IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
AUSTIN DIVISION

LUCIO DEVELOPMENT LLC,	§	
	§	
Plaintiff,	§	Case No: 1:17-cv-1155
	§	
vs.	§	PATENT CASE
	§	
NXP SEMICONDUCTORS USA, INC.,	§	
	§	
Defendant.	§	
	§	

COMPLAINT

Plaintiff Lucio Development LLC ("Plaintiff" or "Lucio") files this Complaint against NXP Semiconductors USA, Inc. ("Defendant" or "NXP") for infringement of United States Patent No. 7,069,546 (hereinafter "the '546 Patent").

PARTIES AND JURISDICTION

1. This is an action for patent infringement under Title 35 of the United States Code. Plaintiff is seeking injunctive relief as well as damages.

2. Jurisdiction is proper in this Court pursuant to 28 U.S.C. §§ 1331 (Federal Question) and 1338(a) (Patents) because this is a civil action for patent infringement arising under the United States patent statutes.

3. Plaintiff is a Texas limited liability company with its office address at 555 Republic Dr., Suite 200, Plano, Texas 75074.

4. On information and belief, Defendant is a Delaware corporation with its principal place of business located at 1109 McKay Drive, San Jose, CA 95131.

5. This Court has personal jurisdiction over Defendant because Defendant has

committed, and continues to commit, acts of infringement in this District, has conducted business in this District, and/or has engaged in continuous and systematic activities in this District.

6. On information and belief, Defendant's instrumentalities that are alleged herein to infringe were and continue to be used, imported, offered for sale, and/or sold in this District.

VENUE

7. Venue is proper in this District pursuant to 28 U.S.C. §1400(b) because acts of infringement are occurring in this District and Defendant has a regular and established place of business in this District. For instance, on information and belief, Defendant has a regular and established place of business at 6501 W. William Cannon Dr., Austin, TX 78735.

COUNT I
(INFRINGEMENT OF UNITED STATES PATENT NO. 7,069,546)

8. Plaintiff incorporates paragraphs 1 through 7 herein by reference.

9. This cause of action arises under the patent laws of the United States and, in particular, under 35 U.S.C. §§ 271, *et seq*.

10. Plaintiff is the owner by assignment of the '546 Patent with sole rights to enforce the '546 Patent and sue infringers.

11. A copy of the '546 Patent, titled "Generic Framework for Embedded Software Development," is attached hereto as Exhibit A.

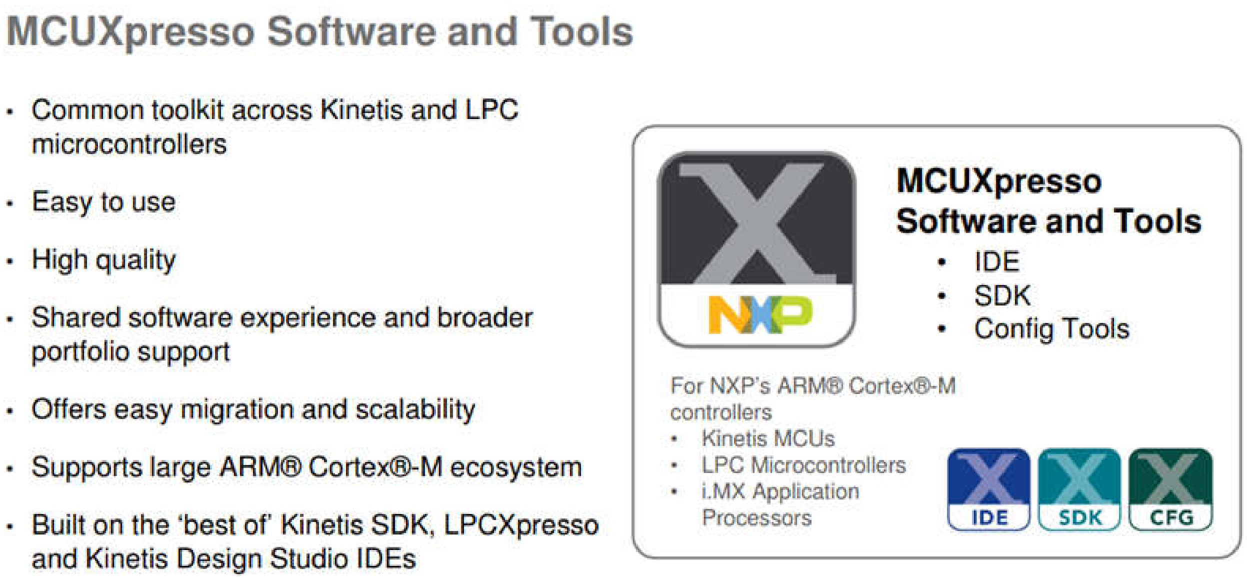
12. The '546 Patent is valid, enforceable, and was duly issued in full compliance with Title 35 of the United States Code.

13. On information and belief, Defendant has infringed and continues to infringe one or more claims, including at least Claim 1, of the '546 Patent by making, using, importing, selling, and/or offering for sale a software platform for embedded software development,

which is covered by at least Claim 1 of the ‘546 Patent. Defendant has infringed and continues to infringe the ‘546 Patent directly in violation of 35 U.S.C. § 271.

14. Defendant, sells, offers to sell, and/or uses embedded software development packages including, without limitation, the MCUXpresso software and tools (including, without limitation, MCUXpresso SDKilinx and MCUXpresso Config Tools), and any similar products (“Product”), which infringe at least Claim 1 of the ‘546 Patent.

15. The Product is a framework that is configured to create embedded software for multiple hardware modules (e.g., versions of a microcontroller, such as Kinetis and/or LPC). Defendant and/or its customers use the Product to produce embedded software. Certain elements of this limitation are illustrated in the screenshots below and in the screenshots referenced in connection with other elements herein.



