NXP Semiconductors
User's guide

Document Number: PEXMC12XSF-MC12XS6UG
Rev. 1.0, 5/2016

Gen4eXtremeSwitch Processor Expert component

Table of Contents

1 Overview

This documentation describes how to install and use Processor Expert in conjunction with the Gen4eXtremeSwitch component.

The Gen4eXtremeSwitch component supports the following analog parts:

- NXP MC12XSF: External industrial lighting 5 MHz SPI-controlled eXtreme Switch (www.nxp.com/MC12XSF)
- NXP MC12XS6: External automotive lighting 5 MHz SPI-controlled eXtreme Switch (www.nxp.com/MC12XS6)

NXP’s FRDM-17XSF5-EVB, FRDM-17XSF4-EVB, FRDM-08XSF4-EVB and FRDM-40XSF5-EVB boards are evaluation platforms based on the MC12XSF and the MC12XS6 devices. See the related user guides and datasheets for detailed information.

2 Gen4eXtremeSwitch compatibility

2.1 Peripheral requirements

Peripherals and resource requirements critical to the MCU's ability to handle a given part are as follows:

SPI communication mode

- SPI module – required for communication (SI, SO, SCLK, CSB pins)
- TPM/FTM timer (PWM, single channel) required for clock generation
- GPIOs – required for the device reset (RSTB) pin and direct input pins (IN1 through IN4)
- ADC and GPIO – required for current sensing (CSNS) and synchronization (CSNS SYNC)

2.2 Supported devices

The 12XSF and 12XS6 families represent the fourth generation of NXP's eXtreme switch products.This product line supports automotive and industrial lighting applications ranging from incandescent bulbs to LEDs.

Common features of both the 12XSF and the 12XS6 family are:

- Operating voltage range of 7.0 V to 18 V, with sleep current < 5.0 μA
- 5.0 MHz 16-bit SPI control of overcurrent profiles, channel control including 8-bit PWM duty-cycles, output ON and OFF open load detections, thermal shutdown and pre-warning, and fault reporting
- Output current monitoring with programmable synchronization signal and supply voltage feedback
- Programmable overcurrent trip levels
- Enhanced output current sense with programmable synchronization signal and battery voltage feedback
- Watchdog and limp home mode
- External smart power switch control (Output 6)
- –16 V reverse polarity and ground disconnect protection
- Compatible PCB foot print and SPI software driver among the family

The 12XSF and 12XS6 family products differ in the number of available outputs and their $R_{DS(on)}$ resistance (see Table 1).

Table 1. Device features

Product Number	No. of Outputs	OUT1	OUT2	OUT3	OUT4	OUT5	OUT6
MC07XSF517 MC07XS6517	5 + 1	17 mΩ	17 mΩ	7 mΩ	7 mΩ	7 mΩ	Yes
MC08XSF421 MC08XS6421	4 + 1	21 mΩ	21 mΩ	8 mΩ	8 mΩ	Not connected	Yes
MC10XS6200	2 + 1	Not connected	Not connected	10 mΩ	10 mΩ	Not connected	Yes
MC10XS6225	2 + 1	25 mΩ	Not connected	10 mΩ	Not connected	Not connected	Yes
MC10XS6325	3 + 1	25 mΩ	Not connected	10 mΩ	10 mΩ	Not connected	Yes
MC17XSF400 MC17XS6400	4 + 1	17 mΩ	17 mΩ	17 mΩ	17 mΩ	Not connected	Yes
MC17XSF500 MC17XS6500	5 + 1	17 mΩ	17 mΩ	17 mΩ	17 mΩ	17 mΩ	Yes
MC25XSF300	3 + 1	25 mΩ	25 mΩ	25 mΩ	Not connected	Not connected	Yes
MC40XSF500 MC40XS6500	5 + 1	40 mΩ	40 mΩ	40 mΩ	40 mΩ	40 mΩ	Yes

2.3 Supported MCUs

The Gen4eXtremeSwitch component supports the MCUs listed in Table 2. The listed MCUs are a subset of MCUs supported by Processor Expert for Kinetis using the logic device driver (LDD) layer.

Compatibility with other MCUs and MCU boards depends on their peripherals and pin connections.

Table 2. Supported MCUs

Supported MCUs	KDS examples	Limitations
FRDM-KL25Z	Yes	Yes
FRDM-K64F	Yes	Supports only one device (1 CS pin). No support for direct inputs.
FRDM-KV10Z	Yes	Supports only one device (1 CS pin). No support for monitoring (CSNS and CSNS_SYNC not connected).

2.4 Possible Freedom board settings

The FRDM-17XSF5-EVB evaluation board features the MC17XSF500, a member of NXP's 12XSF high-side switch products family. The board incorporates the following:

- up to five outputs with freewheeling diode protection and LED indicators
- output for external smart power switch control output
- on-board voltage regulator and level shifter (from 3.3 V to 5.0 V logic) and switch to enable operation of up to four boards in parallel with separate chip-select pins (parallel SPI operation)

Figure 1 shows a typical FRDM-17XSF5-EVB configuration. The eXtreme switch expansion board is mounted on top of a FRDM-KL25Z board. The green and black connectors on the left are for power supplies. The green connector on the right connects to the load.

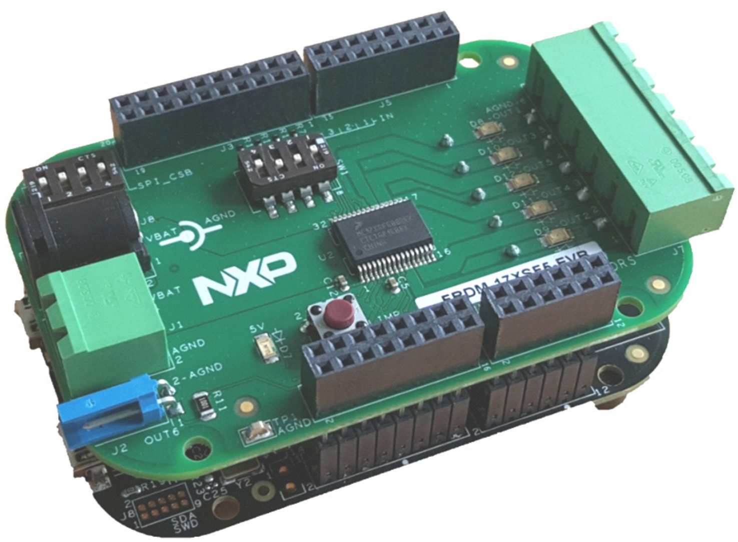


Figure 1. Typical FRDM-17XSF5-EVB configuration

3 The Gen4eXtremeSwitch component

The Gen4eXtremeSwitch component is located under the Components folder in the Processor Expert active projects window (see Figure 2). The functionality of this component depends on the component property settings assigned through Processor Expert.

