CYPRESS
Embedded in Tomorrow™

CY3215-DK

PSoC® 1 In-Circuit Emulator Development Kit Guide

Doc. # 001-66514 Rev. *E

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709

Phone (USA): 800.858.1810
Phone (Intnl): 408.943.2600
http://www.cypress.com

Copyrights

Contents

1. Introduction

Thank you for your interest in the CY3215-DK PSoC® 1 In-Circuit Emulation (ICE) Development Kit (DVK) guide. You can use this kit with PSoC Designer™ or PSoC Programmer. The CY3215-DK provides debugging functionality for PSoC 1 and manages all the emulation communication between the debugger software running on the computer (PSoC Designer) and the target PSoC chip.

PSoC Designer is PSoC 1's GUI-based integrated design environment (IDE), which includes a powerful set of tools for developing code, prototyping, and debugging. This document gives an overview of the hardware configurations that can be used to set up a debugging system for PSoC 1 using the CY3215-DK kit. The Code Examples on page 26 demonstrate the use of the critical software elements of the debugging system.

The CY3215-DK kit supports the following PSoC 1 families.

- CY8C20x34
- CY8C20xx6A
- CY8C21x23
- CY8C21x34
- CY8C22xxx/CY8C21x45
- CY8C23x33
- CY8C24x23A
- CY8C24x94
- CY8C27x43
- CY8C28xxx
- CY8C29x66
- CY8C95xx

1.1 Kit Contents

The CY3215-DK kit includes the following:

- ICE-Cube in-circuit emulator
- ISSP cable
- USB 2.0 cable
- Blue Cat-5e cable
- MiniEval programming board
- Two units of 28-pin DIP samples (CY8C29466-24PXI)
- ZIF socket
- CY3250 flex cable
- Backward compatibility adapter
- 29000-28 PDIP kit

- 12-V 1-A adapter
- CY3215-DK kit CD
  - PSoC Designer installation file
  - PSoC Programmer installation file
  - Bridge Control Panel installation file (packaged along with PSoC Programmer)
  - Kit guide
  - Quick start guide
  - Release notes

Inspect the contents of the kit; if any parts are missing, contact your nearest Cypress sales office for further assistance.

1.2 Additional Learning Resources

Visit http://www.cypress.com for additional learning resources in the form of datasheets, technical reference manual, and application notes. For more information, go to:

- PSoC Designer functionality and releases:
  http://www.cypress.com/go/psocdesigner
- PSoC Programmer, supported hardware and COM layer:
  http://www.cypress.com/go/psocprogrammer
- PSoC Designer-related trainings:
  http://www.cypress.com/go/psocdesignertraining
- CY3250-29xxxQFN ICE Pod Schematic:
  http://www.cypress.com/go/CY3250-29xxxQFN
- CY3250-FLEXCABLE Mechanical Layout Drawing:
  http://www.cypress.com/go/CY3250-29xxxQFN
- CY3250-29xxxQFN ICE Pod Mechanical Layout Drawing:
  http://www.cypress.com/go/CY3250-29xxxQFN
- AN73212 Debugging with PSoC 1: http://www.cypress.com/?rID=57555

1.3 Documentation Conventions

Table 1-1. Document Conventions for Guides

Convention	Usage
`Courier New`	Displays file locations, user entered text, and source code: `C:\...cd\icc\`
Italics	Displays file names and reference documentation: Read about the *sourcefile.hex* file in the *PSoC Designer User Guide*.
[**Bracketed, Bold**]	Displays keyboard commands in procedures: [**Enter**] or [**Ctrl**] [**C**]
File >> Open	Represents menu paths: File >> Open >> New Project
Bold	Displays commands, menu paths, and icon names in procedures: Click the **File** icon and then click **Open**.
Times New Roman	Displays an equation: 2 + 2 = 4
Text in gray boxes	Describes cautions or unique functionality of the product.

2. Getting Started

This chapter describes how to install the CY3215-DK PSoC 1 In-Circuit Emulator Development Kit.

2.1 Kit Installation

To install the CY3215-DK kit, follow these steps:

1. Insert the kit CD into the CD/DVD drive of your PC. The CD is designed to auto-run and the kit installer startup screen appears.

   Note You can also download the latest kit installer from http://www.cypress.com/go/CY3215-DK. Three different types of installers are available for download.

   a. CY3215-DK_ISO: This file (ISO image) is an archive file of the optical disc provided with the kit. You can use this to create an installer CD or extract information using WinRar or similar tools.

   b. CY3215-DK_Setup.exe: This executable file installs the contents of the kit CD, which includes PSoC Programmer, PSoC Designer, and user documents.

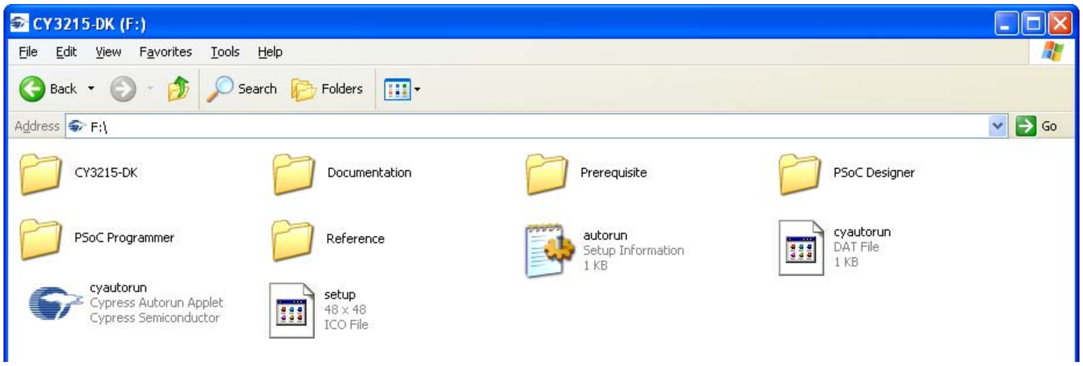
   c. CY3215-DK_SetupOnlyPackage.exe: This executable file installs only the kit contents, which includes kit user documents.

2. Click **Install CY3215-DK** to start the installation, as shown in Figure 2-1.

Figure 2-1. Kit Installer Startup Screen

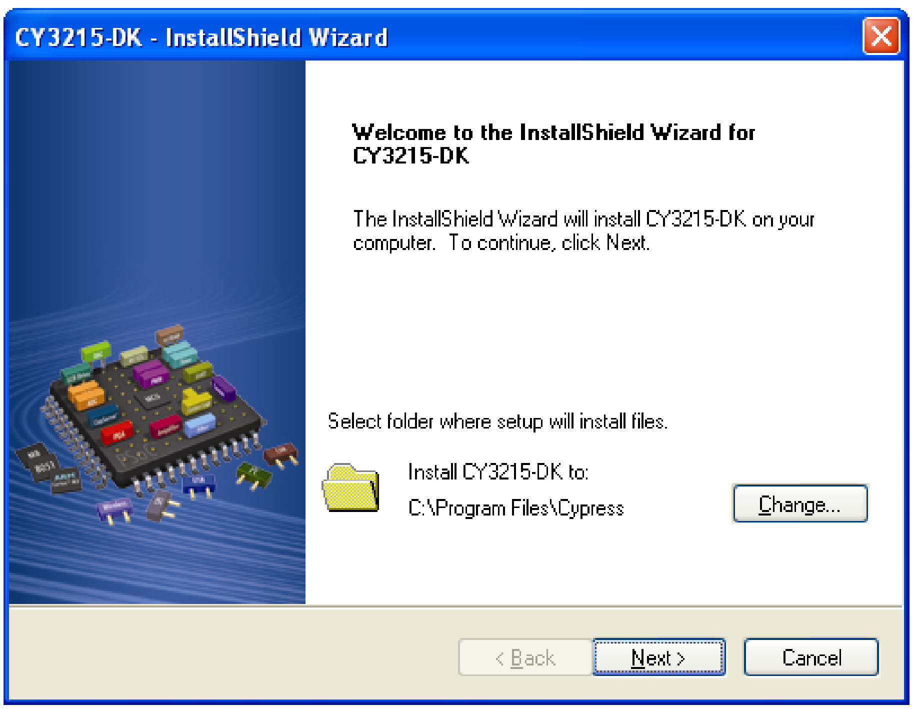
Note If auto-run does not execute, double-click the *cyautorun.exe* file on the root directory of the CD, as shown in Figure 2-2. To access the root directory, click **Start > My Computer > CY3215-DK <drive:>**.

Figure 2-2. CD Root Directory



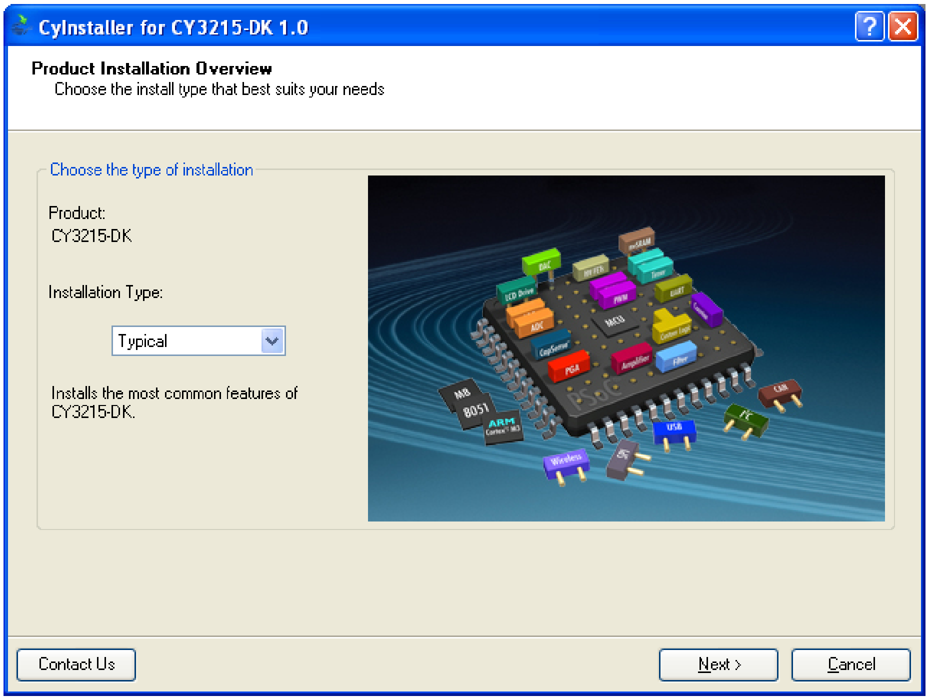
3. In the **InstallShield Wizard**, choose the folder location to install the setup files. You can change the location of the folder for the setup files using **Change**, as shown in Figure 2-3.
4. Click **Next** to launch the kit installer.

