S32K MICROCONTROLLER PLATFORM LAUNCH: ACCELERATING AUTOMOTIVE SOFTWARE DESIGN

Mike Cao 曹学余

GC AUTOMOTIVE FAE

XUEYU.CAO@NXP.COM

+86 18616552690

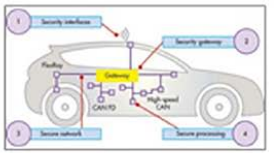
Agenda

- S32K MCU Family Overview
- Key features
  - CAN-FD
  - Security
  - Safety
  - Low power
- Development tools
- Applications
- Availability

OVERVIEW

NXP AMP Product Lines

C&S
(Connectivity & Security)



Gateway

- #1 in Vehicle Networking with leading networking and security IP
- #1 in Automotive HW Security with Strong IP and broad portfolio
- End to end portfolio of networking devices (MCU/MPU, TX/RX)

Products:
MPC564xB/C
MPC574xG

ADAS
(Advanced Driver Assistance Systems)

**Radar, LIDAR
Vision
Sensor Fusion**

- #1 in Radar with strong IP and system knowledge
- High performance low power accelerators
- Scalable high performance roadmap for central processing

Products:
- S32R - Radar
- S32V - Vision

VDS
(Vehicle Dynamics & Safety)

**Chassis & Safety
Powertrain & Hybrid/EV**

- Long term Innovator in Chassis and Powertrain Control.
- Significant Growth in Safety as Autonomous Control Drives Robust Fault Tolerant Systems

Products:
MPC56xx
MPC57xx

GPIS
(General Purpose & Integrated Solutions)

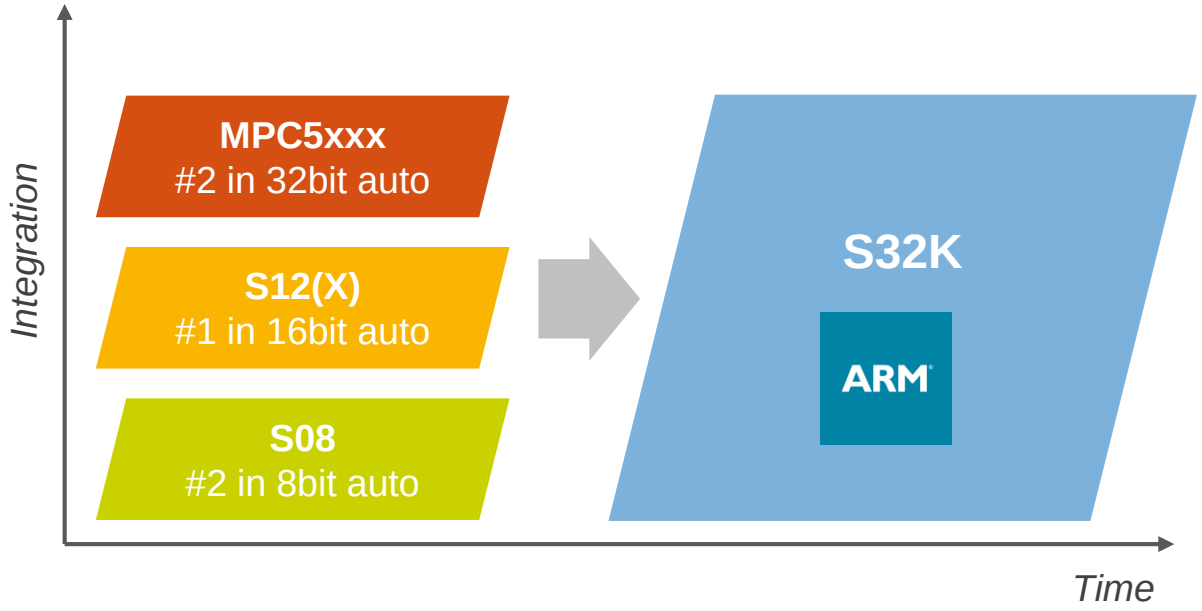
**Body Electronics
Edge Nodes**

- 500+ customers
- Broadest portfolio of integrated MCU+HV mixed-signal solutions
- Complete Tools & Software enablement

Products:
S08/S12/PPC → ARM
KEA – S32K
S12 MagniV – S32M

New General Purpose Automotive S32K Portfolio

- Automotive MCUs designed for SW engineers
- Moving to the ARM Cortex-M architecture
- Future-proofing through superior performance and advanced feature sets

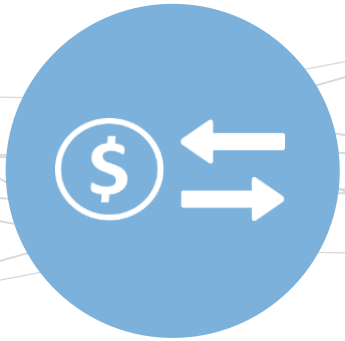


General Purpose Automotive – Design Challenges

Connectivity, Security, Safety & Low Power

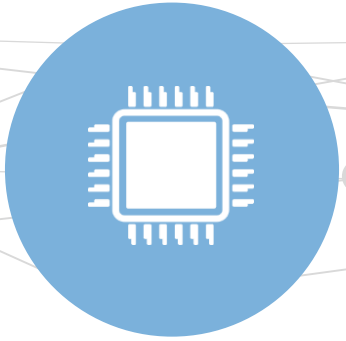
- Increased software content & re-programming requirements (frequency, efficiency & security)
- ISO26262 compliance requirements
- Maintaining low power targets

Software Complexity



- 60% of project resources spent on SW R&D
- Integration challenges of incompatible software from multiple sources

Hardware & Software Reuse



- No compatible MCU platform from 8 to 32 -bit
- Limited software reuse across applications and MCU families

S32K1 Family – Accelerating Automotive Software Design

Performance & Integration

Future proof designs

- ARM Cortex M4F and M0+ cores
- ISO CAN-FD, CSEc hardware security, ISO26262 ASIL-B functional safety
- Ultra low power

Automotive-grade SW

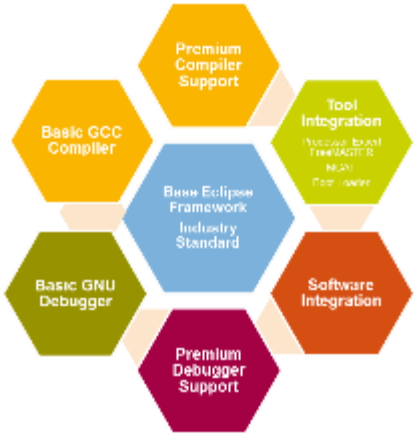
Minimized complexity

- Automotive-grade Software Development Kit (SDK)
- S32 Design Studio IDE
- Autosar MCAL & OS, 3rd party ecosystem

Broad Portfolio

Maximised reuse

- 128KB to 2MB, 32 to 176 pins
- H/w and S/w compatibility
- AEC Q100 grade 1 qualified (125°C), min. 15 year longevity

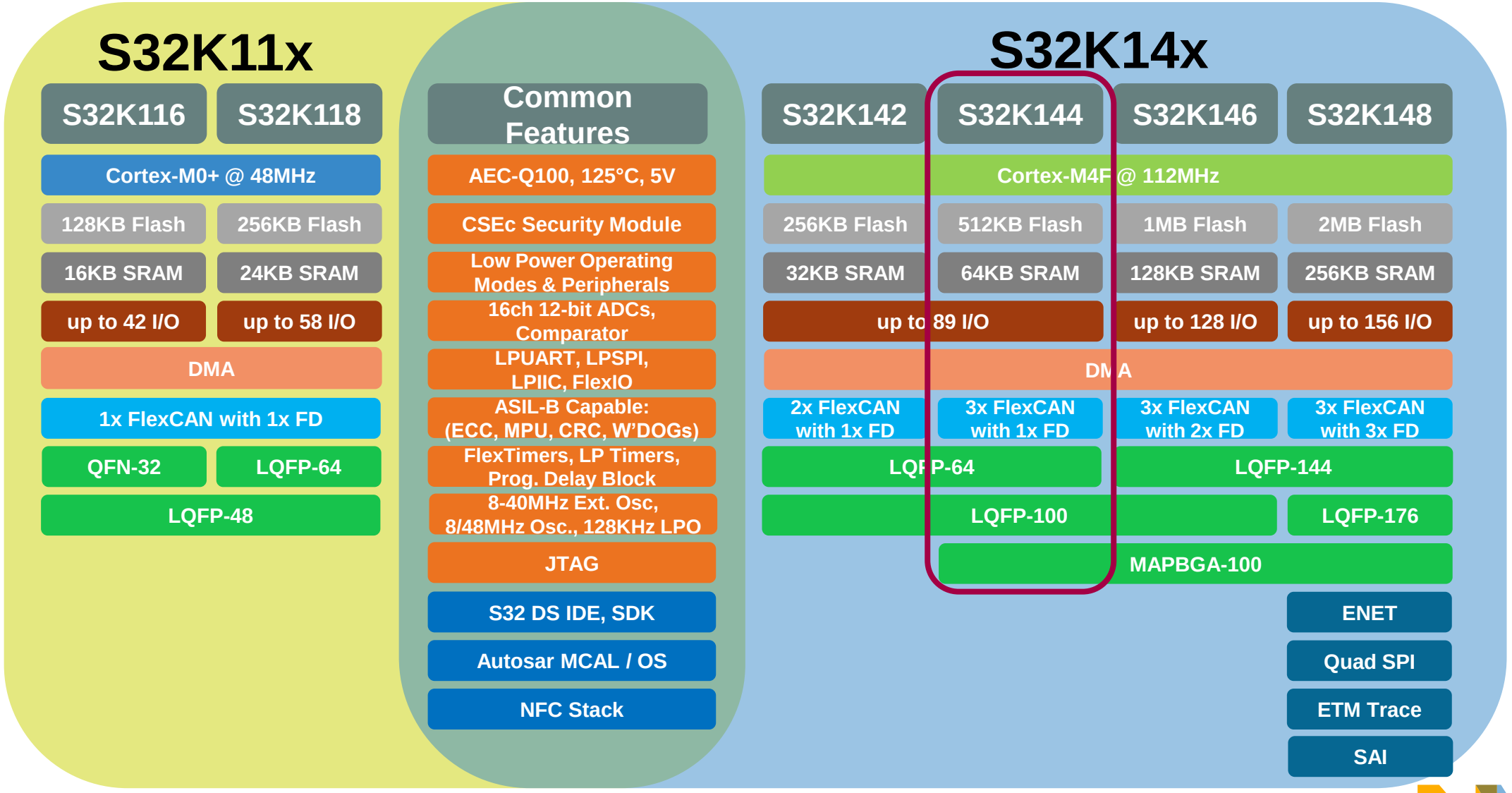


Product Longevity

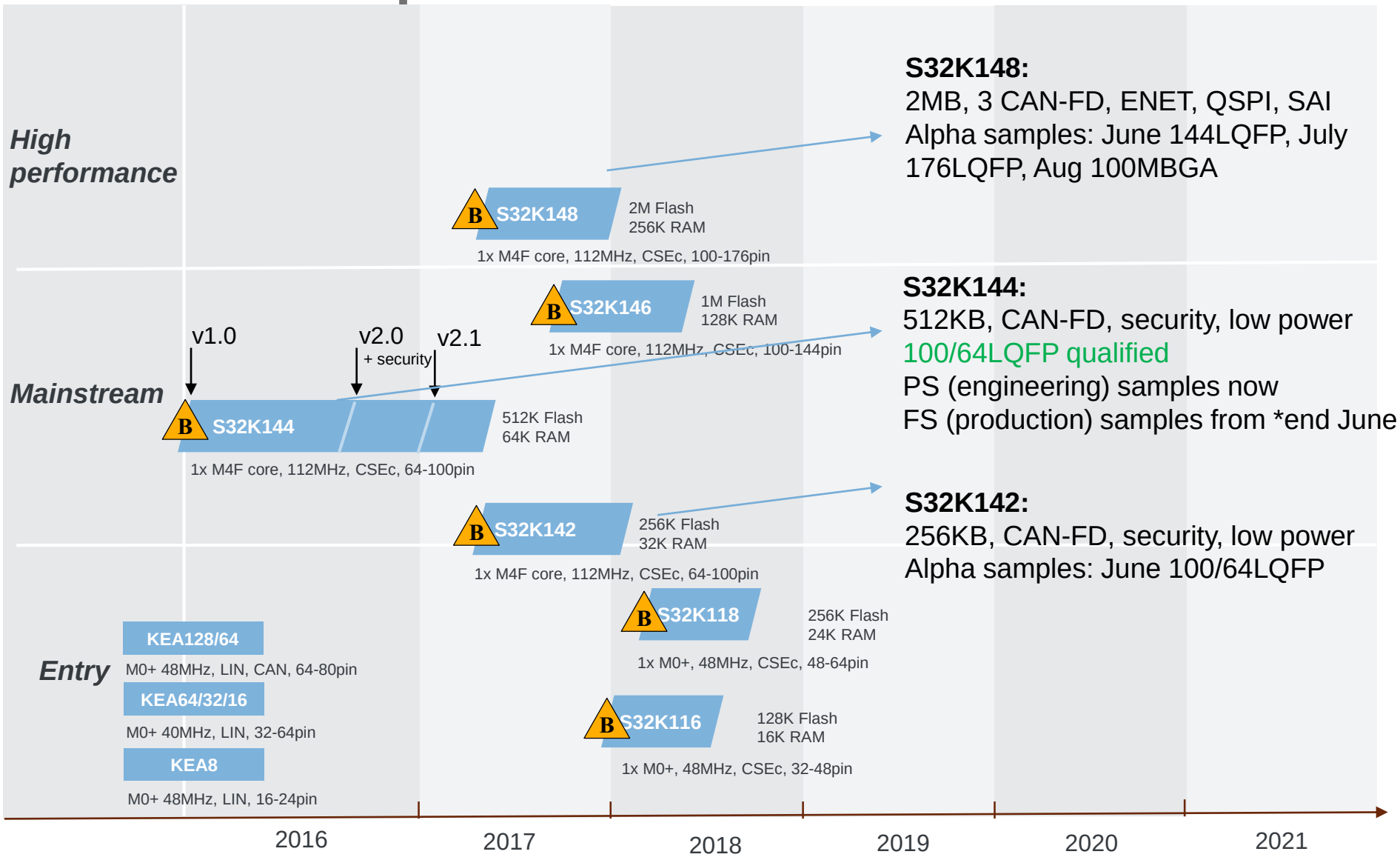
MCU FAMILIES

S32K1 Family Overview

Launched March 2017



S32K Roadmap



S32K148:
2MB, 3 CAN-FD, ENET, QSPI, SAI
Alpha samples: June 144LQFP, July 176LQFP, Aug 100MBGA

S32K144:
512KB, CAN-FD, security, low power
100/64LQFP qualified
PS (engineering) samples now
FS (production) samples from *end June

S32K142:
256KB, CAN-FD, security, low power
Alpha samples: June 100/64LQFP

ARM

First Sample Date (left edge)

Product Idea
Concept
Development
Production

Product Qualification (right edge)

X ASIL Level

DUAL Dual fab on S32K1xx

Last Updated 04 MAY17 subject to change

S32K144 Block Diagram

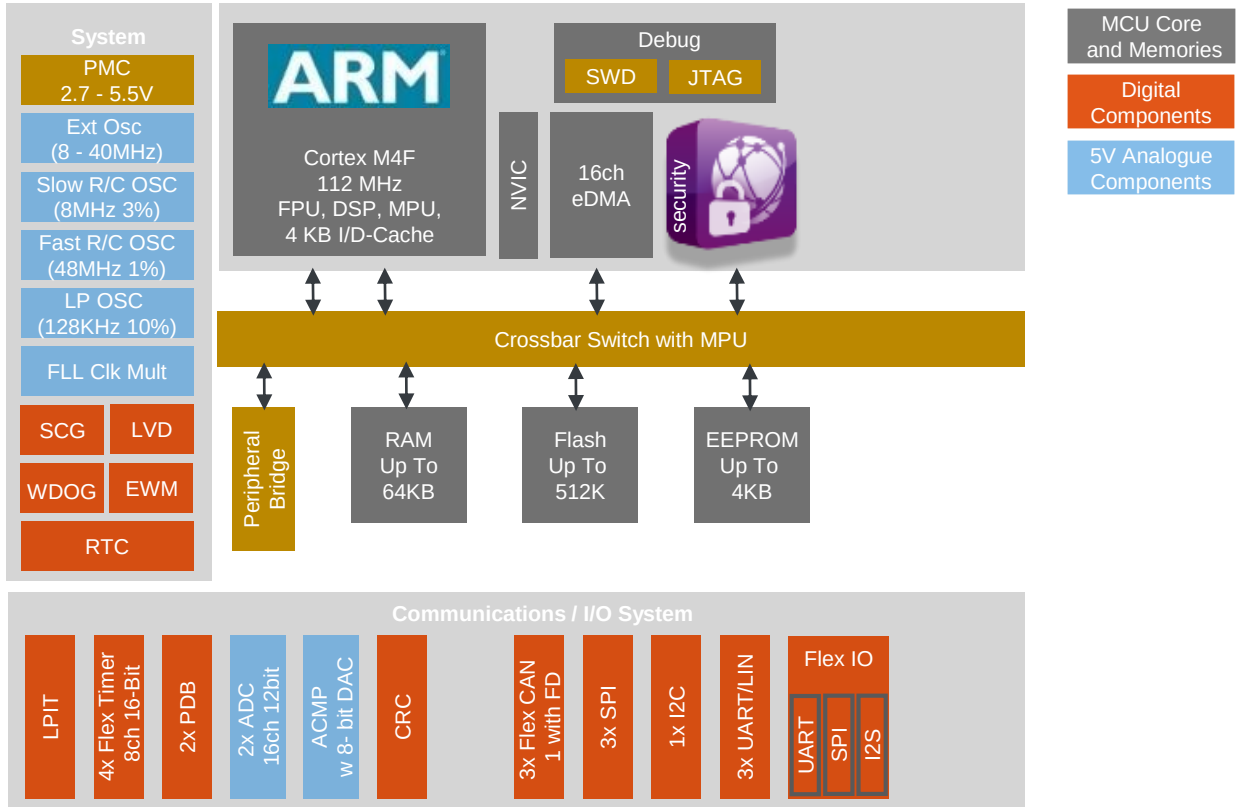
High performance
- ARM Cortex M4F up to 112MHz w FPU
- eDMA from 57xxx family

Software Friendly Architecture
- High RAM to Flash ratio
- Independent CPU and peripheral clocking
- 48MHz 1% IRC – no PLL init required in LP
- Registers maintained in all modes
- Programmable triggers for ADC no SW delay counters or extra interrupts

Functional safety
- ISO26262 support for ASIL B or higher
- Memory Protection Unit, ECC on Flash/Dataflash and RAM
- Independent internal OSC for Watchdog
- Diversity between ADC and ACMP, SPI/SCI and FlexIO
- Core self test libraries
- Scalable LVD protection, CRC

Low power
- Low leakage technology
- Multiple VLP modes and IRC combos
- Wake-up on analog thresholds

Security
- CSEc (SHE-spec)



Operating Characteristics
- Voltage range: 2.7V to 5.5V
- Temperature (ambient): -40°C to +125°C

Packages & IO
- Open-drain for 3.3 V and hi-drive pins
- Powered ESD protection
- Packages: 100 BGA, 64 LQFP, 100 LQFP

S32K148 Block Diagram

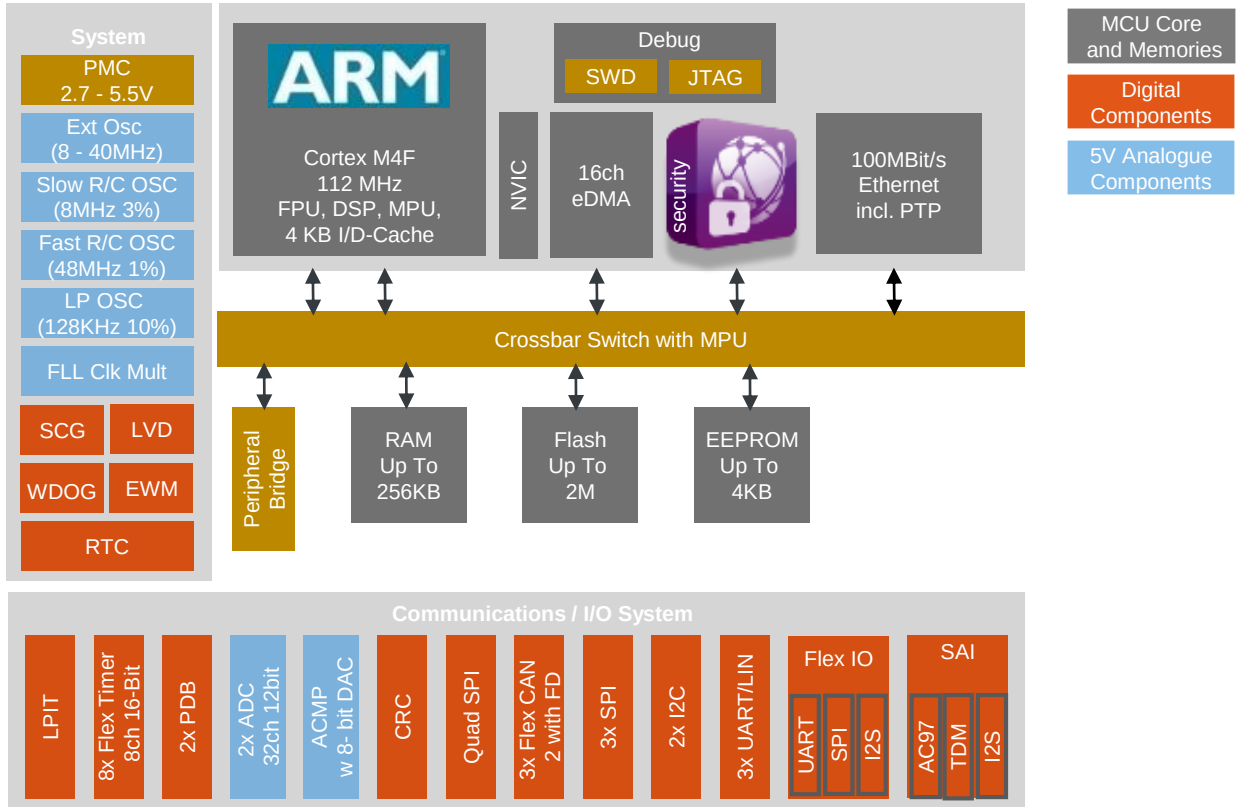
High performance
- ARM Cortex M4F up to 112MHz w FPU
- eDMA from 57xxx family

Software Friendly Architecture
- High RAM to Flash ratio
- Independent CPU and peripheral clocking
- 48MHz 1% IRC – no PLL init required in LP
- Registers maintained in all modes
- Programmable triggers for ADC no SW delay counters or extra interrupts

Functional safety
- ISO26262 support for ASIL B or higher
- Memory Protection Unit, ECC on Flash/Dataflash and RAM
- Independent internal OSC for Watchdog
- Diversity between ADC and ACMP, SPI/SCI and FlexIO
- Core self test libraries
- Scalable LVD protection, CRC

Low power
- Low leakage technology
- Multiple VLP modes and IRC combos
- Wake-up on analog thresholds

Security
- CSEc (SHE-spec)



Operating Characteristics
- Voltage range: 2.7V to 5.5V
- Temperature (ambient): -40°C to +125°C

Packages & IO
- Open-drain for 3.3 V and hi-drive pins
- Powered ESD protection
- Packages: 100 BGA, 144 LQFP, 176 LQFP

S32K146 Block Diagram

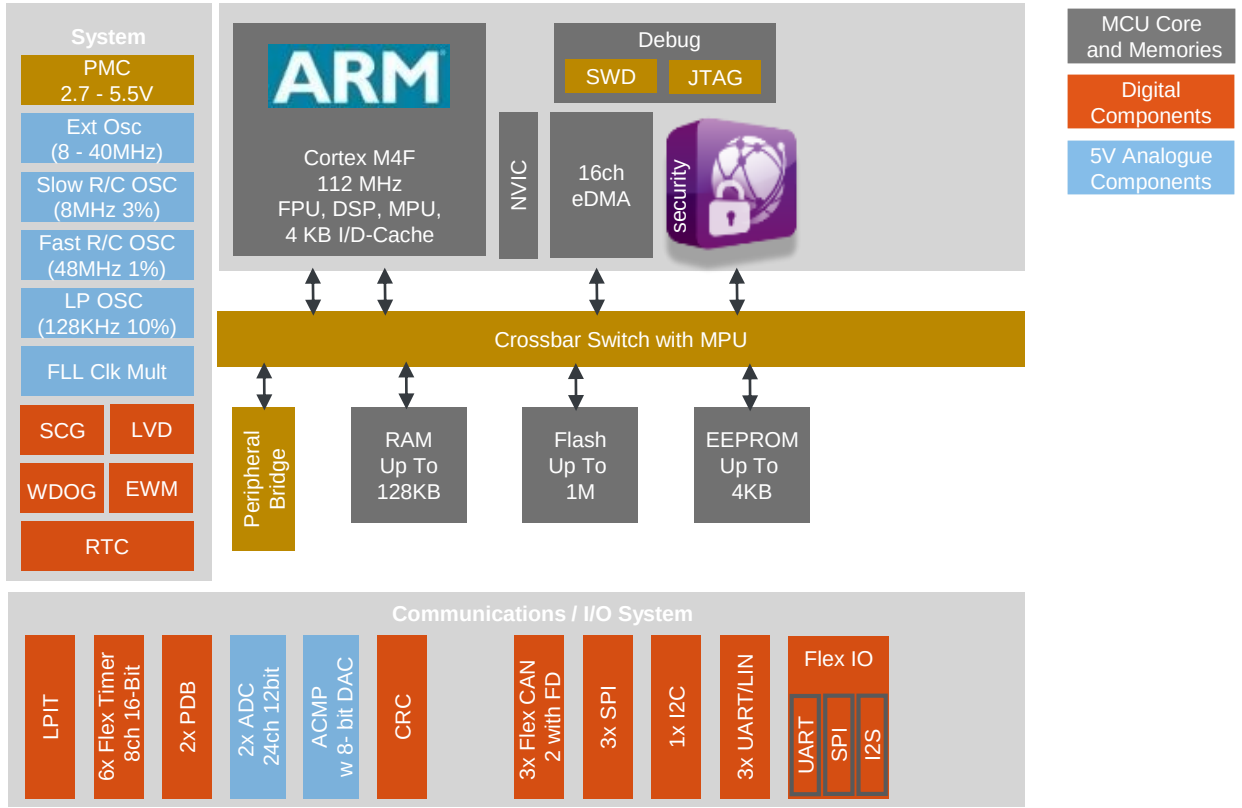
High performance
- ARM Cortex M4F up to 112MHz w FPU
- eDMA from 57xxx family

Software Friendly Architecture
- High RAM to Flash ratio
- Independent CPU and peripheral clocking
- 48MHz 1% IRC – no PLL init required in LP
- Registers maintained in all modes
- Programmable triggers for ADC no SW delay counters or extra interrupts

Functional safety
- ISO26262 support for ASIL B or higher
- Memory Protection Unit, ECC on Flash/Dataflash and RAM
- Independent internal OSC for Watchdog
- Diversity between ADC and ACMP, SPI/SCI and FlexIO
- Core self test libraries
- Scalable LVD protection, CRC

Low power
- Low leakage technology
- Multiple VLP modes and IRC combos
- Wake-up on analog thresholds

Security
- CSEc (SHE-spec)



Operating Characteristics
- Voltage range: 2.7V to 5.5V
- Temperature (ambient): -40°C to +125°C

Packages & IO
- Open-drain for 3.3 V and hi-drive pins
- Powered ESD protection
- Packages: 100 BGA, 100 LQFP, 144 LQFP, 176 LQFP

S32K142 Block Diagram

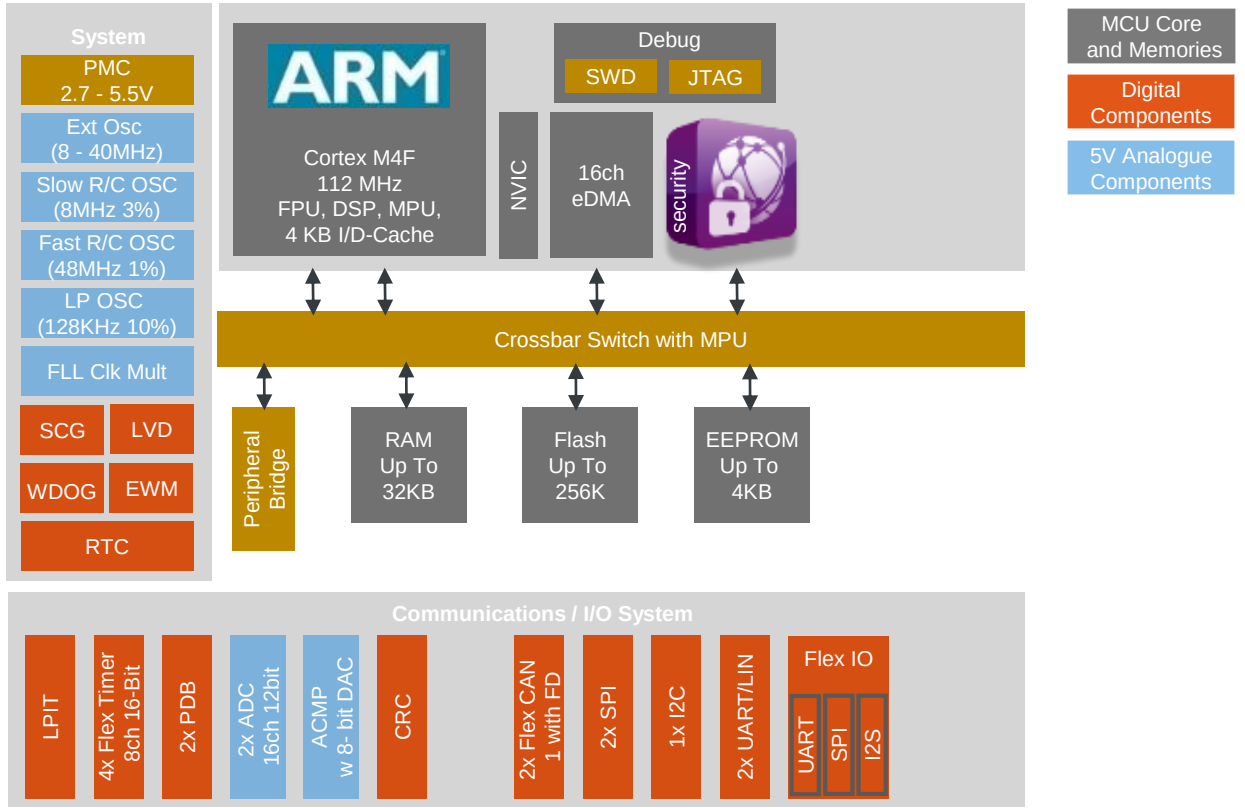
High performance
- ARM Cortex M4F up to 80MHz w FPU
- eDMA from 57xxx family

