LEARN FROM OUR EXPERIENCE! HINTS, TIPS AND TRICKS TO SPEED YOUR QORIQ® LS1012A, LS1043A AND LS1046A DESIGN CYCLE USING NXP SUPPORT

MARLAN WINTER
QORIQ SW & SERVICES

AMF-NET-T2776 | AUGUST 2017

SECURE CONNECTIONS FOR A SMARTER WORLD

AGENDA

- NXP - More than “chips”
- Tips from the techies
  - Bring up your LS1
  - Tips to get full advantage of pin compatibility
- Projects are too big to “go it alone”

01.
NXP IS FAR MORE THAN “JUST CHIPS”

Enabling Customers with Software & Solutions

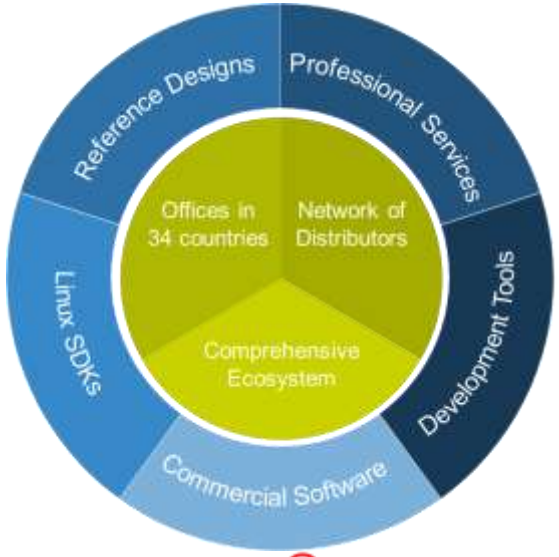
- Significant investment in Software R&D
  - Over 700 networking software engineers
  - Focus on emerging technologies – SDN, NFV, IoE, 5G
  - Top 10% Company Contributor to kernel.org for the last 7+ years!
  - Field hardened & widely deployed >10 Million copies deployed to date
- Driving ARM64 ecosystem in close collaboration with ARM Inc
- Most mature ARMv8 64-bit Software - hardened over 2 years
- Broad range of software products
- SDK, acceleration kits, middleware, commercial & production-ready SW
- Comprehensive HW+SW solutions for next generation networks
- Professional software services and support
- Active involvement and leadership in consortiums, industry forums and working groups

ONF OPEN NETWORKING FOUNDATION — *Chipmaker's Advisory Board*

InCNTRE

Linaro — *Founding Member*

OpenDataPlane.org



NEC, BROCADE, FUJITSU, Visteon, CISCO, ADTRAN, Google, FOXCONN, KANGURU, Raytheon, CRYPTO, ERICSSON, ViaSat, Rubicon, cohere technologies, NETSCOUT, TATA POWER, GECO, Airvana, SPIRENT Communications, SAGEM

*Representative customer logos

QorIQ/LS Software and Services Group

Software Products and Custom Services

Development Tools	Runtime Products	Solutions Reference	Linux® Services	Integration Services
• CodeWarrior – IDE – Debug – Compiler – Trace – Scenarios Tools – DDrV – SerDes V	• VortiQa Software Products – Application Identification Software (AIS) – SDN Solutions – Mobile Transport – Layer 1 • Application Solution Kits (ASK) – NAS, BHR • Application Development Kits (ADK) • Network Services Switch	• Storage Controller • SDN Switch • Wireless LAN • Data Concentrator • Smart Converged Gateway • Digital Signage • Proof of Concept • OpenWRT+	• Commercial Support • Frozen Branch • Application-Specific Hardening • Feature Acceleration • Forward Port/Kernel Refresh • WiFi • Performance	• Systems Consulting • Discovery Contract • Security (All Levels) • Design Services • Porting • Emergency Boot • Uncommon IP • Functional Simulators
nxp.com/**cw4net**	nxp.com/**vortiqa**	nxp.com/**solutions-reference**	nxp.com/**networking-services**	

CodeWarrior QorIQ

VortiQa

02.

Tips from the techies

Assumptions

- QSPI flash is primary boot location
  - location for RCW, bootloader, Linux
- Bootloader relocates to DDR after initial execution
- Primary Core: Core0
- NXP debug tools used: CodeWarrior, CodeWarriorTAP
- NXP software used: NXP SDK
- No Secure Boot
- Attendees have prior silicon bring up experience

Frequently Used Acronyms

IFC	Integrated Flash Controller
DDR	Double Data Rate (interchangeably used with DDR)
DDRv	DDR Validation Tool
LS	LayerScape
PBI	Pre-Boot Initialization
PBL	Pre-Boot Loader
QCVS	QorIQ Configuration and Validation Suite
RCW	Reset Configuration Word
SDK	Software Development Kit

CodeWarrior Development Suites for Networked Applications

- **Overview**
  - Enhance the success of your networked design by utilizing one of the suites from the CodeWarrior Development Suites for Networked Applications. The following are the benefits you will gain from our networking-focused suites

- **Benefit**
  - The development suites support multiple architectures including Power Architecture® technology, ARM® technology and StarCore DSP cores
  - Merges the best aspects of Eclipse, GDB, GCC & NXP into a single IDE
  - Use the tools that NXP uses to simplify & speed support
  - Far easier bring-up than uboot brute force
  - Registers name/info from users manual by reg/bit field available in debugger
  - Continuity with all NXP QorIQ processors using CW
  - Multiple debug configs possible to fit every situation

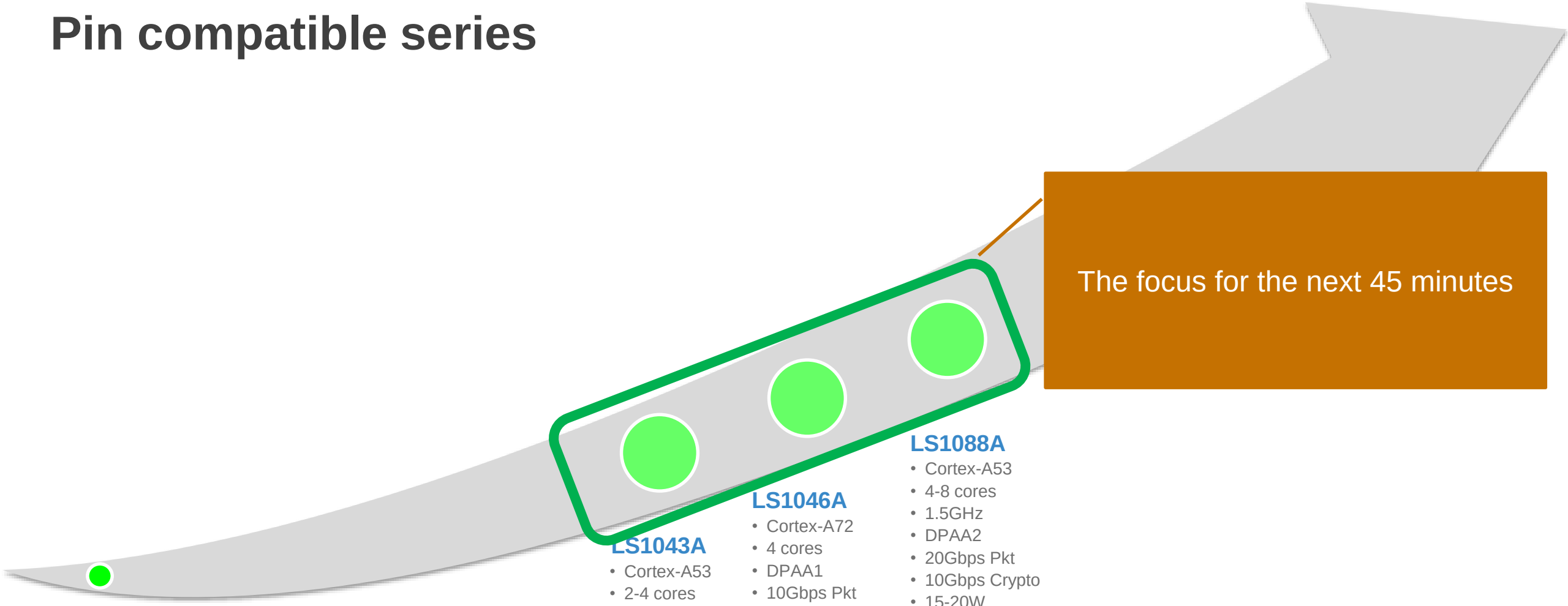
    - For more information please visit http://nxp.com/cw4net.

If you use CodeWarrior…

You'll be using the same tools we do

It is just one less thing to deal with.

Pin compatible series



The focus for the next 45 minutes

LS1012A
- Cortex-A53
- 800MHz
- 2Gbps Packet
- 1Gbps Crypto
- 1-2W
- Lowest power 64-bit ARM

LS1043A
- Cortex-A53
- 2-4 cores
- 10Gbps Pkt
- 5Gbps Crypto
- 5-10W
- 1st 64-bit ARM processor for gateways and access points

LS1046A
- Cortex-A72
- 4 cores
- DPAA1
- 10Gbps Pkt
- 10Gbps Crypto
- 10-12W
- 1st Value Tier A72 ARM for gateways and routers

LS1088A
- Cortex-A53
- 4-8 cores
- 1.5GHz
- DPAA2
- 20Gbps Pkt
- 10Gbps Crypto
- 15-20W
- 1st 8x A53 ARM Next gen programmable offload

Finding the right collateral

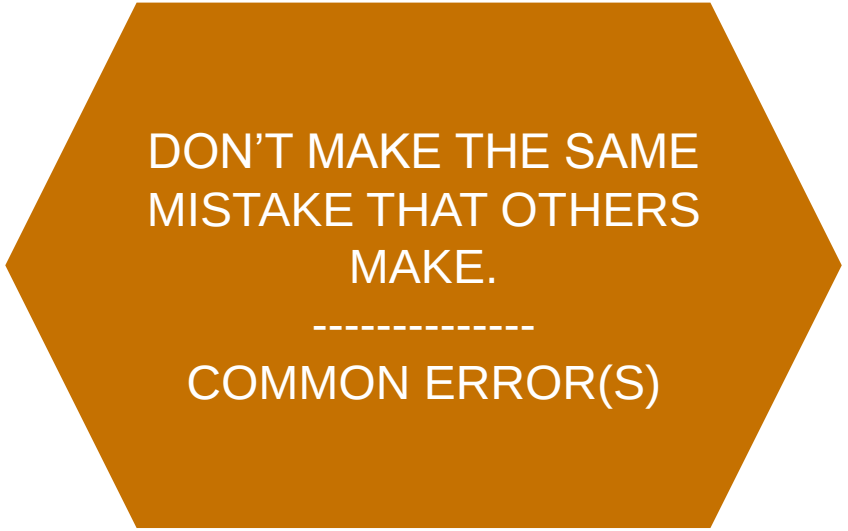
- All information is available on NXP.com
- Product page
- Reference design board
- Collateral needed
  - *QorIQ LS1043A Reference Manual*
  - *QorIQ LS1046A Reference Manual*
  - *QorIQ LS1088A Reference Manual*
  - *QorIQ LS1043A, LS1023A Data Sheet*
  - *QorIQ LS1046A, LS1026A Data Sheet*
  - *QorIQ LS1088A Data Sheet*
  - *AN5012 - LS1043A Design Checklist*
  - *AN5252 - LS1046A Design Checklist*
  - *AN5144 - LS1088A Design Checklist*
  - *AN5097 - Hardware and Layout Design Considerations for DDR4 SDRAM memory interfaces*
  - *AN5226 – Common Board Design for LS1046A, LS1043A and LS1088A Processor*

DON'T MAKE THE SAME MISTAKE THAT OTHERS MAKE.

