

# GET STARTED FAST WITH THIS COMPREHENSIVE ENABLEMENT OFFERING FOR LPC800 MCUS

BRENDON SLADE  
DIRECTOR, LPC ECOSYSTEM

JUNE 13, 2018



SECURE CONNECTIONS  
FOR A SMARTER WORLD

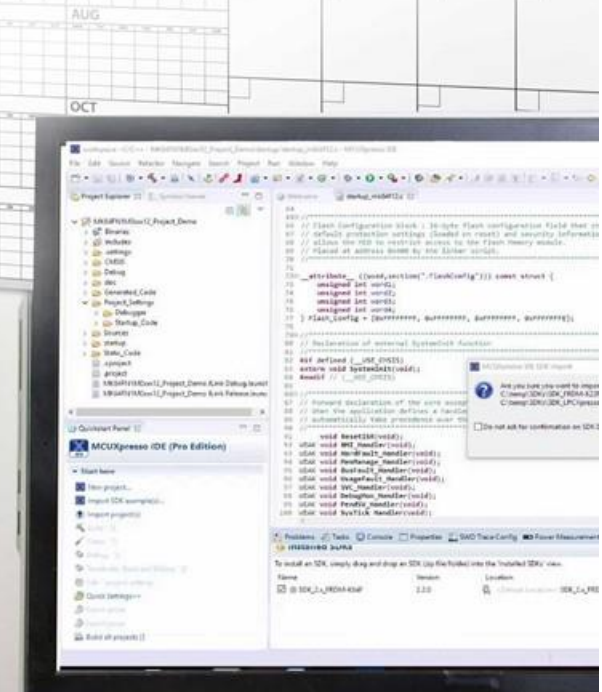
NXP and the NXP logo are trademarks of NXP B.V. All other product or service names are the property of their respective owners. © 2017 NXP B.V.

PUBLIC



# GET TO MARKET FAST WITH LPC800 MCUS - A LOW-COST, ENTRY-LEVEL, 8-BIT ALTERNATIVE FOR YOUR NEXT DESIGN

- Part I: Thursday, May 31, 10 AM Central (1 hour)  
**Thinking about migrating from 8-bit? Wait no longer - LPC80x MCUs are your 32-bit answer**
- Part II: Thursday, June 7, 10 AM Central (1 hour)  
**Creative ways to leverage the LPC804 MCU's integrated programmable logic feature**
- Part III: Thursday, June 14, 10 AM Central (1 hour)  
**Get started fast with this comprehensive enablement offering for LPC800 MCUs**
- Part IV: Thursday, June 21, 10 AM Central (1 hour)  
**Got NFC? LPC8N04 does – learn how to leverage this unique feature in your next design?**



# Agenda

- LPC800 Recap
- MCUXpresso SW & Tools Overview
- Live demo of MCUXpresso tools
- Conclusion / Where next



# LPC Microcontrollers

Broad Market Leader

## Architecting Scalable MCU Families with Flexible Integration Enabling Fast Time & Platform Re-use

1

Innovative ARM  
MCU Portfolio.

2

Ecosystem &  
Partners.

3

Supply, Longevity  
& Quality.

4

Local  
Support Network.

5

Extensive  
Software & Tools.

» Accelerating Transition from 8-bit to **Entry-level Cortex-M0+ based MCUs**

# LPC 32-bit Microcontrollers for the Mass Market

Over 1B units shipped

>400 part numbers

Thriving ecosystem

Complementary professional development suite (HW/SW)