Software Friendly Architecture
- High RAM to Flash ratio
- Independent CPU and peripheral clocking
- 48MHz 1% IRC – no PLL init required in LP
- Registers maintained in all modes
- Programmable triggers for ADC no SW delay counters or extra interrupts

Functional safety
- ISO26262 support for ASIL B or higher
- Memory Protection Unit, ECC on Flash/Dataflash and RAM
- Independent internal OSC for Watchdog
- Diversity between ADC and ACMP, SPI/SCI and FlexIO
- Core self test libraries
- Scalable LVD protection, CRC

Low power
- Low leakage technology
- Multiple VLP modes and IRC combos
- Wake-up on analog thresholds

Security
- CSEc (SHE-spec)



Operating Characteristics
- Voltage range: 2.7V to 5.5V
- Temperature (ambient): -40°C to +125°C

Packages & IO
- Open-drain for 3.3 V and hi-drive pins
- Powered ESD protection
- Packages: 100 BGA, 64 LQFP

S32K118 Block Diagram

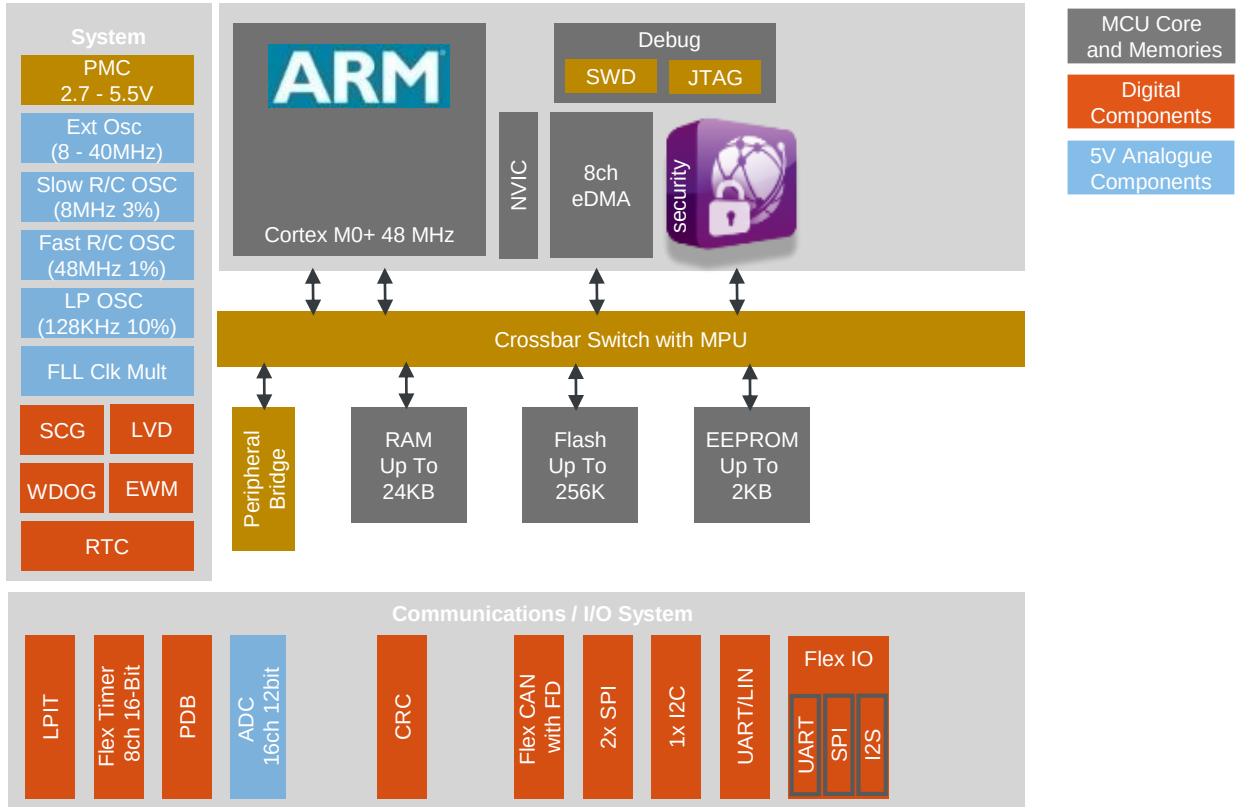
High performance
- ARM Cortex M0+ up to 48MHz
- eDMA from 57xxx family

Software Friendly Architecture
- High RAM to Flash ratio
- Independent CPU and peripheral clocking
- 48MHz 1% IRC – no PLL init required in LP
- Registers maintained in all modes
- Programmable triggers for ADC no SW delay counters or extra interrupts

Functional safety
- ISO26262 support for ASIL B or higher
- Memory Protection Unit, ECC on Flash/Dataflash and RAM
- Independent internal OSC for Watchdog
- Diversity between ADC and ACMP, SPI/SCI and FlexIO
- Core self test libraries
- Scalable LVD protection, CRC

Low power
- Low leakage technology
- Multiple VLP modes and IRC combos
- Wake-up on analog thresholds

Security
- CSEc (SHE-spec)



Operating Characteristics
- Voltage range: 2.7V to 5.5V
- Temperature (ambient): -40°C to +125°C

Packages & IO
- Open-drain for 3.3 V and hi-drive pins
- 32 pin compatible within series
- Packages: 64 LQFP, 48LQFP

S32K116 Block Diagram

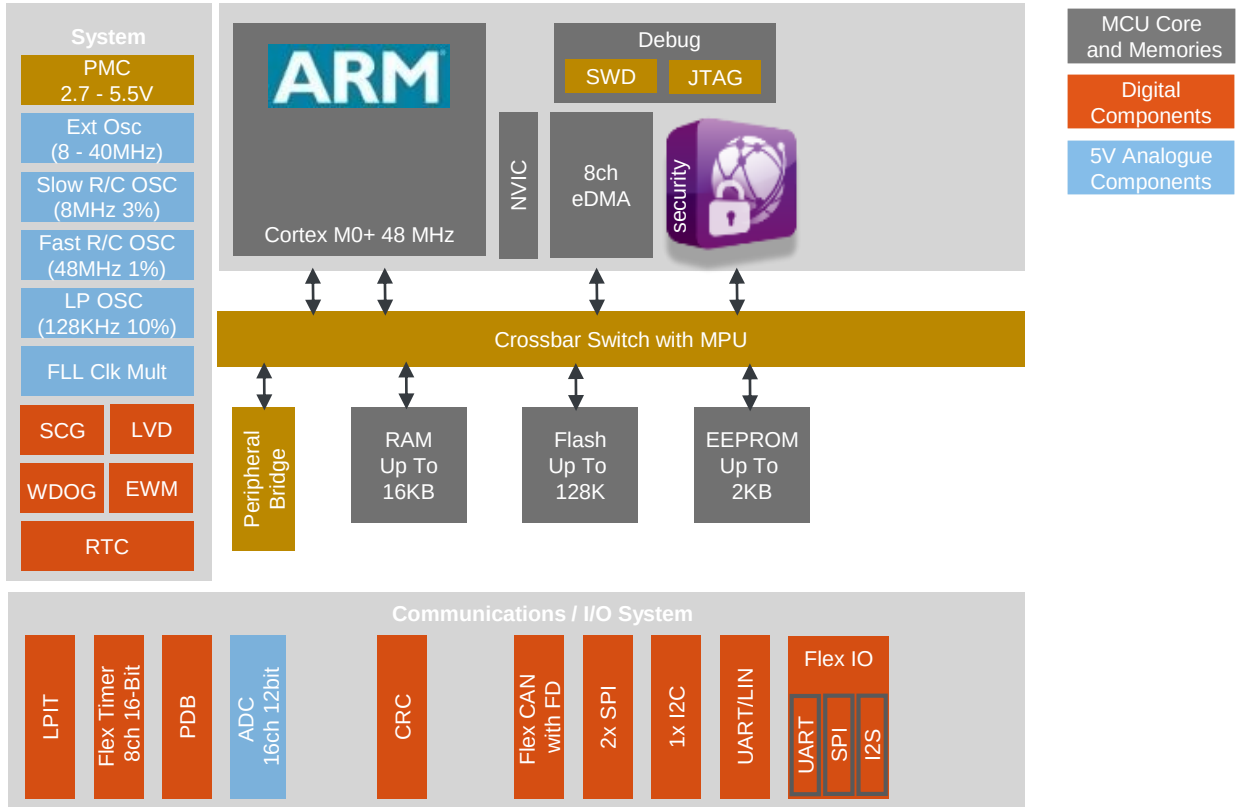
High performance
- ARM Cortex M0+ up to 48MHz
- eDMA from 57xxx family

Software Friendly Architecture
- High RAM to Flash ratio
- Independent CPU and peripheral clocking
- 48MHz 1% IRC – no PLL init required in LP
- Registers maintained in all modes
- Programmable triggers for ADC no SW delay counters or extra interrupts

Functional safety
- ISO26262 support for ASIL B or higher
- Memory Protection Unit, ECC on Flash/Dataflash and RAM
- Independent internal OSC for Watchdog
- Diversity between ADC and ACMP, SPI/SCI and FlexIO
- Core self test libraries
- Scalable LVD protection, CRC

Low power
- Low leakage technology
- Multiple VLP modes and IRC combos
- Wake-up on analog thresholds

Security
- CSEc (SHE-spec)



Operating Characteristics
- Voltage range: 2.7V to 5.5V
- Temperature (ambient): -40°C to +125°C

Packages & IO
- Open-drain for 3.3 V and hi-drive pins
- 32 pin compatible within series
- Packages: 48 LQFP, 32 QFN

S32K144 – Sample Availability

Attribute	Rev2.0 (0N47T)	Rev2.1 (0N57U / Production)
R-gate /Qual.	-	100/64LQFP: Now, 100BGA: Oct '17

Availability	'PS' (engineering)
	• Small qty of 100/64LQFP samples left
	  • PS32K144UAT0VLLT (100LQFP)
	  • PS32K144UFT0VLHT (64LQFP, no CSEc)
	• Suitable for non Auto OEM related opptys
	• No functional difference vs. Rev2.1 – minor quality change for single-event latch-up performance (ppm related – not functionality related)
		'PS' (engineering)
		• 100/64 LQFP: ~12K July (6K each pckg)
		  • PS32K144UAT0VLLA (100LQFP)
		  • PS32K144UAT0VLHA (64LQFP)
		• 100MBGA: from end June (very small qtys)
		  • PS32K144UAT0VMHA (100MBGA)
		'FS' (production)
		• See pns on following slide
		• LQFP: from end June (lead ctms) / from mid Sept (general Distribution)
		• BGA: from Oct

PPAP Doc.	-	End June
S.O.P	-	End June (on allocation)

MHz: **H** = 80MHz / **U** = 112MHz
Features: **F** = CAN-FD, FlexIO, max. RAM
A = CAN-FD, FlexIO, **Security,** max. RAM

S32K144 – Production Part Numbers

PN	Comment	Package	Freq. (MHz)	CAN-FD	Security (CSEc)	FlexIO	Temp	Availability
FS32K144UAT0VLHT	Superset, general sampling	64LQFP	112	X	X	X	105C	From Sept
FS32K144UAT0VLLT	Superset, general sampling	100LQFP	112	X	X	X	105C	From Sept
FS32K144UAT0VMHT	Superset, general sampling	100MBGA	112	X	X	X	105C	From Oct

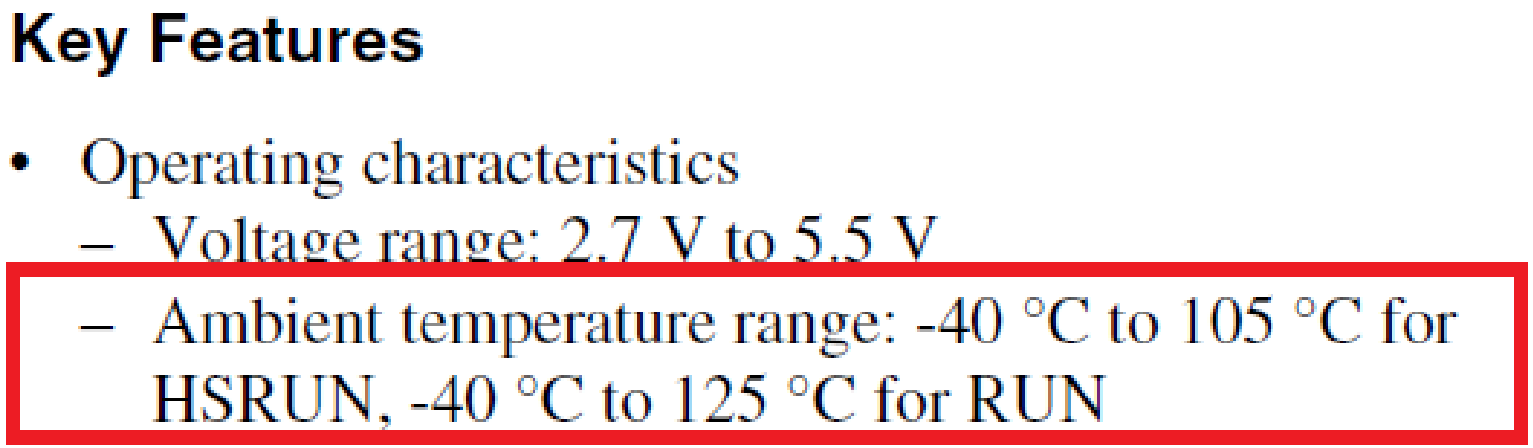
- S32K144 family consists of multiple possible pn options (speed/feature/temp/package/T&R)
- For mfg logistical reasons a limited set of production part numbers is preferred for Disty customers
- For customer who require a non superset part number, a MOQ (tbc) and minimum lead time shall apply
  - MPQs:
    - 100LQFP: 450# (tray), 1000# (T&R)
    - 64LQFP: 800# (tray), 1500# (T&R)
    - 100MBGA: tbc
  - LT: 26wks

- Orders (for sampling & production qtys)
  - Qty <MPQ: order via e-commerce
  - Qty >=MPQ: order via SAP/Distributor
  - Orders should be placed ASAP for 2017 demand

Last Updated 30 AUG 2016

S32K144 – MHz/Temp Specification

- 112MHz (HSRUN) is an overclocking mode which should be used only for critical tasks in “burst” mode. Normal device operation should be at 80MHz
- Customers wishing to use S32K144 @ M temp (125C) should restrict operation to 80MHz as per the guidance in the data sheet
- Customers who require M temp production parts should order the 80MHz (H) version and not the 112MHz (U) version



Key Features

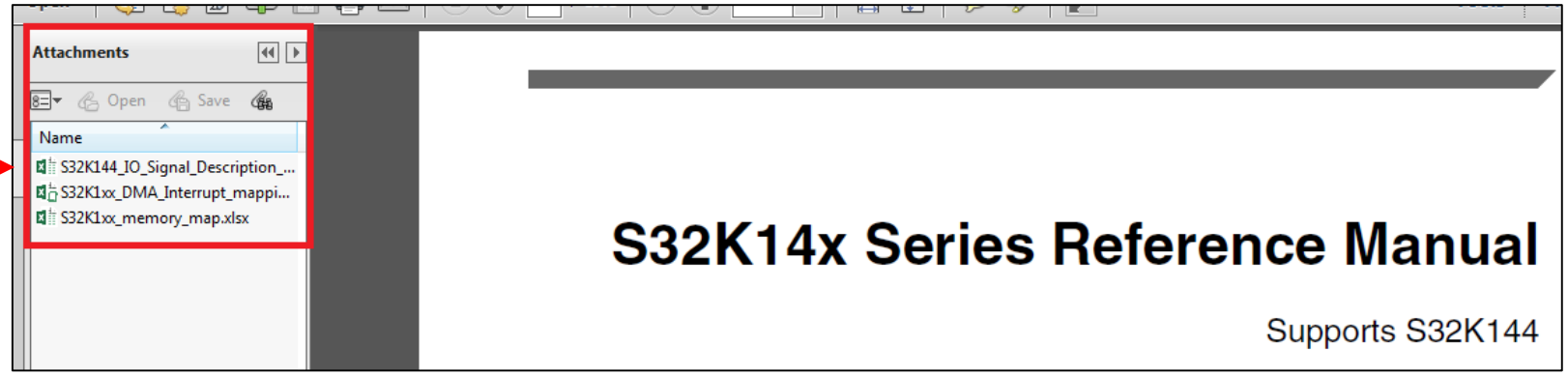
- Operating characteristics
  - Voltage range: 2.7 V to 5.5 V
  - Ambient temperature range: -40 °C to 105 °C for HSRUN, -40 °C to 125 °C for RUN

S32K1 – Documentation @ www.nxp.com/S32K

Document	Products Covered	Availability
S32K1xx_Product Brief	All S32K1xx MCUs	Now – on web
S32K144_Data Sheet	All S32K1xx MCUs	Now – on web
S32K144_Reference Manual	All S32K1xx MCUs	Now – on web
S32K1xx_Safety Manual	All S32K1xx MCUs	Now – on web

Note: RM Attachment Files:

- IO Signal Descriptions
- DMA Interrupt Mapping
- Memory Map



- **App Notes**
  - S32K14x CookBook: s/w examples for CLKs, Interrupts, DMA, FTM, ADC, UART, SPI, CAN-FD. Built using S32DS & tested on S32K144EVB
  - Others: CSEc, Power Mgmt, Clock Calculator, FlexTimer, DMA, H/W Design Guidelines, EE, Codewarrior to S32_DS Migration

Production Part Numbers – S32K142 & 144

Flash Size (kB)	PN	Core Frequency (MHz)	CAN-FD (yes/no)	FlexIO (yes/no)	RAM Size (kB)	Security (yes/no)	Ethernet / Audio	x	xx	x
256	FS32K142MNT0xxxx	64	no	no	32	no	no	C = -40 to 85oC V = -40 to 105oC M = -40 to 125oC	LH = 64LQFP LL = 100LQFP	T = Trays/ Tubes R = Tape & Reel
	FS32K142MFT0xxxx		yes	yes	32	no	no			
	FS32K142MST0xxxx		no	no	32	yes	no			
	FS32K142MAT0xxxx		yes	yes	32	yes	no			
	FS32K142HNT0xxxx	80	no	no	32	no	no			
	FS32K142HFT0xxxx		yes	yes	32	no	no			
	FS32K142HST0xxxx		no	no	32	yes	no			
	FS32K142HAT0xxxx		yes	yes	32	yes	no			
	FS32K142UNT0xxxx	112	no	no	32	no	no			
	FS32K142UFT0xxxx		yes	yes	32	no	no			
	FS32K142UST0xxxx		no	no	32	yes	no			
	FS32K142UAT0xxxx		yes	yes	32	yes	no			
512	FS32K144MNT0xxxx	64	no	no	48	no	no	C = -40 to 85oC V = -40 to 105oC M = -40 to 125oC	LH = 64LQFP LL = 100LQFP MH = MAPBGA100	T = Trays/ Tubes R = Tape & Reel
	FS32K144MRT0xxxx		no	no	64	no	no			
	FS32K144MFT0xxxx		yes	yes	64	no	no			
	FS32K144MST0xxxx		no	no	64	yes	no			
	FS32K144MAT0xxxx		yes	yes	64	yes	no			
	FS32K144HNT0xxxx	80	no	no	48	no	no			
	FS32K144HRT0xxxx		no	no	64	no	no			
	FS32K144HFT0xxxx		yes	yes	64	no	no			
	FS32K144HST0xxxx		no	no	64	yes	no			
	FS32K144HAT0xxxx		yes	yes	64	yes	no			
	FS32K144UNT0xxxx	112	no	no	48	no	no			
	FS32K144URT0xxxx		no	no	64	no	no			
	FS32K144UFT0xxxx		yes	yes	64	no	no			
	FS32K144UST0xxxx		no	no	64	yes	no			
	FS32K144UAT0xxxx		yes	yes	64	yes	no			

Production Part Numbers – S32K146 & 148

Flash Size (kB)	PN	Core Frequency (MHz)	CAN-FD (yes/no)	FlexIO (yes/no)	RAM Size (kB)	Security (yes/no)	Ethernet / Audio	x	xx	x
1024	FS32K146MNT0xxxx	64	no	no	96	no	no	C = -40 to 85oC V = -40 to 105oC M = -40 to 125oC	LQ = 144LQFP LL = 100LQFP MH = MAPBGA100	T = Trays/ Tubes R = Tape & Reel
	FS32K146MRT0xxxx		no	no	128	no	no			
	FS32K146MFT0xxxx		yes	yes	128	no	no			
	FS32K146MST0xxxx		no	no	128	yes	no			
	FS32K146MAT0xxxx		yes	yes	128	yes	no			
	FS32K146HNT0xxxx	80	no	no	96	no	no			
	FS32K146HRT0xxxx		no	no	128	no	no			
	FS32K146HFT0xxxx		yes	yes	128	no	no			
	FS32K146HST0xxxx		no	no	128	yes	no			
	FS32K146HAT0xxxx		yes	yes	128	yes	no			
	FS32K146UNT0xxxx	112	no	no	96	no	no			
	FS32K146URT0xxxx		no	no	128	no	no			
	FS32K146UFT0xxxx		yes	yes	128	no	no			
	FS32K146UST0xxxx		no	no	128	yes	no			
	FS32K146UAT0xxxx		yes	yes	128	yes	no			
2048	FS32K148HNT0xxxx	80	no	no	192	no	no	C = -40 to 85oC V = -40 to 105oC M = -40 to 125oC	LQ = 144LQFP LU = 100LQFP MH = MAPBGA100	T = Trays/ Tubes R = Tape & Reel
	FS32K148HRT0xxxx		no	no	256	no	no			
	FS32K148HFT0xxxx		yes	yes	256	no	no			
	FS32K148HST0xxxx		no	no	256	yes	no			
	FS32K148HAT0xxxx		yes	yes	256	yes	no			
	FS32K148HET0xxxx		no	no	256	no	yes			
	FS32K148HJT0xxxx		yes	yes	256	yes	yes			
	FS32K148UNT0xxxx	112	no	no	48	no	no			
	FS32K148URT0xxxx		no	no	64	no	no			
	FS32K148UFT0xxxx		yes	yes	64	no	no			
	FS32K148UST0xxxx		no	no	64	yes	no			
	FS32K148UAT0xxxx		yes	yes	64	yes	no			
	FS32K148UET0xxxx		no	no	256	no	yes			
	FS32K148UJT0xxxx		yes	yes	256	yes	yes			

S32K Part Numbering: NXP GPIS Products

Ordering Part number (always 16 characters)

F/P	1	*Product status*
S32	2-4	*Product Type/Brand*
K	5	*Product Line*
1	6	*Series/Family (incl. generation)*
0	7	*Core platform / Performance indicator*
0	8	*Memory Size*
X	9	*Option #1: letter*
Y	10	*Option #2: letter*
F0	11-12	*Fab and Mask rev letter*
M	13	*Temperature Suffix*
LC	14-15	*Package Suffix*
R	16	*Tape and Reel Indicator*

1st Character
Product Status for ordering and marking

P for prototype and
F for qualified ordering P/N

2nd, 3rd & 4th Character
Product Type / Brand

S32 for Automotive 32Bit MCU

5th Character
Product Line

K = ARM Cortex MCUs

6th Character
Series / Family

1 = 1st product family
2 = 2nd product family

7th Character
Core platform / Performance indicator

1 = M0+
4 = M4F

8th Character
Product/Memory

	1	2	3	4	5	6	7	8	9
M0+						128k		256k	
M4F		256k		512k		1M		2M	

9th & 10th Characters
Ordering options

X: Speed
B = 48 MHz without DMA (only for S32K11x)
L = 48 MHz with DMA (only for S32K11x)
M = 64 MHz
H = 80 MHz (112 MHz for K144 v1.0)
U = 112 MHz

Y: Optional feature
N = No / none
R = RAM max.
F = CAN FD & FlexIO incl. max. RAM
S = Security incl. max. RAM
A = CAN FD, FlexIO & Security incl. max. RAM
E = Ethernet & Audio incl. max. RAM
J = CAN FD, FlexIO, Security, Ethernet & Audio incl. max. RAM

11th & 12th Character
Fab and Mask rev

Fx = ATMC
Tx = GF
XX = Flex #

x**0** = 1st fab revision
x**1** = 2nd fab revision

13th Character
Temperature

C = -40°C to 85°C
V = -40°C to 105°C
M = -40°C to 125°C
W = -40°C to 150°C – not available

14th & 15th Character
Package Suffix

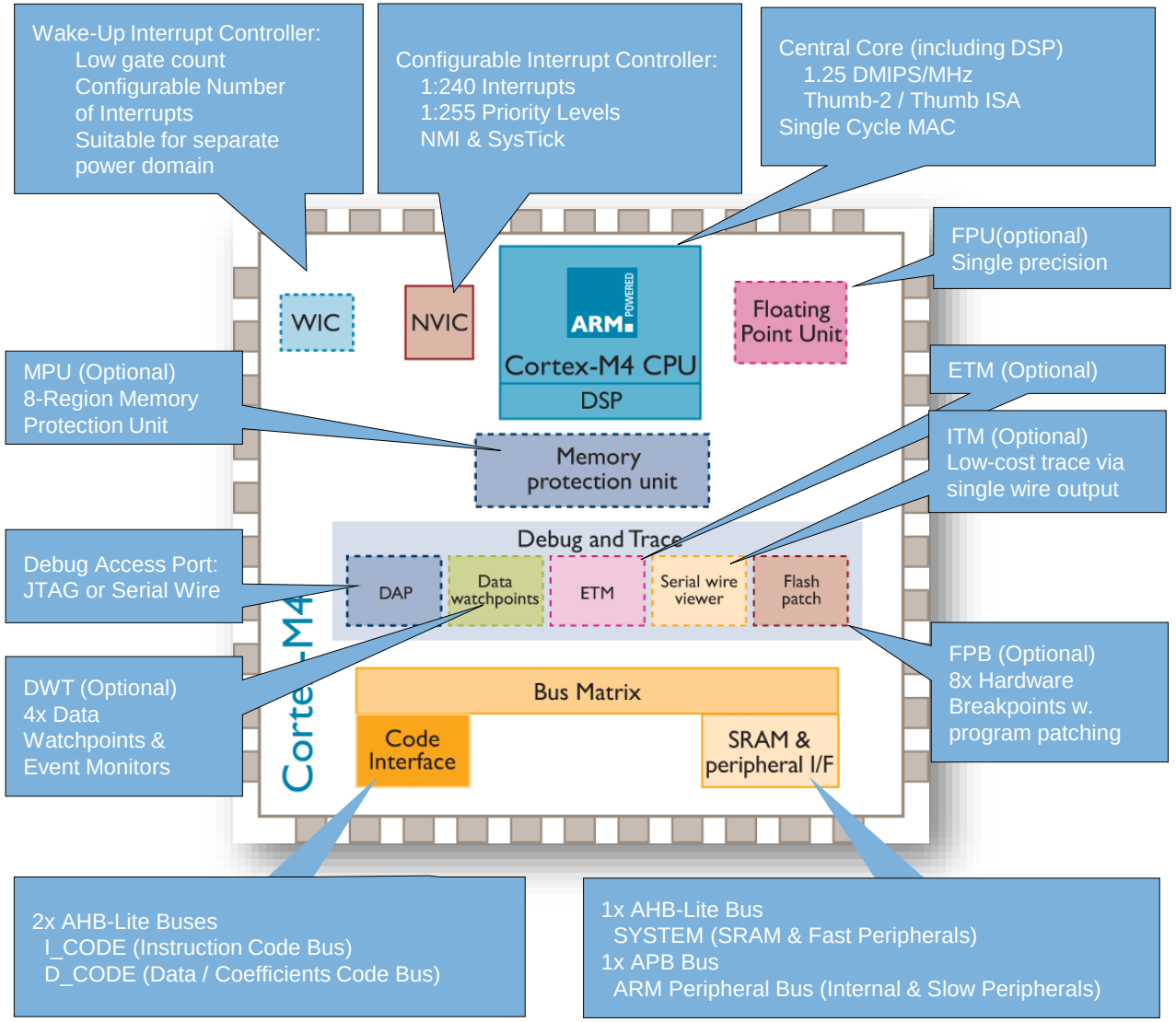
pins	TSSOP	LQFP	LQFP-EP	QFN	BGA
16	TG	-	-	-	-
32	-	LC	-	FM	-
48	-	LF	KF	FT	-
64	-	LH	KH	-	-
80	-	LK	KK	-	-
100	-	LL	-	-	MH
144	-	LQ	-	-	-
176	-	LU	-	-	-

16th Character
Tape & Reel

T = Trays/Tubes
R = Tape & Reel

KEY FEATURES

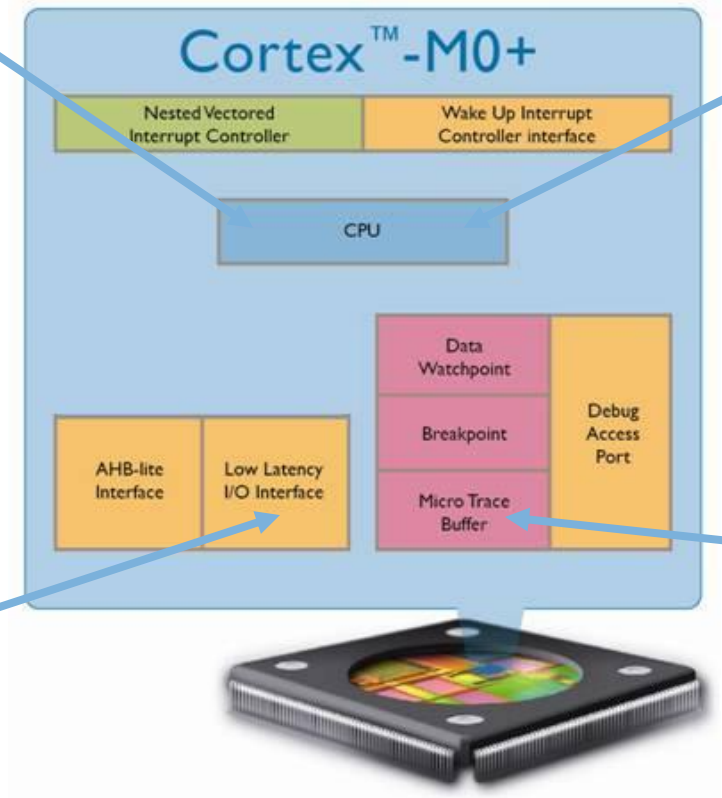
ARM Cortex-M4F Processor



ARM Cortex-M0+ Processor

Energy-Efficiency
•2-stage pipeline – reduced cycles per instruction (CPI) enabling faster branch instruction and ISR entry
•Program memory access on alternate cycles

Single-Cycle I/O Port
•50% higher GPIO toggling frequency than standard I/O
•Improves reaction time to external events allowing bit-banding and software protocol emulation
•Save precious cycles, e.g. set faster peripherals for low-power access
•Access GPIO/peripherals while processor fetches the next instruction



Processing
•Only 56 Instructions, mostly coded on 16-bit. Option for fast MUL 32x32 bit in 1 cycle
•Cortex-M0/3/4 compatible
•1.77CM/MHz
•Best-in-class code density vs. 8/16-bit architectures – reduced cost, power consumption and pin-count

Micro Trace Buffer
•Powerful, lightweight trace solution enabling fast debug
•Non-intrusive – trace information stored in small area of MCU SRAM (size defined by programmer)
• Trace read over Serial Wire /JTAG (CPU stopped)

90nm TFS (Thin Film Storage) with FlexMemory Technology

NXP's next-generation microcontrollers will offer exceptional low-power capability, performance and flexibility through our 90nm thin film storage (TFS) flash process with FlexMemory technology.

