Using MCUXpresso Config Tools with Custom Hardware

Generating a Board Support Configuration (.mex)

This section covers using the MCUXpresso Config Tools to create a custom configuration for a new board.

- 1. Create a new LPC845 based project and enable MCUXpresso Config Tools
 - 1.1. Open MCUXpresso IDE, shortcut located on desktop.
 - 1.2. Use the Quickstart Panel to create and "New project..." and select the LPC845 device.
 - Expand "LPC84x" from the list of "SDK MCUs" Targets
 - Select LPC845 from the menu, but <u>do not</u> select a photo of a board under "Available boards".
 - Select Next
 - 1.3. Under "Device Packages" select the LPC845M301JBD48 package.
 - 1.4. Under "driver" check the boxes next to:

clock, common, gpio, power, reset, swm, and usart.

- 1.5. Select Finish
- 1.6. Use the Project Explorer menu bar to launch Config Tools and Open Pins 🛛 🗷 🝷
- 1.7. Ensure the correct project is from the drop-down menu in the tool bar



- 2. Use MCUXpresso Config Tools to mux pin signals
 - 2.1. Ensure the Pins Tool is open using the perspective icons located in the upper right:
 - 2.2. Add GPIOs Muxing for LEDs
 - To enable the GREEN LED as GPIO, click on the pin name that matches the schematic (PIO1 0) under the "GPIO" column in the Pins window.
 - Enter GREEN as the identifier for the pin.
 - In the Routed Pins window, specify the Direction as Output (*optional*).
 - Repeat for BLUE (PIO1_1) and RED (PIO1_2) LEDs.







2.3. Add PINTs Muxing for Pushbuttons

- To enable to the USER button, click on the pin name that matches the schematic (PIO0_4) under the "PINT" column in the Pins window.
- Select (PINT, 0) in the resulting selection dialog
- Enter USER as the identifier for the pin.
- Repeat for ISP (PIO0_12) -> (PINT, 1) button.



AMF-SOL-T3532 Version 2.0 May 2, 2019 For SDK2.5.0 and MCUXpresso 10.3.1

- 2.4. Add ADC for onboard potentiometer
 - To enable ADC, click on the pin name that matches the schematic (ADC_0) under the "ADC" column in the Pins window.
 - Enter ADC as the identifier for the pin.
- 2.5. Optional pin muxing steps:
 - Configure Pin 42 as ACMP_I5 (Label: CAPY)
 - Configure Pin 9 as CAPT_X0 (Label: CAPX)
- 3. Use MCUXpresso Config Tools to change the clock configuration
 - 3.1. Switch to the Clock Tools using the perspective icon in the upper right:
 - 3.2. Select the "Clock Diagram" view and select the "System clock" to highlight the clock tree path



*note: you will need to scroll down to see the origin of the "fro" input signal

- 3.3. With the "System clock" highlighted, review the "Details" panel for specific clock configurations related to this clock path.
- 3.4. Change the **fro** setting to "**direct from FRO oscillator**", this provides a 30MHz System Clock, but introduces an error with the PLL, which has a max frequency of 25MHz (see error message and tooltip).
- 3.5. Resolve the PLL error by changing the SYSPLLCLKSEL input source from "fro" to "fro_div"
- 3.6. Optionally change the **PLL Multiplier** to ***2** to set the SYSPLL clock output to the 30MHz maximum.
- 4. Update MCUXpresso IDE project with new Config Tools setting
 - 4.1. Use the **Update Project** to save the config settings (.mex) and generate the pin and clock source files.
 - 4.2. Review the generated source files.
- 5. Save the .mex file to an easy to access location for further use in the workshop.

AMF-SOL-T3532 Version 2.0 May 2, 2019 For SDK2.5.0 and MCUXpresso 10.3.1



🗙 🏢 🔟

5.1. Find the .mex file in the Project Explorer and save a copy by copying it to the Desktop (select the file and press ctrl + C, then paste on Desktop). X LPC845_Project.mex

Apply Board Support Configuration to a New Project

This section covers applying a board support configuration to a blank new project.

- 6. Create another "New Project..." for the LPC845 device
 - 6.1. Repeat steps 1.2 1.6 to create an additional project, but provide a unique suffix to the project name
 - 6.2. Use the Project Explorer menu bar to enable the Config Tools and Open Pins
- 7. Apply the previously create configuration to this new project
 - 7.1. Import .mex file created in Step 1 to the new project
 - File \rightarrow Import.
 - Under MCUXpresso Config Tools select "Import Configuration (*.mex)" and click Next.
 - Select the .mex file created previously as the Existing configuration, and the name of the current project as the Target Project. Click **Finish**.

🔄 Update Project

- 7.2. Observe the pins that are activated and notice that the configuration is the same as in the previous case. Open the Clock Tool to see that the System clock has also been updated to 30 MHz.
- 7.3. Save and Update Project with the Config Tools settings.
- 7.4. Examine resulting project files.
- 8. Build and Debug application on LPC Breakout Board
 - 8.1. From the Quickstart Panel select "Build"
 - 8.2. From the Quickstart Panel select "Debug" and follow resulting dialog windows to setup the correct debug interface.
 - 8.3. Use the Debug control buttons to "Resume" the application from the initial main() breakpoint.



