



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



## Contact us

Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China





CY3210-PSoCEVAL1

# PSoC<sup>®</sup> 1 Evaluation Kit Guide

Doc. #: 001-66768 Rev. \*E

Cypress Semiconductor  
198 Champion Court  
San Jose, CA 95134-1709  
Phone (USA): 800.858.1810  
Phone (Intl): 408.943.2600  
<http://www.cypress.com>

**Copyrights**

© Cypress Semiconductor Corporation, 2011-2015. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.

PSoC Designer™ is a trademark and PSoC® is a registered trademark of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

**Flash Code Protection**

Cypress products meet the specifications contained in their particular Cypress PSoC Data Sheets. Cypress believes that its family of PSoC products is one of the most secure families of its kind on the market today, regardless of how they are used. There may be methods, unknown to Cypress, that can breach the code protection features. Any of these methods, to our knowledge, would be dishonest and possibly illegal. Neither Cypress nor any other semiconductor manufacturer can guarantee the security of their code. Code protection does not mean that we are guaranteeing the product as 'unbreakable'.

Cypress is willing to work with the customer who is concerned about the integrity of their code. Code protection is constantly evolving. We at Cypress are committed to continuously improving the code protection features of our products.

# Contents



<b>1. Introduction</b>	<b>5</b>
1.1 Kit Contents .....	5
1.2 Additional Learning Resources .....	6
1.2.1 PSoC Designer .....	7
1.2.2 Code Examples .....	8
1.2.3 PSoC Designer Help .....	10
1.2.4 Technical Support.....	10
1.3 Documentation Conventions.....	12
<b>2. Getting Started</b>	<b>13</b>
2.1 Kit Installation .....	13
2.2 PSoC Designer .....	17
2.3 PSoC Programmer .....	18
<b>3. Kit Operation</b>	<b>19</b>
3.1 Evaluating the PSoC 1 Device.....	19
3.1.1 Programming Specifications and Connections .....	20
<b>4. Hardware</b>	<b>23</b>
4.1 CY3210-PSoCEVAL1 System Block Diagram.....	23
4.2 Functional Description .....	24
4.2.1 Power Supply System .....	24
4.2.2 Programming Interface.....	27
4.2.3 PSoC 1 Parts .....	27
4.2.4 RS-232 Interface .....	28
4.2.5 Prototyping Area.....	29
4.2.6 Character LCD Interface .....	30

<b>5. Code Examples</b>	<b>31</b>
5.1 My First Code Example .....	31
5.1.1 Project Description.....	31
5.1.2 Creating My First PSoC 1 Project.....	31
5.1.3 Verifying Output using LCD.....	46
5.1.4 Verifying Output using UART .....	47
5.2 Code Example 2: ASM_Example_ADC_UART_LCD.....	50
5.2.1 Project Description.....	50
5.2.2 Creating the Project .....	50
5.2.3 Hardware Connections .....	51
5.2.4 Code Example 2 Flowchart.....	52
5.2.5 Verifying Output using LCD.....	53
5.2.6 Verifying Output using UART .....	53
5.3 Code Example 3: ASM_Example_Blink_LED .....	55
5.3.1 Project Description.....	55
5.3.2 Hardware Connections .....	56
5.3.3 Code Example 3 Flowchart .....	56
5.3.4 Verifying Output .....	57
5.4 Code Example 4: ASM_Example_DAC_ADC .....	57
5.4.1 Project Description.....	57
5.4.2 Hardware Connections .....	59
5.4.3 Code Example 4 Flowchart.....	59
5.4.4 Verifying Output .....	60
5.5 Code Example 5: ASM_Example_Dynamic_PWM_PRS .....	61
5.5.1 Project Description.....	61
5.5.2 Hardware Connections .....	63
5.5.3 Code Example 5 Flowchart.....	64
5.5.4 Verifying Output .....	65
5.6 Code Example 6: ASM_Example_LED_Logic.....	66
5.6.1 Project Description.....	66
5.6.2 Hardware Connections .....	67
5.6.3 Code Example 6 Flowchart.....	67
5.6.4 Verifying Output .....	67
<b>A. Appendix</b>	<b>69</b>
A.1 Schematic.....	69
A.1.1 CY3210-PSoCEVAL1 Board Schematic.....	69
A.2 Board Layout .....	70
A.3 Bill of Materials .....	73
A.4 Replace CY8C29466-24PXI with CY8C27443-24PXI.....	74
<b>Revision History</b>	<b>77</b>
Document Revision History .....	77

# 1. Introduction



Thank you for your interest in PSoC<sup>®</sup> 1. This user guide will help you get started with the CY3210-PSoCEVAL1 PSoC 1 Evaluation Kit; provide hardware description, instructions on software installation, and kit operation; and walk you through the code examples.

