This course covers PowerQUICC II Pro MPC8321, MPC8321E, MPC8323 and MPC8323E

Objectives

- The course clarifies the architecture of the MPC832XE and MPC8323E, particularly the operation of the coherency module that interconnects the e300 to memory and high-speed interfaces.
- Cache coherency protocol is introduced in increasing depth.
- The e300 core is viewed in detail, especially the MMU.
- The boot sequence and the clocking are explained.
- The course focuses on the hardware implementation of the MPC832XE.
- A long introduction to DDR SDRAM operation is done before studying the DDR1/2 SDRAM controller.
- Communication between CPUs through the PCI message unit is clarified.
- Interacting with the Security Engine through descriptors is studied as well as direct access to SEC registers.
- The course describes the sophisticated QoS mechanisms supported by the UCC Ethernet Controller.
- Regarding MPC8323E, a dedicated part on ATM controllers is proposed on request.
- Generation of a Linux image and Root File System by using LTIB can also be included into the training.

A more detailed course description is available on request at training@ac6-training.com

Prerequisites and related courses

- The knowledge of the following interconnect standards may be required:
  - PCI, see our course reference IC1 - PCI 3.0course
  - Gigabit Ethernet, see our course reference N1 - Ethernet and switchingcourse
  - USB 2.0, see our course reference IP2 - USB 2.0course

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.

Evaluation modalities

- The prerequisites indicated above are assessed before the training by the technical supervision of the traineein his company, or by the trainee himself in the exceptional case of an individual trainee.
- Trainee progress is assessed by quizzes offered at the end of various sections to verify that the trainees have assimilated the points presented
- At the end of the training, each trainee receives a certificate attesting that they have successfully completed the course.

- In the event of a problem, discovered during the course, due to a lack of prerequisites by the trainee a different or additional training is offered to them, generally to reinforce their prerequisites,in agreement with their company manager if applicable.

Plan

INTRODUCTION TO MPC832XE

Overview

- Internal architecture
- Highlighting data paths inside the MPC832XE, benefit of a dual-DDR controller system
- Block diagram: characteristics of each of the 3 internal modules e300 core, Platform, QuiccEngine
- Software migration from MPC8XX/MPC82XX/MPC85XX families
- Application examples

e300

THE INSTRUCTION PIPELINE

- Superscalar operation
- Branch processing unit
- Branch instructions
- Coding guidelines

DATA AND INSTRUCTION PATHS

- Load / store architecture
- Load / store buffers
- Sync and eieio instructions, determining where eieio is really required

CACHES

- Cache basics
- Cache locking
- PLRU algorithm, highlighting the difference between a True LRU and the PLRU replacement algorithms
- Shared resource management, lwarx and stwcx. instructions
- Cache coherency mechanism, snooping
- Memory coherency required attribute
- The MEI state machine
- Basic snoop requests: clean / flusk / kill
- Management of cache enabled pages shared with PCI DMAs
- Cache related instructions
- Software enforced cache coherency

SOFTWARE IMPLEMENTATION

- PowerPC architecture specification, the 3 books UISA, VEA and OEA
- addressing modes, load / store instructions
- Integer instructions
- Rotate instructions
- PowerPC EABI
- Linking an application with Diab Data, parameterizing the linker command file

THE MMU

- Thread vs process

- Introduction to real, block and segmentation / pagination translations
- Real mode restrictions
- Memory attributes and access rights definition
- Virtual space benefit
- TLBs organization
- Segment-translation: process ID definition
- Page-translation
- MMU implementation in real-time sensitive applications

THE EXCEPTION MECHANISM

- Supervisor registers description
- Save / restore registers for non-critical interrupts
- Critical interrupt, automatic nesting
- Exception management mechanism
- Requirements to allow exception nesting

THE DEBUG PORT

- JTAG emulation, restrictions
- Code instrumentation
- Hardware breakpoints

THE PLATFORM CONFIGURATION

POWER, RESET AND CLOCKING

- DC and AC electrical characteristics
- Configuration signals sampled at reset
- Reset configuration words source
- Utilization of the I2C boot sequencer
- PCI Host / Agent configuration
- Boot memory space
- Clocking in PCI Host mode, system clock domains
- External clock inputs
- System PLL ratio

PLATFORM CONFIGURATION

- Address translation and mapping, local memory map, local access windows
- Arbiter and bus monitor
- Sequencer
- General purpose inputs / outputs
- Timers

THE DDR2 MEMORY CONTROLLER

- DDR-SDRAM operation: a 128-Mbits DDR-SDRAM from Micron is used as an example
- Jedec specification basics, mode register initialization, bank selection and precharge
- On-Die termination and calibration
- Differences between DDR1 and DDR2
- Command truth table
- Bank activation, read, write and precharge timing diagrams, page mode
- DDR-SDRAM controller overview
- Initial configuration following Power-on-Reset
- Address decode
- Timing parameters programming
- Initialization routine

LOCAL BUS CONTROLLER

- Multiplexed or non-multiplexed address and data buses
- Burst support
- Dynamic bus sizing
- GPCM, UPMs states machines
- Interfacing to ZBT SRAMs