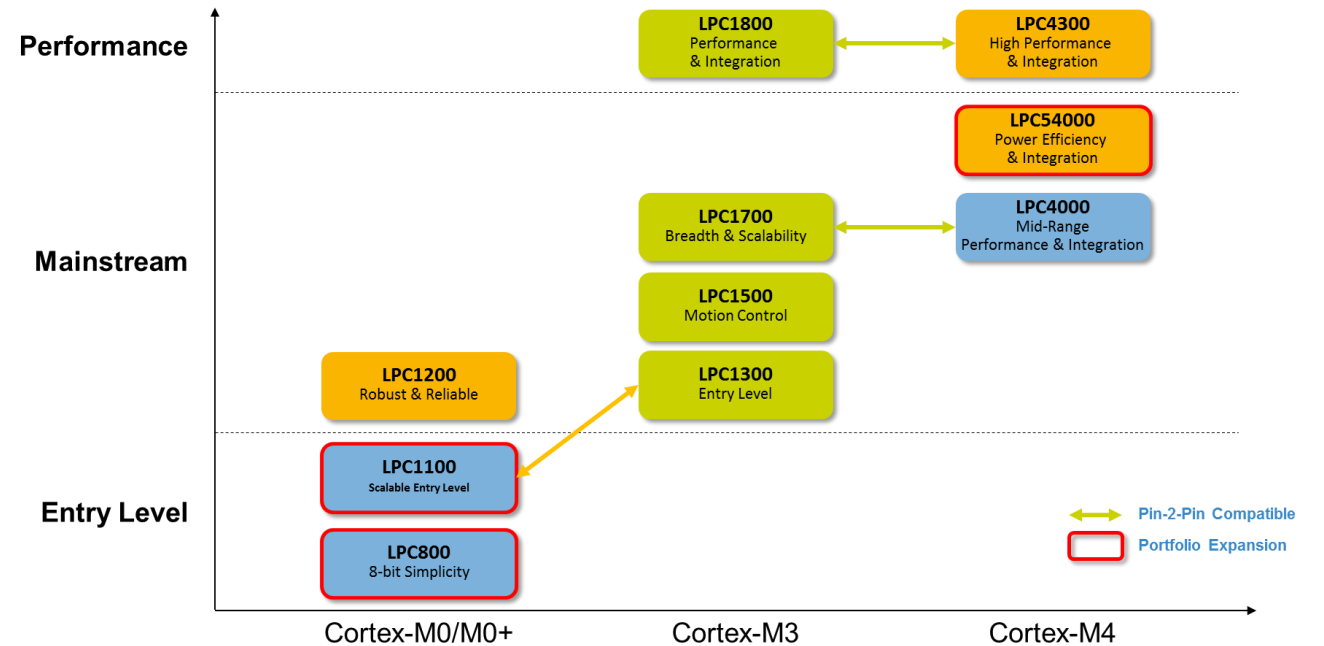
## Open Development Environment

- MCUXpresso IDE with Easy to Use Software Code Bundles
- Development, Debug & Expansion Boards
- Developer Community



Easy Development

## Complete Portfolio of Cortex-M MCUs



Scalable Expansion

# LPC800 Enablement Overview

## Runtime Software

NXP Solutions:



**MCUXpresso Software and Tools**

- IDE
- SDK
- Config Tools

For NXP Cortex-M controllers

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





## Software Development Tools

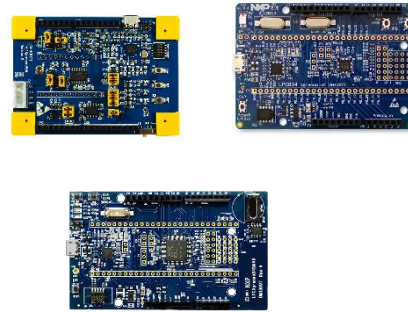
IDE / Toolchains:

RTOS, Middleware Partners:



## Hardware Development Tools

Evaluation Kits:



Partner Solutions



## Application Specific

- Cloud connectivity
- NFC / NTAG
- PLU integration

## Connectivity Solutions



THINGSTREAM

## Support

Broad Market:

- OOB Walkthroughs
- NXP Community
- Solution Designs
- Application Notes
- Schematics



High Touch:

- Professional Support
- Professional Services



Comprehensive frameworks and solutions for low-power, connected, and secure embedded systems

Industry leading IDE support and intuitive software configuration tools to accelerate application development

Low cost hardware platforms for evaluation and application development. Partner solutions for hardware debugging solutions

Software frameworks and development tools for targeted applications and certified connectivity solutions

Get started quickly and get the support you need, when you need it

# MCUXPRESSO OVERVIEW



# Introducing ...



## MCUXpresso Software and Tools

for Kinetis and LPC microcontrollers



### MCUXpresso IDE

Edit, compile, debug and optimize in an intuitive and powerful IDE



### MCUXpresso SDK

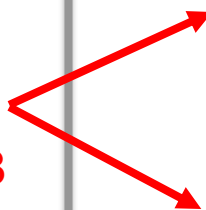
Runtime software including peripheral drivers, middleware, RTOS, demos and more



