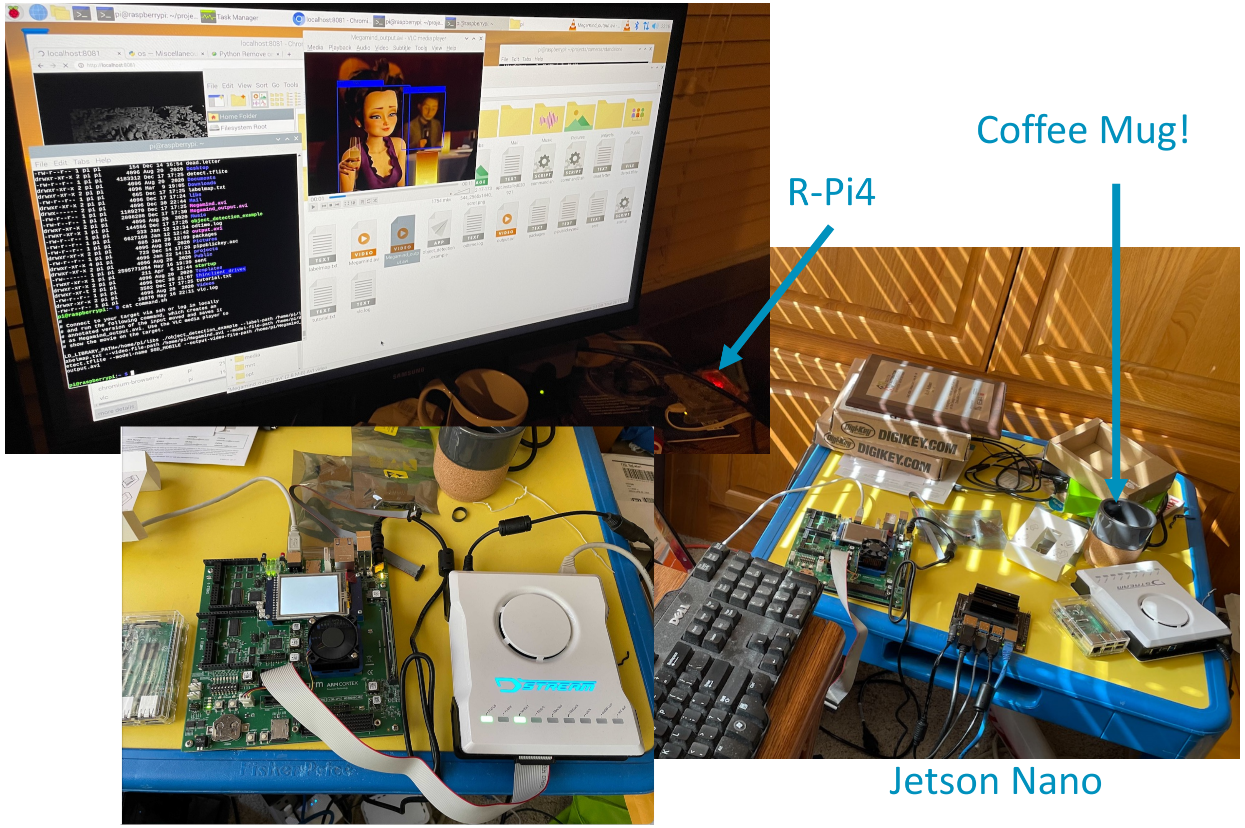
Prototype Object Detection App Running on R-Pi 4 Linux Board

Application has high application performance dynamic range to exercise accelerator options

- AISS representative ML/AI App
  - Object Detection built on ArmNN library
    - Plus a load of other libraries
  - Can be Mapped to MP 1-4
  - Can be Mapped to MP + GPU/NPU
  - Automated Application build world

```
while display.IsStreaming():
        img = camera.Capture()
        detections = net.Detect(img)
        display.Render(img)
        display.SetStatus("Object Detection | Network {:.0f} FPS".format(net.GetNetworkFPS()))
```