Figure 2-3. InstallShield Wizard



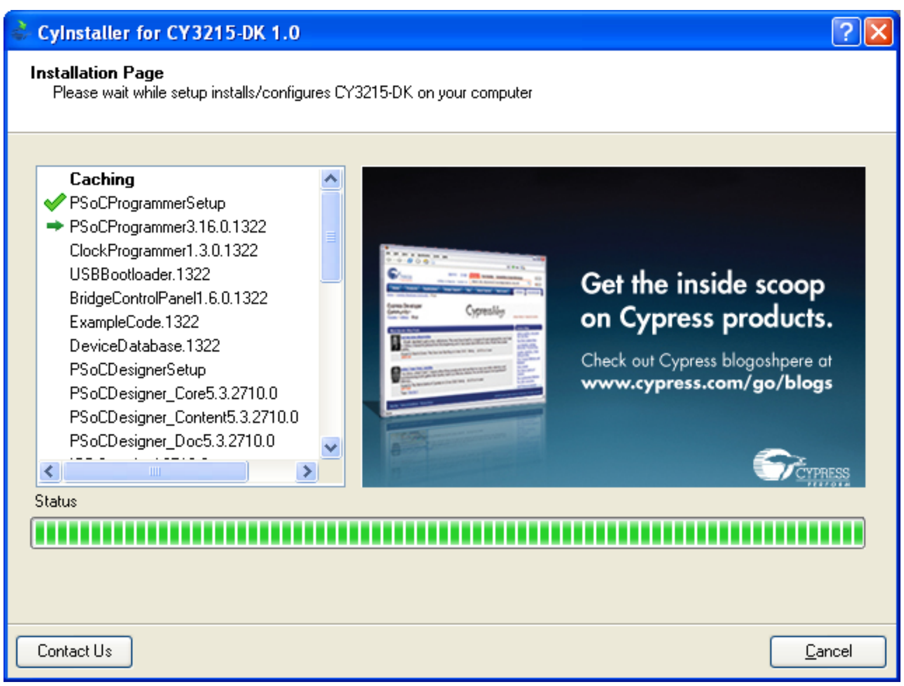
5. On the **Product Installation Overview** screen, select the installation type that best suits your requirement. The drop-down menu has three options: **Typical**, **Complete**, and **Custom**, as shown in Figure 2-4.
6. Click **Next** to start the installation.

Figure 2-4. Installation Type Options



7. When the installation begins, a list of packages appears on the **Installation Page**. A green check mark appears adjacent to every package that is downloaded and installed, as shown in Figure 2-5.
8. Wait until all the packages are downloaded and installed successfully.

Figure 2-5. Installation Page



9. Click **Finish** to complete the kit installation, as shown in Figure 2-6.

Figure 2-6. Installation Complete Page

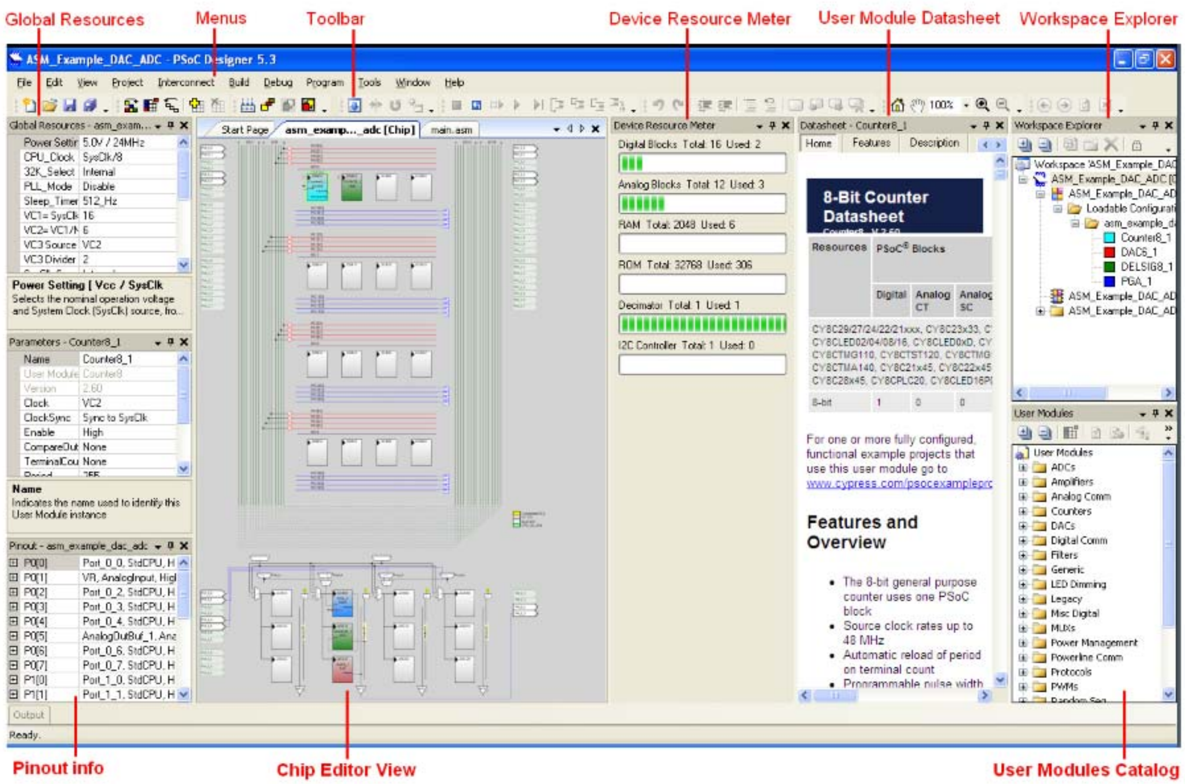
After software installation, verify that you have all hardware and drivers set up for the CY3215-DK kit by connecting the kit to your PC via its USB interface. Because this is the first time you have connected this board to this PC, initial drivers are installed. If you encounter a prompt related to Windows Logo Testing, click the **Continue Anyway** button to allow the driver installation to finish. Follow the instructions to complete the installation process.

2.2 PSoC Designer

PSoC Designer is the revolutionary integrated design environment (IDE) that helps to customize PSoC 1 to meet specific application requirements. PSoC Designer software accelerates system bring-up and time-to-market.

1. To open PSoC Designer, click **Start** > **All Programs** > **Cypress** > **PSoC Designer <version>** > **PSoC Designer <version>**.
2. To create a new project in PSoC Designer, click **File** > **New Project**.
3. To open an existing project in PSoC Designer, click **File** > **Open**.

Figure 2-7. PSoC Designer Interconnect View



To experiment with the code examples, go to Code Examples on page 26.

Note For more details on PSoC Designer, see the PSoC Designer IDE Guide located at: `<Install_directory>:\PSoC Designer\<version>\Documentation.`

See Additional Learning Resources on page 5 for links to PSoC Designer training.

The PSoC Designer quick start guide is available at: http://www.cypress.com/?rID=47954.

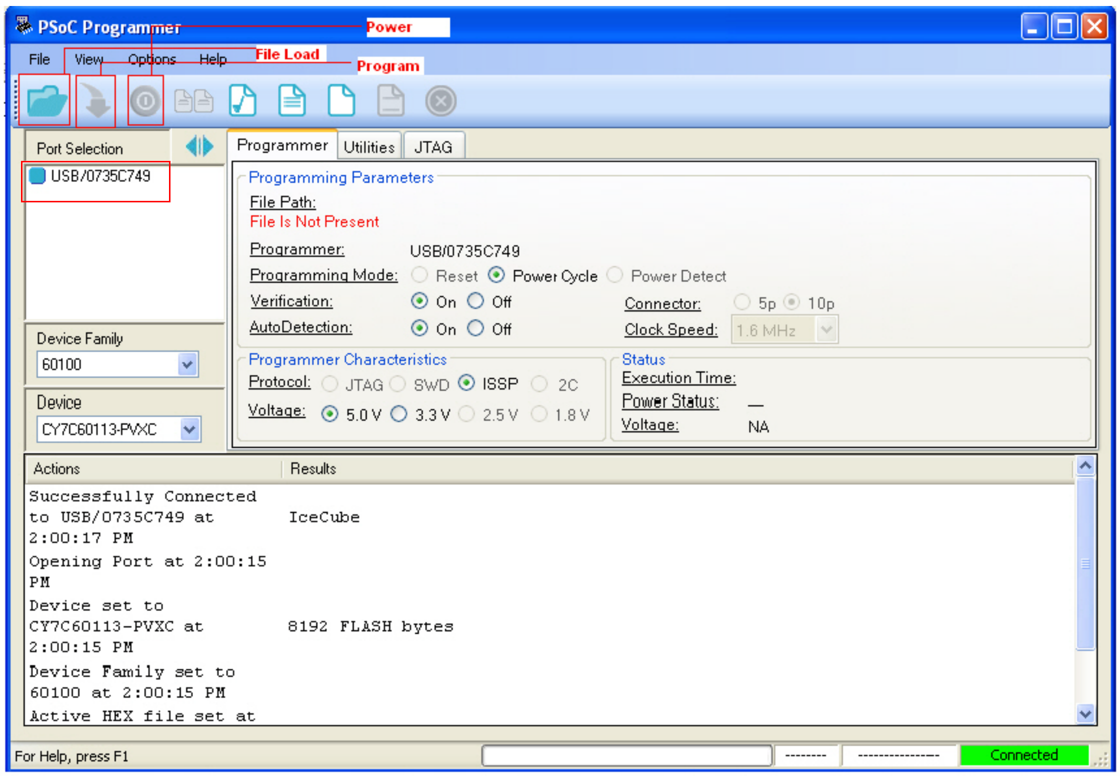
2.3 PSoC Programmer

To open PSoC Programmer, click **Start** > **All Programs** > **Cypress** > **PSoC Programmer <version>** > **PSoC Programmer <version>**.

To successfully program the device, follow these steps:

1. Select the ICE-Cube connectivity in **Port Selection**, as shown in Figure 2-8.

Figure 2-8. PSoC Programmer Window



2. Click the **File Load** button to load the hex file.
3. Select the device and device family from the list.
4. Click the **Program** button to program the selected PSoC 1 device.
5. Close PSoC Programmer.

