

# **ARMv8: The Next Generation**

**Minlin Fan & Zenon Xiu  
December 8, 2015**

# Introducing Ourselves

---



**Minlin Fan**  
**Application Engineering Manager**



**Zenon Xiu**  
**Application Engineering Software  
Team Lead**

# ARM Partner Enablement

## ARM Partner Enablement

Helping you through every phase of your design project

### Technical Training

- Expand employees' knowledge and capabilities
- Reduce time to market
- Fully exploit product features

### Support and Maintenance

- Worldwide support
- Design efficiently and effectively
- Increase productivity



### Documentation

- Accessible product information
- Comprehensive and up-to-date
- Supplementary online resources

### Active Assist

- End-to-end project assistance
- Optimize subsystem designs
- Reduce risk

# Where to Find ARM Documentation

- ARM's documentation can be found at <http://infocenter.arm.com/>

The screenshot shows the ARM Infocenter website. At the top, the ARM logo is followed by the tagline "The Architecture for the Digital World®". Navigation tabs include Products, Support, Community, Markets, About, and Careers. A search bar is located in the top right corner. Below the navigation, a breadcrumb trail indicates "You are here: Home > Support > Documentation". A secondary search bar is present, with a dropdown menu set to "All documents" and a search button labeled "Advanced Search". The main content area is divided into two columns. The left column, titled "Contents", lists various document categories such as "Using this site", "ARM Glossary", "ARM Technical Support Knowledge Articles", "ARM architecture", "ARM Software development tools", "Keil embedded development tools", "Development boards", "Developer Guides and Articles", "Application Notes and Tutorials", "Research Papers", "Cortex-A series processors", "Cortex-R series processors", "Cortex-M series processors", "ARM11 processors", "ARM9 processors", "ARM7 processors", "AMBA", "CoreLink controllers and peripherals", and "CoreSight on-chip trace and debug". The right column, titled "Using this site", features a sub-header "ARM Infocenter" and a welcome message: "Welcome to the ARM Infocenter. The Infocenter contains all ARM non-confidential<sup>†</sup> Technical Publications, including:". Below this, a bulleted list of document types is provided: "ARM Architecture Reference Manuals", "Cortex-A, Cortex-R, Cortex-M, ARM11, ARM9, and ARM7 Technical Reference Manuals", "AMBA specifications and design tools and CoreLink peripherals and controllers product manuals", "CoreSight on-chip debug and trace TRMs and Architecture documentation", "ARM Software Development tools and Modeling tools documentation", and "Application Notes and Technical Support Knowledge Articles (FAQs)". Three expandable sections are listed: "ARM Forums and knowledge articles", "Frequently asked questions", and "Giving feedback". On the far right, a "Related information" sidebar contains two promotional boxes. The first is for "mali™", with the text "Visit the Mali Developer Center for more information on the ARM Mali graphics processors. Find out more »". The second is for "KEIL™", with the text "See the Keil website for more information on the Keil range of embedded software development tools. Find out more »". At the bottom of the sidebar, it mentions the "ARM DesignStart program" and "The ARM Physical IP DesignStart program."

- Useful sections:**

- ARM architecture – ARM and GIC architecture reference manuals
- ARM Technical Support Knowledge Articles – FAQs
- Cortex<sup>®</sup>-A/R/M series processors – Technical Reference Manuals
- Developer Guides and Articles – Detailed discussions of TrustZone<sup>®</sup>, barriers...

# Global ARM Support



# Active Assist



# ARM Training

**ARM** The Architecture for the Digital World®

Contact ARM | English | [Login](#) | [Register](#) | [Help](#)

[Products](#) | [Support](#) | [Community](#) | [Markets](#) | [About](#) | [Careers](#)

Search our site

You are here: [Home](#) > [Support](#) > [Training](#)

**Support**

- ARM Self-Service
- Training**
  - ARM Training Courses
  - Training Partner Courses
- Support and Maintenance
- Active Assist
- ARM Accredited Engineer Program
- University Program
- Contact Support

## Training

Expand your employees' knowledge and capabilities so you can reduce time-to-market.

### Why ARM Training?



ARM provides training on a wide range of ARM technology topics, written and delivered by the world's most experienced ARM technology trainers. With public, private and live remote course options available, our courses are flexible too. [Features and benefits of ARM Training...](#)

**Book ARM training courses**

[Public Courses](#) [Private Courses](#) [Remote Courses](#)

We also offer an e-learning course on [ARM Architecture Fundamentals](#).

#### Testimonials

"It was good to communicate with well-qualified specialists who can answer real questions quickly and effectively."