‘Complete’ Application Code



Strategic Projects Program. Research into How to build SoCs

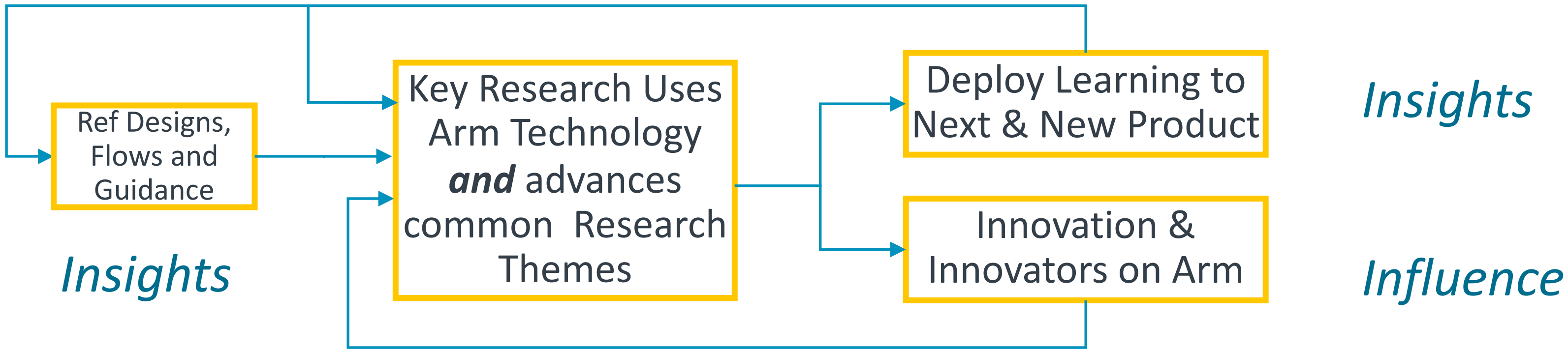
Friction-Free deployment of secure, performance/power optimized Arm devices.

Align and engage funded programs, collaborative consortia including ecosystem partners

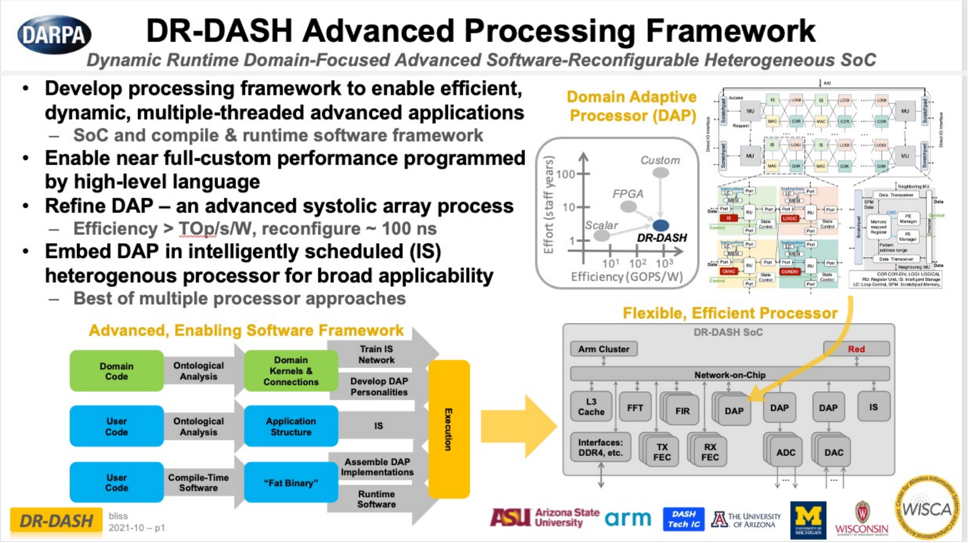
Experienced team across technology stack, transistors through application software

Business development to bring aligned funding to academic/ecosystem consortia

Research Themes	Current Key Projects, Activities & Initiatives
SoC HW & SW Workflow Automation for Better Results at Reduced Design NRE	AISS program key driver (2019 onward – phase 1 complete, phase 2 through 2023) Other DARPA/academic engagements OpenRoad eFabless
Secure & Safe Heterogenous SoC System Architectures	JPL/NASA Space Flight program key driver (design study 2021, SoC to be awarded 2022) Extensible HW SW Platforms for Secure Autonomous devices (AISS/DARPA/academia)
Co-Design with Cloud hosted Device Development & Deployment	Zero Trust Design Manufacture and Deployment Cloud hosted IP Selection Integration and Implementation for Arm SoCs (eFabless AISS OpenROAD)


Ref Designs, Flows and Guidance
Key Research Uses Arm Technology *and* advances common Research Themes
Deploy Learning to Next & New Product
Innovation & Innovators on Arm
Insights
Insights
Influence