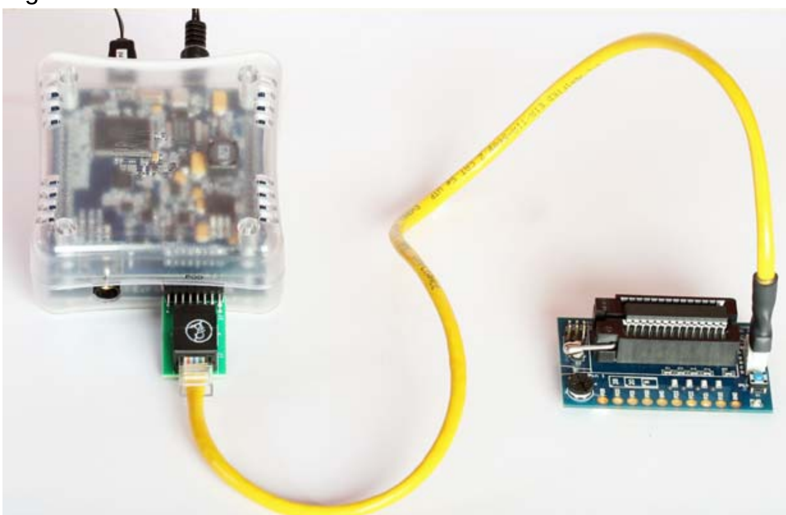
Note For more details on PSoC Programmer, see the user guide at the following location:
<Install_directory>\Cypress\Programmer\<version>\Documents

3. Using ICE-Cube

3.1 Connecting the ICE-Cube

PSoC Designer supports the ICE-Cube. This new in-circuit emulator replaces the ICE-4000 and the USB adapter for seamless USB connection, debugging, and programming.

Figure 3-1. ICE-Cube



3.1.1 Connect using a USB Port

The ICE-Cube connects to any computer using a standard USB 2.0 cable, included in Cypress development kits. To connect the ICE-Cube to your computer, plug the USB cable into your computer and attach the other end to the ICE-Cube.

The ICE-Cube is a plug-and-play device and it should be recognized automatically by any computer with PSoC Designer and PSoC Programmer installed. If a USB connection problem occurs, refer to Microsoft Windows Help for troubleshooting Windows connectivity issues.

Figure 3-2. USB Port in ICE-Cube

3.1.2 Connect using a Flex Cable

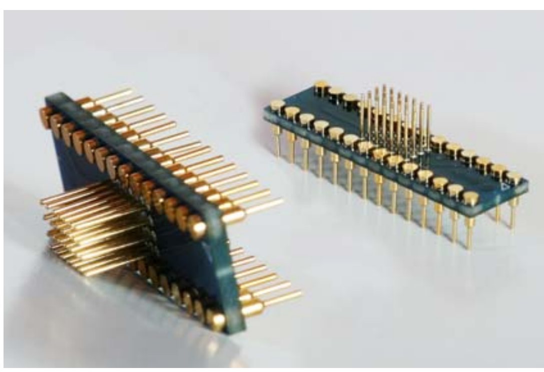
A flex cable is used to connect the ICE-Cube to the pod main board.

- **Pod:** Pods connect to target circuits via foot. As each pod contains a fully functional PSoC bond-out device, pods may be used instead of devices for test purposes. Simply plug a pod into the circuit without connecting it to an ICE base station. The pod power LED lights up when it is powered and operational.

Figure 3-3. Flex Cable and 29000-28 DIP

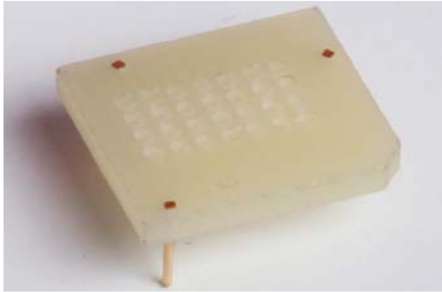
- **Foot:** A foot is used to connect the pod to the MiniEval board. Each foot has a pinout that models a PSoC, for example, a 28-pin DIP. A foot that emulates surface-mount components must be soldered to target circuits. The main board of the pod can then be attached or removed, as desired.

Figure 3-4. 28-Pin DIP Feet



- **Plastic Mask:** The plastic mask is used to orient the foot. Plastic masks are provided to expose only the pins that connect to the foot.

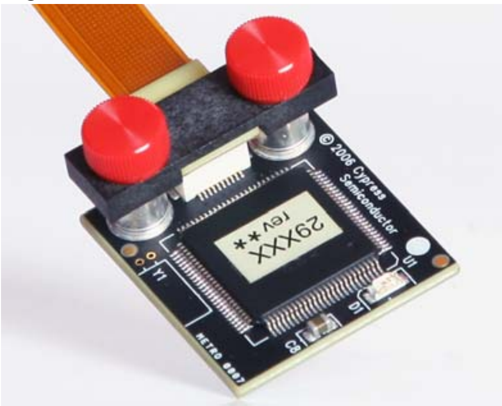
Figure 3-5. Plastic Mask



Before connecting the flex cable to the ICE-Cube, assemble the three pieces (pod, plastic mask, and foot), as shown in Figure 3-6.

1. Select a foot. The foot should match the pinout of the PSoC 1 device used in the target circuit.
2. Next, select the mask that matches the desired foot.
3. Insert the mask into the bottom of the pod, aligning the chamfered corners of the mask to the pin-1 triangle on the pod.
4. Insert the foot through the plastic mask. Use the alignment triangles to orient the foot to the pod.
5. Plug the pod into the MiniEval board via ZIF socket.

Figure 3-6. Assembled Pod with Flex Cable



6. Connect the other end of the flex cable to the ICE-Cube pod connector, as shown in Figure 3-7.

Figure 3-7. Flex Pod Connected to ICE-Cube Pod

Table 3-1. Pin Description of Pod Connector

Pin No.	Pin Name	Pin Description
1	POD EXTRA3	Future use
2	GND	Ground
3	–	–
4	POD_OCDDE	POD_OCD even data I/O
5	GND	Ground
6	POD_OCDD)	POD_OCD odd data output
7	POD EXTRA1	Future use
8	POD_XRES	Reset signal (required only for Reset programming mode)
9	GND	Ground
10	POD_OCDHC	POD_OCD high speed clock output
11	GND	Ground
12	POD_OCDCC	POD_OCD CPU clock output
13	POD EXTRA4	Future use
14	PODVCC	Supply voltage
15	–	–
16	PODVCC	Supply voltage

3.1.3 Connect using a Backward Compatibility Adapter

A backward compatibility adapter can be used to debug and program the PSoC 1 chips.

Figure 3-8. Backward Compatibility Adapter

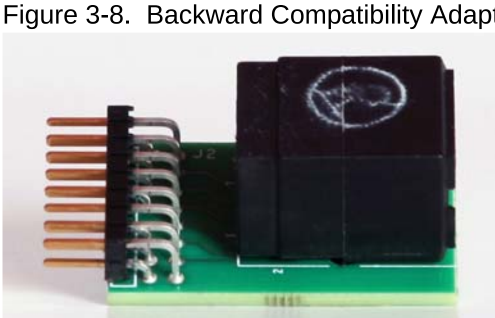
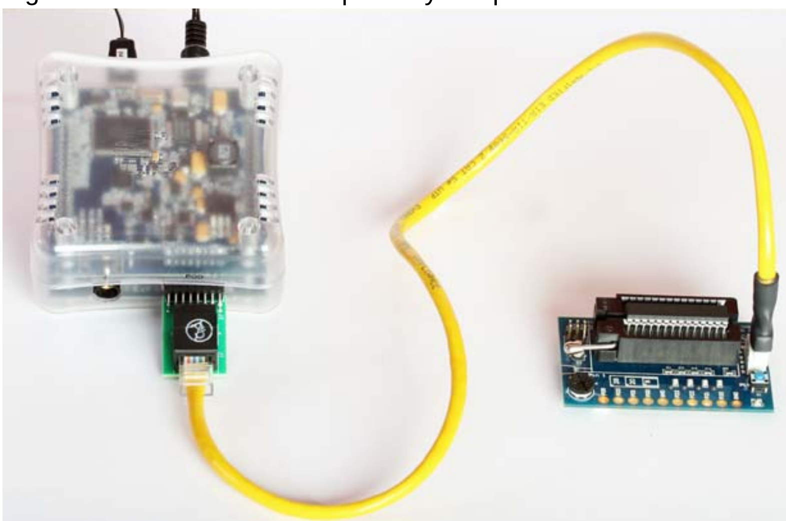


Figure 3-9. Connector in ICE-Cube

To program the device, follow these steps:

1. Using a USB cable, plug the ICE-Cube into the installed application. Make sure the ICE-Cube is powered on.
2. Connect the ISSP cable to the ICE-Cube through the backward compatibility adapter.
3. Place the 5-pin end of the ISSP cable on the 5-pin header(J2) of the MiniEval board.
4. Place the CY8C29466 chip on the MiniEval board via ZIF socket.
5. Launch PSoC Programmer.
6. Load the hex file using the **File Load** button.
7. Program it successfully.

Figure 3-10. Backward Compatibility Adapter Connected to ICE-Cube Pod

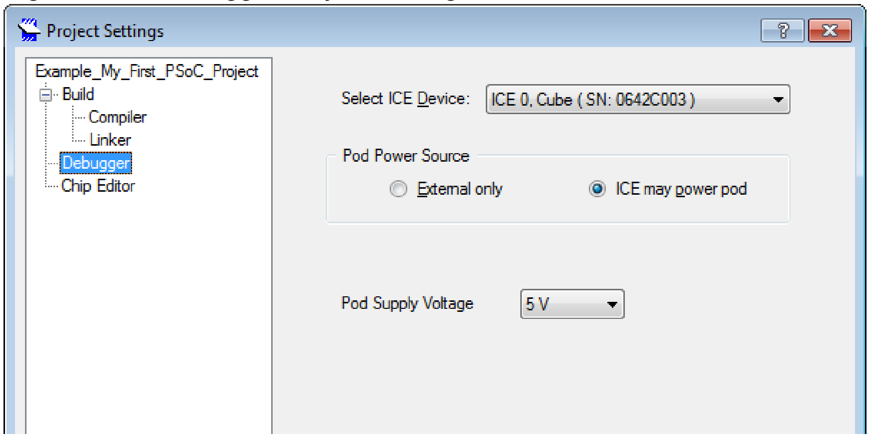


3.1.4 Software Configuration

After the physical connection is made, you are ready to configure the internal connection from the computer to the ICE. The ICE enables communication and debugging between PSoC Designer and the pod. To connect the ICE from within PSoC Designer, perform the following steps:

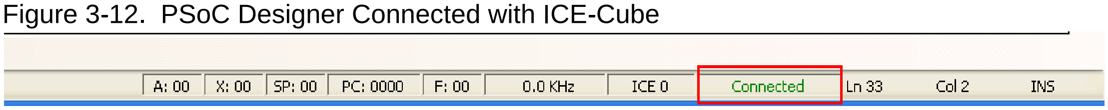
1. Confirm that the flex cable and pod are attached to the ICE-Cube.
2. Confirm that the ICE is powered from the power adapter.
3. Confirm that the USB cable is connected from the ICE-Cube to the PC.
4. Create a project using PSoC Designer. For more information, see My First Code Example on page 26.
5. To access the debugger subsystem, go to **Project > Settings > Debugger** and click the **Debugger** icon.
6. Select the correct port from the drop-down window. The target board may either be powered by ICE-Cube or an external source. Select **5V** as **Pod Supply Voltage** if the board is powered by ICE-Cube.
7. Click **OK**.

