Document Number: PEXMC36XSDSWUG
Rev. 2.0, 10/2016

36 V eXtreme switch Processor Expert component

Contents

1 Overview

This documentation describes how to install and use Processor Expert in conjunction with the 36VeXtremeSwitch component.

The 36VeXtremeSwitch component supports the MC36XSD 36 V Industrial Low $R_{DS(on)}$ eXtreme Switch. NXP's TWR-MC36XSDEVB tower platform and FRDM-MC36XSD-EVB Freedom evaluation boards are based on this chip. See the related user guides and data sheets for detailed information.

2 36VeXtremeSwitch compatibility

2.1 Peripheral requirements

The following peripherals and resource requirements are critical to the MCU's ability to handle a given part:

- **SPI Module** is required for communication (SI, SO, SCLK, CSB)
- **GPIO** is required for device reset (RSTB)

The following peripherals are not critical, but may be required to support other features:

- **TPM/FTM** timer (PWM, single channel) is required for external clock generation
- **ADC** and **GPIO** are required for current sensing (CSNS) and synchronization (CSNS SYNC)

2.2 Supported devices

This section describes which analog devices are supported by this component. It also compares the features and capabilities of the supported devices.

The MC36XSD is part of a family of intelligent, dual high-side eXtreme switches designed for 36 V systems. This product family offers 1.0 A to 12 A DC devices that are footprint and software compatible within the PQFN subfamily. Each device can be programmed to control a broad range of loads, including lamps, LEDs, motors and solenoids.

The 36XSD family products differ in $R_{DS(on)}$ resistance of the outputs (see Table 1).

Table 1. Device features

Device	Common Features	Device Specific Features
MC06XSD200	• 3.3 V and 5.0 V compatible 16-bit SPI port for device control, configuration and diagnostics at rates up to 8.0 MHz • Separate bulb and DC motor latched overcurrent handling • Sleep mode with minimal supply current (< 10 µA at 24 V) • Parallel output operating mode with improved switching synchronization • Individually programmable internal/external PWM clock signals (switching frequency, duty cycle, slew rate, switch-on time-shift) • Overcurrent, short-circuit and overtemperature protection with programmable auto-retry functions • Accurate temperature and current sensing (high/low sensing ratios/offset compensation) • Normal operating range: 8.0 V to 36 V, extended operating range: 6.0 V to 58 V • Open load detection (channel in OFF and ON state), also for LED applications (7.0 mA typ.)	• Two fully-protected 6.0 mΩ (at 25 °C) high-side switches • Up to 9.0 A steady-state current per channel
MC10XSD200		• Two fully protected 10 mΩ (at 25 °C) high-side switches • Up to 6.0 A steady-state current per channel
MC16XSD200		• Two fully-protected 16 mΩ (at 25 °C) high-side switches • Up to 3.0 A steady-state current per channel
MC50XSD200		• Two fully-protected 50 mΩ (at 25 °C) high-side switches • Up to 1.2 A steady-state current per channel

2.3 Supported MCUs

The 36VeXtremeSwitch component supports the products listed in Table 2.

Table 2. Supported products

Supported Products	Kinetis Design Studio Support
TWR-K20D72M	Yes
TWR-K70F120M	Yes
TWR-K64F120M	Yes
FRDM-K64F	Yes
FRDM-KL25Z	Yes

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

2.4 Tower board settings

The TWR-MC36XSDEVB is a peripheral module circuit board that provides a means of exercising eXtreme Switch functions in a Tower System environment. The Tower System includes a debug port and a communication port that connect to a PC and allow for the downloading and debugging of application code using the Kinetis Design Studio IDE.

The board features:

- Four eXtreme switch devices: MC06XSD200, MC10XSD200, MC16XSD200, MC50XSD200
- SPI communication in daisy chain mode with four devices (selectable by switch)
- Single SPI communication support
- On-board LEDs indicating ON/OFF status for each high-side channel
- An MCU A/D converter pin for current/temperature sensing of four devices

To run the example projects that are part of the component package, the TWR-MC36XSDEVB board must be in a Tower System configuration with one of the supported MCU boards (TWR-K20D72M or TWR-K70F120M). Figure 1 shows how to connect the hardware.