Vadim Balashov - Milandr

# Training Options From ARM



## Private courses

- Onsite, anywhere delivery
- Face-to-face with ARM experts
- Customized agendas



## Public courses

- ARM hosted
- Public schedule
- Open enrolment



## Live remote courses

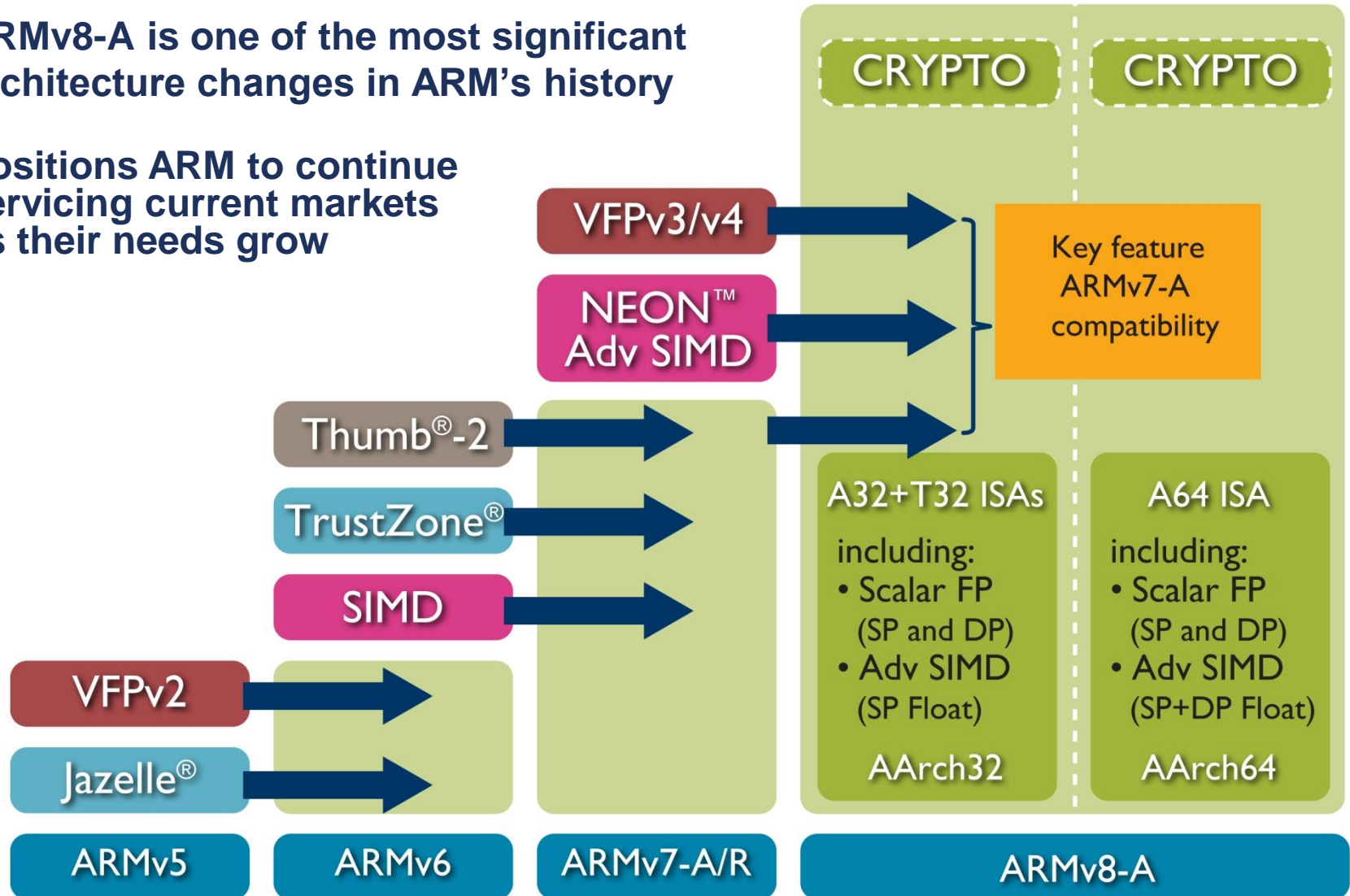
- Live broadcast from ARM
- Distributed teams
- Targeted, agile delivery



# ARMv8-A Overview

# Development of the ARM Architecture

- ARMv8-A is one of the most significant architecture changes in ARM's history
- Positions ARM to continue servicing current markets as their needs grow