COMMON ERROR(S)

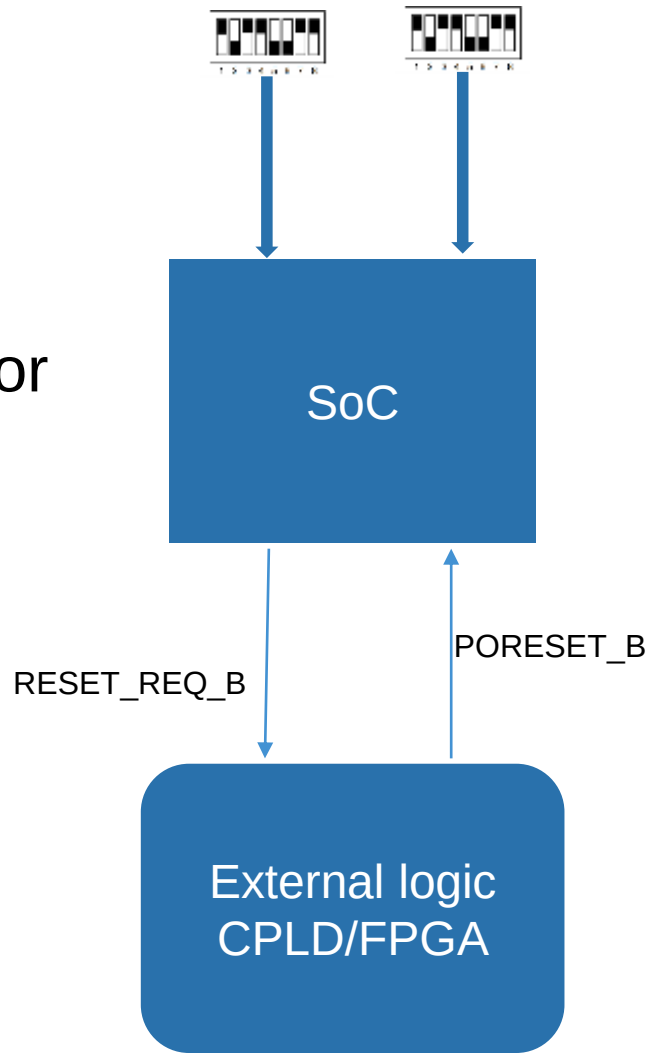
Design Considerations

- Power supply requirements
- Boot source
  - IFC: NOR Flash, NAND Flash - Avail, but not a pin-compatible choice
  - QSPI
  - eSDHC/MMC
- Using CPLD/FPGA for reset and signal routing
  - Reset_req_b to be connected to FPGA/CPLD
- Clock sources
  - SYSCLK and DDRCLK must always be driven
  - ECn_GTX_CLK125 must be driven if RGMII mode is used on the respective ECn port
  - SerDes reference clocks (SDn_REF_CLKn and SDn_REF_CLKn_B) must be driven if the corresponding SerDes bank is enabled in the RCW
  - Optional input clock sources include RTC, USBCLK, and TSEC_1588_CLK_IN



PORESET Signals

- Various chip functions are initialized by sampling certain signals during the assertion of PORESET_B.
- These power-on reset (POR) inputs are pulled either high or low during this period.
- While these pins are generally output pins during normal operation, they are treated as inputs while PORESET_B is asserted.



Reset configuration name	Functional interface	Functional Signal Name	Default
cfg_rcw_src[0:7]	IFC	IFC_AD[8:15]	1111 1111
cfg_rcw_src[8]	IFC	IFC_CLE	1 [1]
cfg_ifc_te	IFC	IFC_TE	1
cfg_dram_type	IFC	IFC_A[21]	1
cfg_gpinput[0:7]	IFC	IFC_AD[0:7]	1111 1111
cfg_eng_use0	IFC	IFC_WE0_B	1

EXTRA TIME HERE IS WORTH IT

RCW, PBI and PBL

- **PBL:** Preboot loader, a state machine logic responsible for reading RCW and PBI from external memory source, checking their integrity and loading the registers with information contained in RCW and PBI prior to boot. PBL can read RCW and PBI from IFC (NOR and NAND FCM), eSDHC/eMMC and qSPI.

- **RCW:** 512-bit long sequence of data containing configuration information. Read from external memory(normal operation). Ex: PLL, SerDes, DDR and pin muxing information

- **PBI:** PBI are set of commands/instructions that can be used to set up DDR and other interfaces, implement errata etc.

- **Note:** You can perform a RCW override using CodeWarrior. This will write RCW values from a file on your PC to the SoC before HRESET thus overriding RCW fetched from cfg_rcw_src configuration

UNDERSTANING THIS PROCESS IS IMPORTANT

Description of PBL functionality

- **PBL RCW phase**
  - PBL starts to load RCW data form specified source (cfg_rcw_src[0:n])
  - PLLs begin to lock
  - All 512 bits of RCW are loaded
  - Sequence completes

UNDERSTANING THIS PROCESS IS IMPORTANT

- **PBL PBI phase**
  - PBL switches to platform clock
  - PBL checks RCW[PBI_SRC]
    - **If PBI is disabled,** then PBL is done
    - **If PBI is enabled,** proceed to fetch PBI data from the source.
  - PBL finishes the PBI
  - System Ready state
  - Peripheral interfaces are released to accept external requests
  - Release the core0 to fetch instruction if RCW[BOOT_HO] is 0.
    - The boot code location is specified in SCRATCHRW2 (for non-secure boot) and SCRATCHRW1 should be 0
    - PBI should write the boot location to scratch registers in LS1 devices
  - ASLEEP negates

- For both RCW and PBI phase, if there is any error, the boot stops and /RESET_REQ is asserted
- PBI can also be used to implement any features or errata
- FTF-DES-N1832 on how to build and deploy RCW and PBI

POR Configs

Significant POR configs	LS1046A	LS1043A	LS1088A
Cfg_rcw_src[n]	cfg_rcw_src[0:8] of LS1046A/ LS1043A is mapped to cfg_rcw_src[1:8, 0] of LS1088A. For common board cfg_rcw_*src[n]* should have provision for both pull up/down.		
Cfg_eng_use[0:2]	Mapped to same balls on all three devices. Pull high or low as per input clock selection.		
Cfg_dram_type	Plan on DDR4 Should be pulled low for DDR4 in LS1043A. Pull low in LS1046A and LS1088A		

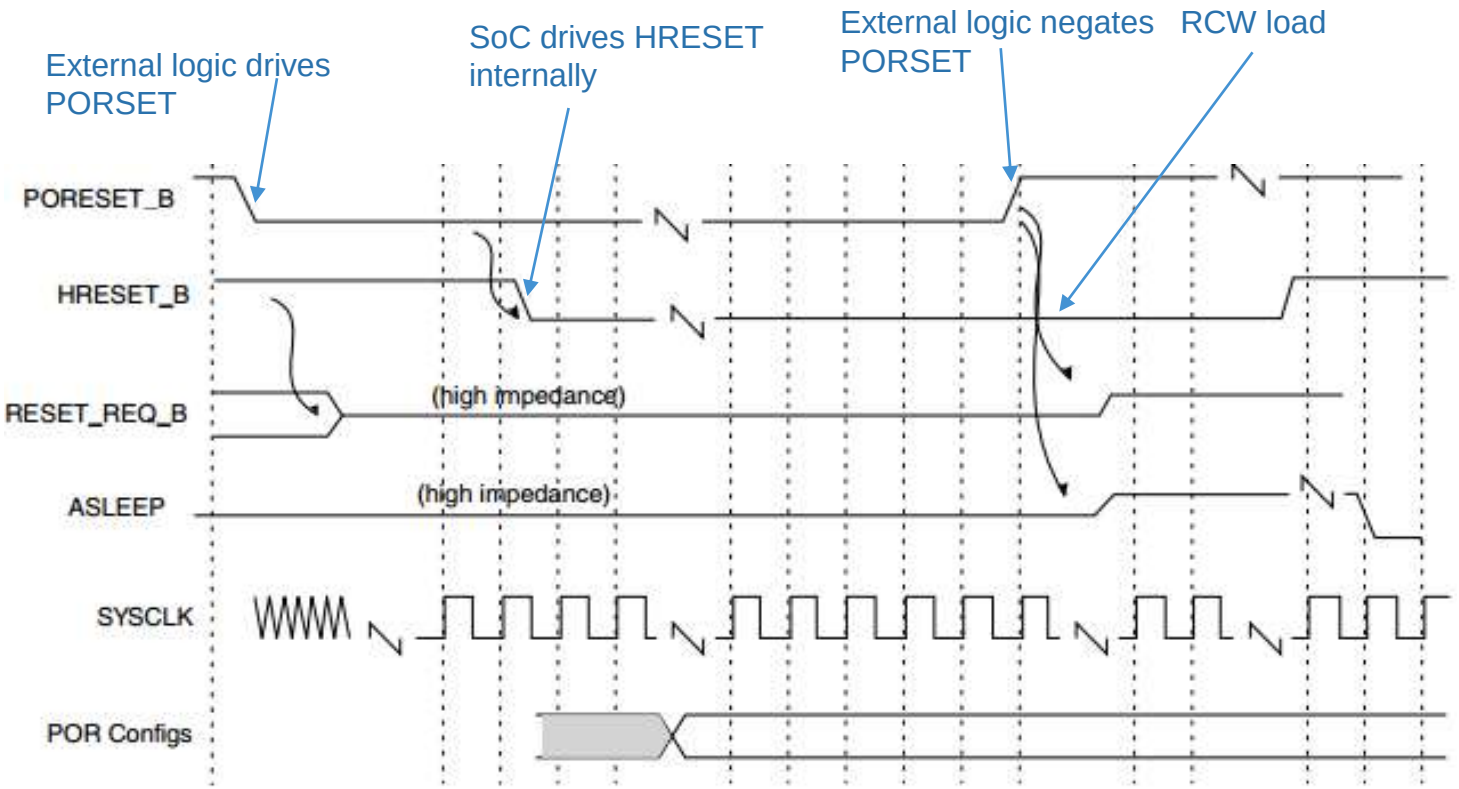
Boot Sources

Boot Sources	LS1046A	LS1043A	LS1088A	Remarks
NOR Flash memory (8 bit)	✓	✓	×	
NOR Flash memory (16 bit)	✓	✓	✓	16 bit NOR flash is not pin compatible across the three devices, it will require a complex on board logic to provision booting.
NAND Flash Memory	×	✓	×	Booting from NAND device is not supported on LS1046A and LS1088A
SD/eMMC	✓	✓	✓	Fully compatible
QSPI	✓	✓	✓	Fully compatible

Since only 8-bit NAND interface is pin compatible, hence no IFC boot mechanism is common across the three devices. It is recommended to use SD/eMMC or QSPI as boot source and storage device on the common board.
However if a common board is designed to support LS1043A and LS1046A only, IFC NOR may be considered.

TIPS AND CONSIDERATIONS TO GET TO A SINGLE BOARD

Power up sequence



DON'T MAKE THE SAME MISTAKE THAT OTHERS MAKE.

