GETTING STARTED WITH THE SILICON LABS PRECISION32™ APPBUILDER

1. Introduction

Precision32™ AppBuilder is a software program that generates configuration code for MCU devices. The software dynamically generates configuration code for peripherals, clocks, and pins in user-definable modes, greatly simplifying the device initialization process. This application note overviews the AppBuilder software for the Silicon Labs 32-bit microcontrollers.

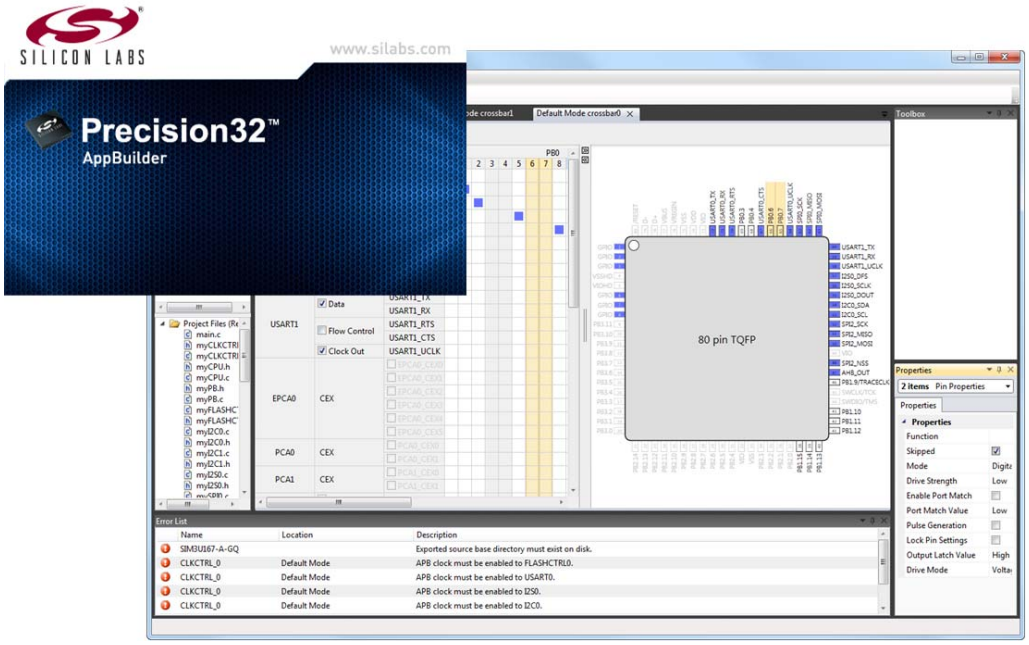


Figure 1. Precision32 AppBuilder

2. Key Points

- Creating and navigating a new project
- Configuring peripherals, clocks, pins, and modes
- Exporting a project

3. Relevant Documentation

Precision32 Application Notes are listed on the following website: www.silabs.com/32bit-mcu.

- AN663: Precision32™ MCU Family Clocking Options
- AN664: Precision32™ CMSIS and HAL User's Guide
- AN667: Getting Started with the Silicon Labs Precision32™ IDE
- AN671: Precision32™ Port I/O Crossbar Decoder

Copyright © 2012 by Silicon Laboratories

4. Creating a New Project

1. Start a new project by selecting **New Project…** from the File menu, or click the **New Project…** link on the Start Page. This will display a **New Project** window.

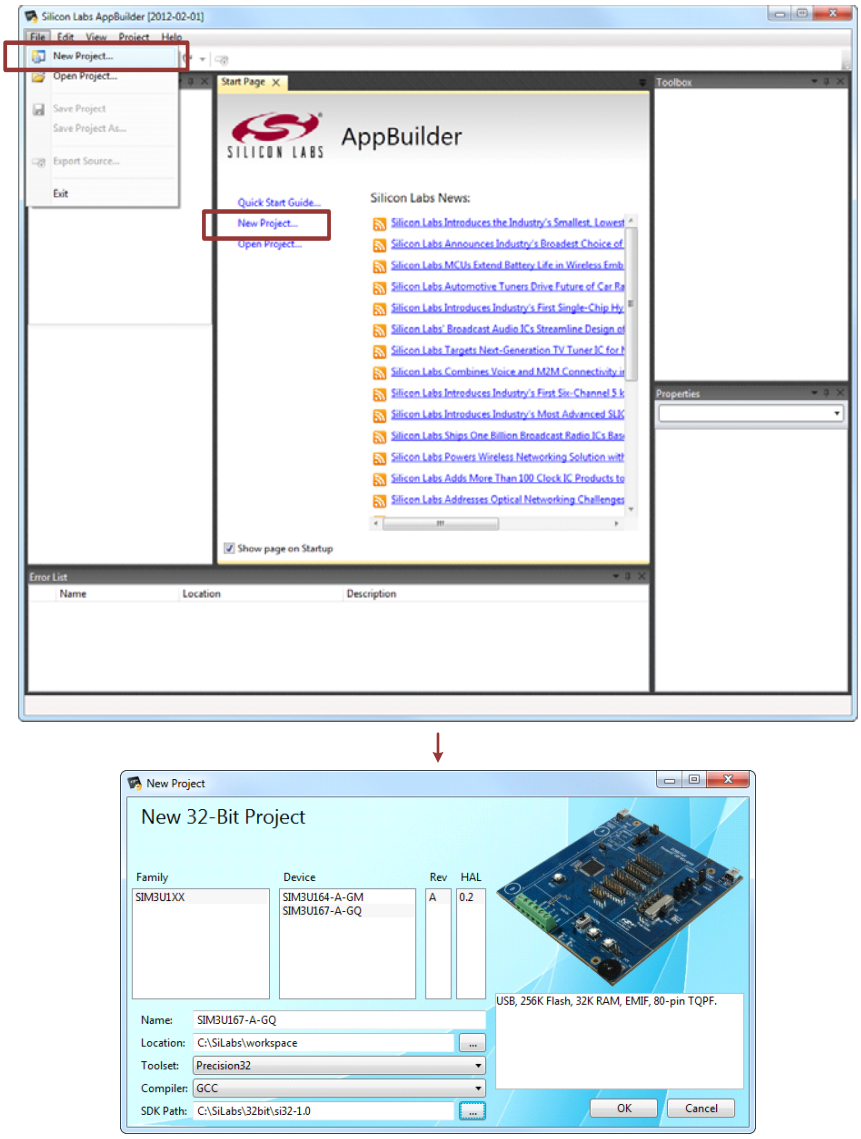


Figure 2. Creating a New Project

2. In the **New Project** window, select the Family, Device, Revision, and SDK version. Each selection will reveal all other possible selections.
   a. The Revision selection is the MCU device revision.
   b. The SDK version determines the functions used by AppBuilder. This selection should match the SDK base path.
3. Name the project and select the location where the generated source will be exported.
4. Select the toolset (IDE) and compiler.
5. Select the Silicon Labs 32-bit SDK base path (**C:\SiLabs\32bit\si32-x.y** by default). This selection should match the SDK version.
6. Click OK.

5. Migrating a Project

AppBuilder includes a project migration feature to update a created project with a new device family, new device within the same family, device revision, or SDK version. This allows existing and complicated projects to be easily ported without redoing all the peripheral, clock, and pin configurations manually.

To migrate a project:

1. Go to the **Project** menu and select **Migrate Project...**.

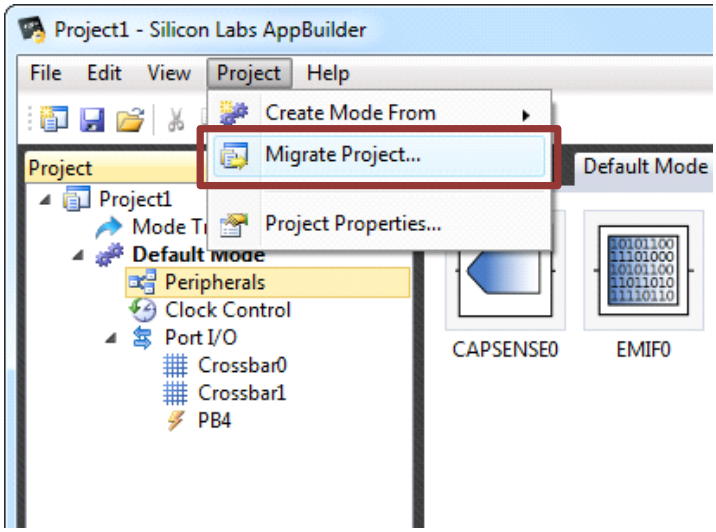


Figure 3. Migrating a Project

2. In the **Migrate Project** dialog, select the new settings. The **Change family, device, or rev** checkbox must be checked to change the hardware platform. If the hardware platform changes, the project should be generated to a new location to prevent any build errors. The **Show Report** and **Create Backup** options are enabled by default.