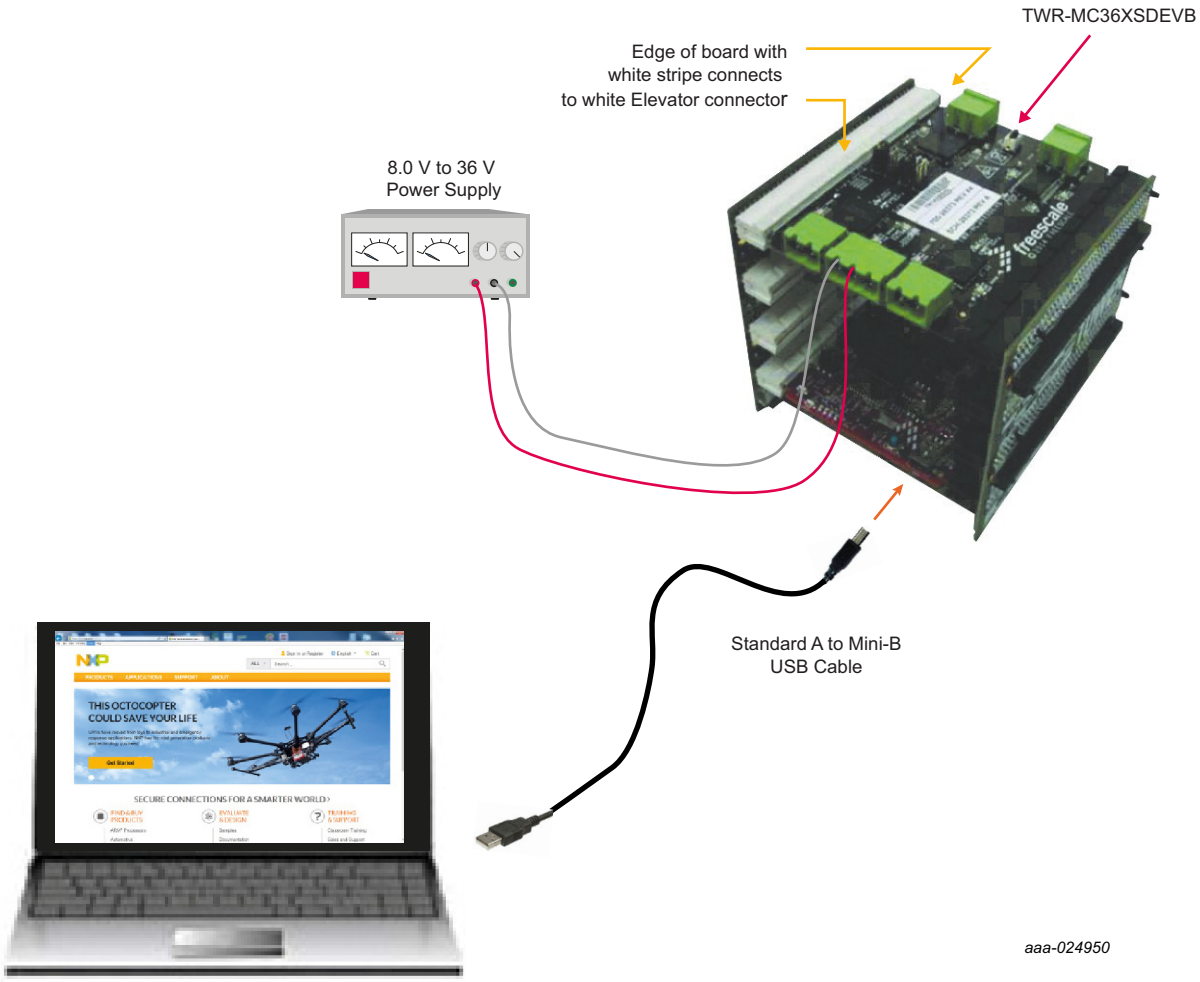

Figure 1. Hardware configuration

Switch SW2 on the TWR-MC36XSDEVB board must be set as shown in Table 3.

Table 3. TWR-MC36XSDEVB switch settings

SPI Work Mode	Description	Connection			
		SW2 1	SW2 2	SW2 3	SW2 4
Single	Connect MC06XSD200 only	ON	OFF	ON	OFF
Daisy Chain	Connect the four devices by daisy chain	OFF	ON	OFF	ON

2.5 Freedom board settings

The FRDM-MC36XSD-EVB consists of four power high-side channels driven through a parallel and SPI interface. The two devices on board are daisy chained. The board can be configured for use with a FRDM-KL25Z board and the 36VeXtremeSwitch component.

Note:

When using the FRDM-MC36XSD-EVB, make sure that the maximum supply voltage (VPWR) stays within the 5.0 V to 36 V range. Operating outside this range may cause damage to the board.

To configure the FRDM-MC36XSD-EVB for use with the FRDM-KL25Z and Kinetis Design Studio, do the following:

1. Connect the FRDM-MC36XSD-EVB to the FRDM-KL25Z using the Arduino™ connectors on each board.
2. Connect the USB cable (not supplied with the kit) between the PC and the USB port labeled **SDA** on the FRDM-KL25Z board.
3. With the power switched off, attach the DC power supply to the VBAT and GND screw connector terminal (J20) on the evaluation board.
4. Connect the load to the screw terminal (J21). Figure 2 illustrates the hardware configuration using a FRDM-KL25Z.

For more details on setup of the FRDM-KL25Z, refer to the FRDM-KL25Z tool summary page at http://www.nxp.com/FRDM-KL25Z or the material at http://www.element14.com/community/docs/DOC-49219

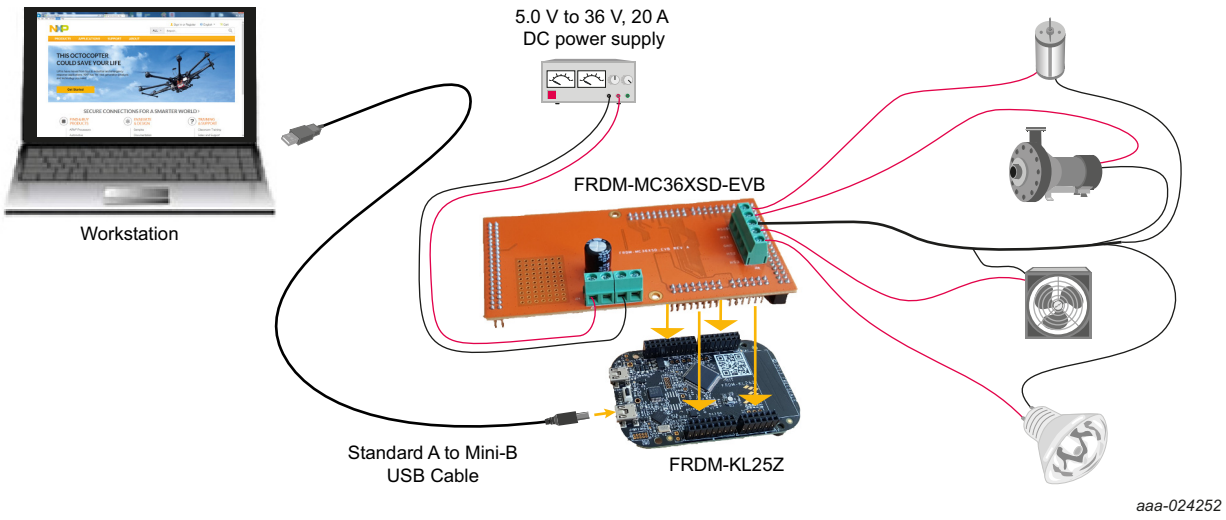

Figure 2. FRDM-MC36XSD-EVB configured for use with a FRDM-KL25Z board

3 The 36VeXtremeSwitch component

The 36VeXtremeSwitch component is located in the Components folder in the active projects window (see Figure 3). This folder contains two subfolders:

- **Referenced_Components** contains components to configure SPI communication properties.
- **XSD1:36VeXtremeSwitch** contains the component to configure the 36 V eXtreme Switch component features.

The functionality of the 36VeXtremeSwitch depends on the component property settings assigned through Processor Expert.

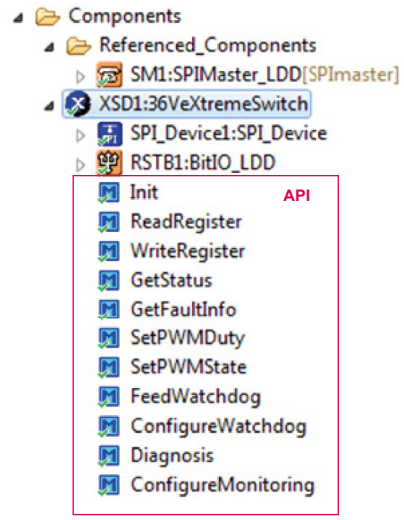

Figure 3. 36VeXtremeSwitch Processor Expert component

The component includes Help documentation, which can be accessed by right-clicking the 36VeXtremeSwitch component in the component tree. The **Help on Component** window provides information on all properties and methods of the component. Access the **Typical Usage** section to view examples showing how to work with API methods.

3.1 Component settings

Selecting the 36VeXtremeSwitch component in the component tree gains access to properties in the Component Inspector. These properties determine the component's general settings and its behavior after initialization. Some of these properties can later be changed in the application code by using the provided API.