Thin Film Storage: split-gate non-volatile memory

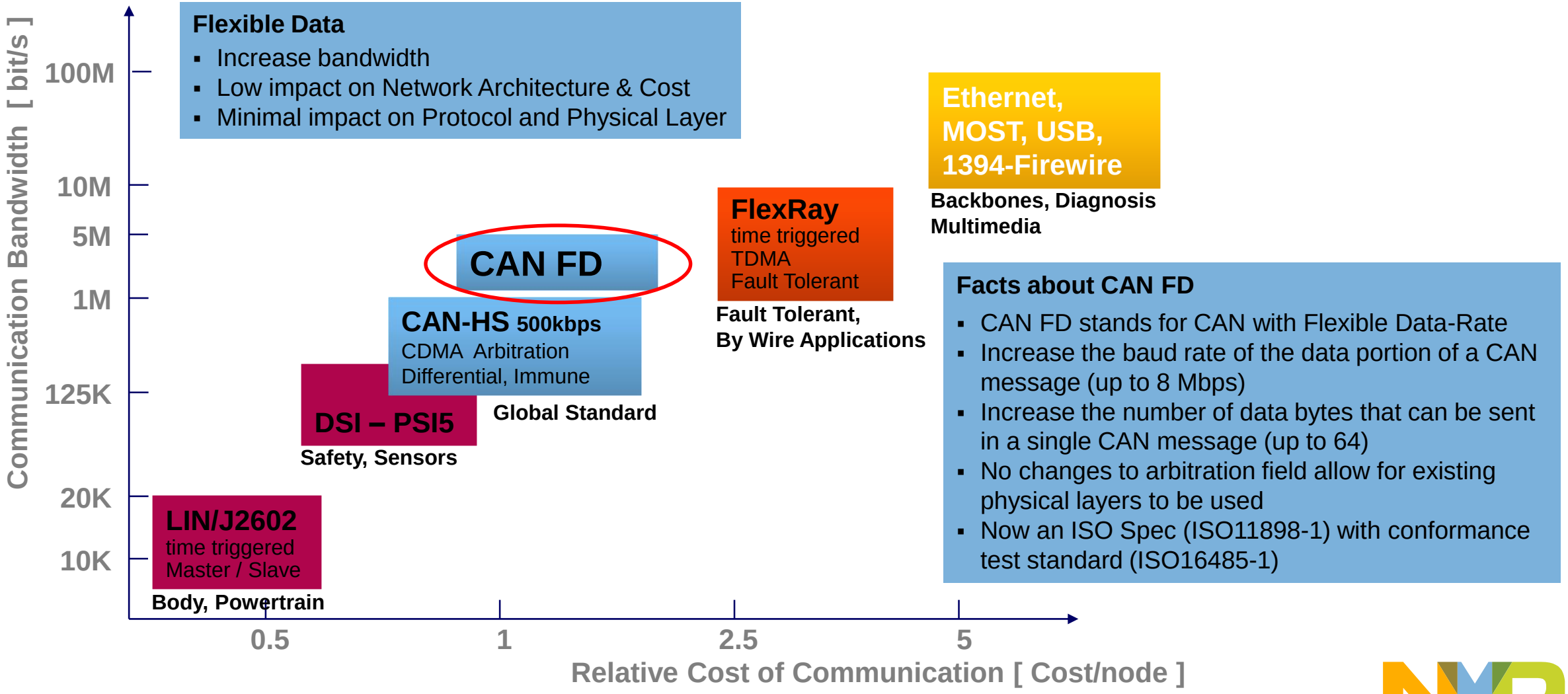
- ***Low power*:** significantly reduced run and standby currents vs. existing 90nm flash technologies
- ***Scalable:*** competitive for both low and high density arrays
- ***Reliable:*** unique nanocrystal technology provides unparalleled reliability
- ***Efficiency:*** 30% fewer added masks compared to competing 90nm flash technologies, fast program/erase operations and high area-efficiency

FlexMemory: integrated enhanced EEPROM capability

- ***Robust:*** extreme high endurance byte write/erase EEPROM with exceptional flexibility and performance
- ***Configurable:*** user programmable as EEPROM and/or additional flash memory
- ***Easy:*** seamless EEPROM read/write operations

CAN FD – CAN Flexible Data Rate

Increasing Bandwidth but not cost

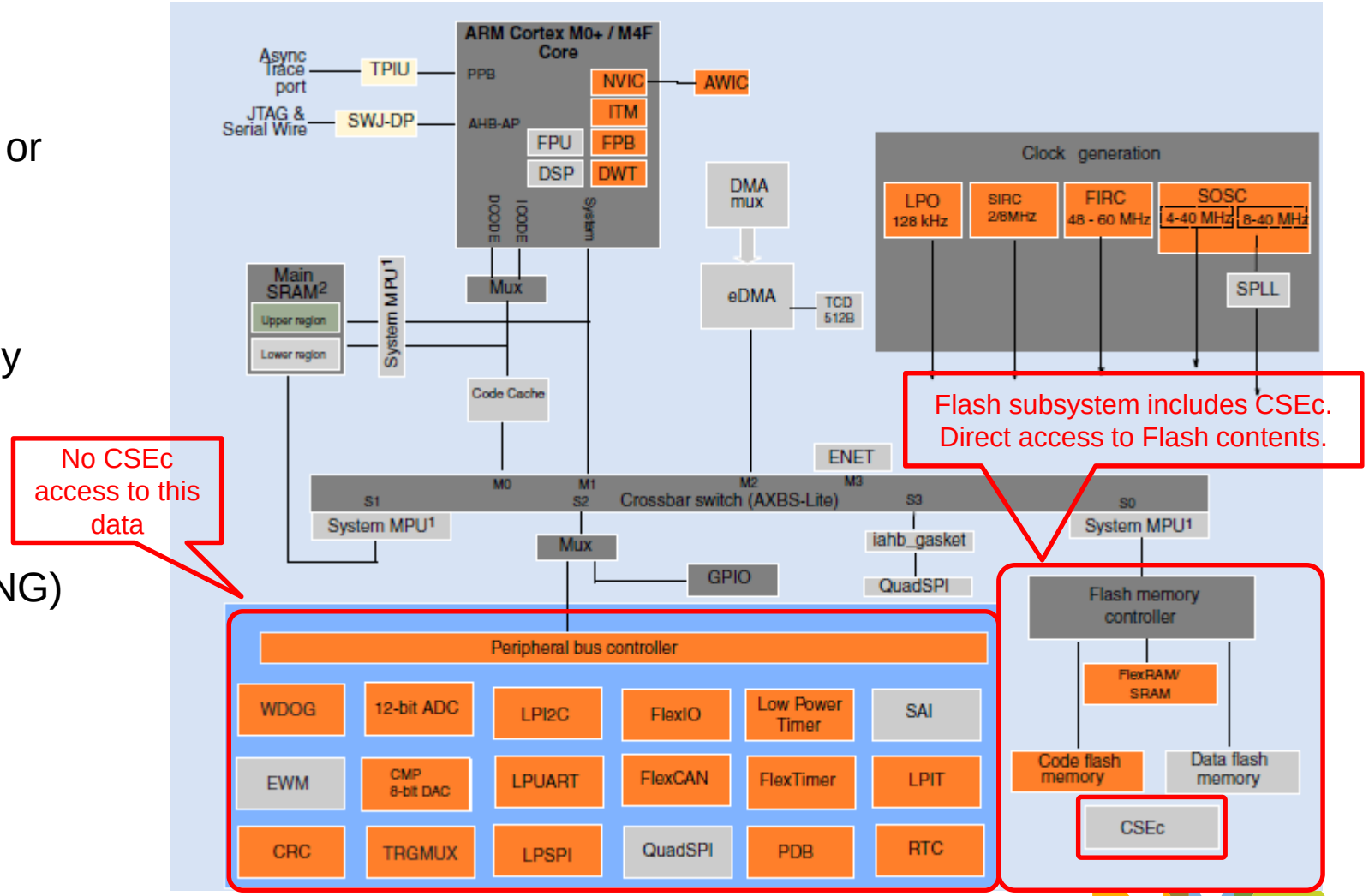


Facts about CAN FD

- CAN FD stands for CAN with Flexible Data-Rate
- Increase the baud rate of the data portion of a CAN message (up to 8 Mbps)
- Increase the number of data bytes that can be sent in a single CAN message (up to 64)
- No changes to arbitration field allow for existing physical layers to be used
- Now an ISO Spec (ISO11898-1) with conformance test standard (ISO16485-1)

CSEc (Cryptographic Services Engine compressed)

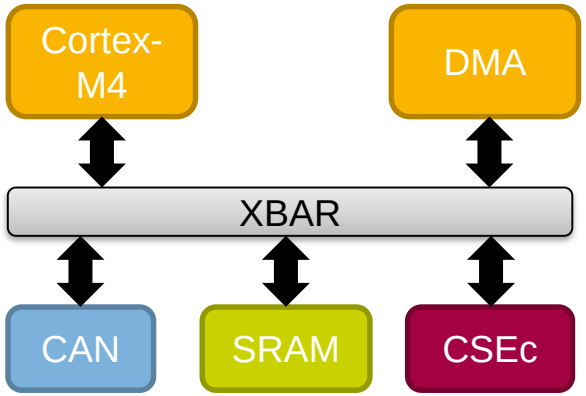
- 32-bit security co-processor
- Direct access to Flash data, data in SRAM/Peripherals accessable via core or DMA transfers
- Supports all SHE(+) & EVITA-Low standards – encode/decode, CMAC calulation and verification, loading of key values
- AES-128 hardware cipher block
- Secure Key storage
- Random number generator (TRNG/PRNG)
- 120-bit unique identification (UID)



CSEc – Module Interaction & Data-Flow

Scenario: Rx-CAN Message Authentication

1. CAN data stored in local buffer
2. FlexCAN triggers interrupt to CORE/DMA
3. Transfer Data to CSEc Memory
4. Trigger CSEc CMAC calculation/verification
5. CSEc triggers interrupt to Core
6. Core processes message data



Application Notes:

- AN4234 - Using the Cryptographic Service Engine (CSE)
- AN4235 - Using CSE to protect your Application Code via a Chain of Trust

CSEc – Secure Boot Use Case

Check Boot Loader for Integrity and Authenticity

Step 1: After power on: CSEc module reads the bootloader via its bus master interface.

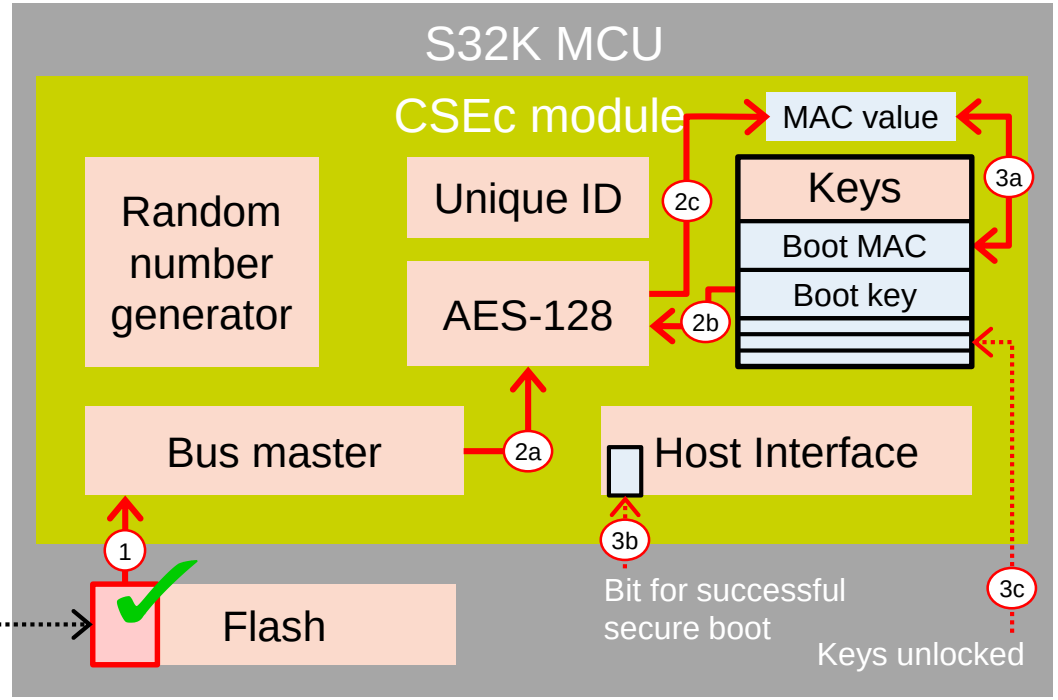
Step 2: CSEc module uses the boot key to calculate the MAC value of the bootloader.

Step 3: CSEc module compares calculated MAC with stored boot MAC. If identical: successful secure boot → set respective bit in host interface and unlock keys

Step 4: MCU always starts bootloader.

Bootloader:
Part of flash memory

Start bootloader



- MAC protects against modification of bootloader and depends on the (secret) boot key → integrity and authenticity of bootloader.
- Only if calculated MAC value matches stored boot MAC value: successful secure boot → set respective bit in host interface and unlock keys for further usage (see next demos)

FlexIO Peripheral

- FlexIO = **Flexible input and output peripheral**
- Programmable logic for complex output waveform generation
- Emulation of standard communication interfaces:
  - UART, SPI, I2C, I2S, LCD RGB, PWM, SENT, etc.
- Low CPU overhead
- DMA support
- Bare metal drivers available

FlexIO Features

- Allows emulation of standard communication interfaces.
- Supports a wide range of protocols and peripherals including:
  - UART
  - I2C
  - SPI
  - I2S
  - LCD RGB
  - CMT (carrier modulator transmitter)
  - PWM/waveform generation
  - SWD (single wire debug)
  - CAN
  - LIN
- Creates an interlink between GPIO method of software emulation and exact hardware peripheral module.

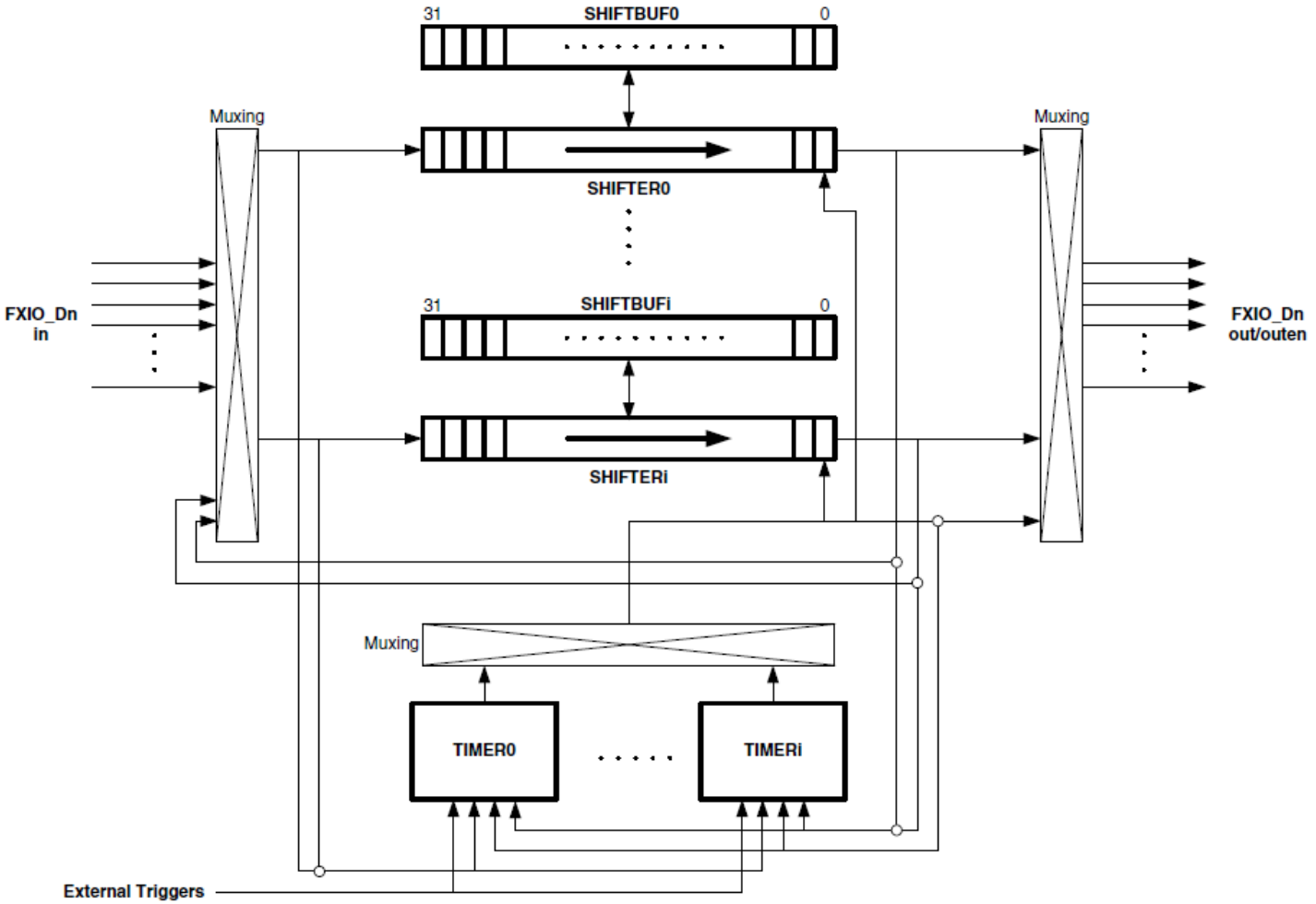
FlexIO

FlexIO main components:

- Shifters
- Timers
- Muxing interfaces

FlexIO: I/O Operation

- Each timer and shifter can be configured to use any FlexIO pin as:
  - Input
  - Output data
  - Output enable (Open Drain)
  - Bidirectional output

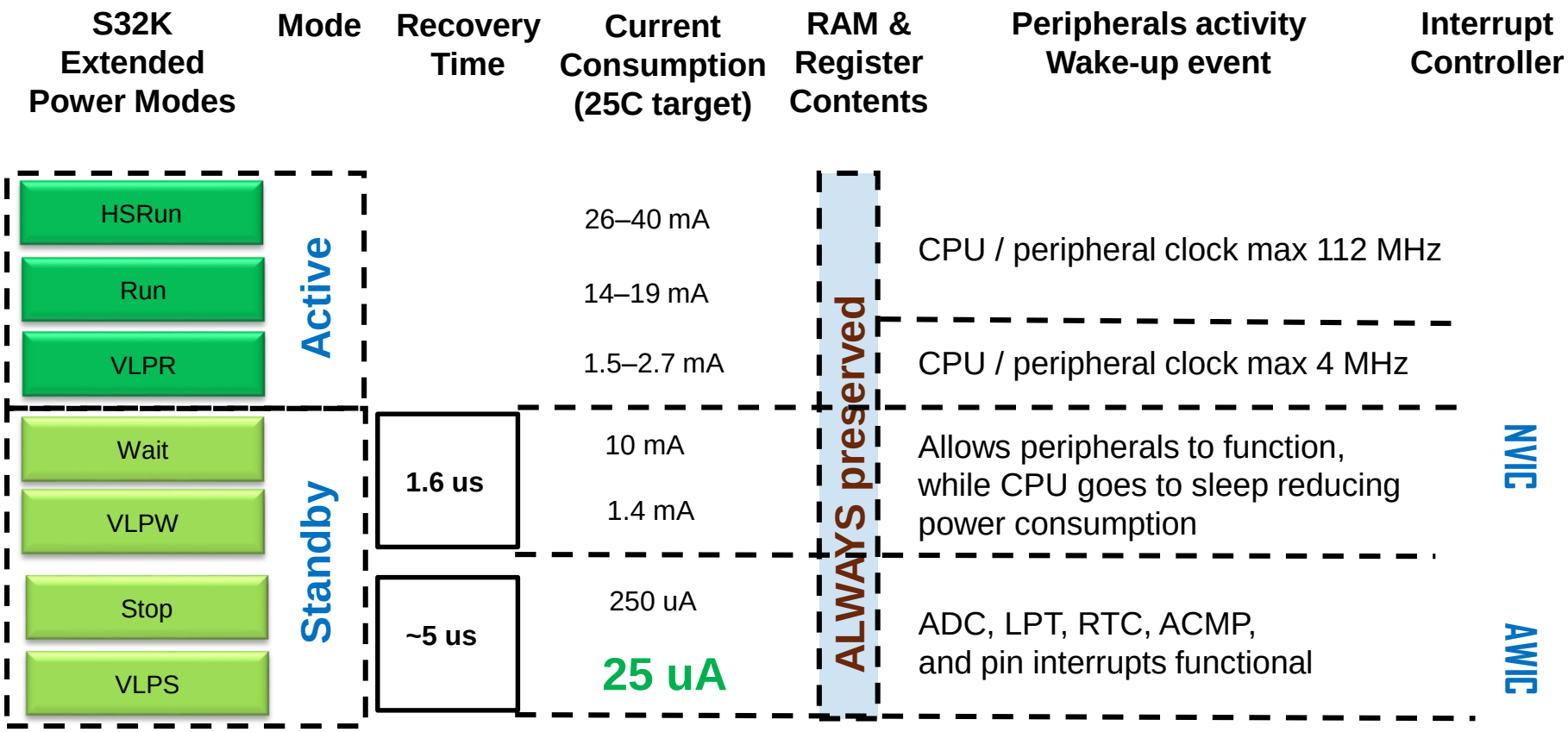


Autonomous Low Power Peripherals

- Strategy: switch on as little as possible:
  - CPU and clock tree is the most power hungry area on an MCU
  - Need to switch to OFF when possible
  - Put more intelligence into peripherals

- The S32K Autonomous peripherals help achieve this:
  - API / RTC — Autonomous Periodic Interrupt
    - Allows device to recover from very low power state at selectable time intervals
  - DMA — Direct Memory Access
    - Allows data transfer between peripherals minimizing CPU activity
  - ADC — Analog Digital Converter
    - Continual conversion while running in low power
    - Triggers wake-up when signal reaches certain level
  - ANLCMP — automatic cyclic compare functionality
  - LPUART — intelligent Lin management, available in low power modes
  - LPSPI — autonomous operation, regardless of core activity
  - LPIIC — ability to continue operation in low power modes
  - LINFlex — Intelligent LIN management, minimizing CPU interrupts

Low Power Modes



ALL memory and ***ALL*** registers and ***ALL*** I/O are ***ALWAYS*** maintained in ***ALL*** modes!!!

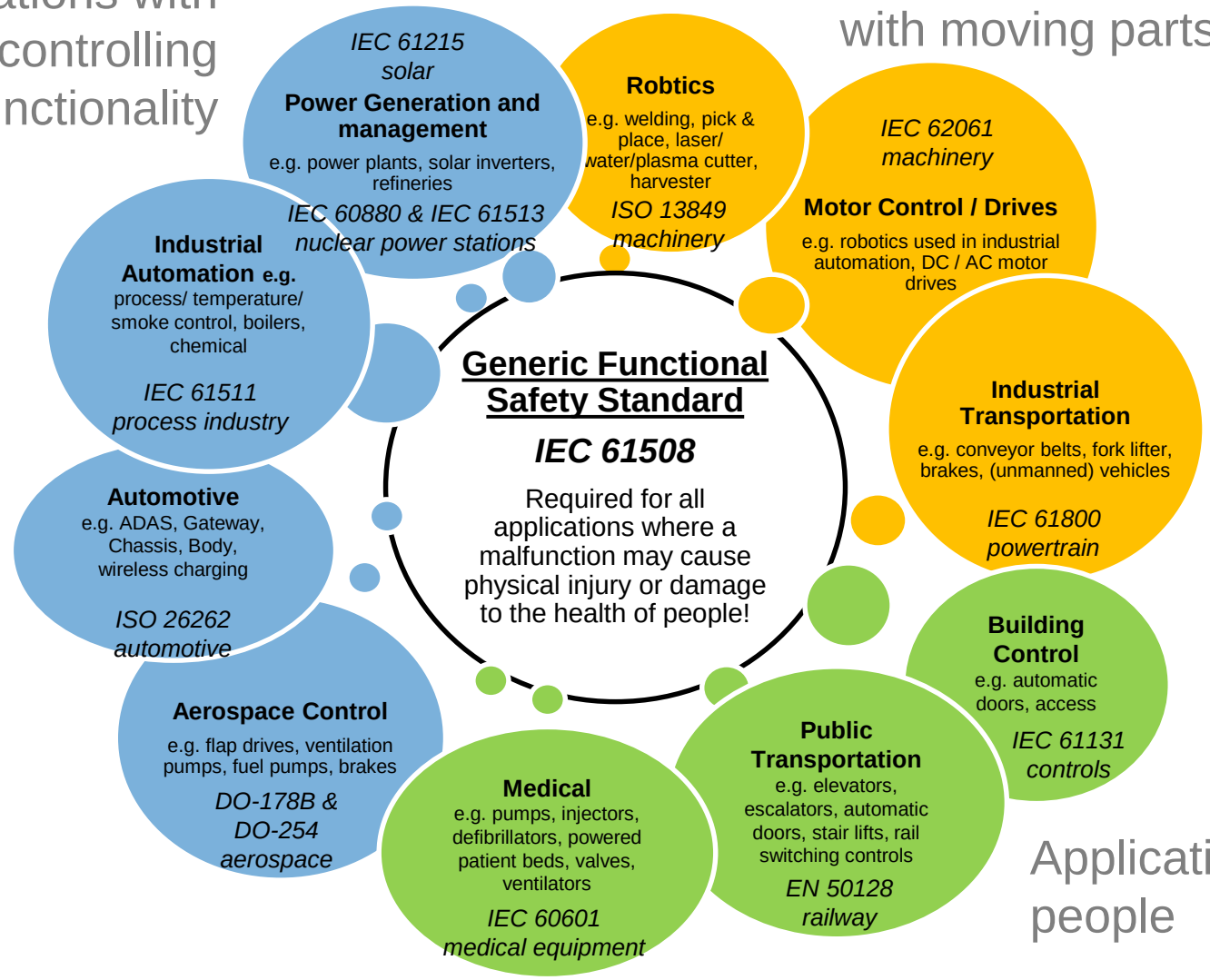
Functional Safety

- ✓ Prevents risks of electronic system malfunctions
- ✓ Measures failures, mitigates impact, predicts effects
- ✓ Industry defined standard: ISO 26262 for EE systems

Functional Safety is Required Everywhere!

