Designing with i.MX6 Family

Course Description

Designing with Freescale i.MX6 family is a 3 days Freescale official course.

The course provides all necessary theoretical and practical know-how for start developing platforms based on i.MX6 family.

The course begins with an introduction to i.MX6 family and its target applications, and focuses on Cortex-A9 MPCore architecture.

The course continues with an in-depth study of the device architecture, memory organization, image processing unit (IPU), video processing unit (VPU), graphic processing unit (GPU), security, I/O muxing, clock management, reset management, power management, and audio interfaces.

The course covers also the new IP blocks in i.MX6 family, external memory interfaces and configurations, boot modes, low power modes and system management.

An optional day extension will be provided upon customer’s request to cover major OS such as Embedded Linux, WINCE and Android.

Course Duration

3 days (4th optional day)
Goals

1. Become familiar with i.MX6 family and its applications
2. Become familiar with ARM Cortex-A9 architecture
3. Become familiar with i.MX6 SoC architecture
4. Be able to configure correctly clocks, reset and power
5. Be able to connect external memories and understand the tradeoffs
6. Become familiar with image processing unit (IPU) and its configuration
7. Become familiar with video processing unit (VPU) and its configuration
8. Become familiar with graphic processing unit (GPU) for 3D and 2D applications
9. Become familiar with i.MX6 new IPs
10. Become familiar with i.MX6 security architecture
11. Become familiar with i.MX6 boot options including secure boot
12. Become familiar with hardware and software design tools
13. Work with Firmware libraries

Target Audience

Software and hardware engineers who would like start developing with i.MX6 processors

Prerequisites

- Computer architecture background
- Experience in developing embedded systems

Course Material

Demonstration system
Course book
Agenda

Day #1

- **Introduction to i.MX6 Family**
  - Main Features and Road Map
    - i.MX family roadmap
    - Qualification tiers (industrial, commercial, automotive, custom)
    - Freescale’s product longevity program
  - Target Markets Applications
    - General embedded
    - Medical
    - Tablet and eReader
    - Auto infotainment
    - Home energy
    - Real products example
  - i.MX6 Series Overview
    - i.MX6 solo, solo lite, dual, dual lite, quad core comparison and main features
    - i.MX6 block diagrams overview
    - Packaging
    - Development boards
    - Linux, Android and graphics software roadmaps
  - Device Architecture Overview
    - i.MX6 devices block diagrams in details (core and accelerators)
    - Dual core use cases and performance comparison
    - Video Capabilities and SW codecs (standards, performance)
    - Graphics capabilities (2D & 3D, openGL vs openVG, openCL support, GPU performance with iMX family)
    - Display support (display resolution, multiple display support)
    - Multimedia processing chain (integration of ARM core, IPU, VPU, and GPU)
  - Memory System and DDR Controller
    - Multi-Mode DDR Controller (MMDC) main features
    - MMDC low power support
    - MMDC calibration
    - DDR standards supported
    - DDR configuration examples
    - Low latency and bandwidth issues
    - MMDC configuration
    - LPDDR2 Two-channel interleaving
    - DDR channels and address mapping

When innovation meets expertise...
o MMDC modes and basic settings
o DDR3 DLL OFF mode
o Recommended MMDC settings for best performance
o From bus system to DDR controller routing and bandwidth
o Quality of service
o Bus/DDR arbiter
o Bus/DDR profiling

- i.MX6D/Q Interfaces
  o Memory and mass storage interfaces (NAND, EIM, LPDDR2/3, serial ePROM/serial NAND, SD/MMC, SATA)
  o High speed connectivity (USB, Ethernet Controller, PCI-E)
  o Audio system connectivity (S/PDIF, SSI, AudioMUX, EAI, ASRC)
  o General purpose connectivity (Ecspi, Keypad, UART, GPT, PWM, SDMA, I2C, GPIO, One-Wire)
  o Camera interface and displays

- i.MX6 Dual/Quad ARM Cortex A9 MP core platform
  o introduction to ARM Architecture
  o Platform overview (Core, Neon, private timer, watch dog, L1 Cache, L2 cache, SCU, GIC, TrustZone, MMU, CoreSight debug)
  o Platform configuration (single, dual, Quad and SCU configuration, endian support, memory parity error support)

- Clocking & Reset
  o Clock generation scheme
  o Clock configuration
  o Clock change procedures
  o Reset scheme

- Power Management
  o i.MX6 series power management overview
  o integrated PMU
  o system power tree
  o PFUZE PMIC advantages
  o i.MX6 power needs
  o PFUZE100 overview
  o Temperature monitor
  o Low power modes
  o Wake-up events
  o Power saving techniques
Day #2

- **i.MX6 Multimedia**

  - **Image Processing Unit (IPU)**
    - IPU Block Diagram and Architecture
    - Multiple Display setup and operation
    - Display ports muxing
    - Max display port resolutions
    - Video input ports
    - Display interface (DI)
    - Display controller (DC)
    - DMFC
    - Image DMA controller (IDMAC)
    - Display processor (DP)
    - Image converter (IC)
    - Image rotator (IRT)
    - Video de-interlacer or combiner (VDIC)
    - Basic combining capabilities
    - Off-line combining
    - Maximal on-the-fly combining to a single display
    - On-the-fly combining using 2xIPU
    - IPU programming steps
    - Camera sensor interface (CSI)
    - 16 bit camera support
    - Control module (CM)
    - Use case examples
    - Memory Bandwidth/considerations

