

ARM ATC Software Courses

ARM Architecture Fundamentals

Course Duration: 1 day

Summary:

This course is designed to provide hardware and software engineers new to ARM processors with the essential basic information that they will need to be aware of to start ARM-based designs. It can also be useful to non-technical staff who require a basic understanding of ARM technology.

Prerequisites:

- A basic understanding of microprocessors or microcontrollers.
- A basic understanding of digital logic or ASIC design issues would be useful, but not essential.
- A basic understanding of assembly / C programming would be useful, but not essential.
- No prior knowledge of the ARM processor is assumed.

Modules:

- Introduction to ARM
- Programmers model
- ARM and Thumb Instruction Sets
- ARM Cores Overview (ARM7TDMI, ARM9TDMI, ARM9E-S, ARM10, ARM11)
- Advanced Microcontroller Bus Architecture (AMBA)
- ARM Software Development Tools
- Exception Handling
- Embedded ARM software issues

ARM Cortex-A5 Software Development

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This course is designed for Software engineers developing for ARM's Architecture V7 with a specific focus on the features of the Cortex-A5.

Prerequisites:

- Some knowledge of coding for embedded systems
- Some experience using C/C++

Audience:

Software engineers looking to gain a good understanding of the features of the Cortex-A5, from Reset through to Application development.

Modules:

- Introduction to the ARM Architecture
- ARM Tools Overview
- RVDS Introductory Workbook
- Assembler Programming for ARM Processors
- ARM Assembler Workbook
- Exception Handling
- Exception Handling Workbook
- ARM Caches and TCMs
- Memory Management
- Synchronization Support
- C/C++ Compiler Hints & Tips
- Linker & Libraries Hints & Tips
- Compilation Tools Workbook Sections 1-4
- Software Engineers' Guide to the Cortex-A5
- Neon Overview
- Further Compiler/Linker Hints & Tips
- Compilation Tools Workbook Sections 5-6
- Introduction to TrustZone
- Embedded Software Development
- Scatter Loading Workbook
- ARM Debug and Trace
- CoreSight Overview

ARM Cortex-A8 Software Development

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This course is designed for software engineers designing applications for platforms based around the ARM Cortex-A8 processor core.

Prerequisites:

- Some knowledge of embedded systems
- Familiarity embedded programming in C and assembler
- A basic awareness of ARM is useful but not essential

Audience:

Software design engineers who need to understand the issues involved when developing application software for the ARM Cortex-A8 processor core.

Modules:

- Introduction to the ARM Architecture
- ARM Tools Overview
- RVDS Introductory Workbook
- Assembler Programming for ARM Processors
- ARM Assembler Workbook
- Exception Handling
- Exception Handling Workbook
- ARM Caches and TCMs
- Memory Management
- Synchronization Support
- C/C++ Compiler Hints and Tips
- Linker & Libraries Hints and Tips
- Compilation Tools Workbook Sections 1-4
- Software Engineers' Guide to the Cortex-A8
- Neon Overview
- Power Management for Cortex-A/R Cores
- Further Compiler/Linker Hints and Tips
- Compilation Tools Workbook Sections 5-6
- Introduction to TrustZone
- Embedded Software Development
- Scatter Loading Workbook
- ARM Debug and Trace
- CoreSight Overview

ARM Cortex-A9 MPCore Software Training

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This training course covers the issues involved in developing software for platforms powered by the ARM Cortex-A9 MPCore application processors.

Prerequisites:

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

Audience:

Much of the content is relevant to users of 3rd party tools but we cannot undertake to cover them in any detail.

Modules:

- Introduction to the ARM Architecture
- ARM Tools Overview
- RVDS Introductory Workbook
- Assembler Programming for ARM Processors
- ARM Assembler Workbook
- Exception Handling
- Exception Handling Workbook
- ARM Caches and TCMs
- Memory Management
- Synchronization Support
- C/C++ Compiler Hints & Tips
- Linker & Libraries Hints & Tips
- Compilation Tools Workbook Sections 1-4
- Introduction to TrustZone
- Software Engineers' Guide to the Cortex-A9
- Neon Overview
- Power Management for Cortex-A/R Cores
- Further Compiler/Linker Hints & Tips
- Compilation Tools Workbook Sections 5-6
- Embedded Software Development
- Scatter Loading Workbook

- Software Engineers' Guide to MPCore Processors
- Implementing a simple SMP micro-kernel for MPCore
- ARM Debug and Trace
- CoreSight Overview

ARM Cortex-A9 Software Development

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This training course covers the issues involved in developing software for platforms powered by the ARM Cortex-A9 application processors.