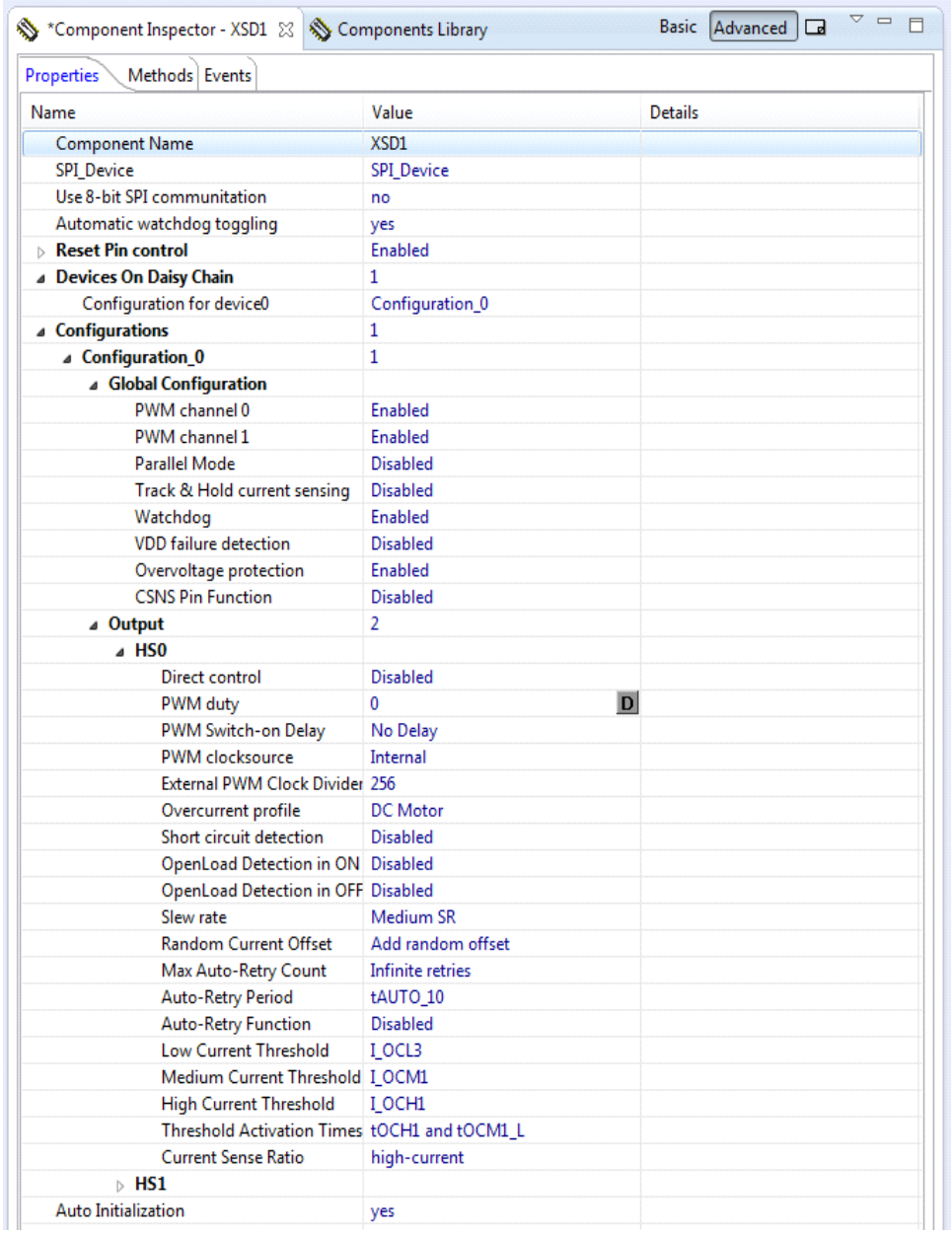

Figure 4. 36VeXtremeSwitch component properties

SPI communication – SPI_Device link and 8/16-bit SPI selection. The eXtreme switch uses 16-bit communication, but for MCUs with only 8-bit communication, the 16-bit frame can be divided into two 8-bit frames.

Watchdog toggling – When automatic watchdog toggling is enabled, the watchdog bit is toggled with every SPI read/write operation. When **Watchdog toggling** is disabled, application code must call the **FeedWatchdog** method periodically.

Reset Pin control – Links to the **BitIO_LDD** component used as the reset pin.

Devices On Daisy Chain – The number of devices connected in a daisy chain. Each device can have its own configuration or all devices can share the same configuration.

Device global configuration - Device configuration or common features for both channels.

- **PWM channel 0** and **PWM channel 1** – These properties activate the PWM control mode on channels 0 and 1. To improve switching, synchronize between these channels by enabling **Parallel Mode**.
- **Watchdog** – When enabled, watchdog feeding must be sent to the device to avoid a watchdog timeout. **VDD failure detection** and **Overvoltage protection** enable failure detection on VDD and VPWR.
- **CSNS pin** – This property outputs an analog signal to indicate current on channel 0, current on channel 1, summed current of both channels and temperature. When **Track & Hold current sensing** is activated, the value of the channel's load current is kept available after turn-off.

Output configuration – Channel specific configuration such as PWM duty, overcurrent profile and threshold, or open load detection.

- Each channel is controlled either internally by PWM or externally by using direct inputs (IN1, IN2). External control is enabled by the **Direct control** property. When **Direct control** is disabled, the direct input pins have no function.
- For PWM control, set the clock period values for the **PWM duty** and the **PWM Switch-on Delay** properties. Also set the **PWM clock source** for either an internal or an external clock. If an external clock is selected, the **External PWM Clock Divider** property must be set to the appropriate value.
- The **Output** configuration section contains settings for short-circuit and open load detection, slew rate and current thresholds. Properties in this section set **Current Sense Ratio** values and **Random current offset** values for current sensing.

Auto Initialization – When auto initialization is enabled, the **Init** method is called automatically within the PE initialization function **PE_low_level_init()**.

3.2 SPI configuration

The 36VeXtremeSwitch uses the SPI communication protocol to communicate with the MCU. This protocol is implemented by the **SPIMaster_LDD** component, which resides in the referenced components (shared components) folder in the Components panel (see Figure 5). However, this component does not handle arbitration for simultaneous communication requests on the SPI bus. This functionality is implemented by the **SPI_Device** component, which is exclusively inherited by the **36VeXtremeSwitch** component.

In **SPIMaster_LDD**, the (**MISO**, **MOSI**, **CLK**) pins and timing settings must comply with the specifications in the relevant MC36XSD data sheet. The maximum admissible communication frequency is 8 MHz.