# What's New in ARMv8-A?

---

- **ARMv8-A introduces two execution states: AArch32 and AArch64**
- **AArch32**
  - Evolution of ARMv7-A
  - A32 (ARM) and T32 (Thumb) instruction sets
    - ARMv8-A adds some new instructions
  - Traditional ARM exception model
  - Virtual addresses stored in 32-bit registers
- **AArch64**
  - New 64-bit general purpose registers (X0 to X30)
  - New instructions – A64, fixed length 32-bit instruction set
    - Includes SIMD, floating point and crypto instructions
  - New exception model
  - Virtual addresses now stored in 64-bit registers

# Agenda

---

Architecture versions

- **Privilege levels**

AArch64 Registers and A64 Instruction Set

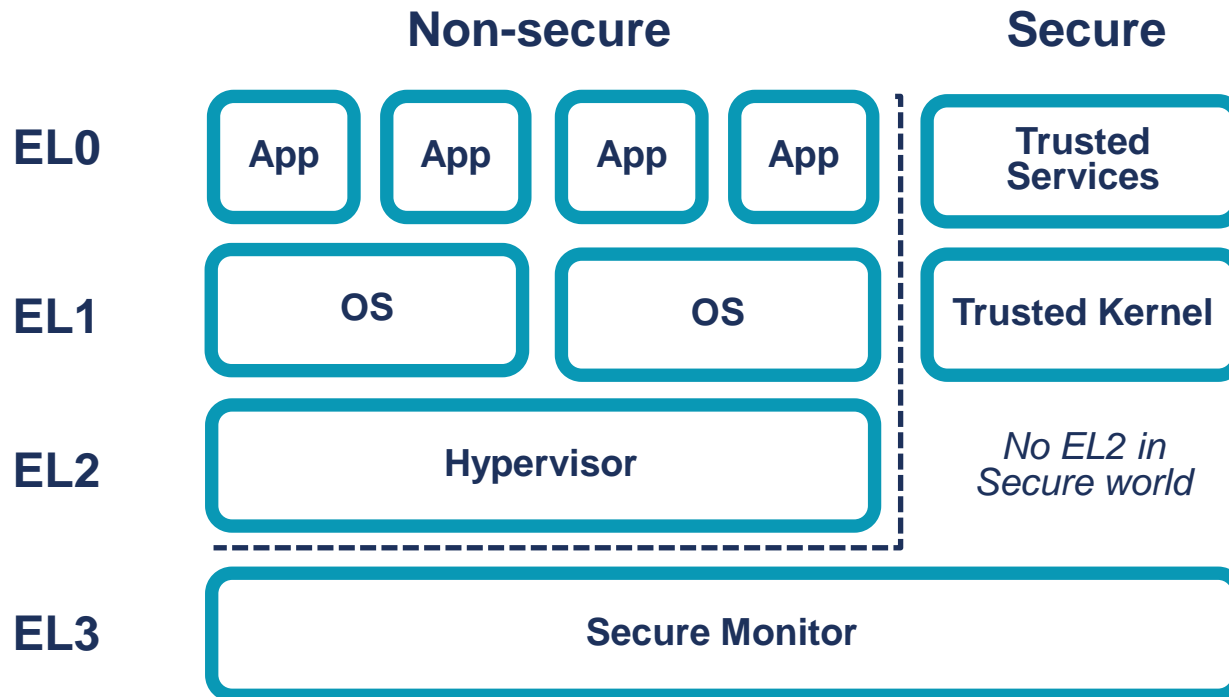
AArch64 Exception Model

AArch64 Memory Model

ARMv8.1

# AArch64 Privilege Model

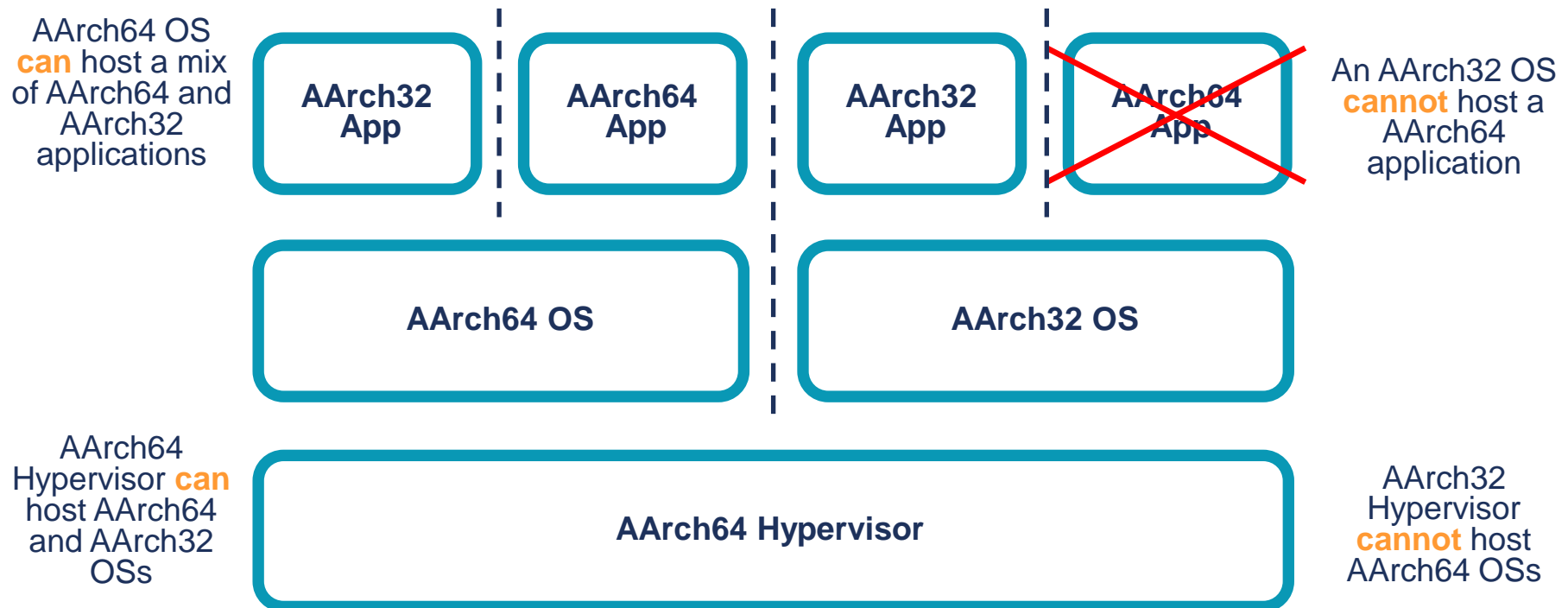
- AArch64 has four exception levels, and two security states
