Advanced Features and Techniques of Embedded Systems Design

Course Description

Advanced Features and Techniques of Embedded Systems Design provides embedded systems developers the necessary skills to develop complex embedded systems and enables them to improve their designs by using the tools available in the Vivado IP Integrator. This course also helps developers understand and utilize advanced components of embedded systems design for architecting a complex system in the Zynq All Programmable System on a Chip (SoC), Zynq UltraScale+ MPSoC, or MicroBlaze soft processor.
This course builds on the skills gained in the Embedded Systems Design course. Labs provide hands-on experience with developing, debugging, and simulating an embedded system. Utilizing memory resources and implementing high-performance DMA are also covered. Labs use demo boards in which designs are downloaded and verified.

Level: Embedded Hardware 4

Training Duration: 2 days

Who Should Attend?

Hardware, firmware, and system design engineers who are interested in Xilinx embedded systems development flow.

Prerequisites:

- Embedded Systems Design course or experience with embedded systems design and the Vivado® Design Suite
- Basic C programming
- Working knowledge of the Zynq® All Programmable SoC (Cortex™-A9 processor), Zynq UltraScale+™ processors (Cortex-A53 or Cortex-R5 processors) or MicroBlaze™ processor

Software Tools:

Vivado Design or System Edition 2017.3

Hardware:

- Architecture: Zynq-7000 All Programmable SoC and Zynq UltraScale+ MPSoC*
- Demo board: Zynq-7000 All Programmable SoC ZC702 or ZedBoard*

* This course focuses on the Zynq-7000 All Programmable SoC, Zynq UltraScale+ MPSoC, and MicroBlaze processor architectures. Check with your local Authorized Training Provider for the specifics of the in-class lab board or other customizations. Zynq UltraScale+ MPSoC designs target QEMU rather than a specific board.

Skills Gained: After completing this training, you will be able to:

- Assemble an advanced embedded system
- Take advantage of the various features of Zynq All Programmable SoC, Zynq UltraScale+ MPSoC, and Cortex and MicroBlaze processors, including the AXI interconnect and various memory controllers
- Apply advanced debugging techniques, including the use of the Vivado analyzer tool for debugging an embedded processor system and HDL system simulation for processor-based designs
- Identify the steps involved in integrating a memory controller into an embedded system using the Cortex-A9 and MicroBlaze processors
- Integrate an interrupt controller and interrupt handler into an embedded design
- Design a flash memory-based system and boot load from off-chip flash memory

Course Outline

1. Overview of Embedded Hardware Development

Provides an overview of embedded hardware development.

2. Hardware-Software Flow

3. Software Overview

Provides a thorough understanding of how the integrated design environment works, including how the compiler and linker behave, basics of make files, DMA usage, and variable scope.

4. Zynq-7000 All Programmable SoC Architecture Overview

5. MicroBlaze Processor Architecture Overview

6. Zynq UltraScale+ MPSoC Architecture Overview

7. Debugging: Hardware Introduction

Introduces the need and offers a solution for in-chip testing of hardware designs.

Lab Description

1. Hardware-Software Flow

Illustrates how design information generated during the hardware development process is moved into the SDK tool realm.

2. Zynq-7000 All Programmable SoC Architecture Overview

Overview of the Zynq-7000 All Programmable SoC architecture.

3. MicroBlaze Processor Architecture Overview

Overview of the MicroBlaze microprocessor architecture.

4. Zynq UltraScale+ MPSoC Architecture Overview

Overview of the Zynq UltraScale+ MPSoC architecture.

Cont...

Advanced Features and Techniques of Embedded Systems Design

Course Outline

Cont...

8. Debugging: Marking Nets

9. Debugging: Hardware-Software Co-Debugging (Cross-Triggering)

10. Memory Types: Memory Overview

Provides a brief overview of the different types of memory available, as well as when one type of memory would be selected over another.

11. Memory Types: Block RAM Controllers

Introduces two versions of block RAM controllers and how and why they are needed.

12. Memory Types: Static Memory Controllers

Discusses static memory controllers in general and the SMC implementation in the Zynq-7000 family of devices.

13. Memory Types: DDRx Memory Operation

Provides additional details regarding how DDRx memory interfaces with a controller.