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

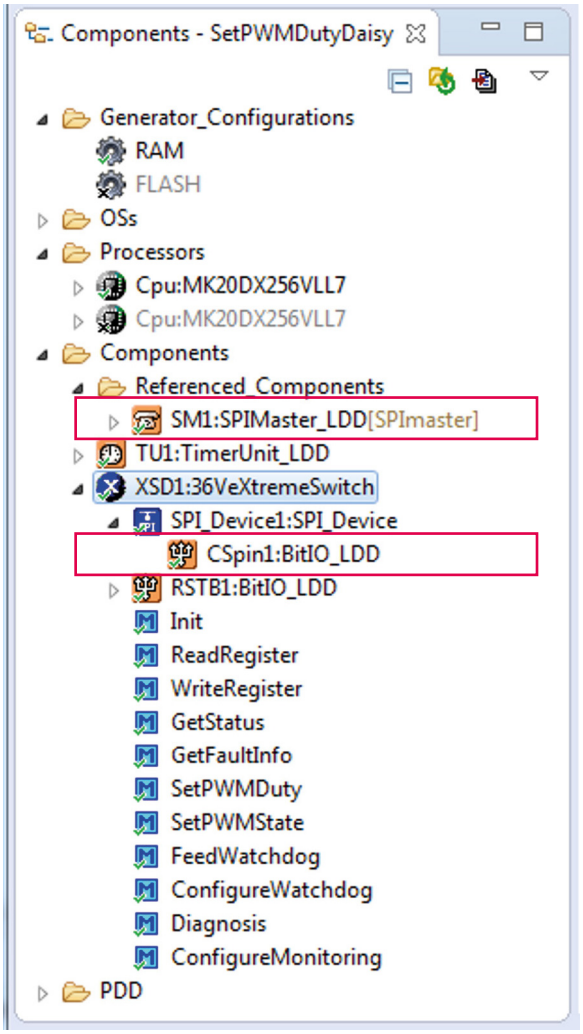

Figure 5. SPI chip select configuration

3.3 Component API

The 36VeXtremeSwitch component contains API functions that allow 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 in the **Components Panel** (see Figure 3).

Methods and events marked with green tick marks are included when source code is generated; methods and events marked with black crosses are not included. To change these settings, go to the **Component Inspector** panel and select the **Methods** tab. Note that some methods/events are always generated because they are needed for proper functionality. This forced generation of methods/events depends on various combinations of component property settings.

For a summarization of available API methods and events and their descriptions, see Table 4.

Table 4. 36VeXtremeSwitch component API

Method	Description
Init	Initializes the device and applies settings selected in the component properties. This includes initialization of inherited components and other features.
Deinit	Deinitializes the device; sets the reset pin to low and consequently clears all device registers (puts the device in Sleep mode)
GetQuickStatus	Returns the current status of the device
GetChannelStatus	Returns fault information for all available channels
ReadRegister	Reads a value from the selected register via the SPI
WriteRegister	Writes a value to the selected register via the SPI
GetStatus	Returns current general status information
GetFaultInfo	Returns fault status information
SetPWMDuty	Sets the PWM duty cycle for the specified channel
SetPWMState	Enables or disables the PWM channel
FeedWatchdog	Feeds the watchdog to avoid a watchdog time-out when the watchdog is enabled
ConfigureWatchdog	Enables or disables the watchdog
Diagnosis	Runs a diagnosis routine that returns the channel configuration, product identification, current logic status of direct inputs, report of external clock failure (if external clock is enabled) and report of calibration failure
ConfigureMonitoring	Sets current/temperature monitoring option of CSNS pin

3.4 Components used by the 36VeXtremeSwitch component

The 36VeXtremeSwitch uses several components that are shared or inherited. Figure 6 illustrates these components and their relationship to each other. A description of the inherited and referenced components appears immediately below Figure 6. The functionality of the 36VeXtremeSwitch in terms of communication, control, etc. depends on these components.

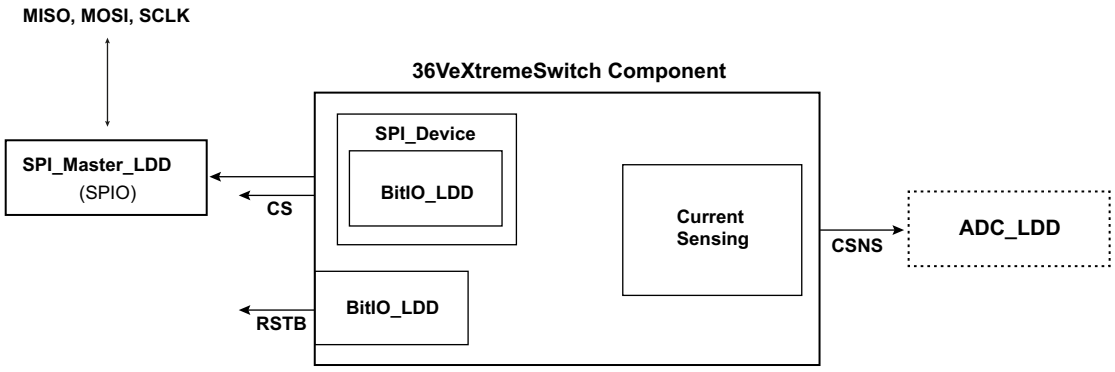

Figure 6. Components used by the 36VeXtremeSwitch

Referenced components

SM1:SPIMaster_LDD	Configuration of SPI communication—Referenced by **SPI_Device**

XSF1:36VeXtremeSwitch (configured features of 36XSD)

SPI_Device1:SPI_Device	Adds bus allocation of SPI communication
CSPin1:BitIO_LDD	Software chip select
RTSB1:BitIO_LDD	Input/output reset pin

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 the following 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 3.2.0 in this guide). The procedure for CodeWarrior installation is very similar.

NOTE

The component and some examples in the component package are intended for Kinetis Design Studio 3.2.0 (and above) and CodeWarrior 10.6 (or above). If Kinetis Design Studio 3.2.0 and CodeWarrior 10.6 are already installed on the system, skip this section.

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

4.2 Downloading the components and example projects

This section describes how to get the 36VeXtremeSwitch component and the example projects from NXP's website.