MCUXpresso Software and Tools

- Common toolkit across Kinetis and LPC microcontrollers
- Easy to use
- High quality
- Shared software experience and broader portfolio support
- Offers easy migration and scalability
- Supports large ARM® Cortex®-M ecosystem
- Built on the ‘best of’ Kinetis SDK, LPCXpresso and Kinetis Design Studio IDEs

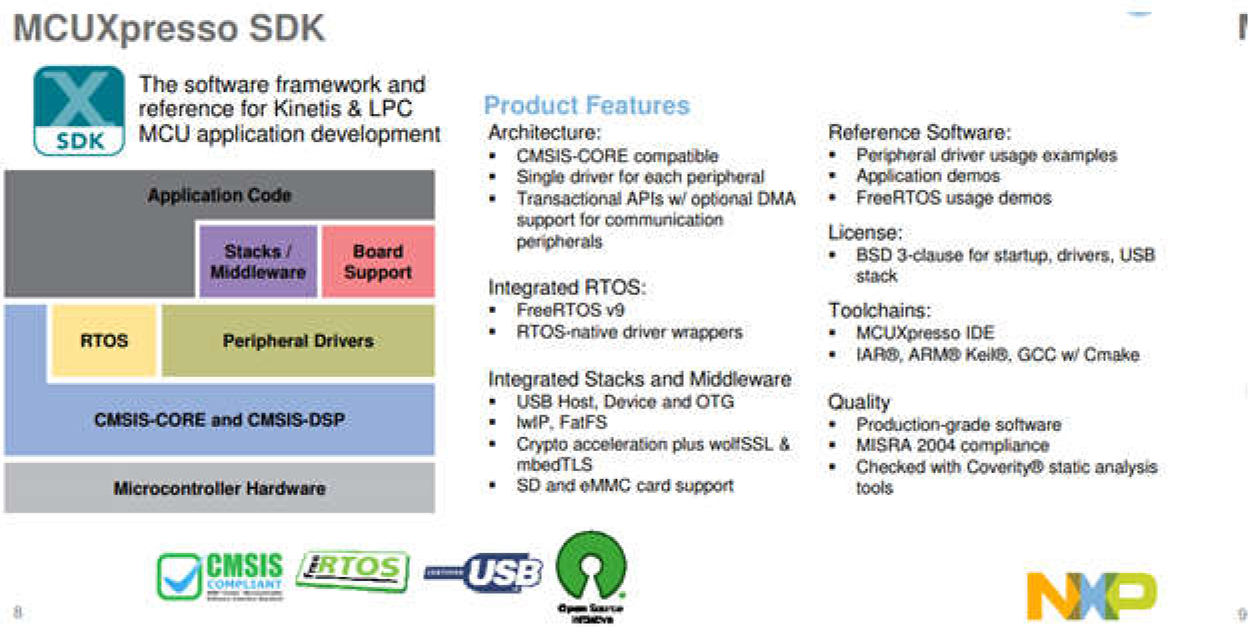
MCUXpresso Software and Tools

- IDE
- SDK
- Config Tools

For NXP’s ARM® Cortex®-M controllers

- Kinetis MCUs
- LPC Microcontrollers
- i.MX Application Processors

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf



MCUXpresso SDK

The software framework and reference for Kinetis & LPC MCU application development

Product Features

Architecture:
- CMSIS-CORE compatible
- Single driver for each peripheral
- Transactional APIs w/ optional DMA support for communication peripherals

Integrated RTOS:
- FreeRTOS v9
- RTOS-native driver wrappers

Integrated Stacks and Middleware
- USB Host, Device and OTG
- lwIP, FatFS
- Crypto acceleration plus wolfSSL & mbedTLS
- SD and eMMC card support

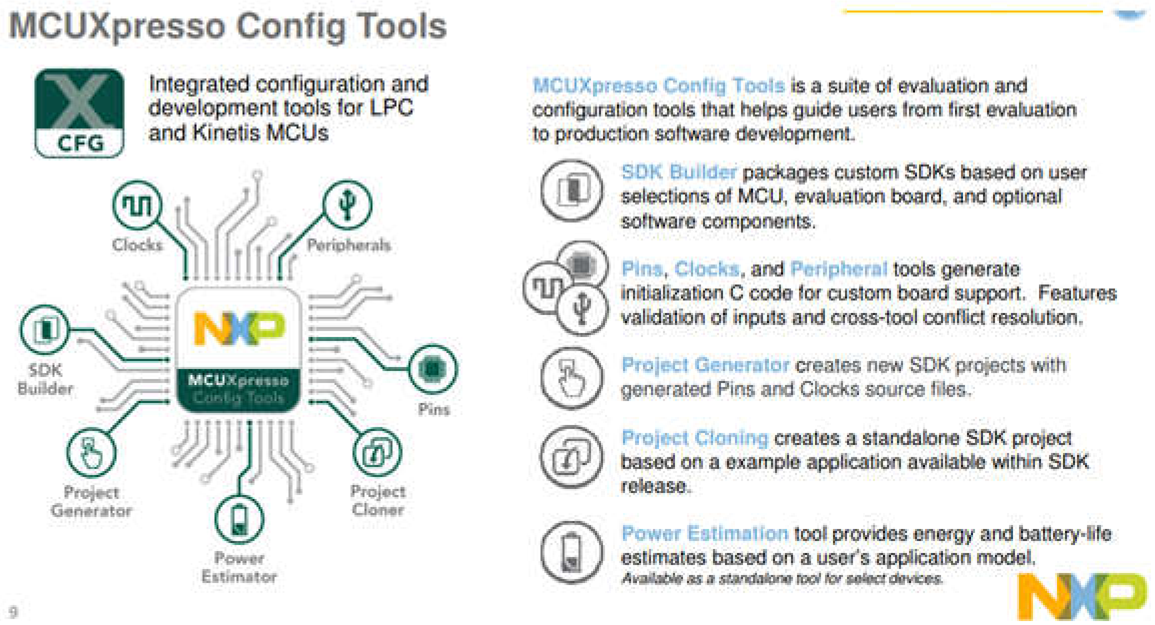
Reference Software:
- Peripheral driver usage examples
- Application demos
- FreeRTOS usage demos