PCI BUS INTERFACES

- Bridge features
- Read prefetch and write posting FIFOs
- Inbound transactions handling, Outbound transactions handling in both modes
- PCI bus arbitration
- PCI hierarchy configuration when operating as host

INTEGRATED DMA CONTROLLER

- Priority between the 4 channels
- Scatter / gathering
- Selectable hardware enforced coherency
- Concurrent execution across multiple channels with programmable bandwidth control
- Messaging unit

INTEGRATED PROGRAMMABLE INTERRUPT CONTROLLER

- Interrupt sources
- Definition of interrupt priorities
- System critical interrupt
- Interrupt management, vector register
- Requirements to support nesting
- Machine check interrupts

SECURITY ENGINE

- Introduction to DES, 3DES and AES algorithms
- Data descriptor
- Crypto channels
- Link tables
- Operation of DEU, MDEU and AESU
- Snooping by caches

LOW SPEED PERIPHERALS

- Description of the NS16452/16552 compliant Uarts
- FIFO mode
- Flow control signal management
- I2C protocol fundamentals
- Transfer timing diagrams, SCL and SDA pins
- Transmit and receive sequence

QUICC ENGINE

SYSTEM INTERFACE AND CONNECTION TO EXTERNAL COMMUNICATION PORTS

- Serial DMA
- Multi-threading
- NMSI vs TDM

- Enabling connections to TSA or NMSI
- CMX registers
- Baud-rate generators

BUFFER MANAGEMENT

- Utilization of Buffer Descriptors
- Chaining descriptors into rings
- Interrupt management
- Parameter RAM independent of protocol

SERIAL PERIPHERAL INTERFACE [On-request]

- Introduction to SPI protocol
- SPI modes of operation
- SPI buffer descriptor
- Transmit and receive sequence

UNIFIED COMMUNICATION CONTROLLERS

- UCC feature set
- Handling UCC interrupts
- Initialization sequence
- UCC for slow communications controllers, UART mode
- UCC for fast protocols, virtual FIFOs
- Defining Tx- and Rx-FIFO thresholds

UCC ETHERNET CONTROLLER

- Physical interfaces to transceiver
- Auto-negotiation
- IP header checksum
- Flow control
- Frame filtering and address recognition, high level description of parse command descriptors
- Header parsing
- Quality of Service
- Interrupt coalescing
- Ethernet scheduler, traffic shaper
- BD and Parameter RAM description
- Ethernet statistics, MIB
- Ethernet host command set

QUICC MULTI-CHANNEL CONTROLLER [On request]

- QMC and serial interface
- Memory organization
- UCC Base and Global multichannel parameters
- Channel-specific HDLC parameters
- QMC exceptions
- QMC host commands

USB [On request]

- Host controller limitations
- Packet-level interface
- Transaction-level interface
- Endpoint parameters block pointer
- USB BD ring
- Host commands

THE ATM CONTROLLER [On request, MPC8323E only]

ATM BASICS

- ATM benefit compared to X.25 or ISDN
- Standardization and related links
- UNI and NNI network interfaces
- Cell format
- Virtual connection
- Layer model
- AAL1 layer: circuit emulation
- AAL3/4: used by the service providers
- AAL5: packet transfer

ATM TRAFFIC MANAGEMENT

- The 5 service classes defined by the ATM forum: CBR, VBRrt, VBRnrt, UBR, ABR
- The QoS ATM attributes: PCR/CDVT, CLR, CTD/CDV
- Traffic policy
- Traffic shaping
- Early packet discard

UTOPIA L2 BUS CONTROLLER

- Connection to 1 device through one UL2 Standard bus I/F
- Cell level handshake support
- Internal rate features
- Tx scheduling
- Rx cell transfer

THE UCC ATM CONTROLLER

- Introduction: the adaptation layers and the service classes supported by the UCC
- APC unit: schedule tables, GCRA algorithm for VBR traffic
- VCI/VPI of incoming cells lookup
- OAM AAL0 cells management
- Performance monitoring
- ATM/TDM interworking
- ATM controller parameter RAM description
- RxBD and TxBD format according to the adaptation layer

SERIAL ATM CONTROLLER

- Interworking between QMC and Serial ATM
- Transmit SAM features, payload scrambling
- Receive SAM features, cell delineation
- Run-time statistics
- Microcode TC Layer [MTC]

INVERSE MULTIPLEXING FOR ATM - IMA

- IMA frame, control cells, filler cells
- IMA User Plane functions
- Transmit queue operation
- Cell reception task
- low-level statistic counters

GENERATING THE LINUX KERNEL IMAGE

- Introducing the tools required to generate the kernel image
- What is required on the host before installing LTIB
- Common package selection screen
- Common target system configuration screen
- Building a complete BSP with the default configurations
- Creating a Root Filesystems image
- e-configuring the kernel under LTIB
- Selecting user-space packages
- Setup the bootloader arguments to use the exported RFS
- Debugging Uboot and the kernel by using Trace32
- Command line options
- Adding a new package
- Other deployment methods
- Creating a new package and integrating it into LTIB
  - A lot of labs have been created to explain the usage of LTIB

Renseignements pratiques

Inquiry : 5 days