Figure 3-11. Debugger Project Setting



8. On successful connection, you receive a notification in the Output tab of the Status window; a green indicator displays **Connected** in the lower-right corner of the program.

Figure 3-12. PSoC Designer Connected with ICE-Cube



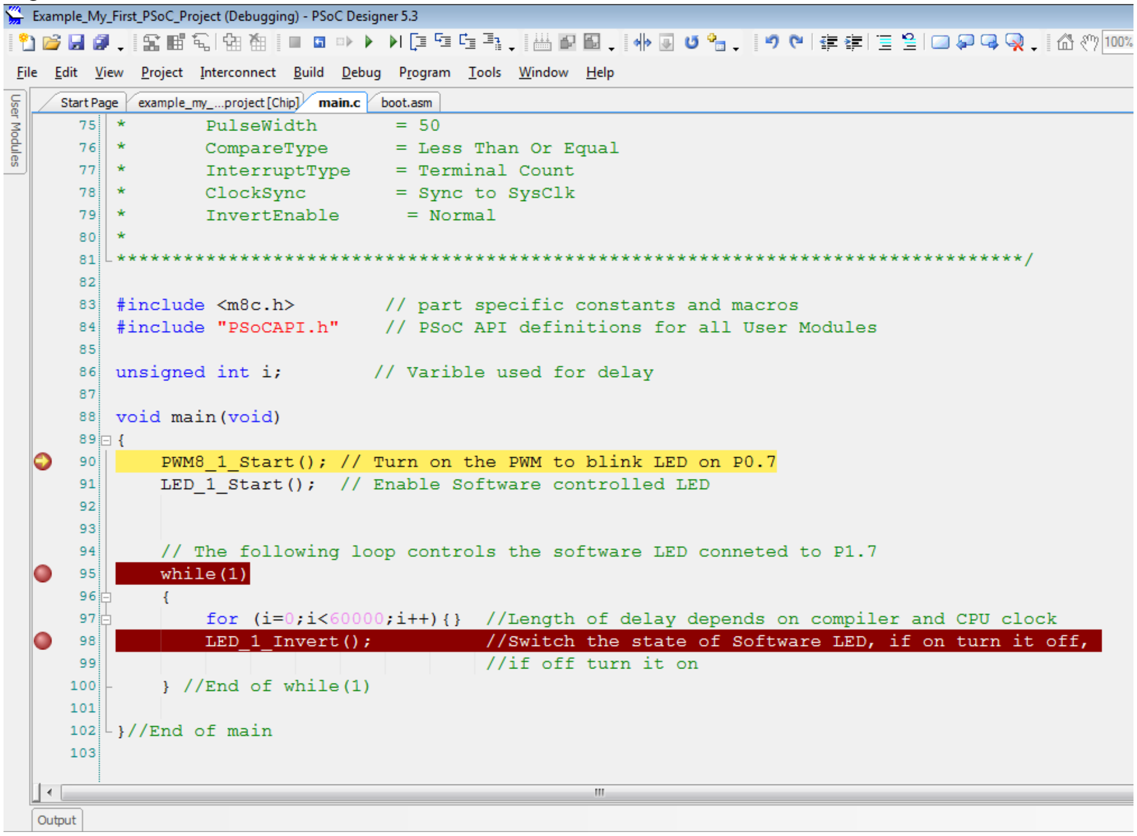
3.1.5 Debug a Project

To successfully debug a project, follow these steps:

1. Connect the ICE-Cube to the PSoC 1 development board.
2. Click **Start > All Programs > Cypress > PSoC Designer <version> > PSoC Designer <version>**.
3. Create a new project in PSoC Designer by clicking **File** > **New Project**.

   Note To open an existing project, click **File** > **Open**.
4. To connect the ICE-Cube and PSoC 1 development board, go to **PSoC Designer > Debug**.
5. Click **Connect/Disconnect** or press **F9**.
6. Right-click a line in the project from where the debugging process should start. The **Insert/Delete Breakpoint** option appears.

Figure 3-13. Break Points in main.c

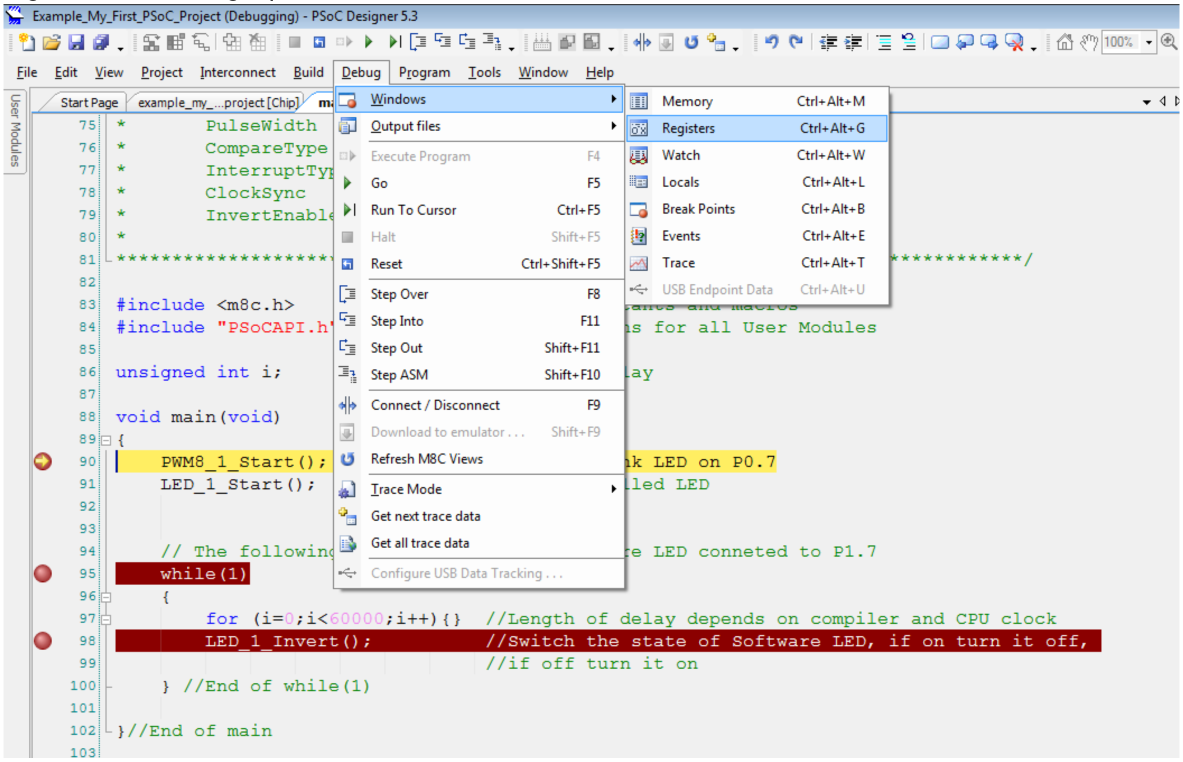


7. To view memory, registers, and watch variables at a particular location, go to **Debug > Windows**.
8. To start the debugging process, go to **Debug > Go** or press **F5**.

   Use one of the following options for the debugging process:

   a. **Debug > Step Over** (or press **F8**): Steps over next statement
   b. **Debug > Step Into** (or press **F9**): Steps into next statement
   c. **Debug > Step Out** (or press **Shift + F11**): Steps out of current function
   d. **Debug > Step ASM** (or press **Shift + F10**): If the current line is C code, the line is located in the first file and that line is executed

Figure 3-14. Debug Options

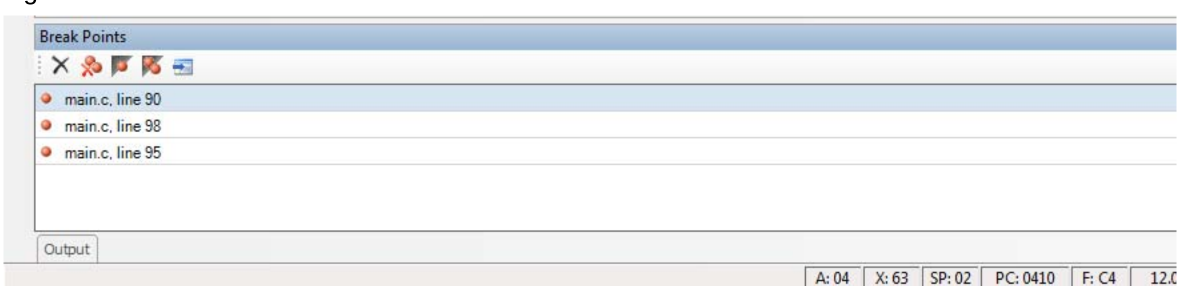


3.1.5.1 Break Points

The break point feature allows you to stop program execution at predetermined address locations. When a break point is encountered, the program stops at the address of the break point, without executing the address code. The program is restarted using the available menu or icon options.

To set break points, first open the file to debug. Right-click the mouse at specific points and select **Insert Break Point**. You can view and remove active break points in the Break Points window. To open the Break Points window, select **Debug > Windows > Break Points**.

Figure 3-15. Break Points Window

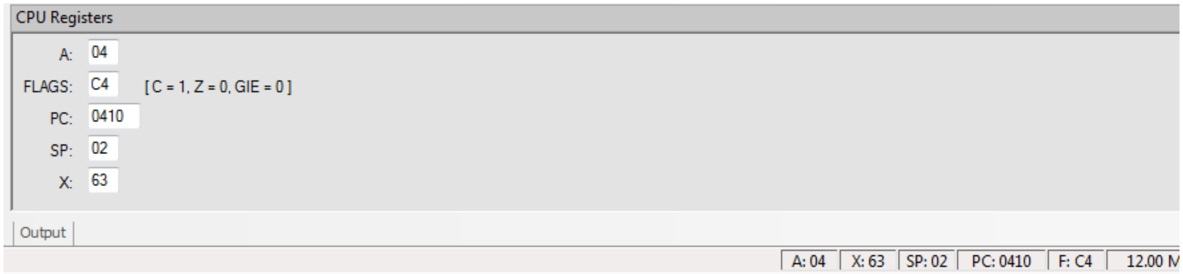


3.1.5.2 CPU and Register Views

During debugging, you can read and write in five areas: CPU registers, bank registers 0, bank registers 1, RAM, and flash. The CPU registers are shown in their own window (**Debug > Windows > Registers**) and in the notification area at the bottom of PSoC Designer. The other four areas can be viewed in the Memory Window (**Debug > Windows > Memory**). Select one of the four memory areas from the **Address Space** box.