COMMON ERROR(S)

- The successful completion of the reset sequence is indicated by the ASLEEP signal being driven low as shown in the timing diagram

- If this does not occur, then there is an issue with the reset sequence – usually with some basic hardware function – and it must be debugged using low-level hardware debug tools and techniques (logic analyzer and oscilloscopes)

Power up Sequence (continued)

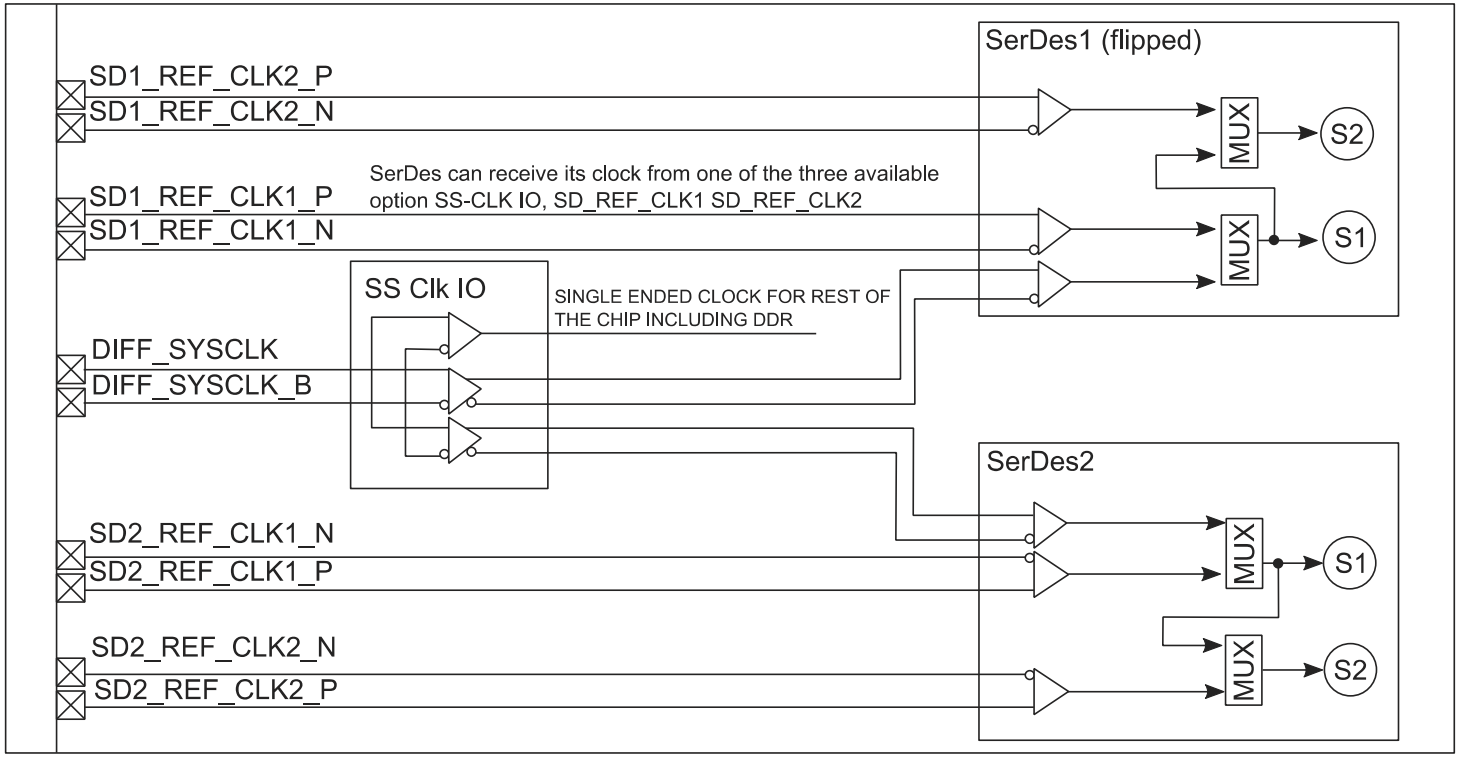
- Things to check if the Power up sequence does not complete include:
  - **Voltage Rails:** Ensure the all the required voltage levels are provided and meet the specified levels and tolerances
    - Ensure that the recommended power rail sequence is followed
  - **SYSCLK:** Ensure it is present and meets the voltage level, slew rate, frequency, duty cycle, and jitter requirements specified
  - **Reset Signals:** Ensure PORESET is driven for a minimum of 1 ms and that it is driven before the core and platform voltages are powered up
    - If HRESET is driven externally, ensure it is released as expected; if driven just by the SoC, confirm it is released after PORESET desertion
  - Confirm the **RCW device is being read** after ASLEEP is driven high
    - If not, check that the cfg_rcw_src signals are driven as expected when the PORESET signal is released
  - **Confirm RCW contents are as expected**. The specifics of the RCW must match the system configuration
  - If the RCW device is blank, a tool such as CodeWarrior must be used to program this. Instructions for doing this are provided in later in another session. However, it is recommended to confirm the hardware operation as much as possible before connecting this tool. Confirming that the LS1043A at least attempts to read the RCW device is a good checkpoint
  - Also good to have a mechanism like a switch, that can stop external logic to assert PORESET on assertion of RESET_REQ by SoC

DON'T MAKE THE SAME MISTAKE THAT OTHERS MAKE.

COMMON ERROR(S)

Clocking Differences

Clocking - Single source clocking scheme as in LS1088A



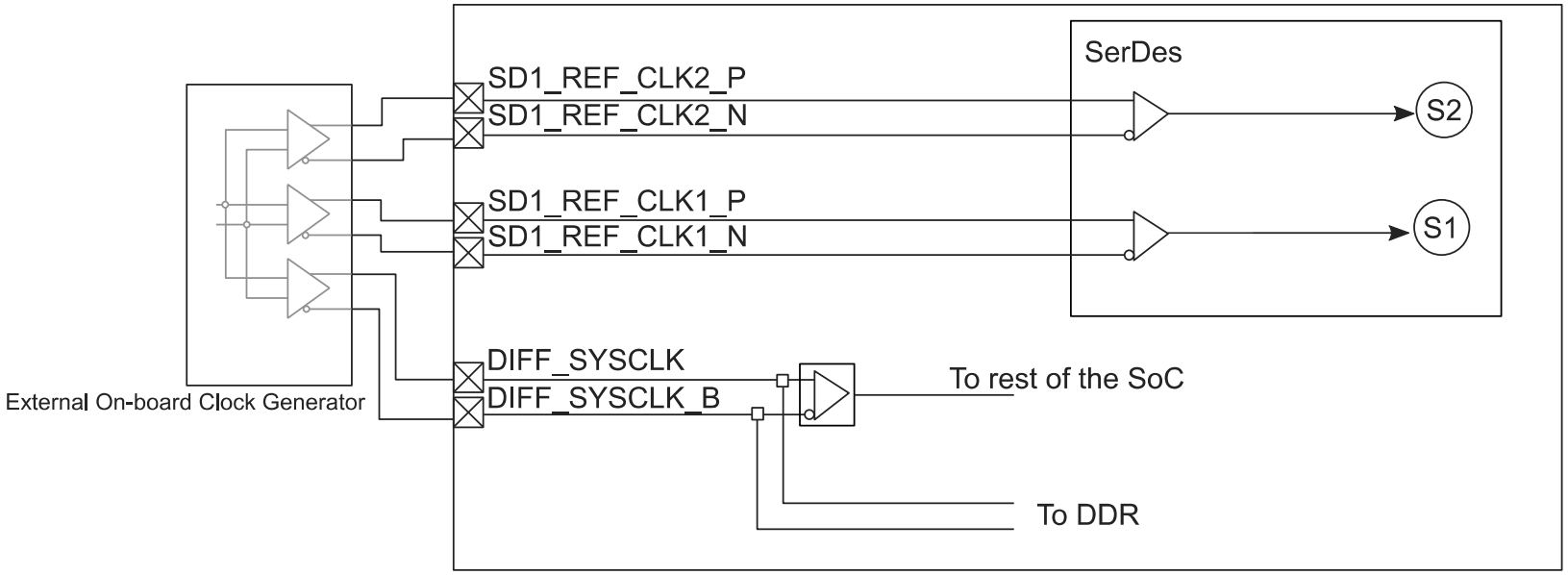
Single clocking scheme in LS1088A

The Differential SYSCLK clock input can feed Core PLLs, Platform PLL, DDR PLL and USB PLL .

Additionally it can be used to feed the four SerDes PLL, selectable option through RCW bits 952, 953, 954 and 955

TIPS AND CONSIDERATIONS TO GET TO A SINGLE BOARD

Clocking - Single source clocking scheme as in LS1043A

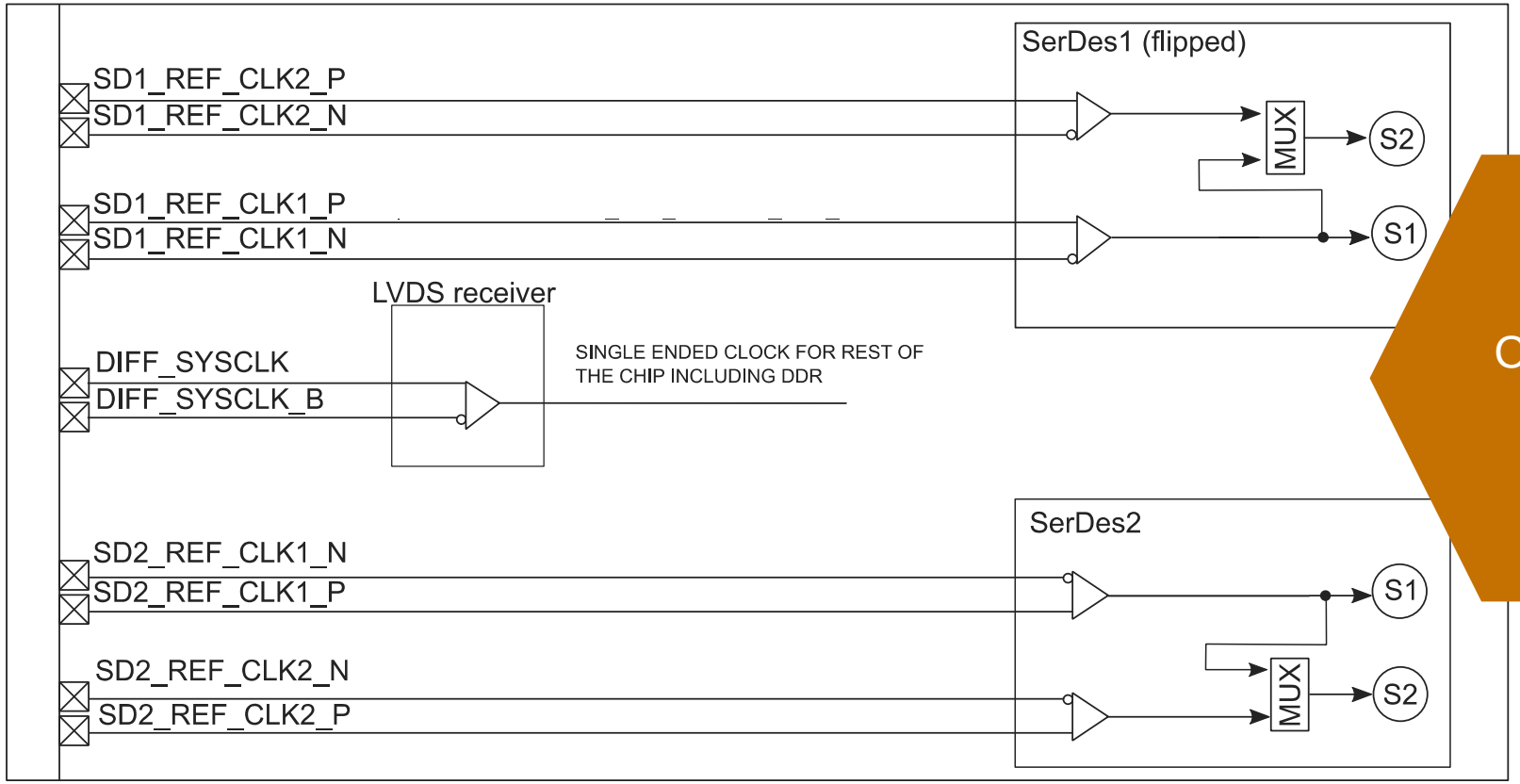


Single clocking scheme in LS1043A

The Differential SYSCLK clock input can feed Core PLLs, Platform PLL, DDR PLL and USB PLL .
The two SerDes PLL require individual clock inputs from board.