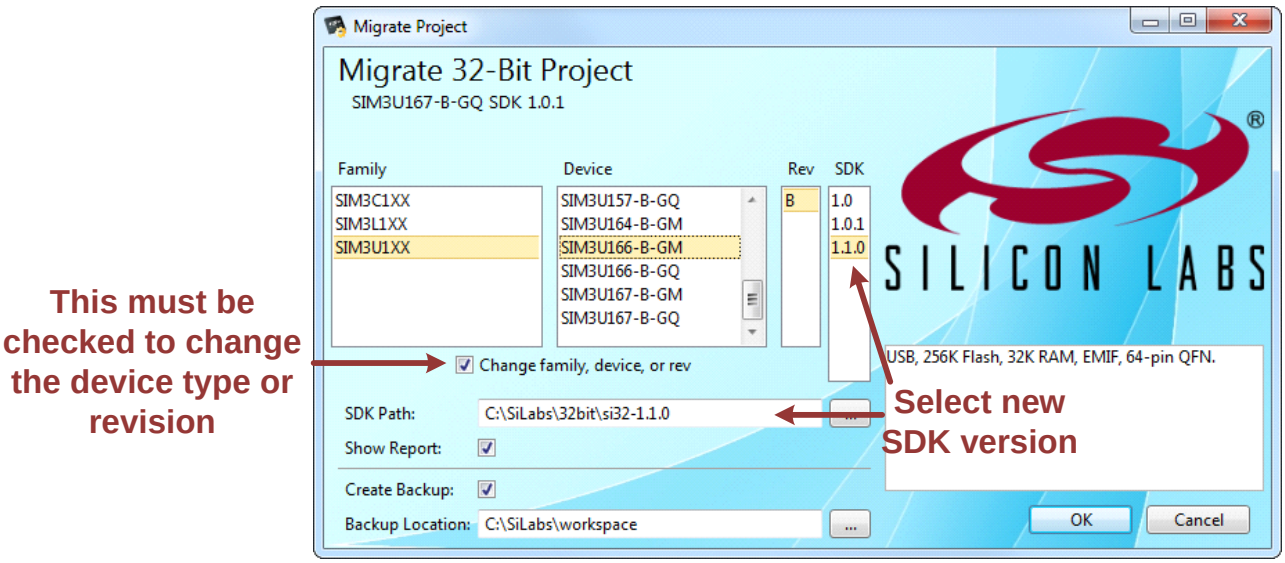


Figure 4. Migrate Project Options

3. Press **OK**.
4. If the **Show Report** option remains enabled, AppBuilder generates a report of the migration results. This report contains a summary of the migration process and any Errors, Warnings, or Messages that resulted from the operation.

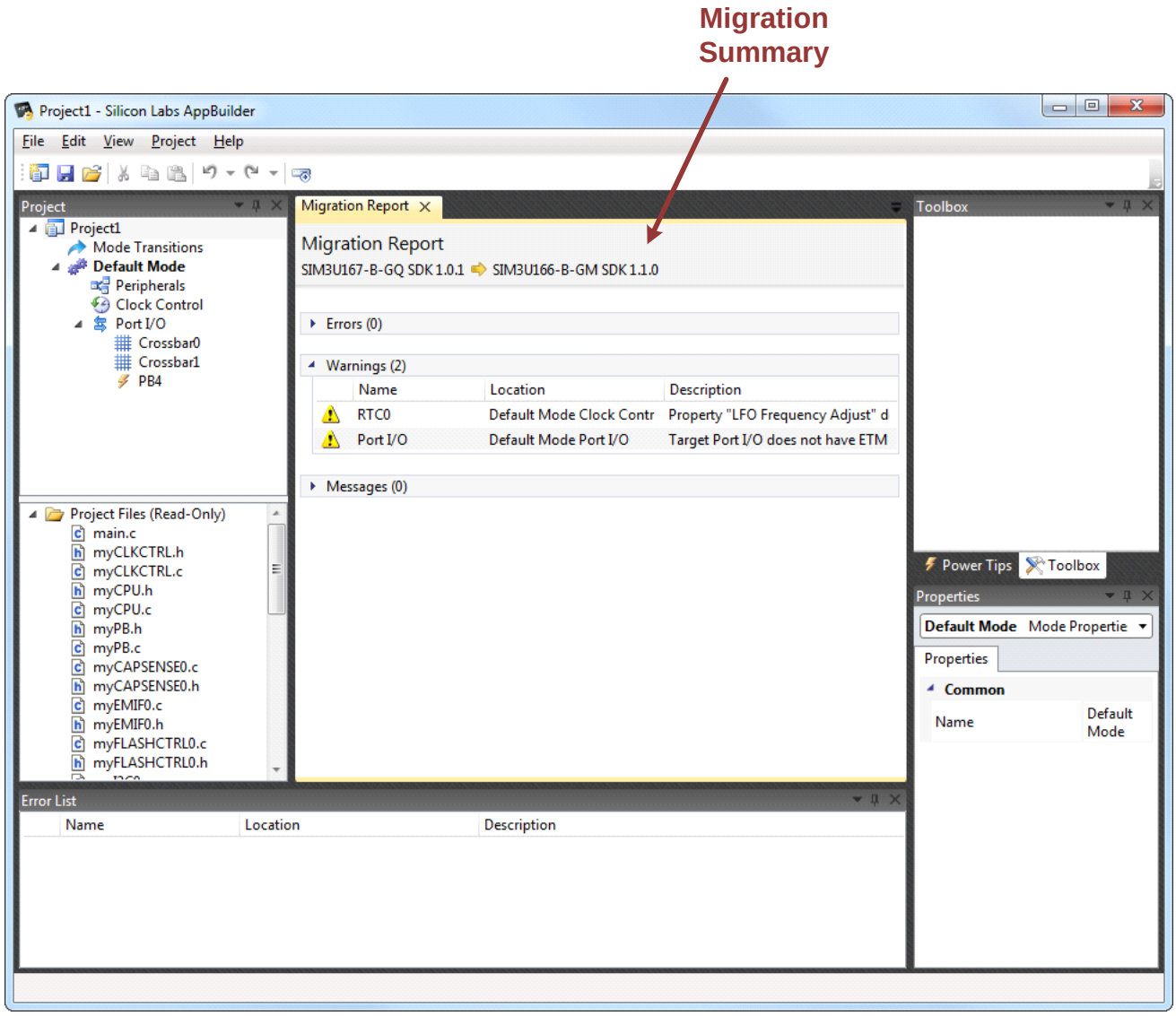


Figure 5. Project Migration Report

6. Navigating in AppBuilder

Figure 6 shows the basic AppBuilder components: Project window, Source Layout, Error and Warning list, Canvas, Toolbox window, Power Tips, and Properties window.

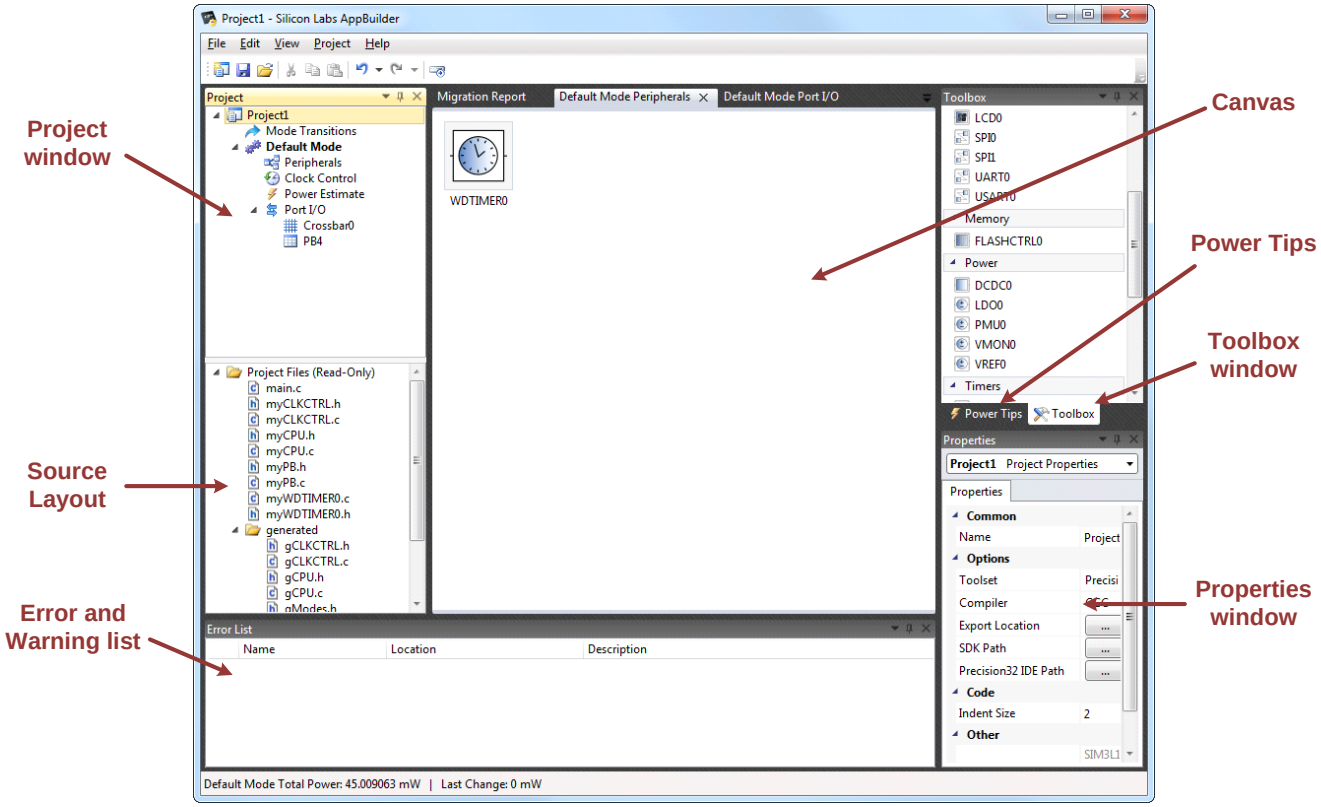


Figure 6. AppBuilder Project Components

6.1. Project Window

The **Project Window** contains the project modes. Each mode has a set of peripherals, clock settings, and port I/O configurations.

Clicking on the part number at the top of the window will open the project properties configured in the **New Project** window and in the **Properties** window.

