



System Configuration Tools for NXP's Kinetis, LPC and i.MX RT product families

MCUXpresso Config Tools

Integrated suite of configuration tools, including pins, clocks, and peripherals tools that help guide users from first evaluation to production software development.

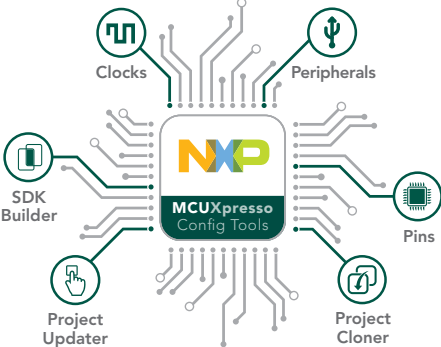
Designing with NXP's Kinetis and LPC microcontrollers based on Arm® Cortex®-M cores is made easier with the MCUXpresso Config Tools, which allow developers to quickly build a custom SDK, update existing IDE projects, and leverage pins, clocks and peripherals tools to generate initialization C code for custom board support.

FEATURES

MCUXpresso Config Tools are available directly within the MCUXpresso IDE. For other IDE options, a standalone desktop version of the Config Tools is available for download. The MCUXpresso Config Tools include:

- ▶ **SDK Builder** packages custom SDKs based on user selections of MCU, evaluation board, and optional software components for use with MCUXpresso software and tools.
- ▶ **Pins tool** assigns internal signals to external pins, sets electrical properties, I/O conflict resolution options and generates ANSI-C source code that drops into the MCUXpresso SDK environment.
- ▶ **Clocks tool** for a graphical representation of the MCU clock tree system and interactive user controls as well as assistance with system fine-tuning.

MCUXpresso CONFIG TOOLS



- ▶ **Peripherals tool** generates initialization code for GPIO, UART, ADC and other peripherals for use inside SDK drivers. Peripherals tool can also configure higher level application code for USB projects.
- ▶ **Project Cloning** creates a standalone SDK project based on an example application available within the SDK.
- ▶ **Project Update** works directly with existing SDK-based IDE projects with generated Pins, Clocks, and Peripheral source files.



SDK BUILDER

The MCUXpresso SDK Builder generates a downloadable SDK archive based on user selections of device, development board, toolchain, host OS, middleware and more. With a few simple configuration selections, the MCUXpresso SDK is ready for download and use with desktop MCUXpresso software and tools. These custom packages are inherently smaller in size and make the MCUXpresso SDK much more manageable than other “one size fits all” SDKs.

SDK Builder
Generate a downloadable SDK archive for use with desktop MCUXpresso Tools.

Current Configuration
FRDM-K64F

Review Optional Middleware
Items here will be included in your SDK download. These selections can be edited using the Tools-> Configurations Settings page.

Selected Middleware
CMSIS DSP Library, FatFS, USB stack, IWP, FreeRTOS.

Hardware Details

Board	FRDM-K64F
Device	MK64F12
Core Type / Max Freq	Cortex-M4F / 120MHz
Memory Size	1024 KB Flash 256 KB RAM

SDK Details

SDK Version:	KSDK 2.2.0 API Reference
Host OS:	Windows
Toolchain:	MCUXpresso IDE

Click the link below to request this specific MCUXpresso SDK Build
In general, SDK builds should complete within a few minutes.
You will be notified via email and notifications in the upper right corner of this webpage.

Package Name
SDK_2.2_FRDM-K64F

Request Build

Feedback

Privacy Policy | Terms of Use | Contact

© 2016 NXP Semiconductors. All rights reserved.

PINS TOOL

The MCUXpresso Pins Tool is used for pin routing configuration, validation, and code generation. It provides pin settings for signal muxing, electrical properties, and run-time configurations. Selections can be easily captured using the graphical package view or searchable/sortable spreadsheet view. The MCUXpresso Pins Tool generates easy to read ANSI-C initialization code suitable for C or C++ projects.