License:
- BSD 3-clause for startup, drivers, USB stack

Toolchains:
- MCUXpresso IDE
- IAR®, ARM® Keil®, GCC w/ Cmake

Quality
- Production-grade software
- MISRA 2004 compliance
- Checked with Coverity® static analysis tools

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf



MCUXpresso Config Tools

Integrated configuration and development tools for LPC and Kinetis MCUs

MCUXpresso Config Tools is a suite of evaluation and configuration tools that helps guide users from first evaluation to production software development.

SDK Builder packages custom SDKs based on user selections of MCU, evaluation board, and optional software components.

Pins, Clocks, and Peripheral tools generate initialization C code for custom board support. Features validation of inputs and cross-tool conflict resolution.

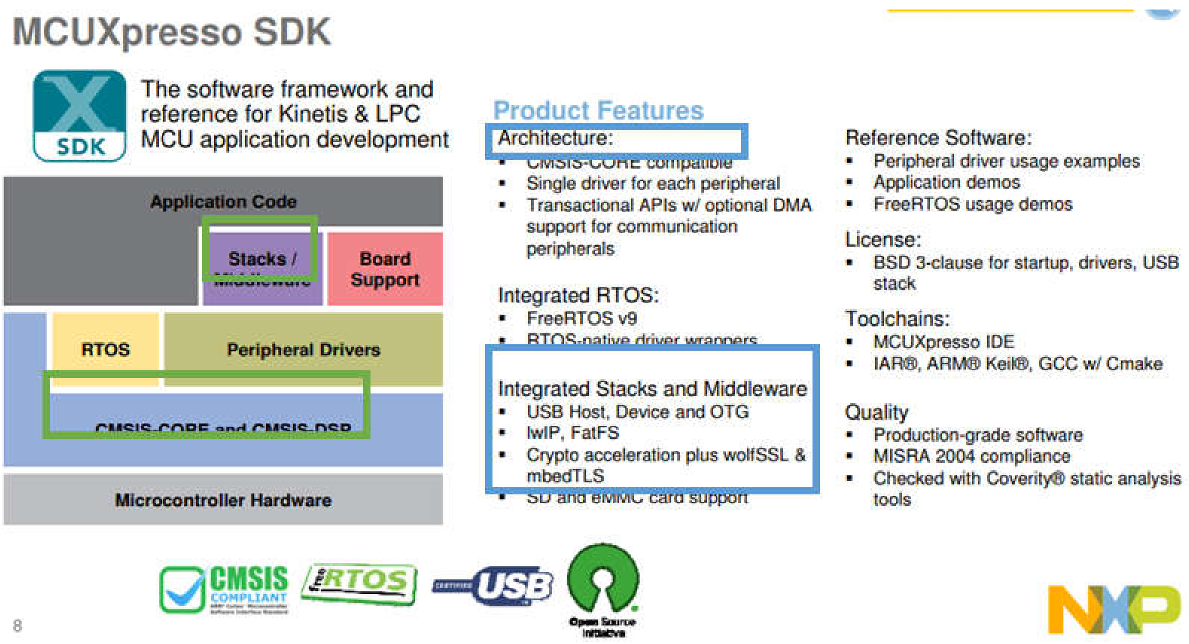
Project Generator creates new SDK projects with generated Pins and Clocks source files.

Project Cloning creates a standalone SDK project based on a example application available within SDK release.

Power Estimation tool provides energy and battery-life estimates based on a user's application model.
Available as a standalone tool for select devices.

16. The Product provides one or more generic application handler programs (e.g., MCUXpresso includes a CMSIS-CORE and CMSIS-DSP Hardware Abstraction Layer (HAL) as well as middleware that provide multiple generic application programming interfaces

(APIs). The generic programs comprise computer program code for performing generic application functions common to multiple types of hardware modules used in a communication environment (e.g., the generic code provides common and generic functions to multiple hardware modules (versions of a microcontroller, such as the Kinetis and/or LPC)). CMSIS provides vendor-independent hardware abstraction layer with generic drivers such as USB, USART, SPI, NAND, Flash, I2C and Ethernet. NXP Semiconductors includes CMSIS HAL code and middleware code in each MCUXpresso SDK distribution as well as in each custom SDK generated using MCUXpresso Config Tools. Such code is contained in at least the "CMSIS" and "middleware" folders. Certain elements of this limitation are illustrated in the screenshots below and in the screenshots referenced in connection with other elements herein.