TIPS AND CONSIDERATIONS TO GET TO A SINGLE BOARD

Clocking - Single source clocking scheme as in LS1046A



Single clocking scheme in LS1046A

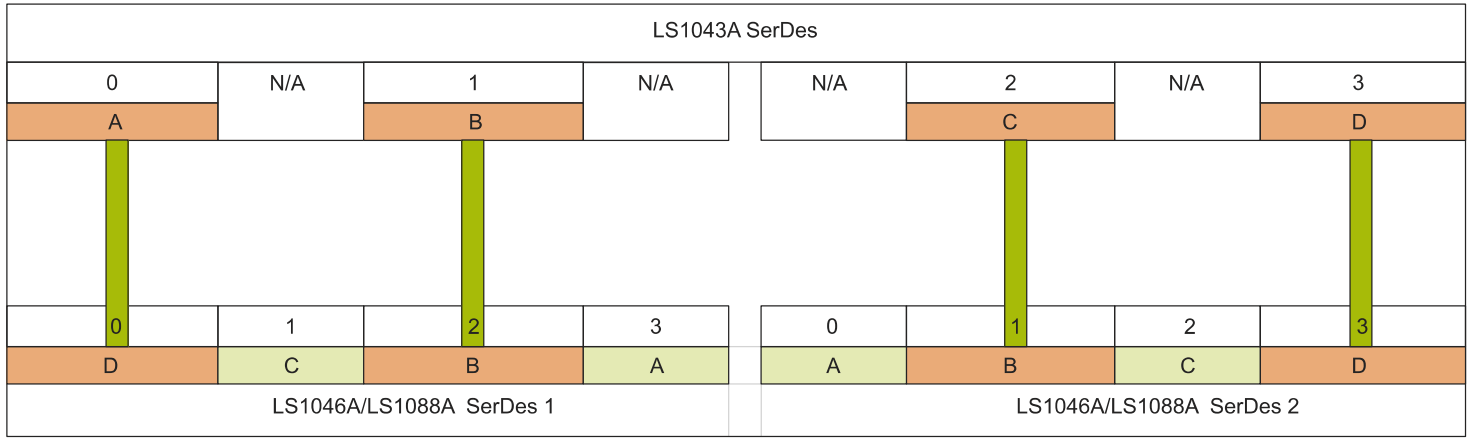
TIPS AND CONSIDERATIONS TO GET TO A SINGLE BOARD

The Differential SYSCLK clock input can feed Core PLLs, Platform PLL, DDR PLL and USB PLL.
SerDes PLL1 clock input can be used to clock the SerDes PLL2, selectable through RCW bits.

SerDES

SerDes

SerDes Lane mapping



SerDes Clock mapping

LS1043A	NC	SD1_REF_CLK1	NC	SD1_REF_CLK2
LS1046A/ LS1088A	SD1_REF_CLK1	SD1_REF_CLK2	SD2_REF_CLK1	SD2_REF_CLK2

- LS1046A and LS1088A SerDes are pin compatible
- Connect the SerDes reference clocks as per the requirements of chosen SerDes protocol for common board
- Terminate unused SerDes lanes and reference clocks as per the guidelines in design checklist
- SD2_IMP_CAL_RX and SD1_IMP_CAL_TX pins in LS1046A/LS1088A are mapped to NC pins in LS1043A. They should be terminated as per the guidelines in LS1046A/LS1088A datasheet
- SD1_PLL2_TPA/TPD and SD2_PLL1_TPA/TPD pins in LS1046A/ LS1088A are mapped to NC pins in LS1043A. They can be left as NC or terminated to a test point

Tools

If you use CodeWarrior…

You'll be using the same tools we do

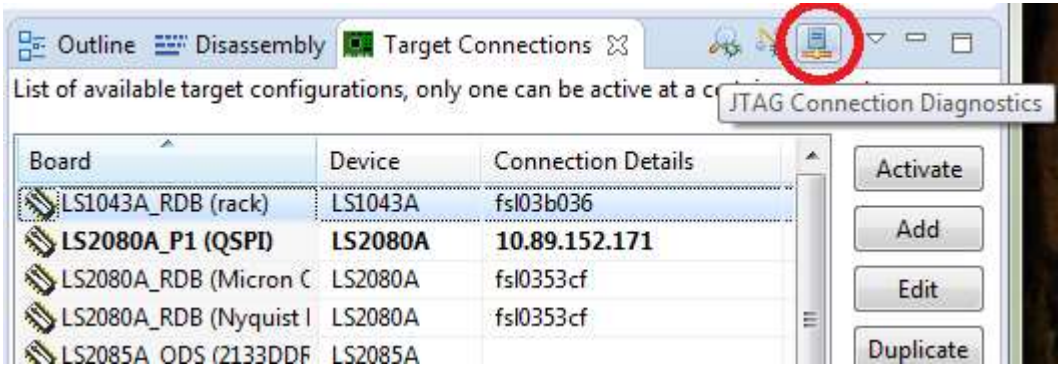
It is just one less thing to deal with.

CodeWarrior Debug

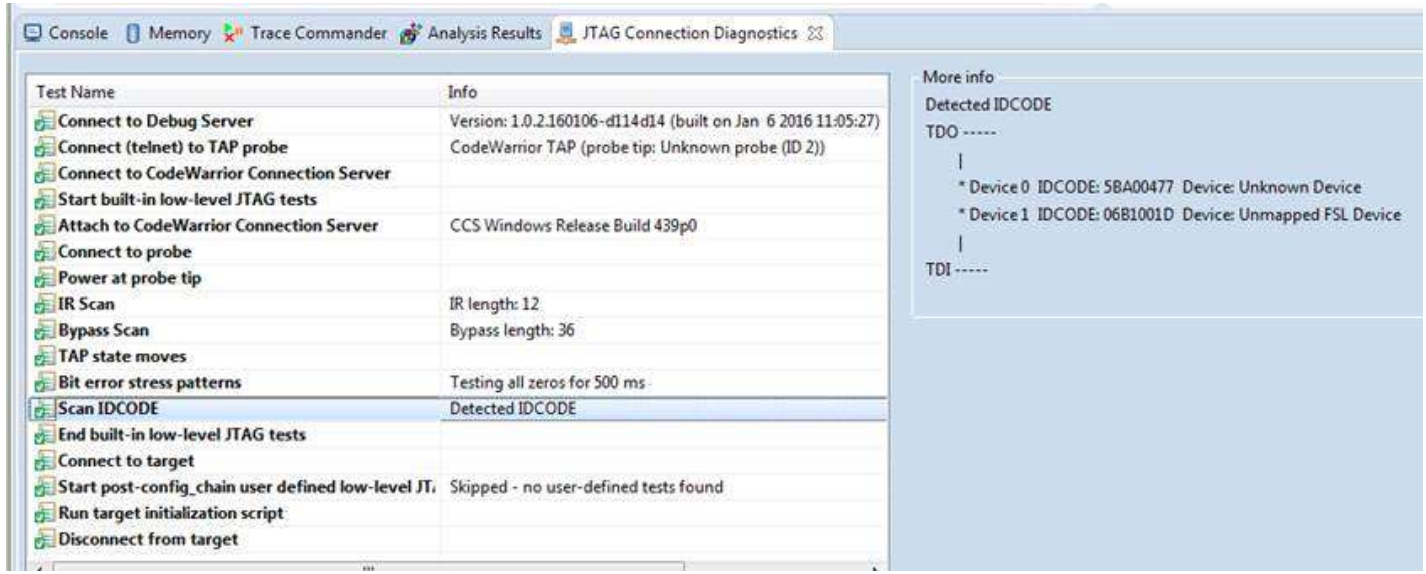
- Checking your SoC using CCS

- Board must be powered with CodeWarriorTAP connected

- In CW4NET ARMv8 v2016.01, open the Target Connections view, select the Target Connection from the menu (“LS1043A_RDB (rack)”, in the example below) and click on ‘JTAG Connection Diagnostics’.



CodeWarrior Debug..contd..



- Clicking on any line of the report will display more information, if available. “Scan IDCODE” shown here.

- This is the recommended method for running JTAG diagnostics in CW ARMv8.

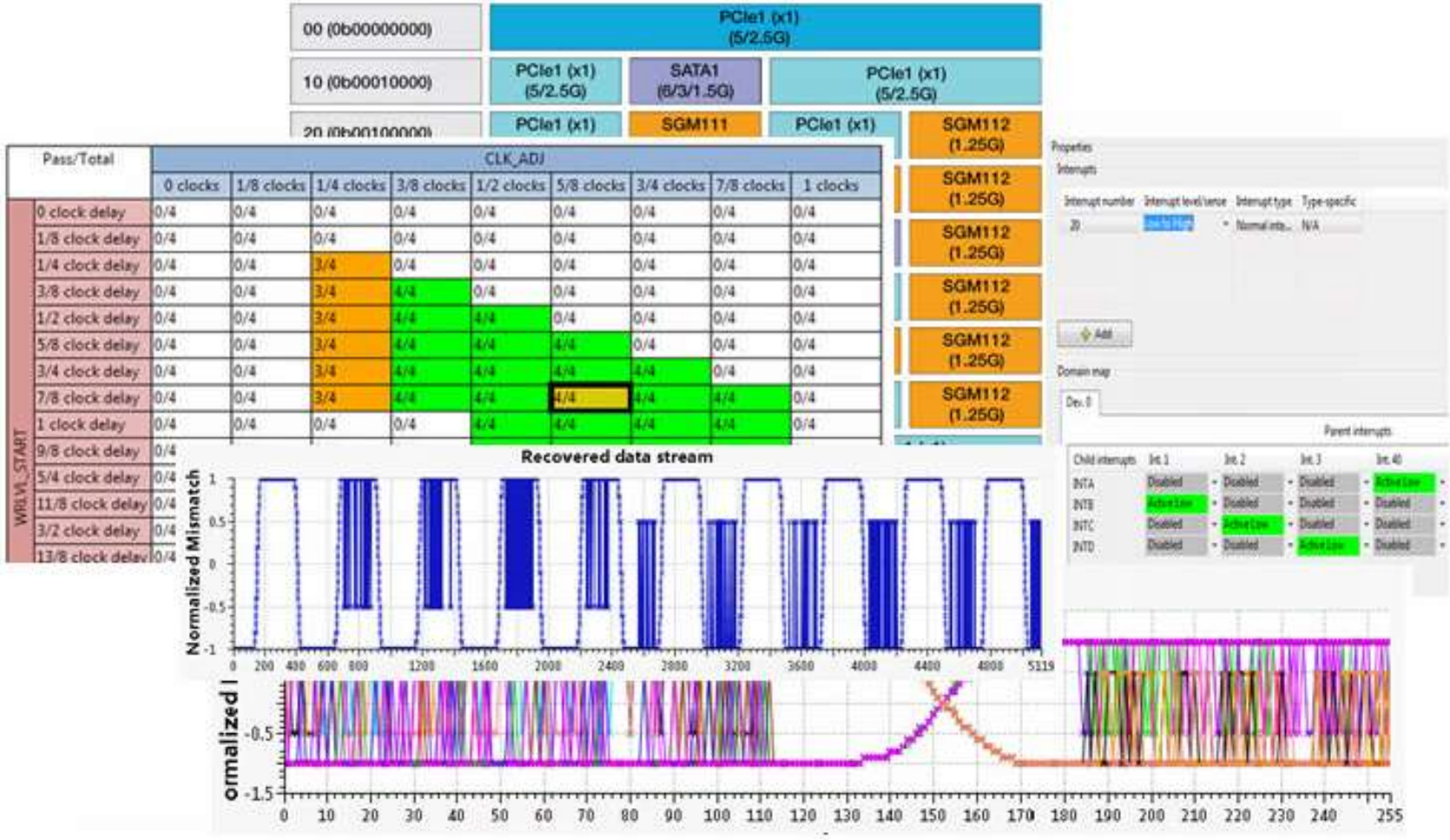
CodeWarrior Debug contd..