14. Memory Types: Dynamic Memory Controller (Zynq-7000 Device)

Covers how the DMC is implemented as well as many of its key behaviors.

15. Interrupt Concepts: Introduction to Interrupts

Introduces the concept of interrupts, basic terminology, and generic implementation.

16. Interrupt Concepts: Interrupts and the Zynq-7000 Device

Presents the details of how the Zynq-7000 platform uses interrupts from both a hardware and software perspective.

17. Interrupt Concepts: General Interrupt Controller

Introduces the general interrupt controller (GIC), its features, and some examples of its use.

18. Interrupt Concepts: Interrupts and the MicroBlaze Processor

Describes how interrupts are handled within the MicroBlaze processor system from a hardware perspective.

19. Interrupt Concepts: AXI Interrupt Controller for the MicroBlaze Processor

Introduces the AXI Interrupt Controller, which augments the MicroBlaze processor's interrupt capabilities by managing multiple interrupt sources.

20. AXI Concepts: AXI Streaming Introduction

Provides the context and background for the the streaming configuration of the AXI protocol.

21. AXI Concepts: MicroBlaze Processor Streaming Ports

Describes and illustrates how data streaming is performed using the MicroBlaze processor.

22. AXI Concepts: AXI Streaming FIFO

Introduces the AXI Streaming FIFO and its capabilities.

5. Debugging: Marking Nets

Reviews the process of marking nets to show which signals should be monitored without having to explicitly instantiate ILA cores.

6. Debugging: Hardware-Software Co-Debugging (Cross-Triggering)

Describes how to enable events in hardware to pause the software execution and breakpoints in software to cause an ILA trigger.

7. PS Peripherals - High-Speed: Gigabit Ethernet

Introduces the Gigabit Ethernet high-speed peripheral.

8. PS Peripherals - Low-Speed: Overview

Introduces the low-speed peripherals in the Zynq All Programmable SoC.

9. Sharing PS Resources (Hardware Perspective)

Illustrates from the hardware design perspective how a master in the PL can leverage resources within the PS.

10. Booting: PL

Introduces the concepts behind configuring the PL at boot.

Cont...

Advanced Features and Techniques of Embedded Systems Design

Course Outline

Cont…

23. AXI Concepts: Connecting AXI IP
Focuses on the relationships between different types of AXI interfaces and how they can be connected to form hierarchies.

24. AXI Concepts: DMA
Introduces various IP that supports DMA and DMA-like functionality.

25. Zynq-7000 Device PS-PL Interface
Discusses the various connection points between the PS and PL.

26. PS Peripherals - High-Speed: USB

Introduces the USB high-speed peripheral.

27. PS Peripherals - High-Speed: Gigabit Ethernet

28. PS Peripherals - Low-Speed: Overview

29. PS Peripherals - Low-Speed: CAN
Introduces the CAN low-speed peripheral.

30. PS Peripherals - Low-Speed: I2C
Introduces the I2C low-speed peripheral.

31. PS Peripherals - Low-Speed: SD/SDIO

Introduces the SD/SDIO low-speed peripheral.

32. PS Peripherals - Low-Speed: SPI

Introduces the SPI low-speed peripheral.

33. PS Peripherals - Low-Speed: UART

Introduces the UART low-speed peripheral.

34. Utility Logic

Covers the IP that provides basic logic support within the block design.

35. Sharing PS Resources (Hardware Perspective)

36. Multi-Processor Hardware Architecture

Addresses some of the mechanisms that a designer can leverage to support cross-processor communications.

37. Caching

Introduces the concept of caching and describes how this technique is implemented using the Xilinx processor systems.

38. Processor Caching and SCLR

Introduces the concepts behind processing caching and the System-Level Control Register.

39. Accelerator Coherency Port

Describes the purpose and general behavior of the accelerator coherency port (ACP).

40. Booting: Flow

Provides a low-level view of the booting process.

41. Booting: PL

42. Booting: Flash Image Generation

Introduces the Flash Image Generator tool, which is used to collect up a variety of files and order them properly in the Flash so that the FSBL can correctly read them.

43. QEMU: Introduction

Introduction to the Quick Emulator, which is the tool used to run software for the Zynq device when hardware is not available.