Pin Configuration Table:

Pin	Pin name	Label	Identifier	GPIO	UART	FT
7	PTB6/SP1L_PCS3...	J15[G1]/SD_CAR...	SDHC_CD	PTE6	UART3_CTS_b	FT
10	USB0_DP	J22[3]/K64_MIC...	USB_DP			
11	USB0_DM	J22[2]/K64_MIC...	USB_DM			
28	XTAL32	Y3[1]/XTAL32_R...	XTAL32K			
29	EXTAL32	Y3[2]/EXTAL32_...	EXTAL32K			
31	ADCO_SE17/PTE...	J2[20]/U8[4]/I2...	ACCEL_SCL	PTE24	UART4_TX	
32	ADCO_SE18/PTE...	J2[18]/U8[6]/I2...	ACCEL_SDA	PTE25	UART4_RX	
33	PTE26	J2[1]/D12[4]/L...	LED_GREEN	PTB26	UART4_CTS_b	
38	PTA4/LLWU_P3/...	SW3		PTA4		FT
39	PTA5/USB_CLK1...	U13[17]/RMD0_R...	RMD0_RXER	PTA5		FT
42	CMP2_IN0/PTA...	U13[12]/RMD0_R...	RMD0_RXD1	PTA2		FT
43	CMP2_IN1/PTA...	U13[13]/RMD0_R...	RMD0_RXD0	PTA13		FT
44	PTA14/SPD_PCS...	U13[15]/RMD0_...	RMD0_CRS_DV	PTA14	UART0_TX	
45	PTA15/SPD_SCK...	U13[19]/RMD0_T...	RMD0_TXEN	PTA15	UART0_RX	
46	PTA16/SPD_SO...	U13[20]/RMD0_T...	RMD0_TXD0	PTA16	UART0_CTS_b[...]	
47	ADCO_SE17/PTA...	U13[21]/RMD0_T...	RMD0_TXD1	PTA17	UART0_RTS_b	
50	EXTAL0/PTA18/...	U13[16]/RMD0_R...	EXTAL0_RMD0_RX...	PTA18		FT
53	ADCO_SE8/ADC...	U13[10]/RMD0_...	RMD0_MDC0	PTB9		FT
54	ADCO_SE9/ADC...	U13[11]/RMD0_...	RMD0_MDC	PTB1		FT
62	PTB16/SPB_SO...	U17[4]/UART0_RX	DEBUG_UART_RX	PTB16	UART0_RX	FT
63	PTB17/SPB_SIN...	U10[1]/UART0_TX	DEBUG_UART_TX	PTB17	UART0_TX	FT
67	PTB21	D12[2]/LEDRGB...	myLED	PTB21		
68	PTB22	D12[1]/LEDRGB...	LED_RED	PTB22		
78	CMP0_IN0/PTC...	U8[11]/SW2	SW2_ACCEL_INT1	PTC6		
85	PTC13/UART4...	U8[9]	ACCEL_INT2	PTC13	UART4_CTS_b	

Routed Pins Table:

#	Peripheral	Signal	Route to	Label	Identifier	Direction	Slew rate	Open drain	Drive strength	Pull select	Pull enable	Passive filter	Digital filter
67	GPIOB	GPIO_21	PTB21	D12[3]/LEDRGB_BLUE	myLED	Output	Slow	Disabled	Low	Pullup	Disabled	Disabled	n/a
68	GPIOB	GPIO_22	PTB22	D12[1]/LEDRGB_RED	LED_RED	Output	Slow	Disabled	Low	Pullup	Disabled	Disabled	n/a
33	GPIOE	GPIO_26	PTE26	J2[1]/D12[4]/LEDRGB_GREEN	LED_GREEN	Output	Slow	Disabled	Low	Pullup	Disabled	Disabled	n/a

CLOCKS TOOL

The MCUXpresso Clocks Tool allows the user to easily configure the initialization of the system clocks (core, system, bus, peripheral clocks) and to generate C code with MCUXpresso SDK clock initialization functions and configuration structures.