MCUXpresso SDK

The software framework and reference for Kinetis & LPC MCU application development

Product Features

Architecture:
- CMSIS-CORE compatible
- Single driver for each peripheral
- Transactional APIs w/ optional DMA support for communication peripherals

Integrated RTOS:
- FreeRTOS v9
- RTOS-native driver wrappers

Integrated Stacks and Middleware
- USB Host, Device and OTG
- lwIP, FatFS
- Crypto acceleration plus wolfSSL & mbedTLS
- SD and eMMC card support

Reference Software:
- Peripheral driver usage examples
- Application demos
- FreeRTOS usage demos

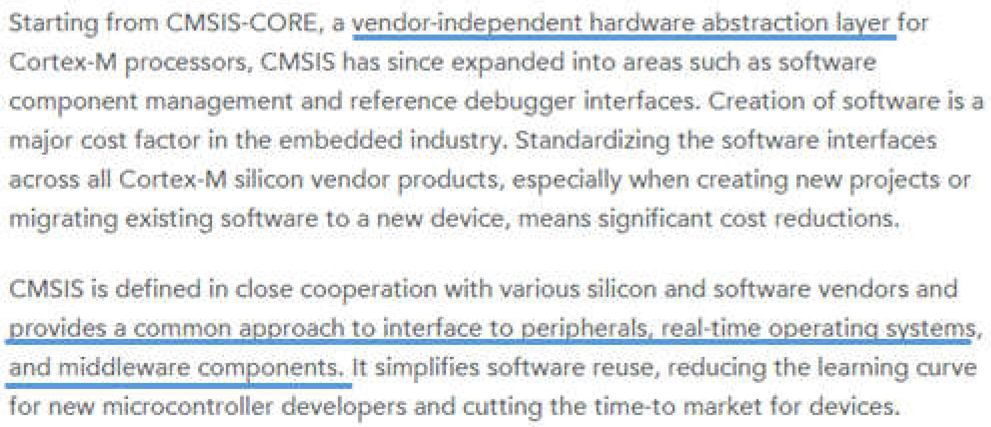
License:
- BSD 3-clause for startup, drivers, USB stack

Toolchains:
- MCUXpresso IDE
- IAR®, ARM® Keil®, GCC w/ Cmake

Quality
- Production-grade software
- MISRA 2004 compliance
- Checked with Coverity® static analysis tools