Prerequisites:

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

Audience:

This course concentrates on the use of ARM's RealView Developer Suite (RVDS). Much of the content is relevant to users of 3rd party tools but we cannot undertake to cover them in any detail.

Modules:

- Introduction to the ARM Architecture
- ARM Tools Overview
- RVDS Introductory Workbook
- Assembler Programming for ARM Processors
- ARM Assembler Workbook
- Exception Handling
- Exception Handling Workbook
- ARM Caches and TCMs
- Memory Management
- Synchronization Support
- C/C++ Compiler Hints & Tips
- Linker & Libraries Hints & Tips
- Compilation Tools Workbook Sections 1-4
- Software Engineers' Guide to the Cortex-A9
- Neon Overview

- Power Management for Cortex-A/R Cores
- Further Compiler/Linker Hints & Tips
- Compilation Tools Workbook Sections 5-6
- Introduction to TrustZone
- Embedded Software Development
- Scatter Loading Workbook
- ARM Debug and Trace
- CoreSight Overview

ARM Cortex-M3/M4 Software Design

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This course is designed for hardware engineers and software engineers developing software for platforms based around the ARM Cortex-M3 and Cortex-M4 cores. Including an introduction to the ARM product range and supporting IP, the course covers the ARM core range, programmer's model and instruction set architecture as well as the Cortex-M3/M4 debug architecture is also covered. The course includes a number of worked examples and hands-on practical exercises to reinforce the lecture material.

Prerequisites:

- Some knowledge of embedded systems
- A basic awareness of ARM is useful but not essential
- Knowledge of programming in C
- Experience of assembler programming is not required but would be beneficial

Audience:

Software engineers writing application and system software for platforms using the ARM Cortex-M3 processor core.

Modules:

- Cortex-M3 Overview
- RealView Developer Suite Overview (and workbook)
- Cortex-M3 Processor Core
- Instruction Set
- Migrating Legacy Code to Cortex-M3
- Cortex-M3 Interrupts
- Cortex-M3 Exception Handling

- Memory Types
- Memory Protection Unit
- Embedded Software Development (and workbook)
- CoreSight Debug Architecture Overview
- Cortex-M3 Debug Infrastructure
- Cortex-M3 Invasive Debug
- Cortex-M3 Non-Invasive Debug (Trace)
- Compiler Hints and Tips

ARM Cortex-R4 Software Development

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This training course covers the issues involved in developing software for platforms powered by the ARM Cortex-R4 processor.

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

Audience:

Much of the content is relevant to users of 3rd party tools but we cannot undertake to cover them in any detail.

Modules:

- Introduction to ARM Architecture
- ARM Tools Overview
- RVDS Introductory Workbook
- Assembler Programming for ARM Processors
- ARM Assembly Workbook
- Exception Handling
- Exception Handling Workbook
- ARM Caches and TCMs
- Memory Management
- C/C++ Compiler Hints & Tips

- Linker & Library Hints and Tips
- Compilation Tools Workbook Sections 1-4
- Software Benchmarking and Profiling
- Software Engineer's Guide to Cortex-R4
- Further Compiler/Linker Hints and Tips
- Compilation Tools Workbook Sections 5-6
- Embedded Software Development
- Power Management for Cortex-A/R Cores
- Scatter loading Workbook
- ARM Debug & Trace
- CoreSight Overview

ARM Embedded Software Optimization

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

A new 3 day embedded software optimisation course focused on ARM Application class processors. The course uses ARM's RealView Development Suite (RVDS) tool chain and covers the following main areas:

- Use of ARM code generation and profiling tools to get the best results
- Methods and metrics to measure performance and identify bottlenecks
- Examination of System and Hardware issues and their impact on performance

Prerequisites:

- Knowledge and experience of coding in C and Assembler
- Familiarity with the ARM instruction set
- Experience of ARM development tools is not required but would be beneficial

Audience:

Software engineers involved in any development where performance is likely to be a critical issue and who need to learn techniques which will help them get the best out of the tools and platforms available.

Modules:

- Introduction to Xvid & RVDS
- Software Benchmarking Overview
- Using RVCT
- Focus your efforts - Introducing the Profiler
- Compiler and Linker Optimizations
- Architecture Optimizations
- System level optimizations

ARM1136/76 Software Development

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This training course covers the issues involved in developing software for platforms powered by the ARM1136J(F)-S, ARM1156T2(F)-S and ARM1176JZ(F)-S application processors.