Applications with controlling functionality

Applications with moving parts



Applications for people

SAFE ASSURE™

NXP

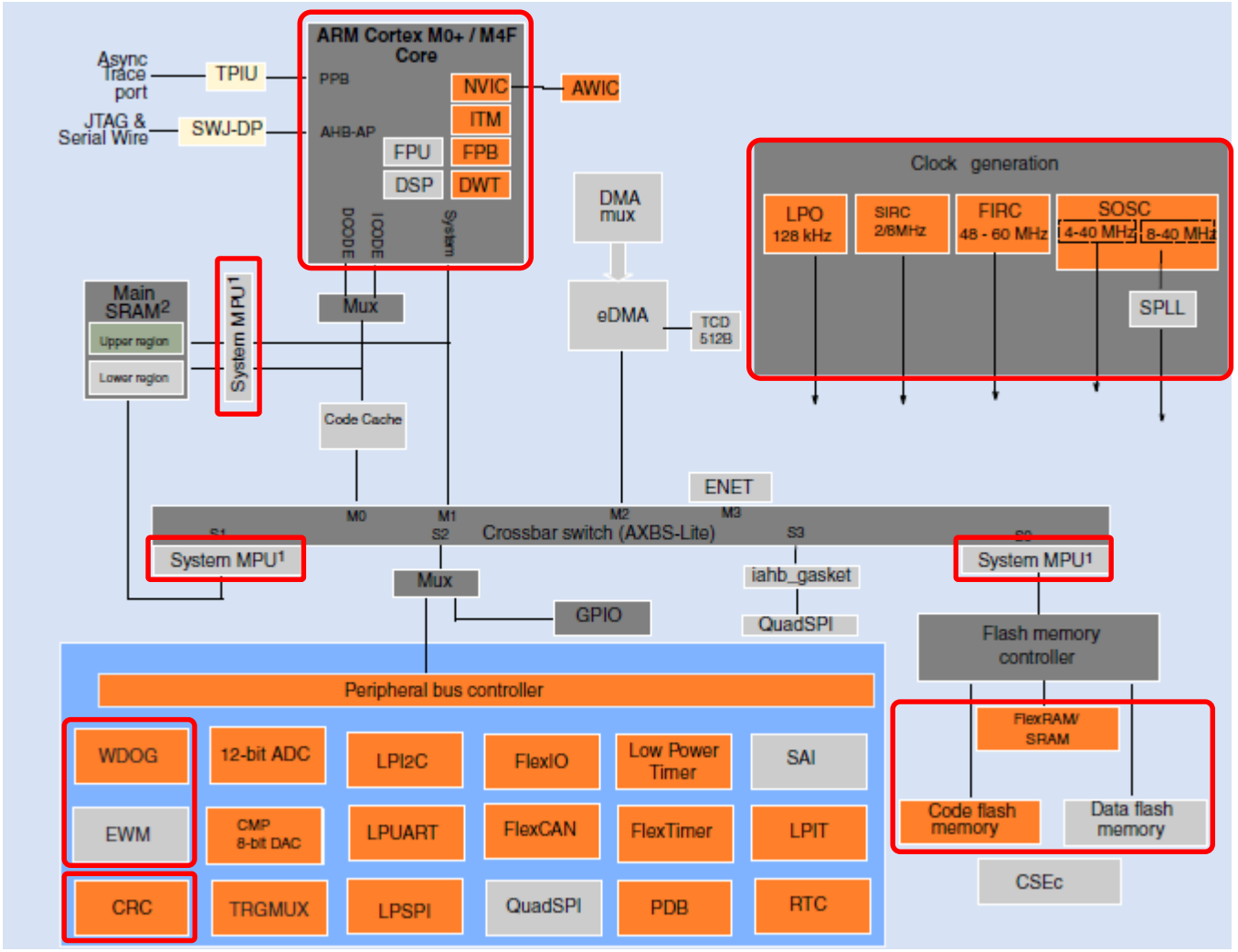
NXP SafeAssure™ Program http://www.nxp.com/safeassure

- NXP simplifies the process of system compliance for automotive and industrial functional safety standards
- Reduces the time and complexity required to develop safety systems that comply with ISO 26262 and IEC 61508 standards
- Supports the most stringent Safety Integrity Levels (SILs)
- Zero defect methodology from design to manufacturing to help ensure our products meet the stringent demands of safety applications

- Functional safety activities address:
  - Safety process (FMEA, FTA, FMEDA) integrated into development process
  - Safety hardware (safety manual) BIST, ECC, etc
  - Safety software (safety manual) Autosar MCAL, OS, core self tests, etc.
  - Safety support – training, documentation and tech support

S32K144 MCU – ASIL-B Safety Features

- ECC in Flash & RAM
- Power & Clock Monitoring
- Temporal protection – Software Watchdog
- MPU, CRC, register protection
- Diversity of safety levels
  - LPSPI or LPSCI v FlexIO: alternate communication paths / high parallelisation
  - Analogue input monitoring: signal measuring via completely independent system resources (references, peripherals) and monitoring protection schemes
- Safety Process
  - ISO 26262 development process
- Safety Support
  - FMEDA, Safety manual & Technical support
- Safety Software
  - S32K core self-test SW, PT lib



Safety – What you Get

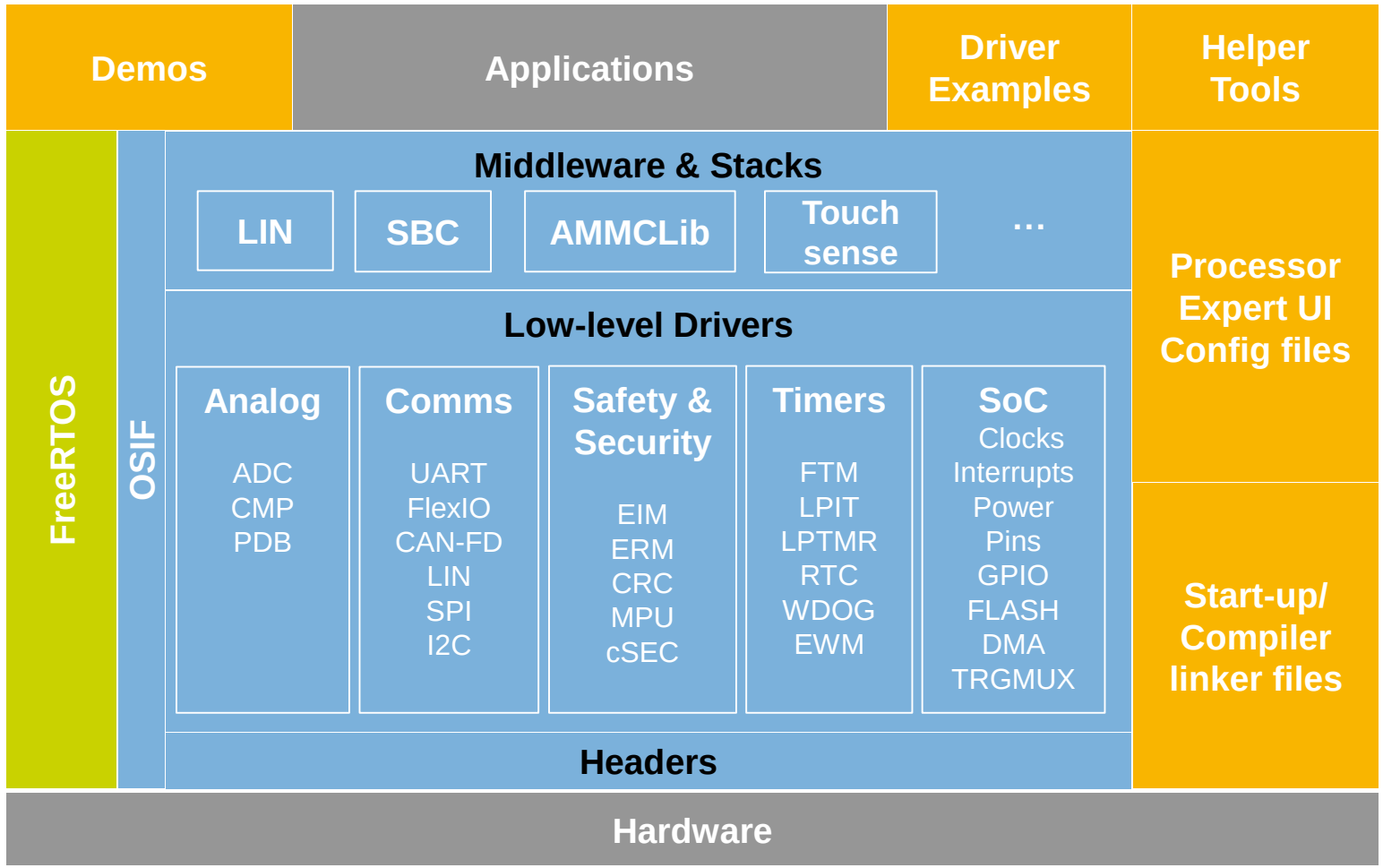
- To support the customer to build their safety system, the following deliverables are provided **as standard** for **all** ISO 26262 developed products.

- **Public Information available via NXP Website**
  - NXP Quality Certificates
  - Safety Manual
  - Reference Manual
  - Data Sheet

- **Confidential Information available under NDA**
  - Safety Plan
  - ISO26262 Safety Case
  - ISO26262-10 Table A.8 Checklist
  - Permanent Failure Rate data (Die & Package) - IEC/TR 62380 or SN29500
  - Transient Failure Rate data (Die) - JEDEC Standard JESD89
  - FMEDA & Report
  - DFA & Report
  - PPAP
  - Confirmation Measures Report (summary of all applicable confirmation measures)

S32K SDK

S32 SDK Product Overview



SW Quality Class

- Class B
- Class C
- Class D

Features

- Integrated **Non-Autosar SW Production-grade** software
- Graphical-based Configuration
- Layered Software Architecture
- Documented Source Code and Examples
- Integrated with S32 Design Studio and other IDEs
- Featuring various Middleware
- FreeRTOS integration
- Multiple toolchains supported
- Several examples and demos

S32K_SDK – Examples

New in v0.9.0 Beta release

Examples	Description
ADC Hardware Trigger	Uses PDB to trigger an ADC conversion with a configured delay and sends the result to host via LPUART.
ADC Software Trigger	Uses software trigger to periodically trigger an ADC conversion and sends the result to host via LPUART.
CMP	Configures the analog comparator to compare the input from the potentiometer with the internal DAC (configured to output half of the reference voltage) and shows the result using the LEDs found on the board.
CRC	The CRC is configured to generate the cyclic redundancy check value using 16 and 32 bits wide result
eDMA	Demonstrates the following eDMA use cases: single block memory to memory transfer, a loop memory to memory transfer, memory to memory transfer using scatter/gather, LPUART transmission/reception using DMA requests.
EWM	Shows the usage of the EWM driver.
Flash	Writes, verifies and erases data on Flash.
FlexIO (I2C Example)	Demonstrates FlexIO I2C emulation. Use one instance of FlexIO and one instance of LPI2C to transfer data on the same board.
FlexIO (SPI Example)	Demonstrates FlexIO SPI emulation for both master and slave configurations. Use one instance of FlexIO to instantiate master and slave drivers to transfer data on the same board.
FlexIO (UART Example)	Demonstrates FlexIO UART emulation examples will show the user how to configure and transfer data using the UART protocol emulated with FlexIO.
FlexIO (I2S Example)	Demonstrates FlexIO I2S emulation examples will show the user how to configure and transfer data using the I2S protocol emulated with FlexIO.
FTM (PWM)	Uses FTM PWM functionality using a single channel to light a LED on the board. The light's intensity is increased and decreased periodically.
FTM (Combined PWM)	Uses FTM PWM functionality using two combined channels to light two LEDs on the board with opposite pulse width. The light's intensity is increased and decreased periodically.

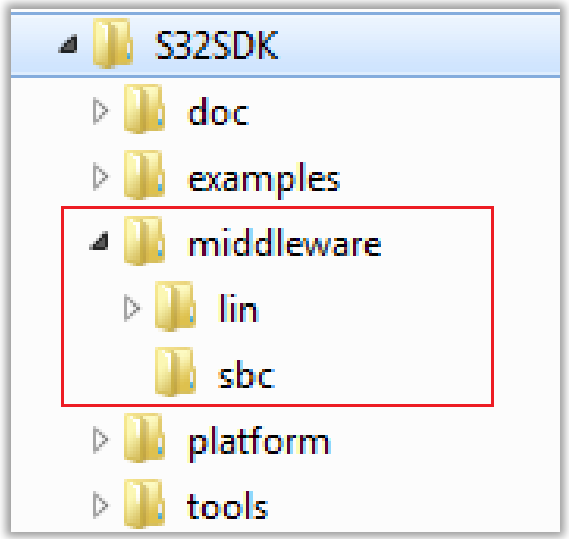
Examples	Description
FTM (Timer)	Uses FTM Timer functionality to trigger an interrupt at a given period which toggles a LED.
FTM (Signal Measurement)	Using one FTM instance the example application generates a PWM signal with variable frequency which is measured by another FTM instance configured in signal measurement mode.
LPI2C (Master Example)	Shows the usage of the LPI2C driver in Master configuration
LPI2C (Slave Example)	Shows the usage of the LPI2C driver in Slave configuration
LPIT	Shows how to initialize the LPIT to generate an interrupt every 1 s. It is the starting point for any application using LPIT.
LPSPI	Uses one instance of the LPSPI as slave to send ADC data to the master LPSPI instance which is on the same board. The master uses data received to feed a FlexTimer PWM.
LPTMR (Timer)	Exemplifies to the user how to initialize the LPTIMER so that it will generate an interrupt every 1 second. To make the interrupt visible a LED is toggled every time it occurs.
LPTMR (Pulse Counter)	Shows the LPTIMER pulse count functionality by generating an interrupt every 4 rising edges.
LPUART	Simple example of a basic echo using LPUART.
MPU	Configures MPU to protect a memory area and demonstrates that read access is correctly restricted.
PDB	Configures the Programmable Delay Block to generate an interrupt every 1 second. This example shows the user how to configure the PDB timer for interrupt generation. The PDB is configured to trigger ADC conversions in ADC_HwTrigger_Example.
RTC	Show the frequently used RTC use cases such as the generation of an interrupt every second and triggering an alarm.
WDOG	Shows the basic usage scenario and configuration for the Watchdog.

S32K_SDK – Demos

New in v0.9.0 Beta release

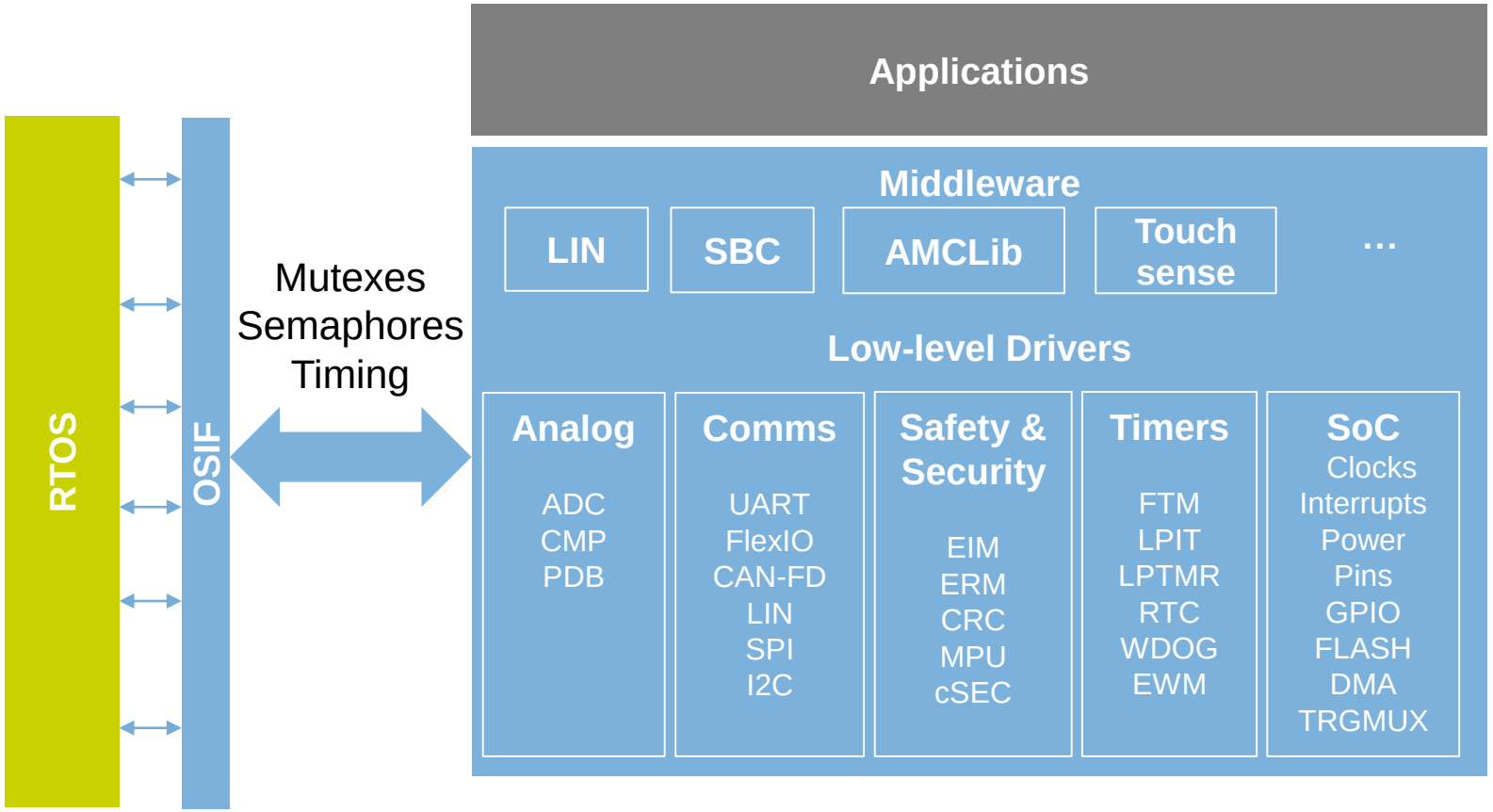
Demos	Description
Blinking LED	This is a simple application created to show configuration for supported compilers (GCC, IAR, GHS …).
Clock setup	Start-up project with maximal clock settings for the platform.
Secure CAN	Uses two boards to demonstrate FlexCAN functionality with Flexible Data Rate on. LEDs on a board are toggled depending on the buttons actioned on the other board. Also demonstrates the use of SBC driver to configure the CAN transceiver from EVB board. Secure communication can be switched on/off by pressing a button on the board. When enabled, commands sent to the other board are encrypted using the CSEc driver. Received commands will be decrypted and, if valid, the corresponding LED will be toggled.
FTM-PDB-ADC demo	This application will show how to use the hardware features to offload the CPU
FreeRTOS	This demo application demonstrates the usage of the SDK with the included FreeRTOS. Uses a software timer to trigger a led and waits for a button interrupt to occur.
LIN	This demo application shows the usage of LIN stack. There are slave and master applications, both configuration having baremetal and FreeRTOS variants.
ADC Low Power	This demo shows the user how to reduce CPU overhead and power usage by triggering ADC conversions with the LPIT via TRGMUX. The CPU is set in the STOP mode via the Power Manager API, with the wakeup condition being the validity of the ADC conversion result, the latter being a value greater than half of the ADC reference voltage achieved by using the hardware compare functionality. If the condition is met, the value in the form of a graph is sent using LPUART and DMA to further reduce the CPU usage.
FreeMASTER BDM	This demo uses the FreeMASTER Run-Time Debugging Tool to visualize ADC conversions and allows the user to monitor the ADC sampling rate for different ADC configurations (ADC sampling time and resolution can be controlled through FreeMASTER Variable Watch). The application uses BDM for communication.

S32 SDK– Middleware & Stacks Layer



- Platform independent API/implementation
- Calls SDK device drivers API
- Configurable with Processor Expert

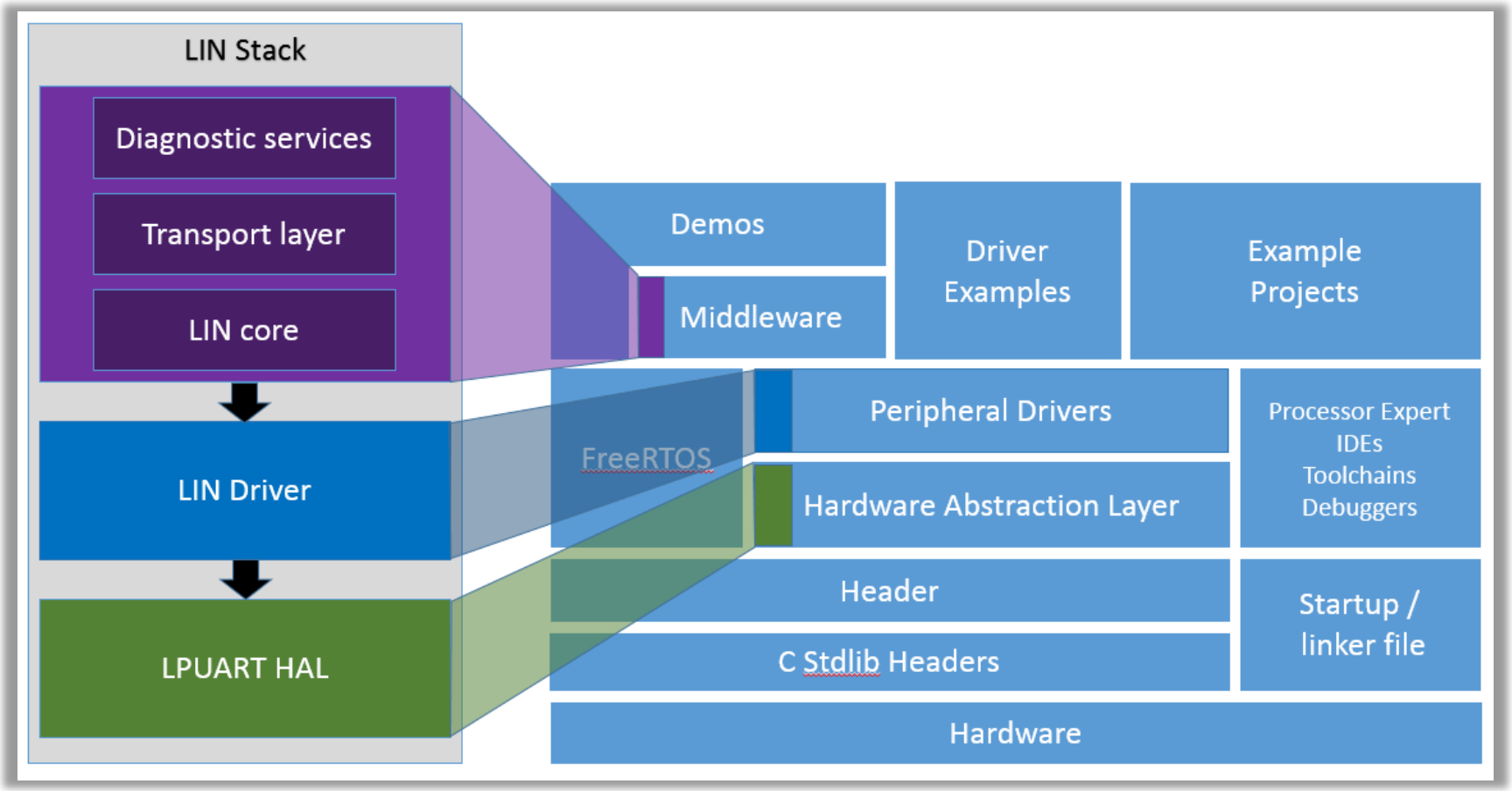
OS Interface Layer (OSIF)



Features

- **RTOS independent**
  Provides some typical OS services to :
  - Drivers
  - Middleware & Stacks
- **Automatic configuration**:
  OSIF added by PEx framework
- **Internal interface**
  Application uses this indirectly via SDK elements (drivers, middleware …)

LIN Protocol Stack

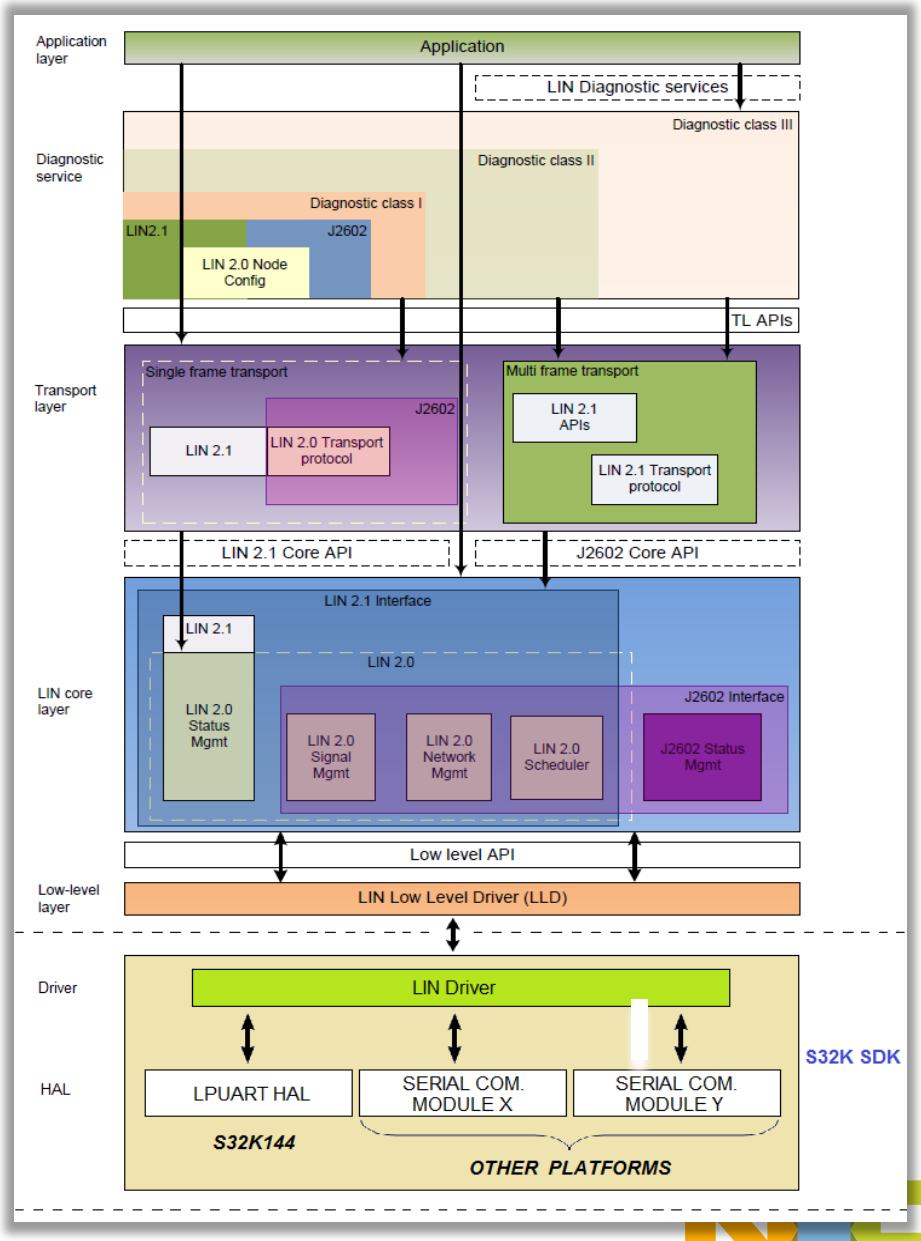


Features:

- LIN2.x and J2602 standards
- Master/slave support
- Diagnostic classes I, II and III
- Autobaud for slave nodes
- Layered software architecture:
  - Low level layer
  - Core API layer
  - Transport layer
  - Diagnostic services
- Node Configuration Tool (NCF) – PC based script for LIN Stack configuration generation
- LIN Protocol demo applications

LIN Protocol Stack (cont')

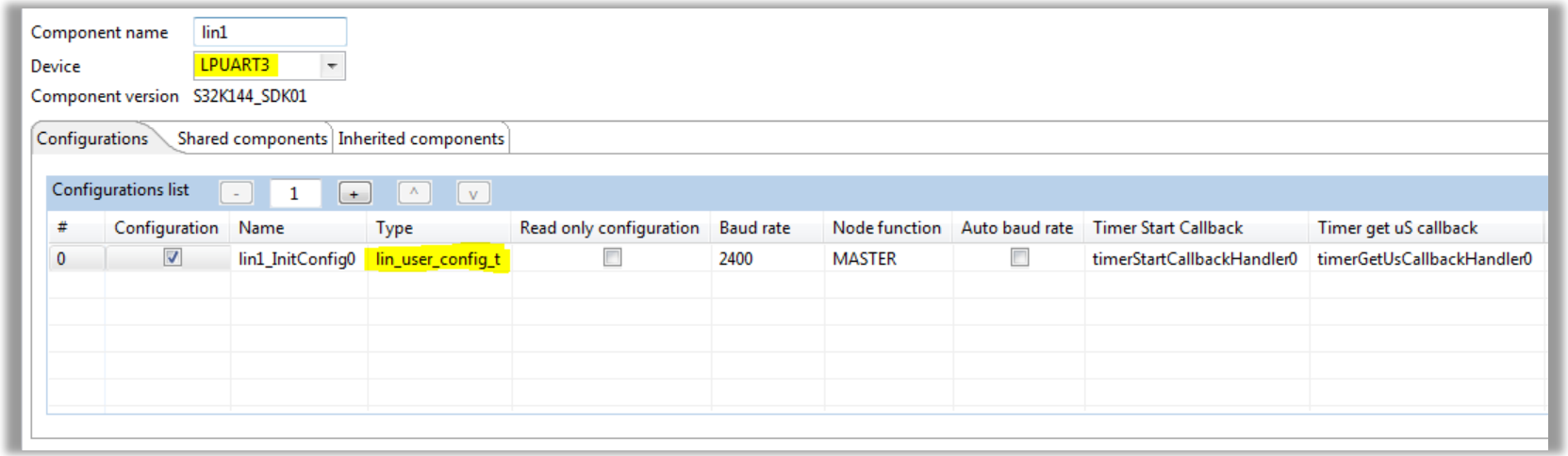
- **Application** – user code
- **Diagnostic services** – implements all diagnostic classes defined in LIN specification
- **Transport layer** – Single/multi frame transmission
- **Core layer** – implements API functions as defined by the LIN specification
- **Low level layer** – wrapper functions for hw-dependent implementation
- **Driver** – basic node/frame configuration & control
- **HAL** – low-level IP specific functions – register access, get/set hw features



LIN Protocol Stack Configuration

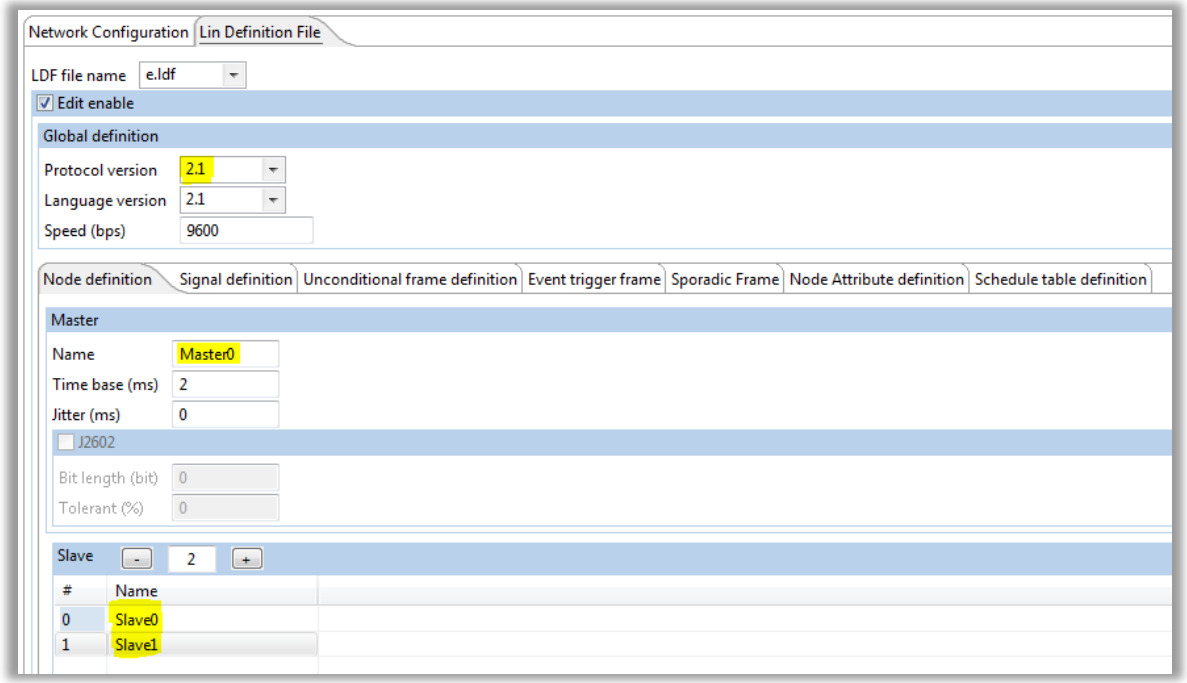
- LIN Driver
  - Inherits serial communication HAL
  - Configures LIN node
  - Generates configuration structure, as defined in fsl_lin_driver.h