6.2. Source Layout

The **Source Layout** window shows the current files included in the project. Each of the files can be opened to view their current state.

Note: The source files in the **Source Layout** window are read-only and do not exist on disk until the project is exported.

6.3. Error and Warning List

The **Error and Warning** list provides notifications for any configuration errors flagged by AppBuilder. Double-clicking on an error will automatically open the configuration window that contains the error, and the erroneous setting will be highlighted in red.

6.4. Properties Window

The **Properties** window contains the configuration settings for the peripheral or module currently selected.

6.5. Toolbox Window

The **Toolbox** window is active when the mode canvas is selected. This box contains the peripherals available on the device. To add a peripheral to the mode, drag and drop it to the **Canvas**.

6.6. Canvas

The **Canvas** is the active area of the selected module or mode.

6.7. Power Tips

The Power Tips window shows helpful information on how to configure the selected module to reduce power consumption.

7. Adding and Configuring Peripherals

To add a peripheral to a mode:

1. Double-click the **Peripherals Canvas** from the **Project** window under the desired mode.

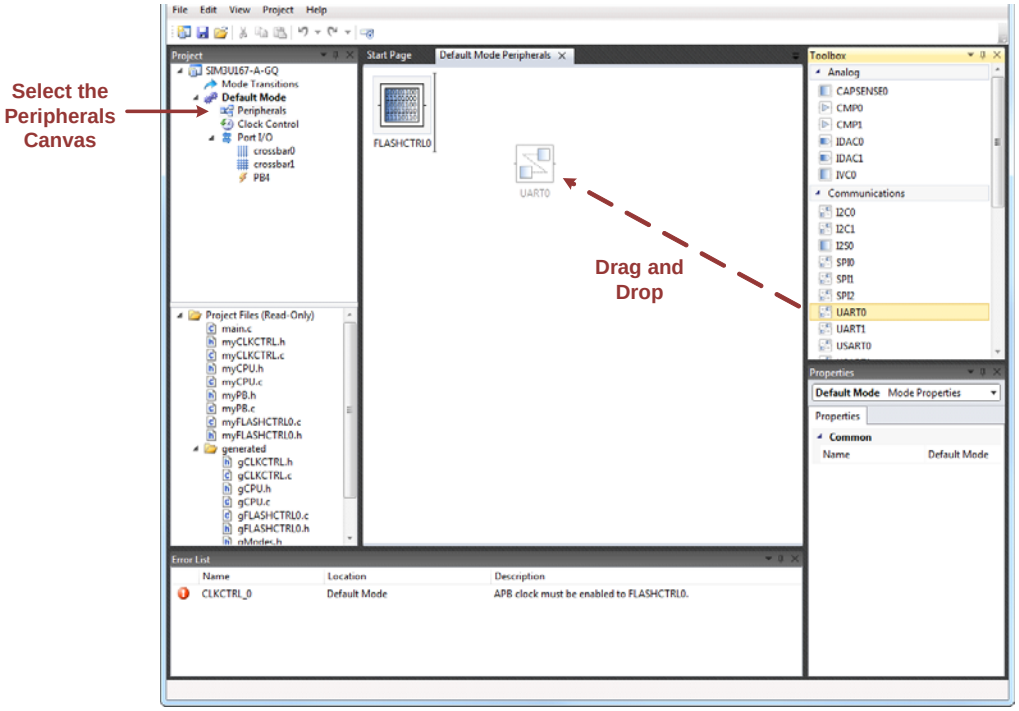


Figure 7. Selecting the Peripherals Canvas

2. Click and drag the desired peripheral from the **Toolbox** window into the **Canvas**.
3. The peripheral will now display in the **Canvas** window, and the corresponding peripheral files automatically add to the **Source Layout** window. When it is clicked and selected, the **Properties** window displays the configurable properties for that peripheral.

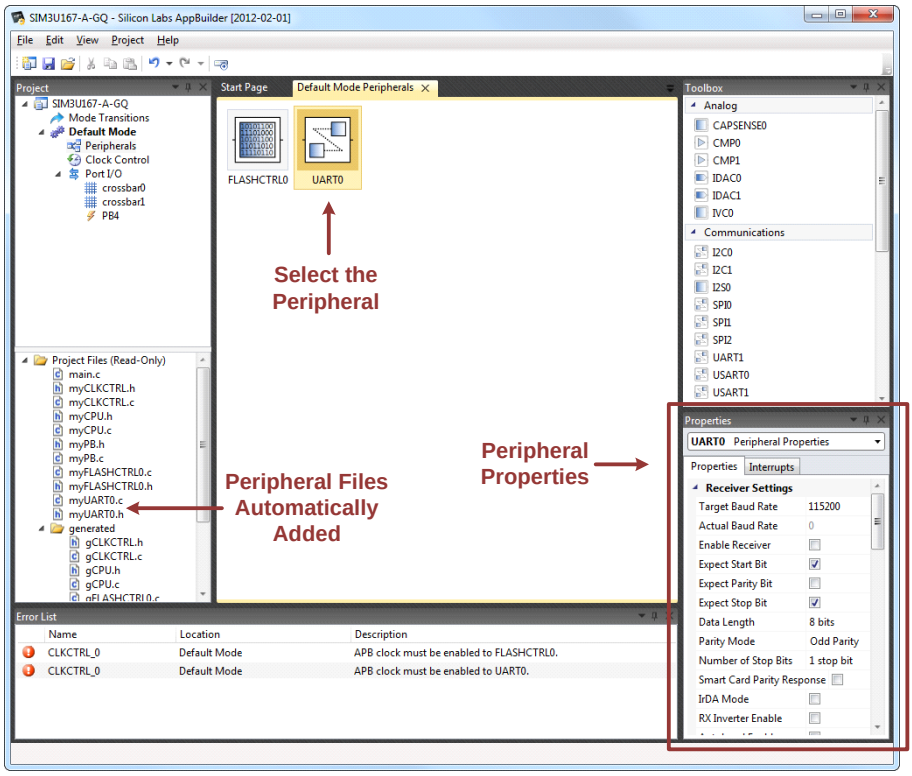


Figure 8. Peripheral Properties

4. The **Error and Warning** list will display any errors in the configuration. Double-clicking on one of the errors in the list will jump to and highlight the property that needs to be corrected.

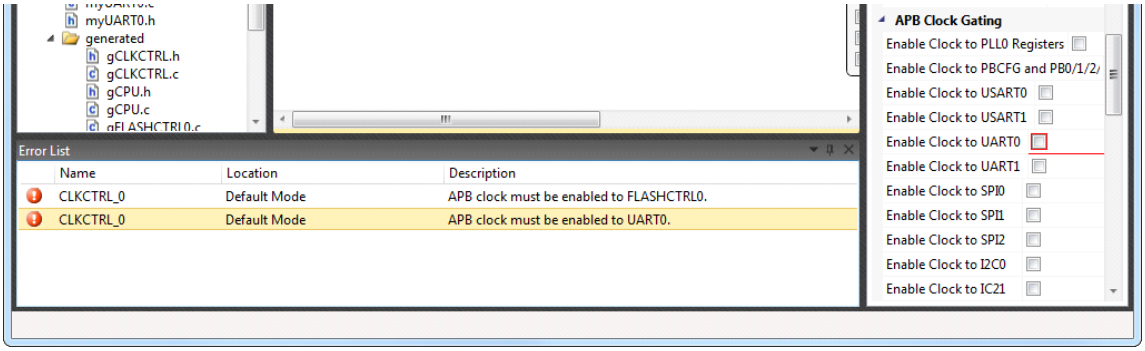


Figure 9. Error and Warning List

8. Configuring the Device Clocks

To configure the clocks in a mode:

1. Double-click on the **Clock Control Canvas** in the **Project** window to configure the clock control module for the selected mode.