### MCUXpresso Config Tools

Online and desktop tool suite for system configuration and optimization

LPC800 Family  
support launched  
in May 2018



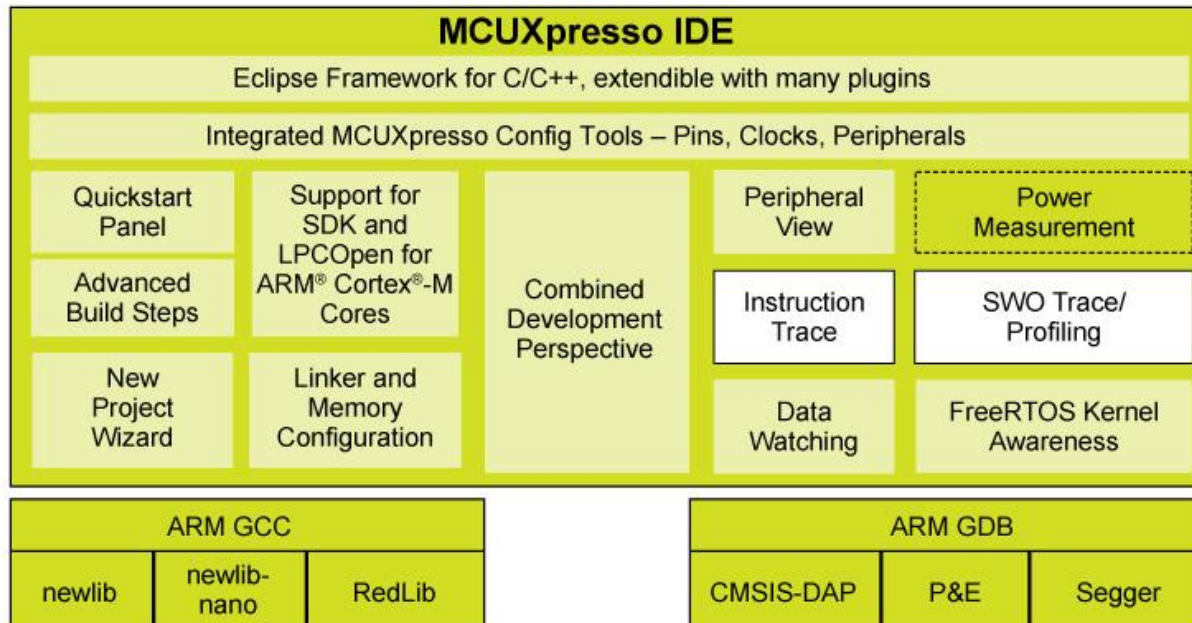




# MCUXpresso IDE



Free Eclipse and GCC-based IDE for C/C++ development on Kinetis and LPC MCUs



 For supported boards     With supported probes

## Product Features

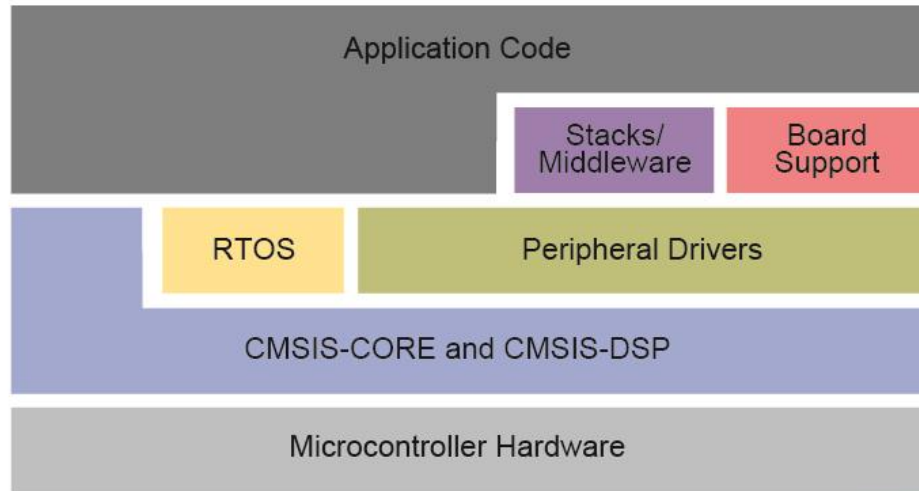
- **Feature-rich, unlimited code size**, optimized for **ease-of-use**, based on **industry standard Eclipse** framework for **NXP's Kinetis** and **LPC MCUs**, and **i.MX RT1050 Crossover Processors**
- **Application development** with Eclipse and GCC-based IDE for advanced **editing, compiling** and **debugging**
- **Integrated configuration tools** for easy project updating
- Supports **custom** development boards, **Freedom, Tower** and **LPCXpresso** boards with debug probes from **NXP, P&E** and **Segger**
- **Advanced Trace Features**, including instruction trace, SWO trace and profiling
- Non-intrusive, **real-time data watch and printf** via SWD
- **No activation** needed and community based support



# 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
- Amazon Web Service IoT
- QCA WiFi Stacks
- USB Type-C Power Delivery Stack
- 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

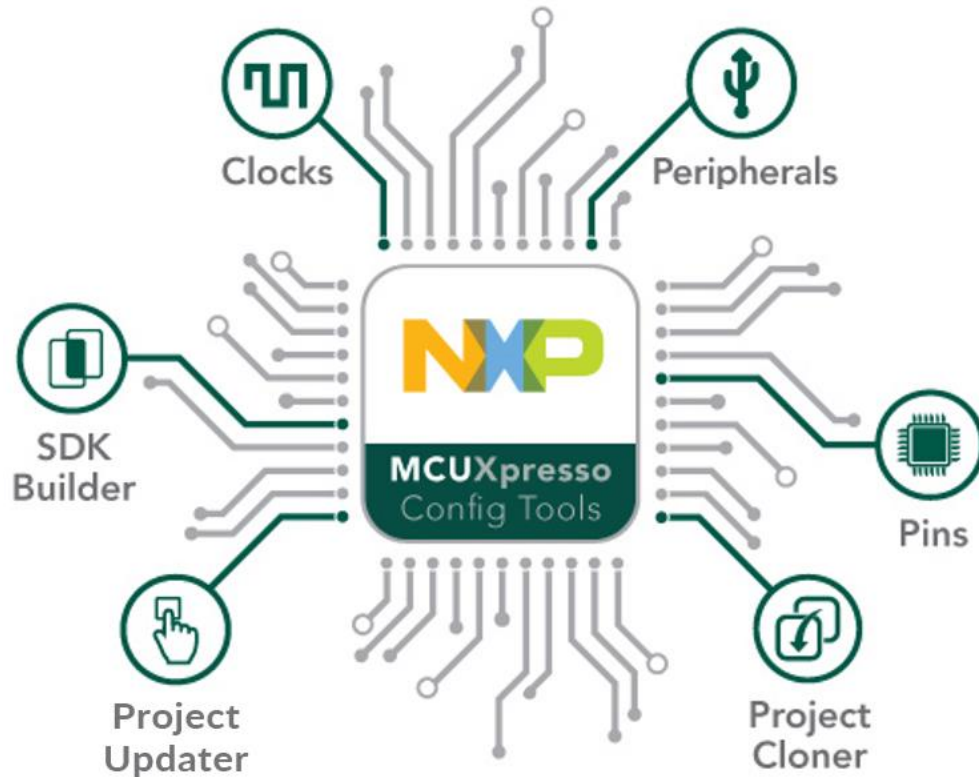
- Production-grade software
- MISRA 2004 compliance
- Checked with Coverity® static analysis tools