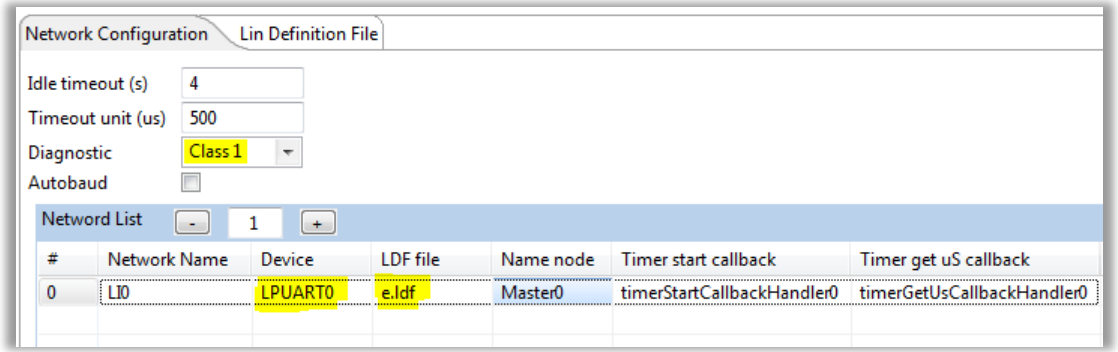
```
/*! lin1 configuration structure */
lin_user_config_t lin1_InitConfig0 = {
  .baudRate = 2400U,
  .nodeFunction = MASTER,
  .autobaudEnable = false,
  .timerStartCallback = timerStartCallbackHandler0,
  .timerGetUsCallback = timerGetUsCallbackHandler0,
};
```



LIN Configuration Component

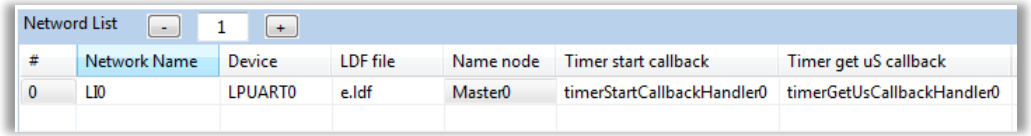
- LIN Stack
  - Network configuration
  - LIN cluster definition:
    - Nodes
    - Signals
    - Frames
    - Node attributes
    - Schedule tables





LIN Configuration Component

- LIN Stack
  - UI changes update npf/ldf on-the-fly
  - PEx script calls jar to generate .c & .h configuration files, based on npf/ldf – feeds into NCF tool




```
/*** NETWORK DEFINITION ***/
network {
    idle_timeout                    = 4 s;
    diagnostic_class                = 1;
    resynchronization_support       = no;
    autobaud_support                = no;
    LI0 {
        node                        = Master0;
        file                        = "e.ldf";
        device                      = LPUART0;
    }
}
```

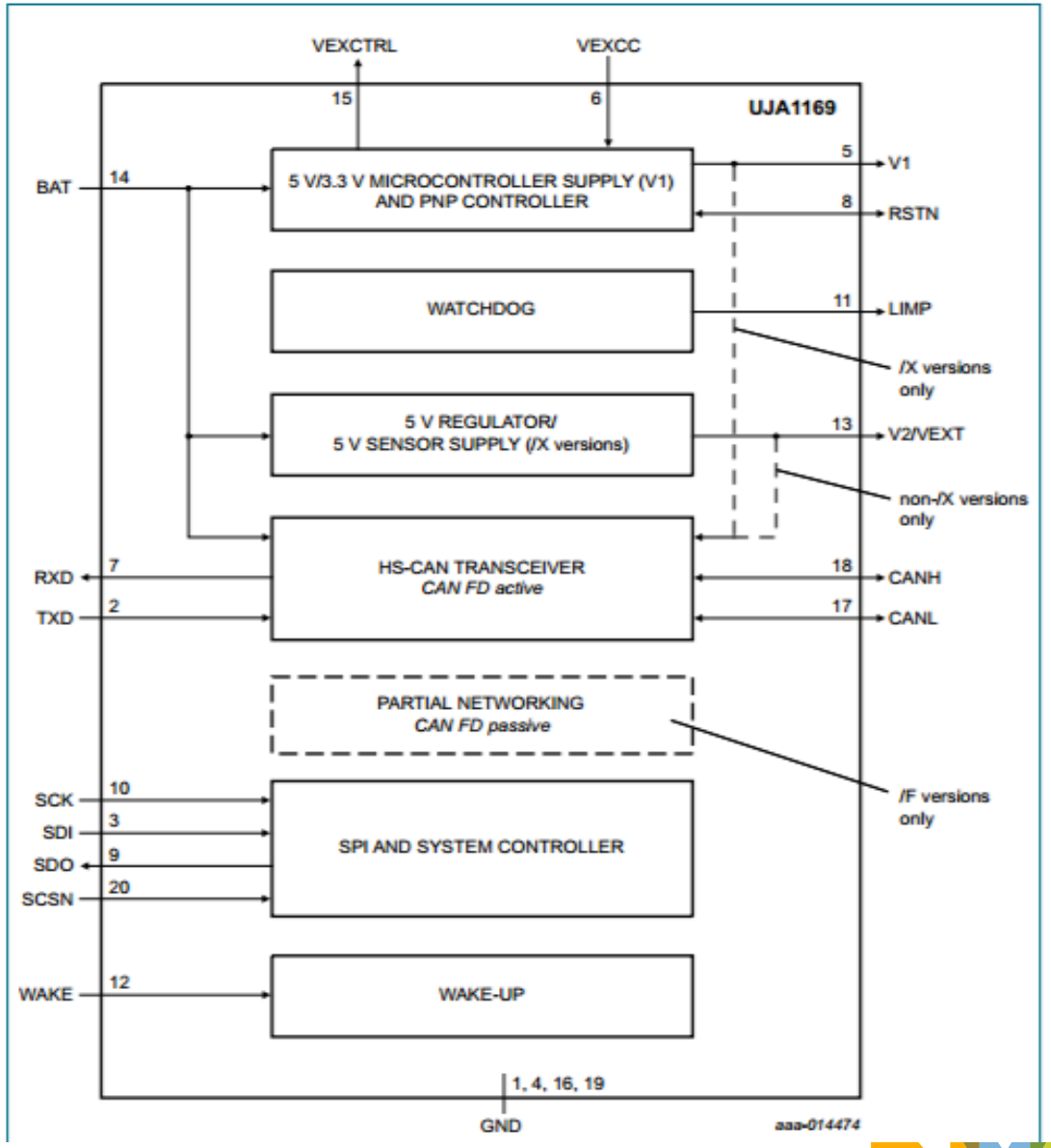
```
#define TIME_OUT_UNIT_US 500
#define TIME_OUT_UNIT_S  0.0005
#define TIME_BASE_PERIOD   500
#define HARDWARE_INSTANCE_COUNT    4
/* Define operating mode */
#define _MASTER_MODE_      1
#define _SLAVE_MODE_       0
#define LIN_MODE   _MASTER_MODE_
/* Define protocol version */
#define PROTOCOL_21        0
#define PROTOCOL_J2602     1
#define PROTOCOL_20        2
#define LIN_PROTOCOL    PROTOCOL_21
#define LIN_NUM_OF_IFCS  1       /* For master */

/* max idle timeout for all networks = idle_timeout_value*1000000/time_base_period */
#define _MAX_IDLE_TIMEOUT_    8000      /* idle_timeout_value = 4s */

/*********************************************************************/
/***************          Diagnostic class selection  ****************/
/*********************************************************************/
#define _DIAG_CLASS_I_          0
#define _DIAG_CLASS_II_         1
#define _DIAG_CLASS_III_        2

#define _DIAG_CLASS_SUPPORT_    _DIAG_CLASS_I_
```

SBC – UJA1169 family features

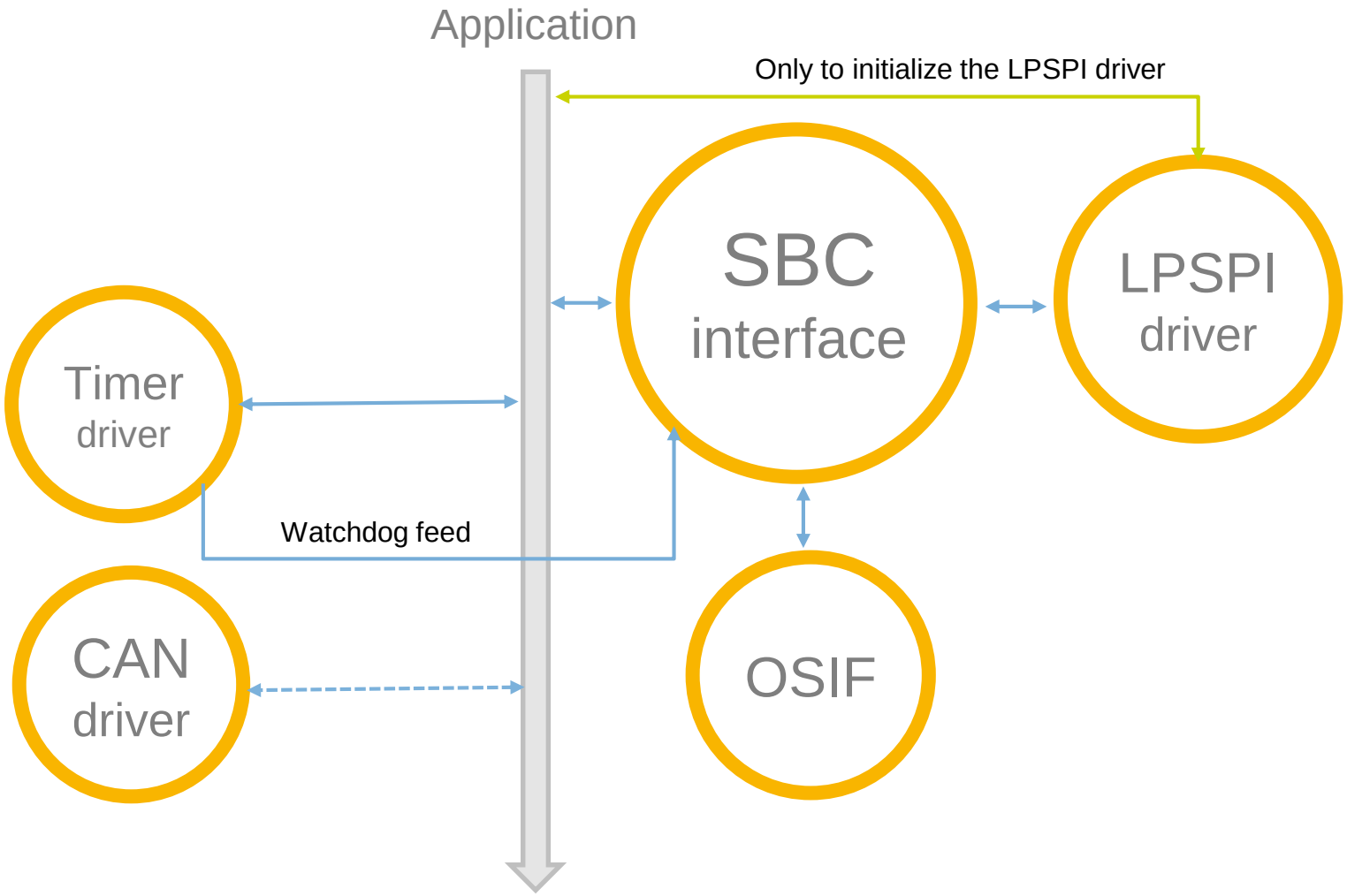
- **Transceiver** for CAN FD up to 2MBs and partial networking support

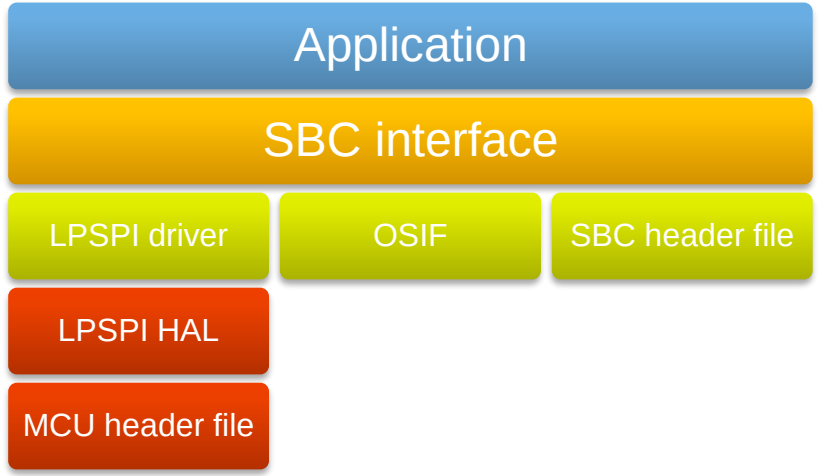
- **Voltage regulators:**
  - LDO (5V/3.3V 250mA)
  - CAN supply (5V 100mA)
  - Sensor supply (5V 100mA)

- **Power management**
  - Modes: Normal, Standby, Sleep
  - Wake-up sources (pin, CAN frame)



SBC Application Flow and Software Architecture



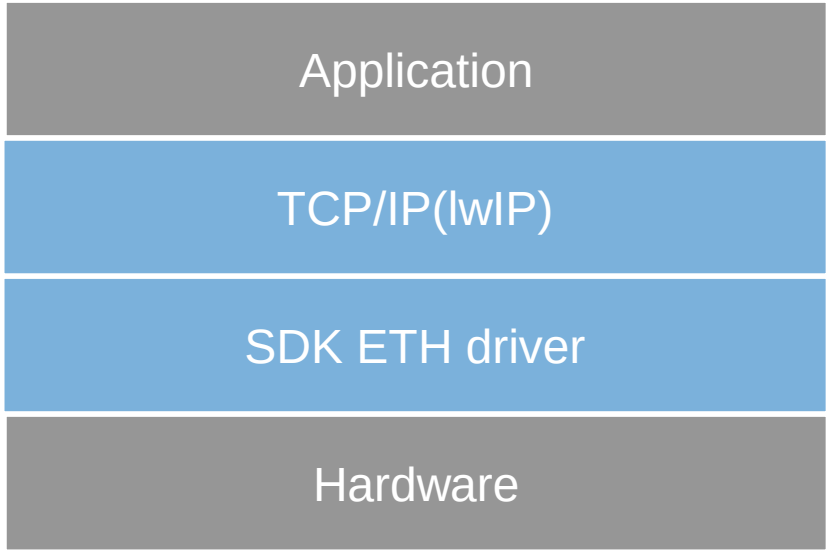


TCP/IP Protocol Stack

lwIP(free, complete, widely used, no OS dependency, compatible with AVB, compatible with SSL/TLS stacks, also used by Kinetis SDK 2.0, not applicable to ASR) + **Class C, configurator**

Main features include:
- Protocols: IP, ICMP, UDP, TCP, IGMP, ARP, PPPoS, PPPoE
- DHCP client, DNS client, AutoIP/APIPA (Zeroconf), SNMP agent (private MIB support)
- APIs: specialized APIs for enhanced performance, optional Berkeley-alike socket API
- Extended features: IP forwarding over multiple network interfaces, TCP congestion control, RTT estimation and fast recovery/fast retransmit
- Addon applications: HTTP server, SNTP client, SMTP client, ping, NetBIOS nameserver

- lwIP is licenced under a BSD-style license



lightweight IP – SDK integration

Time Synchronization (802.1AS, gPTP)

gPTP

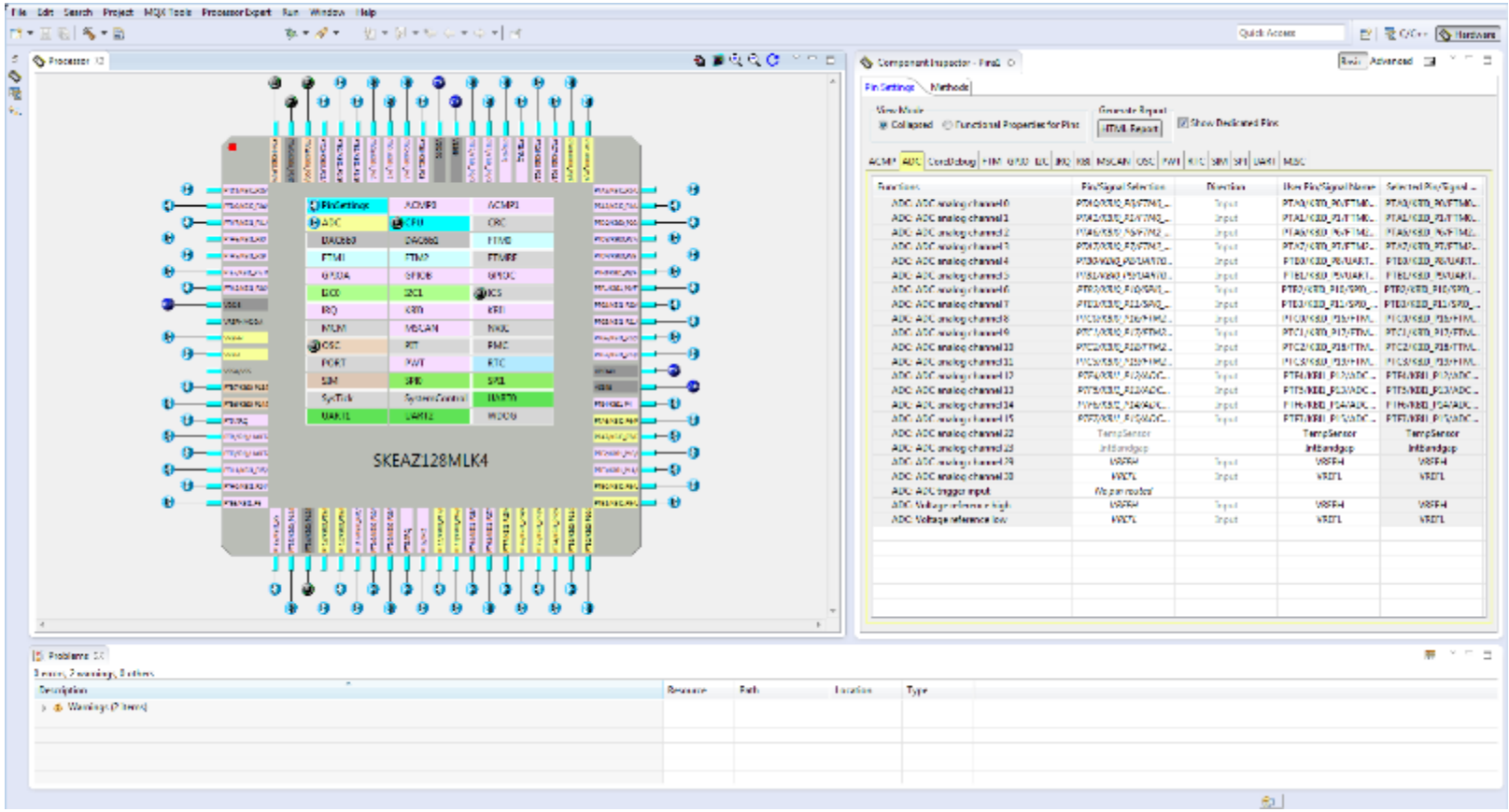
Implementation of IEEE 802.1AS-2011, layer 2 protocol, subset of IEEE 1588

- Applications
  - Synchronous Audio/Video recording/playback
  - Noise cancellation
  - Synchronous sensor sampling/sensor fusion

- gPTP achieves synchronization precision of roughly 100 ns/hop, depending on implementation (time stamp accuracy, path delay symmetry…)

Processor Expert Tool – Initialization and SDK configuration

- GUI based tool for MCU configuration
- Configure MCU clocks
- Setup peripheral initialization configurations
- Configure drivers in the SDK for S32K MCUs
- Provides consistency checks for invalid configuration detection
- Generates code that works with static initialization functions
- Pin-Muxing tool:
  - Assign functions to pins
  - Set up pin electrical characteristics
  - Generate code to initialize pins to desired functions



DEVELOPMENT TOOLS

S32K1 Enablement

Software

- *Free* S32 Design Studio IDE – Eclipse based, supports multiple compiler & debugger plug-ins
- *Free* S32 SDK – Automotive-grade, pre-qualified, multiple low-level drivers, optional middleware (LIN, NFC, TSI),
- Math, Motor Control & Core Self Test Libraries, MATLAB based design tools

Hardware

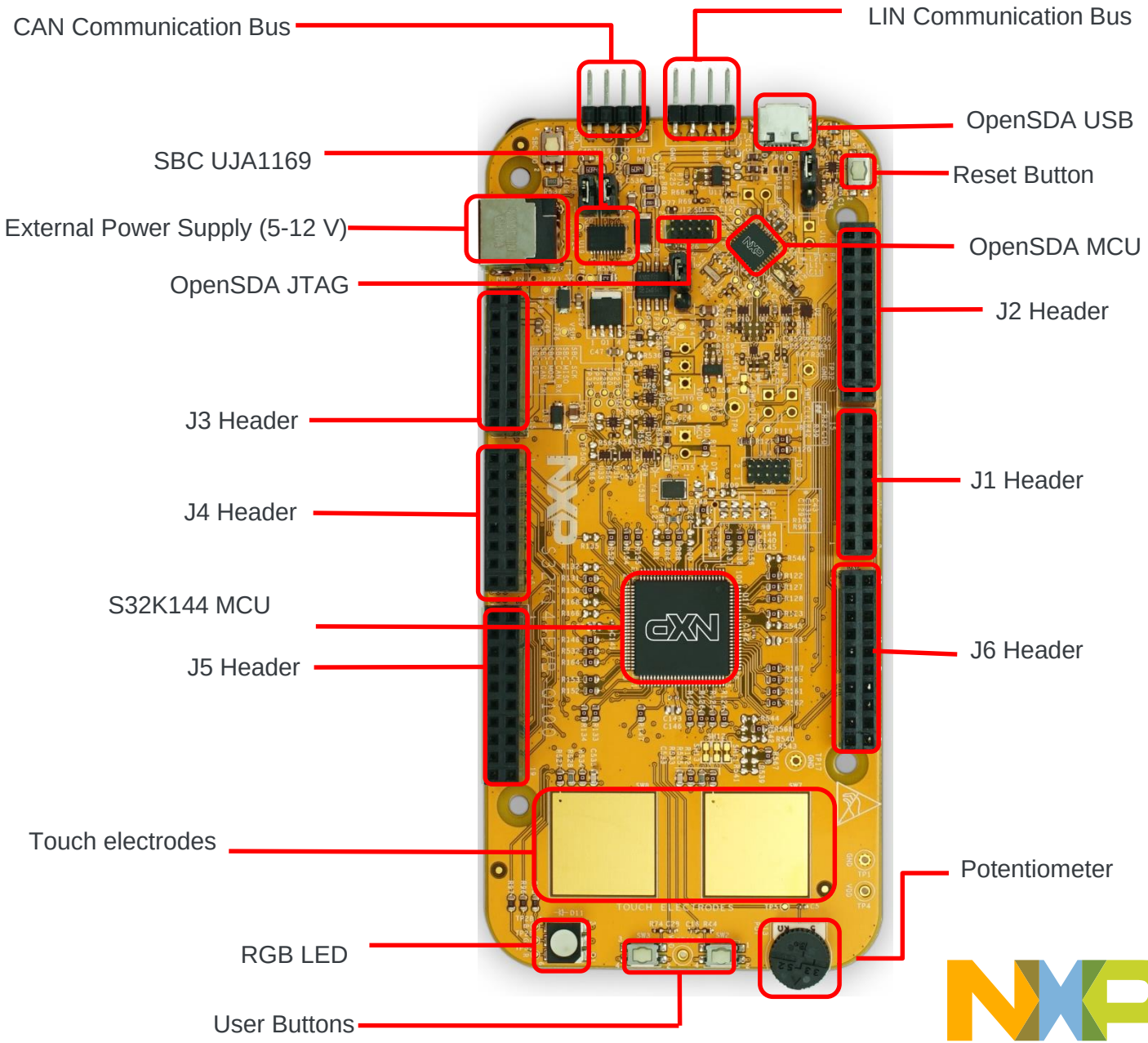
- S32K144EVB-Q100X / -Q100
- NXP & 3rd party SW tool compatible, out-of-box examples for fast start-up & prototyping
- Arduino UNO compatible with expansion “shield” support
- $49 resale

Ecosystem

- Premium level IDE, complier & debugger tools
- NXP & 3rd party MCAL / AUTOSAR + *new* ARCCORE Starter Kit
- S32K, S32DS & SDK Communities http://www.nxp.com/community

S32K144 MCU – EVB

- $49 USD SRP
- Supports S32K144 100LQFP MCU
- Small form factor size 6” x 4”
- Arduino™ UNO footprint-compatible with expansion “shield” support
- Integrated open-standard serial and debug adapter (OpenSDA) with support for several industry-standard debug interfaces
- Easy access to all the MCU I/O pins for prototyping
- On-chip connectivity for CAN, LIN, UART/SCI
- SBC UJA1169 and LIN phy TJA1027
- Flexible power supply options
  - microUSB or external 12 V power supply

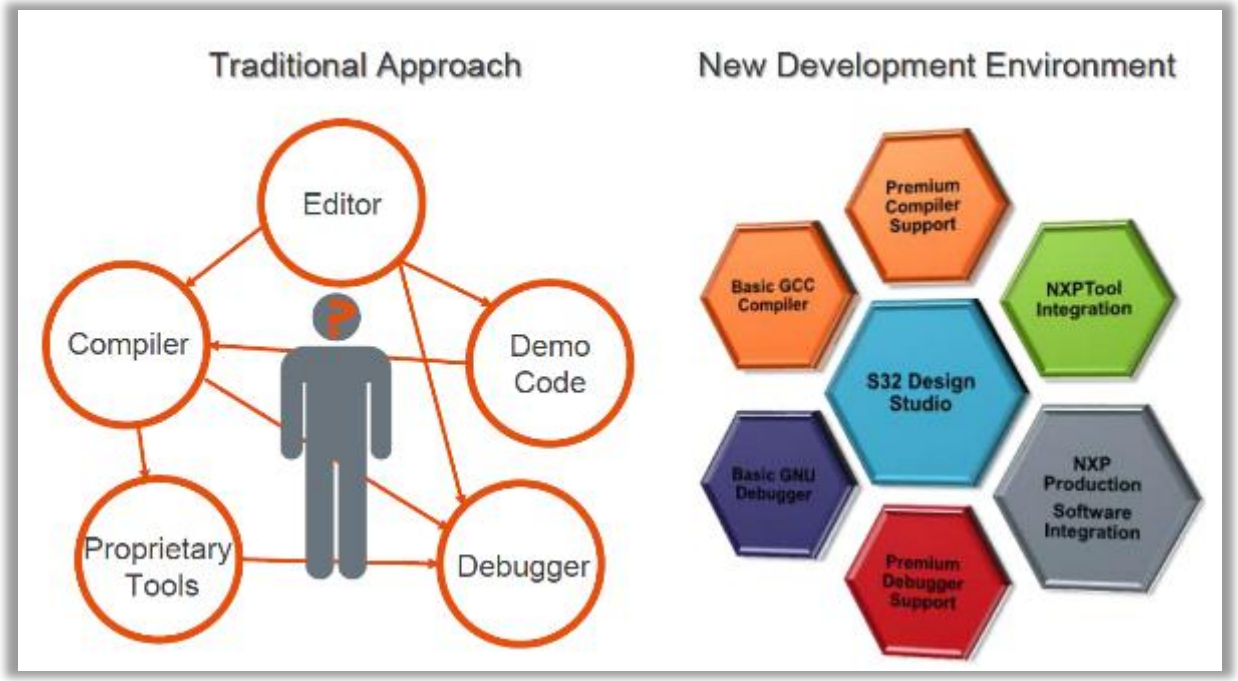


S32K14x – EVBs & Documentation

Dev Tool	MCU	Silicon	Availability
S32K144EVB-Q100X	K144, 100LQFP	Rev2.0 (0N47T)	~500 left in stock, will move to non X version once finished
S32K144EVB-Q100	K144, 100LQFP	Rev2.1 (0N57U)	T1 Auto ctms: now Disty ctms: contact AMP Mktg
S32K148EVB (pn tbc)	K148, 176LQFP	Rev (x)	144LQFP: now, T1 ctms 176LQFP: Q3 T1 ctms, Q4 Disty ctms
S32K142EVB (pn tbc)	K142, 100LQFP	Rev (x)	T1 Auto ctms: now Disty ctms: contact AMP Mktg

S32 Design Studio IDE - Overview www.nxp.com/S32DS

- Supports S32K and Power Architecture (MPC) products
- Free of charge, unlimited code size
- Eclipse based environment
- GNU compiler & debugger integrated
- S32 SDK integrated (graphical configuration)
- Processor Expert integrated (automatic code generator)
- Can use with 3rd party compliers & debuggers (IAR) via Connection Utility
- Not a replacement for NXP's CodeWarrior IDE
- Does not compete with premium 3rd party IDEs