- Create a RCW override file using information in the video link below
  - http://www.nxp.com/video/how-to-create-a-rcw-override-configuration-file-for-use-in-a-CodeWarrior-qoriq-debugger:RCW-QORIQ-DEBUGGER
- Use the RCW override file to override the RCW using CodeWarrior
  - http://www.nxp.com/video/how-to-use-qoriq-rcw-override-feature-in-codewarrior:HOW-TO-USE-QORIQ-RCW-CODEWARRIOR
- More details can be found in the User guide for CodeWarrior ARM v8
  - http://cache.nxp.com/files/soft_dev_tools/doc/user_guide/CWARMv8TM.pdf?fpsp=1&WT_TYPE=Users%20Guides&WT_VENDOR=FREESCALE&WT_FILE_FORMAT=pdf&WT_ASSET=Documentation&fileExt=.pdf

DDRv

- Allows configuration of DDR controller memory mapped (in CCSR) registers
- View DDR controller memory mapped registers on a bit field level
- Can read DDR configuration from various sources (memory dump, DIMM's SPD, directly from target [in the making…])
- Generates DDR initialization code in various formats: uBoot data structure, plain C code, GDB script
- Exports DDR registers configuration dump to various formats
- Same advantages as PBL tool
  - Same user experience across processors
  - No need for reference manual
  - Incorporates errata
  - Validates input against known constraints
- DDR validation should be done before SDK phase so as to plug in the right DDR register settings for optimum and stable performance of DDR. Performing DDRv will eliminate border cases where certain boards work while others do not

Processor Expert Sotfware: QorIQ and Validation Suite



http://www.nxp.com/products/software-and-tools/software-development-tools/codewarrior-development-tools/suite-for-networked-applications/qoriq-configuration-and-validation-suite:PE_QORIQ_SUITE#

DDR Controller

DDR SDRAM Byte lane	LS1046A LS1088A	LS1043A
Byte lane 0	MDQ[0:7]	MECC[0:3] + 4x NC
Byte lane 1	MDQ[8:15]	MDQ[24:31]
Byte lane 2	MDQ[16:23]	MDQ[16:23]
Byte lane 3	MDQ[24:31]	MDQ[8:15]
Byte lane 4	MDQ[32:39]	8x NC
Byte lane 5	MDQ[40:47]	8x NC
Byte lane 6	MDQ[48:55]	8x NC
Byte lane 7	MDQ[56:63]	8x NC
ECC lane	MECC[0:7]	MDQ[0:7]

- LS1043A supports a 32-bit DDR3L/DDR4
- LS1046A/LS1088A support a 64-bit DDR4
- 32-bit DDR4 w/ECC memory is common

Note that one of the byte lane of LS1043A is connected to ECC lane of memory. If user chooses to not use ECC, one of the byte lanes will be unused. Unpopulated byte lane will create stub which can affect signal integrity.
User should either use 32-bit DDR4 with ECC OR do a thorough simulation to analyze affects of stubs

- References:
  - AN5226
  - AN5097

TIPS AND CONSIDERATIONS TO GET TO A SINGLE BOARD

SDK phase

Suggested sessions: FTF-DES-N1851,N1834,N1843

Docs!!!!!

DON’T MAKE THE SAME MISTAKE THAT OTHERS MAKE.

COMMON ERROR(S)

- Extensive documentation online: nxp.com/linux
- Even finer grained docs: http://www.nxp.com/products/developer-resources/run-time-software/linux-sdk/layerscape-software-development-kit:LAYERSCAPE-SDK?tab=Documentation_Tab
- (sorry for the long URL)

U-boot environment settings and memory map

```
=> print
baudrate=115200
bootargs=console=ttyS0,115200 root=/dev/ram0 earlycon=uart8250,0x21c0500,115200
bootcmd=cp.b $kernel_start $kernel_load $kernel_size && bootm $kernel_load
bootdelay=3
console=ttyAMA0,38400n8
eth1addr=00:e0:0c:00:77:01
eth2addr=00:e0:0c:00:77:02
eth3addr=00:e0:0c:00:77:03
eth4addr=00:e0:0c:00:77:04
eth5addr=00:e0:0c:00:77:05
eth6addr=00:e0:0c:00:77:06
ethact=FM1@DTSEC1
ethaddr=00:e0:0c:00:77:00
ethprime=e1000#0
fman_ucode=60300000
gatewayip=192.168.1.1
hwconfig=fsl_ddr:bank_intlv=auto
ipaddr=192.168.1.100
kernel_addr=0x100000
kernel_load=0xa0000000
kernel_size=0x2800000
kernel_start=0x61100000
loadaddr=0x80100000
ramdisk_addr=0x800000
ramdisk_size=0x2000000
serverip=192.168.1.1
stderr=serial
stdin=serial
stdout=serial
```

QUICK REF!

Start address	End address	Image	Max size
0x60000000	0x6001FFFF	RCW (current bank)	128KB
0x60100000	0x6017FFFF	u-boot (current bank)	512KB
0x60300000	0x6031FFFF	Fman ucode (current bank)	128KB
0x61100000	0x6203FFFF	Linux.uImage, Device Tree and Ramdisk (current bank) FIT Image	15MB
0x64000000	0x6401FFFF	RCW (alt bank)	128KB
0x64100000	0x6417FFFF	u-boot (alt bank)	512KB
0x64300000	0x6431FFFF	Fman ucode (alt bank)	128KB
0x65100000	0x6603FFFF	Linux.uImage, Device Tree and Ramdisk (alt bank) FIT Image	15MB

Memory Map

Start address	End address	Image	Max size
0x60000000	0x6001FFFF	RCW (current bank)	128KB
0x60100000	0x6017FFFF	u-boot (current bank)	512KB
0x60300000	0x6031FFFF	Fman ucode (current bank)	128KB
0x61100000	0x6203FFFF	Linux.uImage, Device Tree and Ramdisk (current bank) FIT Image	15MB
0x64000000	0x6401FFFF	RCW (alt bank)	128KB
0x64100000	0x6417FFFF	u-boot (alt bank)	512KB
0x64300000	0x6431FFFF	Fman ucode (alt bank)	128KB
0x65100000	0x6603FFFF	Linux.uImage, Device Tree and Ramdisk (alt bank) FIT Image	15MB

- Two ways to transfer images to the board
  - CodeWarrior
    - If you have a blank board
  - TFTP
    - Assume working U-Boot in current bank
    - Flash to altbank from current bank

QUICK REF!

Transferring Images to the board: TFTP

- TFTP can be used to flash images to alternate bank from current bank
- Programming a new U-Boot

  =>tftp 82000000 <u-boot_file_name>.bin

  =>protect off 64100000 +$filesize

  =>erase 64100000 +$filesize

  =>cp.b 82000000 64100000 $filesize

  =>protect on 64100000 +$filesize

  =>cpld reset altbank
- Programming a new RCW

  =>tftp 82000000 <rcw_file_name>.bin

  =>protect off 64000000 +$filesize

  =>erase 64000000 +$filesize

  =>cp.b 82000000 64000000 $filesize

  =>protect on 64000000 +$filesize

  =>cpld reset altbank
- Programming a new FMan ucode

  =>tftp 82000000 <fman_ucode_file_name>.bin

  =>protect off 64300000 +$filesize

  =>erase 64300000 +$filesize

  =>cp.b 82000000 64300000 $filesize

  =>protect on 64300000 +$filesize

  =>cpld reset altbank

QUICK REF!

U-Boot: Ethernet and PCI Sanity Checks

```
=> mdio list
FSL_MDIO0:
1 - RealTek RTL8211F <--> FM1@DTSEC3
2 - RealTek RTL8211F <--> FM1@DTSEC4
4 - Vitesse VSC8514 <--> FM1@DTSEC1
5 - Vitesse VSC8514 <--> FM1@DTSEC2
6 - Vitesse VSC8514 <--> FM1@DTSEC5
7 - Vitesse VSC8514 <--> FM1@DTSEC6
FM_TGEC_MDIO:
1 - Aquantia AQR105 <--> FM1@TGEC1
```

QUICK REF!

PCIe Information

```
=> pci 0
Scanning PCI devices on bus 0
BusDevFun  VendorId   DeviceId   Device Class       Sub-Class
_____________________________________________________________
00.00.00   0x1957     0x8084     Bridge device           0x04
=> pci 1
Scanning PCI devices on bus 1
BusDevFun  VendorId   DeviceId   Device Class       Sub-Class
_____________________________________________________________
01.00.00   0x8086     0x10d3     Network controller      0x00
=>
```

Linux Images

- Options for loading Linux images onto board
  - Use TFTP from PC
  - Use Serial Port from PC

QUICK REF!

Using TFTP to Load Linux Images and Boot

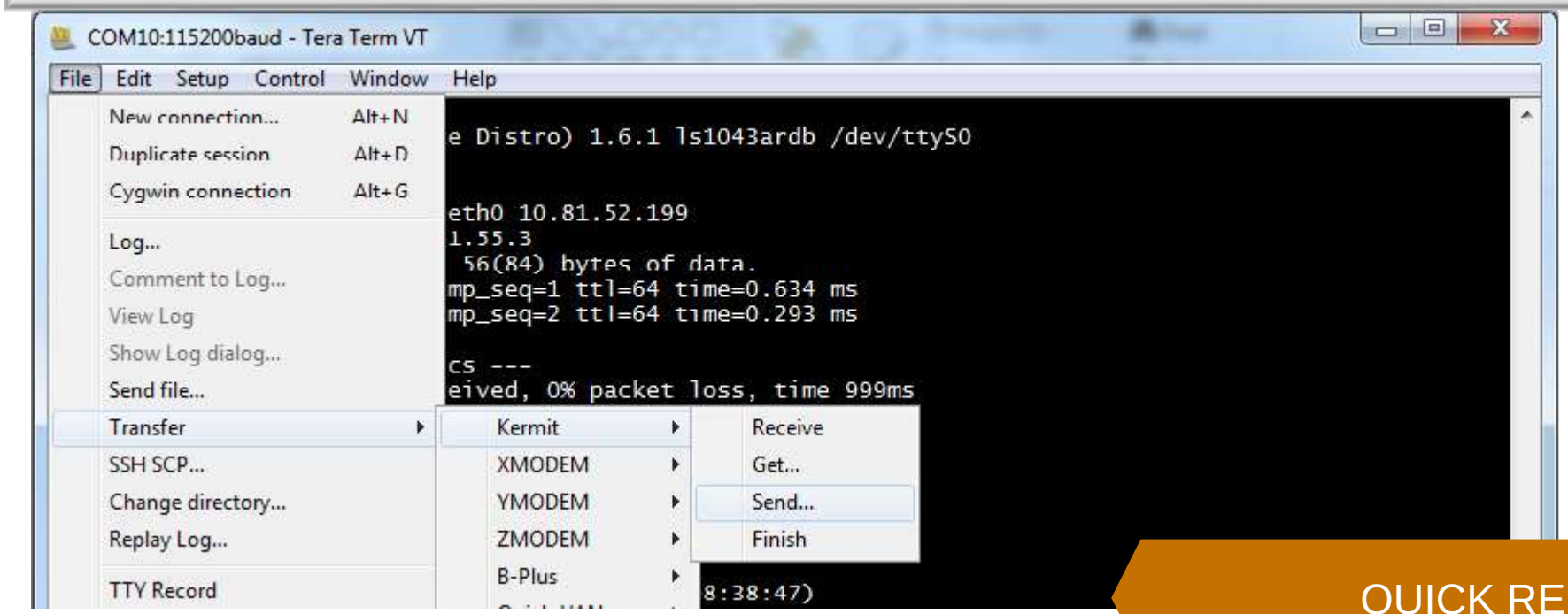
- Set up TFTP server on your PC

QUICK REF!