  - EL0 = least privileged, EL3 = most privileged
  - Secure state and non-secure (or Normal) state



- **EL2 and EL3 are optional**
  - A processor may not implement EL2/3 if Security or Virtualization are not required

# Moving Between AArch32 & AArch64

- Execution state can *only* change on exception entry or return
  - Moving to a lower EL, execution state can stay the same *or switch to AArch32*
  - Moving to a higher EL, execution state can stay the same *or switch to AArch64*



# Agenda

---

Architecture versions

Privilege levels

- **AArch64 registers and the A64 Instruction Set**

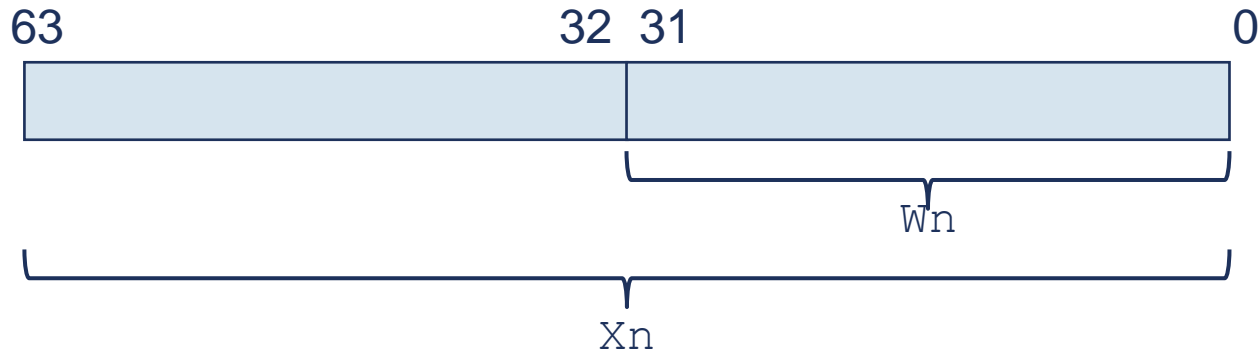
AArch64 Exception Model

AArch64 Memory Model

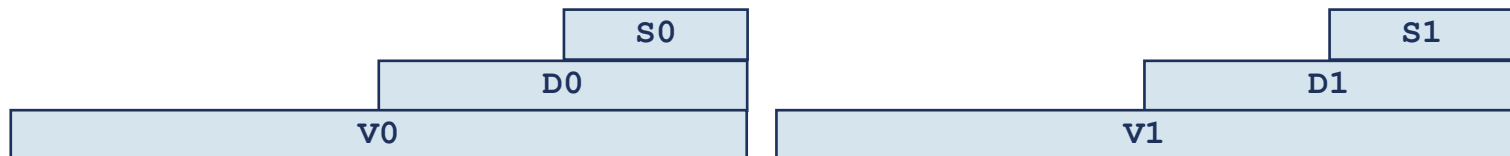
ARMv8.1

# Register Banks

- AArch64 provides 31 general purpose registers
  - Each register has a 32-bit ( $w0-w30$ ) and 64-bit ( $x0-x30$ ) form



