Arm-Cortex-A-Software Development

Summary
Arm Cortex-A software training courses are designed to help engineers working on new or existing Cortex-A system designs. Whether you’re working on design, verification or validation, for a Cortex-A system, the course can be configured according to your team’s needs.

Courses include fundamental topics to enable a solid platform of understanding. The rest of the course then builds on from this with optional topics and can be tailored appropriately. Some key topics are delivered via pre course on-demand video.

Learning activities such as interactive workbooks, walkthrough examples and quizzes are incorporated into the training to help bring the learning to life.

A pre course call with the engineer delivering the training will help you discuss your team’s individual training requirements.

At the end of the course delegates will be able to:

- Describe different Cortex-A processor features and their use
- Understand the programmer’s model of a Cortex-A processor
- Identify and solve key Cortex-A system design issues
- Program simple bare-metal code in both C and Arm assembly language using an AWS test environment
- Debug issues on Cortex-A processors

<table>
<thead>
<tr>
<th>Course Length</th>
<th>Delivery Method</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>2 – 4 days</td>
<td>Classroom</td>
<td>Virtual or Onsite</td>
</tr>
</tbody>
</table>

Audience

- System architects
- Real-time operating system developers
- Device driver developers
- Low level software developers
- Engineers writing low level test code

Prerequisites

- A basic understanding of microprocessor systems
- Familiarity with assembler or C programming
- Experience of embedded system development is helpful but not essential
- A basic awareness of Arm is an advantage but not required

Related Products
Topics

Agendas will be created from the following list of fundamental and optional topics

### Fundamental Topics

- **Introduction to the Arm Architecture** and feature set of your chosen Cortex-A processor. A discussion of the programmer’s model, register layout and architectural features
- **Assembly programming**. Introducing the instruction set and assembly directive available and how to use them
- A discussion of the interrupt and exception architecture, including how to handle IRQs and internal faults and how to program the interrupt controller
- **Cortex-A memory model**, covering Arm memory types, interactions with caches and how to program the Memory Management Unit (MMU)
- An advanced discussion of Arm memory accesses, including memory barrier instructions and Load/Store Exclusive instructions for inter-process synchronisation
- Effective use of compilation tools with a Cortex-A system. Covering writing effective and efficient C code and basic linker layout. Including the bare metal software boot flow from reset to C main() via an hands on AWS based session
- **Embedded virtualization**. Interrupt virtualization, instruction trap-and-emulate and the two stage MPU

### Optional Additional Topics

Optional Topics can be used to tailor the training to meet your specific, mobile, infrastructure or architectural design.

- **Arm DynamIQ** technology
- Scalable Vector Extension, SVE2,
- Transactional Memory Extension (TME)
- Memory Tagging Extension (MTE)
- **NEON overview** covering the Single Instruction Multiple Data (SIMD) instructions available for Cortex-A processors
- **Debug**. A discussion of the Cortex-R debug architecture focusing on the low-level feature that enable a debugger to connect to and debug your CPU.
- Using the Arm **Generic Interrupt Controller** (GIC) in your Cortex-A system
- Security using TrustZone in your Cortex Processor based system
- Auto enhanced (AE)
- OS Support
- Software Test Libraries (Armv8-A processors only)
- Power management for Cortex-A

### Related face-to-face and on-demand courses

- Introduction to Armv7-A
Example Training Courses

Examples of how the above topics can be combined into differing course options are given below

Arm A Profile Architecture - New Features – (2 Days)

- An overview of the fundamental features of the AArch64 architecture
- Improved features that have been introduced and their effect on the on the virtualization behavior and the memory model.
- A discussion of the new security features
- An overview of the changes to the debug architecture
- Transactional Memory
- Scalable Vector Extension (SVE and SVE 2)

Arm Cortex A72 Software Development – (3 Days)

- Cortex-A72 Processor Overview
- Introduction to Armv8-A
- AArch64 A64 ISA Overview
- A64 ISA Workbook
- AArch64 Exception Model
- Armv8 Exception Model Workbook
- Armv8-A Memory Management
- Armv8-A Memory Model
- Caches and Branch Prediction
- MMU and Cache Initialization Workbook
- Barriers
- Synchronization
- Cache Coherency
- OS Support
- Software Engineer Guide to the Cortex-A72
- Booting
- Power management for Cortex-A
- Virtualization
- Security
- GIC Programming
Arm DynamIQ Software for Cortex A77, A76, A75, A55, - (3 Days)

- Introduction to Arm
- Armv8-A Overview
- Introduction to DynamIQ
- A64 ISA Overview
- Armv8-A AArch64 Exception Model
- Memory Management
- Memory Model
- Caches and Branch Prediction
- Barriers
- Synchronization
- Cache Coherency
- RAS support
- Software Engineer Guide to the DynamIQ Shared Unit (DSU)
- Software Engineer Guide to DynamIQ CPU(s)
- Booting
- Power Management
- Virtualization
- Security
- Software Engineer’s Guide to System Fabric (Optional)