To download the project and its associated components:

1. Go to www.nxp.com/36V-EXTREMESWITCH-PEXPERT and download the example projects and the 36VeXtremeSwitch component zip file.
2. Unzip the downloaded file and check that the folder contains the files listed in Table 5

Table 5. Contents of the downloaded zip file

Folder Name	Folder Contents
Component	Processor Expert components folder
36VeXtremeSwitch_Bxxx.PEupd	36VeXtremeSwitch component update file
SPI_Device_Bxxx.PEupd	SPI_Device component update file
KDS_Examples	Example projects folder for Kinetis Design Studio
FRDM-MC36XSD-EVB	Examples using MC36XSD Freedom board (two devices in daisy chain)
FRDM-K64F or FRDM-KL25Z	Examples for specific MCU Freedom board (K64 or KL25)
XSD_FRDM-xxx_ChannelMonitoring	Current and temperature measurement with console output
XSD_FRDM-xxx_FreeMASTER	Changing duty cycle, current/temperature measurement and fault monitoring using FreeMASTER GUI
XSD_FRDM-xxx_SetPWMDutyDaisy	Simple example showing changing duty cycle
TWR-MC36XSDEVB	Examples using MC36XSD Tower board (one or four devices)
TWR-K20D72M or TWR-K64F120M or TWR-K70F120M	Examples for specific MCU Tower board (K20, K64 or K70)
XSD_TWR-xxx_ChannelMonitoring	Current and temperature measurement with console output
XSD_TWR-xxx_FreeMASTER	Changing duty cycle, current/temperature measurement and fault monitoring using FreeMASTER GUI (only one device)
XSD_TWR-xxx_SetPWMDutyDaisy	Simple example showing changing duty cycle (four devices)
XSD_TWR-xxx_SetPWMDutySingle	Simple example showing changing duty cycle (one device)

4.2.1 Importing 36VeXtremeSwitch components into the Processor Expert library

To install the 36VeXtremeSwitch component:

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 **36VeXtremeSwitch_PEx_SW\Component**. Select 36VeXtremeSwitch_bxxxx.**PEupd** and **SPI_Device_b**xxxx.PEupd files, the click **Open** (see Figure 7).
3. Select the repository where the imported packages are to reside, then confirm by clicking **OK**. (Note that, in CodeWarrior, components are imported to a predefined default repository. Kinetis Design Studio offers users a choice to select the repository.)

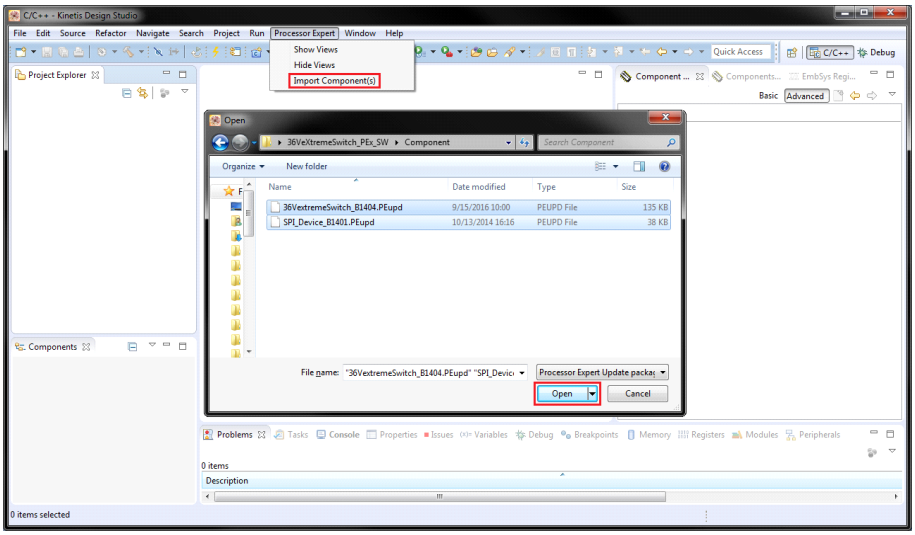

Figure 7. Importing the 36VeXtremeSwitch component

4. If the import is successful, the 36VeXtremeSwitch component is in **Components Library** -> **Software** -> **User Components** (see Figure 8). The component is ready to use.

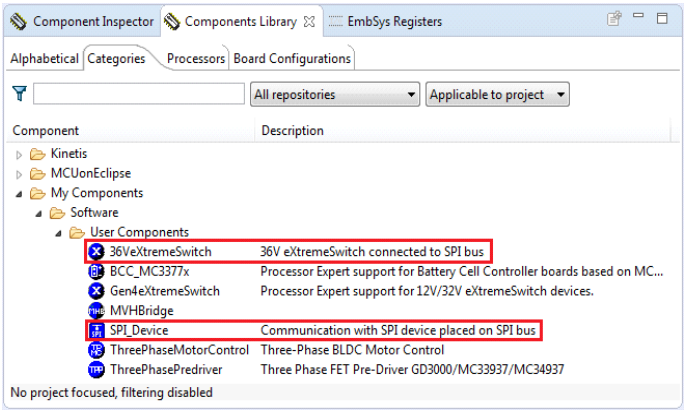

Figure 8. 36VeXtremeSwitch component location after importing into Kinetis Design Studio

4.2.2 Importing an example project into Kinetis Design Studio

The following steps show how to import an example from the downloaded zip file into Kinetis Design Studio.

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

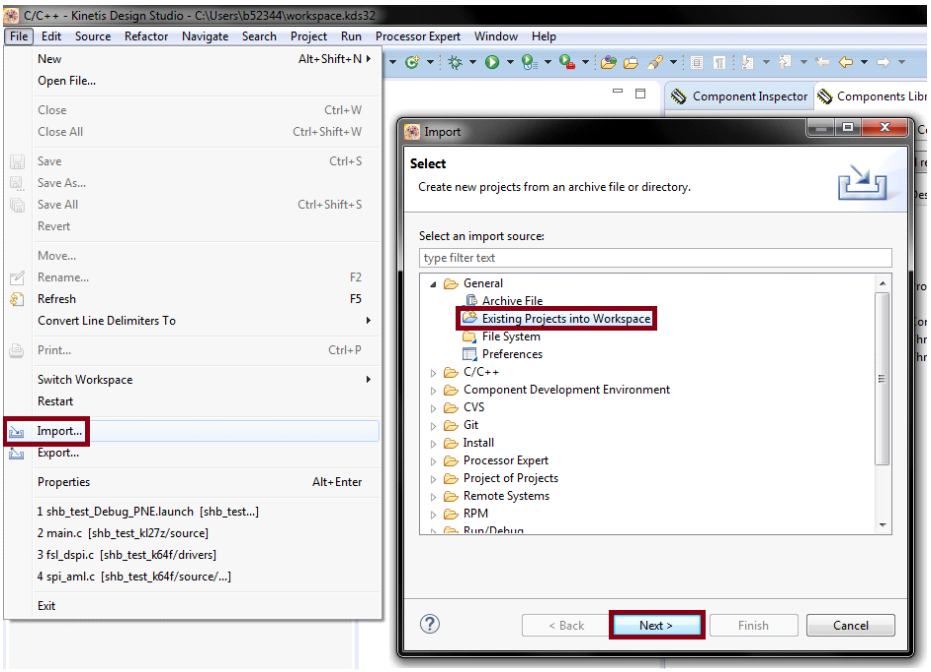

Figure 9. Importing an example project

2. Click Browse and locate the folder where the downloaded example files were unzipped. Find the folder **36VeXtremeSwitch_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 the **Copy projects into workspace** checkbox. Then click **Finish**. The project is now in the Kinetis Design Studio workspace where it can be built and run.