- Separate register file for floating point, SIMD and crypto operations -  $v_n$ 
  - 32 registers, each 128-bits
    - Can also be accessed in 32-bit ( $s_n$ ) or 64-bit ( $d_n$ ) forms





# AArch64 ↔ AArch32 Register Mappings

x0-x7	x8-x15	x16-x23	x24-x30
R0	R8_usr	R14_irq	R8_fiq
R1	R9_usr	R13_irq	R9_fiq
R2	R10_usr	R14_svc	R10_fiq
R3	R11_usr	R13_svc	R11_fiq
R4	R12_usr	R14_abt	R12_fiq
R5	R13_usr	R13_abt	R13_fiq
R6	R14_usr	R14_und	R14_fiq
R7	R13_hyp	R13_und	

- **When moving from AArch32 to AArch64**
  - Registers accessible in both registers widths
    - Top 32 bits: UNKNOWN
    - Bottom 32 bits: The value of the AArch32 register
  - Registers that are not accessible in AArch32 retain value from previous AArch64 execution

# A64

- **AArch64 introduces new A64 instruction set**
  - Similar set of functionality as traditional A32 (ARM) and T32 (Thumb) ISAs
- **Fixed length 32-bit instructions**
- **Syntax similar to A32 and T32**

ADD W0, W1, W2                      ← w0 = w1 + w2 (32-bit addition)

ADD X0, X1, X2                      ← x0 = x1 + x2 (64-bit addition)

- **Most instructions are not conditional**
- **Optional floating point and Advanced SIMD instructions**
- **Optional cryptographic extensions**
  - Low level acceleration for AES, SHA1, SHA224 and 256

# Agenda

---

Architecture versions

Privilege levels

AArch64 registers and the A64 Instruction Set

- **AArch64 Exception Model**

AArch64 Memory Model

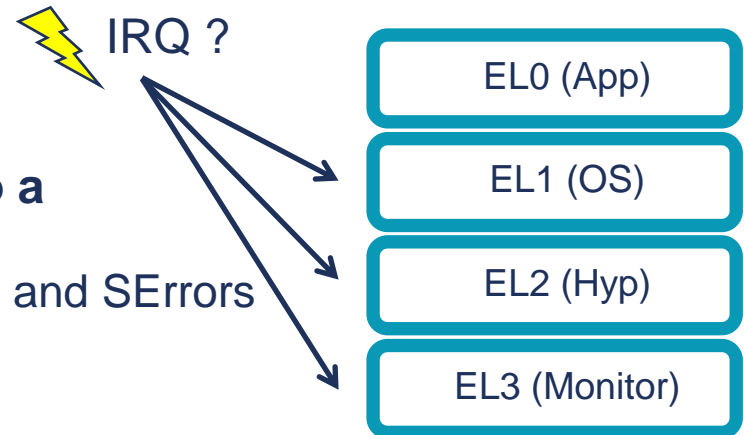
ARMv8.1

# AArch64 Exceptions

- In AArch64 exceptions are split between:
  - Synchronous
    - Data aborts from MMU, permission/alignment failures, service call instructions, etc.
  - Asynchronous
    - IRQ/FIQ
    - SError (System Error)

- On taking an exception the EL can stay the same OR get higher
  - Exceptions are never taken to EL0

- Asynchronous exceptions can be routed to a specific EL
  - Separate bits to control routing of IRQs, FIQs and SErrors