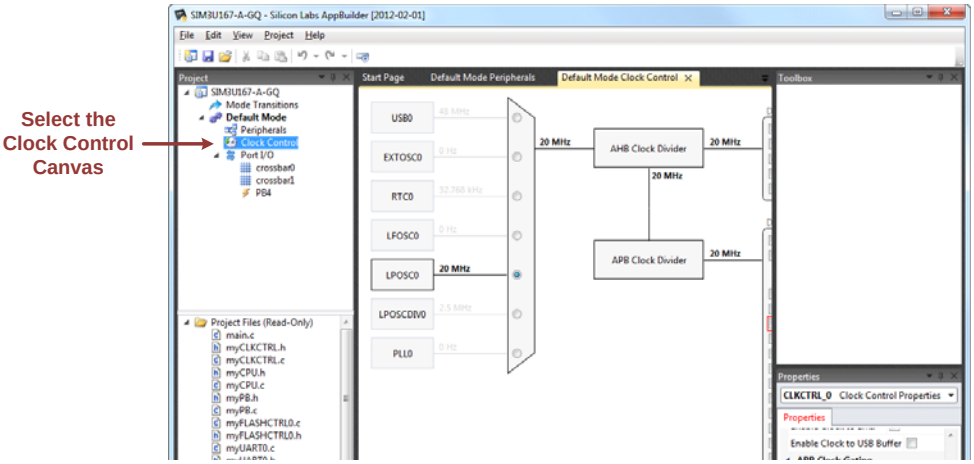


Figure 10. Selecting the Clock Control Canvas

2. Each clock source on the left can be selected to set the configurable properties in the **Properties** window.

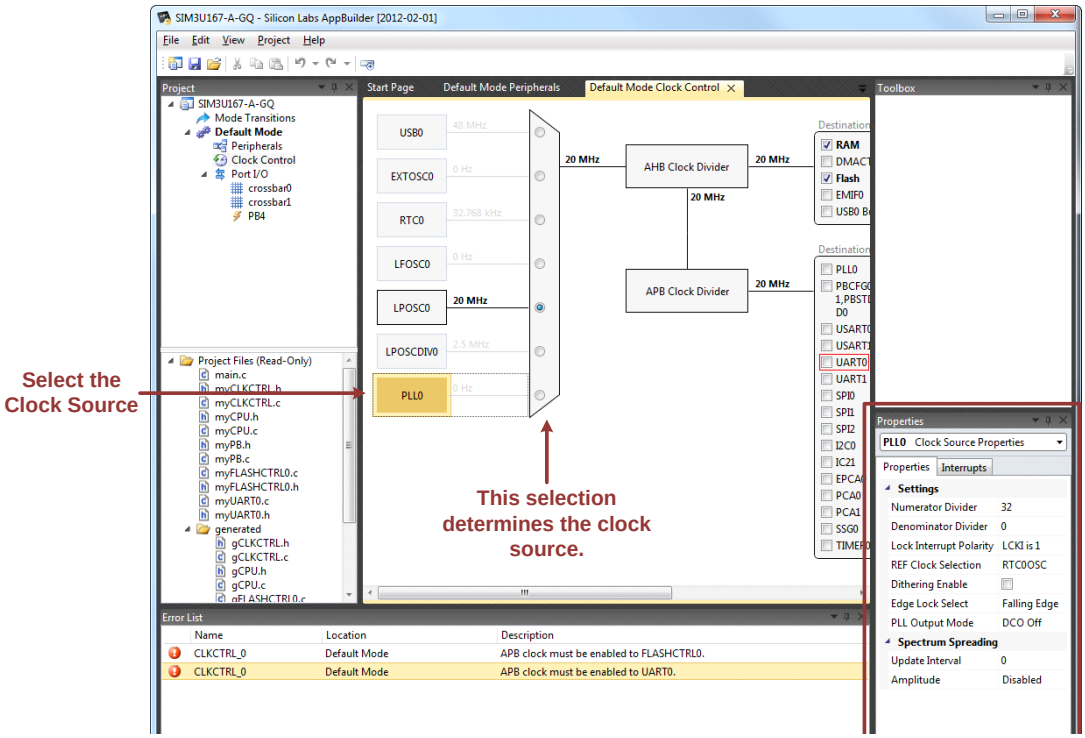


Figure 11. Modifying the Clock Control Canvas

3. The multiplexor selection determines the selected clock source for the system. The clock frequencies shown in the diagram will update automatically based on the selected clock source.
4. Each clock destination on the right can be enabled by clicking the checkbox to enable the clock to the module.

9. Configuring the Pins

There are two types of pins supported by AppBuilder: those on a crossbar, and those not on a crossbar.

9.1. Crossbar Pins

To configure the crossbar in a mode:

1. Double-click on the corresponding **Crossbar Canvas** from the Port I/O section in the **Project** window.

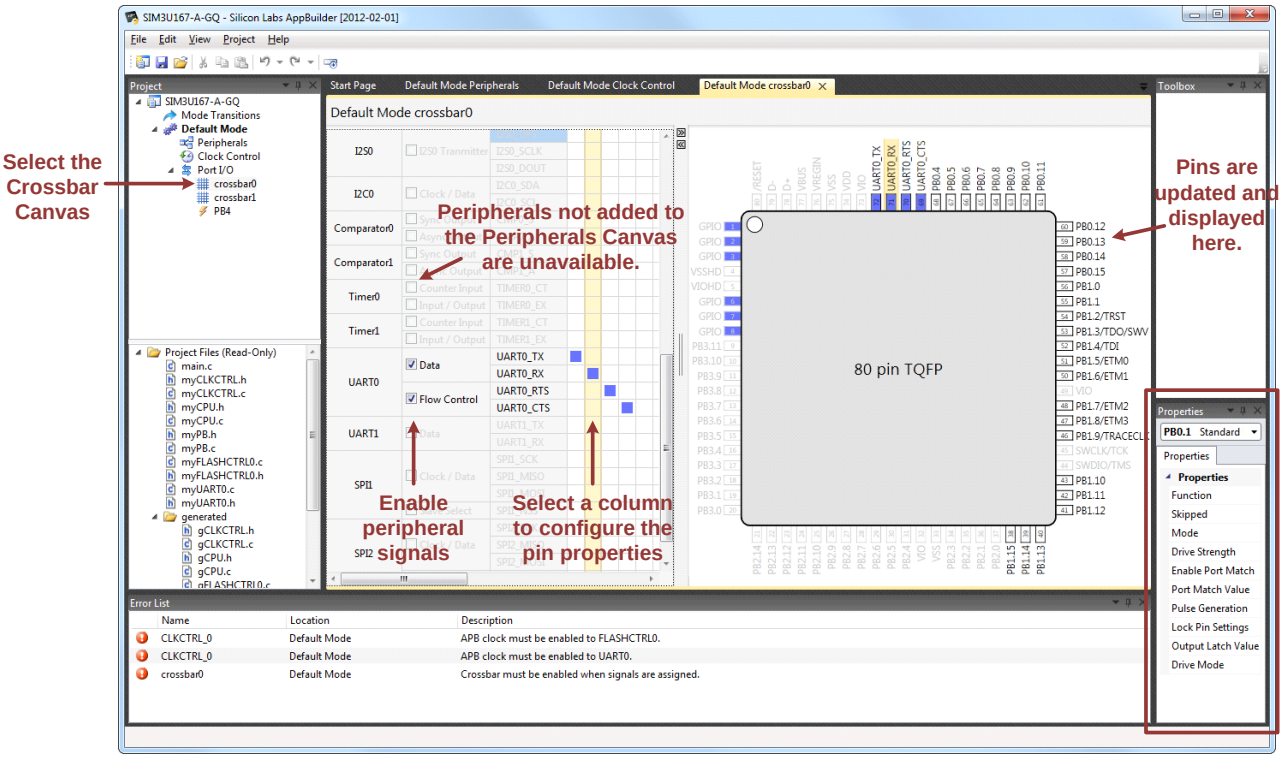


Figure 12. Crossbar Canvas

2. A peripheral is available to add to the crossbar if it's placed on the **Peripherals Canvas**.
3. Click the checkboxes next to the peripheral name to enable the signals on the crossbar.
4. Select a pin column to configure the properties for that pin. Multiple pins can be selected and configured at the same time.
5. The pins will update in the pinout view as they are configured.
6. The pinout and crossbar views can be toggled by using the **<<** and **>>** arrows in the center of the **Crossbar Canvas**.

9.2. Non-crossbar Pins

To configure pins not in a crossbar:

1. Double-click the non-crossbar pin configuration Canvas (PB4 on the SiM3U167 example device).

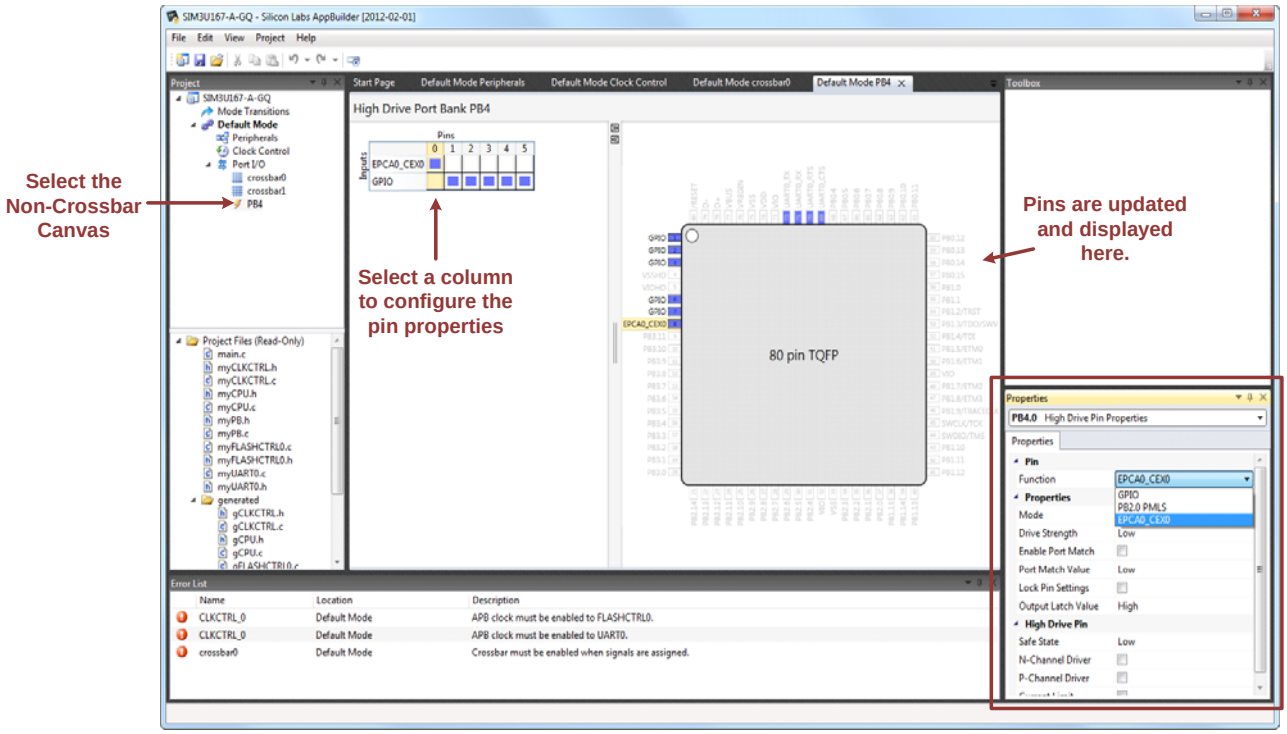


Figure 13. Non-Crossbar Canvas

2. Select a pin column to configure the properties for that pin. Multiple pins can be selected and configured at the same time.
3. Set the pin properties using the **Properties** window.
4. The pins will update in the pinout view as they are configured.
5. The non-crossbar pin grid will expand as pins are assigned to unique functions.