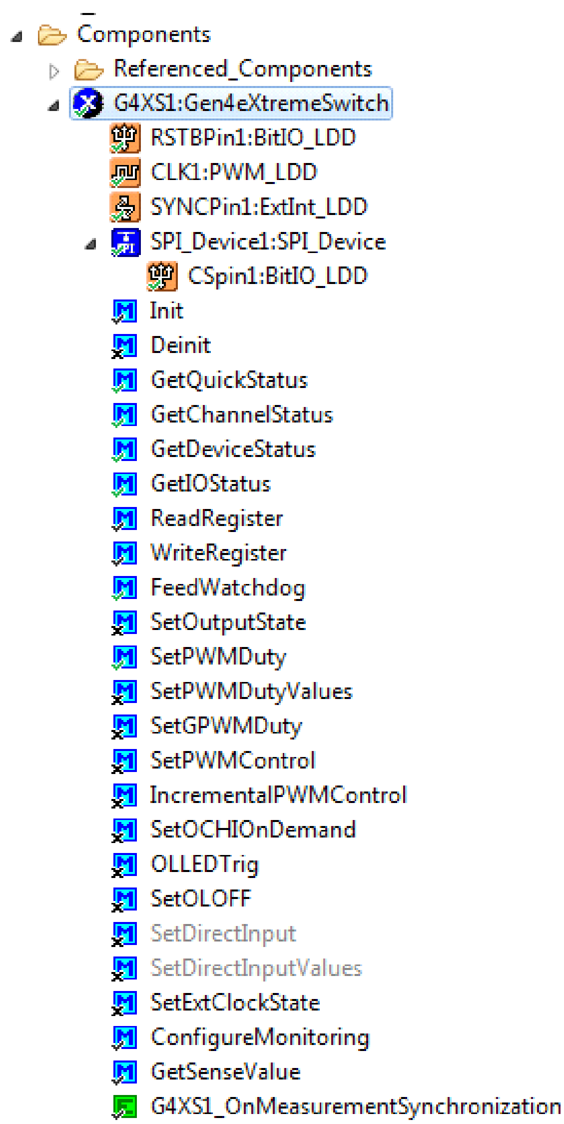


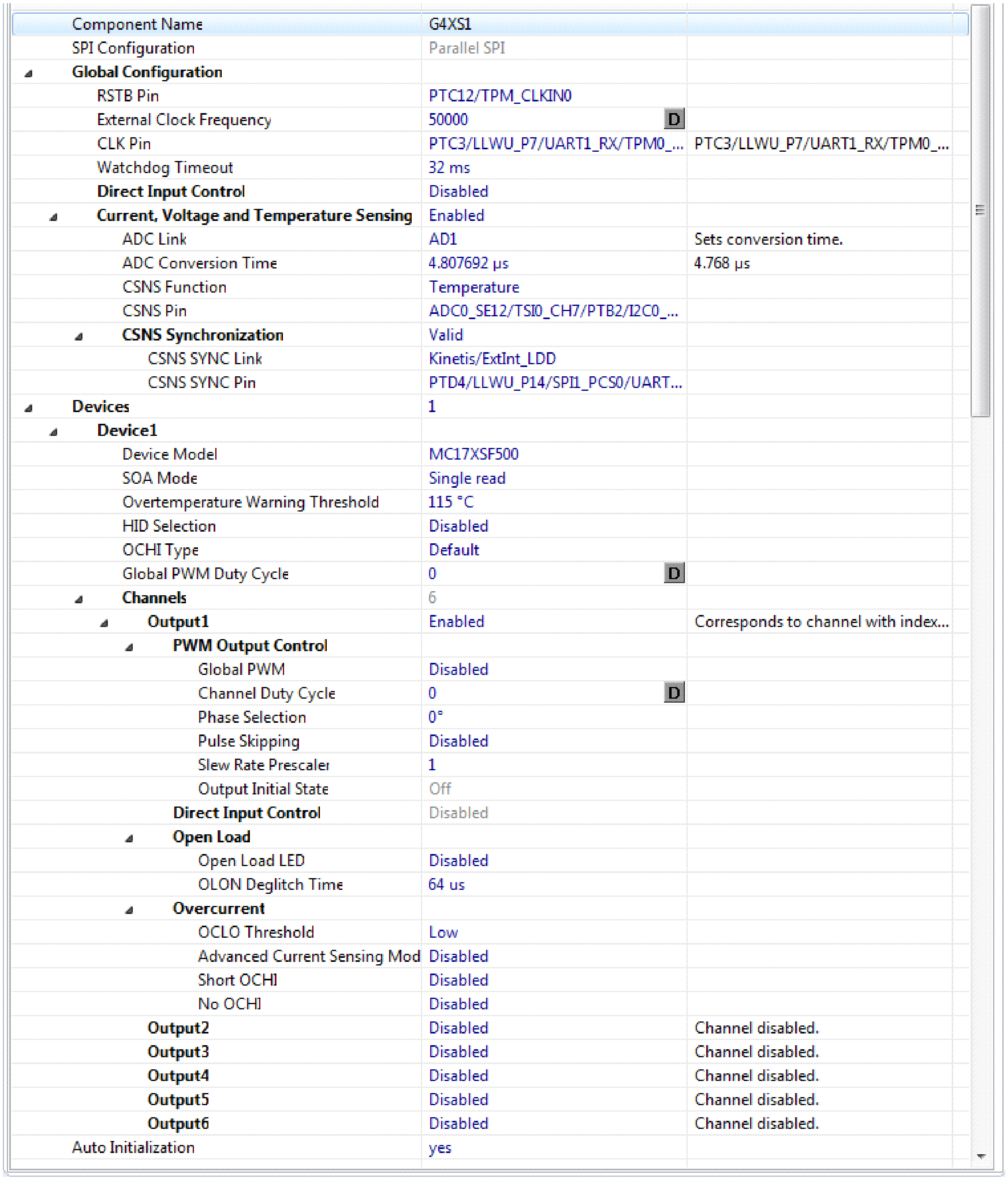
Figure 2. Gen4eXtremeSwitch Processor Expert component

The Gen4eXtremeSwitch component includes a Help function, which can be accessed by right-clicking on **G4XS1:Gen4eXtremeSwitch** in the component tree. The **Help on Component** window provides information on all the component's properties and methods. Access the **Typical Usage** section to view examples showing how to work with API methods.

3.1 Component settings

Selecting a component in the component tree provides access to the component's properties in the **Component Inspector**. These properties determine the component's general settings and its behavior after initialization. Application code can later change some of these properties using the provided API.

See Figure 3 and the associated text for a definition of the available properties.



Component Name	G4XS1	
SPI Configuration	Parallel SPI	
Global Configuration		
RSTB Pin	PTC12/TPM_CLKIN0	
External Clock Frequency	50000	
CLK Pin	PTC3/LLWU_P7/UART1_RX/TPM0_...	PTC3/LLWU_P7/UART1_RX/TPM0_...
Watchdog Timeout	32 ms	
Direct Input Control	Disabled	
Current, Voltage and Temperature Sensing	Enabled	
ADC Link	AD1	Sets conversion time.
ADC Conversion Time	4.807692 µs	4.768 µs
CSNS Function	Temperature	
CSNS Pin	ADC0_SE12/TSI0_CH7/PTB2/I2C0_...	
CSNS Synchronization	Valid	
CSNS SYNC Link	Kinetis/ExtInt_LDD	
CSNS SYNC Pin	PTD4/LLWU_P14/SPI1_PCS0/UART...	
Devices	1	
Device1		
Device Model	MC17XSF500	
SOA Mode	Single read	
Overtemperature Warning Threshold	115 °C	
HID Selection	Disabled	
OCHI Type	Default	
Global PWM Duty Cycle	0	
Channels	6	
Output1	Enabled	Corresponds to channel with index...
PWM Output Control		
Global PWM	Disabled	
Channel Duty Cycle	0	
Phase Selection	0°	
Pulse Skipping	Disabled	
Slew Rate Prescaler	1	
Output Initial State	Off	
Direct Input Control	Disabled	
Open Load		
Open Load LED	Disabled	
OLON Deglitch Time	64 us	
Overcurrent		
OCLO Threshold	Low	
Advanced Current Sensing Mod	Disabled	
Short OCHI	Disabled	
No OCHI	Disabled	
Output2	Disabled	Channel disabled.
Output3	Disabled	Channel disabled.
Output4	Disabled	Channel disabled.
Output5	Disabled	Channel disabled.
Output6	Disabled	Channel disabled.
Auto Initialization	yes	

Figure 3. Gen4eXtremeSwitch component properties

SPI Configuration – Configures SPI communication with devices. Two configurations are supported:

- **Parallel SPI** – Specifies the SPI mode as Parallel SPI (common MISO, MOSI, SCLK, separate CS pins) with common RST, CLK, CSNS and CSNS SYNC pins. For parallel SPI applications, each with its own set of SPI pins, additional instances of this component must be included.
- **Daisy Chain SPI** – Specifies the SPI mode as daisy chain SPI. Devices connected in daisy chain (the MISO of the first device is connected to the MOSI of the second device, and so forth, common CS pin) with common RST, CLK, CSNS and CSNS SYNC pins. Daisy chain SPI with these pins separated is not supported. This configuration does not support the FRDM-17XSF5-EVB.