Also available as CMSIS Device Family Pack

# MCUXpresso Config Tools



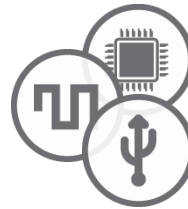
Integrated configuration and development tools for LPC and Kinetis MCUs



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

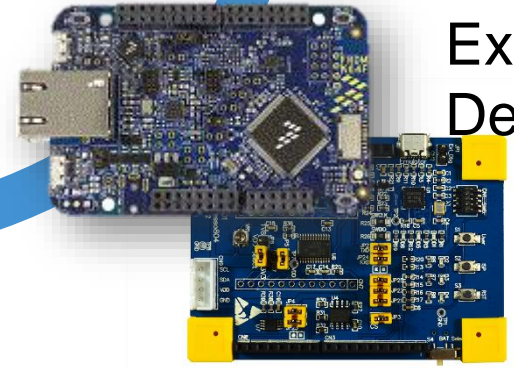
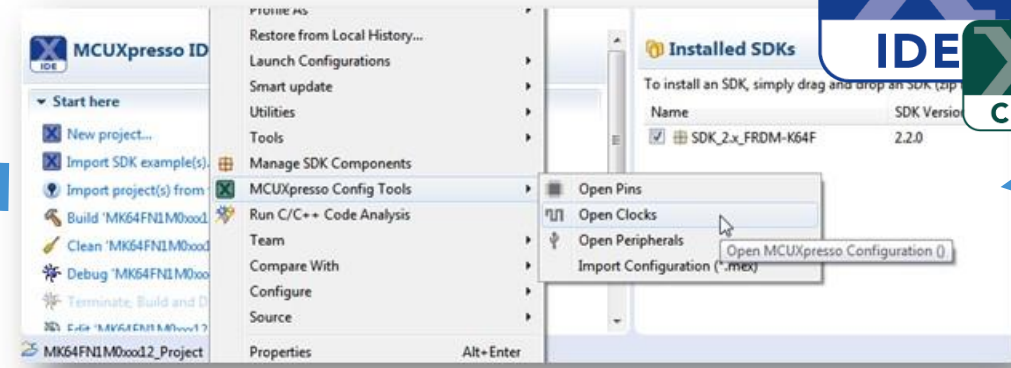
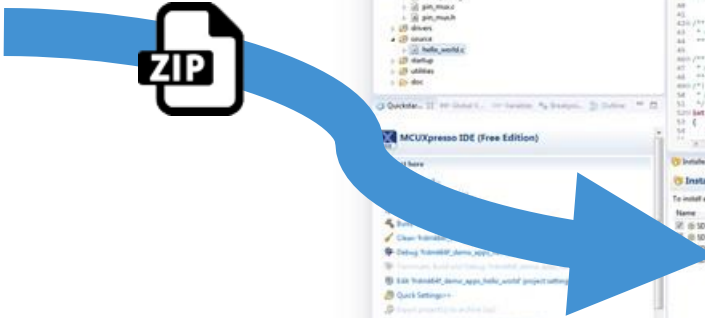
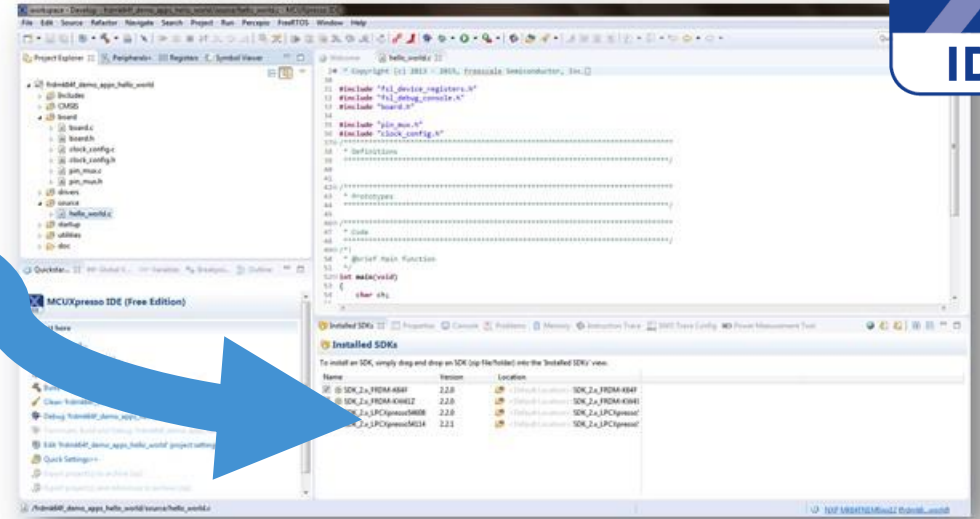