4.3 Create a new project with Processor Expert and 36VeXtremeSwitch component

If the example project is not used, the following instructions describe how to create and setup a new project that uses the 36VeXtremeSwitch component. If the 36VeXtremeSwitch component is not already in the Processor Expert library, follow the steps in Section 4.2.1, "Importing 36VeXtremeSwitch components into the Processor Expert library".

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, then click **Next**.
2. In the **Devices** dialog box, select the MCU class to use (in Fig 9 MKL25Z128xxx4 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 the compiler to be used (GNU C Compiler has been selected in Figure 10).

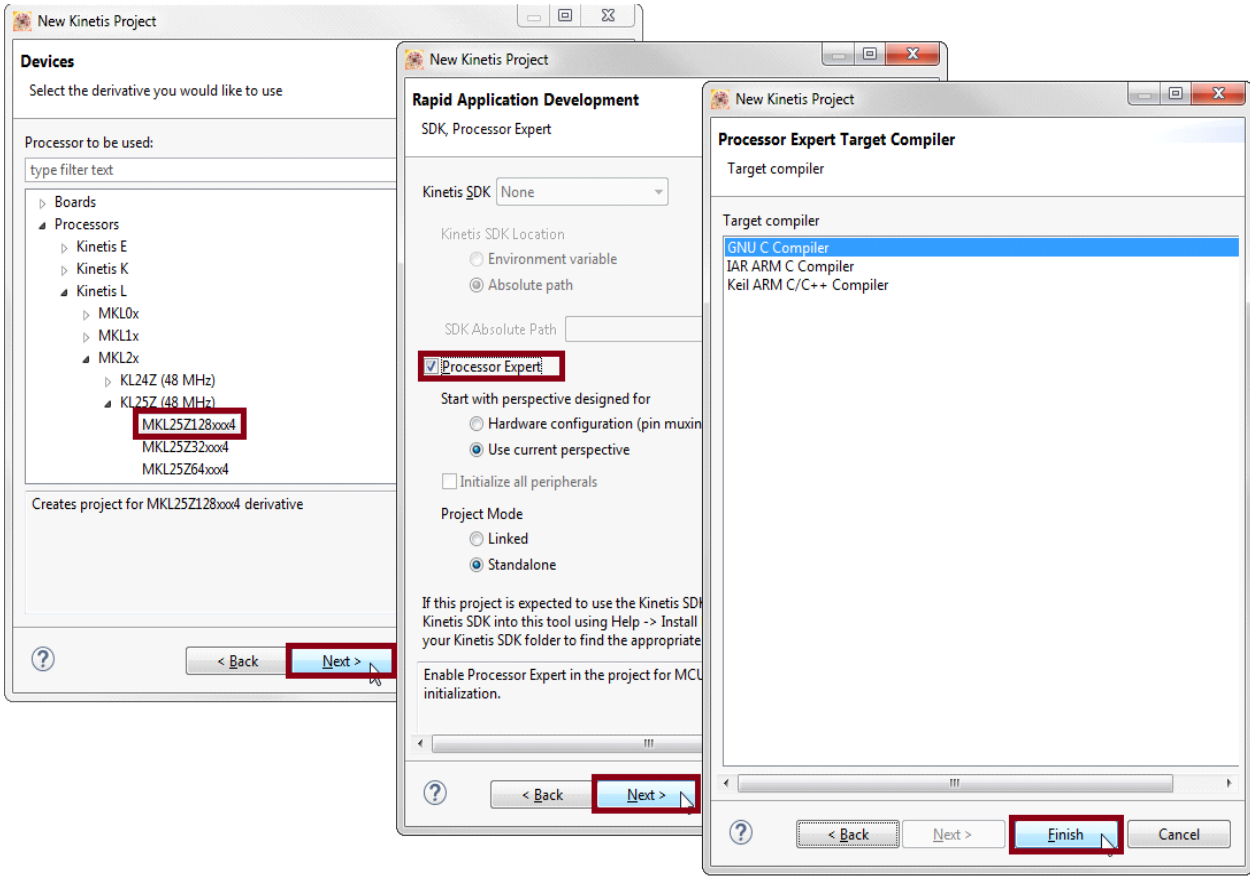

Figure 10. Selecting a device, the Rapid Application Development options and compiler

5. Figure 11 shows the Project Explorer panel and the Components panel after the project has been successfully created.

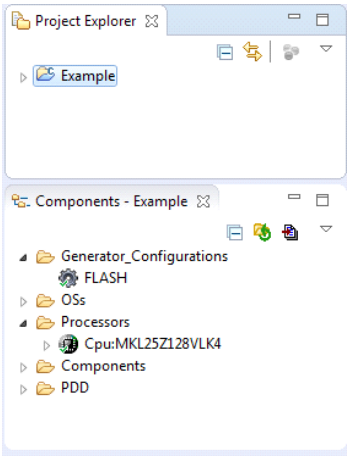

Figure 11. Project Explorer and Components panel with the project created

4.4 Setting up the Project

Once the new project has been created and 36VeXtremeSwitch component added into it, it must be set up. Read Section 3.1, "Component settings" describing the capabilities of the component and what is required to configure it properly.

36VeXtremeSwitch uses several components (see Figure 12). Configure all the components in this order:

1. Set up referenced **SPI_Master_LDD** component.
2. Set up **CS pin** under inherited **SPI_Device** component.
3. Set up **36VeXtremeSwitch** component.

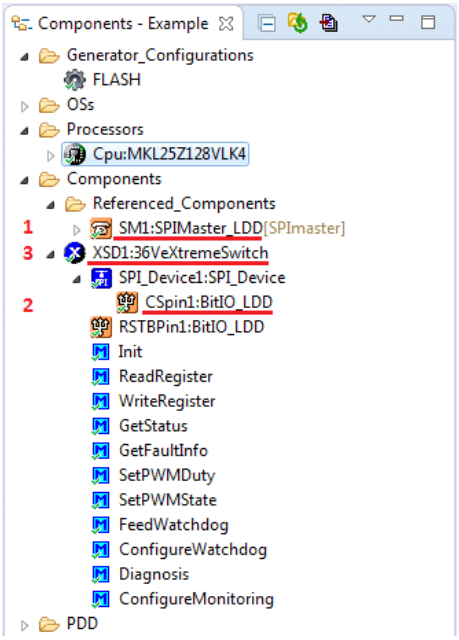

Figure 12. Setting up the components

4.5 Generating Driver Source Code

After the components are configured, the next step is to generate the driver code that will be incorporated into the user application. The process is as follows:

1. Click on the **Generate Processor Expert Code** icon in the upper right corner of the **Components** panel.

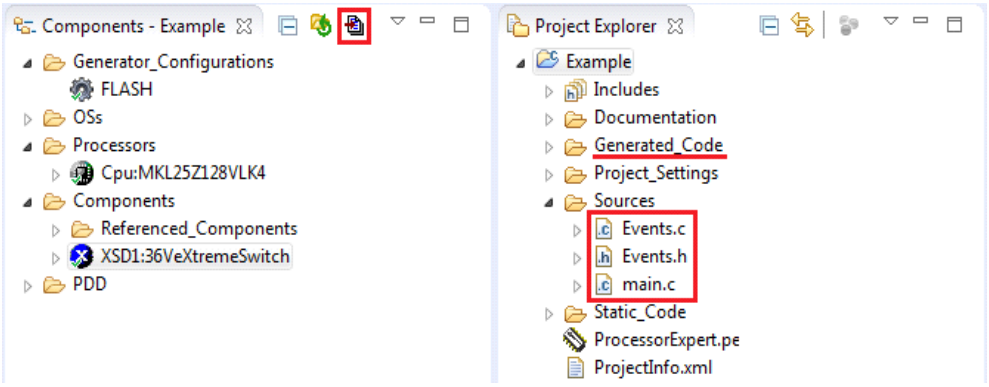

Figure 13. Generating the source code and the code location

The driver code for the eXtreme Switch 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. Figure 13 shows the locations of the generated driver source and the application code.

4.6 Writing Application Code

All application code must reside in the Sources folder in the user's project directory. The user may modify the code in **main.c** and **Events.c**, but must retain the original comments related to usage directions.

To add a component method into the application source code:

1. In the **Components** view for the project, click on **Components** folder and select a component. Find the method to be added 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, the user can open the 36VeXtremeSwitch component method list, drag and drop **SetPWMDuty** to **main.c** and add the necessary parameters (see Figure 14).

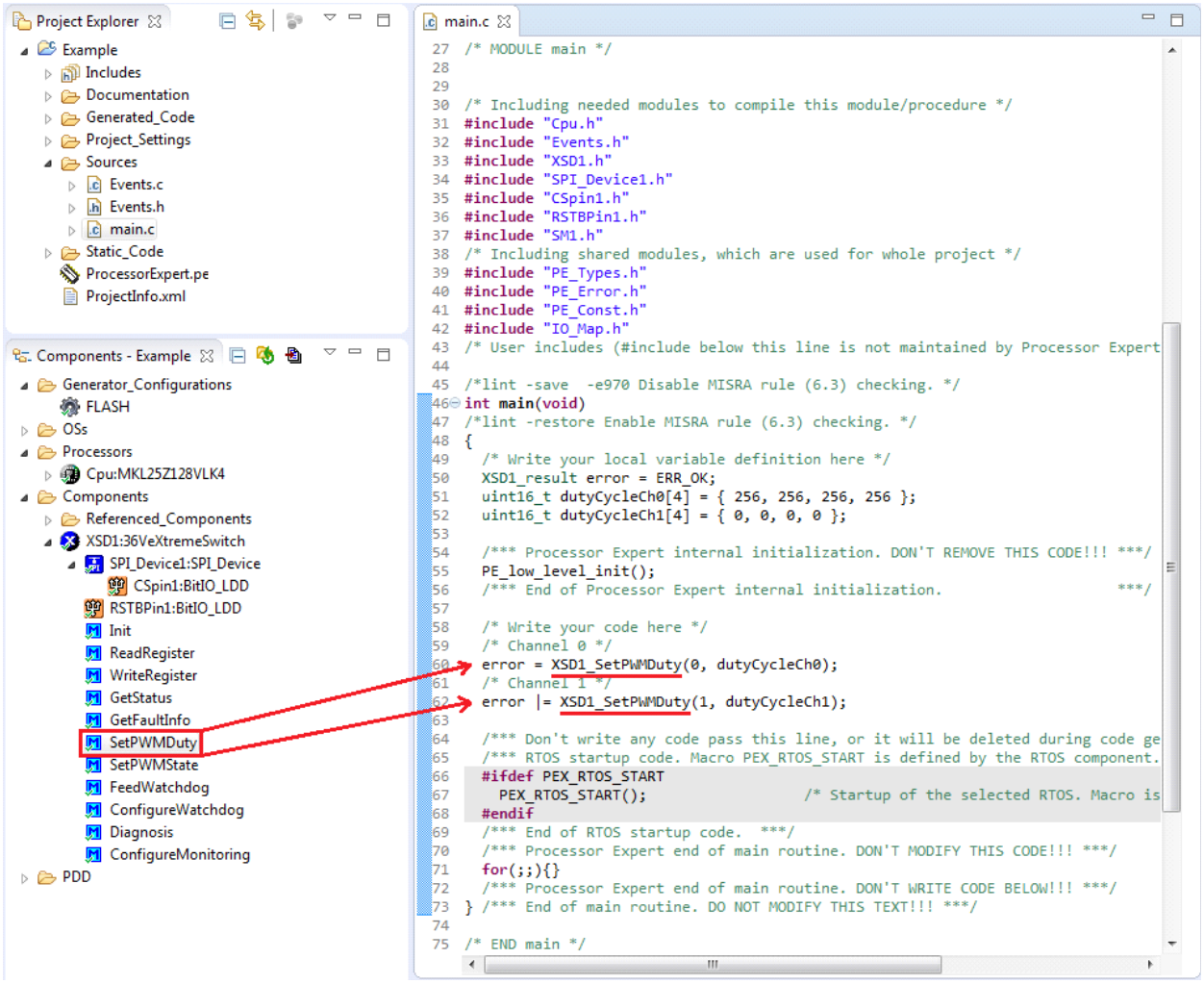

Figure 14. Adding component methods

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

36VeXtremeSwitch component encompasses a help, which describes component properties, methods and typical usage. To show the help, do the following:

- In the **Components** view, right click the **36VeXtremeSwitch** component and select **Help on Component**.

A web page displays with the Help information.

4.7 Compiling, Downloading and Debugging

To compile a project, click the compile icon in the toolbar (see Figure 15).

The process for downloading an application onto a board in Kinetis Design Studio may differ according to the type of MCU board being used. For more information, see the Kinetis Design Studio user's guide. To download and debug on a FRDM-KL25Z MCU board, do the following:

1. Click the arrow next to the debug icon in the toolbar and select **Debug Configurations…** (see Figure 15).

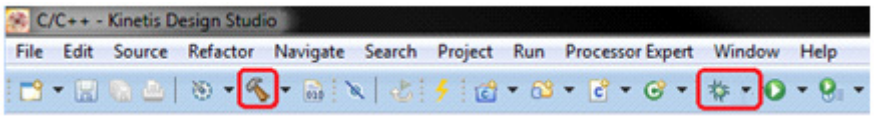

Figure 15. Compile and debug

2. In the **Debug Configurations** dialog box, click **Example_Debug_PNE** under **GDB PEMicro Interface Debugging** (see Figure 16).
3. Make sure that **C/C++ Application** contains the path to the .elf file of the project (see Figure 16).