8.4. "Hello World" should output on the console.

AMF-SOL-T3532 Version 2.0 May 2, 2019 For SDK2.5.0 and MCUXpresso 10.3.1



Installed SDKs □ Properties □ Console ☎ Problems ➡ Debug LPC845_Project_test LinkServer Debug [C/C++ (NXP Semiconductors) MCU Ap [MCUXpresso Semihosting Telnet console for 'LPC845_Project_

Hello World

Apply Board Support Configuration to an Existing Example Project

This section covers applying a customized board support configuration to an existing SDK example project.

- 9. Open / Import an existing MCUXpresso SDK Example project for an NXP development board
 - 9.1. Use the Quickstart Panel to Import SDK Example 🛛 🔣 Import SDK example(s)...
 - Select "LPC84x" from the list of "SDK MCUs" targets
 - Locate and select the picture of the LPCXpresso845max board. Select Next.
 - Under "demo_apps" select the example called "led_blinky".
 - Select Finish.
- 10. Convert project to correct device package (impacts to IDE, SDK, and Config Tools)

10.1. Convert IDE project to correct Package (Note, the IDE doesn't really care, this just helps avoid other conflicts)

- Expand **lpcxpresso845max_led_blinky** -> **Project Settings** -> **MCU**, note chip and package values
- Right-click MCU, select Change package from the menu, then select LPC845M301JBD48
- 10.2. Enable Pins Tools
 - Use the Project Explorer menu bar to launch 🛛 🛛 🗸 Config Tools and Open Pins.
 - Use the Switch Package button toolbar in the top right corner of the Package window to select the "LQFP 48 package".
 - Optionally, you can right-click the error message in the Config Tools problem window to switch the package type directly.

- 11. Apply the previously create configuration to this new project
 - 11.1. Import .mex file created in Step 1 to the new project
 - File \rightarrow Import.
 - Under MCUXpresso Config Tools select "Import Configuration (*.mex)" and click **Next**.
 - Select the .mex file created previously as the Existing configuration, and the name of the current project as the Target Project. Click **Finish**.

🖸 Update Project

- 11.2. Observe the pins that are activated and notice that the configuration is the same as in the previous case. Open the Clock Tool to see that the System clock has also been updated to 30 MHz.
- 11.3. Save and **Update Project** with the Config Tools settings.
- 11.4. Examine resulting project files.
- 12. Edit source code to "connect" new LED signal
 - 12.1. Open the pinmux.h file under the board folder in the example project.
 - 12.2. Scroll down to find the definitions for each of the LEDs enabled earlier. Find:

#define BOARD_INITPINS_GREEN_PORT 1U

#define BOARD_INITPINS_GREEN_PIN 1U

12.3. In the led_blinky.c file, find the definitions:

#define BOARD_LED_PORT 0U

#define BOARD_LED_PIN 12U

- 12.4. Copy the variable names **BOARD_INITPINS_GREEN_PORT** and **BOARD_INITPINS_GREEN_PIN** from the pinmux.h file to replace the hard coded values of the definitions in led_blinky.c (0U and 12U respectively).
- 12.5. Build, Debug, and Run the application to see the green LED flashing.
- 12.6. End the debug session using the Terminate (red stop) button.