Global Configuration – Configures shared settings for all channels on all devices

- **RSTB Pin** – Selects the pin that initializes the device configuration and fault registers (when high). This pin also places the device in a low-current Sleep mode (low).
- **External Clock Frequency** – Selects the PWM clock in Hz. Admissible frequency range is 25.6 kHz to 102.4 kHz. The frequency entered is divided by 256 when the device is in normal operating mode.
- **CLK Pin** – Selects the reference PWM clock, which is divided by 256 when the device is in normal operating mode
- **Watchdog Timeout** – Specifies the timeout period used to detect a loss of SPI communication. An SPI communication fault is detected if the WD bit is not toggled with each SPI message or if the WD timeout is reached. If a SPI communication error occurs, the device is switched into fail mode. Shorter (32 ms) or longer (128 ms) WD timeouts can be used depending on the number of connected devices. For more devices, use longer WD timeouts.
- **Direct Input Control** – Contains pins used as direct input pins IN1 to IN4. Direct input pins are used to directly control the corresponding channel in fail mode. During normal mode, SPI programming can be used to enable the input control pins to control the outputs.
- **Current, Voltage and Temperature Sensing** – Provides settings for feedback measurement. The analog feedback circuit provides load and device diagnostics during normal mode. The feedback monitor provides a current proportional to the current of the selected output, a voltage proportional to the battery supply voltage or the output voltage of the temperature sensor that monitors the average control die temperature. Settings include AD conversion time, feedback monitor functions and sensing synchronization.

Devices – Properties related to the number of devices connected to the MCU. Each device has its own set of properties.

- **Device Model** – Model of the eXtreme Switch device. The number of displayed outputs depends on this property.
- **SOA Mode** – Specifies the reading mode of the device
  - **Single Read** – Specifies that the programmed SO address is used for a single read command. After the reading, the SO address returns to quick status register.
  - **All Reads** – Specifies that the programmed SO address is used for the next and all further read commands until a new SO address is programmed.
- **Overtemperature Warning Threshold** – Specifies the maximum safe operating temperature in Celsius for the power transistors. A temperature sensor is located on each power transistor to protect the transistor and to provide SPI status monitoring. When the overtemperature warning threshold is exceeded, the outputs remain in their current state but an overtemperature warning is reported. The overtemperature threshold can be set low (115 °C) or high (135 °C).
- **HID Selection** – Enables/disables smart high-intensity discharge (HID). The smart overcurrent window control feature turns on an HID ballast even when the power-on reset time is long. Smart HID can be enabled/disabled for all channels, channel 3 only, or channels 3 and 4.
- **OCHI Type** – Selects the OCHI type. To minimize the electro-thermal stress inside the device when a short-circuit occurs, the OCHIx levels can be either dynamically adjusted with respect to the control die temperature or evaluated during the OFF-to-ON output transition, depending on the output voltage. For output protection, the OCHI level can be set to depend on the control die temperature, the output voltage or both temperature and voltage.
- **Global PWM Duty Cycle** – In addition to the individual PWM register, each channel can be assigned independently to a global PWM register. When a channel is assigned to global PWM, the global PWM duty cycle is applied but the switching phase, the prescaler and the pulse skipping conform to the corresponding output channel setting.

Depending on the selected device model, all available outputs are displayed and can be enabled/disabled individually. Each output has its own set of properties. Not all features are available for all output channels. The properties are grouped as follows:

- **PWM Output Control** – Properties influencing the PWM output signal
  - **Global PWM** – In addition to the individual PWM register, each channel can be assigned independently to a global PWM register. When a channel is assigned to global PWM, the global PWM duty cycle is applied but the switching phase, the prescaler and the pulse skipping conform to the corresponding output channel setting.
  - **Channel Duty Cycle** – Specifies the duty cycle of the PWM when the channel is controlled individually

   - **Phase Selection** – Selects the phase assignment of the output channel
   - **Pulse Skipping** – Enables/disables the pulse skipping feature. Due to the output pulse shaping feature and the resulting switching delay time of the smart switches, the device cannot generate duty cycles close to 0 or respectively 100 percent. When enabled, the pulse skipping feature interpolates the output duty cycle range in normal mode.
   - **Slew Rate prescaler** – Depending on the programming of the prescaler setting register, the switching speeds of the outputs are adjusted to the output frequency range of each channel.
   - **Output Initial State** – Selects the state of the output (on or off) after initialization
- **Direct Input Control** – If enabled, output is assigned to direct input pin control. To use direct input pin control, enable the **Direct Input Control** property in the global configuration part of the component and set up the pins. The channel output initial state has to be set to On in properties or in code by method **SetOutputState**.
- **Open Load** – Properties related to the open load detection features
   - **Open Load LED** – Enables/disables a special low current detection mode that detects small load currents (for example, an LED) in the on state of the switch.
   - **OLON Deglitch Time** – When an open load is detected, the output remains in the on state. To meet the requirements of different load types, the deglitch time of the open load in the on state can be controlled individually for each output. Short deglitch times (64 μs) for bulbs and long deglitch time (2 ms, converter mode) are implemented.
- **Overcurrent** – Properties related to overcurrent protection. This feature guards against an ultra-low resistive short-circuit condition resulting from a smart overcurrent profile and severe short-circuit protection.
   - **OCLO Threshold** – The static overcurrent threshold can be programmed individually for each output in two levels (low, high) to adapt low duty cycle dimming and a variety of loads.
   - **Short OCHI** – The length of the OCHI windows can be shortened by a factor of 2 to accelerate the availability of the CSNS diagnosis and to reduce the potential stress inside the switch during an overload condition.
   - **No OCHI** – To accelerate the availability of the CSNS diagnosis, the switch on process of an output can be done without an OCHI window.
- **Current Sensing** – Properties related to current monitoring features
   - **Advance Current Sensing Mode** – An advanced current sense mode (ACM) diagnoses LED loads in normal mode and improves current sense accuracy for low current loads. In the ACM mode, the offset sign of the current sense amplifier is toggled on every CSNS SYNCB rising edge. The error amplifier offset contributes to the CSNS error. This contribution can be fully eliminated from the measurement result by averaging each set of two sequential current sense measurements. In addition, the Advanced Current sensing Mode divides the selected OCLO threshold by 2.

Auto Initialization - Selects whether component initialization is automatically called from within the CPU component initialization function **PE_low_level_init** or whether the user is responsible for calling the initialization method.

3.2 SPI configuration

Gen4eXtremeSwitch uses the SPI communication protocol to communicate with the MCU. This protocol is implemented by the **SPIMaster_LDD** component located under **Referenced Components** in the Processor Expert **Components** panel (Figure 2). **SPIMaster_LDD** does not handle arbitration for more simultaneous communication requests on the SPI bus. This functionality is implemented by the **SPI_Device** component, which is exclusively inherited by **Gen4eXtremeSwitch** component. The component supports up to four devices connected in parallel with an **SPI_Device** component configured for each device.

In **SPIMaster_LDD**, the MISO, MOSI and CLK pins and timing settings have to be properly selected according to 12XSF/12XS6 datasheet recommendations (Figure 4). The maximum admissible communication frequency is 5 MHz.