9.3. Printing the Pin Diagram

Once the pinout is complete, right-click on the package area and select **Print...** to print the package drawing with peripheral assignment information.

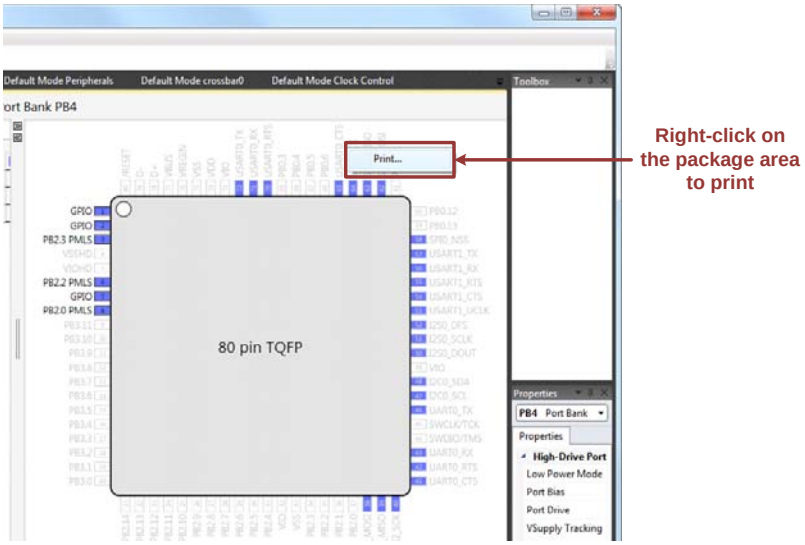


Figure 14. Printing a Pinout

10. Configuring Modes

Modes consist of sets of peripheral, clock, and pin settings. AppBuilder can specify multiple modes and the transitions between them. As an example, a system might contain a normal operating mode and a low-power operating mode. A new project always starts with a single **Default Mode** shown in the **Project** window.

10.1. Adding or Configuring a Mode

To add a mode to the project, right-click on a mode already in the project in the **Project** window. This will allow the mode to be cloned into a new mode. Alternatively, right-clicking on the project in the **Project** window provides options to create a new mode from any mode in the project or a **Reset Mode**, which is a mode with all the peripheral, clock, and pin reset defaults.

Right-clicking on a mode in the **Project** window also provides two additional options: setting the mode as the default after a reset, and removing the mode from the project.

Figure 15 shows the menus to add or configure modes.

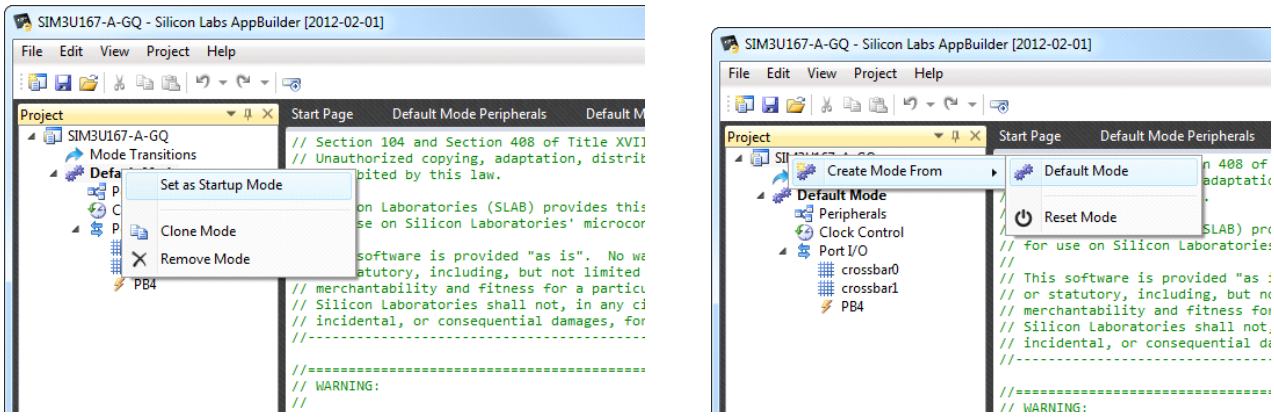


Figure 15. Adding or Configuring a Mode

To rename a mode, left-click on the mode name in the **Project** window and modify the name in the Properties window as shown in Figure 16.

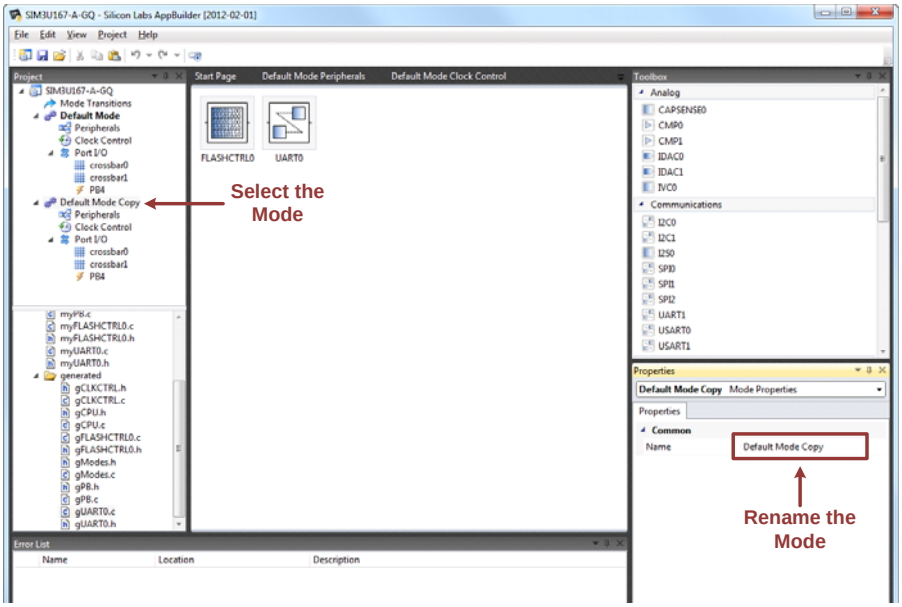


Figure 16. Renaming a Mode

10.2. Defining Mode Transitions

Mode transitions enable a system to move from one user-defined mode to another in an efficient way. Mode transition functions include the minimal amount of code to configure the peripherals from one mode to the next as opposed to regenerating full initialization code.

AppBuilder automatically populates some mode transitions. To define a new mode transition, double-click the **Mode Transitions** option in the **Project** window. Select the mode to transition from and the mode to transition into under **Add Transition**. Then, click the plus (**+**) button.

To remove a transition, click the minus (**–**) button next to the transition.

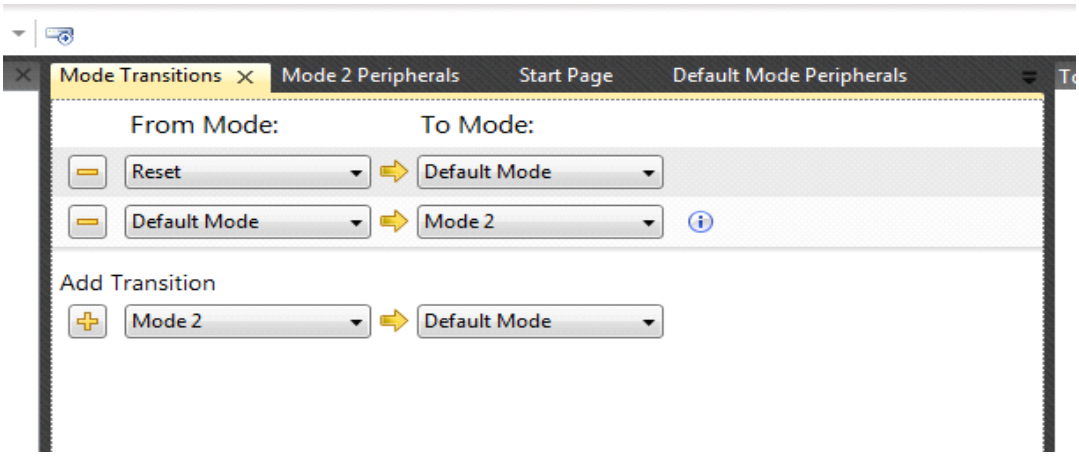


Figure 17. Mode Transitions

Transitions from **Reset** will generate peripheral, clock, and pin configuration code. Transitions between two user-defined modes will generate peripheral transition code only. AppBuilder displays a yellow banner when editing settings for a mode that does not have a transition defined from the **Reset** state as shown in Figure 18. To quickly add a transition from **Reset** to the selected mode, click the **Add the transition** link in the banner.