Strategic Projects – Collaborative Activities

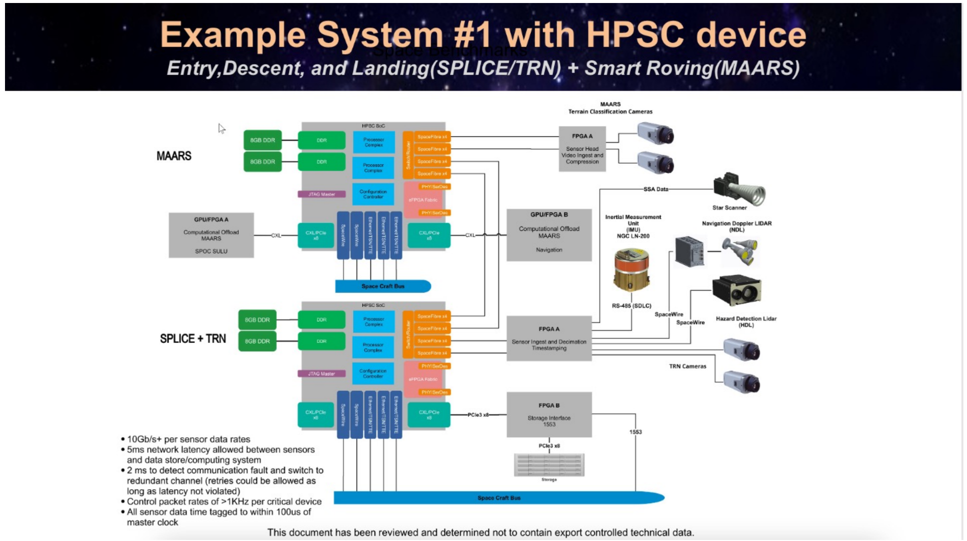


DR-DASH Advanced Processing Framework

Dynamic Runtime Domain-Focused Advanced Software-Reconfigurable Heterogeneous SoC

- Develop processing framework to enable efficient, dynamic, multiple-threaded advanced applications
  - SoC and compile & runtime software framework
- Enable near full-custom performance programmed by high-level language
- Refine DAP – an advanced systolic array process
  - Efficiency > TOp/s/W, reconfigure ~ 100 ns
- Embed DAP in intelligently scheduled (IS) heterogenous processor for broad applicability
  - Best of multiple processor approaches

bliss
2021-10 – p1



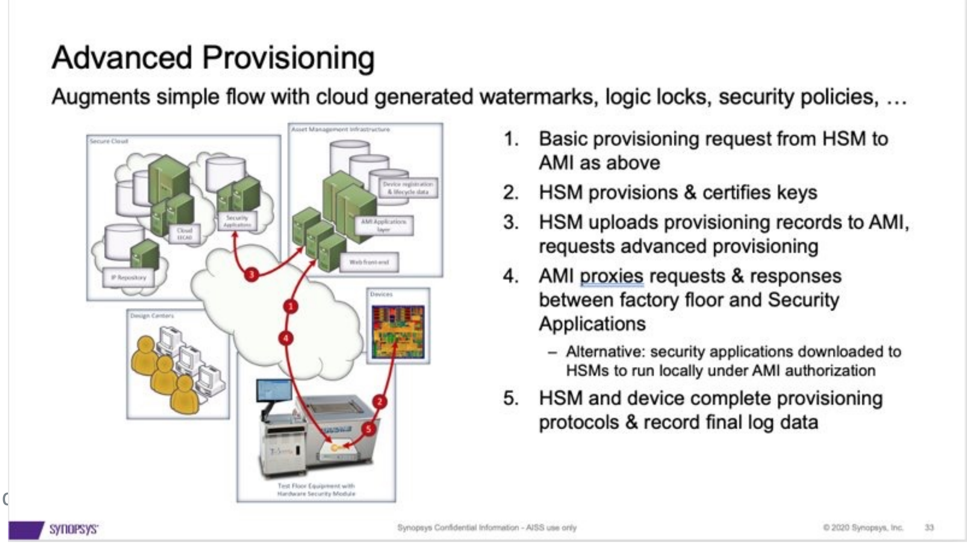
Example System #1 with HPSC device

Entry,Descent, and Landing(SPLICE/TRN) + Smart Roving(MAARS)

MAARS
SPLICE + TRN
HPSC SoC
GPU/FPGA A
Computational Offload MAARS
GPU/FPGA B
Computational Offload MAARS
Navigation
FPGA A
Sensor Head Video Ingest and Compression
MAARS Terrain Classification Cameras
SSA Data
Star Scanner
Inertial Measurement Unit (IMU) NGC LN-200
Navigation Doppler LIDAR (NDL)
Hazard Detection Lidar (HDL)
FPGA A
Sensor Ingest and Decimation Timestamping
TRN Cameras
FPGA B
Storage Interface 1553
PCIe3 x8
1553
Storage
Space Craft Bus
JTAG Master
Configuration Controller
eFPGA Fabric
PHY/SerDes
CXL/PCIe x8
SpaceWire
Ethernet/TSN/TTE