**Project Update** works directly with existing SDK-based IDE projects with generated Pins and Clock source files (IAR/Keil tools)



**Project Cloning** creates a standalone SDK project based on an example application available within SDK release (IAR/Keil tools)

# MCUXpresso Development flow



Examples,  
Demo Apps



# MCUXpresso SDK Builder (<http://mcuxpresso.nxp.com>)



- MCUXpresso Dashboard
- MCUXpresso SDK Builder
- Online MCUXpresso Config Tools

The image displays two screenshots of the NXP MCUXpresso web interface. The top screenshot shows the 'SDK Builder' tool, which allows users to generate a downloadable SDK archive. It includes sections for 'Developer Environment Settings' (Host OS: Windows, Toolchain: IDE), 'Select Optional Middleware' (Add software component), and 'Hardware Details' (Board: FRDM-K64F, Device: MK64F12, Core Type / Max Freq: Cortex-M4F / 120MHz, Memory Size: 1024 KB Flash, 256 KB RAM). The bottom screenshot shows the 'MCUXpresso Dashboard' with a sidebar navigation menu and a main content area titled 'My Recent SDKs'. It lists three configurations:

Configuration	Actions
<b>LPCXpresso54608 with MCUXpresso IDE</b> (NEW) Board: LPCXpresso54608, SDK Version: KSDK 2.3.0, OS: Windows Components: USB stack, FatFS, IWP	Download, Refresh, Configure, Delete
<b>FRDM-KW41Z with MCUXpresso IDE</b> Board: FRDM-KW41Z, SDK Version: KSDK 2.3.0, OS: Windows Components: FreeRTOS, FatFS	Download, Refresh, Configure, Delete
<b>FRDM-K64F with MCUXpresso IDE</b> Board: FRDM-K64F, SDK Version: KSDK 2.3.0, OS: Windows Components: USB stack, FatFS, IWP	Download, Refresh, Configure, Delete

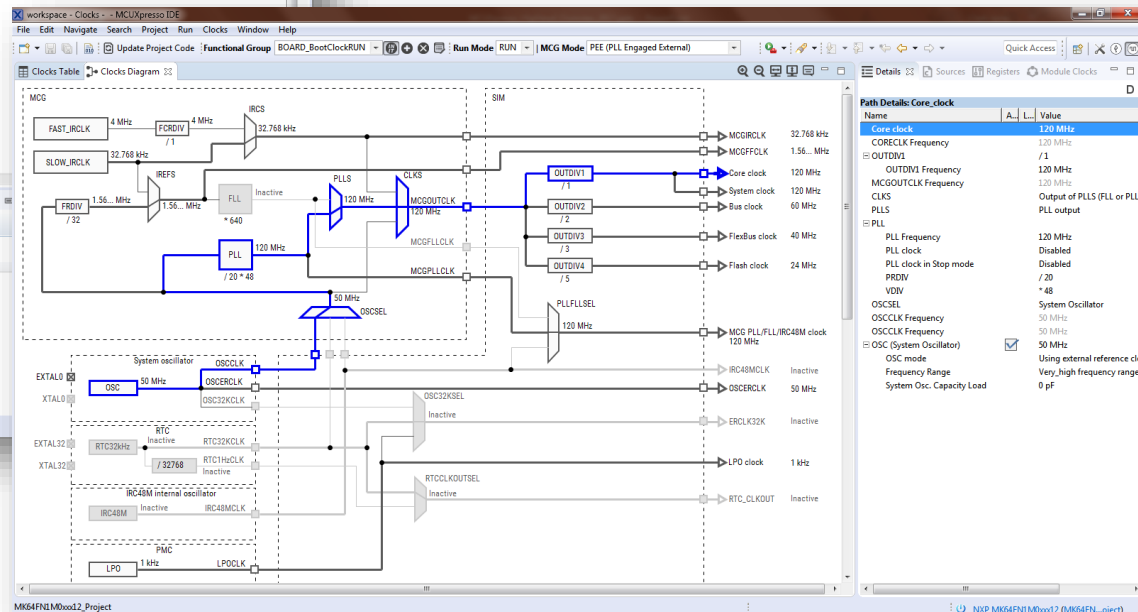
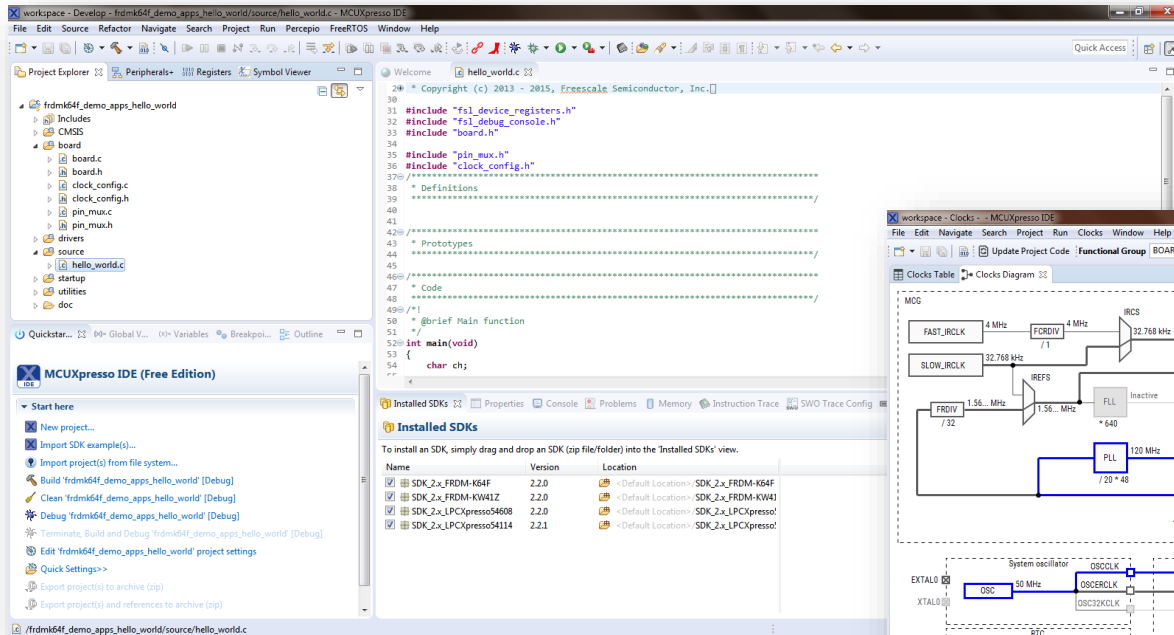
The bottom right of the dashboard shows a 'Clocks Diagram' tool with a 'Clocks Table' and 'Clocks Diagram' views. The 'Clocks Table' displays the following details for the 'Core\_clock':