The code examples implement commonly used peripherals such as analog-to-digital converter (ADC), digital-to-analog converter (DAC), universal asynchronous receiver/transmitter (UART), pulse-width modulator (PWM), pseudo random sequencer (PRS), and LCD. They are functional in PSoC Designer™, which is PSoC 1's GUI-based Integrated Design Environment (IDE). Peripherals in PSoC Designer are implemented as preprogrammed precharacterized 'User Modules'. Evaluate the library of over 100 user modules in PSoC Designer and experience simpler and faster designs.

The evaluation board features a pluggable character LCD module with contrast control, status LEDs, a potentiometer, push button switches, a UART, an RS-232 interface, an ISSP programming header, and prototyping area. The MiniProg unit, included in this kit is required to program PSoC 1 devices directly on the evaluation board. Kit schematics, layout, and bill-of-materials (BOM) are provided in the [Appendix on page 69](#).

## 1.1 Kit Contents

The CY3210-PSoCEVAL1 Evaluation Kit contains:

- PSoCEVAL1 evaluation board
- MiniProg programmer
- CY8C29466-24PXI 28-pin DIP sample
- CY8C27443-24PXI 28-pin DIP sample
- CY3210-PSoCEVAL1 kit DVD
  - PSoC Designer installation file
  - PSoC Programmer installation file
  - Bridge Control Panel installation file (packaged along with PSoC Programmer)
  - Code examples
  - Hardware files
  - Kit guide
  - Quick start guide
  - Release notes
- USB cable
- LCD module
- Single strand jumper wire pack

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

## 1.2 Additional Learning Resources

Cypress provides a wealth of data at [www.cypress.com](http://www.cypress.com) to help you to select the right PSoC device for your design, and to help you to quickly and effectively integrate the device into your design. For a comprehensive list of resources, see the knowledge base article “[How to Design with PSoC<sup>®</sup> 1, PowerPSoC<sup>®</sup>, and PLC – KBA88292](#)”. Following is an abbreviated list for PSoC 1:

- Overview: PSoC Portfolio, PSoC Roadmap
- Product Selectors: [PSoC 1](#), [PSoC 3](#), [PSoC 4](#), [PSoC 5LP](#)
- In addition, PSoC Designer includes a device selection tool.
- Application notes: Cypress offers a large number of PSoC application notes covering a broad range of topics, from basic to advanced level. Recommended application notes for getting started with PSoC 1 are:
  - [Getting Started with PSoC<sup>®</sup> 1 – AN75320](#)
  - [PSoC<sup>®</sup> 1 - Getting Started with GPIO – AN2094](#)
  - [PSoC<sup>®</sup> 1 Analog Structure and Configuration – AN74170](#)
  - [PSoC<sup>®</sup> 1 Switched Capacitor Analog Blocks – AN2041](#)
  - [Selecting Analog Ground and Reference – AN2219](#)

**Note:** For CY8C29X66 devices related Application note please click [here](#).

- Development Kits:
  - [CY3210-PSoCEval1](#) supports all PSoC 1 Mixed-Signal Array families, including automotive, except CY8C25/26xxx devices. The kit includes an LCD module, potentiometer, LEDs, and breadboarding space.
  - [CY3214-PSoCEvalUSB](#) features a development board for the CY8C24x94 PSoC device. Special features of the board include USB and CapSense development and debugging support.

**Note:** For CY8C29X66 devices related Development Kits please click [here](#).

The [MiniProg1](#) and [MiniProg3](#) devices provide interfaces for flash programming and debug.