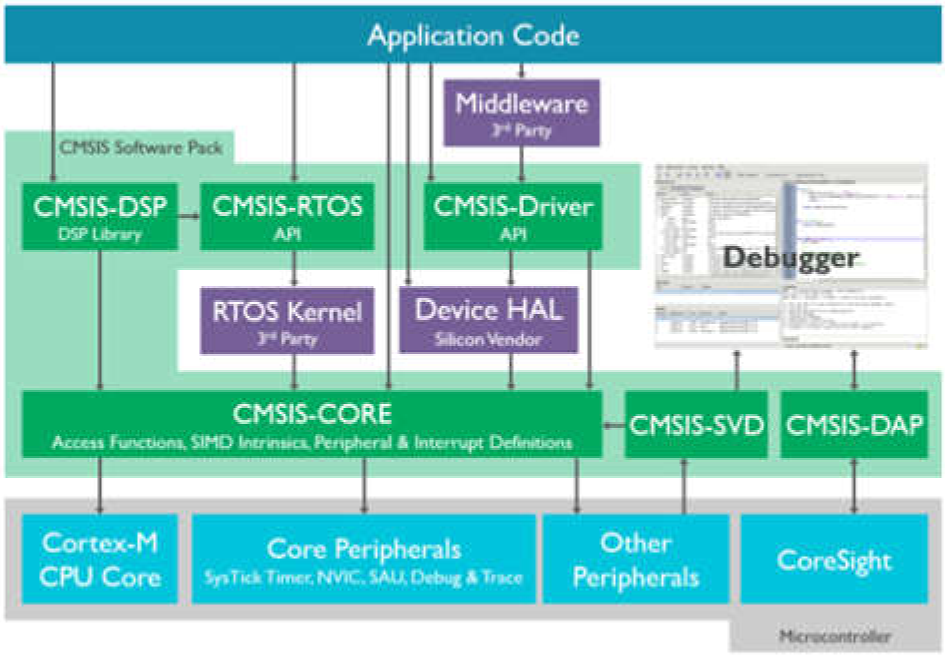
8

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf

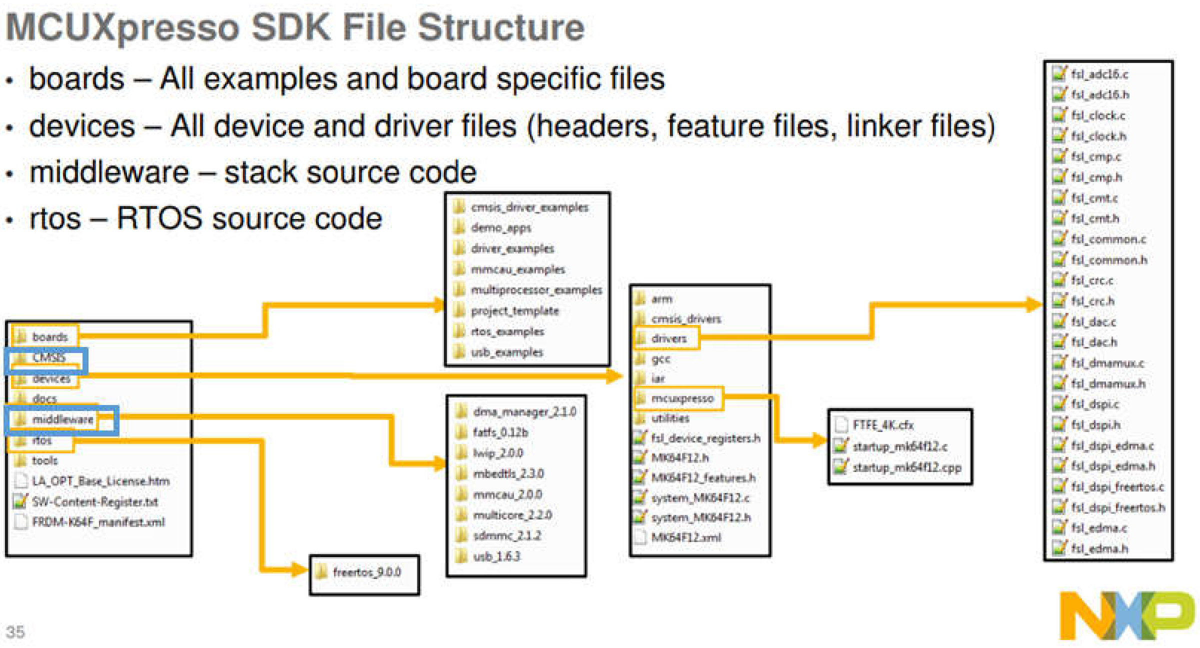


Starting from CMSIS-CORE, a vendor-independent hardware abstraction layer for Cortex-M processors, CMSIS has since expanded into areas such as software component management and reference debugger interfaces. Creation of software is a major cost factor in the embedded industry. Standardizing the software interfaces across all Cortex-M silicon vendor products, especially when creating new projects or migrating existing software to a new device, means significant cost reductions.

CMSIS is defined in close cooperation with various silicon and software vendors and provides a common approach to interface to peripherals, real-time operating systems, and middleware components. It simplifies software reuse, reducing the learning curve for new microcontroller developers and cutting the time-to market for devices.



Source: https://developer.arm.com/embedded/cmsis

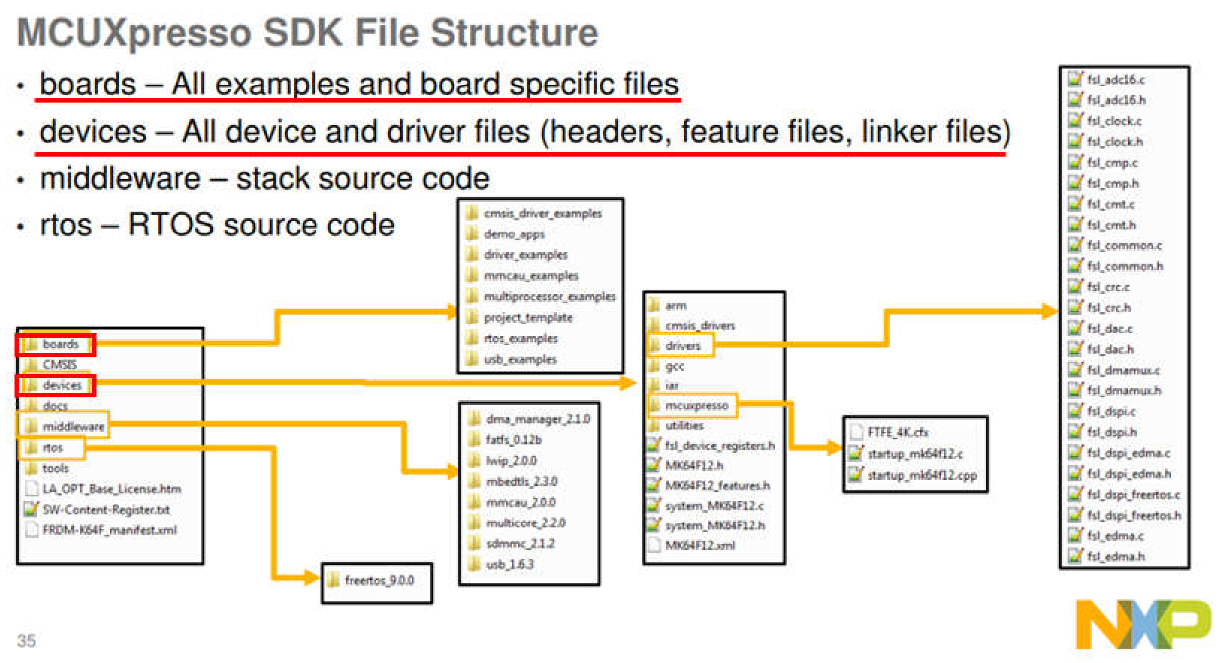


MCUXpresso SDK File Structure

- boards – All examples and board specific files
- devices – All device and driver files (headers, feature files, linker files)
- middleware – stack source code
- rtos – RTOS source code

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf

17. The Product includes generating specific application handler code to associate the generic functions with the specific functions at a device driver for at least one of the types of hardware modules. For example, in addition to the generic CMSIS HAL and middleware code, SDKs distributed and/or generated by NXP Semiconductors using MCUXpresso SDK and/or MCUXpresso Config Tools include board-specific and device-specific files that are specific to particular Kinetis and/or LPC microcontrollers. Certain elements of this limitation are illustrated in the screenshots below and in the screenshots referenced in connection with other elements herein.



MCUXpresso SDK File Structure

- boards – All examples and board specific files
- devices – All device and driver files (headers, feature files, linker files)
- middleware – stack source code
- rtos – RTOS source code

```
boards
CMSIS
devices
docs
middleware
rtos
tools
LA_OPT_Base_License.htm
SW-Content-Register.txt
FRDM-K64F_manifest.xml
```

```
cmsis_driver_examples
demo_apps
driver_examples
mmcau_examples
multiprocessor_examples
project_template
rtos_examples
usb_examples
```

```
dma_manager_2.1.0
fatfs_0.12b
lwip_2.0.0
mbedtls_2.3.0
mmcau_2.0.0
multicore_2.2.0
sdmmc_2.1.2
usb_1.6.3
```

```
freertos_9.0.0
```