Visual inspection of the configured clock paths is available using the graphical clock tree. The MCUXpresso Clock Tool validates clock settings and provide calculations of the resulting clock frequencies.

The screenshot displays the MCUXpresso Clocks Tool interface. On the left, a 'Path Details: Core clock' panel lists various configuration parameters such as Core clock (120 MHz), CORECLK Frequency (120 MHz), and PLL Frequency (120 MHz). The central area features a 'Clocks Diagram' showing a complex network of clock sources, dividers, and multipliers, including components like PLL, PFD, and various clock outputs (MCG, SIM, OUTDIV). On the right, a 'Module Clocks' panel shows a list of generated clock outputs with their respective frequencies. Below the diagram, a 'Code successfully generated.' message is visible, along with a 'Problems' panel showing no issues.

PERIPHERALS TOOL

The MCUXpresso Peripherals Tool allows users to add desired peripherals to their designs, including UART, ADC, SPI, I²C, and more. It will generate initialization structures for the MCUXpresso SDK drivers and offers an easy-to-use quick selection feature which allows starting references to be pre-populated based on peripheral selections. It will also allow the user to easily create example system code for USB applications by establishing the configuration of the device and selecting callbacks to implement on top of the existing MCUXpresso SDK middleware and drivers. In addition, users can also quickly validate their selections to confirm that the settings are conflict free, and an alert will call out conflicts when they arise.

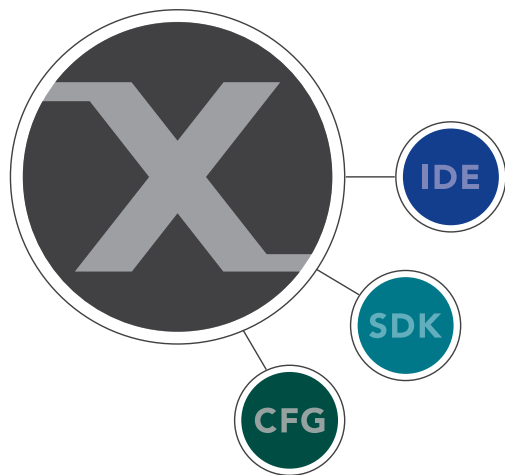
The screenshot displays the MCUXpresso Peripherals Tool interface. On the left, a 'Peripherals' panel shows a list of selected components including 'USB_1', 'ADC_1', 'FlexIO_DCC_1', 'GPIO_1', and 'LPUART_1'. The central area shows the configuration for 'Universal Serial Bus (USB)', including fields for Vendor ID, Product ID, Manufacturer, and Product, as well as a table for 'Supported interfaces' and 'Endpoints'. On the right, a 'Code Preview' panel shows the generated C code for the USB peripheral, including initialization functions and configuration structures. Below the code, a 'Problems' panel shows no issues.

MCUXpresso SOFTWARE AND TOOLS

MCUXpresso Config Tools are part of the cohesive suite of MCUXpresso Software and Tools, and are inherently compatible with MCUXpresso Software Development Kit (SDK) and the MCUXpresso Integrated Development Environment (IDE).

Designed to ease and accelerate embedded system development and optimization, the MCUXpresso Software and Tools bring high quality comprehensive enablement to NXP's Powerhouse Portfolio of microcontrollers built on Arm Cortex-M cores. A shared software environment enables easy migration and scalability between Kinetis, LPC, and i.MX RT product portfolios.

MCUXpresso Software and Tools



GET STARTED:

Learn more:

www.nxp.com/mcuxpresso/config

Join the MCUXpresso Config Tools community: <https://community.nxp.com/community/mcuxpresso/mcuxpresso-config>

Professional Support & Services:

www.nxp.com/services

www.nxp.com/mcuxpresso/config

NXP and the NXP logo are trademarks of NXP B.V. All other product or service names are the property of their respective owners. All rights reserved. © 2018 NXP B.V.

Date of Release: December 2018
Document Number: MCUXPRESSOCFTFS REV 2