- Make sure IP Address of PC is same as serverip set previously
- Ping serverip to make sure a live link is available
  - =>ping $serverip
- TFTP itb image and flash it to the NOR flash vBank0 using following commands
  - =>tftp a0000000 kernel-ls1043ardb.itb
  - =>erase 61100000 +$filesize
  - =>cp.b a0000000 61100000 $filesize
- Use command: “boot” to bring up linux

Using Serial Port to Load Linux Images

```
=> loadb
## Ready for binary (kermit) download to 0x80100000 at 115200 bps...
## Total Size      = 0x00000088 = 136 Bytes
## Start Addr      = 0x80100000
=> md 0x80100000
80100000: 55aa55aa 0001ee01 08000b08 0000000a    .U.U............
80100010: 00000000 00000000 02005514 02400080    .........U....@.
80100020: 005002e0 002400c1 00000000 00000000    ..P...$.........
80100030: 00000000 00880300 00000000 00110000    ................
=>
```

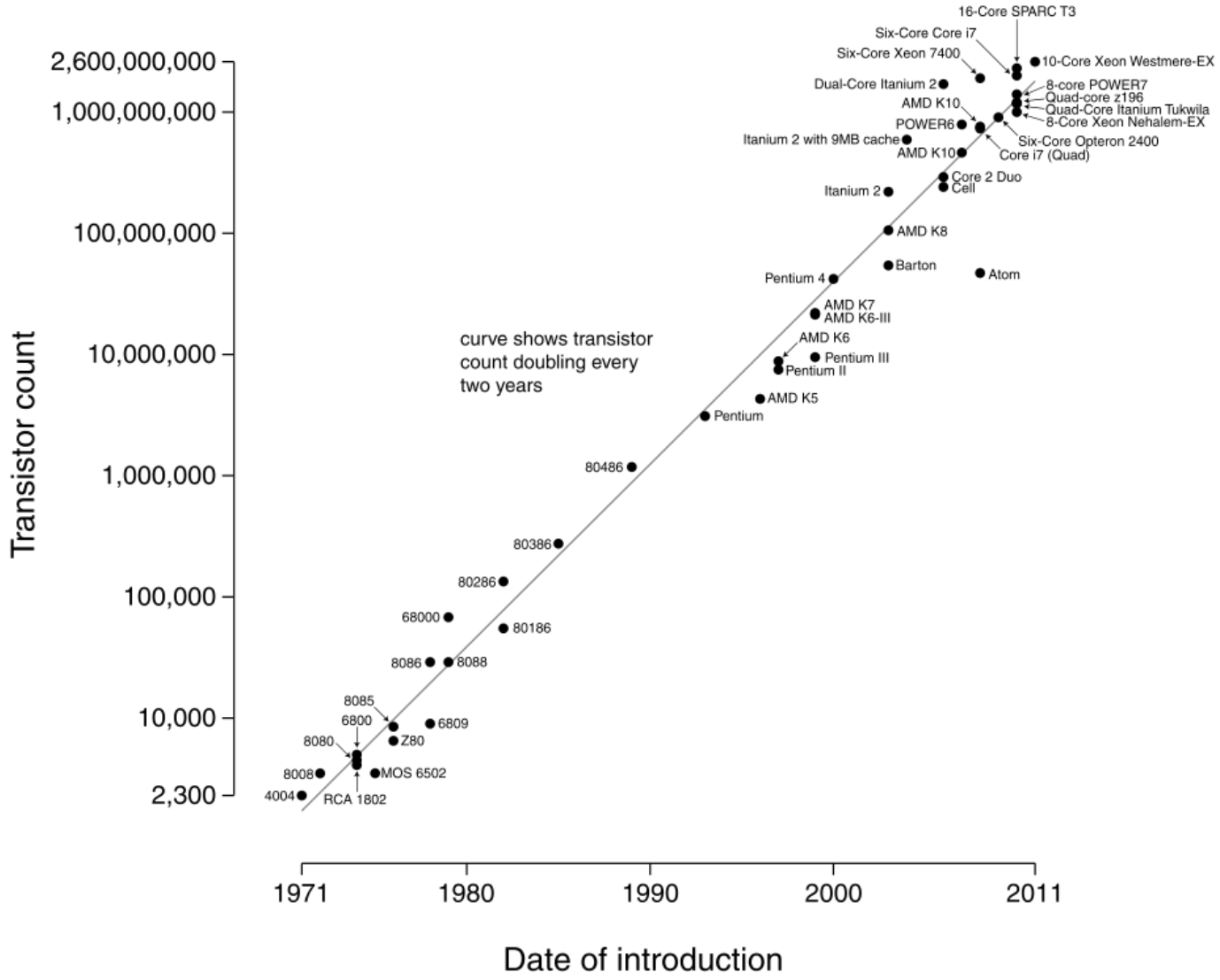


QUICK REF!

03.

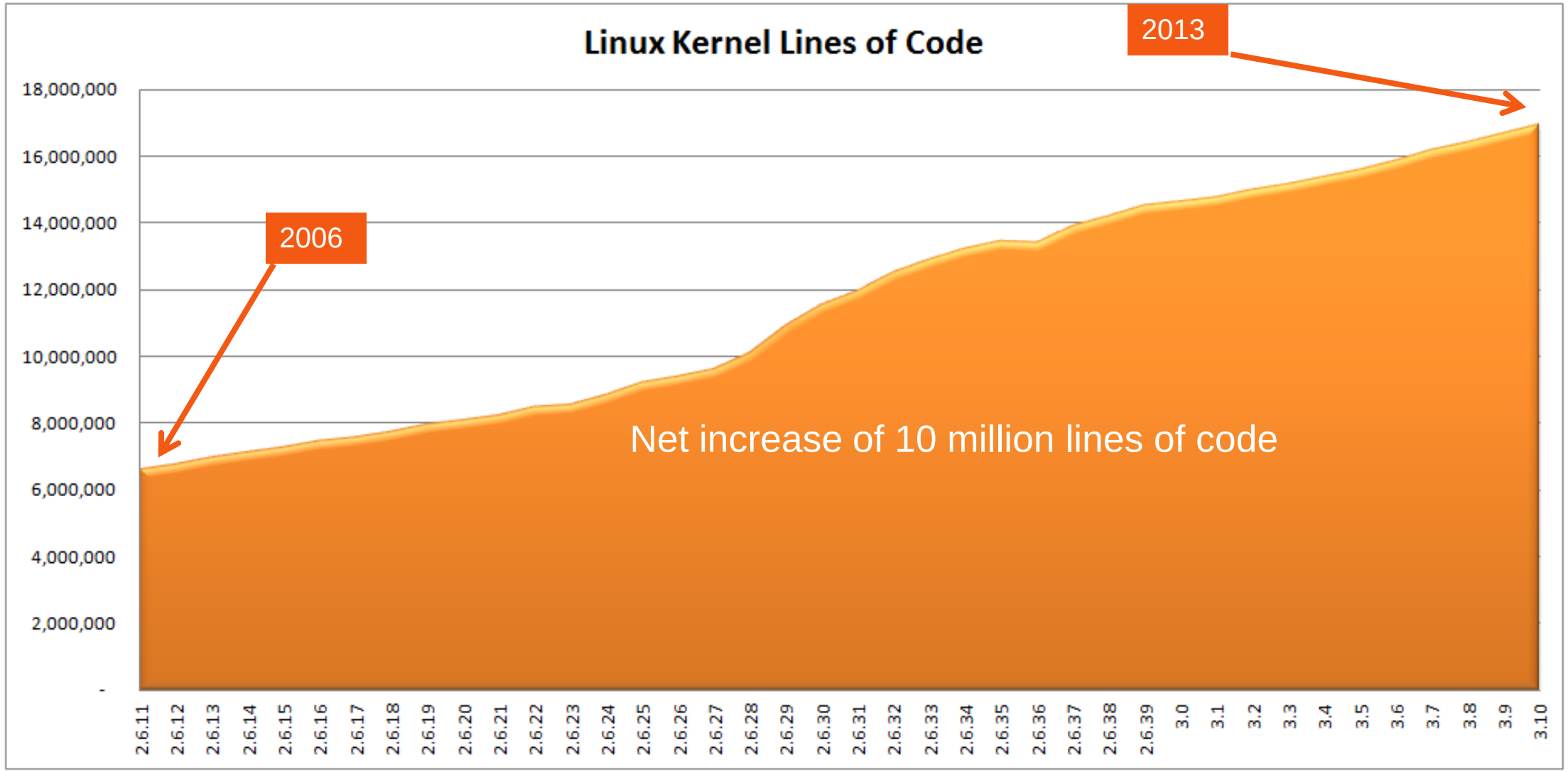
Worthwhile projects are too big to “go it alone”