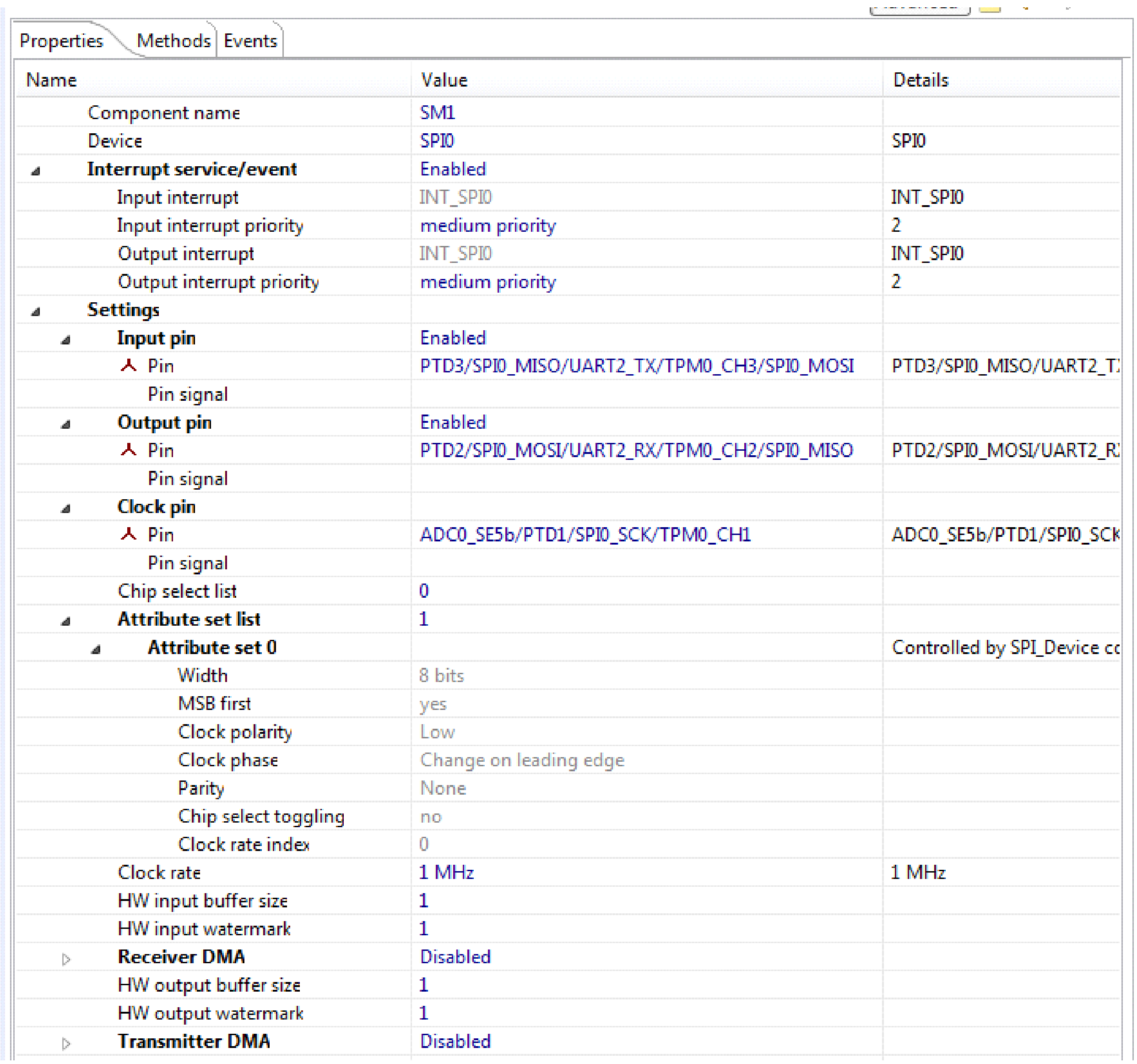


Figure 4. The SPI configuration (pins for FRDM-KL25Z are shown)

The CSB (chip select) pin must be set separately in the **BitIO_LDD** component exclusively inherited by **SPI_Device** (see Figure 5). Because of component implementation limitations, the user must initialize the CSB pin value to 1 as specified in the datasheets.

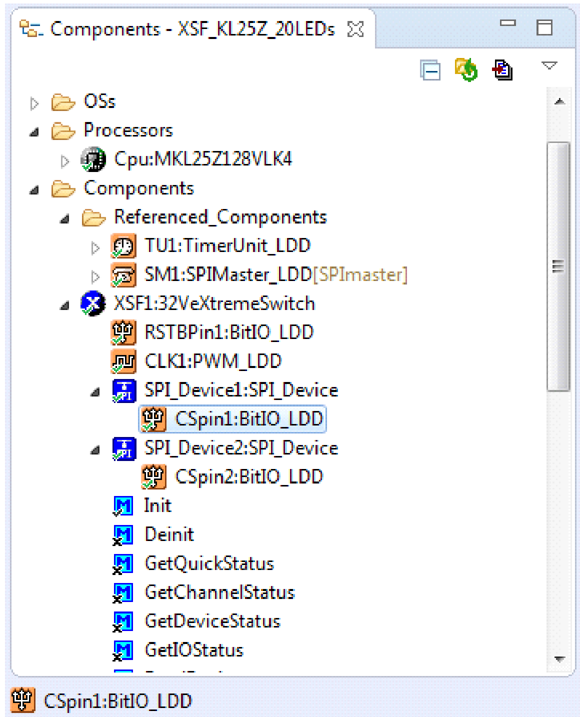


Figure 5. SPI Chip Select configuration

3.3 Component API

The Gen4eXtremeSwitch component provides API functions allowing application code to dynamically configure a device in real-time. To view the available methods and events, click to expand the component in the **Component** folder of the Processor Expert **Components** panel (see Figure 2).

In Processor Expert, methods and events are marked to indicate which of them are included when code is generated. A tick mark beside a method/event indicates that the item is generated; A cross indicates that the item is not generated. Any of these settings can be saved in the Processor Expert Inspector panel. Methods/events with grey text are always generated because they are needed for proper functionality. This forced behavior depends on various combinations of component property settings. For a summarization of available API methods and events and their descriptions, see Table 3.

Table 3. Gen4eXtremeSwitch component API

Method/Event	Description
Init	Initializes the device according to the component properties
Deinit	Deinitializes the device (puts the device in sleep mode)
GetQuickStatus	Returns current status of the device
GetChannelStatus	Gets fault information for all available channels
GetDeviceStatus	Gets information on the status of the device
GetIOStatus	Gets information on the I/O of the device
ReadRegister	Reads the value of the given register via SPI
WriteRegister	Writes the value to the given register via SPI
FeedWatchdog	Toggles the watchdog bit in all devices to avoid watchdog timeout
SetOutputState	Turns on/off channels of the device
SetPWMDuty	Sets PWM duty for the specified channel
SetPWMDutyValues	Sets PWM duty for all available channels of the device
SetGPWMDuty	Sets Global PWM duty cycle value
SetPWMControl	Sets PWM output to Global or individual PWM control
IncrementalPWMControl	Sets LSB step by which the duty cycle is incremented or decremented with every call of this method
SetOCHIOnDemand	Enables the OCHI On Demand (over current high on demand) feature on corresponding channel
OLLEDTrig	Triggers the Open load functionality for LED
SetOLOFF	Enables the OLOFF (open load detection in off state) for the corresponding channel
SetDirectInput	Sets target pin INx high/low voltage The direct input is used to directly control corresponding channel in Fail mode
SetDirectInputValues	Sets target pin INx high/low voltage for all channels simultaneously. The direct input is used to directly control the corresponding channel in fail mode
SetExtClockState	Enables/disables the external clock timer
ConfigureMonitoring	Configures CSNS pin output; sets monitored parameter and synchronization
GetSenseValue	Configures monitoring of current, voltage or temperature on the CSNS pin and returns the corresponding value from the ADC
OnMeasurementSynchronization	Invoked when a measurement synchronization trigger occurs

3.4 Utilized components

The Gen4eXtremeSwitch consists of the Gen4eXtremeSwitch component and a set of referenced components. Figure 6 illustrates these components and their relationship to each other. A description of the components appears immediately below Figure 6. The functionality of the Gen4eXtremeSwitch depends on these components to support functions such as communication and control.

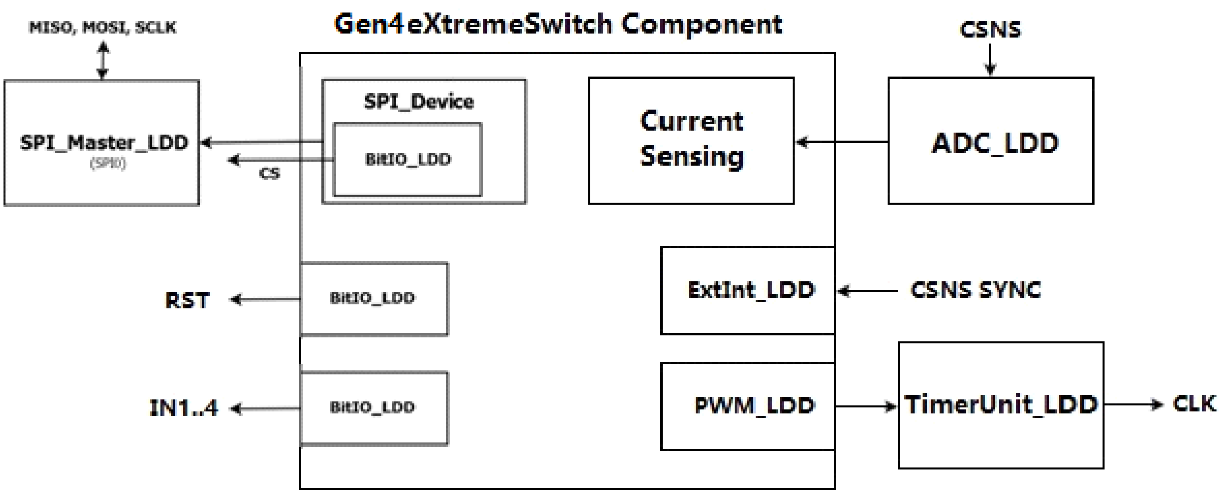


Figure 6. Components used by the Gen4eXtremeSwitch component

The components used by Gen4eXtremeSwitch (sorted as in the component tree) are as follows:

