eCockpit Proof of Concept

Philppe Desblancs

Program Manager MICR Advanced Technologies (AT) Team

October 2018 | AMF-AUT-T3164



 \square



Company Public – NXP, the NXP logo, and NXP secure connections for a smarter world are trademarks of NXP B.V. All other product or service names are the property of their respective owners. © 2018 NXP B.V.

eCockpit POC objective

- Demonstrate i.MX 8QM HW performance & partitioning capabilities by running two concurrent HLOS and a Safety Critical application (RVC, Display, Audio) on the same SoC, without using a SW Hypervisor solution
 - Linux OS Instrument Cluster (IC)
 - Android or Linux OS Head Unit (IVI)
 - FreeRTOS SafeAssure RVC/Display/Audio

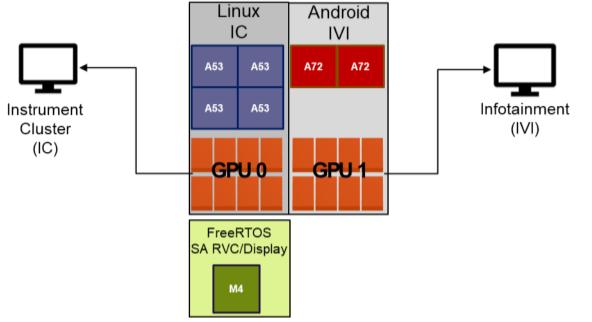
Advantage of eCockpit approach vs. Hypervisor

- Complexity
 - No 3rd party software to add to already complex SW stack
- Cost
 - Provided at zero cost as enablement
 - Professional Services possible to port, productize, support, maintain on customer platform
- Certification
 - Safety critical applications requiring certification (ISO 26262) are properly isolated from the potentially noncertified Instrument Cluster and Head Unit (IVI) applications on same SoC



Systems Architecture concept

- The eCockpit architecture provides a way to implement several (up to 3) HW isolated subsystems on i.MX 8QM with different safety levels.
- Mechanisms are implemented in HW and under the control of System Controller Unit (SCU) via the **xRDC** module

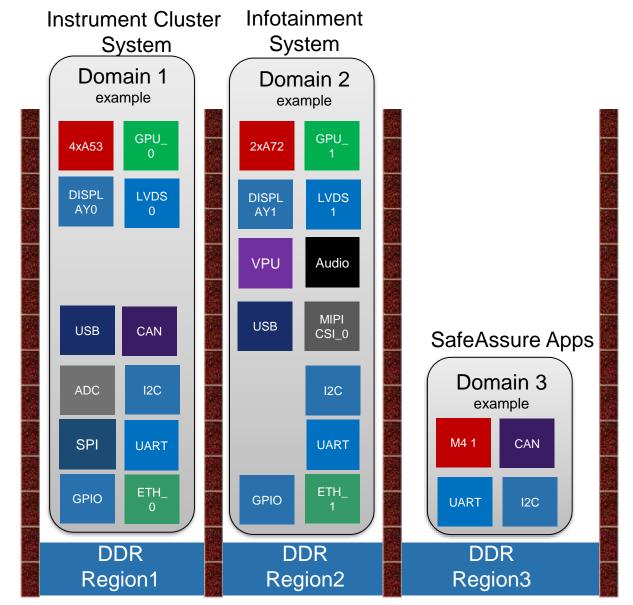


Hardware Isolation Implementation

- Define hardware partitions
 - Each subsystem has a partition
 - Each hardware resource is assigned to a single subsystem's partition
 - Access to a resource by another partition is prevented by default
- Define Memory Regions
 - A memory region is assigned to a single subsystem's partition
 - A region access is allowed to hardware modules from a single partition
- Assign IO pads to hardware partitions
- Route IRQs to their respective subsystem

Hardware isolation example for NXP i.MX 8QM ref platform

- IVI partition
 - Based on the Cortex-A72 cluster
 - Has exclusive control over following HW modules
 - 1 GPU
 - 1 Display Controller
 - VPU
 - Imaging (JPEG, ISI, CSI)
 - Audio (ASRC, Mixer, SAI, ESAI, SPDIF)
 - eMMC main storage media
- IC partition
 - Based on the Cortex-A53 cluster
 - Has exclusive control over following HW modules
 - 1 GPU
 - 1 Display Controller
 - NOR Flash (QSPI)
 - SD card main storage media
- SafeAssure partition
 - Based on one Cortex-M4 CPU