Name	Value
Core clock	120 MHz
CORECLK Frequency	120 MHz
OUTDIV1	/ 1
OUTDIV1 Frequency	120 MHz
MCGOUTCLK Frequency	120 MHz
CLKS	Output of PLL
PLLS	PLL output
PLL	PLL
PLL Frequency	120 MHz
PLL lock	Disabled
PLL lock in Stop mode	Disabled
FRDIV	/ 15
VDIV	* 36
OSCSSEL	System Osc
OSCCLK Frequency	50 MHz
OSCCLK Frequency	50 MHz



# MCUXpresso IDE

- MCUXpresso SDK Installation
- Development Perspective
- Integrated MCUXpresso Config Tools



# MCUXpresso Config Tools



- Pin and Clock Initialization
  - Peripheral support for LPC8xx in future release
- Code Generation
- Conflict Resolution



The screenshot displays the 'Peripherals' and 'Routed Pins' windows. The 'Peripherals' window shows a list of pins and their associated peripheral labels. The 'Routed Pins' window shows a table of routed pins with columns for Peripheral, Signal, Route to, Label, Identifier, Direction, Slew rate, and Open.

Peripheral	Signal	Route to	Label	Identifier	Direction	Slew rate	Open
UART0	RX	U7[0]UART0_RX	DEBUG_UART_RX	signal	Fast		Disa
UART0	TX	U7[0]UART0_TX	DEBUG_UART_TX	signal	Fast		Disa

The screenshot displays the 'Clocks Diagram' and 'clock\_config.h' files. The clock diagram shows the internal clock structure with various PLLs and dividers. The 'clock\_config.h' file contains the configuration code for the clock system.

```

/* TEXT BELOW IS USED AS SETTING FOR TOOLS *****
1) GlobalInfo
product: P1064N10xxxx12
package_id: PK64FN10MXX12
mcu_data: ksdk2_0
processor_version: 2.0.0
board: F064N10
pin_labels:
- (pin_num: '33', pin_signal: PTE26/ENET_1588_CLKIN/UART4_CTS_B/RTC_CLKOUT)
- (pin_num: '68', pin_signal: PTE22/SP12_S0UT/FR_AD03/COMP2_OUT, label: 'O3')
- (pin_num: '67', pin_signal: PTE21/SP12_SCK/FR_AD03/COMP1_OUT, label: 'O2')
- BE CAREFUL MODIFYING THIS COMMENT - IT IS YAML SETTINGS FOR TOOLS *****
*/

#include "fsl_common.h"
#include "fsl_port.h"
#include "pin_mux.h"

/*FUNCTION Name : BOARD_InitBootPins
 * Description : Calls initialization functions.
 *
 * */
void BOARD_InitBootPins(void) {
    BOARD_InitPins();
}

#define PIN16_IDX 16u /*!c Pin number for pin 16 i
#define PIN17_IDX 17u /*!c Pin number for pin 17 i
#define SPT15_UART0_TX 0x00u /*!c UART 0 transmit data so
*/

/* TEXT BELOW IS USED AS SETTING FOR TOOLS *****
- options: (callFromInitBoot: 'true', coreID: core0, enableClock: 'true')
- pin_list:
- (pin_num: '62', peripheral: UART0, signal: RX, pin_signal: PTE16/SP11_
- (pin_num: '63', peripheral: UART0, signal: TX, pin_signal: PTE17/SP11_
- BE CAREFUL MODIFYING THIS COMMENT - IT IS YAML SETTINGS FOR TOOLS *****
*/

/*FUNCTION Name : BOARD_InitPins
 * Description : Configures pin routing and optionally pin electrical fe
 */
    
```