Referenced components

- **SM1:SPIMaster_LDD** – Standard SPI communication model; referenced by SPI_Device
- **TU1:TimerUnit_LDD** – Referenced by CLK1:PWM_LDD
- **AD1:ADC_LDD** – Current sensing

G4XS1:Gen4eXtremeSwitch – configures 12XSF/12XS6 features

- **SPI_Device1:SPI_Device** – Adds bus allocation for SPI communication
  - **CSPin1:BitIO_LDD** – Software chip select
- **RSTBPin1:BitIO_LDD** – Reset pin
- **CLK1:PWM_LDD** – PWM clock generation
- **IN1Pin1:BitIO_LDD** – direct input pin; same for IN2Pin1, IN3Pin1 and IN4Pin1
- **SYNCPin1:ExInt_LDD** – synchronization trigger for ADC

4 Installing the Processor Expert software

This chapter describes the installation of Kinetis Design Studio and the use of Processor Expert for application development. Processor Expert software is available as part of the CodeWarrior Development Studio for Microcontrollers, Kinetis Design Studio or as an Eclipse-based plug-in for installation into an independent Eclipse environment (Microcontroller Driver Suite). For more information about Processor Expert refer to this link: www.nxp.com/ProcessorExpert.

4.1 Installing Kinetis Design Studio

This procedure explains how to obtain and install the latest version of Kinetis Design Studio (version 3.0.0 in this guide). The procedure for CodeWarrior installation is similar.

NOTE

The component and some examples in the component package are intended for CodeWarrior 10.6 (or above) and Kinetis Design Studio 3.0.0 (and above). If CodeWarrior 10.6 and Kinetis Design Studio 3.0.0 are already installed on the system, skip this section.

1. Obtain the latest Kinetis Design Studio installer file from the NXP website here: www.nxp.com/KDS
2. Run the executable file and follow the instructions.

4.2 Downloading the components and example projects

The examples used in this section are based on a pre-configured Kinetis Design Studio project. To download the project and its associated components:

1. Go to the NXP website www.nxp.com/GEN4-EXTREMESWITCH-PEXPERT.
2. Download the zip file containing components and example projects.
3. Unzip the downloaded file and check to see that the folder contains the files listed in Table 4.

Table 4. Downloaded zip file contents

Folder Name	Folder Contents
Component	Processor Expert components folder
Gen4eXtremeSwitch_Bxxx.PEupd	Gen4eXtremeSwitch component update file
SPI_Device_Bxxx.PEupd	SPI_Device component update file
KDS_Examples	Example projects folder for Kinetis Design Studio
G4XS_KL25Z_FreeMASTER	Demo project that demonstrates how to use the FreeMASTER application along with the Gen4eXtremeSwitch component to drive a 32 V eXtreme Switch device and a FRDM-KL25Z
G4XS_KL25Z_IncrementalPWM	Example project that demonstrates the use of incremental PWM feature of a 32 V eXtreme Switch with a FRDM-KL25Z
G4XS_KL25Z_Measurement	Example project that demonstrates how to configure feedback monitoring and how to get measured values with a FRDM-KL25Z
G4XS_KL25Z_ParseStatusData	Example project that demonstrates how to parse and interpret status data using the Gen4eXtremeSwitch component with a FRDM-KL25Z
G4XS_KL25Z_SetPWMDutyParallel	Example project that demonstrates how to set the PWM duty value of selected channels on multiple devices with a FRDM-KL25Z
G4XS_KL25Z_SetPWMDutySingle	Example project that demonstrates how to set the PWM duty value for selected channels on single device with FRDM-KL25Z
G4XS_K64F_FreeMASTER	Demo project that demonstrates the use of the FreeMASTER application along with a Gen4eXtremeSwitch component to drive a 32 V eXtreme Switch device and a FRDM-K64F
G4XS_K64F_IncrementalPWM	Example project that demonstrates how to use the incremental PWM feature of a 32 V eXtreme Switch with a FRDM-K64F
G4XS_K64F_Measurement	Example project that demonstrates how to configure feedback monitoring and how to get measured values with FRDM-K64F.
G4XS_K64F_ParseStatusData	Example project that demonstrates how to parse and interpret status data using the Gen4eXtremeSwitch component with a FRDM-K64F

Table 4. Downloaded zip file contents

Folder Name	Folder Contents
G4XS_K64F_SetPWMDutySingle	Example project that demonstrates how to set the PWM duty value of selected channels on a single device with a FRDM-K64F
G4XS_KV10Z_IncrementalPWM	Example project that demonstrates how to use the incremental PWM feature of a 32 V eXtreme Switch with a FRDM-KV10Z
G4XS_KV10Z_ParseStatusData	Example project that demonstrates how to parse and interpret status data using the Gen4eXtremeSwitch component with a FRDM-KV10Z
G4XS_KV10Z_SetPWMDutySingle	Example project that demonstrates how to set the PWM duty value for selected channels on a single device with a FRDM-KV10Z

4.2.1 Importing the Gen4eXtremeSwitch component into the Processor Expert library

The following steps show how to import the Gen4eXtremeSwitch component (downloaded as part of the zip file) into Kinetis Design Studio. The process for CodeWarrior is similar.

1. Launch Kinetis Design Studio. When the Kinetis Design Studio IDE opens, go to the menu bar and click **Processor Expert** -> **Import Component(s)**.
2. In the pop-up window, locate the component file (.PEupd) in the folder Gen4eXtremeSwitch_PEx_SW\Components. Select the **Gen4eXtremeSwitch _bxxxx.PEupd** and **SPI_Device_bxxxx.PEupd** files. Then click **Open** (see Figure 7).

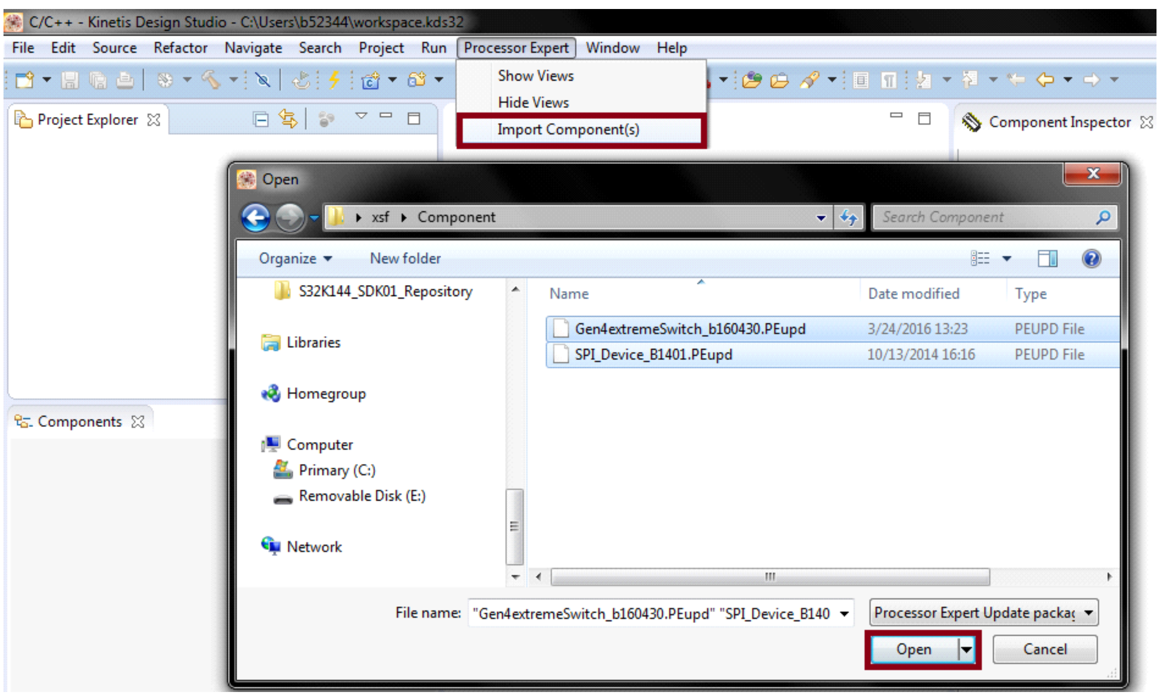


Figure 7. Import the Gen4eXtremeSwitch components

3. Select a repository to import the components into. Then click **OK.**

- In CodeWarrior, components are imported to a predefined default repository.
- In Kinetis Design Studio, you are offered a choice to select a repository.