More functions in a small space

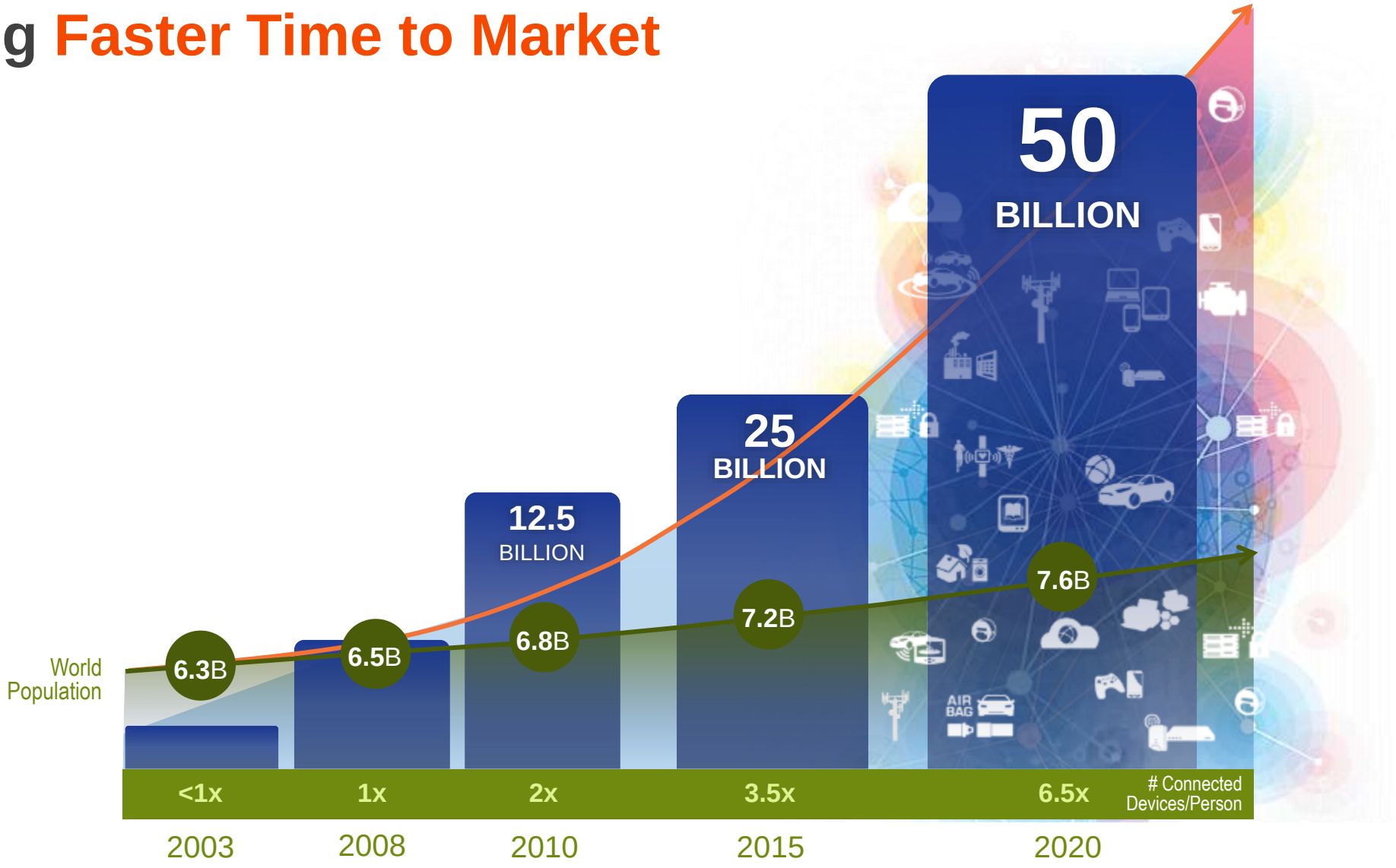


Source: Wikipedia

More code


Linux Kernel Lines of Code
2013
2006
Net increase of 10 million lines of code
18,000,000
16,000,000
14,000,000
12,000,000
10,000,000
8,000,000
6,000,000
4,000,000
2,000,000
-
2.6.11
2.6.12
2.6.13
2.6.14
2.6.15
2.6.16
2.6.17
2.6.18
2.6.19
2.6.20
2.6.21
2.6.22
2.6.23
2.6.24
2.6.25
2.6.26
2.6.27
2.6.28
2.6.29
2.6.30
2.6.31
2.6.32
2.6.33
2.6.34
2.6.35
2.6.36
2.6.37
2.6.38
2.6.39
3.0
3.1
3.2
3.3
3.4
3.5
3.6
3.7
3.8
3.9
3.10

The Explosive Growth In Connected Devices – Requiring Faster Time to Market



* Sources: Ericsson, February 2011; Cisco Internet Business Solutions Group (IBSG), April 2011

All this complexity…

- A typical highly integrated SoC can take several engineers with expertise in different disciplines to support fully
  - Communications for packet management
  - Cryptography and communications for protected communications
  - Trust & cryptography for non-communications usages.
  - DDR memory expert…
  - Etc.

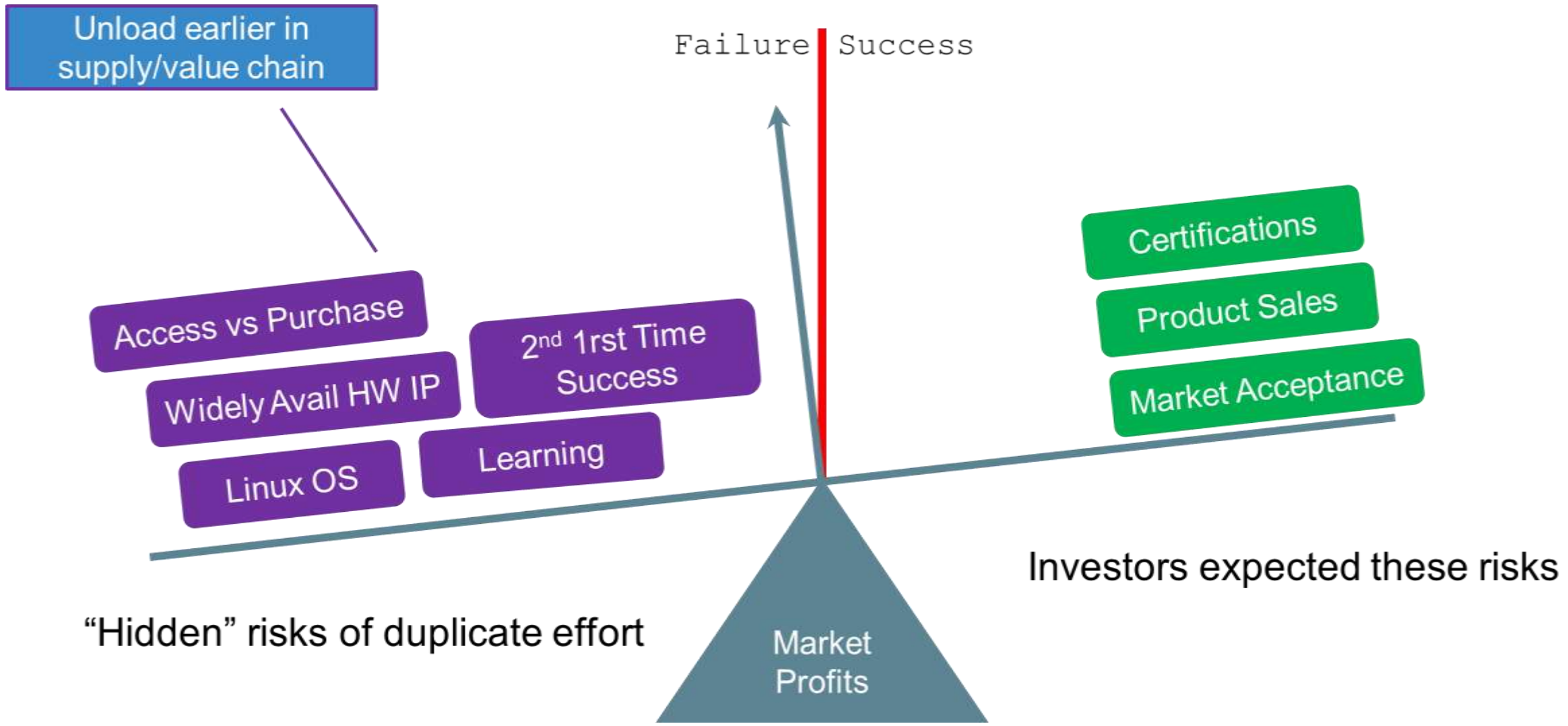
- Now, put SW (many different proprietary and open source packages to choose from for nearly any small section of an SoC

- Now make the SW take full advantage of all the goodness of the IC.

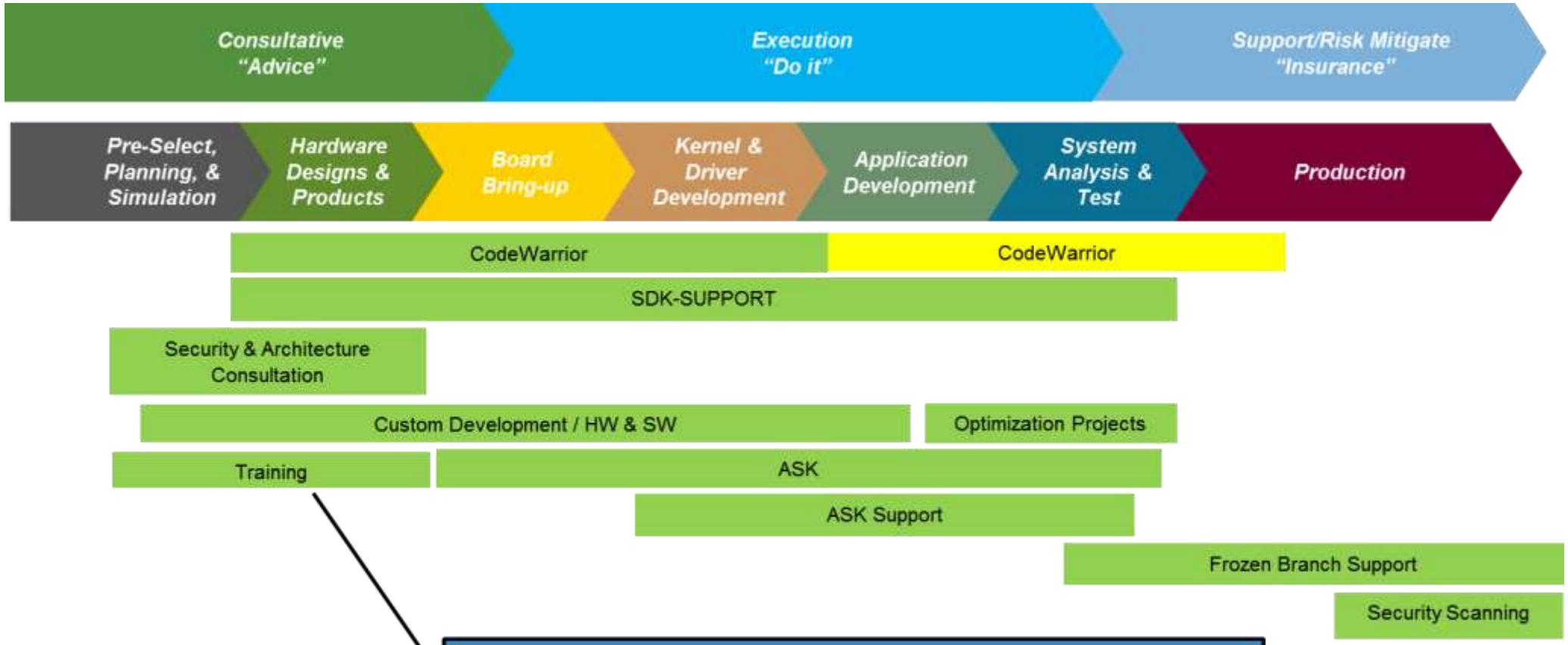
- Remember… there are a 100 companies world wide that want to solve the same problem as you.

- Oh yea, didn't you need to make a differentiated application?

Risk – What risks are you PAID to take versus PAID to avoid?

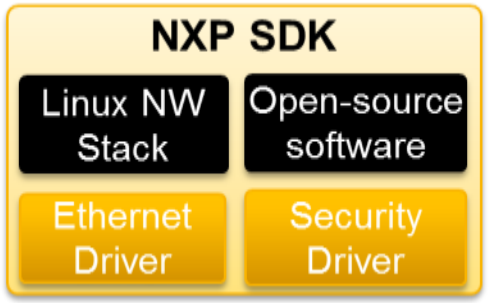


Services, SW & Tools have the most impact at different times in the product lifecycle