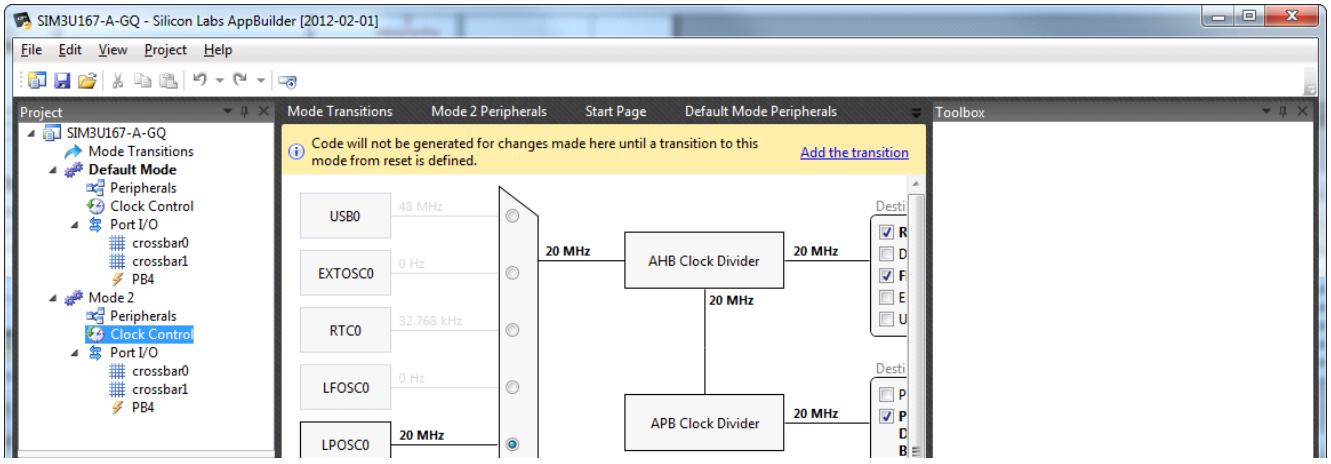


Figure 18. Mode Transition Warning Banner

11. Estimating Power Consumption (SiM3L1xx Devices Only)

The AppBuilder Power Estimator can provide real-time estimates of the SiM3L1xx device power consumption in each mode. This feature is based on a power model developed to estimate power based on dc-dc converter efficiency, LDO output voltage, and enabled peripherals.

11.1. Manipulating the Power Model

Double-click on the corresponding **Power Estimate** for the appropriate mode in the **Project** window to view and modify the power estimate for a mode. Figure 19 shows an example mode Power Estimate.

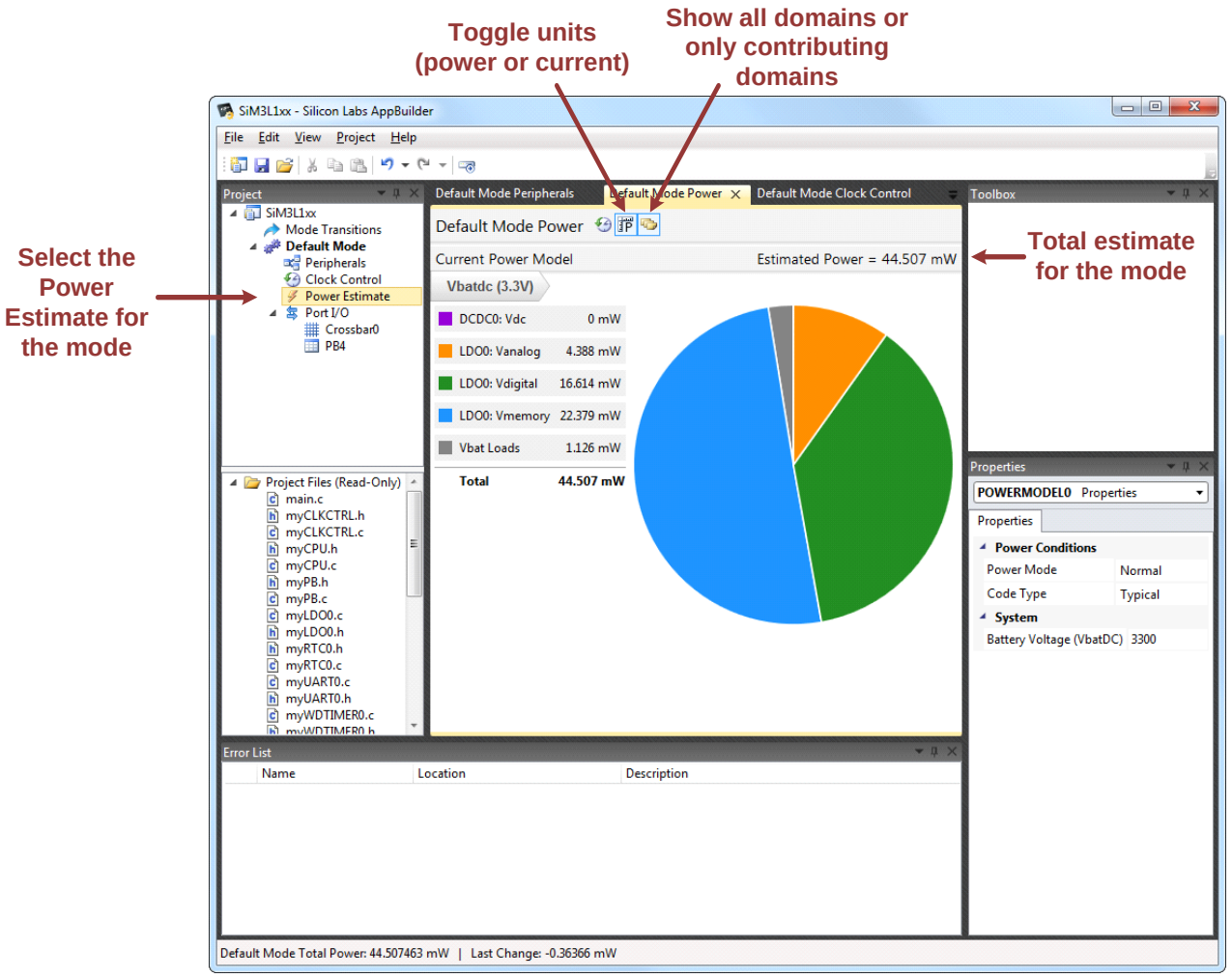


Figure 19. Example Power Estimate

The units are shown in power or current using the toggle button at the top. In addition, the view can switch between all available domains or contributing domains.

Once in the **Power Estimate** view, single-click on a power domain in the list or in the pie chart to show its constituent properties in the **Properties** window.

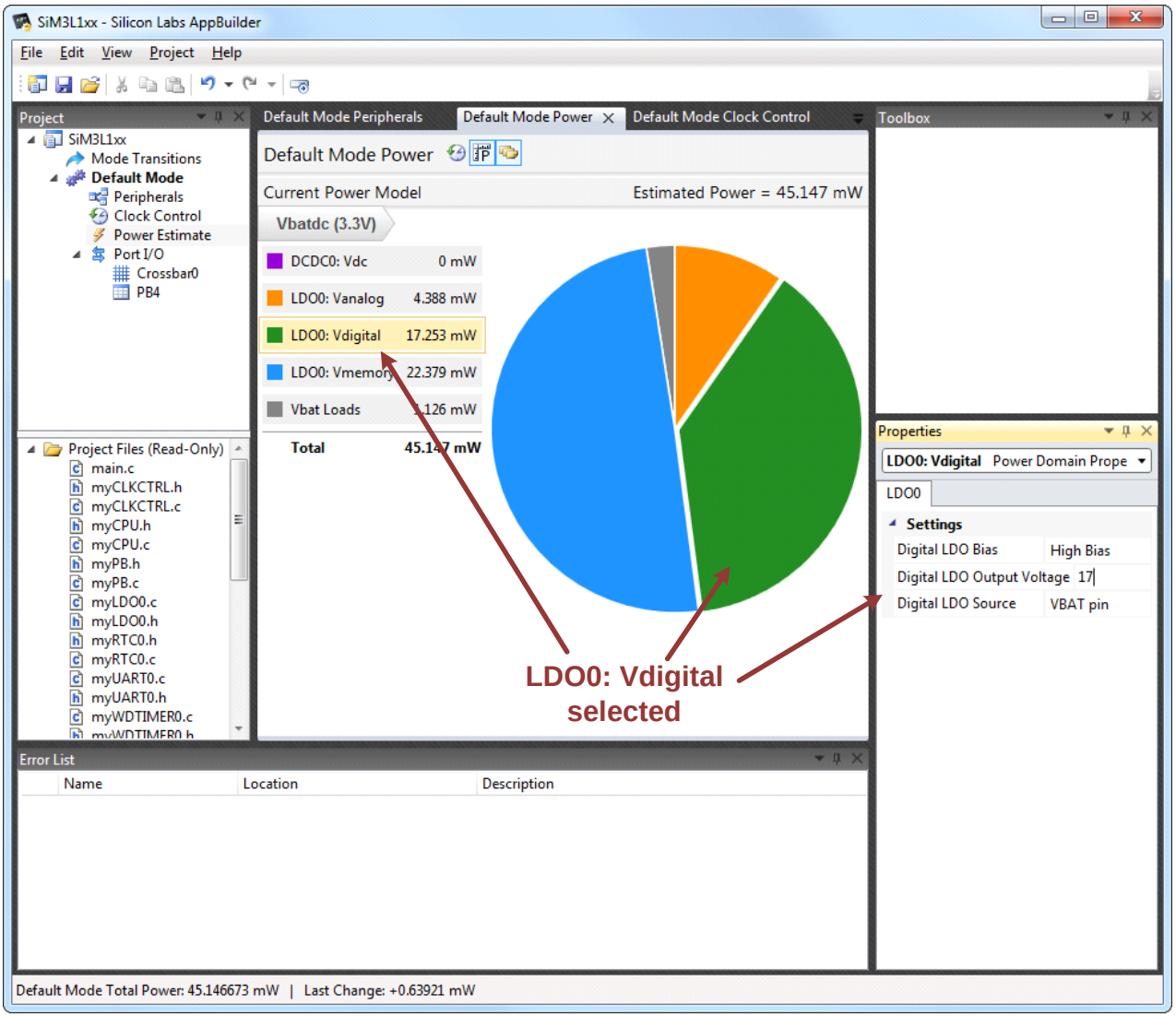


Figure 20. Viewing Power Estimate Properties

Some Properties may be disabled if the peripheral has not been added to the **Peripheral** Canvas or if the peripheral's current configuration does not allow the property to be modified.