# Agenda

---

Architecture versions

Privilege levels

AArch64 registers and the A64 Instruction Set

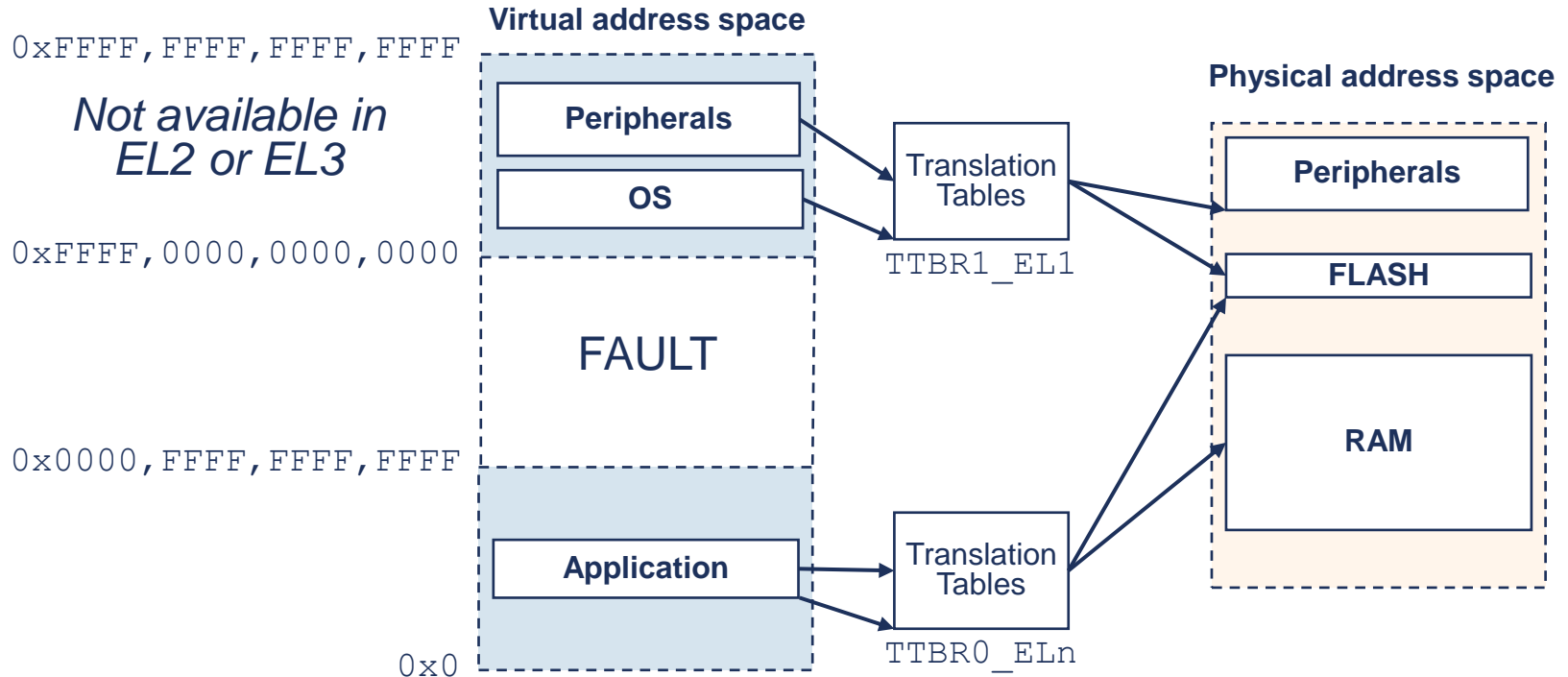
AArch64 Exception Model

- **AArch64 Memory Model**

ARMv8.1

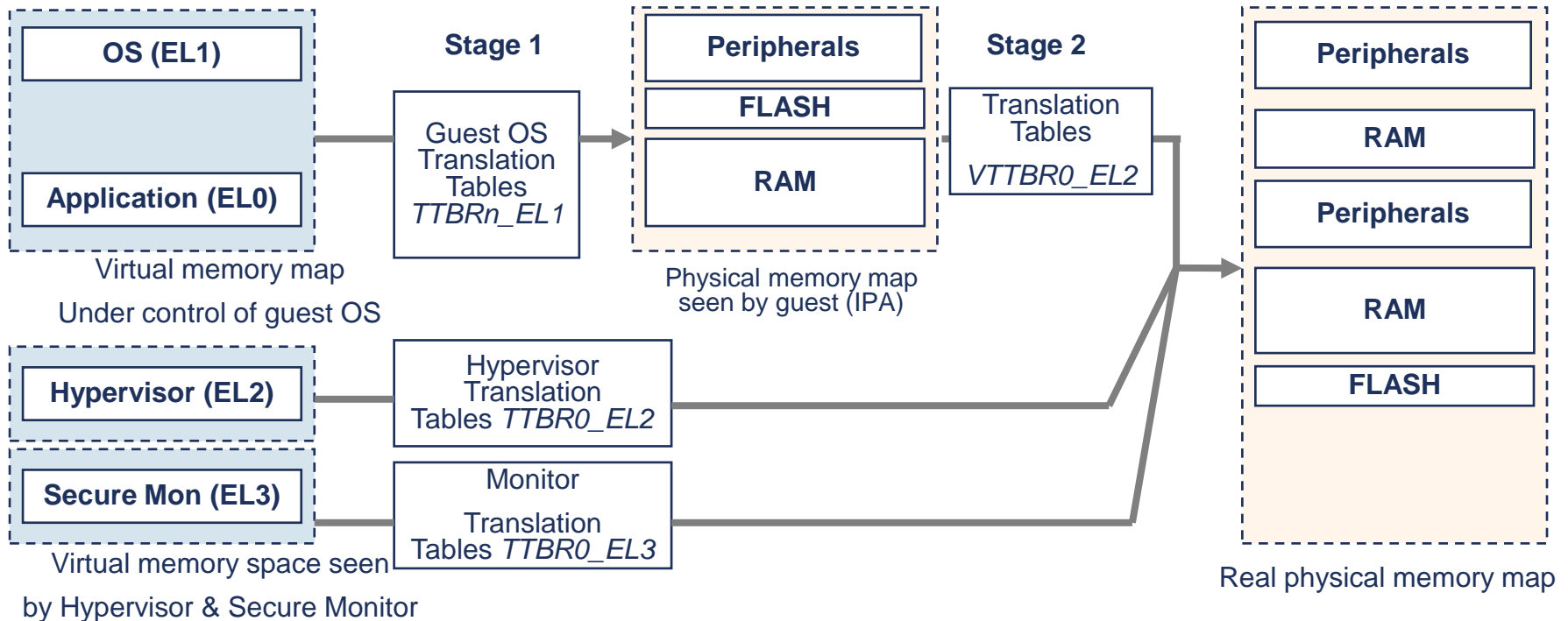
# Virtual Address Space

- Virtual addresses are 64-bit wide, but not all addresses are accessible
  - Virtual memory address space split between two translation tables
    - Each covering a configurable size, up to 48 bits of address space ( $TCR\_ELn$ )
  - Addresses not covered by either translation table generate translation faults