CPU Registers: This window allows you to examine and change the contents of the CPU registers. Data is entered in hexadecimal notation. CPU register values can be viewed across the bottom of PSoC Designer.

Figure 3-16. CPU Register in Memory Window

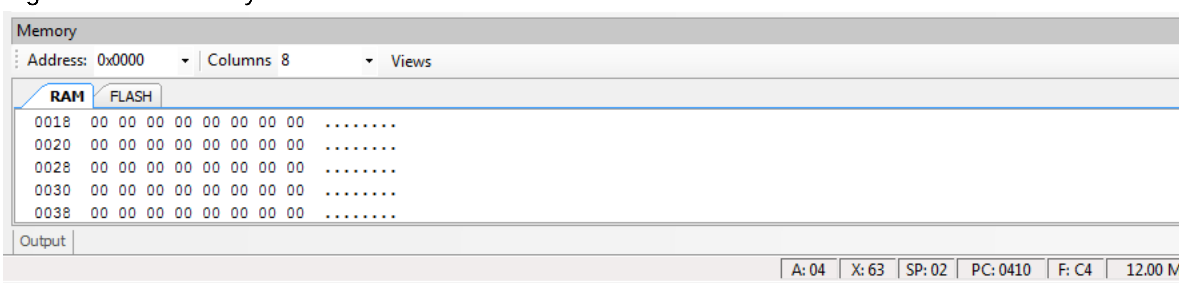


RAM: RAM locations can be modified by clicking the data at the specific location and typing in the new value. Data is entered in hexadecimal notation.

Flash: The flash window displays the data stored in flash. This is the program memory; it is read-only.

Bank Registers 0 and 1: You can scroll through the register bank to view the values in the register bank. Type a new value into the Offset to scroll directly to that offset. Click next to a value and type a new value for the register.

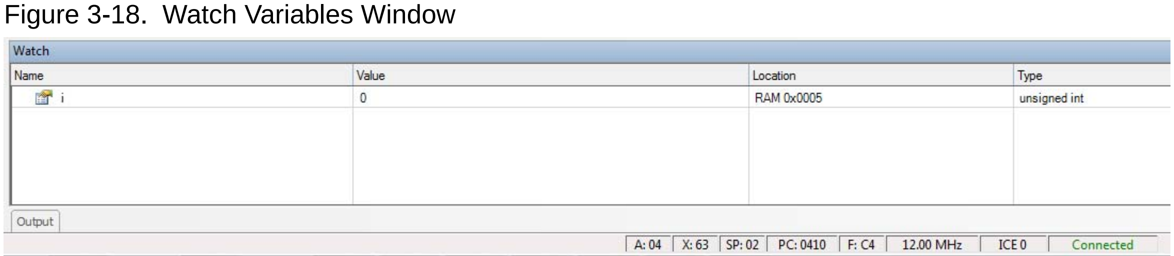
Figure 3-17. Memory Window



3.1.5.3 Watch Variables

To set watch variables, right-click a variable in a source file and select **Add Watch**. You can also select **Global Variables**. Right-click **Add**, **Delete**, or **Properties** in the Watch/Global Name window to add, delete, or modify values.

Figure 3-18. Watch Variables Window

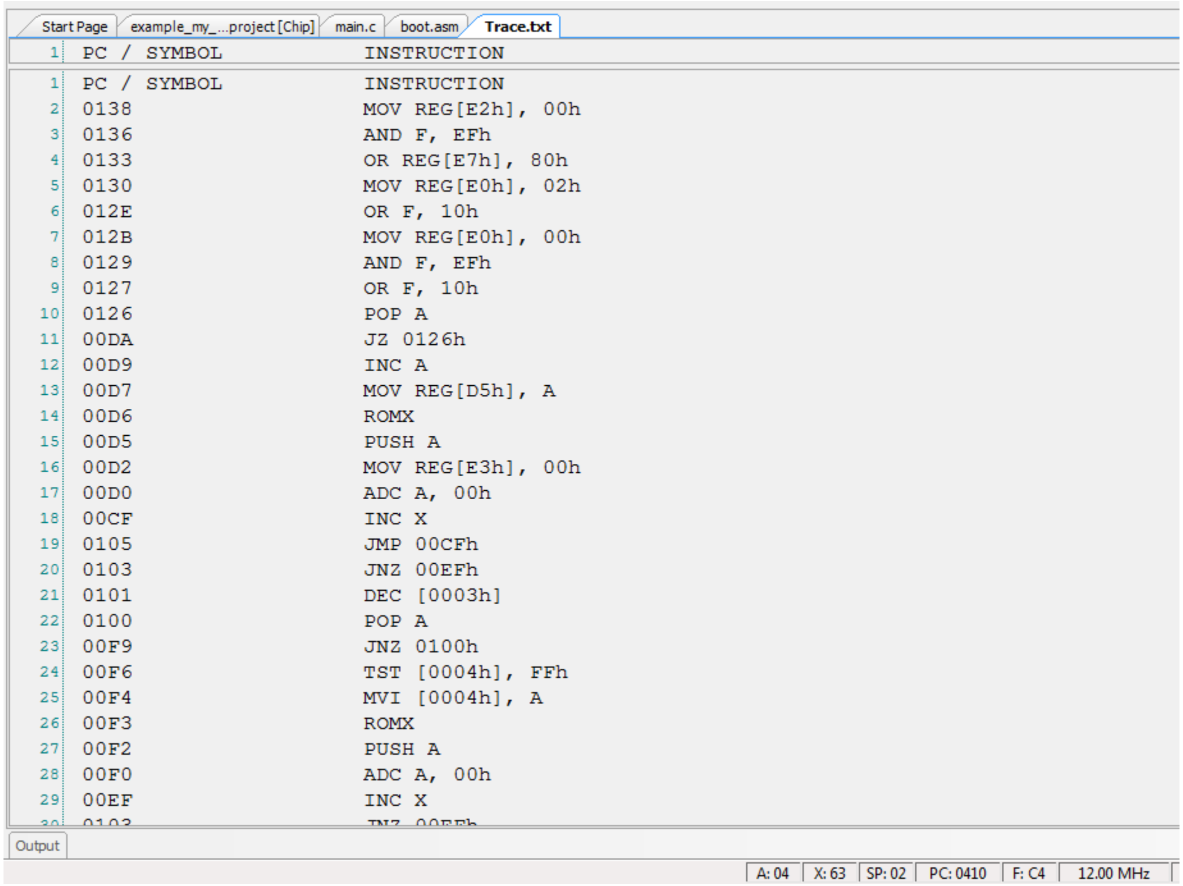


3.1.5.4 Trace

The Trace window is displayed when **Debug > Windows > Trace** is chosen. It displays a continuous, configurable listing of project symbols and operations from the last breakpoint. The trace shows symbolic, rather than address data, to enhance readability.

Each time the program executes, the trace buffer is cleared. When the trace buffer becomes full, it continues to operate and overwrite old data.

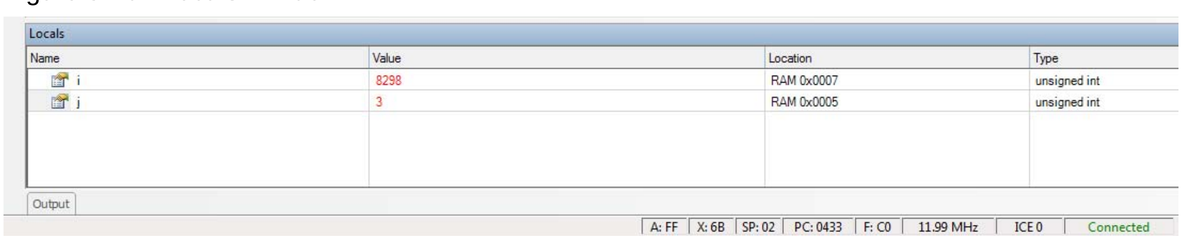
Figure 3-19. Trace Window



3.1.5.5 *Locals*

A separate window is available for local variables. Whenever execution halts, the local variables are updated to the current value.

Figure 3-20. Locals Window

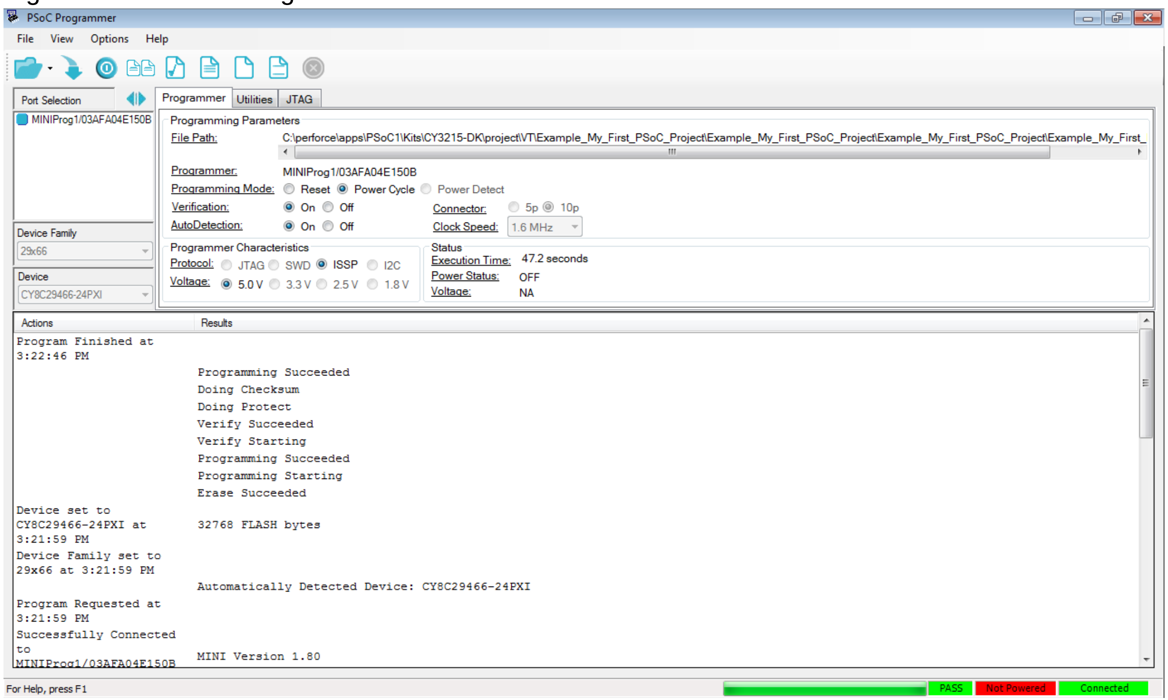


WARNING: The time taken to execute a system supervisory call (SSC) such as FlashRead/Write, Table Read, and Erase Block is significantly more while emulating a code through ICE-Cube when compared to the time taken when running the code on chip.