For a more detailed view of the selected power domain, double-click on the power domain to step into a lower level of the power model. Click the top-level domain (**Vbatdc**) at the top of the window to return to the top-level power estimate.

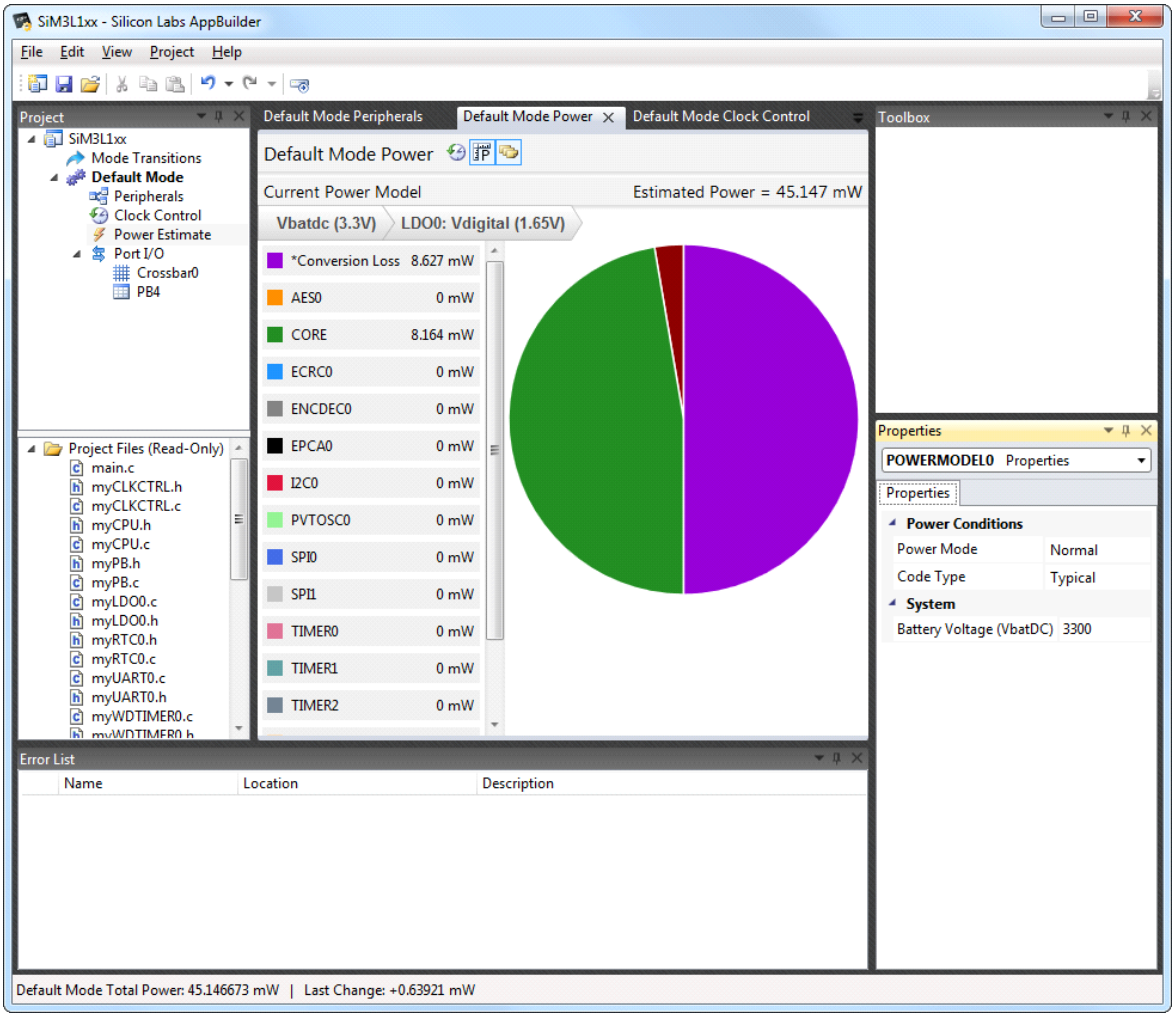


Figure 21. Detailed Power Domain Information

11.2. Comparing Power Estimates

The status bar at the bottom of the AppBuilder window shows the total power consumption for the selected mode as well as the effect (positive or negative) of the last property change on the power estimate.

In addition to estimating the power consumption for the mode's current configuration, AppBuilder can store a previous power estimate to enable quick comparisons of power consumption between groups of property changes.

To see the effects of multiple property changes on the power consumption for a mode:

1. Show the previous power model estimate by clicking the **Show previous power shapshot** button.
2. Press the **Pause updates** button to freeze the previous power estimate.
3. Make any necessary property changes.

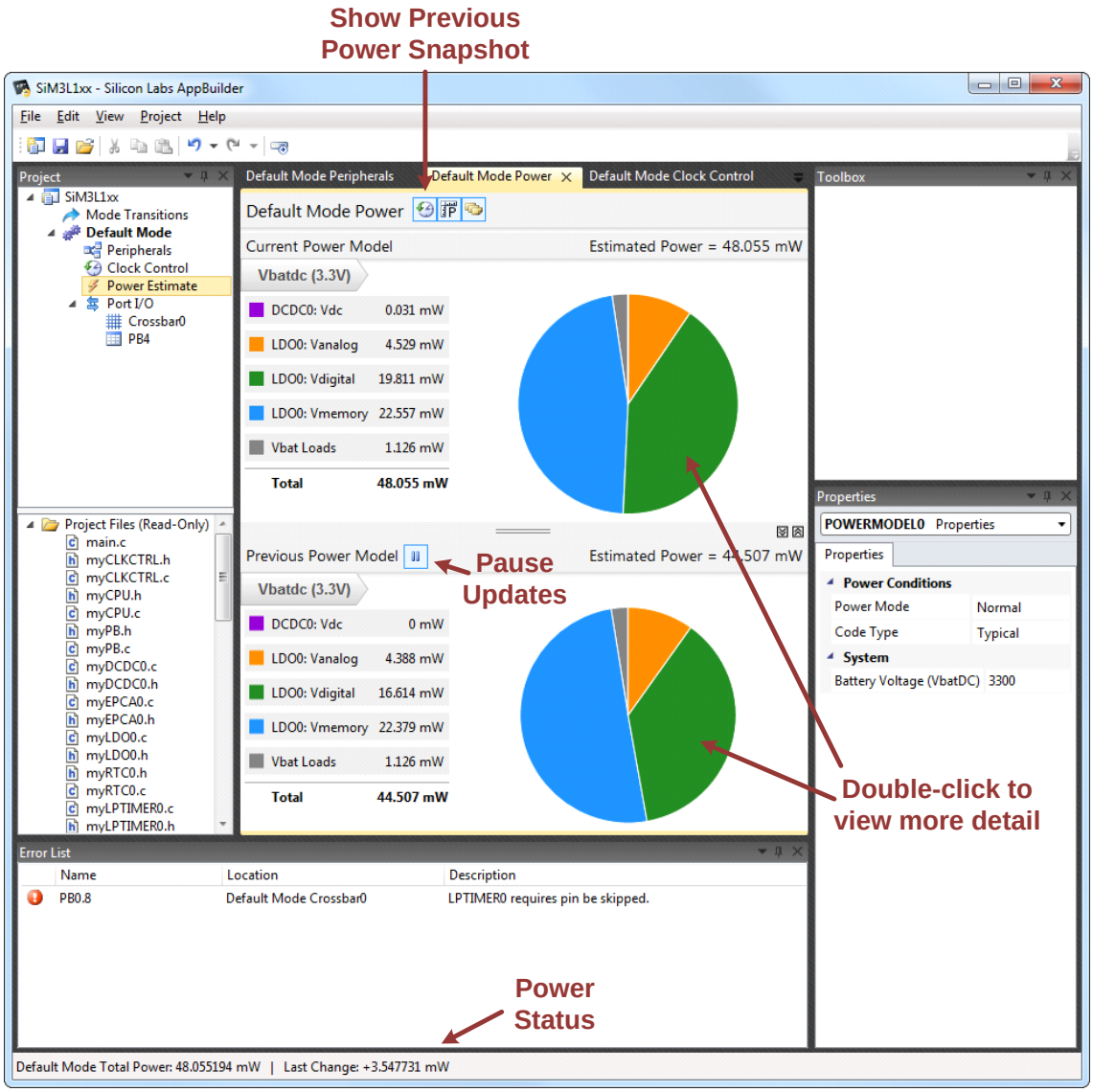


Figure 22. Comparing Power Estimates

As properties and configurations change, the current power estimate will continue updating, while the previous estimate will stay frozen, allowing for in-depth comparison of power domains. The previous power estimate can still be interacted with, but properties displayed in the previous power estimate cannot be modified.

12. Viewing Source Code

The generated source code can be viewed at any time during the configuration process by double-clicking on a file in the **Source Layout** window. The source files in the **Source Layout** window are read-only and do not exist on disk until the project is exported.

Double-click a file in the **Source Layout** window. Generated files will be in the generated folder. This source will dynamically change as properties and settings change during configuration.