Prerequisites:

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

Audience:

Much of the content is relevant to users of 3rd party tools but we cannot undertake to cover them in any detail.

Modules:

- Introduction to ARM Architecture
- ARM Tools Overview
- RVDS Introductory Workbook
- Assembler Programming for ARM Processors
- ARM Assembly Workbook
- Exception Handling
- Exception Handling Workbook
- ARM Caches and TCMs
- Memory Management

- C/C++ Compiler Hints & Tips
- Linker & Library Hints and Tips
- Compilation Tools Workbook Section 1-4
- Software Benchmarking and Profiling
- Software Engineer's Guide to ARM11
- Further Compiler/Linker Hints and Tips
- Compilation Tools Workbook Section 5-6
- Embedded Software Development
- Scatter loading Workbook
- Introduction to TrustZone
- ARM Debug and Trace

ARM7/9 Software Development

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This training course covers the issues involved in developing software for platforms powered by ARM7 or ARM9 processors. The course covers the RealView software development tools, including RealView Profiler. Other subjects include instruction set, programmer's model, exception handling, debug, benchmarking and memory systems.

Optional material on the Operating System Support features of these processors can be included on request.

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

Audience:

This course concentrates on the use of ARM's RealView Developer Suite (RVDS). Much of the content is relevant to users of 3rd party tools but we cannot undertake to cover them in any detail.

Modules:

- The ARM Architecture & Processor Core
- Introduction to ARM Development Tools
- RVDS Introductory Workbook
- Assembler Programming for v4/v5 ARM Cores
- ARM v4/v5 Assembly Workbook
- ARM/Thumb Interworking
- Interworking Workbook
- Exception Handling
- Exception Handling Workbook
- C/C++ Compiler Hints & Tips
- Compilation Tools Workbook
- ARM Debug & Trace
- v4/v5 Caches, TCMs & Memory Management
- Benchmarking and Profiling
- Profiler Workbook
- Embedded Software Development
- Scatter loading Workbook

Modeling with ARM's Fast Model Tools & Portfolio

- **Course Type:** Software
- **Course Length:** 3 day(s)

Summary:

This course is designed for those who are modeling SoCs using System Generator. The course covers generation of Virtual Platforms, writing and debugging component models, using ARM Fast Models, importing C++ models and SystemC export.

Prerequisites:

- Knowledge of embedded system design principles
- Familiarity with embedded software development
- Experience of the C++ programming language

Audience:

Hardware/System Design Engineers who need to model complex system-on-chip designs and Software Engineers who need to develop and simulate applications.

Modules:

- Background to ARM and the Connected Community
- Overview of System Generator and ESL
- ARM's ESL tools, components and licensing
- Creating Virtual Platforms using existing components
- The PV Model Library
- SG modeling technology
- Writing LISA+ models
- PV Bus models
- Modeling hardware with LISA+
- Debugging LISA+ models
- Importing C++ models
- SystemC export

NEON Training Course

- **Course Type:** Software
- **Course Length:** 2 day(s)

Summary:

Hidden inside many of the next generation of Cortex-A8 and Cortex-A9 processors is a sophisticated DSP engine. Fully supported by ARM, GNU and other popular tool chains, this master class will provide a detailed introduction to NEON™ technology, before going on to show how it can be applied in software.

Learn how to take advantage of NEON to optimize common signal processing functions used in Filtering, Sample Rate Conversion, Audio and Video Codec applications.

Modules:

The training modules will cover the following main topics:

Day 1:

- ARM Architecture Overview and Introduction
- Co-processor instruction space and how it relates VFP and NEON
- NEON Introduction & Instruction Set Details
- Why & How to use NEON
- NEON vs DSP dedicated hardware

Day 2

- Compiler and NEON, Working with Intrinsics, Libraries and Vectorising Compiler
- Micro-architecture implementations of NEON in Cortex-A8 and Cortex-A9
- NEON examples, worked examples showing efficient usage of NEON
- How does NEON relate to Debug and Profiling support?

Contact Details:



Aravindan.R
FTD Solutions Pte Ltd.
#159 Kampong Ampat
06-03 KA Place,
Singapore 368328

Tel: (+65) 64882469
Fax: (+65) 6846 1663
Email: info@ftdsolutions.com