3.2 PSoC Programmer

PSoC Programmer is used as a standalone application to program PSoC devices. It can be launched within PSoC Designer or accessed from the desktop as a standalone program.

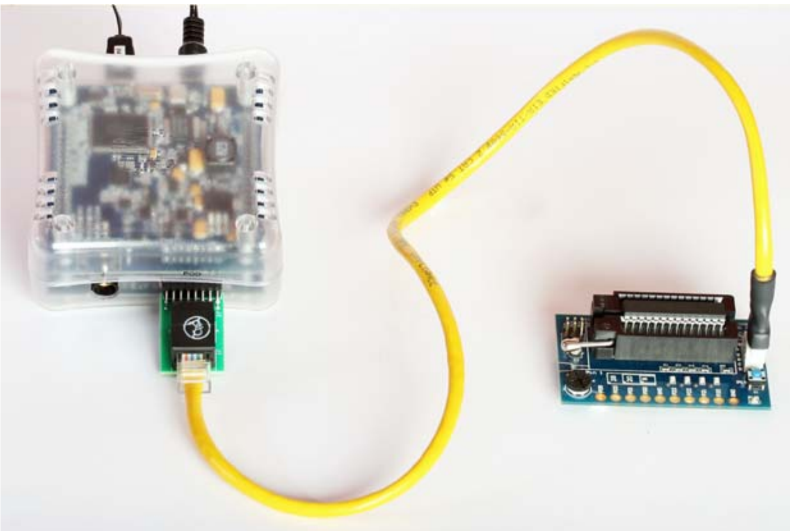
Figure 3-21. PSoC Programmer Interface



To program a target board using PSoC Programmer via ICE-Cube pod connector, follow these steps:

1. Set up all hardware, including the device to be programmed.
2. Disconnect power from the target board.
3. Launch PSoC Programmer:
   a. From the desktop, click **Windows Start > All Programs > Cypress > PSoC Programmer**.
   b. From within PSoC Designer, click **Program > Program Part**.
4. Click **File Load** to select a file for programming.
5. Select the port used to connect the programmer.
6. Select the device family and device used to generate the hex file.
7. Select **Reset** under programming mode.
8. Apply power to the target board.
9. Click **Program** to start device programming.
10. The action window reports the status and success of programming.

Figure 3-22. Hardware Configuration with ICE-Cube



4. Code Examples

4.1 My First Code Example

4.1.1 Project Objective

This project is used to demonstrate blinking an LED at a varying duty cycle using a hardware pulse width modulator (PWM). Another LED is caused to blink using a software delay. The clock dividers VC1, VC2, and VC3 are used to divide the 24-MHz system clock by 16, 16, and 256, respectively. The resulting 366-Hz clock is used as the input to an 8-bit PWM. This in turn produces an LED blink period of 1.4 Hz. The project also demonstrates how an LED can toggle on/off with a delay of approximately 1 second.

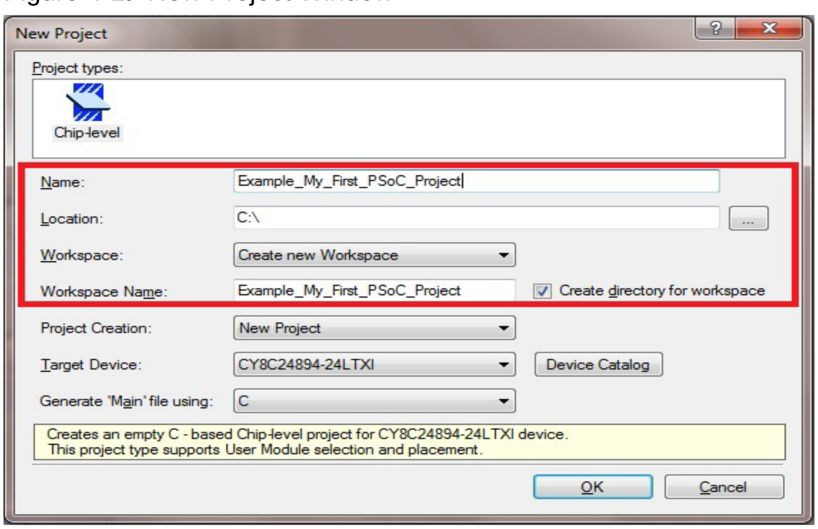
The following user modules are used in this project:

- PWM - An 8-bit PWM is used to generate a 366-Hz signal. An LED is connected to the PWM output. This LED blinks at 1.4 Hz.
- LED - This module is used to toggle an LED on/off.

4.1.2 Creating My First PSoC 1 Project

1. Open PSoC Designer 5.3.
2. To create a new project, click **File** > **New Project**. The New Project window opens.
3. In this window, select the **Chip-level** icon. Name the project **Example_My_First_PSoC_Project**, as shown in Figure 4-1.
4. Click **Browse** and navigate to the directory in which the project should be created.

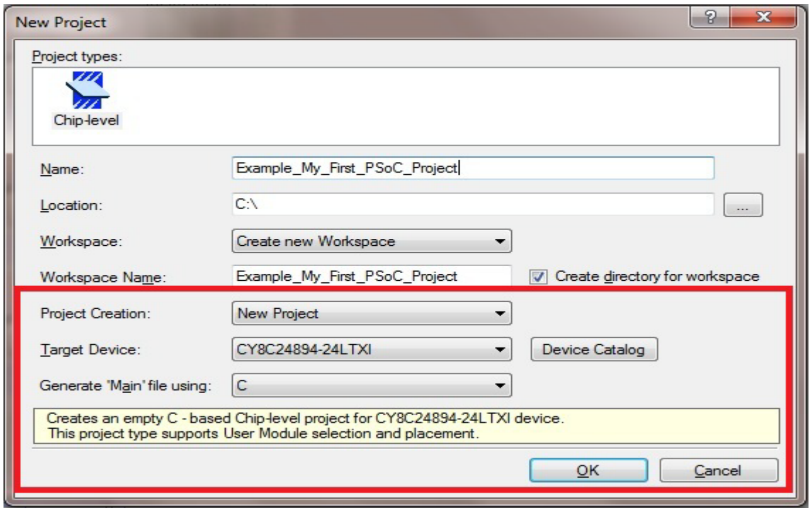
Figure 4-1. New Project Window



5. In the project creation drop down menu select new project, as shown in Figure 4-2

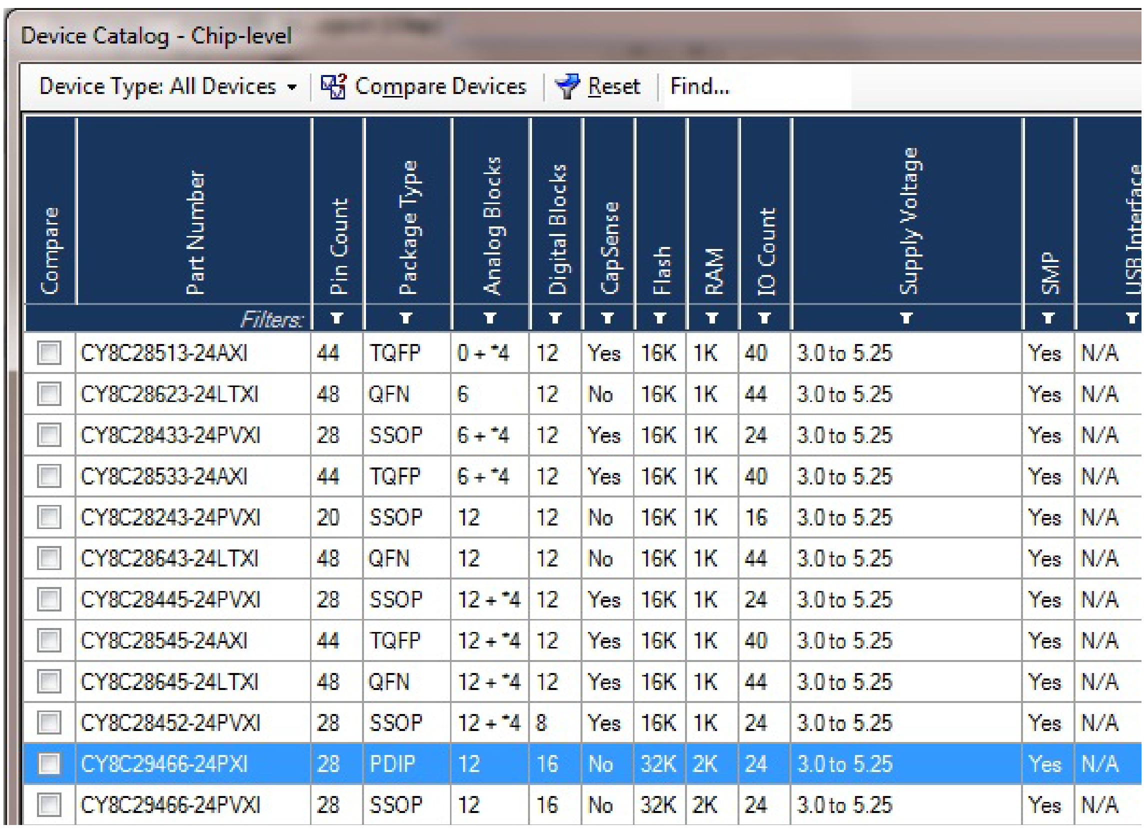
6. In this window, under Select Target Device, click **Device Catalog**.

Figure 4-2. Select Project Type Window



7. The Device Catalog window opens, then scroll down to the **CY8C29466, CY8C29566,…** section.
8. For this project, click **CY8C29466-24PXI** and then click **Select**; see Figure 4-3.