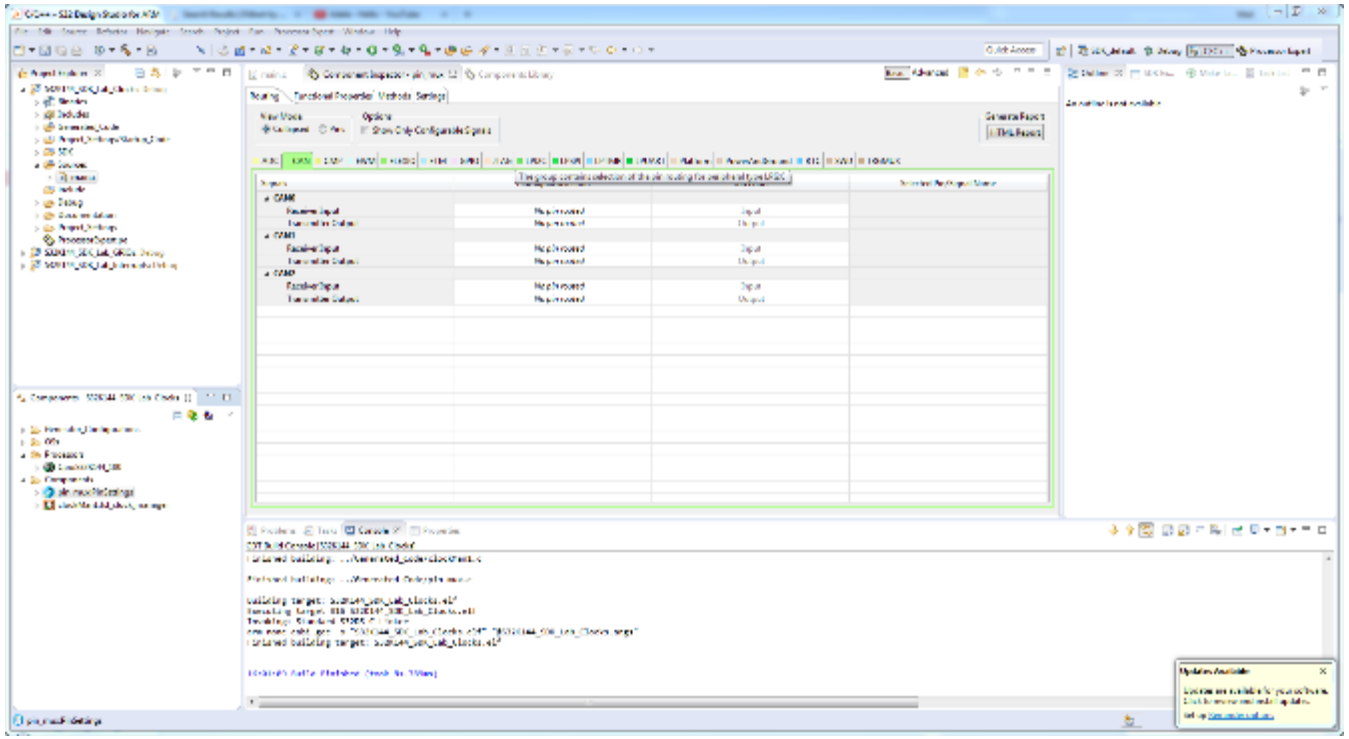
S32 Design Studio IDE – graphical configuration environment

1. **Create a new S32DS IDE New Project Wizard**
   - Select MCU and target package

2. Select **Compiler**
   - GCC or 3rd party Premium Compiler (IAR and GHS)

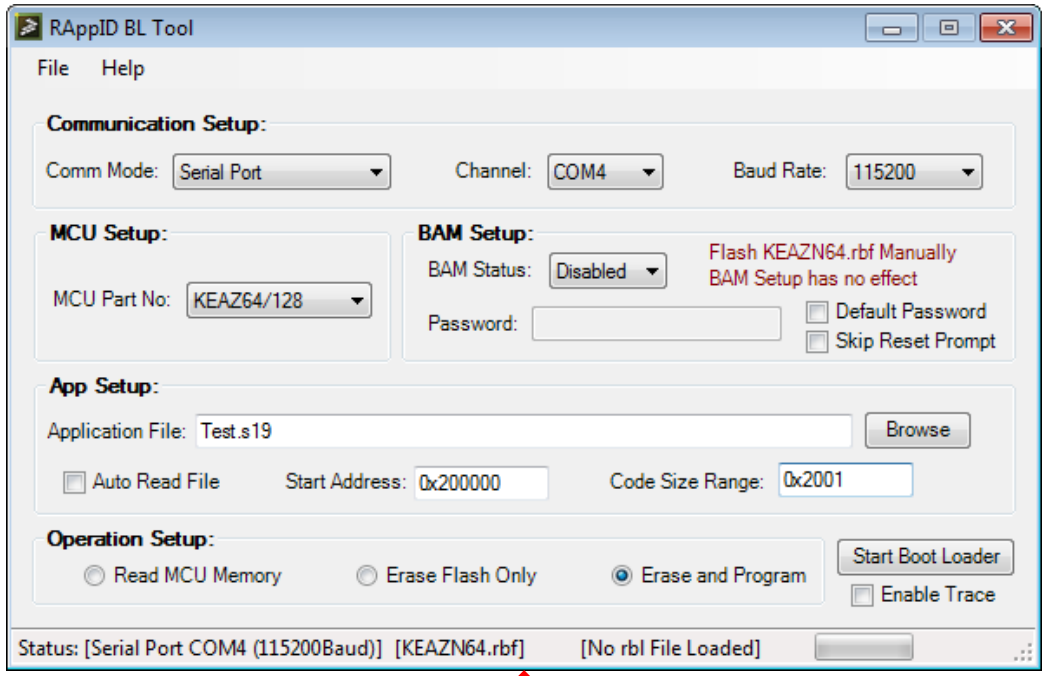
3. Select **Integration NXP tools**
   - Processor Expert
     - Pin Mux Tool
     - Device Configuration
     - SDK Configuration
   - Bootloader
   - FreeMASTER Embedded

4. Select **Software Integration**
   - S32K SDK Integrated with-in the tools
   - KEA SDK Integrated with-in the tools
   - Automotive Math and Motor Control Libraries (AMMCLib)



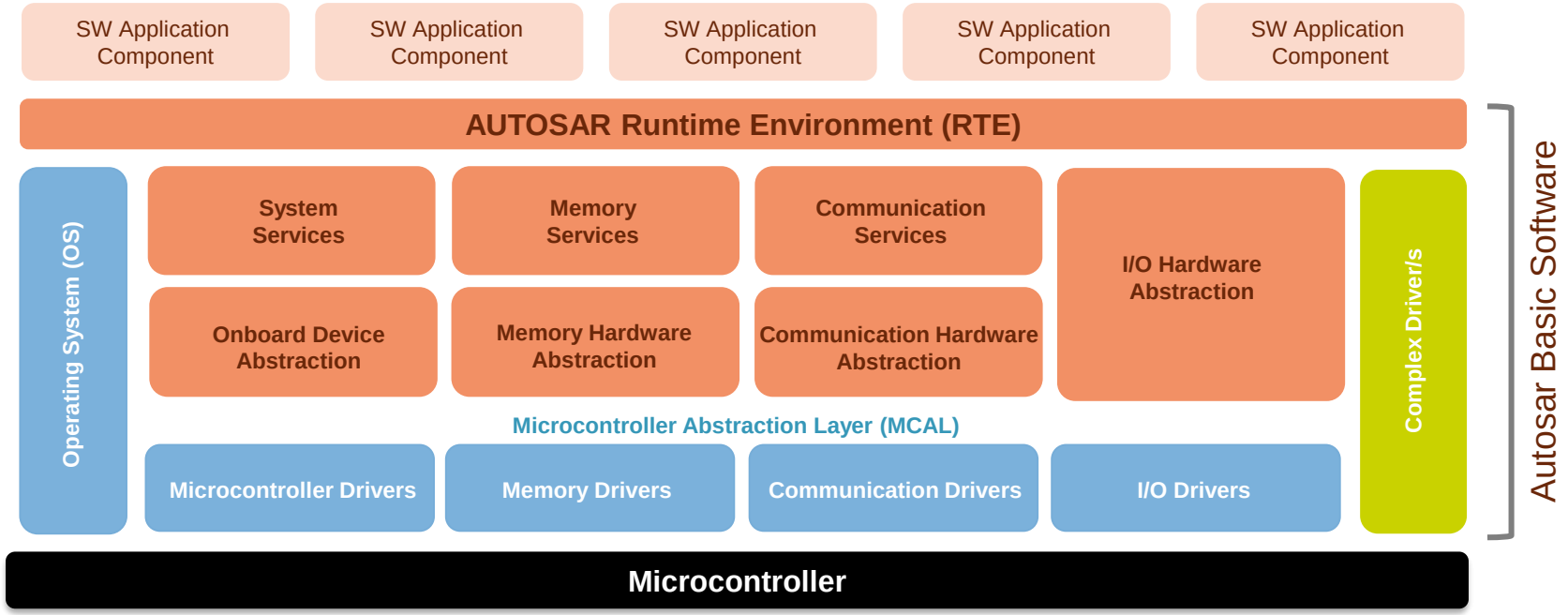
Bootloader Utility

- Boot loader utility to download s19 record to a module
- Bootloader resident in flash memory
- Works through Serial or CAN Interface
- Works through GUI or command line interface
- S32DS IDE makes it very easy to place the bootloader in flash memory for easy use of bootloader utility later



S32K1xx SW Architecture — AUTOSAR

- **NXP Standard Products** — MCAL (source code), OS (source code) and Config Tool (executable) for MCAL and OS.
- **Partner Products** (Elektrobit, Vector, KPIT, etc.) — the rest of AUTOSAR basic software as needed & Integration Services (NXP IP + Partner IP + Customer IP)
- **Complex Drivers** — custom software offered by NXP Consulting & Professional Engineering Services



Developed in respect to **AutoSAR4.x**, **SPICE Level 3** and **ISO26262** standards.

S32K Software Business Model – 5 Components:

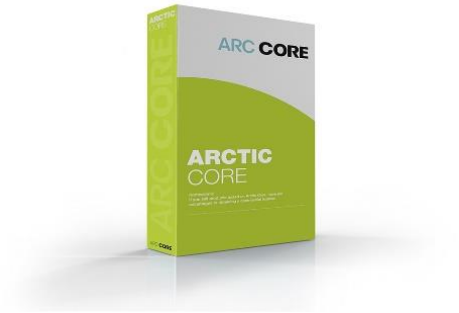
1. DISM Pack	Initial delivery, Support, Updates, Fixes, Quality Reports, Development Lic. etc. ***Annual Renewal***
2. Production Licenses	Production Rights ***One time Purchase***
3. Standard Services	Frozen Branch Releases, Compiler Tests ***Purchase as needed***
4. Automotive S/w Services	Complex Driver / Custom Software Development, Testing, Integration, Consulting ***As needed per customer's requirements***
5. Automotive SDK	Software Development Kit with HAL driver library and FreeRTOS port ***Free of charge offer***

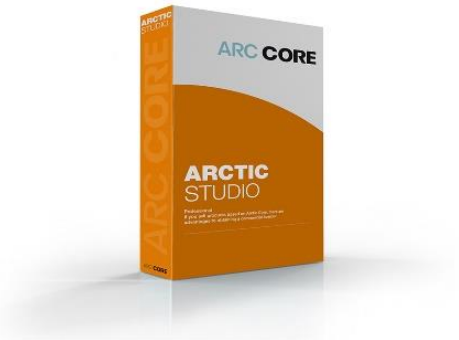
S32K AUTOSAR Starter Kit from ARCCORE

New AUTOSAR Starter Kit for S32K144EVB

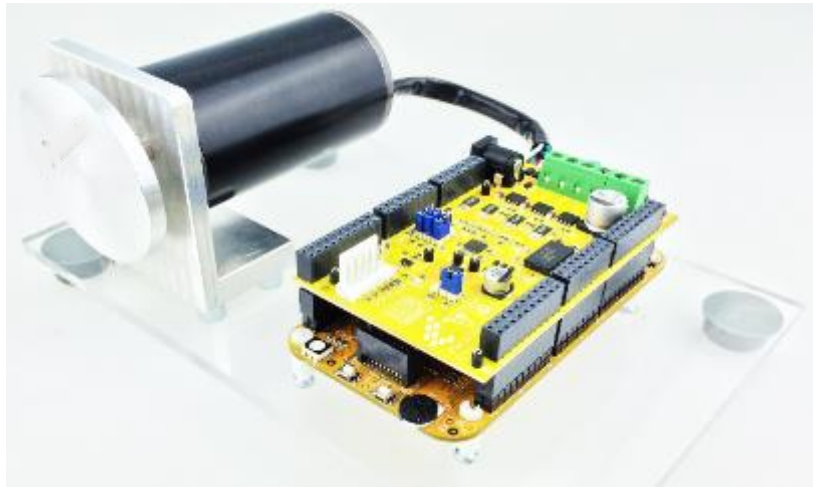
- www.arccore.com
- ARCCORE: Independent 3rd party with strong OEM engagement, ISO26262 safety support and mass mkt customer support model
- Bundle solution: AUTOSAR OS (ARCCORE) + MCAL (NXP)
  - *Arctic Studio* IDE: Eclipse based
  - *Arctic Core* OS: comm. stacks, network mgmt., diagnostics etc.
- 90-day free eval. license – ARCCORE only A'SAR supplier to offer this. With competition ctm needs to buy concept license, typically $10K+
- Launching Q3
  - Sample application for sending and receiving ISO CAN-FD frames every 100ms with processing of incoming data and changing values of outgoing data
  - Delivery format: zip file which can be extracted into S32_DS or Arctic Studio IDEs

- **Lowering the entry barrier for AUTOSAR adoption**

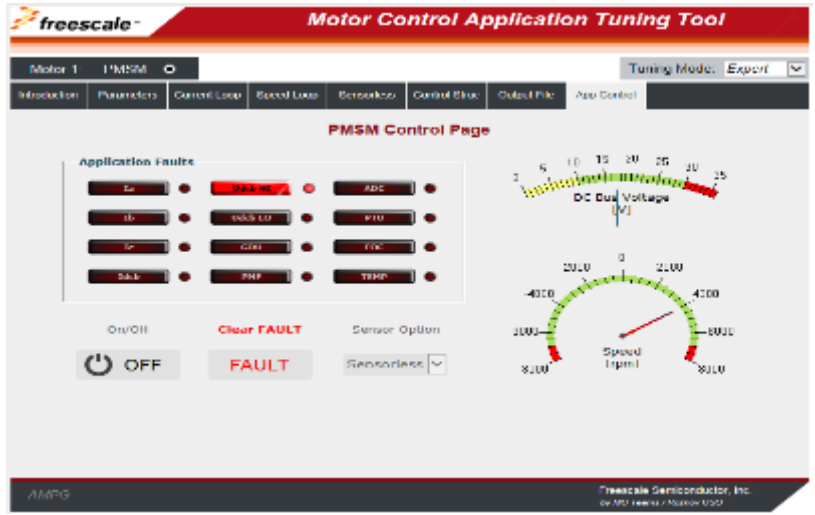




S32K – Motor Control Development Tools



DEVKIT-MOTORGD
Sensorless BLDC/PMSM, due Q217



AMMCLIB

- Automotive Math and Motor Control Library

Motor Control Toolbox

- MATLAB based Toolbox

MCDEVKITs

- HW development Kits

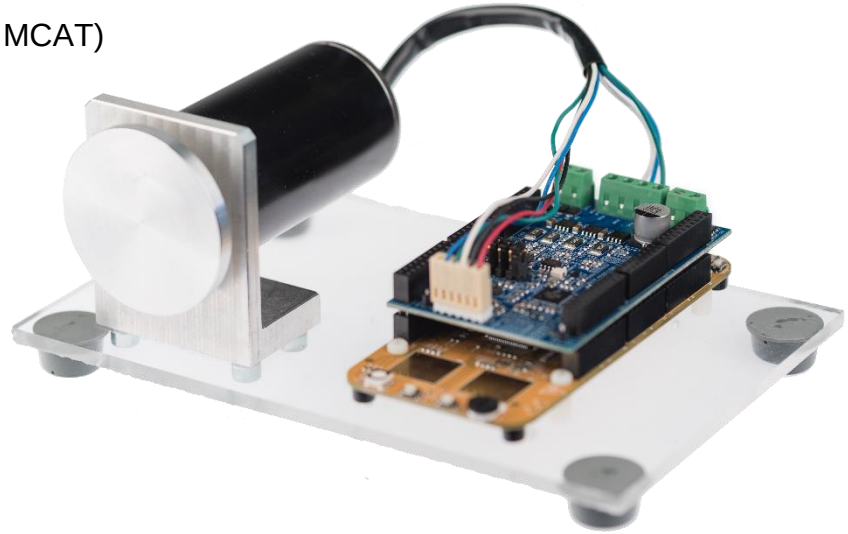
MCAT

- Motor Control Application Tuning Tool

S32K – BLDC/PMSM Motor Control

- **Products:** (based on S32K144EVB and DEVKIT-MOTORGD)
  - **MTRDEVKSBNK144 (BLDC based)**
  - **MTRDEVKSPNK144 (PMSM based)**

- **Key Features:**
  - S32K s/w (S32_Design Studio IDE, S32K SDK), and Motor Control s/w (AMMCLIB, FreeMASTER, MCAT)
  - Sensor and sensorless operation of both BLDC/PMSM motors
  - S32K key features: CAN-FD/PN, Security



- **Technical challenges addressed:**
  - Low cost reference implementation of the sensorless PMSM & BLDC control algorithms
  - Modular approach enabling additional functionality on top of Motor Control
  - Supporting up to 24V MC applications

- **Benefits:**
  - Complete out-of-the box experience, spinning motor in 10 minutes
  - Conveniently and quickly evaluate features & performance
  - Reduce development & prototyping time
  - Faster Time to Market

- **Availability:** Q3 2017

www.nxp.com/AutoMCDevKits

Automotive Math and Motor Control Library Set

Features:

- Precompiled off-the-shelf software library containing the building blocks for a wide range of motor control applications, **significantly decreasing the development time**
- Enable **easy migration between platforms** with minimized effort
- **Production ready SW** (**SPICE Level 3**, **CMMI** and **ISO9001/TS16949**)
- Supporting **fixed-point 32-bit**, **fixed-point 16-bit** and single precision **floating point** arithmetic
- Provided with **Matlab/Simulink® models** for the control loop modeling
- Evaluation version available on **www.nxp.com/AutoMCLib**

Supported Compilers:

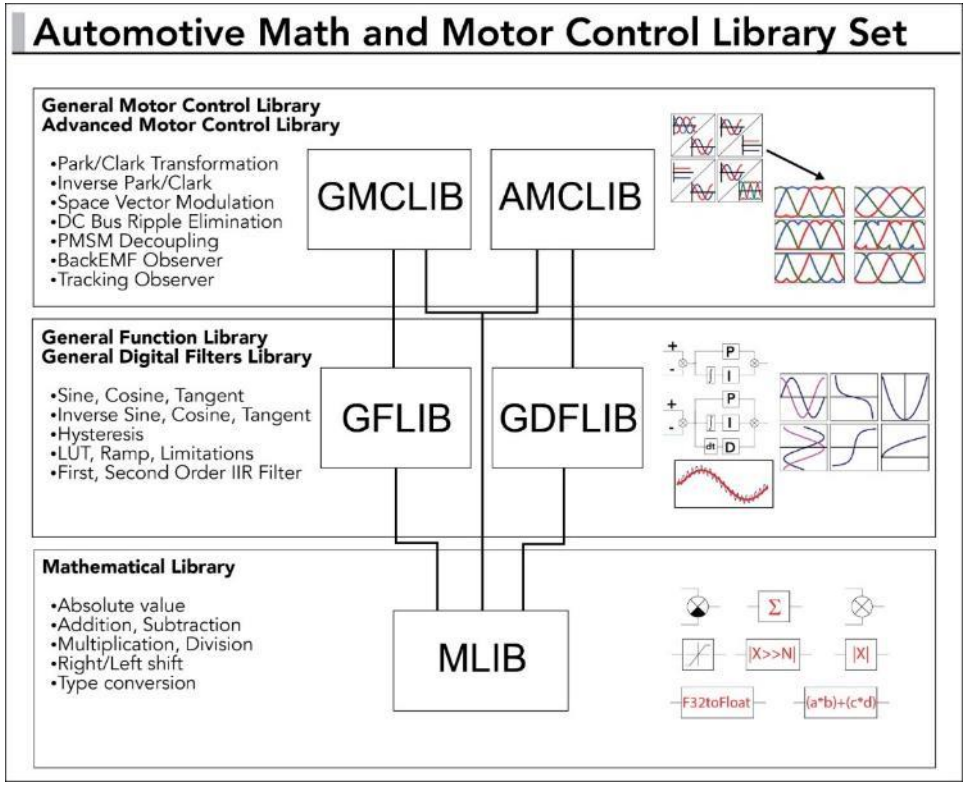
- NXP **CodeWarrior** Eclipse, NXP **S32DS**, **GreenHills** Multi, WindRiver **Diab**, **Cosmic**, **IAR**, **GCC**

Suported Devices:

- KEAx, S12ZVM, S32K14x
- MPC560xP, MPC560xB, MPC564xL, MPC567xF, MPC567xK
- MPC574xC, MPC574xG, MPC574xP, MPC574xR, MPC577xC, MPC577xK, MPC577xM

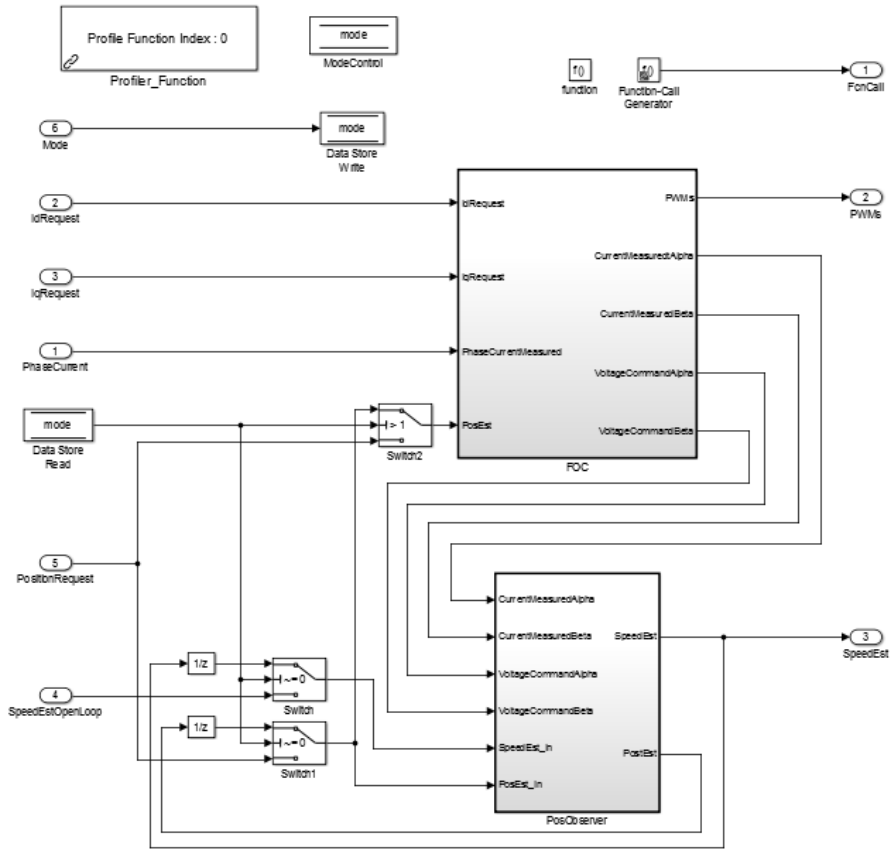
Target Applications:

- Standard & advanced motor/actuator control
- Sensor based and sensor-less applications
- General mathematical applications



Model Based Design Tools (previously Motor Control Toolbox)

- Model-based design environment in MATLAB™/Simulink™ for generating motor control SW on NXP MCUs
- Auto code generation based on the Matlab Embedded Coder
- Includes Automotive Math and Motor Control Library set
- Integration with FreeMASTER for fine tuning

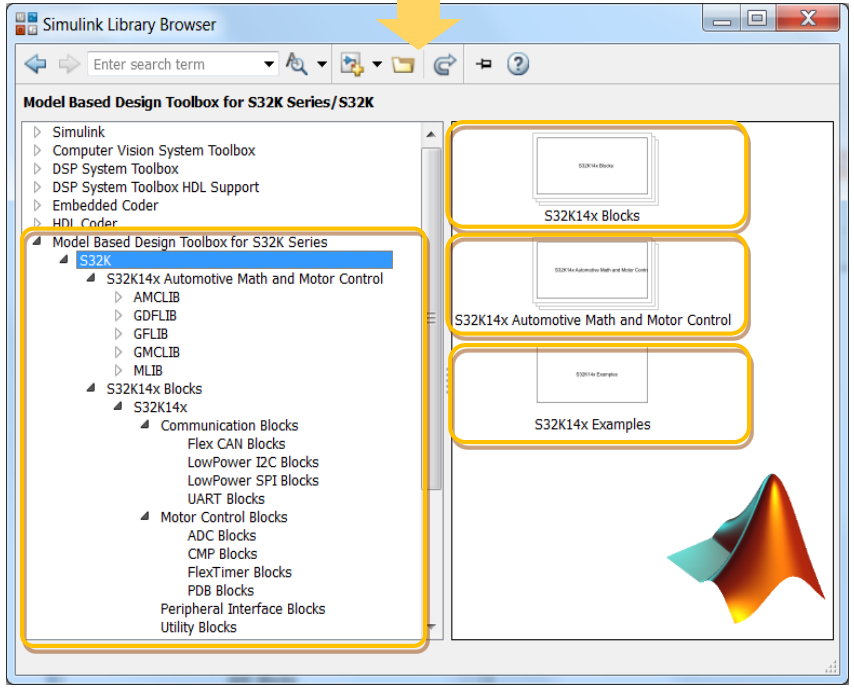


→ **Model-based design & auto code generation**
→ **Reduce development & prototyping time**
→ **Faster Time to Market**
→ **Support migration between platforms**

www.nxp.com/MCToolbox

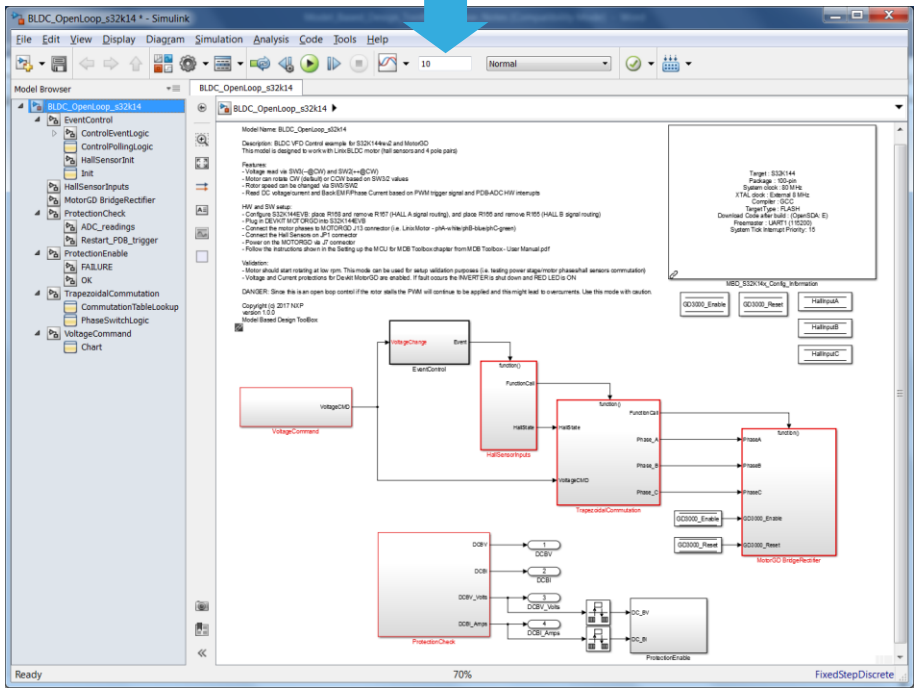
Model Based Design Tools

1 Basic/Advanced Building Blocks



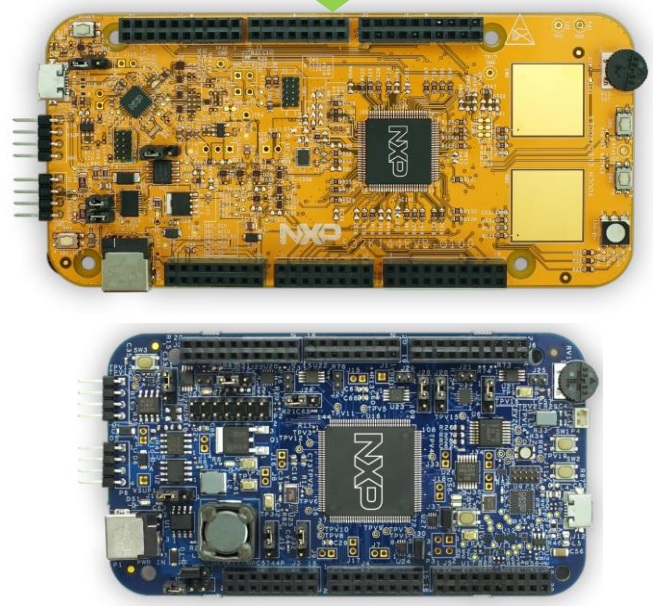
- ✓ **M**ATLAB integration
- ✓ **A**utomatic ANSI C Code Generation
- ✓ **A**bstractization of peripherals
- ✓ **S**upport for: S32K,MPC5744, etc.