4. If the import is successful, the Gen4eXtremeSwitch component appears in **Components Library** -> **Software** -> **User Component** (see Figure 8). The component is now ready to use.

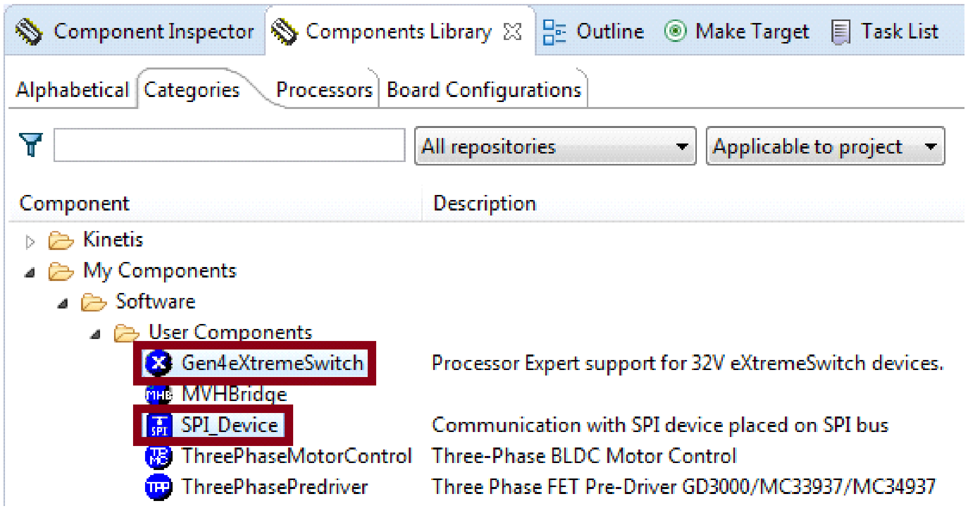


Figure 8. Gen4eXtremeSwitch component location after importing to Kinetis Design Studio

4.2.2 Importing an example project

The following steps show how to import an example project (downloaded with the zip file) into Kinetis Design Studio. The process for CodeWarrior is similar.

1. In the Kinetis Design Studio menu bar, click **File** -> **Import**… In the pop-up window, select **General** -> **Existing Projects into Workspace** and click **Next**.

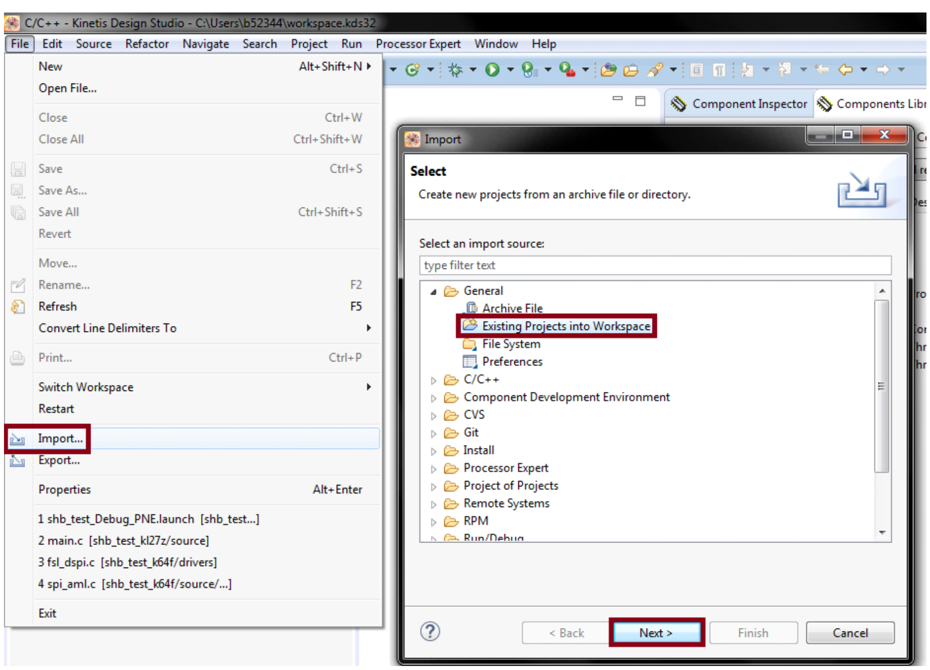


Figure 9. Importing an example file

2. Click **Browse** and locate the folder where the unzipped example files are located. Find the folder **Gen4eXtremeSwitch_PEx_SW\KDS_Examples** and select a project to import. Then click **OK**.

3. With the project now loaded in the **Select root directory** box, click on the **Copy projects into workspace** check box. Then click **Finish**. The project is now in the Kinetis Design Studio workspace where it can be built and run.

4.3 Creating a new project with Processor Expert and the Gen4eXtremeSwitch component

If you elect not to use the example projects, the following instructions describe how to create and setup a new project using the Gen4eXtremeSwitch component. If you do not have Gen4eXtremeSwitch components in the Processor Expert Library, follow the steps in Section 4.2.1, Importing the Gen4eXtremeSwitch component into the Processor Expert library, page 14.

The following procedure describes how to create a new project using Kinetis Design Studio. The process for CodeWarrior is similar.

1. In the Kinetis Design Studio menu bar, select **File** -> **New** -> **Kinetis Project**. When the **New Kinetis Project** dialog box opens, enter a project name into the text box and then click **Next**.
2. In the **Devices** dialog box, select the MCU class for the appropriate MCU. In Figure 10 MKL25Z128xxxx4 has been selected. Then click **Next**.
3. In the **Rapid Application Development** dialog box, make sure that the **Processor Expert** option is selected. Then click **Next**.
4. In the **Processor Expert Target Compiler** dialog box, select a compiler (GNU C Compiler in Figure 10) and click **Finish**.

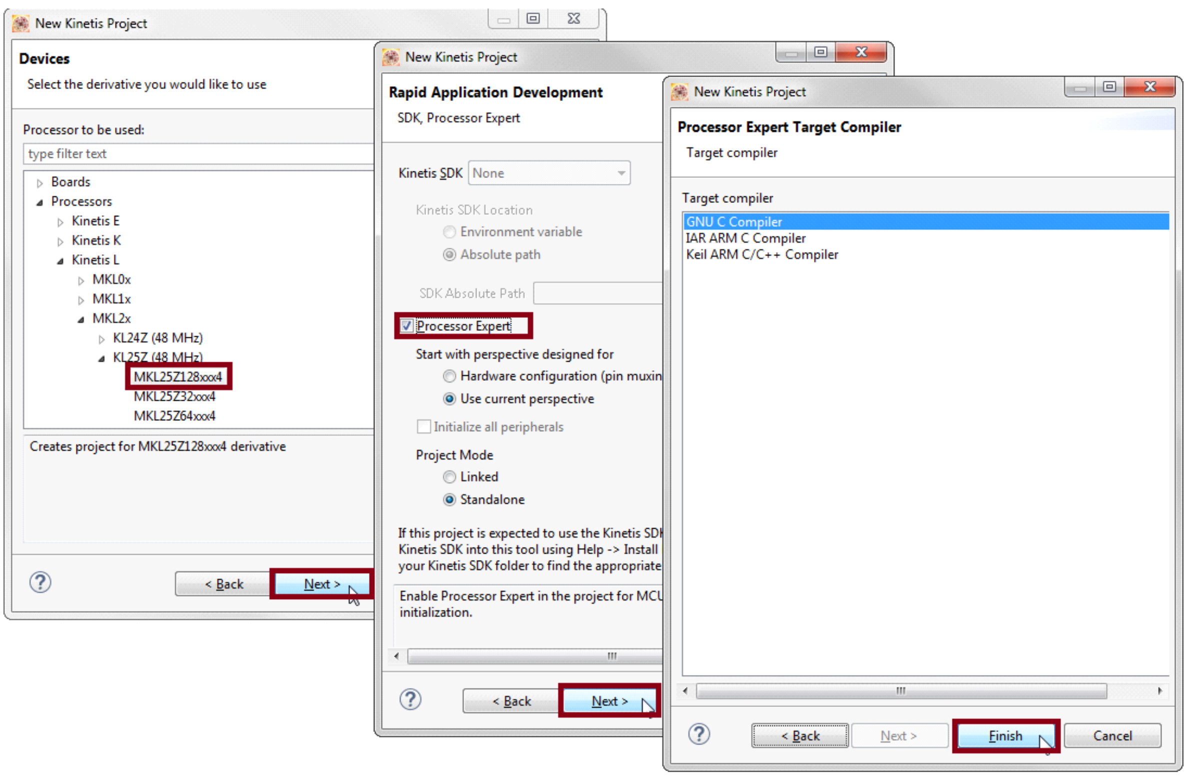


Figure 10. Selecting a device, the Rapid Application Development options and compiler