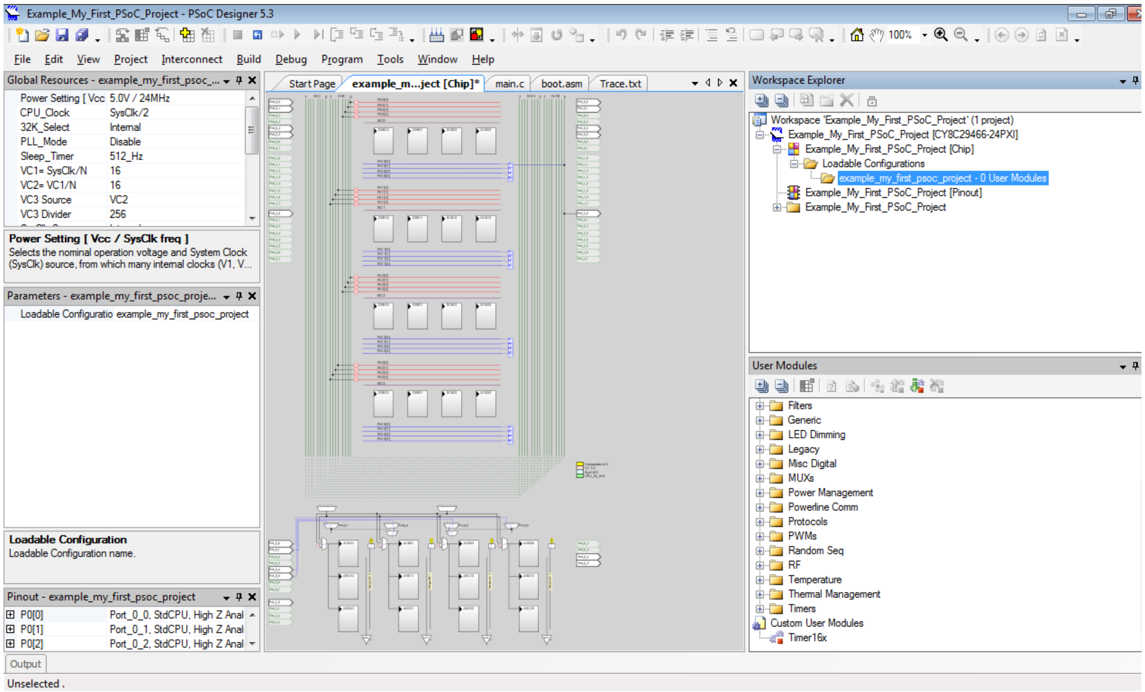
Figure 4-3. Device Catalog Window



Compare	Part Number	Pin Count	Package Type	Analog Blocks	Digital Blocks	CapSense	Flash	RAM	IO Count	Supply Voltage	SMP	USB Interface
	CY8C28513-24AXI	44	TQFP	0 + *4	12	Yes	16K	1K	40	3.0 to 5.25	Yes	N/A
	CY8C28623-24LTXI	48	QFN	6	12	No	16K	1K	44	3.0 to 5.25	Yes	N/A
	CY8C28433-24PVXI	28	SSOP	6 + *4	12	Yes	16K	1K	24	3.0 to 5.25	Yes	N/A
	CY8C28533-24AXI	44	TQFP	6 + *4	12	Yes	16K	1K	40	3.0 to 5.25	Yes	N/A
	CY8C28243-24PVXI	20	SSOP	12	12	No	16K	1K	16	3.0 to 5.25	Yes	N/A
	CY8C28643-24LTXI	48	QFN	12	12	No	16K	1K	44	3.0 to 5.25	Yes	N/A
	CY8C28445-24PVXI	28	SSOP	12 + *4	12	Yes	16K	1K	24	3.0 to 5.25	Yes	N/A
	CY8C28545-24AXI	44	TQFP	12 + *4	12	Yes	16K	1K	40	3.0 to 5.25	Yes	N/A
	CY8C28645-24LTXI	48	QFN	12 + *4	12	Yes	16K	1K	44	3.0 to 5.25	Yes	N/A
	CY8C28452-24PVXI	28	SSOP	12 + *4	8	Yes	16K	1K	24	3.0 to 5.25	Yes	N/A
	CY8C29466-24PXI	28	PDIP	12	16	No	32K	2K	24	3.0 to 5.25	Yes	N/A
	CY8C29466-24PVXI	28	SSOP	12	16	No	32K	2K	24	3.0 to 5.25	Yes	N/A

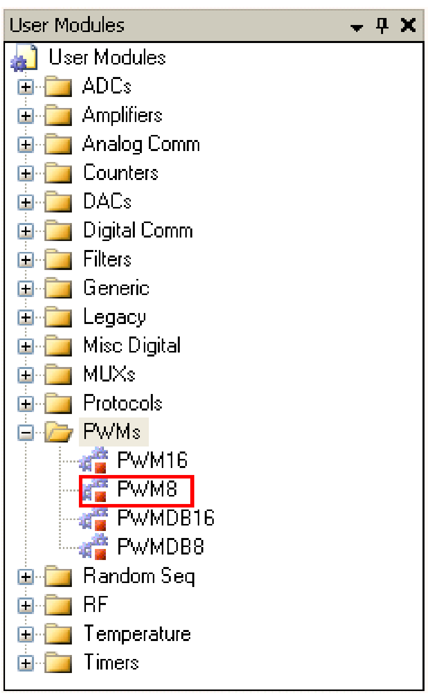
9. Under Generate 'Main' File Using:, select **C**, then click **OK**.
10. By default, the project opens in chip view, as shown in Figure 4-4.

Figure 4-4. Default View



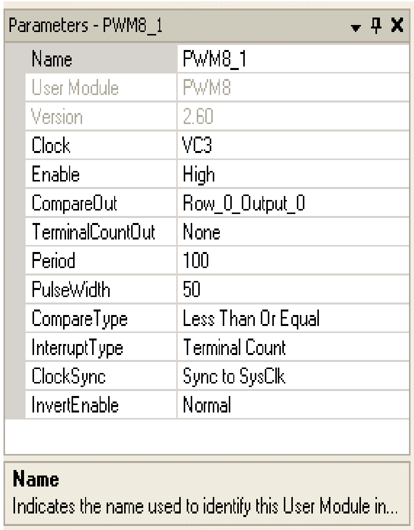
11. In the User Modules window, expand the **PWMs** folder. In this folder, right-click on **PWM8** and select **Place**. The user module (UM) is placed in the first available digital block.

Figure 4-5. User Modules Window



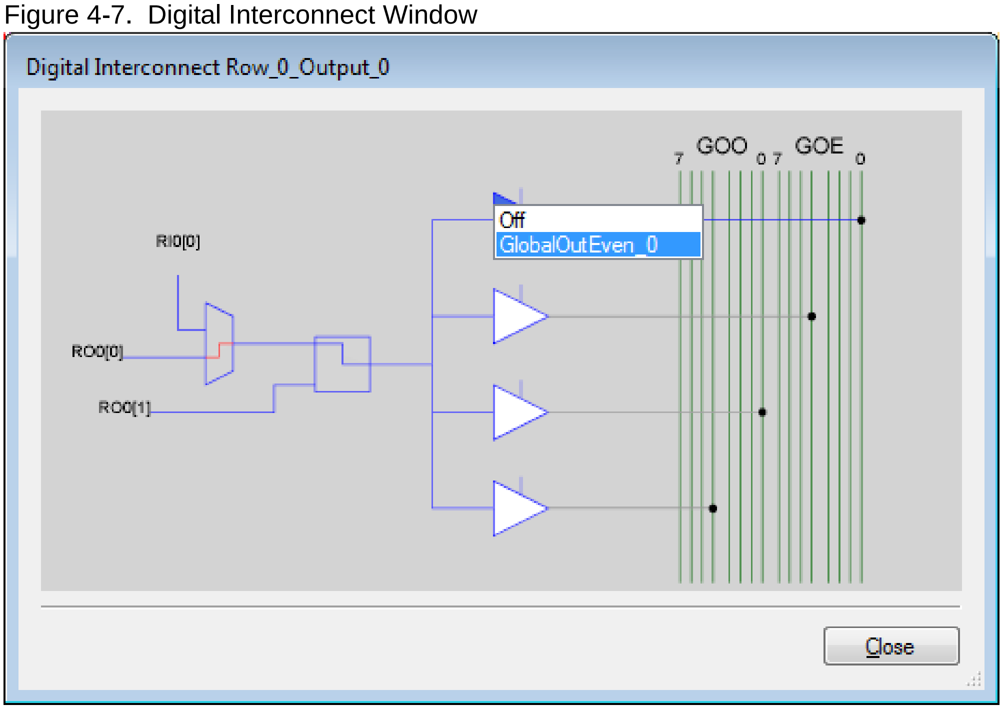
12. Configure the PWM8_1 properties, as shown in the following figure.

Figure 4-6. PWM8 User Module Properties



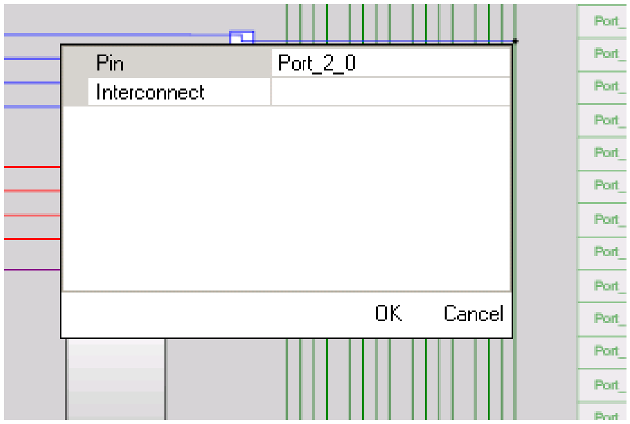
13. Click on the LUT node on **Row_0_Output_0** to open the Digital Interconnect window.
14. In this window, configure **Row_0_Output_0_Drive_0** to connect to **GlobalOutEven_0**.

Figure 4-7. Digital Interconnect Window



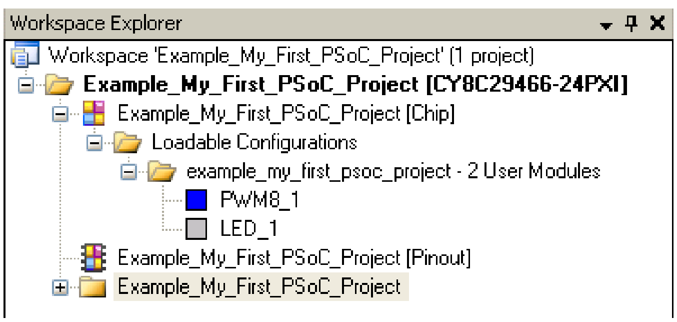
15. Click **Close**.
16. Click **GlobalOutEven_0**. In the window that appears, configure the pin for **Port_2_0**.