  - **Display Content Integrity Checker (DCIC)**
    - Display content authentication
    - Ports muxing: DCIC options

  - **Video Processing Unit (VPU)**
    - VPU performance/capability overview
    - The multimedia processing chain
    - VPU decoding, encoding, multi-streaming, transcoding & full-duplex
    - Codec licensing
    - Video encoding/decoding algorithm
    - VPU Block Diagram and Architecture Overview
    - VPU-IPU interface
    - Bandwidth Considerations
    - VPU programmable engine architecture
    - VPU driver API
    - VPU tiled format support
    - Tiled format handling in Video data order adapter (VDOA)
- JPEG processing unit (JPU)
- VPU software structure
- Video playback using VPU
- Supported streaming protocol

- **Video Data Order Adapter (VDOA)**
  - VDOA block diagram
  - VDOA features
  - VDOA data path
  - VDOA control

- **Graphics Architecture (GPU)**
  - i.MX6 GPU processing units
  - OpenVG (GC355)
  - Composition engine 2D (GC320)
  - OpenGL ES 3D engine (GC2000)
  - Vivante GC architecture overview
  - Vivante GC software architecture overview
  - FMR vs TBR: advantages and optimizations
  - Benchmarking
  - Vivante GPU tools
  - i.MX6 graphics application development
  - OpenCL support

---

**Day #3**

- **i.MX6 New IP Blocks**

- **GPMI & NAND Interface Subsystem**
  - Architecture overview
  - NAND interface
  - Error correction
  - NAND control
  - BCH limitations and assumptions

- **1 Gigabit Ethernet**
  - MAC-NET features
  - IP protocol performance optimization features
  - IEEE1588 features
  - MAC-NET block diagram
  - Media independent interface
  - Ethernet clock generation
• **MIPI Display & Sensor Interfaces**
  - MIPI DSI overview
  - MIPI DSI features
  - MIPI DBI interface
  - MIPI CSI-2
  - MIPI CSI interface payload data format
  - MIPI HSI block diagram
  - MIPI HASI features
  - MIPI HIS operations

• **PCI Express**
  - PCI-E overview
  - PCI-E protocol stack
  - PCI-E link
  - PCI-E end-point
  - PCI-E root complex
  - Address translation

• **HDMI**
  - HMDI general features
  - HDMI architecture overview
  - Audio DMA interface
  - HDCP
  - EDID/HDCP I2C E-DDC interface
  - CEC hardware engine

• **Audio**
  - Connectivity highlights
  - Enhanced serial audio interface (ESAI)
  - ESAI vs SSI
  - Asynchronous sample rate converter (ASRC)

• **i.MX6 Boot**
  - Boot modes and boot sources
  - Boot mode pin settings
  - High level boot sequence
  - Boot from fuses mode
  - Serial downloader mode
  - Internal boot mode
  - Boot security settings
  - GPIO override pads
  - Multi-core boot flow
  - Encrypted image support
  - High assurance boot (HAB)
• Security
  o Security key features
  o Security hardware
  o Common security attacks
  o Security architecture block diagram
  o ARM TrustZone
  o Boot ROM: high assurance boot
  o HAB4 features
  o Cryptographic acceleration and assurance module (CAAM)
  o CAAM block diagram and interfaces
  o Secure memory
  o Algorithm acceleration in CAAM
  o Secure non-volatile storage (SNVS)
  o SNVS block diagram and interfaces
  o Secure JTAG controller
  o Memory and peripheral isolation for TrustZone support
  o Security configuration and lifecycle
  o TrustZone hardware
  o TrustZone multicore support
  o TrustZone software

• Input/Output
  o I/O Muxing
  o Pad Assignments and Configuration
  o Resolving pin conflicts
  o Ensuring voltage consistency per module
  o Alternate I/O mux views

• System on a Module (SOM)
  o Embedded board solutions
  o Make vs buy decision matrix
  o Examples
Day #4 (optional)

- Freescale i.MX6 Software Release
  - i.MX6 Linux Software Release
    - Introduction to Linux
    - Freescale software release overview
    - i.MX6 series Firmware
    - Introduction to LTIB
    - Boot the i.MX6 from SD card/Flash memory/NFS
    - Utilizing Freescale i.MX6 software release based on LTIB
      - U-Boot
      - Linux kernel
      - File system
      - Manufacturing Firmware
    - Freescale root file system
      - LTIB based
      - Gnome mobile Root File System
      - Ubuntu demo rootfs
  - i.MX6 Android Software Release
    - Android overview and demo
    - Freescale i.MX6 Android SW release
  - Board Support Package (BSP)
    - Introduction to BSP and drivers
    - Freescale BSP modifications overview
    - Freescale Drivers overview