	1	2	3	4	5	6	7	8
		PI00 16 CNI 40 PI00 17 50 PI0 VCC 40 PI00 17 50 PI1 P00 30 PI00 19 50 PI2 P00 42 PI00 20 50 PI3 P02 PI3 P03 42 PI00 22 50 PI3 P14 P03 42	VDD 9 PR00 00 8 PR00 01 7 PR00 02 6 PR00 03 5 PR00 04 4 PR00 05		P <u>C845_OFP48</u>		VDD R5 10K PIO0 02 PIO0 03 8 10 PIO0 05 10 PIO0 05 10 PIO0 05 10 PIO0 05 10 PIO0 05 10 PIO0 05 10 PIO0 05 10 PIO0 PIO0 PIO0 PIO0 PIO0 PIO0 PIO0 PIO	Α
3		PIO0 23 PIO PIO PIO PIO PIO0 65 PIO7 RST PO7 <th>3 PIC0 06 2 PIC0 07 1 PIC1 10 0 PIC1 1 9 PIC1 2 8 PIC0 0.15 7 PIC0 1.3 5 PIC0 0.13 5 PIC0 0.13 7 PIC0 0.11</th> <th>PIO0_0/AC PIO0_1/ACMP I2 SWDIA' PIO0_4/ADC_11/TRT PIO0_6/ADC_1 PIO0_6/ADC_1 PIO0_PIO0_1 PIO0_PIO0_1 PIO0_14/ACM PIO0_17/ADC_9 PIO0_1/PIO0_PIO0_PIO0_PIO0_1 PIO0_17/ADC_9 PIO0_1PIO0_PIO0_PIO0_PIO0_PIO0_PIO0_PIO0_PIO0_</th> <th>MP_1I/TDO 36 P100 00 CLKINTDI 10 P100 01 P100 21X 8 P100 02 P100 21X 8 P100 03 P100 21X 8 P100 03 P100 21X 8 P100 04 NWAKEUP 5 P100 06 SET/P100/FF 34 P100 06 07/ADC 03 P100 07 D20 7ADC 00 07 D20 7ADC 00 07 D20 7ADC 01 100 D1200 SCAL 12 P100 D1200 SCAL 2 P100 12 P100 12 2 P100 14 P100 12 2 P100 14 P100 12 2 100 14 P100 15 P100 16 10</th> <th></th> <th></th> <th>VDDA RVI 10K E RII 100</th>	3 PIC0 06 2 PIC0 07 1 PIC1 10 0 PIC1 1 9 PIC1 2 8 PIC0 0.15 7 PIC0 1.3 5 PIC0 0.13 5 PIC0 0.13 7 PIC0 0.11	PIO0_0/AC PIO0_1/ACMP I2 SWDIA' PIO0_4/ADC_11/TRT PIO0_6/ADC_1 PIO0_6/ADC_1 PIO0_PIO0_1 PIO0_PIO0_1 PIO0_14/ACM PIO0_17/ADC_9 PIO0_1/PIO0_PIO0_PIO0_PIO0_1 PIO0_17/ADC_9 PIO0_1PIO0_PIO0_PIO0_PIO0_PIO0_PIO0_PIO0_PIO0_	MP_1I/TDO 36 P100 00 CLKINTDI 10 P100 01 P100 21X 8 P100 02 P100 21X 8 P100 03 P100 21X 8 P100 03 P100 21X 8 P100 04 NWAKEUP 5 P100 06 SET/P100/FF 34 P100 06 07/ADC 03 P100 07 D20 7ADC 00 07 D20 7ADC 00 07 D20 7ADC 01 100 D1200 SCAL 12 P100 D1200 SCAL 2 P100 12 P100 12 2 P100 14 P100 12 2 P100 14 P100 12 2 100 14 P100 15 P100 16 10			VDDA RVI 10K E RII 100
2	CN2 MicroUSB V CC - et D + et D + et D + et B - et Shell - t Shell - t	VDD T1 9012 VBUS R12 R12 R12 R13 R13 R13 VBUS R12 VBUS R13 VBUS VBUS R13 VBUS VBUS VBUS VBUS VBUS VBUS VBUS VBUS	Emulator Emulator Sch USB_EN SWDIO SWDIO SWCLK PPI SWCLK PPI SWCLK PPI SWCLK PPI TXD PPI TXD PPI PPI USB_DM USB_DP USB_VBUS	PIO PIO PIO PIO PIO PIO 23/ADC PIO 20 05 PIO 20 05 PIO 20 25 PIO PIO PIO PIO PIO PIO PIO PIO PIO PIO	¹ 20 ADC - 6 ¹ 44 Pi00 21 ² 21 ADC - 5 ¹ 43 Pi00 22 ² 21 ADC - 4 ¹ 39 Pi00 22 ² 21 ADC - 4 ¹ 39 Pi00 22 ¹ 21 ADC - 4 ¹ 39 Pi00 23 ¹ 19 Pi00 24 ¹ 19 Pi00 25 ¹ 19 Pi00 26 ¹ 19 Pi00 26 ¹ 19 Pi00 26 ¹ 17 Pi00 27 ¹ 7 Pi00 28 ¹ 10 Pi00 28 ¹ 10 Pi00 29 ¹ 10 Pi00 29 ¹ 12 Pi01 2 ¹ 2CAPT X1 ¹ 12 Pi01 2 ¹ 2CAPT X3 ¹ 12 Pi01 3 ¹ 23 Pi01 4 ¹ 27 CAPT X3 ¹ 28 Pi01 6 ¹ CAPT X1 ¹ 28 Pi01 6 ¹ CAPT X1 ¹ 12 Pi01 5 ¹ CAPT X1 ¹ 12 Pi01 5 ¹ CAPT X1 ¹ 12 Pi01 6 ¹ CAPT X1 ¹ 14 Pi01 7 ¹ CAPT X1 ¹ 12 Pi01 8 ¹ CAPT V1 ¹ Pi01 ¹ Pi01	CAPY R CAPX 0[N/A] PIO0_30 0[N/A] PIO0_31 		
		VBUS 3 VIII VOUL *C8 TOUF/16V			$\begin{array}{c} VDDA \\ VREFN \\ VSSA \\ VSSA \\ VDD \\ VSS \\ VDD \\ VSS \\ \hline \\ \hline \\ \hline \\ VSS \\ \hline \\ \hline \\ \hline \\ VSS \\ \hline \\$		CAPY HIGH 19 19 19 10 19 10 10 10 10 10 10 10 10 10 10	
	1	2	3	4	5	6	7	8