5. Figure 11 shows the **Projects Explorer** panel and the **Components** panel after the project has been successfully created. Before the project can be built and run, add the Gen4eXtremeSwitch component (imported in Section 4.2.2, Importing an example project, page 15) into the project. Section 4.3.1, Adding a Gen4eXtremeSwitch component into the project, page 17 outlines this procedure.

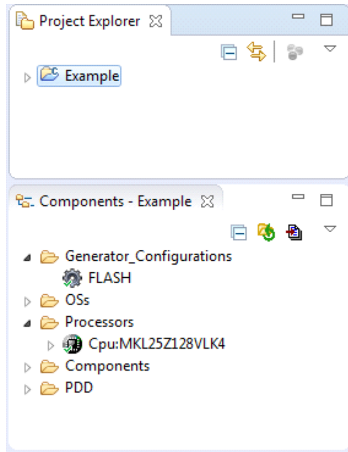


Figure 11. Project Explorer and Components panels with project created

4.3.1 Adding a Gen4eXtremeSwitch component into the project

1. Find the Gen4eXtremeSwitch component in the **Components Library** and add it into the project (see Figure 12). The Gen4eXtremeSwitch component is located in the workspace directory selected when importing the component.

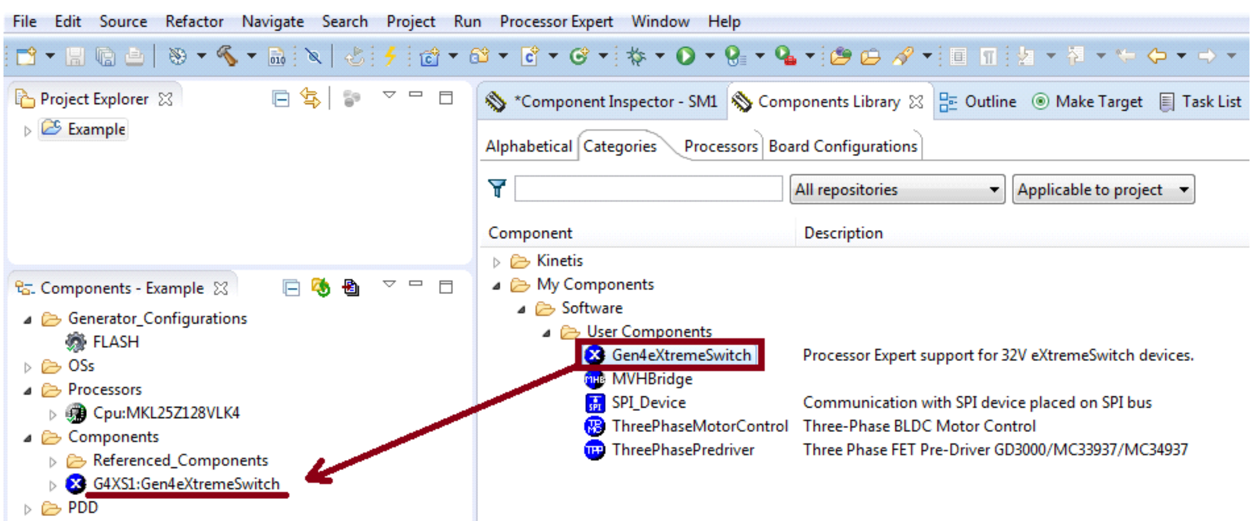


Figure 12. Add the Gen4eXtremeSwitch component to the project

2. To view the **Component Inspector** options, double-click on the Gen4eXtremeSwitch component in the **Components** panel.

4.4 Setting up the project

Once the new project has been created and the Gen4eXtremeSwitch component has been added into it, the component properties in the project must be set up. Make sure to read Section 3.1, Component settings, page 6, which describes the component's capabilities and what must be done to configure its properties.

Gen4eXtremeSwitch uses several components (see Figure 13). Configure all the components in the following order:

1. Set up the referenced **SPI_Master_LDD** component.
2. Set up CS pin under the inherited **SPI_Device** component.
3. Set up the inherited **Gen4eXtremeSwitch** component.

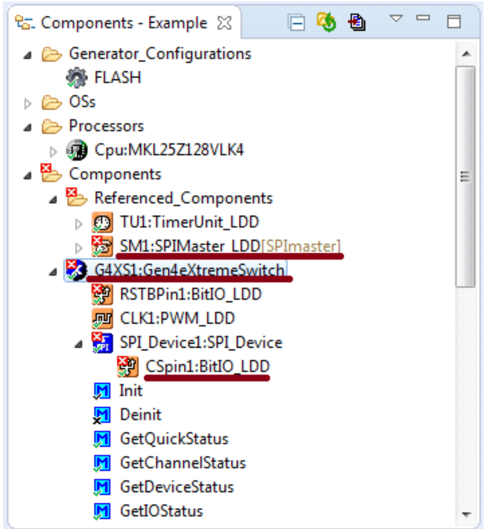


Figure 13. Setting up the components

4.5 Generating driver source code

Once you have completed configuring the components, the application is ready to generate the driver code. The process is as follows:

1. Click on the **Generate Processor Expert Code** icon in the upper right corner of the **Components** panel. (See Figure 14)

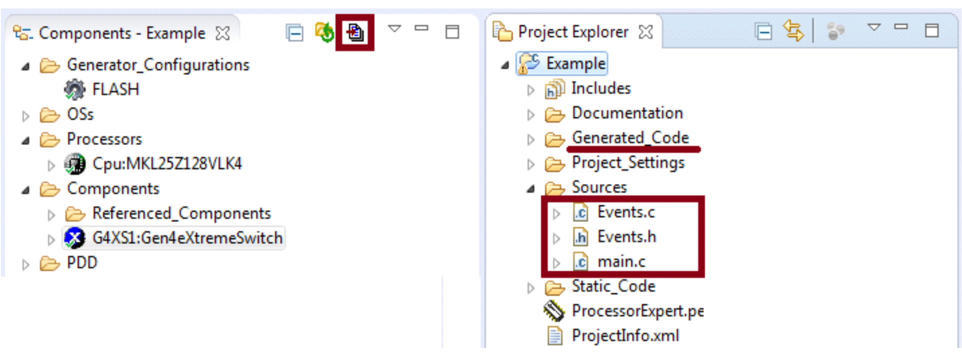


Figure 14. Generating the driver code and code location

2. The driver code for the device is generated into the **Generated_Code** folder in the **Project Explorer** panel. The component only generates the driver code. It does not generate application code (see Figure 14)

4.6 Writing the application code

All of the application code must reside in the **Sources** folder in the project directory. The code in **main.c** and **Events.c** can be modified, but take care to retain the original comments related to usage directions.

To add a component method into the application source code:

1. In the **Components** panel for the project, click on **Components**. Find the desired method to add to the code.
2. Drag and drop the method directly into the source code panel.
3. Add the appropriate parameters to the method. Hovering the mouse over the method displays a list of the required parameters.

For example, open the Gen4eXtremeSwitch component method list, drag and drop **SetPWMDuty** into **main.c** and add the necessary parameters. (See Figure 15).