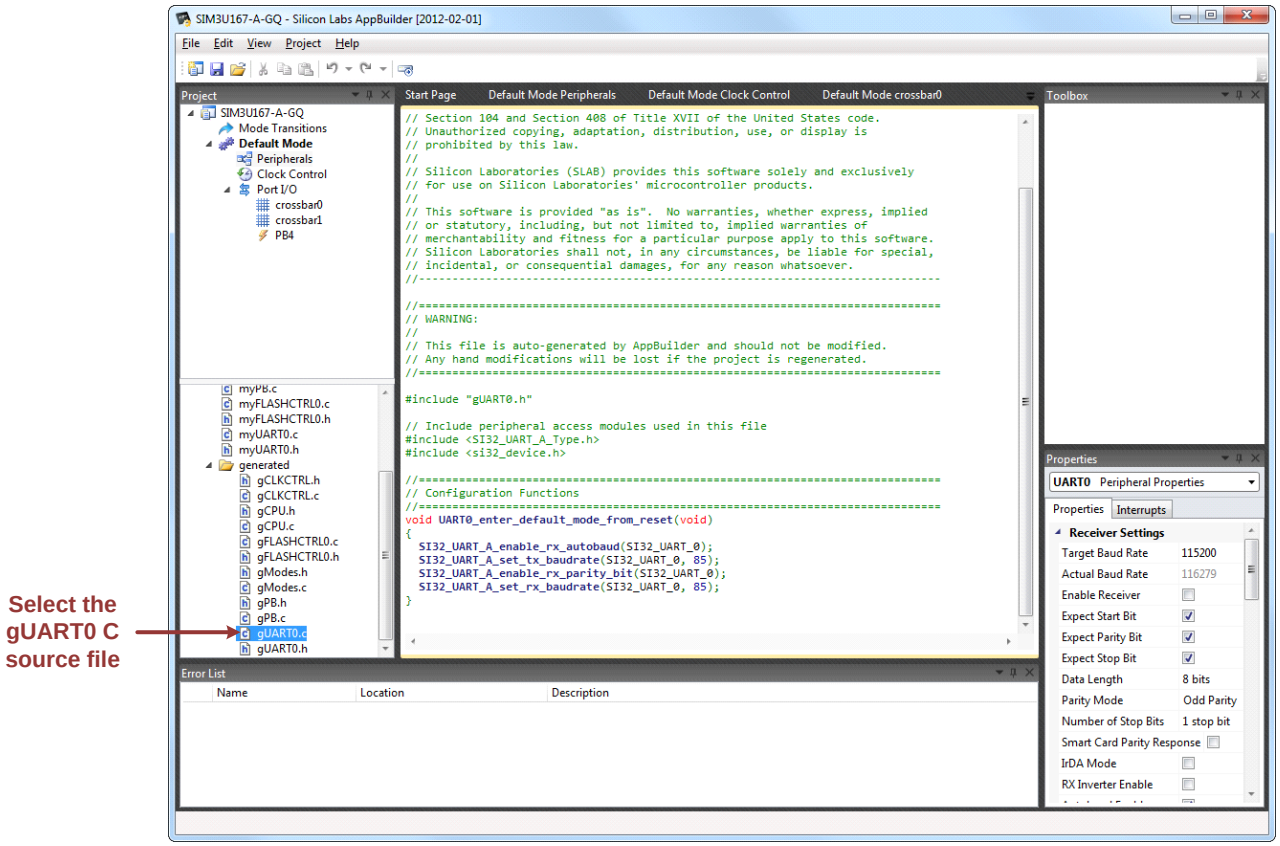


Figure 23. Generated Source Code

13. Generating a Project for Development and Debugging

When configuration is complete, generate a project by selecting **Export Source...** from the file menu or by clicking the **Export Source** button in the toolbar.

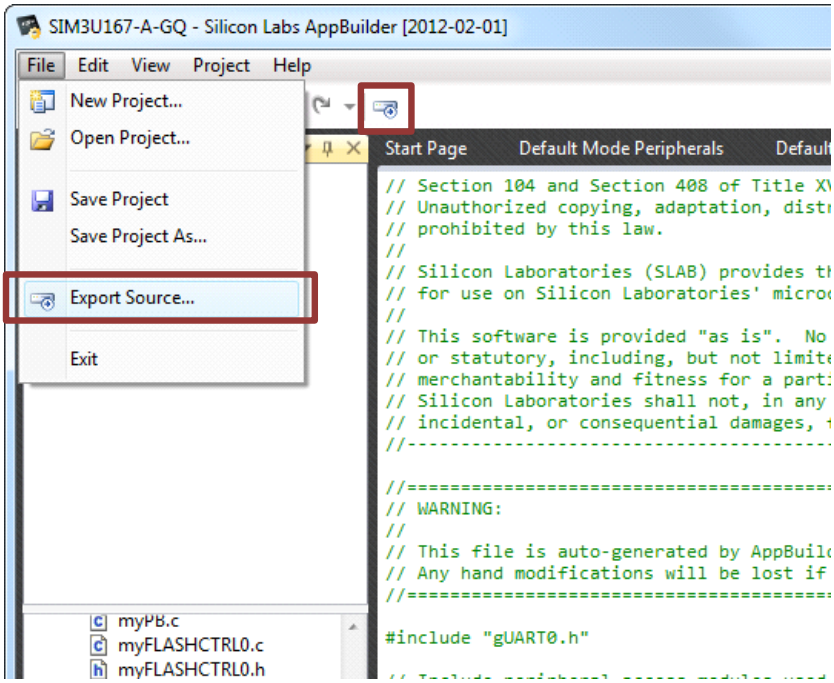


Figure 24. Exporting a Project

This will open the **Export Source** dialog shown in Figure 25.

If the workspace or project does not already exist, AppBuilder will create the workspace and all **my-** and **g-** files. If the workspace or project already exists, AppBuilder will regenerate the **g-** files but leave all existing **my-** files alone. AppBuilder expects that **my-** files are edited by the user and does not overwrite them. AppBuilder will indicate any potential problems with the selected workspace or project and the export file status in the expanded source tree.

Selecting the **Open After Export** checkbox will cause AppBuilder to launch the selected IDE. Clicking the OK button will export the project to the location specified in the project properties.

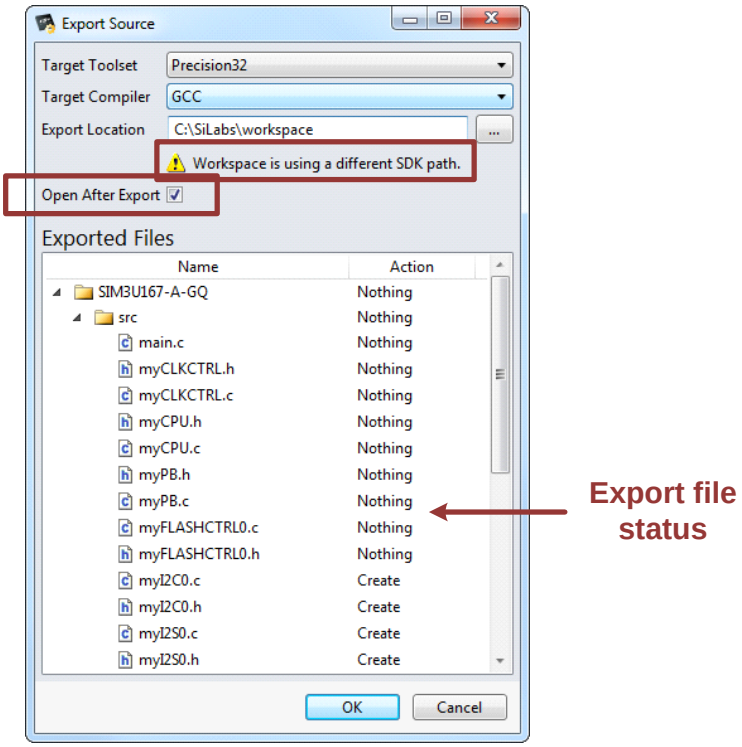


Figure 25. Export Project Dialog

Document Change List

Revision 0.1 to Revision 0.2

- Added descriptions for the Project Migration and Power Estimator features.
- Updated "6. Navigating in AppBuilder" on page 5 and Figure 6 to include Power Tips.

NOTES:

Contact Information

Silicon Laboratories Inc.

400 West Cesar Chavez
Austin, TX 78701
Tel: 1+(512) 416-8500
Fax: 1+(512) 416-9669
Toll Free: 1+(877) 444-3032

Please visit the Silicon Labs Technical Support web page:
https://www.silabs.com/support/pages/contacttechnicalsupport.aspx
and register to submit a technical support request.

The information in this document is believed to be accurate in all respects at the time of publication but is subject to change without notice. Silicon Laboratories assumes no responsibility for errors and omissions, and disclaims responsibility for any consequences resulting from the use of information included herein. Additionally, Silicon Laboratories assumes no responsibility for the functioning of undescribed features or parameters. Silicon Laboratories reserves the right to make changes without further notice. Silicon Laboratories makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does Silicon Laboratories assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. Silicon Laboratories products are not designed, intended, or authorized for use in applications intended to support or sustain life, or for any other application in which the failure of the Silicon Laboratories product could create a situation where personal injury or death may occur. Should Buyer purchase or use Silicon Laboratories products for any such unintended or unauthorized application, Buyer shall indemnify and hold Silicon Laboratories harmless against all claims and damages.

Silicon Laboratories and Silicon Labs are trademarks of Silicon Laboratories Inc.
Other products or brandnames mentioned herein are trademarks or registered trademarks of their respective holders.