# LPCXpresso804 Development board (OM40001)

Host debug and power  
USB connector

CMSIS-DAP Debug probe  
Incl. VCOM port

User Button

ISP Button

User LEDs

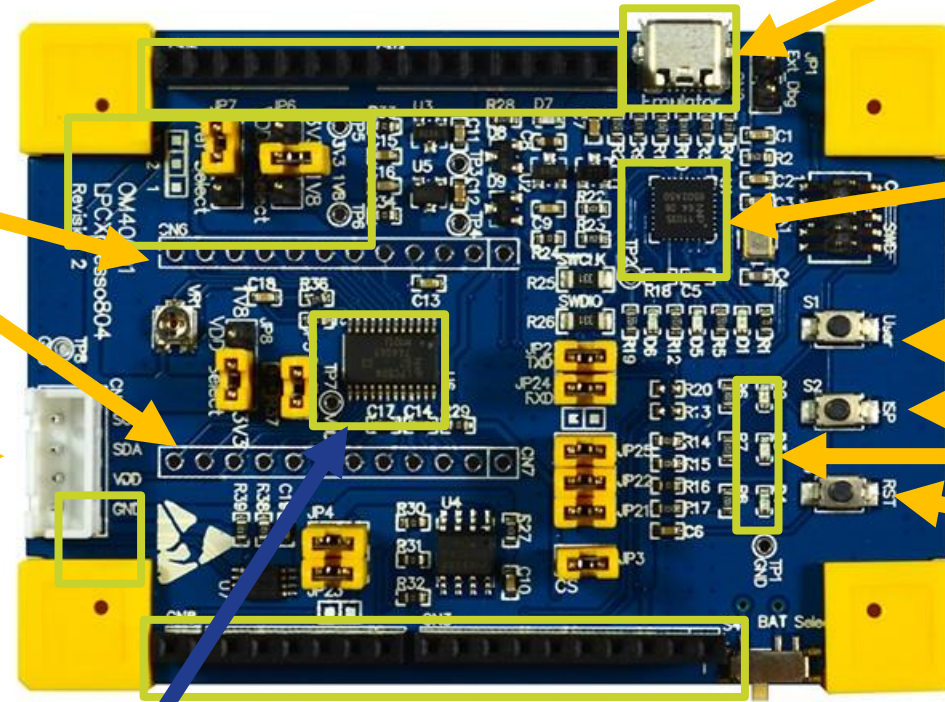
Reset Button

Access header for  
LPC804 pins

Grove I2C  
Connector

LPC804 Cortex M0+ MCU  
SPI, I2C, USART, PLU

Arduino Expansion Header





# LIVE DEMO



# Other software for LPC8xx Devices

## LPCOpen

- Software drivers and libraries for pre-2017 LPC Cortex-M devices
  - LPC81x and LPC82x
- MCU peripheral device drivers
- Common APIs across device families
- <http://www.nxp.com/lpcopen>