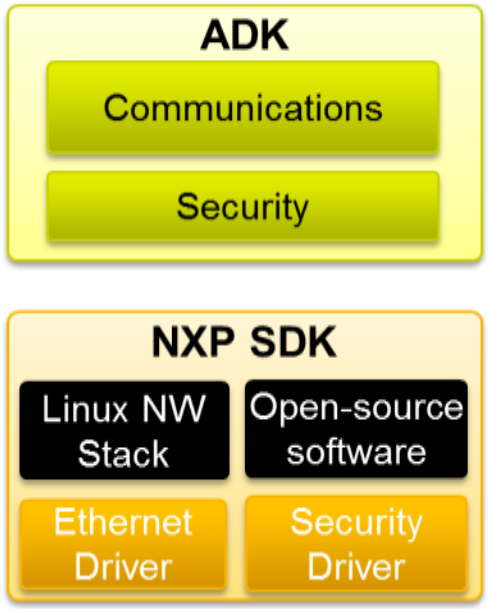


NXP QorIQ Offers Differing Levels of Software APIs

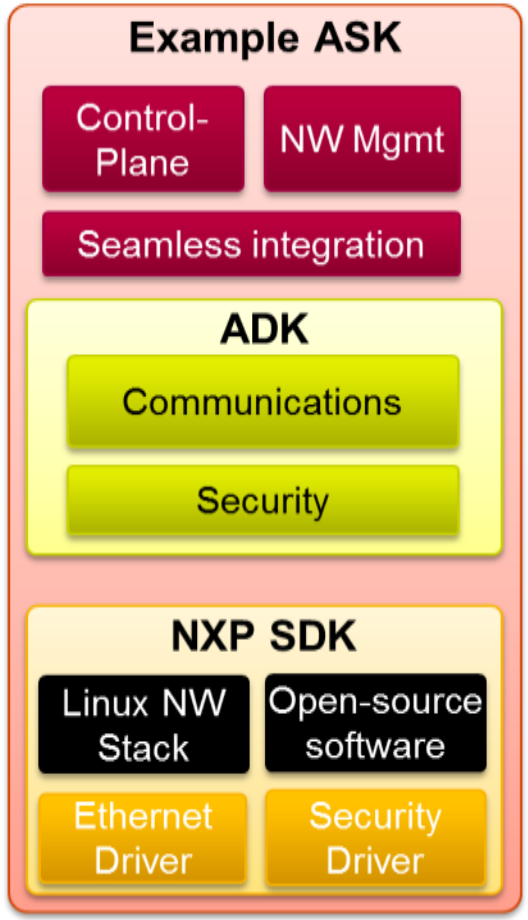
SDK:
- Optimized drivers
- Open-source Linux stacks
- Competitive performance
- Flexibility to add own applications



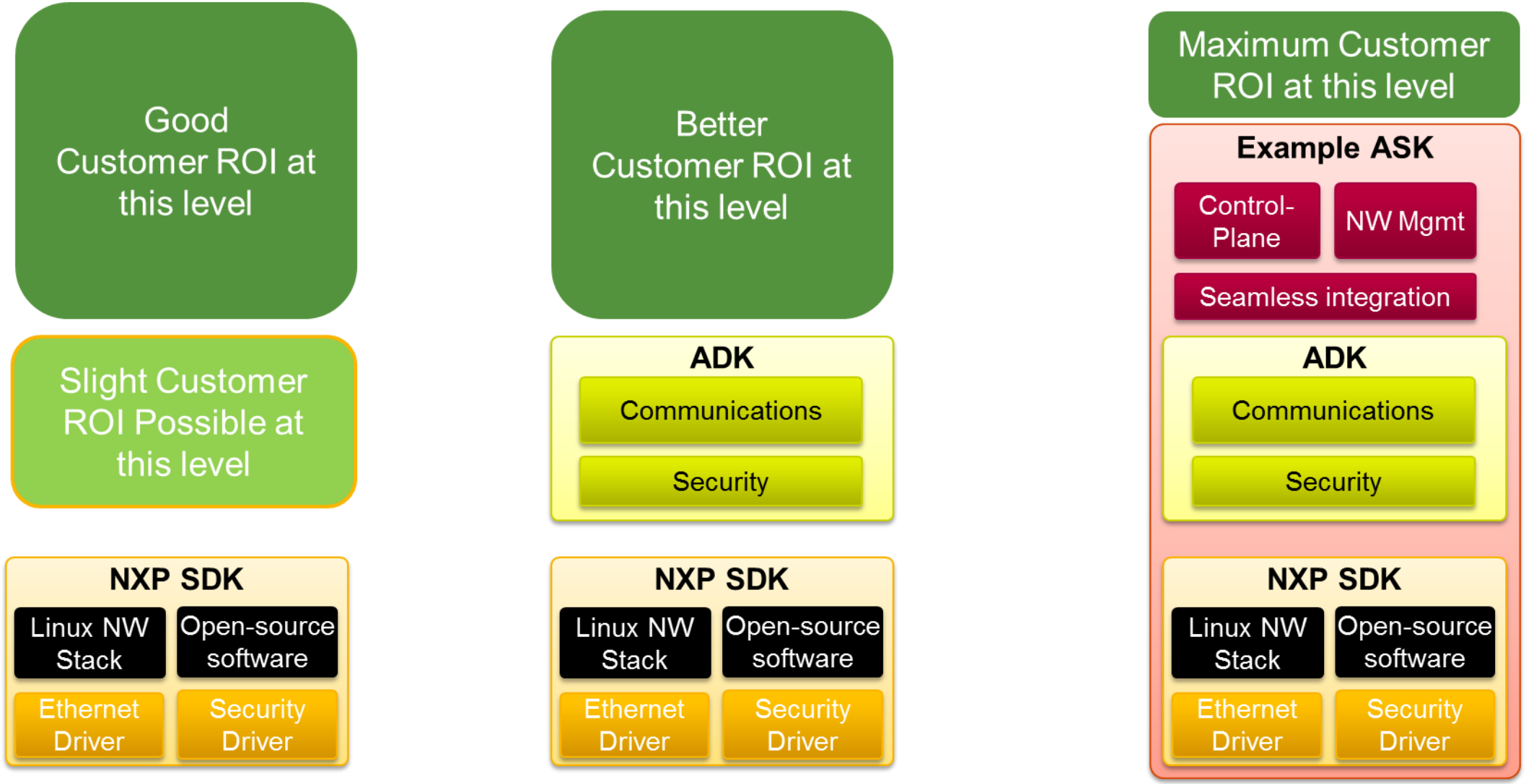
SDK + ADK:
- 4x-8x performance increase
- More headroom in CPU
- Flexibility to integrate own applications
- More features than open-source SW



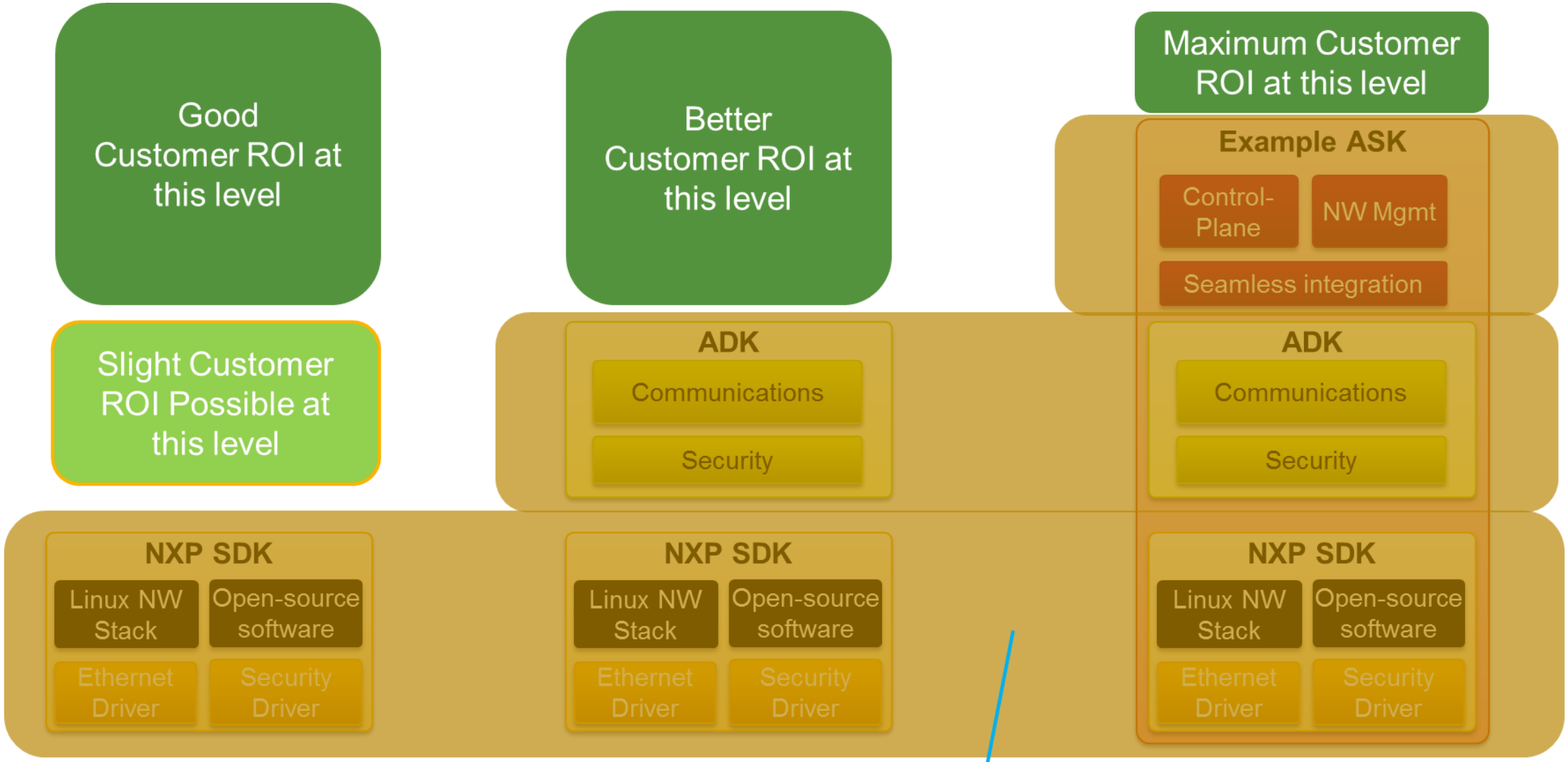
ASK = SDK + ADK + more:
- All the 4x-8x performance increase +
- Full turnkey solution HW & SW
- Fastest time to market



Customers interface to APIs in a way to reduce duplication



Big Projects Have a clean ROI-minded division of labor



QorIQ/LS Software and Services Group

Software Products and Custom Services

Development Tools	Runtime Products	Solutions Reference	Linux® Services	Integration Services
• CodeWarrior – IDE – Debug – Compiler – Trace – Scenarios Tools – DDrV – AIOP Analyzer	• VortiQa Software Products – Application Identification Software (AIS) – SDN Solutions – Mobile Transport – Layer 1 • Application Solution Kits (ASK) – NAS, BHR • Application Development Kits (ADK) • Network Services Switch	• Storage Controller • SDN Switch • Wireless LAN • Data Concentrator • Smart Converged Gateway • Digital Signage • Proof of Concept • OpenWRT+	• Commercial Support • Frozen Branch • Application-Specific Hardening • Feature Acceleration • Forward Port/Kernel Refresh • WiFi • Performance	• Systems Consulting • Discovery Contract • Security (All Levels) • Design Services • Porting • Emergency Boot • Uncommon IP • Functional Simulators
nxp.com/**cw4net**	nxp.com/**vortiqa**	nxp.com/**solutions-reference**	nxp.com/**networking-services**	

CodeWarrior QorIQ

VortiQa

Best ”Success Package”

Software Products and Custom Services

Development Tools	Runtime Products	Solutions Reference	Linux® Services	Integration Services
• CodeWarrior – IDE – Debug – Compiler – Trace – Scenarios Tools – DDrV – SerDes V	• VortiQa Software Products – Application Identification Software (AIS) – SDN Solutions – Mobile Transport – Layer 1 • Application Solution Kits (ASK) – NAS, BHR • Application Development Kits (ADK) • Network Services Switch	• Storage Controller • SDN Switch • Wireless LAN • Data Concentrator • Smart Converged Gateway • Digital Signage • Proof of Concept • OpenWRT+	• Commercial Support • Frozen Branch • Application-Specific Hardening • Feature Acceleration • Forward Port/Kernel Refresh • WiFi • Performance	• Systems Consulting • Discovery Contract • Security (All Levels) • Design Services • Porting • Emergency Boot • Uncommon IP • Functional Simulators
nxp.com/**cw4net**	nxp.com/**vortiqa**	nxp.com/**solutions-reference**	nxp.com/**networking-services**	

CodeWarrior QorIQ

VortiQa

SECURE CONNECTIONS
FOR A SMARTER WORLD