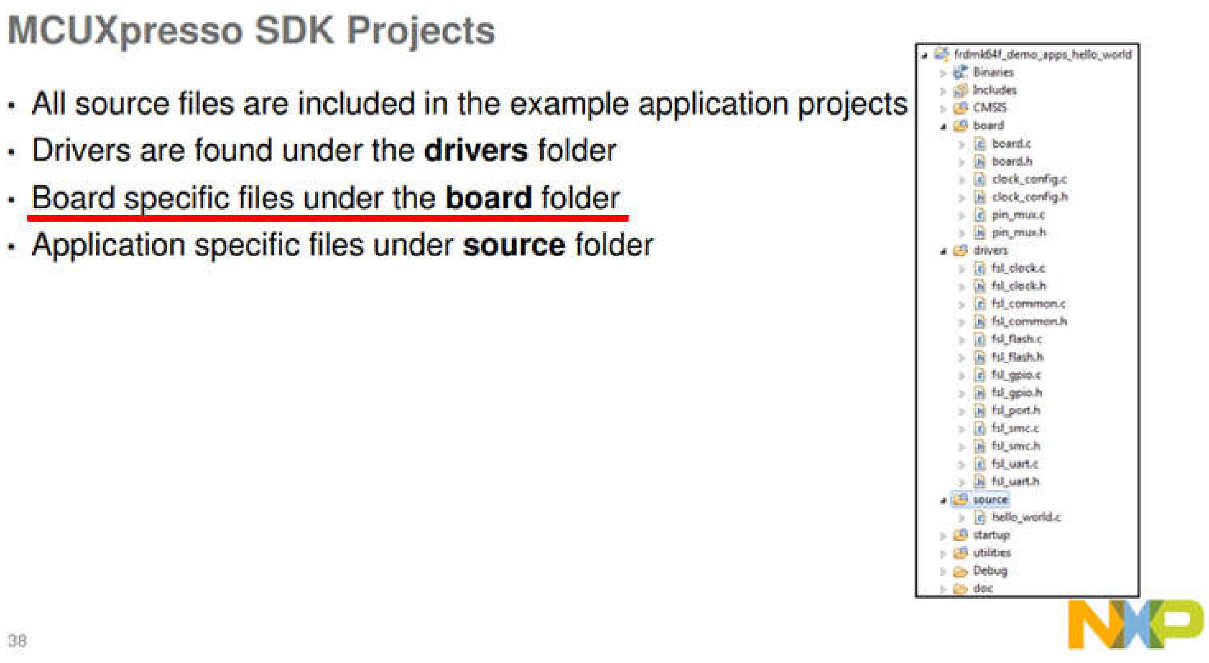
```
arm
cmsis_drivers
drivers
gcc
iar
mcuxpresso
utilities
fsl_device_registers.h
MK64F12.h
MK64F12_features.h
system_MK64F12.c
system_MK64F12.h
MK64F12.xml
```

```
FTFE_4K.cfx
startup_mk64f12.c
startup_mk64f12.cpp
```

```
fsl_adc16.c
fsl_adc16.h
fsl_clock.c
fsl_clock.h
fsl_cmp.c
fsl_cmp.h
fsl_cmt.c
fsl_cmt.h
fsl_common.c
fsl_common.h
fsl_crc.c
fsl_crc.h
fsl_dac.c
fsl_dac.h
fsl_dmamux.c
fsl_dmamux.h
fsl_dspi.c
fsl_dspi.h
fsl_dspi_edma.c
fsl_dspi_edma.h
fsl_dspi_freertos.c
fsl_dspi_freertos.h
fsl_edma.c
fsl_edma.h
```

35

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf



MCUXpresso SDK Projects

- All source files are included in the example application projects
- Drivers are found under the **drivers** folder
- Board specific files under the **board** folder
- Application specific files under **source** folder

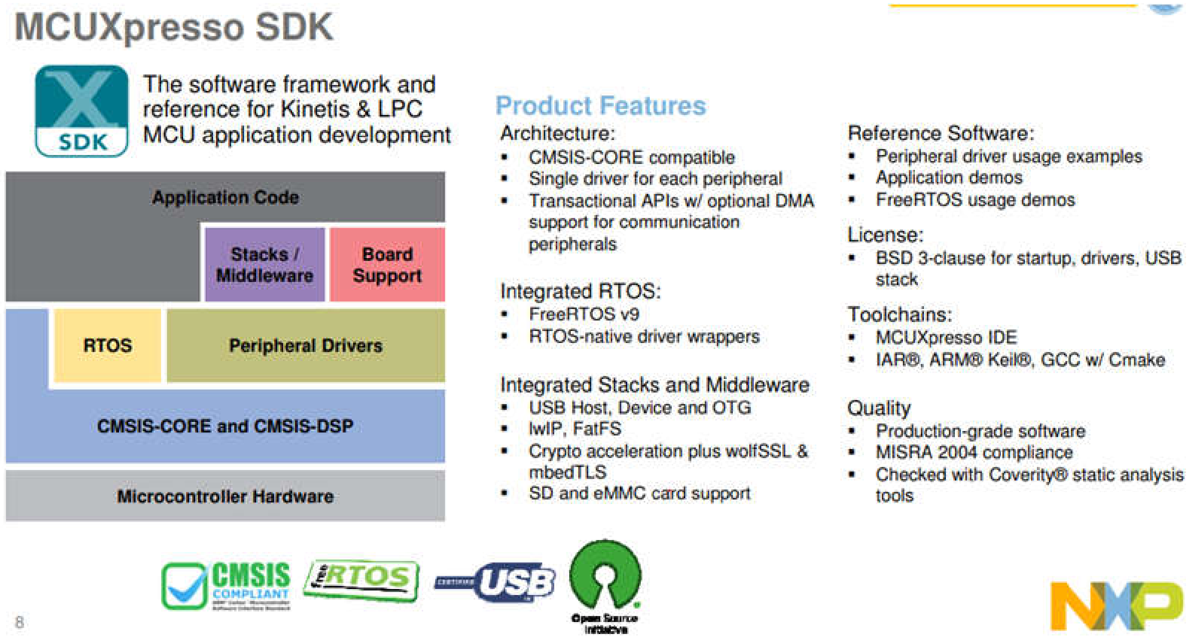
```
frdmk64f_demo_apps_hello_world
  Binaries
  Includes
  CMSIS
  board
    board.c
    board.h
    clock_config.c
    clock_config.h
    pin_mux.c
    pin_mux.h
  drivers
    fsl_clock.c
    fsl_clock.h
    fsl_common.c
    fsl_common.h
    fsl_flash.c
    fsl_flash.h
    fsl_gpio.c
    fsl_gpio.h
    fsl_port.h
    fsl_smc.c
    fsl_smc.h
    fsl_uart.c
    fsl_uart.h
  source
    hello_world.c
  startup
  utilities
  Debug
  doc
```