System Boot Sequence – i.MX 8QM A0

SCU (Cortex-M4)	IC (Cortex-A53 cpu0)	IVI (Cortex-A72 cpu0)	
Boot ROM (load SCUFW+DCD from USB/SD/MMC/Flash to TCM)			
Execute DCD			
Start SCU Firmware			
Board System Config (setup HW partitions #3, #4, #5, Memory Regions)			
Boot A53 (@0x0000 0000)	·		
Board init (phase 2: waits for A53 ROM end of loading, then Copy atf+uboot A72 from A53 RAM to A72 RAM and Boot A72@0xc80000000)	ROM (address 0x0000 0000) (load atf+uboot for A53 & A72 in RAM)		
	Start ATF (address 0x8000 0000) (create SMC)	Ctout ATE	
Assign A53 resources to partition#4 (non-secure)		Start ATF Via trampoline 0x8000 0000 -> 0xC000 0000 (create SMC)	
Assign A72 resources to partition#5 (non-secure)	•		
	Start u-boot	Start u-boot	
	Start Kernel	Start Kernel	

SubSystem (core) SCU FW Function			
IPC (MU)			
Boot			

Non-Secure Boot flow implemented currently.

- Limitation: current Boot ROM supports only a single Cortex-A boot image in the Secondary Boot Container.
 - Workaround: boot IC subsystem (A53 core 0) first, its bootloader (uboot) loads and boots the IVI subsystem image
 - Will be fixed in i.MX 8QM B0.



System Memory map example for i.MX 8QM Val board

	0x80000000	ATE A72 (128LD)	0xC0000000
ATF A53 (128kB)	0x80020000	ATF A72 (128kB)	0xC0020000
U-boot A53 (2432kB)	0x80280000	U-boot A72 (2432kB)	0xC0280000
Linux A53 (45.5 MB)	0,00200000	Linux A72 (45.5 MB)	0,00200000
	0x83000000		0xC3000000
DTB A53 (2.5 MB)	0x83280000	DTB A72 (2.5 MB)	0xC3280000
205.5 MB		205.5 MB	
	0x9000000		0xD0000000
CMA A53 (640 MB)		CMA A72 (640 MB)	
80 MB	0xB8000000	80 MB	0xF8000000
	0xBD000000		0xFD000000
U_boot A53 relocated (48 MB)		U_boot A72 relocated (48 MB)	
Telocated (40 MB)	0xC0000000		0x10000000
COMPANY PUBLIC 6			

Software Components Overview

- Boot SW
 - SCU Firmware
 - SCU is the centralized resource controller for the whole SoC
 - SCU firmware exposes to other CPU cores an API to query the resource usage and locking via Messaging Units (MU)
 - SCU controls access rights to HW resource via the xRDC module.

- ARM Trusted Firmware (ATF)

- Boots A53 with Exception Level 3 (EL3) using Boot Loader 31 (BL31), and creates the first non-secure HW partition.
- Creates the Secure Monitor allows non-secure world to perform platform initialization and power control.
- Enables by default the cache coherency between A53 and A72 clusters
- Boot loader \rightarrow **U-boot** for both IC and IVI sub-systems
- Linux kernel
 - Each IC and IVI partition will run a Linux kernel with separate Device Tree Sources defining HW modules mapped to each HW partition



Current status & next steps

- Proof of concept phase in execution until e/o 2018
 - Succesfully implemented hardware isolation and achieved booting two HLOS (Android & Linux) on an i.MX 8QM MEK platform
 - User triggered partition reset functional independent partition reboot
 - Working on SCU triggered partition reset to support « CPU hang » scenario on any of the 2 Cortex-A clusters
- Deployment phase plan for Q1'19
 - Develop collaterals (Application Note) to guide customers build and use eCockpit enablement SW on NXP reference platform and provide guidelines to properly design HW platforms for eCockpit / HW isolation concept
 - Package eCockpit SW solution as a set of patches on top of NXP generic BSP's (Linux / Android)
 - Exact Software Distribution Details will be provided at the time of i.MX 8QM product launch



Future roadmap items

- Resource sharing across domains
 - Integrate a 3rd safety (ISO26262) partition in the system with FreeRTOS on C-M4 running a SafeAssure application
 - SafeAssure RVC
 - SafeAssure Audio
 - SafeAssure Display rendering of a 2D cluster using only the M4 cores, would allow to "crash" the IC and IVI partitions while still maintaining safety critical information to the display
- Being able to have one Layer with e.g. Navigation content prepared by the IVI Nav apps (with the IVI GPU) and overlayed on the IC Display
 - Can be achieved with the i.MX 8QM Display controller, and Android O MR1 offers such a Virtual Display concept.





SECURE CONNECTIONS FOR A SMARTER WORLD

www.nxp.com

NXP, the NXP logo, and NXP secure connections for a smarter world are trademarks of NXP B.V. All other product or service names are the property of their respective owners. © 2018 NXP B.V.