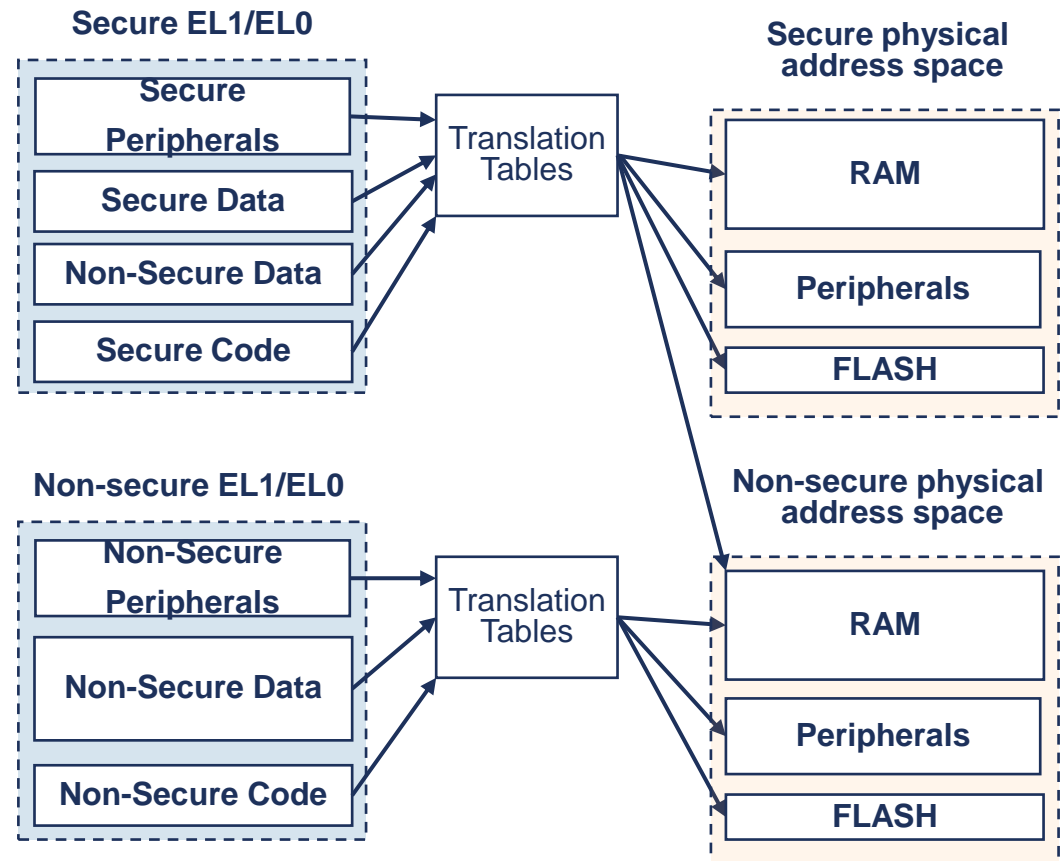
# Multiple Translation Regimes

- The virtualization extensions introduce a *second* stage of translation
  - First stage: Virtual Address (VA) → Intermediate Physical Address (IPA)
    - Operation of MMU appears unchanged for guest OSs, still use `TTBRn_EL1` and `TCR_EL1`
  - Second stage: Intermediate Physical Address (IPA) → Physical Address (PA)
    - Controlled by the Hypervisor
- The Hypervisor and Secure Monitor also have a set of stage 1 Translation Tables
  - Mapping directly from VA to PA



# Physical Address Spaces

- ARMv8-A defines two security states: **Secure** and **Non-secure (Normal)**
  - It also defines two physical address spaces: **Secure** and **Non-secure**
- These are in theory completely separate:
  - SP:0x8000 != NP:0x8000
    - But most systems treat Secure/Non-Secure as an attribute for access control
  - Normal world can only access the non-secure physical address space
  - Secure world can access BOTH physical address spaces
    - Controlled through translation tables





# Agenda

---

Architecture versions

Privilege levels

AArch64 registers and the A64 Instruction Set

AArch64 Exception Model

AArch64 Memory Model

- **ARMv8.1**

# ARMv8.1-A

---

- **The ARM architecture continues to evolve, with the announcement of ARMv8.1-A**
- **Instruction set enhancements**
  - Atomic read-write instructions added to A64
    - For example: Compare and swap
  - Additional SIMD instructions
    - Example use case is colour space conversion
  - Load and stores with ordering limited to a configurable region
- **Virtualization Host Extensions**
  - To improve performance of Type 2 Hypervisors
- **And other enhancements to the memory system architecture, such as Privileged Access Never (PAN) state bit**

# ARMv8-A Overview

# So Much to Say...

---

- **Day 0 – ARMv7-A Architecture**

- The ARM Architecture

- **Day 1**

- Introduction to ARMv8-A
- AArch64 A64 ISA Overview †
- AArch64 Exception Model †
- AArch64 Memory Management
- AArch64 Memory Model
- Caches and Branch Prediction †

- **Day 2**

- Barriers
- Synchronization
- Cache Coherency
- Operating System Support
- SW Engineer's Guide to Cortex-A5x ‡
- Booting

- **Day 3**

- Security
- Virtualization
- Power Management (optional)