38

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf

18. The Product generates specific application handler code and defines a specific element in the specific code to be handled by one of the generic application functions for that hardware module. For example, the board-specific code in each MCUXpresso SDK package is

contained within the folder named "board," the processor-specific code is within the folder named "devices," while the application specific code is contained within the folder named "source." The specific elements such as one or more functions in this specific application handler code are defined to be handled by one or more generic application functions in the CMSIS HAL. When specific functions are written for handling defined specific elements, the specific functions must be registered. The HAL, middleware and/or board- and device-specific code accordingly contains data structures that register and embed the required functions. Certain elements of this limitation are illustrated in the screenshots below and in the screenshots referenced in connection with other elements herein.



MCUXpresso SDK

The software framework and reference for Kinetis & LPC MCU application development

Product Features

Architecture:
- CMSIS-CORE compatible
- Single driver for each peripheral
- Transactional APIs w/ optional DMA support for communication peripherals

Integrated RTOS:
- FreeRTOS v9
- RTOS-native driver wrappers

Integrated Stacks and Middleware
- USB Host, Device and OTG
- lwIP, FatFS
- Crypto acceleration plus wolfSSL & mbedTLS
- SD and eMMC card support

Reference Software:
- Peripheral driver usage examples
- Application demos
- FreeRTOS usage demos

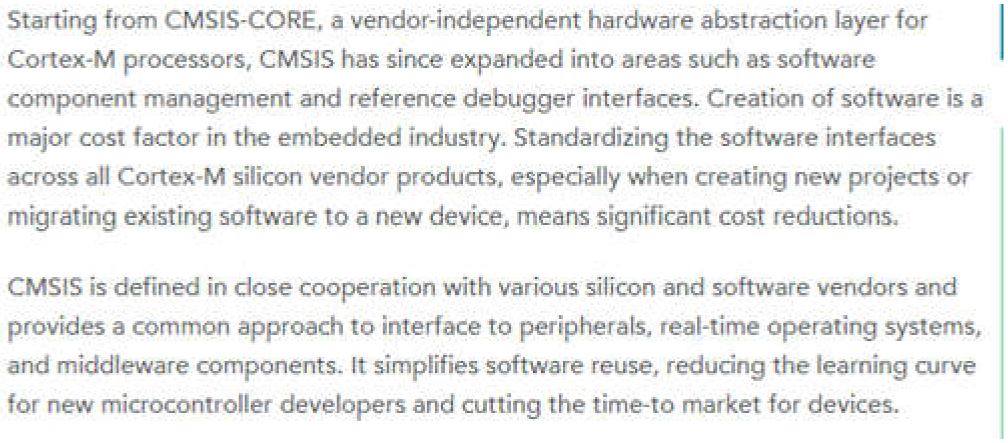
License:
- BSD 3-clause for startup, drivers, USB stack

Toolchains:
- MCUXpresso IDE
- IAR®, ARM® Keil®, GCC w/ Cmake

Quality
- Production-grade software
- MISRA 2004 compliance
- Checked with Coverity® static analysis tools

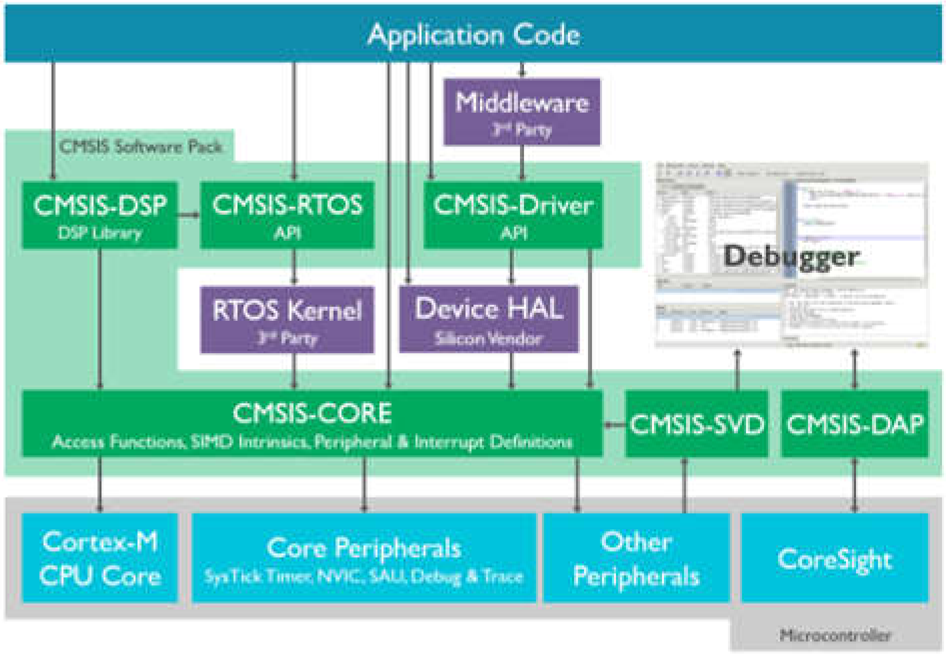
8

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf



Starting from CMSIS-CORE, a vendor-independent hardware abstraction layer for Cortex-M processors, CMSIS has since expanded into areas such as software component management and reference debugger interfaces. Creation of software is a major cost factor in the embedded industry. Standardizing the software interfaces across all Cortex-M silicon vendor products, especially when creating new projects or migrating existing software to a new device, means significant cost reductions.