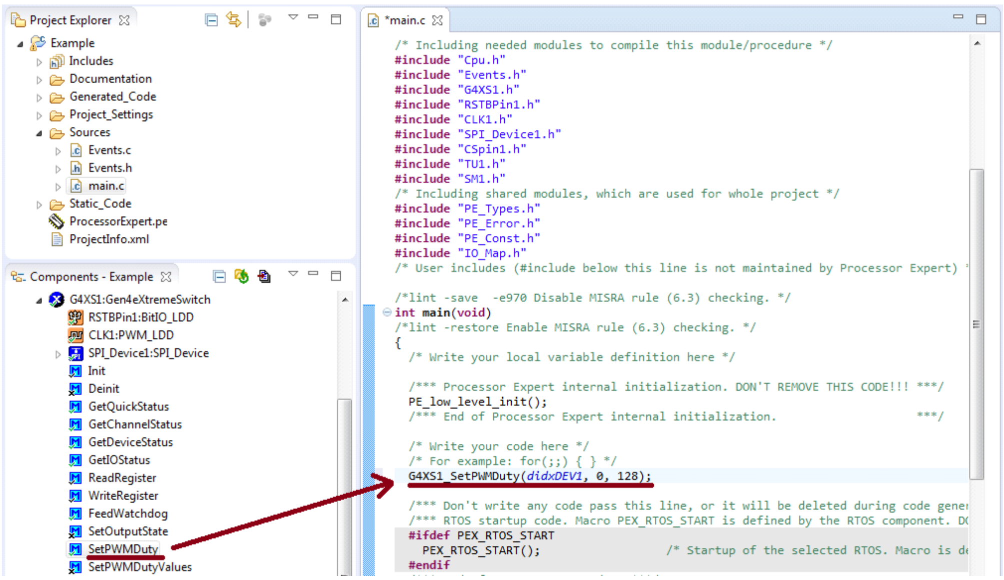


Figure 15. Adding component methods

Hovering the mouse over any of the methods displays a description of the method, including a list of required parameters.

The Gen4eXtremeSwitch component encompasses a Help feature that describes component properties, methods and typical usage. To invoke the Help feature, do the following:

1. In the **Components** view, right-click Gen4eXtremeSwitch component and select **Help on Component**.
2. A web page with the Help information is displayed.

4.6.1 Compiling, downloading and debugging

To compile a project, click on the compile icon in the tool bar (see Figure 16).

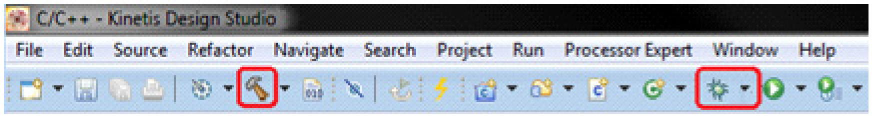


Figure 16. Compiling the project

The process for downloading an application onto a board may differ according to the MCU board used. For any questions, see the Kinetis Design Studio or CodeWarrior user's guide.

To download and debug with a FRDM-KL25Z MCU board, do the following:

1. Click the arrow next to the debug icon in the menu bar and select **Debug Configurations…**
2. In the **Debug Configurations** dialog box, click **Example_Debug_PNE** under **GDB PEMicro Interface Debugging** (see Figure 17).
3. Make sure that **C/C++ Application** contains a path to the .elf file of the project (see Figure 17).

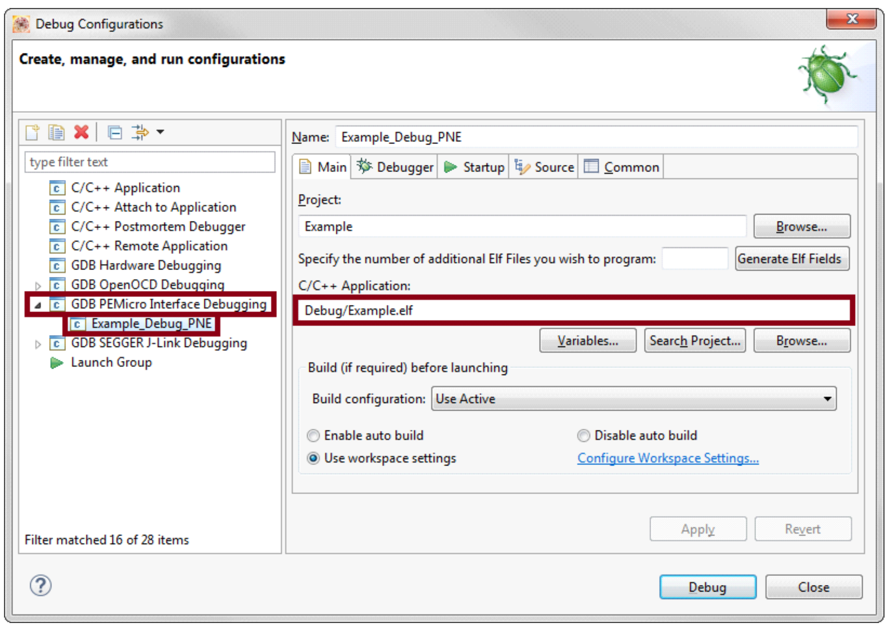


Figure 17. Downloading the application (a)

4. Click the **Debugger** tab and set the **Interface** option to **OpenSDA Embedded Debug - USB Port**. Then click the **Refresh** button next to the **Port** setting to update the list of available USB ports (see Figure 18).
5. Make sure the **Target** is set to **KL25Z128M4**. If not, change the target by clicking the **Select Device** button. In the **Select Target Device** dialog box, go to **Kinetis L** -> **KL2x** -> **KL25Z128M4** and confirm with the **Select** button.
6. Click **Debug**. Kinetis Design Studio downloads and launches the program onto board.

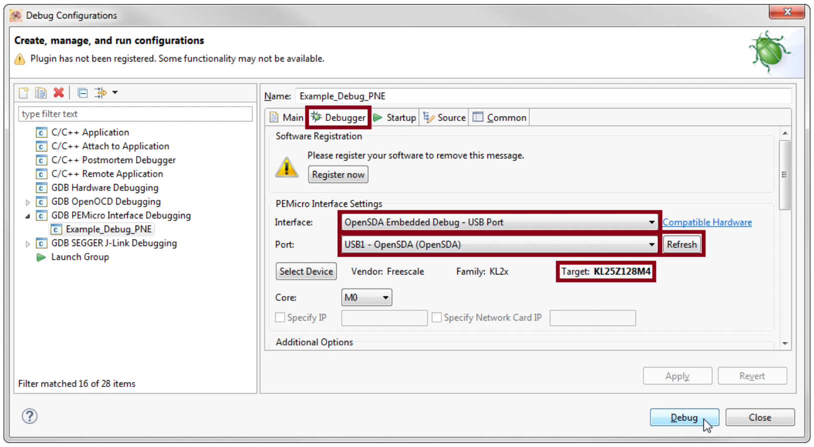


Figure 18. Downloading the application (b)

5 References

Following are URLs where you can obtain information on related NXP products and application solutions:

Table 5. References

NXP.com Support Pages	Description	URL
MC12XSF	Product Summary Page	www.nxp.com/MC12XSF
MC12XS6	Product Summary Page	www.nxp.com/MC12XS6
FRDM-MC12XSF	Tool Summary Page	www.nxp.com/FRDM-MC12XSF
FRDM-KL25Z	Tool Summary Page	www.nxp.com/FRDM-KL25Z
FRDM-K64F	Tool Summary Page	www.nxp.com/FRDM-K64F
FRDM-KV10Z	Tool Summary Page	www.nxp.com/FRDM_KV10Z
Kinetis Design Studio	Software	www.nxp.com/KDS
CodeWarrior	Software	www.nxp.com/codewarrior
Processor Expert	Software	www.nxp.com/ProcessorExpert
Gen4eXtremeSwitch components and example files	Software	www.nxp.com/GEN4-EXTREMESWITCH-PEXPERT

5.1 Support

Visit www.nxp.com/support for a list of phone numbers within your region.

5.2 Warranty

Visit www.nxp.com/warranty to submit a request for tool warranty.

6 Revision history

Revision	Date	Description of Changes
1.0	5/2016	• Initial release

How to Reach Us:

Home Page:
NXP.com

Web Support:
http://www.nxp.com/support

Document Number: PEXMC12XSF-MC12XS6UG
Rev. 1.0
5/2016