- GIC Programming (optional) †
- Debug (optional)

...so little time!

# Where to Go Next?

The screenshot shows the ARM website's Training page. At the top, the ARM logo is followed by the tagline "The Architecture for the Digital World®". Navigation links include "Products", "Support", "Community", "Markets", "About", and "Careers". A search bar is located on the right. The breadcrumb trail reads "You are here: Home > Support > Training". A left-hand sidebar lists support options, with "Training" expanded to show "ARM Training Courses" and "Training Partner Courses". The main content area features a large banner with a man speaking and the text "Expand your employees' knowledge and capabilities so you can reduce time-to-market." Below this is a section titled "Why ARM Training?" which describes the range of training options. A "Book ARM training courses" section includes buttons for "Public Courses", "Private Courses", and "Remote Courses". A testimonial box on the right features a photo of Vadim Balashov and a quote from Milandr.

**ARM** The Architecture for the Digital World®

Contact ARM | English | [Login](#) | [Register](#) | [Help](#)

[Products](#) | [Support](#) | [Community](#) | [Markets](#) | [About](#) | [Careers](#)

You are here: [Home](#) > [Support](#) > [Training](#)

**Support**

- ARM Self-Service
- Training**
  - ARM Training Courses
  - Training Partner Courses
- Support and Maintenance
- Active Assist
- ARM Accredited Engineer Program
- University Program
- Contact Support

## Training

Expand your employees' knowledge and capabilities so you can reduce time-to-market.

### Why ARM Training?


ARM provides training on a wide range of ARM technology topics, written and delivered by the world's most experienced ARM technology trainers. With public, private and live remote course options available, our courses are flexible too. [Features and benefits of ARM Training...](#)

**Book ARM training courses**

[Public Courses](#) [Private Courses](#) [Remote Courses](#)

We also offer an e-learning course on [ARM Architecture Fundamentals](#).

#### Testimonials

 **milandr**

"It was good to communicate with well-qualified specialists who can answer real questions quickly and effectively."

Vadim Balashov - Milandr

**Thank you**

***Any questions?***