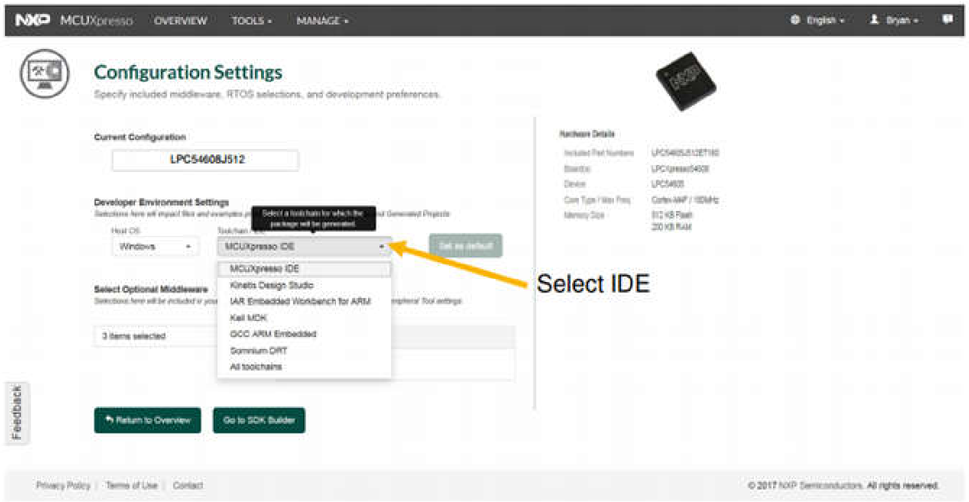
CMSIS is defined in close cooperation with various silicon and software vendors and provides a common approach to interface to peripherals, real-time operating systems, and middleware components. It simplifies software reuse, reducing the learning curve for new microcontroller developers and cutting the time-to market for devices.



Source: https://developer.arm.com/embedded/cmsis

19. When a specific application is needed for a particular hardware, the generic functions and the specific functions are compiled together to yield a machine readable code. NXP Semiconductors and/or its customers compile the generic functions and the specific functions using MCUXpresso IDE or any other IDE supported by the MCUXpresso framework. Certain elements of this limitation are illustrated in the screenshots below and in the screenshots referenced in connection with other elements herein.

Additional Configuration Settings: Choose an IDE



Configuration Settings

Specify included middleware, RTOS selections, and development preferences.

Current Configuration

LPC54608J512

Developer Environment Settings

Selections here will impact files and examples p... ...d Generated Projects

Select a toolchain for which the package will be generated.

Host OS: Windows

Toolchain / IDE: MCUXpresso IDE

- MCUXpresso IDE
- Kinetis Design Studio
- IAR Embedded Workbench for ARM
- Keil MDK
- GCC ARM Embedded
- Somnium DRT
- All toolchains

Select Optional Middleware

Selections here will be included in you... ...eripheral Tool settings

3 items selected

Hardware Details

Included Part Numbers	LPC54608J512ET180
Board(s)	LPCXpresso54608
Device	LPC54608
Core Type / Max Freq	Cortex-M4F / 180MHz
Memory Size	512 KB Flash 200 KB RAM

© 2017 NXP Semiconductors. All rights reserved.

Select IDE

22

Source: https://www.nxp.com/docs/en/supporting-information/APF-DES-T2744-MCUXpressor.pdf

20. Defendant's actions complained of herein will continue unless Defendant is enjoined by this court.

21. Defendant's actions complained of herein are causing irreparable harm and monetary damage to Plaintiff and will continue to do so unless and until Defendant is enjoined and restrained by this Court.

22. Plaintiff is in compliance with 35 U.S.C. § 287.

PRAYER FOR RELIEF

WHEREFORE, Plaintiff asks the Court to:

(a) Enter judgment for Plaintiff on this Complaint on all causes of action asserted herein;

(b) Enter an Order enjoining Defendant, its agents, officers, servants, employees, attorneys, and all persons in active concert or participation with Defendant who receive notice of the order from further infringement of United States Patent No. 7,069,546 (or, in the alternative, awarding Plaintiff a running royalty from the time of judgment going forward);

(c) Award Plaintiff damages resulting from Defendant's infringement in accordance with 35 U.S.C. § 284;

(d) Award Plaintiff pre-judgment and post-judgment interest and costs; and

(e) Award Plaintiff such further relief to which the Court finds Plaintiff entitled under law or equity.

Dated: December 11, 2017

Respectfully submitted,

/s/ Jay Johnson
JAY JOHNSON
State Bar No. 24067322
D. BRADLEY KIZZIA
State Bar No. 11547550
KIZZIA JOHNSON, PLLC
1910 Pacific Ave., Suite 13000
Dallas, Texas 75201
(214) 451-0164
Fax: (214) 451-0165
jay@kjpllc.com
bkizzia@kjpllc.com

ATTORNEYS FOR PLAINTIFF

EXHIBIT A