FA1 - i.MX27 implementation + LTIB

This course describes the i.MX27 multimedia processor and Linux Target Image Builder tool

Objectives

- The course details the hardware implementation of the i.MX27 microcontroller.
- The boot sequence and the clocking are explained.
- The course explains all parameters that affect the performance of the system in order to easily perform the final tuning.
- A description of all internal peripherals is provided.
- An overview of the ARM926EJ-S core helps to understand issues caused by cache and MMU.
- The course ends with practical labs explaining how to generate a Linux image as well as a Root File System, by using a tool called LTIB [Linux Target Image Builder].

- This course has been delivered to several companies developing multimedia equipments.

A lot of programming examples have been developed by ACSYS to explain the boot sequence and the operation of complex peripherals.

- *They have been developed with GNU compiler and are executed under Lauterbach debugger.*
- *Furthermore, a host desktop running Fedora Linux is used to generate Linux image and Root File System during labs on LTIB.*

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

Prerequisites

- This course provides an overview of the ARM926 core. Our course reference R1 - ARM7/9 implementationcourse details the operation of this core.
- The following courses could be of interest:
  - USB Full Speed High Speed and USB On-The-Go, reference IP2 - USB 2.0course
  - Ethernet and switching, reference N1 - Ethernet and switchingcourse

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

ARCHITECTURE OF i.MX27

Overview

- ARM core based architecture
- Clarifying the internal data paths
- Highlighting the purpose of the 2 central interconnect units : MAX and M3IF
- Organization of a board based on i.MX27
- Mapping

CORE PLATFORM

THE ARM926EJ-S CORE

- Presentation of the core
- Operating modes
- Pipeline
- ARM vs Thumb instruction sets, interworking
- Branch instructions
- C-to-Assembly interface
- Exception mechanism
- Debug facilities

THE ARM9 PLATFORM

- AHB slave device latencies
- MAX parameterizing
- ARM Interrupt Controller [AITC]

HARDWARE IMPLEMENTATION

RESET AND CLOCKING

- Clock distribution
- Power-up sequence
- Low power modes, clock gating
- System boot mode selection
- Bootstrap mode operation

SYSTEM CONTROL

- GPIO module
- General Purpose Input interrupt request capability
- Signal description

ACCESSING EXTERNAL MEMORIES

- Description of the Master Arbitration and Buffering [MAB] unit
- Description of the M3IF arbitration [M3A]

- Enhanced DDR SDRAM controller
- NAND flash controller, boot from flash
- Programming the chip-selects

STANDARD PARALLEL INTERFACES

    - ATA controller
- PIO mode
- Ultra DMA mode
- FIFO receive and FIFO transmit alarms
    - MSHC
- Transfer protocol
- Error management
    - SDHC
- Interface to SD cards
- Transfer protocol
- Error management

MULTIMEDIA UNITS

DMA CONTROLLER

- Channel priority definition
- Burst length definition
- 2D memory transfers
- Double-buffering mechanism enabling chained transfers

VIDEO PROCESSING UNITS

    - Video acquisition
- CSI interface
- Configuring the interface to support CCIR656
    - Video pre-processor
- Image resizing
- Color space conversion
    - Video post-processor
- Deblock
- Dering
- Image resizing
- Color space conversion
    - Video codec
- MPEG-4 encoding / decoding
- H.264 AVC encoding / decoding

AUDIO RELATED INTERFACES

    - SSI interfaces
- Connection of Codecs or DSPs
- AC97 support
    - Digital audio multiplexor
- Connecting host interfaces to peripheral interfaces
- Internal network mode

SECURITY MODULES

    - Security Controller
    - SAHARA2 security coprocessor
- Random number generator
- Encryption / decryption sequences

    - Run-Time Integrity Checker
- SHA-1 message authentication
- Segmented data gathering
    - IC Identification Module

COMMUNICATION CONTROLLERS

    - 1-wire interface
    - Configurable SPI
- SPI protocol basics
- Master / slave operation
- Transfer sequence
    - I2C interfaces
- I2C protocol basics
- Master vs slave
- Transfer sequence
    - UART
- IrDA modulation / demodulation
- Support for Smart Card
- Flow control
    - USB
- Explaining what is OTG
- High-speed operation
- EHCI support
- Full speed operation
- Endpoint configuration
    - Fast Ethernet Controller [FEC]
- Buffer management, based on Buffer Descriptors
- Incoming frame filtering mechanisms
- VLAN support

LCD CONTROL

    - LCDC
- LCD screen format
- Standard panel interface for common LCD drivers
- Graphic window on screen
    - SLCDC
- Interface to an external display controller
- Transferring images and controls from DDR to the external controller

LTIB

GENERATING THE LINUX 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
- Re-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
- Adding a new package
- Other deployment methods
- Creating a new package and integrating it into LTIB

Renseignements pratiques

Duration : 4 days
Cost : 0 € HT