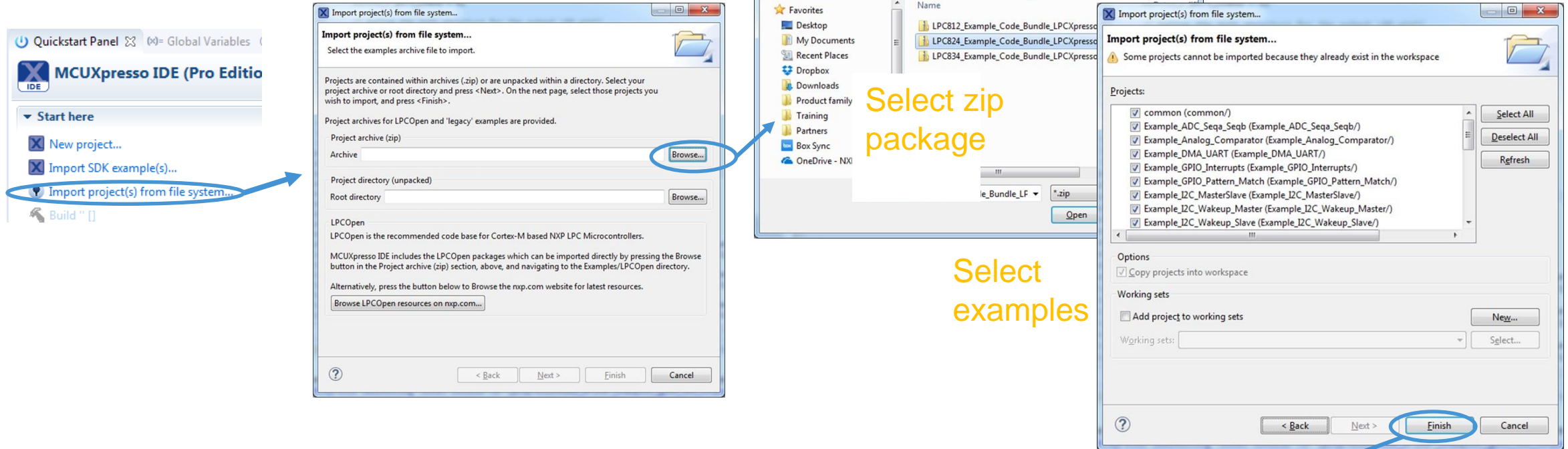
## Code Bundles

- Available for all LPC8xx devices
- Drivers and examples
- Simple, register-level examples
- Ideal for customers transitioning from 8 or 16 bit MCUs
- <https://www.nxp.com/LPC800-Code-Bundles>

*LPCOpen and Code Bundles do not require SDK package installation in MCUXpresso IDE*

# Using Code Bundles / LPCOpen packages

- LPC800 Code Bundles are included in the IDE installation
  - Also available from [nxp.com/lpc800-code-bundles](http://nxp.com/lpc800-code-bundles)
- Simple to install:



Select zip package

Select examples

Click Finish to import

Edit/Build/Debug, as with SDK

# PROGRAMMABLE LOGIC UNIT (LPC804)



# PLU Design Tool

Mapping of inputs, outputs and flops

LUT configuration

The screenshot displays the PLU Config Tool interface for an LPC80x device. It features several key components:

- Mapping Table:** A table on the left showing the mapping of schematic names to hardware primitives.
- Schematic Capture Area:** A central workspace showing a logic diagram with two inputs (in1, in2) connected to a LUT3 primitive, which is then connected to an output (outa).
- LUT Configuration Table:** A table on the right showing the configuration for the selected LUT3 primitive.
- Generated Source:** A window at the bottom right showing the C code generated from the design.
- Resource Counter:** A bar at the bottom of the schematic area showing the remaining counts for various primitives.

Schematic Name	Mapped
in1	IN0
in2	IN1
lut	LUT3
outa	OUT0

index	E	D	C	B	A	OUT
0x0	x	x	x	0	0	0
0x1	x	x	x	0	1	1
0x2	x	x	x	1	0	1
0x3	x	x	x	1	1	0
0x4	x	x	x	x	x	x
0x5	x	x	x	x	x	x

```
plu_tool.c
/*
 * Copyright 2017 NXP
 * 6 Inputs 26 LUTs 4 flip flops and 8 outputs
 */
/* LUT3 (lut) */
LPC_PLU0->LUT_MUX[3].INP[0] = 0x00000001;
LPC_PLU0->LUT_MUX[3].INP[1] = 0x00000000;
LPC_PLU0->LUT_MUX[3].INP[2] = 0x0000003F;
LPC_PLU0->LUT_MUX[3].INP[3] = 0x0000003F;
LPC_PLU0->LUT_MUX[3].INP[4] = 0x0000003F;
LPC_PLU0->LUT_TRUTH[3] = 0x66666666; /* lut

LPC_PLU0->OUTPUT_MUX[0] = 0x00000003; /* LI
```

IN Remaining: 4    OUT Remaining: 7    FF Remaining: 4    LUT Remaining: 25

Schematic capture area

LUT, I/O and Flip-flop primitives

Generated source



**MCUXpresso**  
Software and Tools

COMMON TOOLKIT  
FOR THOUSANDS  
OF KINETIS® & LPC  
MICROCONTROLLERS



[www.nxp.com/mcuxpresso](http://www.nxp.com/mcuxpresso)

# MCUXpresso Software and Tools

## Additional Resources

### Web pages

- LPC80x Webpage: <http://www.nxp.com/lpc80x>
- MCUXpresso Software and Tools – [www.nxp.com/mcuxpresso](http://www.nxp.com/mcuxpresso)
- MCUXpresso SDK – [www.nxp.com/mcuxpresso/sdk](http://www.nxp.com/mcuxpresso/sdk)
- MCUXpresso IDE – [www.nxp.com/mcuxpresso/ide](http://www.nxp.com/mcuxpresso/ide)
- MCUXpresso Config Tools – [www.nxp.com/mcuxpresso/config](http://www.nxp.com/mcuxpresso/config)

**Supported Devices:** [Supported Devices Table \(Community Doc\)](#)

### Communities

- LPC microcontrollers –  
<https://community.nxp.com/community/lpc>

MCUXpresso Software and Tools -  
<https://community.nxp.com/community/mcuxpresso>



**SECURE CONNECTIONS  
FOR A SMARTER WORLD**