- 10Gb/s+ per sensor data rates
- 5ms network latency allowed between sensors and data store/computing system
- 2 ms to detect communication fault and switch to redundant channel (retries could be allowed as long as latency not violated)
- Control packet rates of >1KHz per critical device
- All sensor data time tagged to within 100us of master clock

This document has been reviewed and determined not to contain export controlled technical data.



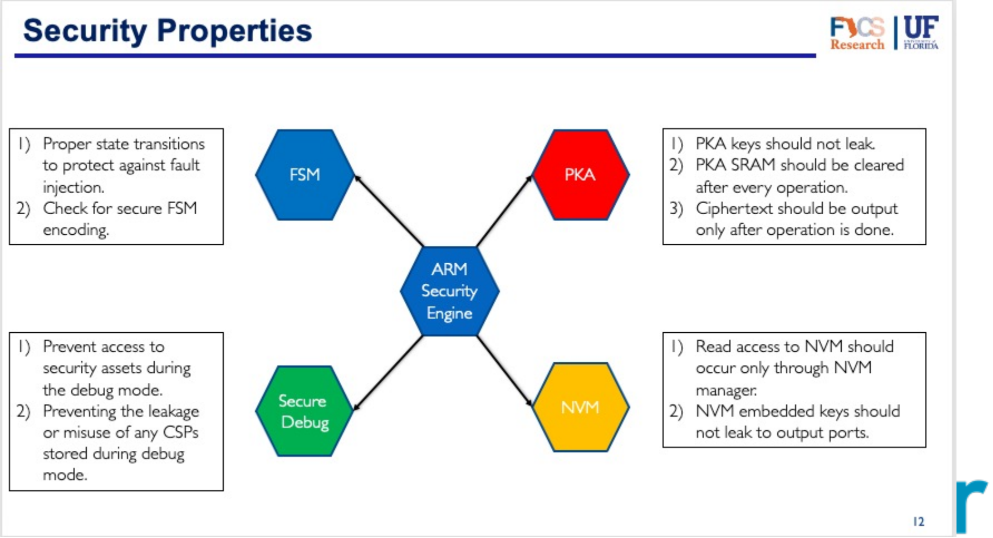
Advanced Provisioning

Augments simple flow with cloud generated watermarks, logic locks, security policies, …

1. Basic provisioning request from HSM to AMI as above
2. HSM provisions & certifies keys
3. HSM uploads provisioning records to AMI, requests advanced provisioning
4. AMI proxies requests & responses between factory floor and Security Applications
   - Alternative: security applications downloaded to HSMs to run locally under AMI authorization
5. HSM and device complete provisioning protocols & record final log data

Secure Cloud
Asset Management Infrastructure
Device registration & lifecycle data
AMI Applications layer
Web front-end
Security Applications
Cloud HSM
IP Repository
Design Centers
Devices
Test Floor Equipment with Hardware Security Module
1
2
3
4
5

SYNOPSYS
Synopsys Confidential Information - AISS use only
© 2020 Synopsys, Inc.
33



Security Properties

1) Proper state transitions to protect against fault injection.
2) Check for secure FSM encoding.

1) PKA keys should not leak.
2) PKA SRAM should be cleared after every operation.
3) Ciphertext should be output only after operation is done.

1) Prevent access to security assets during the debug mode.
2) Preventing the leakage or misuse of any CSPs stored during debug mode.

1) Read access to NVM should occur only through NVM manager.
2) NVM embedded keys should not leak to output ports.

12

Collaboration Opportunities

Take the Platform, Use the Platform Evolve the Platform & Tooling

Operate at the AP or Secure SCP level combine both

Use

- *Bolt an interesting accelerators on*
- *Integrate a new security capability*
- *Look at performance*
- *Fault tolerance*
- *Bake FPGA or Si & Deploy into a Service context*

Evolve

- *Extend the Design Patterns*
- *Change the application platform topology*

- *Power Modeling*
- *Tooling to Render unquified RTL*
- *Pre Silicon assurance tooling*

Thank You

Danke

Gracias

Grazie

谢谢

ありがとう

Asante

Merci

감사합니다

धन्यवाद

Kiitos

شكرًا

ধন্যবাদ

תודה