Figure 4-8. Configur Pin for Port_2_0



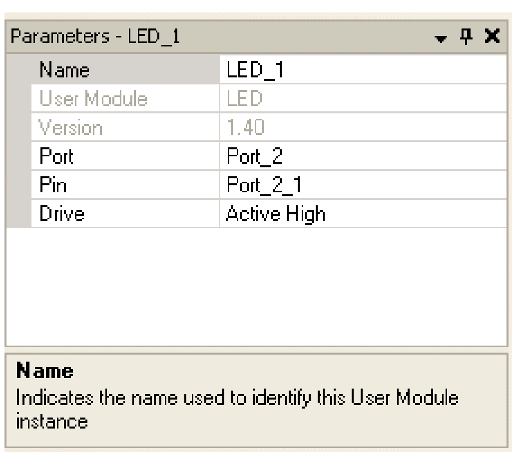
17. Click **OK** to continue.
18. In the User Modules window, expand the **Misc Digital** folder. In this folder, right-click **LED** and select **Place**; this adds the UM to the project. This UM does not use digital or analog blocks. It appears in **Workspace Explorer > Example_My_First_PSoC_Project[CY8C29466-24PXI] > Example_My_First_PSoC_Project[Chip] > Loadable Configurations > example_my_first_psoc_project - 2 User Modules**.

Figure 4-9. Workspace Explorer



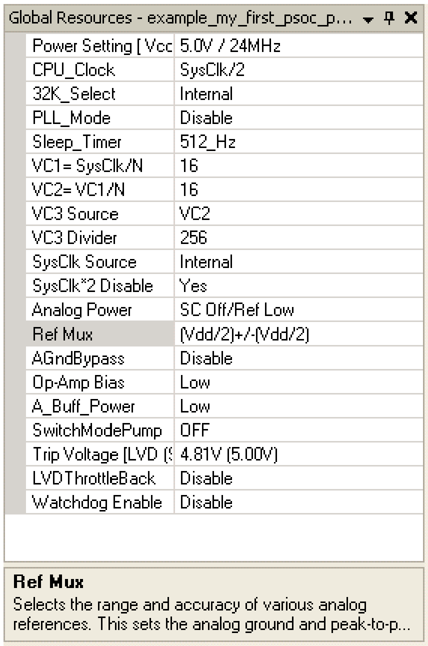
19. Configure the LED properties, as shown in the following figure.

Figure 4-10. LED User Module Properties



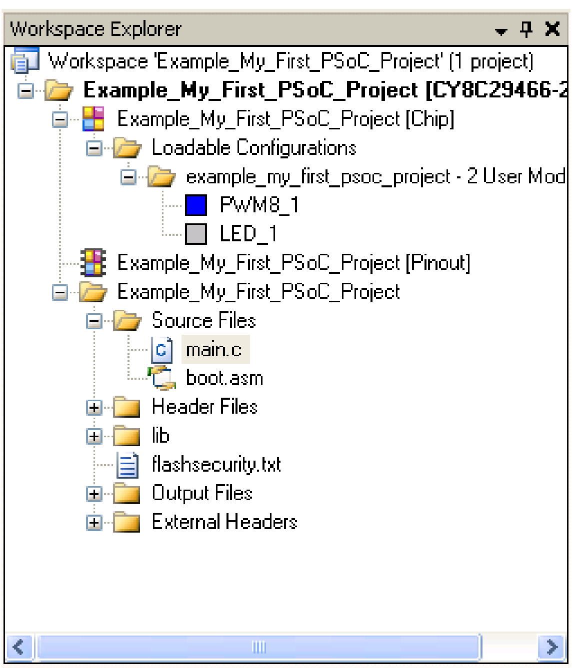
20. Configure the Global Resources window to match the following figure.

Figure 4-11. Global Resources Window



21. Open the existing *main.c* file in Workspace Explorer. Replace the existing *main.c* content with the content of the *My_First_Example_Project_Main.c* file, which is available as an attachment to this PDF document.

Figure 4-12. Workspace Explorer Window



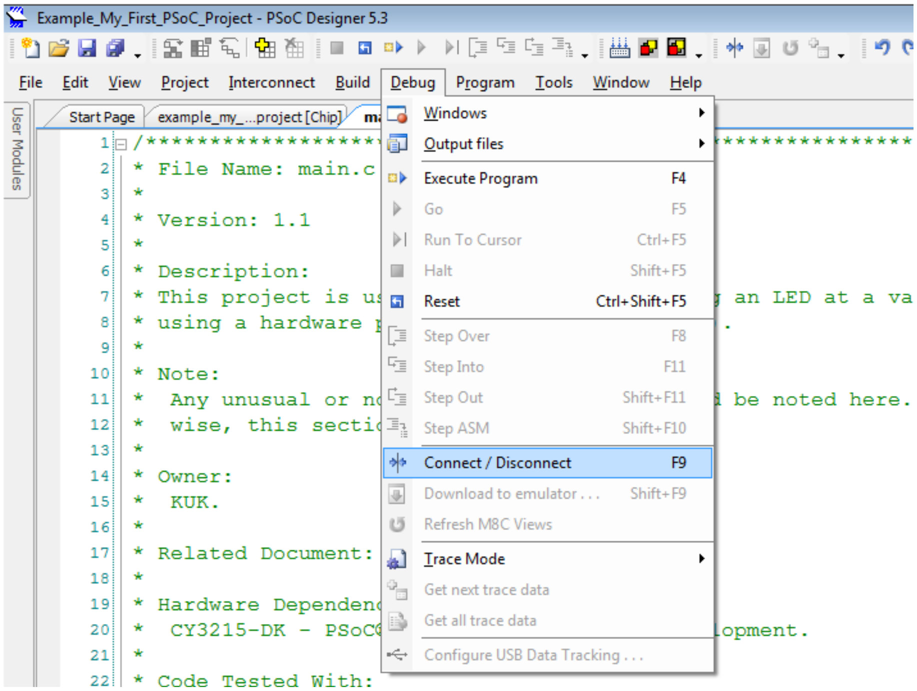
22. Save the project.

23. Click **Build > Generate/Build 'Example_My_First_PSoC_Project'** .

24. Connect the ICE-Cube to the PC. Connect the ICE-Cube to the MiniEval board, as explained in Connecting the ICE-Cube on page 14.

25. In PSoC Designer, select **Debug > Connect/Disconnect** to connect the ICE to PSoC Designer.

Figure 4-13. Connecting ICE to PSoC Designer

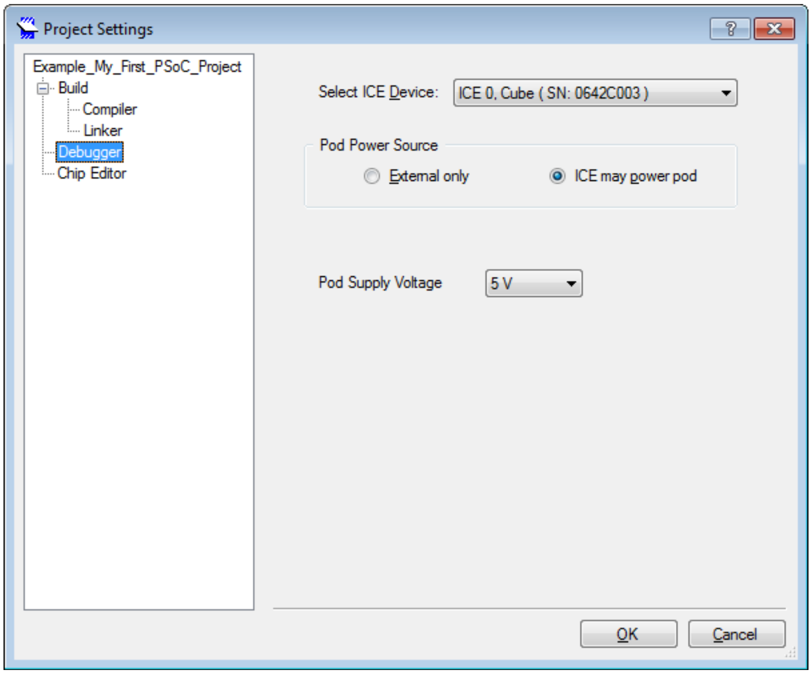


26. On successful connection, the status bar at the bottom of the PSoC Designer IDE shows the status as 'Connected'.

Figure 4-14. ICE-Cube Connected

27. If there is an error, ensure that the Debugger settings in **Project > Settings** match the following figure.

Figure 4-15. Debugger Settings



Note The pod may be powered by the ICE-Cube or through external power. If external power is used, ensure that the power is on for the ICE to get connected.

28. Click **Debug > Execute Program** or the keyboard shortcut **F4**. The project starts compiling; it is downloaded to the pod and starts running.

4.1.3 Verify Output

Verify the output as follows:

1. Two LEDs connected to P2[0] and P2[1] start blinking.
2. Open main.c file, right-click any line in the while(1) loop in the code and select **Insert/Delete Breakpoint** to include a breakpoint at that point. The execution of the code stops at the point where breakpoint is inserted (LED controlled by software (LED connected to P2[1]) stops blinking). The line is highlighted in yellow when the code execution stops.
3. At this point, you may either continue running the code by selecting **Debug > Run** (keyboard shortcut **F5**) or step through the code by selecting either **Debug > Step Over** (keyboard shortcut **F8**) or **Debug > Step Into** (keyboard shortcut **F11**).

   Note The Step Over command executes statement one by one and does not enter into the function calls whereas the Step Into command enters the function calls as well.

Use the debug windows (see Debug a Project on page 19) to explore the other debugging features offered in PSoC designer. For more details on debugging, debug menu options, and debug strategies, see **Help > Help Topics > Debugger**.

Revision History

Document Revision History

Document Title: CY3215-DK PSoC® 1 In-Circuit Emulator Development Kit Guide Document Number: 001-66514				
Revision	**ECN#**	**Issue Date**	**Origin of Change**	**Description of Change**
**	3128877	01/27/2011	RKPM	Initial version of kit guide
*A	3256318	05/11/2011	RKPM	Added Code Examples chapter. Updated images in section 2.1. Content updates throughout the document.
*B	3508496	01/30/2012	PAVA	Added information on the devices that CY3215-DK supports in the Introduction chapter. Updated Figure 2-2.
*C	3915589	02/27/2013	KUK	Updated images in chapters 2, 3, and 4. Minor content updates throughout the document.
*D	4241128	01/09/2014	PAVA	No content updates; sunset review
*E	5629678	02/13/2017	AARA	Updated logo and copyrights/disclaimer. Moved revision history to the end of the document.