2 Ideas & Designs



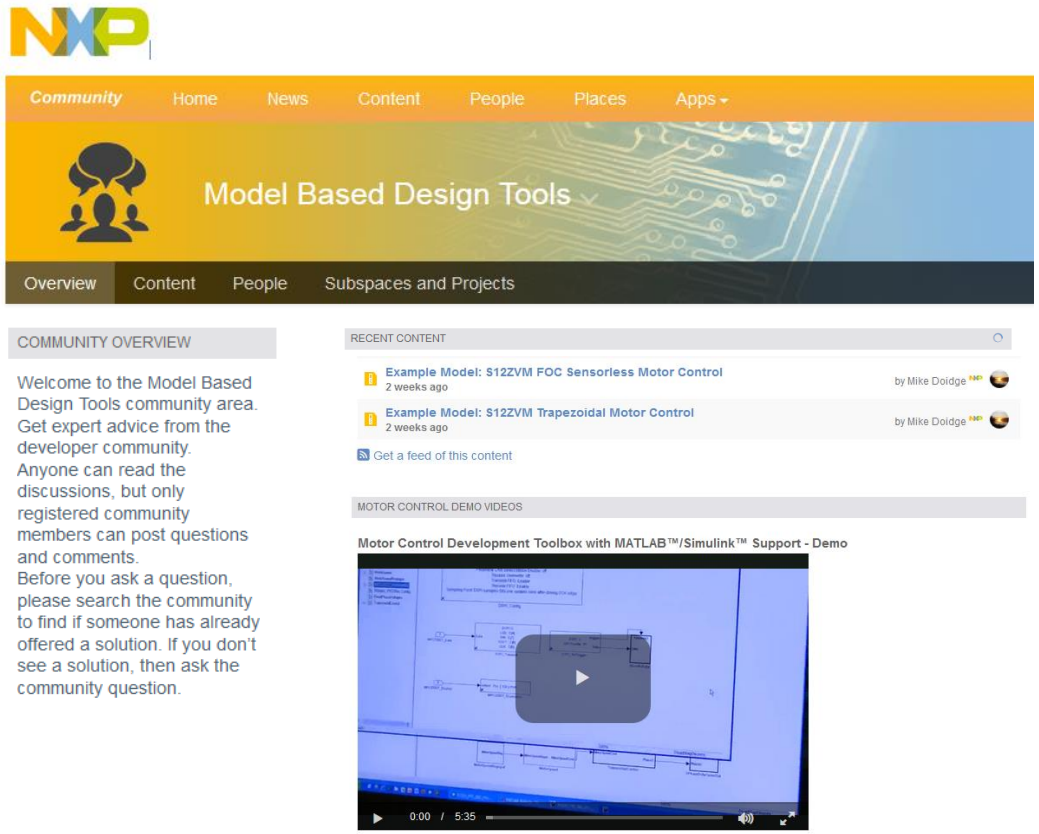
- ✓ **M**inimal knowledge about hardware
- ✓ **U**ltra fast development – drag&drop approach
- ✓ **S**imulation environment for validation
- ✓ **T**echnical expertise

3 Easy Prototyping



- ✓ **Easy migration between NXP solution**
- ✓ **Matching solution for TI, Arduino, Raspberry**

Model Based Design Community
https://community.nxp.com/community/mbdt



- Peripheral **initialization** through UI configuration from a Model Based Design environment like Simulink™
- Supported **platforms**: MagniV S12ZVMx and S12ZVC, MPC56xx, MPC57xx, S32K, DSC and Kinetis families
- S32K community https://community.nxp.com/docs/DOC-334170
- Customer **support** and **training**:

MBDT: Library Contents

On-Chip Peripherals

- General
  - ADC conversion
  - Digital I/O
  - PIT timer
  - ISR
- Communication Interface
  - CAN driver
  - SPI driver
  - I2C
  - UART
- Motor Control Interface
  - Cross triggering unit
  - PWM
  - eTimer block(s)
  - Sine wave generation
  - ADC Command List
  - GDU (Gate Drive Unit)
  - PTU (Programable Trigger Unit)
  - TIM Hall Sensor Port
  - FTM (Flex Timer Module)
  - PDB (Programmable Delay Block)

Configuration/Modes

- Compilers Supported
  - CodeWarrior
  - Wind River DIAB
  - Green Hills
  - Cosmic
  - IAR
  - GCC
  - RAM/FLASH targets
- Simulation Modes
  - Normal
  - Accelerator
  - Software in the Loop (SIL)
  - Processor in the Loop (PIL)
- MCU Option
  - Multiple packages
  - Multiple Crystal frequencies

Utilities

- FreeMASTER Interface
  - Data acquisition
  - Calibration
  - Customize GUI
- Profiler Function
  - Exec. time measurement
  - Available in PIL
  - Available in standalone

Embedded MCU Support

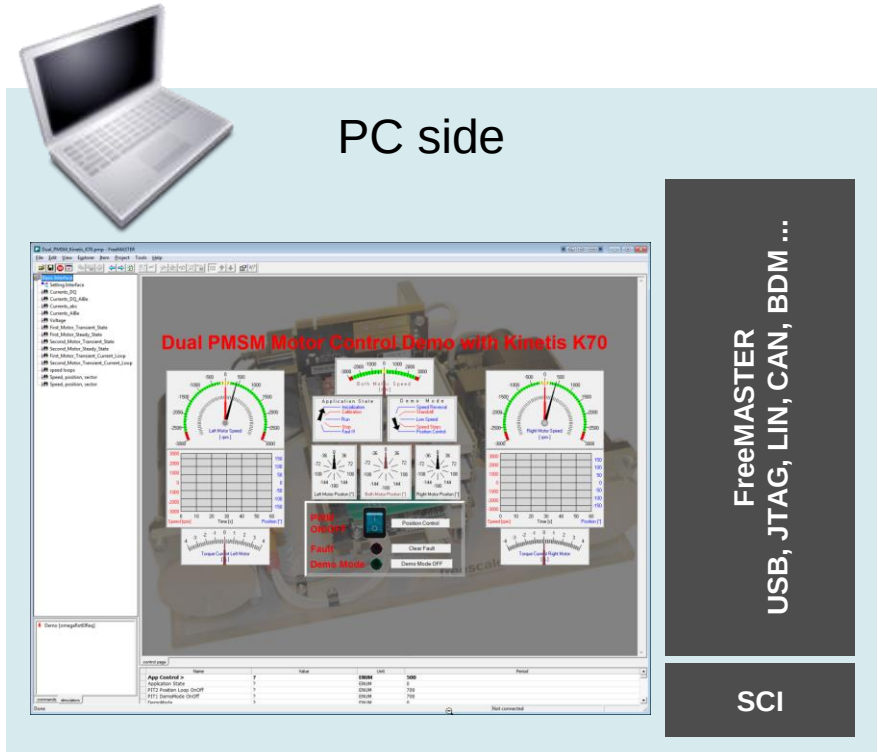
- MPC5643L
- MPC567xK
- MPC574xP
- S12ZVM
- KV10Z
- 56F82xx
- KV31/30/40/50
- S32K

NOTE: Peripheral Blocks and compiler support is dependent on MCU use.

AMP Motor Control Software Pricing

License	Pricing as of Oct 2016
Motor Control Toolbox (evaluation)	Obsolete (no timelimit)
Motor Control Toolbox (license)	Free
AMMCLIB Production (Object Code License)	$5k per device family
AMMCLIB Production (Source Code License)	$30k per device family

FreeMASTER – Real-Time Debugger



Runtime configuration & tuning tool :

- **Connect**: to target MCU
- **Monitor**: variables in run-time
- **Control**: set variables, send commands
- **Share**: Excel, Matlab, etc.

→ **Comprehensive debugging at application level**

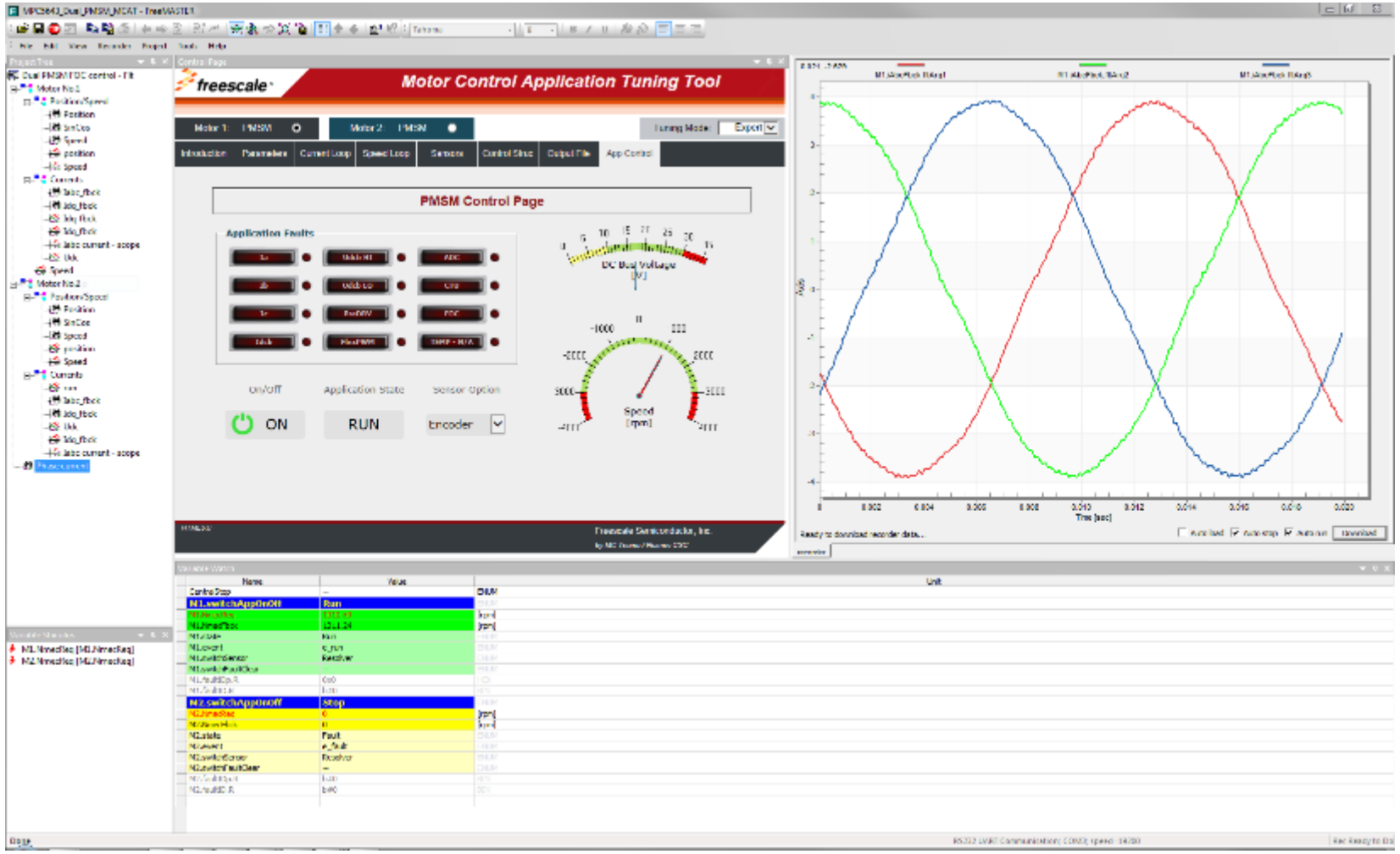
→ **Reduce development & prototyping time**

→ **Faster Time to Market**

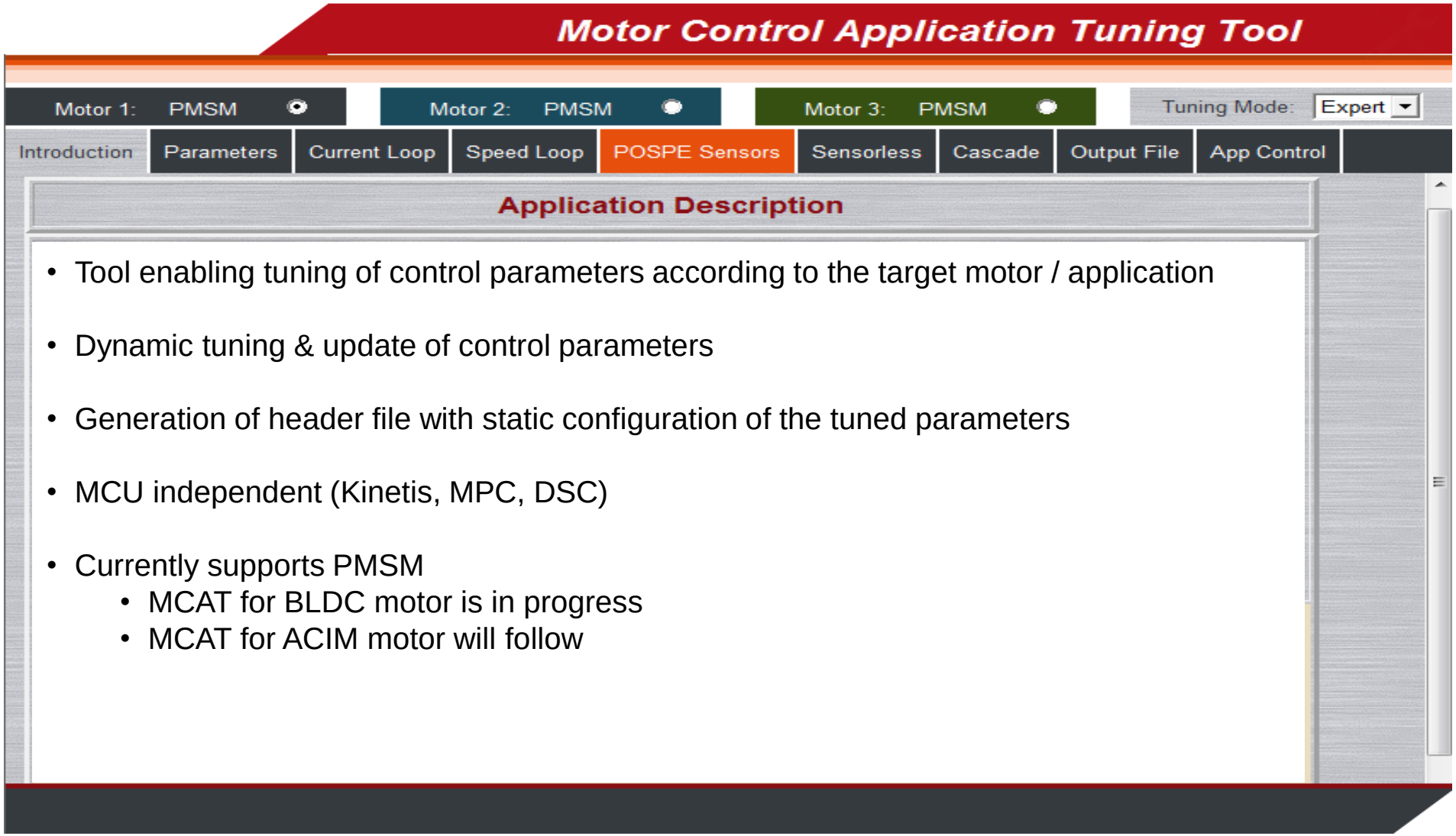
www.nxp.com/FreeMASTER

FreeMASTER Tool

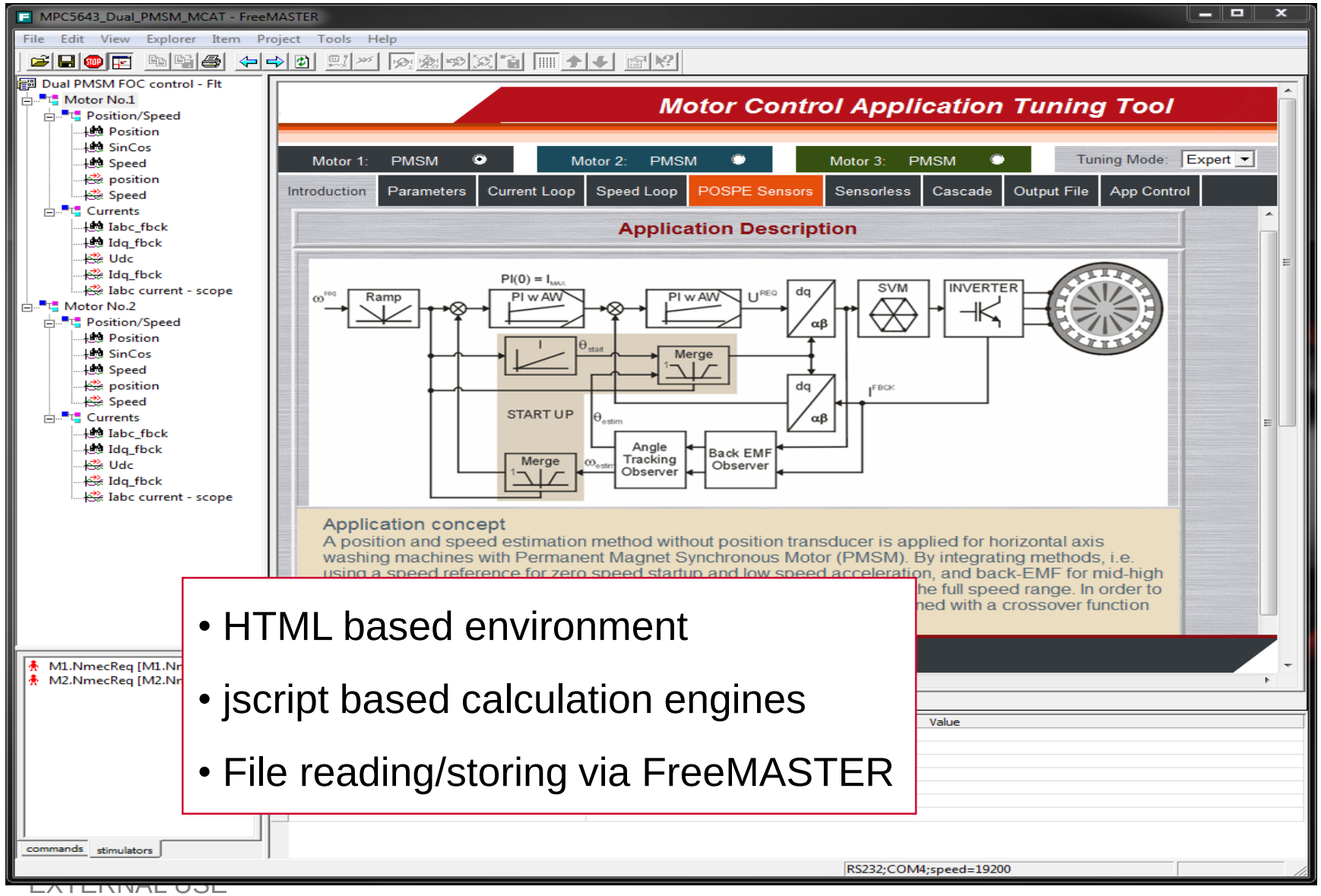
- Real Time Data Visualization, data logging and parameter tuning tool
- Connects to target through Serial or CAN interfaces
- Samples data on target, limited by communication channel data rate only
- Provides Data Visualization of variables on target
- Allows for data logging of the variables on the target
- Allows for tuning variables in the target
- S32DS IDE comes with embedded library



Motor Control Application Tuner (MCAT) Introduction & Features



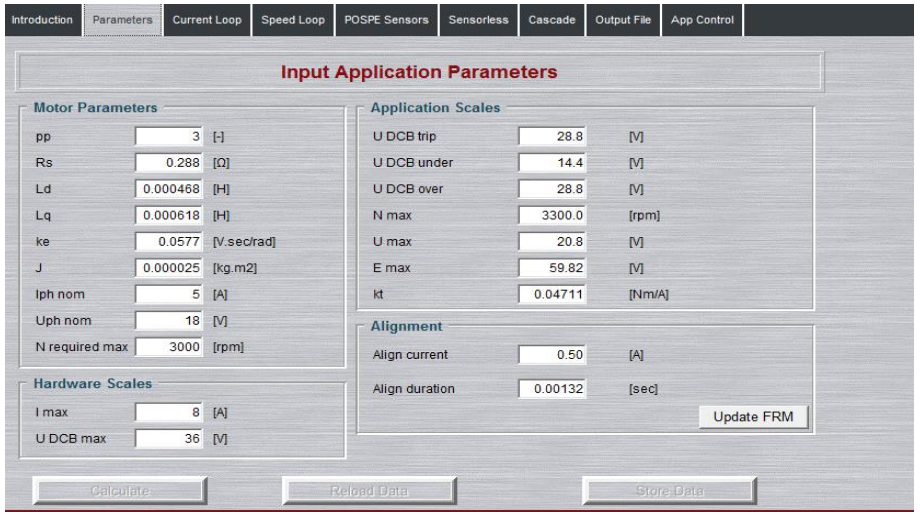
FreeMASTER with MCAT



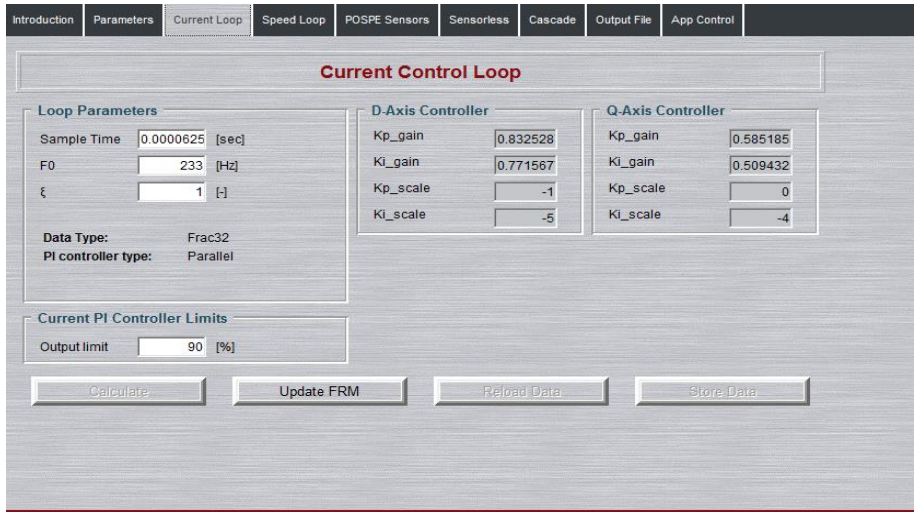
- HTML based environment
- jscript based calculation engines
- File reading/storing via FreeMASTER

Steps to Tune the Current Loop

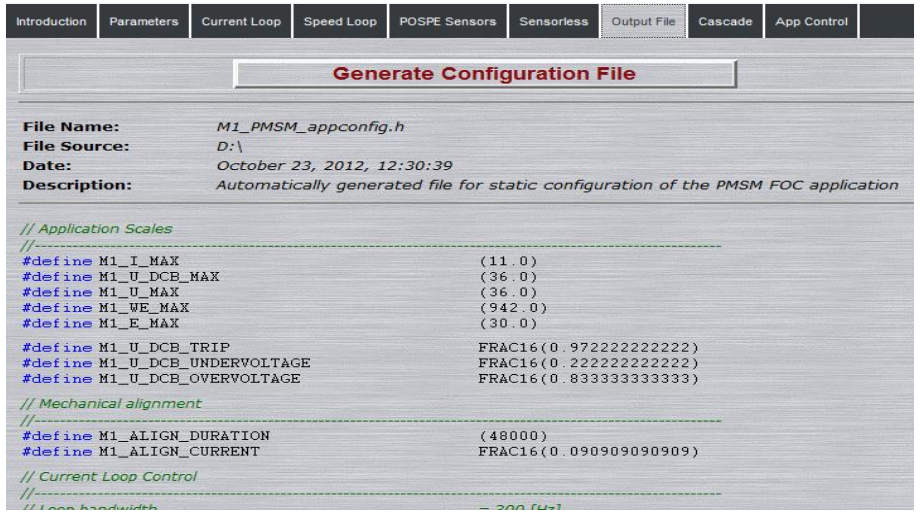
1. Parameter Setting-Up



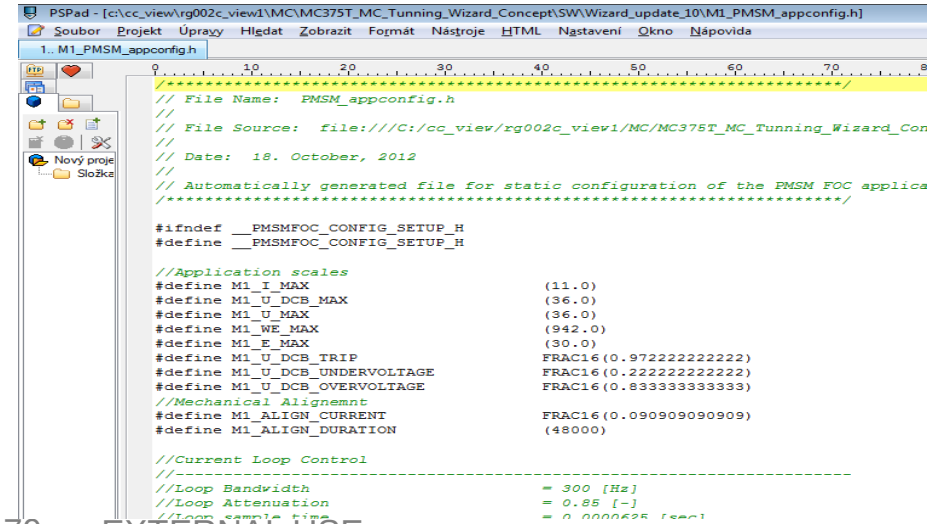
2. Control Loop Tuning



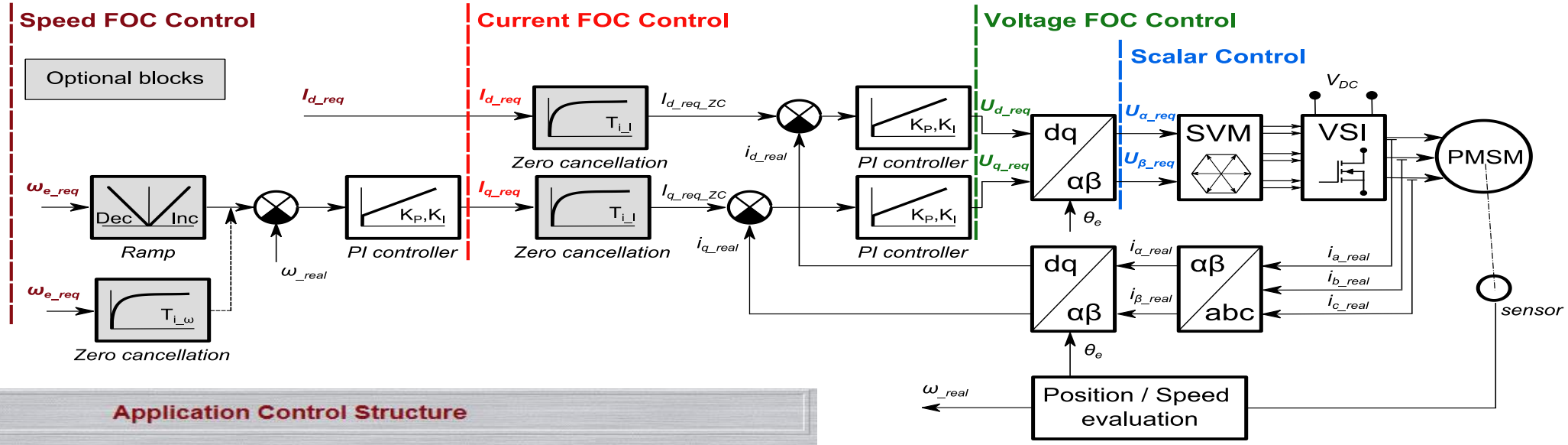
3. Output Control Constant Preview

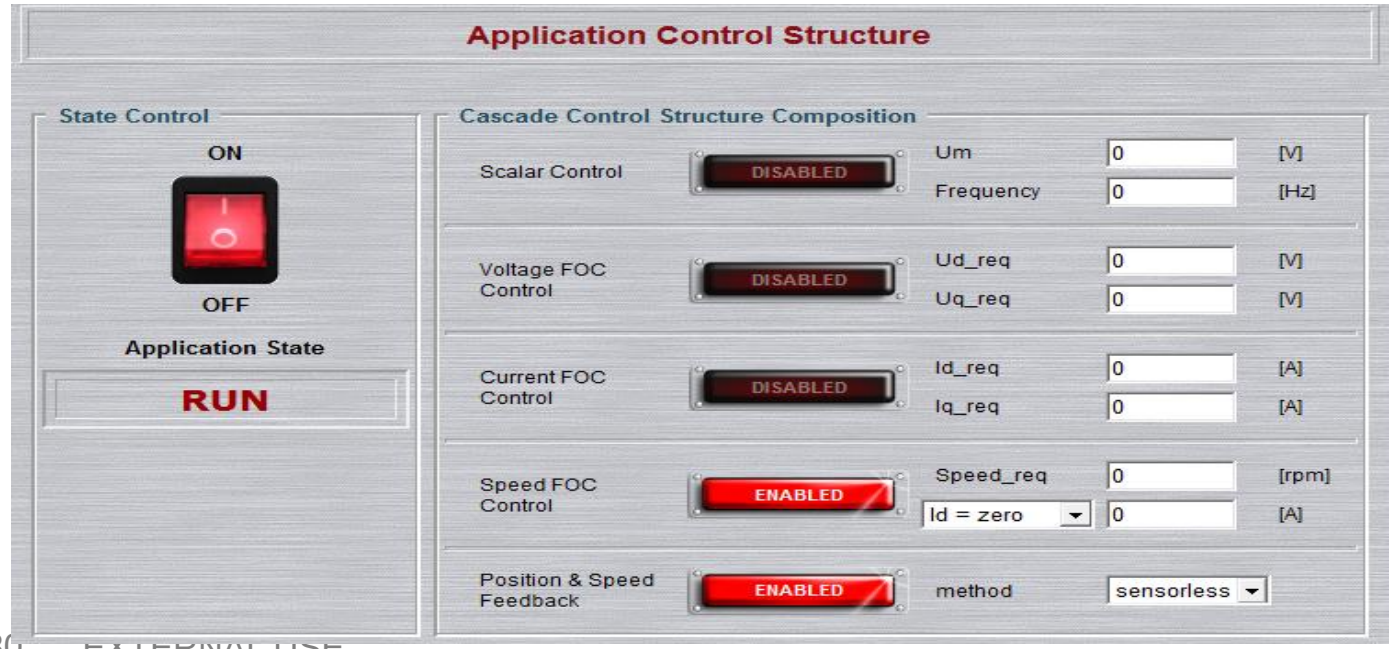


4. Generated .h file



MCAT Control Structure Selector





Open loop control
no need any current, position or speed feedback

Voltage control – position required
no need any current and speed feedback

Current control – current, position required
no need any speed feedback

Speed control - current, position and speed required

S32K144 MCU – IDEs & App Notes

Software	Features
S32 Design Studio IDE v1.3 www.nxp.com/S32DS	• Free of charge, zero code limit, Supports GCC and 3rd party compilers, • Processor Expert framework • Compatible with AMMCLIB (Advanced Math & Motor Control Library), FreeMASTER, Motor Control Toolbox (model based design tools)
S32K SDK www.nxp.com/S32DS (SDK is embedded within S32_DS download)	• Automotive grade, MISRA 2012 & SPICE Level 3 compliant • Free low-level drivers & freeRTOS • Middleware licensable • Driver examples support 3rd party IDEs, Uses Processor Expert as a config. tool

Category	Description	Availability
App Note	S32K144 Hardware Design Guidelines	March / Q2
App Note	S32K144 Cookbook (multiple small App Notes combined)	March / Q2
App Note	S32K CSEc Security Module	March / Q2
App Note	FlexIO module use cases	March / Q2
App Note	AN5258: Using DMA for pulse counting on S32K	March / Q2
App Note	AN5303: Features and Operation Modes of the FlexTimer	March / Q2
App Note	S32K Family vs. KEA MCU Family Comparison	March / Q2
App Note	S12X to ARM Migration	March / Q2
App Note	S32K144 vs. Kinetis MCU Family comparison	March / Q2
Tool	S32K144 FTM PWM Calculator	March / Q2
Tool	S32K144 FlexMemory (EEPROM) Calculator	March / Q2