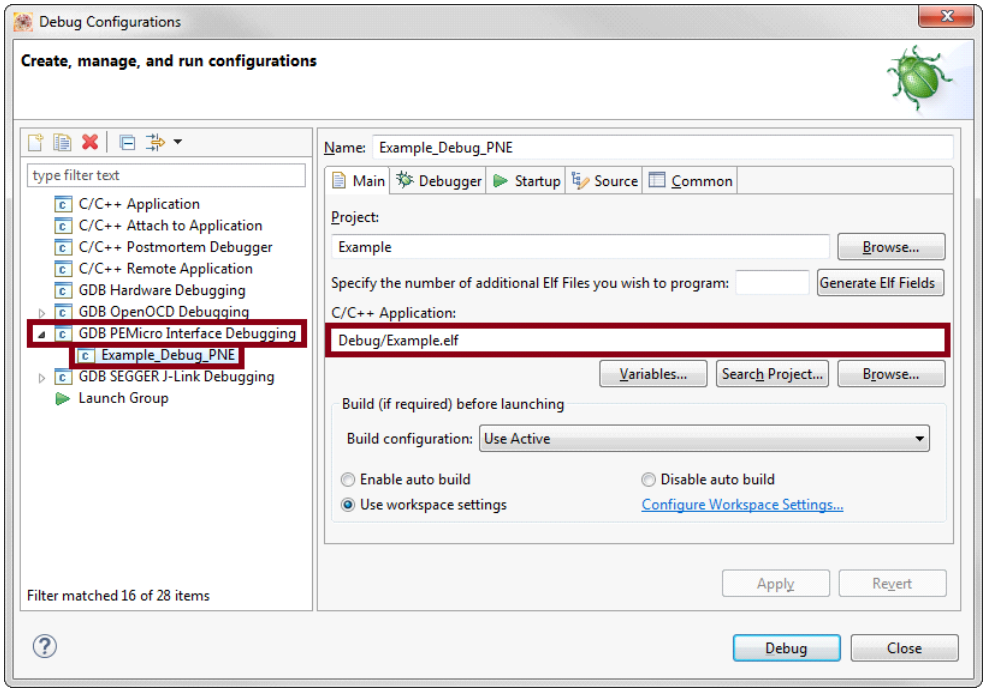

Figure 16. Downloading the application (a)

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 17).
5. Make sure that 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 by clicking the **Select** button.
6. Click **Debug**. Kinetis Design Studio will download and launch the program onto the board.

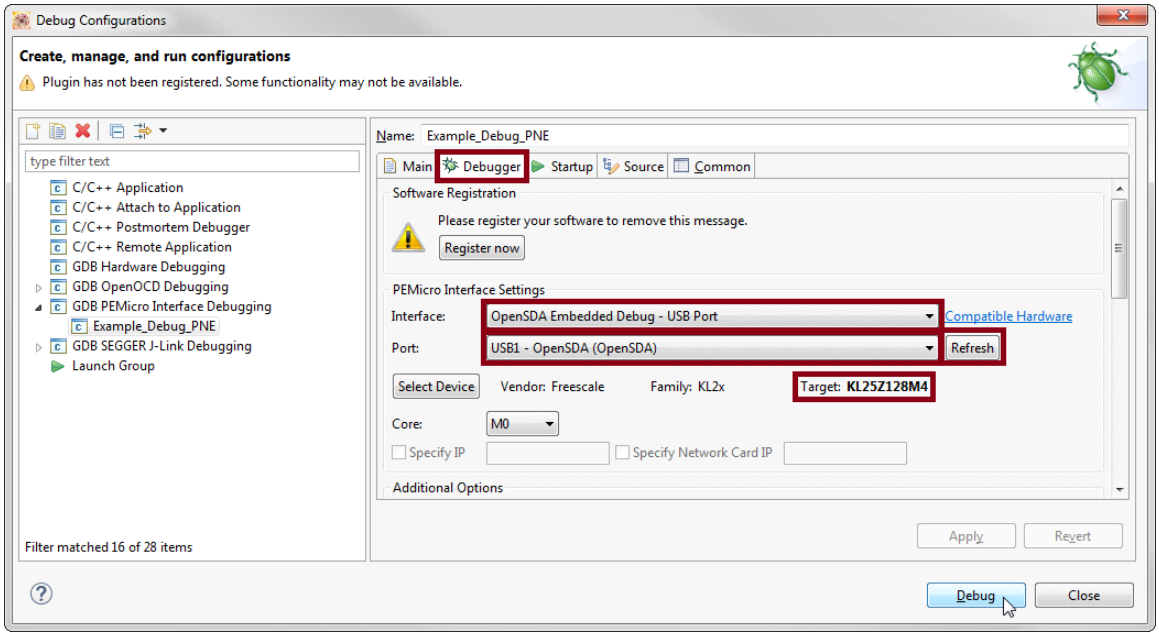

Figure 17. Downloading the application (b)

5 References

The following are URLs for additional information related to NXP products and application solutions:

Table 6. References

NXP.com Support Pages	Description	URL
MC36XSD	Product Summary Page	www.nxp.com/MC36XSD
TWR-MC36XSDEVB	Tool Summary Page	www.nxp.com/TWR-MC36XSDEVB
FRDM-MC36XSD-EVB	Tool Summary Page	www.nxp.com/FRDM-MC36XSD-EVB
Tower System	Tower System Modular Development Board Platform	www.nxp.com/tower
Kinetis Design Studio	Software	www.nxp.com/KDS
CodeWarrior	Software	www.nxp.com/codewarrior
Processor Expert Code Model	Code Walkthrough Video	www.nxp.com/video/processor-expert-code-model-codewarrior-code-walkthrough:PROEXPCODMODCW_VID

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	2/2016	• Initial release
2.0	10/2016	• Added support for FRDM-MC36XSD-EVB • In the Supported products table (Table 2) – Added rows for FRDM-64F and FRDM-KL25Z – Deleted references to CodeWarrior support – Added KDS support for all listed boards • Replaced jumper setting references to switch setting references • Added section on FRDM-MC36XSD-EVB settings • Added new version of Section 4, "Installing the Processor Expert software" • In Section 5, "References" – Added references to FRDM-MC36XSD-EVB – Removed references to CodeWarrior

RoHS

How to Reach Us:

Home Page:
NXP.com

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

Document Number: PEXMC36XSDSWUG
Rev. 2.0
10/2016

NXP