



## FC4 - MPC8610 implementation

*This course covers NXP MPC8610 Power CPU*

### Objectives

- The course clarifies the architecture of the MPC8610, particularly the operation of the coherency module that interconnects the e600 to memory and high-speed interfaces.
- Cache coherency protocol is introduced in increasing depth.
- The e600 core is viewed in detail, especially the AltiVec units that enable vector processing.
- The boot sequence and the clocking are explained.
- The course focuses on the hardware implementation of the MPC8610.
- A long introduction to DDR2 SDRAM operation is done before studying the DDR SDRAM controller.
- An in-depth description of the PCI-Express port is done.
- The course highlights both hardware and software implementation of integrated peripherals.
- This course has been delivered to companies involved in the design of avionics equipments.

A more detailed course description is available on request at [training@ac6-training.com](mailto:training@ac6-training.com)

### Prerequisites

- Experience of a 32-bit processor or DSP is mandatory.
- Knowledge of PCI Express bus (see our [IC4 - PCI Express 3.0](#) course) is recommended.

### Course Environment

- Theoretical course
  - PDF course material (in English) supplemented by a printed version for face-to-face courses.
  - Online courses are dispensed using the Teams video-conferencing system.
  - The trainer answers trainees' questions during the training and provide technical and pedagogical assistance.
- At the start of each session the trainer will interact with the trainees to ensure the course fits their expectations and correct if needed

### Target Audience

- Any embedded systems engineer or technician with the above prerequisites.

## Course Outline

### MPC8610 OVERVIEW

#### Overview

- Key features
- e600 core
- Coherency Module
- High-speed IO interfaces
- Examples of data flow through the MPC8610
- Understanding the operation of OCeAN switches
- 36-bit internal addressing
- Address map, local access windows
- Outbound and inbound address translation windows

# e600 CORE

## PIPELINE

- Introduction to e600 pipeline
- e600 pipeline implementation
- Issue queue resource requirements
- Execution model
- Dispatch conditions, completion conditions
- Execution serialization
- Branch management
- Guarded memory

## INTERNAL DATA AND INSTRUCTION PATHS

- L1 and L2 cache loading, hit under miss
- The MSS [Memory Sub System]
- The load fold queue
- The store miss merging mechanism
- The BIU [Bus Interface Unit]
- Purpose of sync and eieio instructions

## L1 AND L2 CACHES

- Cache basics
- Cache related page / block attributes
- e600 L1 cache
- Transient load instructions benefits
- L2 cache organization
- Cache coherency basics
- The MESI L1 data line states
- MESI snooping sequences involving the e600 and a PCI Express master
- Cache related instructions

## e600 PROGRAMMING

- User and supervisor registers
- Branch instructions
- The system call communication path between applications and RTOS
- Integer load / store instructions, boolean semaphore management
- Integer arithmetic and logic instructions
- IEEE754 basics
- FPU operation
- The EABI
- Code and data sections, small data areas benefits

## ALTIVEC

- AltiVec introduction, SIMD processing
- Intra vs inter element instructions
- ANSI C extension to support vector operators
- Vector load / store instructions
- Vector integer instructions
- Vector float instructions
- Vector permut instructions
- Data streams management

- EABI extension to support Altivec

## **THE MEMORY MANAGEMENT UNIT**

- MMU goals
- The PowerPC address processing
- 32-bit or 36-bit real address size selection
- WIMG attributes definition
- Process protection through VSID selection
- TLB organization
- Page translation
- Software vs hardware TLB reload
- MMU implementation in real-time sensitive applications

## **THE EXCEPTION MECHANISM**

- Supervisor registers : MSR, DAR, DSISR
- Exception state saving and restoring
- Exception management
- Recoverable vs non recoverable interrupts
- Registers updating related to the exception cause
- Requirements to support exception nesting

# **MPC8610 INFRASTRUCTURE**

## **RESET AND CLOCKING**

- Platform clock
- Power-on reset sequence
- Boot page translation
- Power management

## **MPX COHERENCY MODULE**

- I/O arbiter
- Transaction queue
- Global data multiplexor

## **PROGRAMMABLE INTERRUPT CONTROLLER**

- Open PIC architecture compatibility
- Interrupt nesting
- Description of the 4 timers / counters
- Message interrupts

## **DDR-SDRAM MEMORY CONTROLLER**

- DDR2 operation
- Jedec specification basics
- Hardware interface
- Bank activation
- ECC error correction
- On-die termination and driver calibration
- Introduction to the DDR-SDRAM controller
- Address decode
- Timing parameters programming
- Initialization routine

## **ENHANCED LOCAL BUS CONTROLLER**

- Multiplexed or non-multiplexed address and data buses
- Burst support
- GPCM, UPMs states machines
- Interfacing to ZBT SRAMs
- Interfacing to DSP host ports
- NAND flash controller

## **INTEGRATED DMA CONTROLLERS**

- Priority between the 4 channels
- Support for cascading descriptor chains
- Scatter / gathering
- Ability to start DMA from external 3-pin interface

## **PCI INTERFACE**

- Bridge features
- Inbound transactions handling, Outbound transactions handling
- PCI-to-memory and memory-to-PCI streaming
- Host vs agent configuration

## **PCI EXPRESS INTERFACE**

- Modes of operation, Root Complex / Endpoint
- Byte swapping
- Transaction ordering rules
- Programming inbound and outbound ATMUs

## **PERFORMANCE MONITOR AND DEBUG FEATURES**

- Event counting
- Chaining, triggering
- Watchpoint facility
- Trace buffer

# **MPC8610 INPUT / OUTPUT PERIPHERALS**

## **DISPLAY INTERFACE UNIT**

- Display interfaces
- Display color depth
- Plane blending
- Utilization of area descriptor
- Moving images through the dedicated DMA channel

## **I2C CONTROLLERS**

- I2C protocol fundamentals
- Transfer timing diagrams, SCL and SDA pins
- Transmit and receive sequence

## **SERIAL INTERFACE**

- Introduction to UART protocol
- Description of the NS16552 compliant Uarts

- Flow control signal management

## **SPI**

- SPI protocol fundamentals
- Transmit sequence
- Receive sequence

## **SYNCHRONOUS SERIAL CONTROLLER**

- Independent clock and frame sync signals for each receiver and transmitter
- I2S analog interface support
- Time Division Multiplexed support