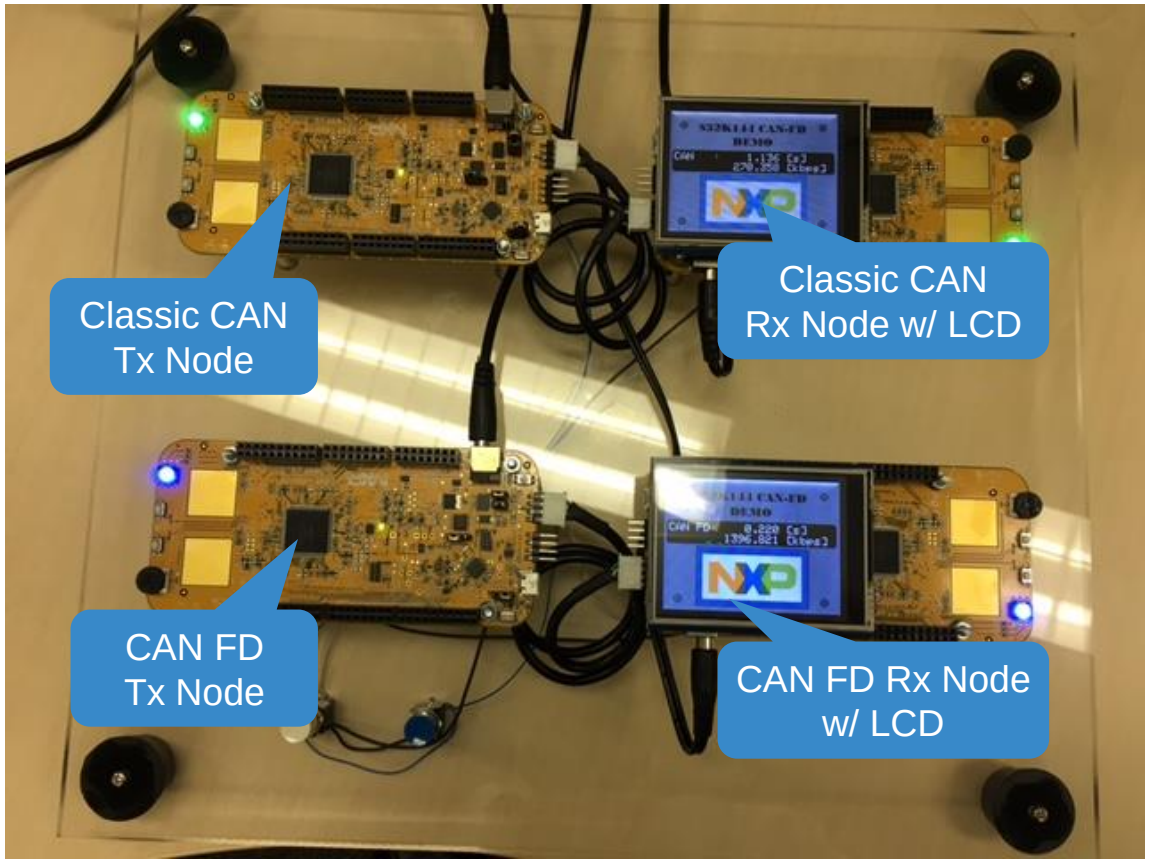
S32K Technical Collateral

Product	Collateral/Deliverable	Type	On web @ launch?	Comment	Status
	AppNotes/Trainings				
S32K144	S32K144 HW Design Guidelines AppNote	AN-ext	Yes	Draf in Autopad S32K144ANs, working on final version.	Done
S32K144	S32K144 HW Design Guidelines Training	PPT	no	Autopad S32K144 Technical sessions	Done
S32K144	Bolero to S32K144 Migration Training	PPT	no	Autopad S32K144 Technical sessions	Done
S32K144	S32K144 vs. Bolero Technical Value Proposition	PPT	no	Autopad S32K144 Technical sessions	Done
S32K144	S32K144 Cookbook	AN-ext	Yes	Autopad S32K144ANs	Ongoing
S32K144	S32K144 Safety Training	PPT	no	Autopad S32K144 Technical sessions	Done
S32K14x	S32K144 vs. MA512 Differences	AN-Int	no	Autopad S32K144 V2.0 documentation	Done
S32K14x	S32K144 and Kinetis family comparison	AN-ext	Yes	Draft available	
S32K14x	S32K144 FTM PWM Calculator	Tool	tbd	https://community.nxp.com/docs/DOC-104756	Done
S32K14x	S32K144 Getting started with FreeRTOS	?	?	As part of SDK. Need clarification of AN existance	Done
S32K14x	S32K144 vs. Kinetis Peripheral Re-use	PPT	no	Senna, L5-K. Presentation for internal use only	Done
S32K144	S32K144 Video Tutorials		Yes	For disti launch, partially with SDK and EVB out of the box video	Open
S32K144	S12X to ARM AppNote migration note	AN-ext	no	won't be ready for the launch	Open
S32K144	S32K144 FlexMem (EEPROM) Calculator	Tool		S32K1xx E2 calculator site	Done
S32K144	S32K vs KEA AppNote	AN-ext	Yes	Autopad S32K144ANs	Done
S32K14x	S32K PDB, FTM, ADC Interaction	PPT		Autopad S32K144 Technical sessions	Done
S32K14x	FlexIO use cases	AN-ext	Yes	Autopad S32K144ANs	Done
S32K14x	S32K E² Emulation	AN-ext	TBD	Autopad S32K144ANs	Done
S32K14x	S32K Security	AN-ext	Yes	Draft available and in review	Ongoing
S32K14x	AN5258: Using DMA for pulse counting on S32K - Application Note	AN-ext	Yes	Autopad S32K144ANs	Done
S32K14x	AN5303: Features and Operation Modes of the FlexTimer Module - Application Note	AN-ext	Yes	Autopad S32K144ANs	Done
S32K & KEA	S32K and KEA for motor control	PPT-int	Yes	Training presentation is available (Motor Control Summit 2016)	Done
S32K & MagniV	FTF - Faster Motor Control Development	PPT-ext	Yes	Training presentation is available (NXP FTF Connects 2016)	Done
S32K14x	TriggerMUX	AN	TBD	Still to be defined whether we need an app note	
S32K14x	S32K power management	AN-ext	TBD	Still to be defined whether we need an app note	
S32K14x	Optimizing performance on S32K series MCUs	AN-ext	TBD		
All	Using NXP's LIN stack - including an S32K example as well	AN	TBD	Still to be defined whether we need an app note	
S32K14x	Using Rappid bootloader for S32K	AN	TBD	Still to be defined whether we need an app note	

S32K Demo – CAN-FD vs. CAN

Overview

- Shows the performance difference between CAN message transmission using classic CAN at 500kbps with 8-byte payload vs. CAN-FD at 2Mbps with 64-byte payload
- Transmission nodes (EVBs on left-hand-side) transmit the data of NXP-logo bitmap image via a CAN interface
- Receiver nodes (EVBs on right-hand-side) receive the bitmap data and transfer to LCDs message-by-message
- **Bitmap data transmission using CAN-FD (0.2s) is much faster than classic CAN (1.1s)**



Key Messages

- All S32K1x MCUs from 128KB to 2MB include the same ISO-certified CAN FD controller so software migration is easy throughout the family
- The CAN driver software is included free of charge in the SDK (Software Development Kit)
- S32K144EVB includes a CAN-FD compliant SBC that supports data rates up to 2Mbps

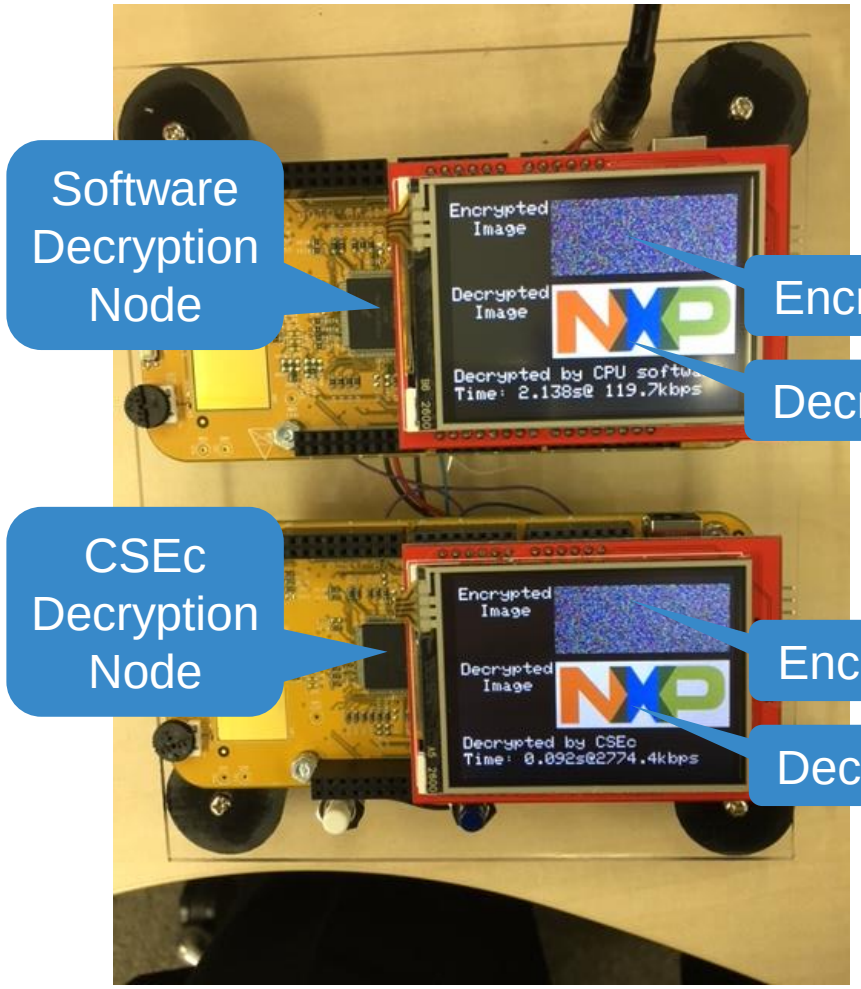
S32K Demo – CSEc Image Decryption

Overview

- Shows the performance (decryption speed) difference between decryption using software (core) vs. using the CSEc module with AES-128
- The NXP-logo bitmap image is pre-encrypted (CBC algorithm) and stored in the MCU memory.
- After the MCU reset, the pre-encrypted image is displayed on the upper side of LCD. The picture is not recognizable since it's encrypted.
- After pushing START button, both nodes start to decrypt the image data by 1) CPU software, and by 2) CSEc AES-128 hardware, and gradually draw the picture from top to bottom
- After the completion of entire bitmap image data decryption, elapsed time with processing data rate is displayed
- **Image decryption using CSEc (0.09s) is much faster than using software/core (2.14s)**

Key Messages

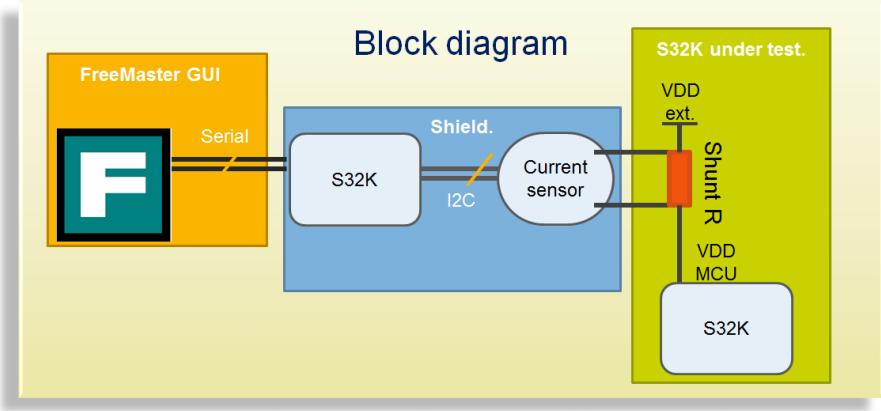
- All of S32K1xx family products from 128KB to 2MB flash products include the CSEc module so migration is easy throughout the family
- The CSEc driver software is included free of charge in the SDK (Software Development Kit)



S32K Demo – Low Power

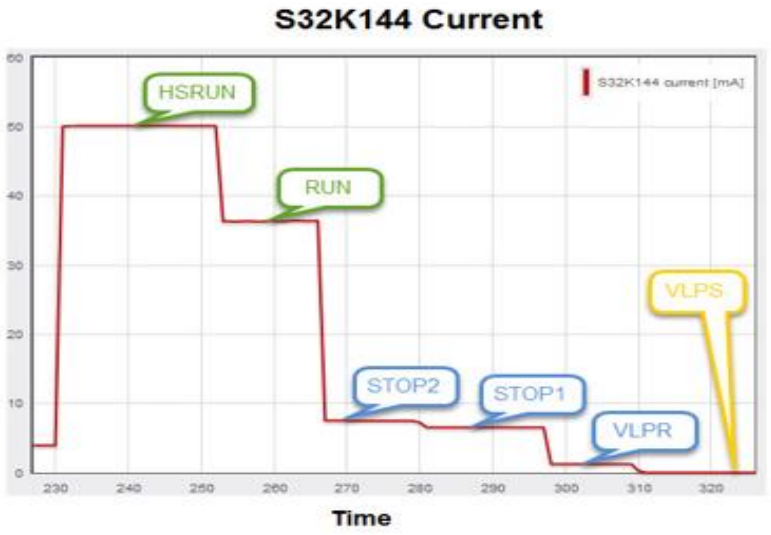
Overview

- S32K144EVB + low power plug-in ‘shield’ board
- Demonstrates all MCU power modes:
  - HSRUN, RUN, STOP2, STOP1,VLPR,VLPS
  - Showcases application use cases
- Uses FreeMASTER interface (real-time, non-intrusive debug monitor)



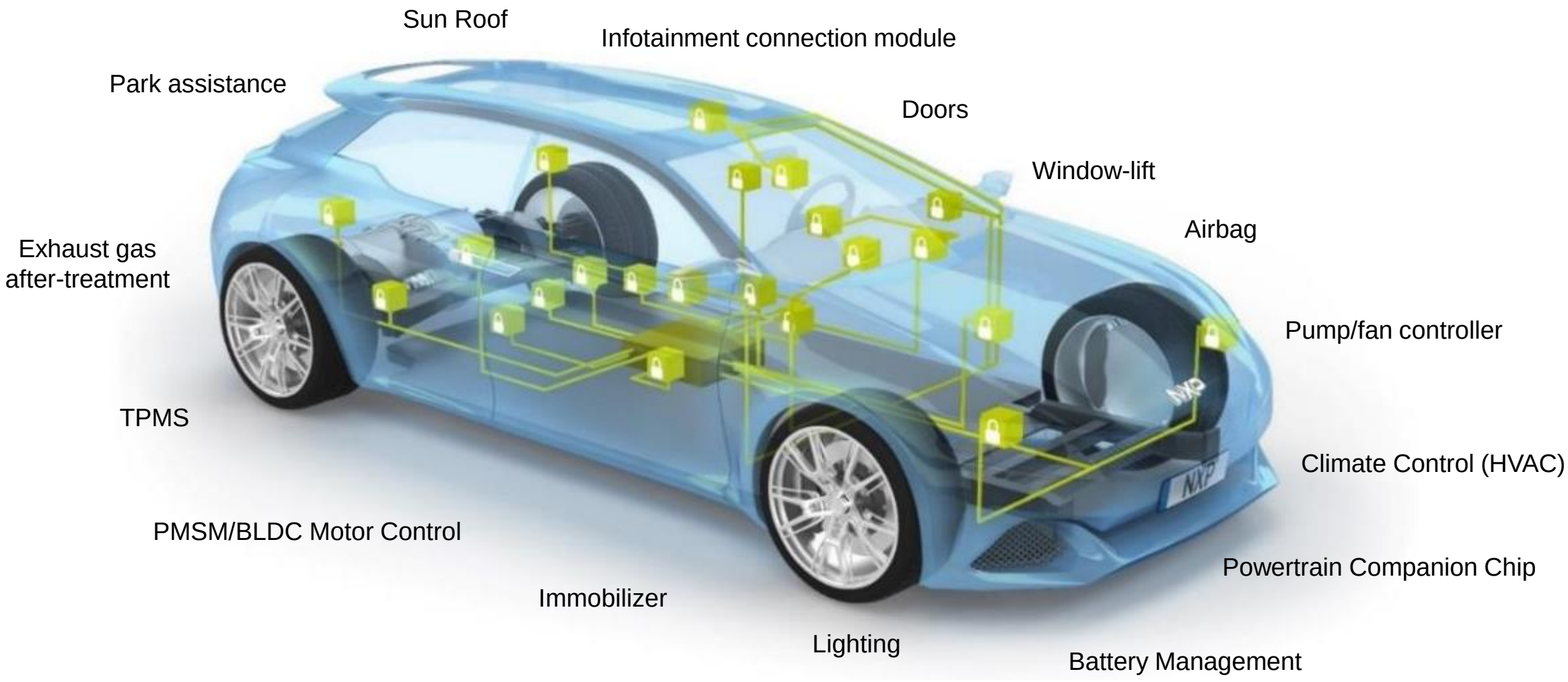
Key Messages

- <100uA average power consumption with periodic wake-ups and ADC measurements/SCI communication
- <100mA in HSRUN mode @ 112MHz with all peripherals active



APPLICATIONS

S32K MCUs in Automotive Applications

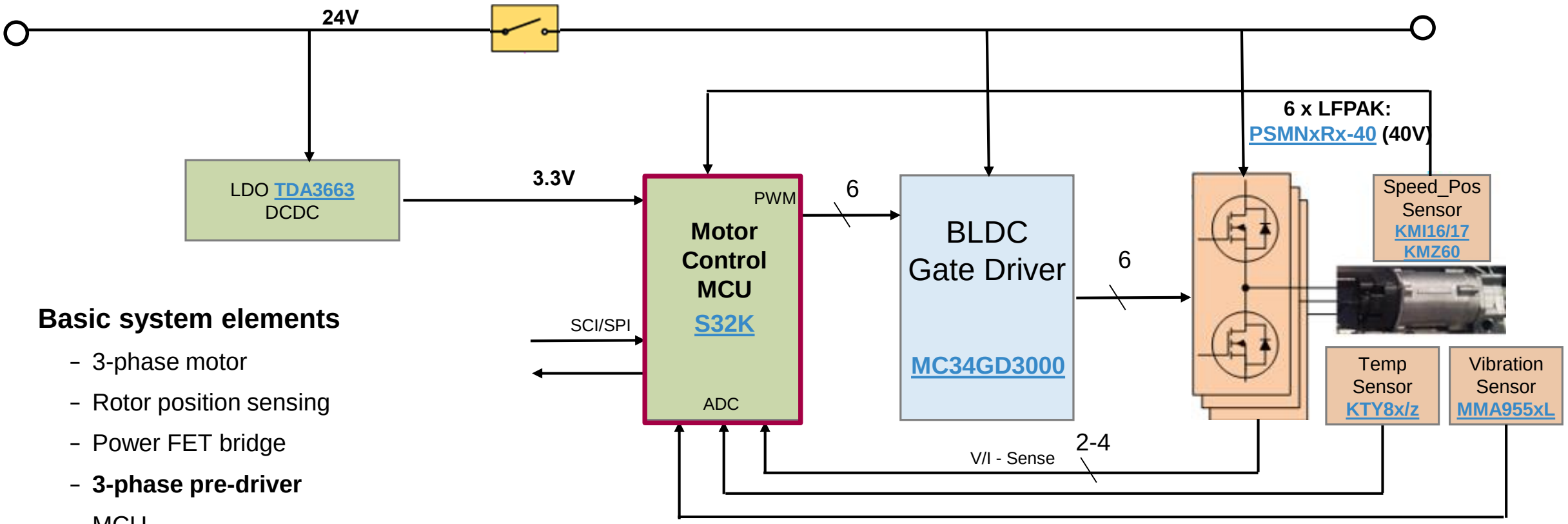


S32K MCUs in Industrial Applications

S32K MCUs deliver quality, reliability and safety for challenging environments found in industrial, automation, communications, transportation, medical and A&D applications.

- **Mimumum** 15 year product longevity
- Extended temperature -40 – 125°C
- CAN-FD & Hardware Security – fast, secure industrial networking
- Safety baseline is ISO26262, IEC61508 additions included for certain products
  - Developed according to ASIL B, similar to SIL2
  - Structural Core Self Test Library available
- Below 1ppm defect quality
- Vreg 2.7 to 5.5V, not limited to 3.3V
- Production grade software tools
  - S32DesignStudio IDE + SDK
  - Freemaster, MCAT, Motor Control Toolbox...

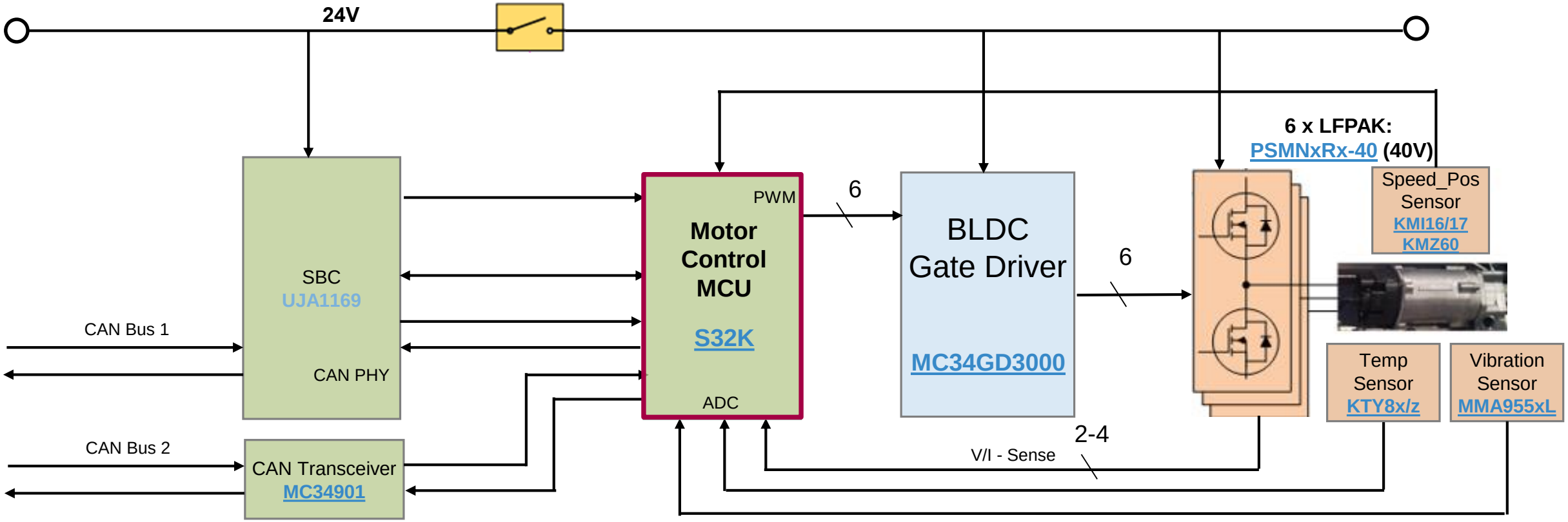
3-Phase Motor Control System – 24V



Basic system elements

- 3-phase motor
- Rotor position sensing
- Power FET bridge
- **3-phase pre-driver**
- MCU
- System power management
- Application system communication interface (optional)

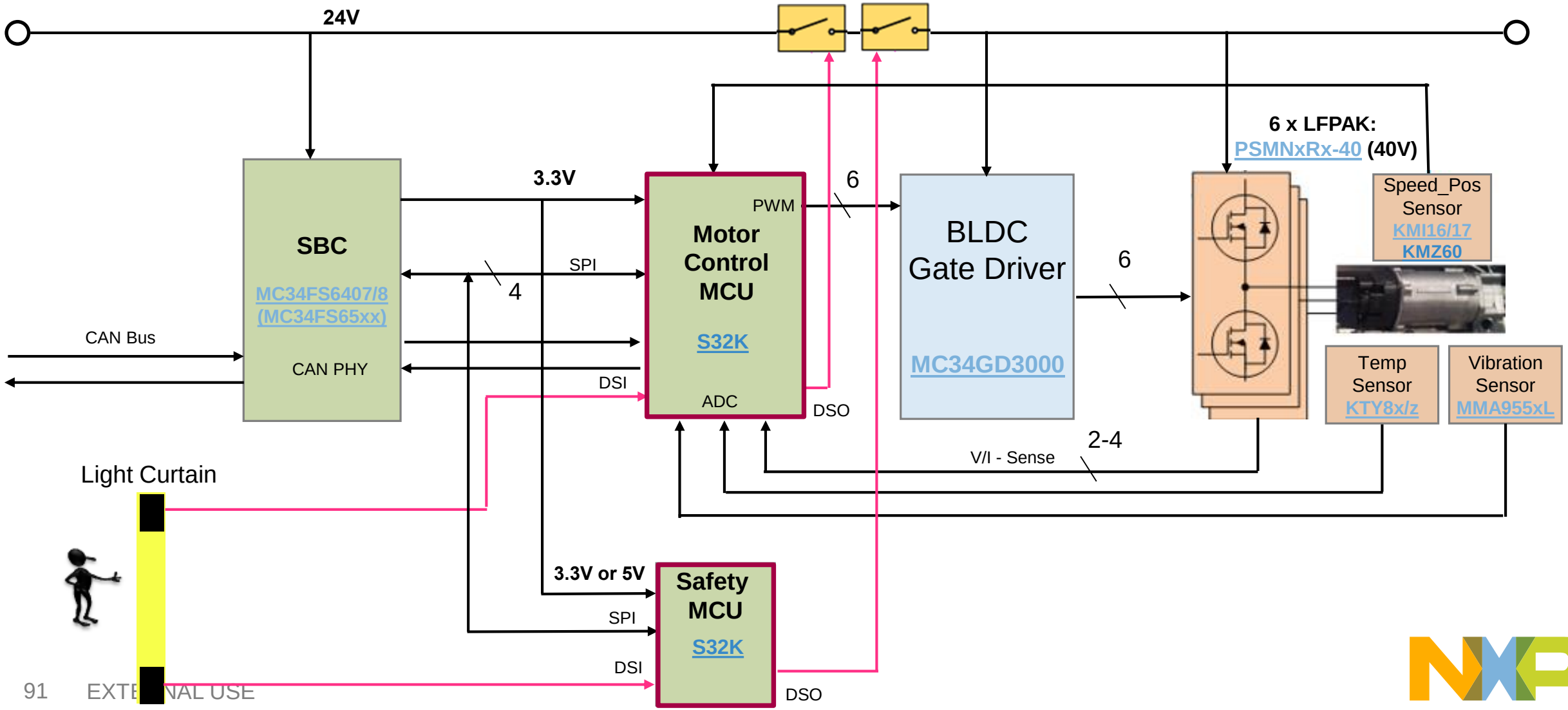
3-Phase Motor Control System – 24V – with CAN



System upgrades

- Add local monitors for fault detection at 3PP, MCU, and PMIC
- Add fault detection message exchange paths to monitor each other
- Add local FailSafe control at 3PP, MCU, and PMIC and the MCU is the system FailSafe control center

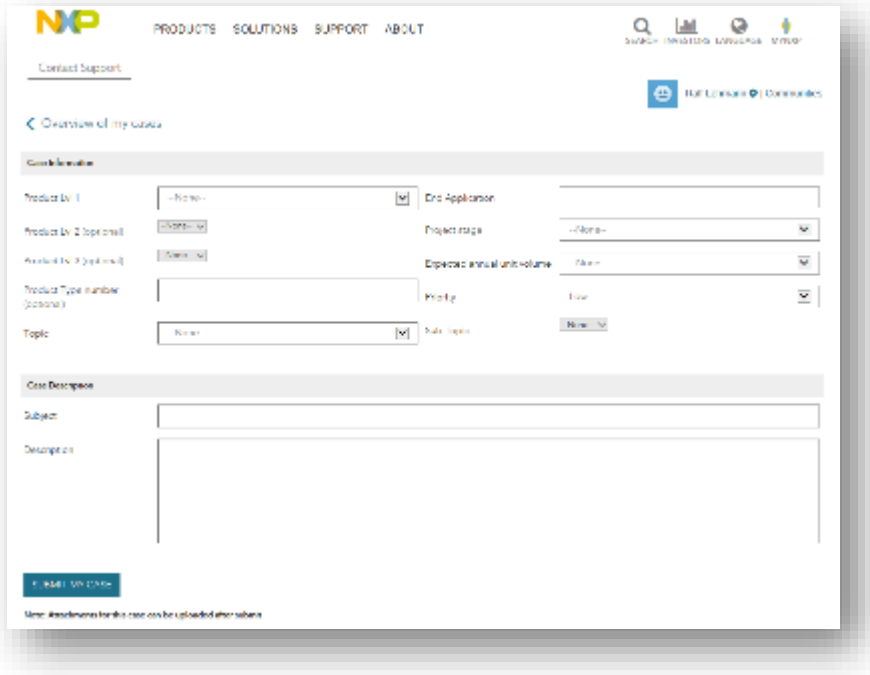
3-Phase Motor Control System (24V) with CAN and 2nd safety MCU



S32K Technical Support http://www.nxp.com/support

- **Communities**
  - **S32K MCU Community**
    - https://community.nxp.com/community/s32/s32k
    - S32K MCU & SDK related topics
  - **S32_Design_Studio IDE Community**
    - https://community.nxp.com/community/s32/s32ds
    - S32DS IDE related topics

- **TIC (Technical Information Center)**
  - Log in with your NXP Communities UN & PWD
  - If new user, please register. If no verification email is received, please check your spam folder. Email is sent from engineers.corner@nxp.com
  - Enter your support CASE - All fields are mandatory



Why S32K?....the answer is SIMPLE

Security & Safety

Integration

M-Cortex

Power Consumption

Longevity

Enablement, Extended Temp., EEPROM

www.nxp.com/S32k

NXP Automotive FAE interface for TW customer

Mike Cao 曹学余
Xueyu.cao@nxp.com
Control&Network, Automotive Customer Application Solution and Support
NXP (China) Management Co.,Ltd
Mobile: +86 186 1655 2690
Tel: +86 21 2205 2745
地址: 中国上海市裕通路100号宝矿洲际商务中心20层　邮编: 200070
Add: 20F, BM InterContinental Business Center, 100 Yu Tong Road, Shanghai, P.R.C 200070

SECURE CONNECTIONS
FOR A SMARTER WORLD