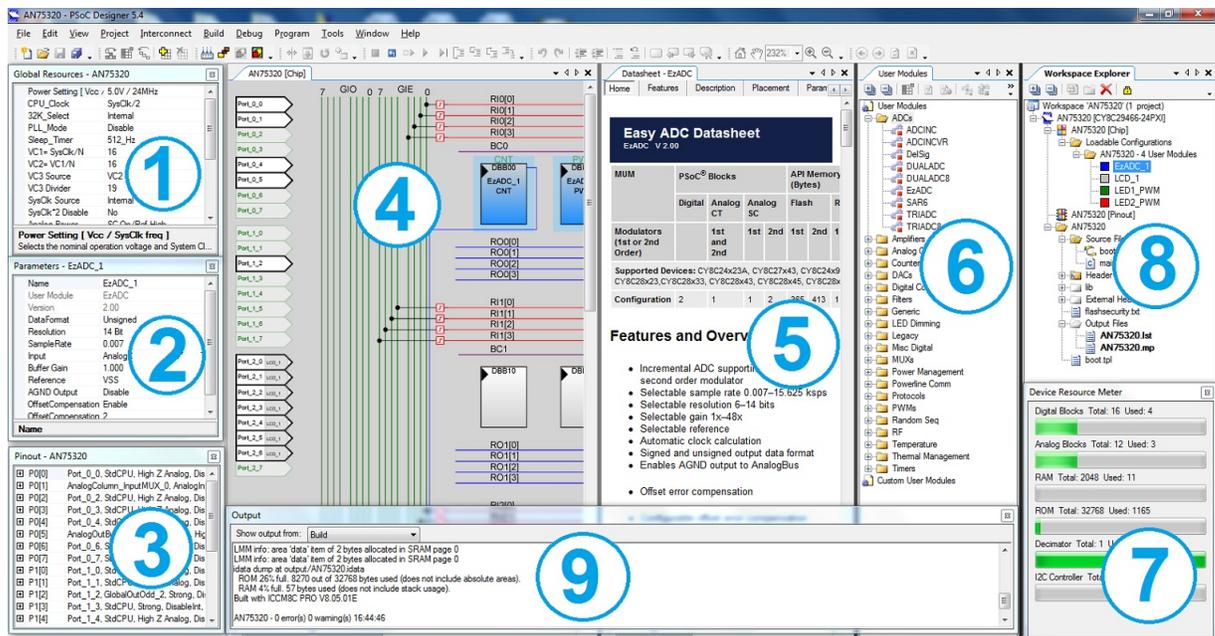
## 1.2.1 PSoC Designer

PSoC Designer is a free Windows-based Integrated Design Environment (IDE). Develop your applications using a library of pre-characterized analog and digital peripherals in a drag-and-drop design environment. Then, customize your design leveraging the dynamically generated API libraries of code. **Figure 1-1** shows PSoC Designer windows. **Note:** This is not the default view.

1. **Global Resources** – all device hardware settings.
2. **Parameters** – the parameters of the currently selected User Modules.
3. **Pinout** – information related to device pins.
4. **Chip-Level Editor** – a diagram of the resources available on the selected chip.
5. **Datasheet** – the datasheet for the currently selected UM
6. **User Modules** – all available User Modules for the selected device.
7. **Device Resource Meter** – device resource usage for the current project configuration.
8. **Workspace** – a tree level diagram of files associated with the project.
9. **Output** – output from project build and debug operations.

**Note:** For detailed information on PSoC Designer, go to PSoC® Designer > Help > Documentation > Designer Specific Documents > IDE User Guide.

Figure 1-1. PSoC Designer Layout



## 1.2.2 Code Examples

The following webpage lists the PSoC Designer based Code Examples. These Code Examples can speed up your design process by starting you off with a complete design, instead of a blank page and also show how PSoC Designer User modules can be used for various applications.

<http://www.cypress.com/go/PSoC1Code Examples>

To access the Code Examples integrated with PSoC Designer, follow the path **Start Page > Design Catalog > Launch Example Browser** as shown in [Figure 1-2](#).

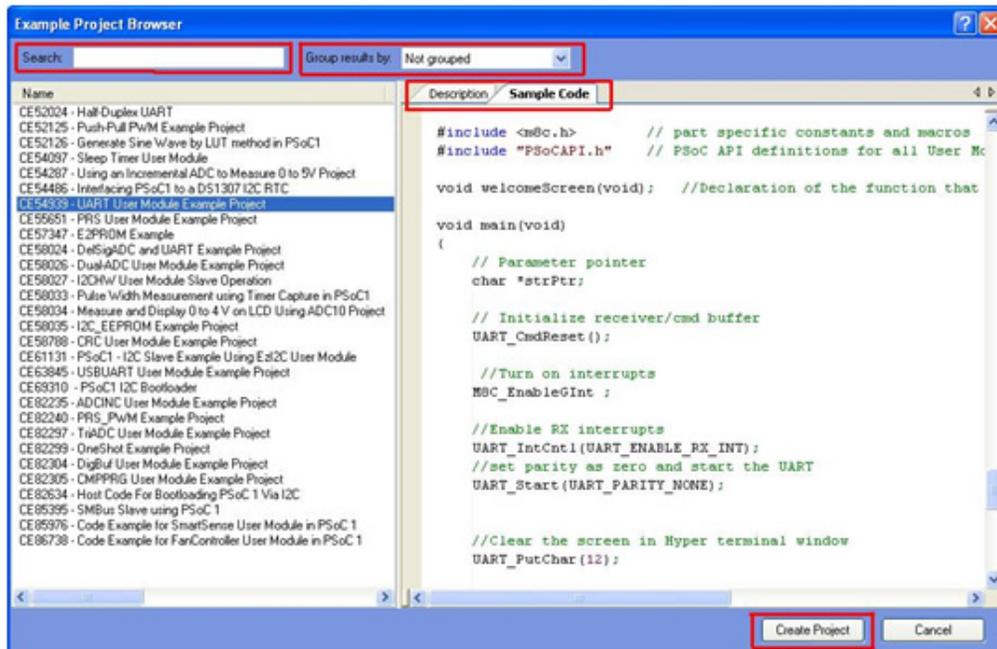
Figure 1-2. Code Examples in PSoC Designer



In the Example Projects Browser shown in [Figure 1-3](#), you have the following options.

- Keyword search to filter the projects.
- Listing the projects based on Category.
- Review the datasheet for the selection (on the Description tab).
- Review the code example for the selection. You can copy and paste code from this window to your project, which can help speed up code development, or
- Create a new project (and a new workspace if needed) based on the selection. This can speed up your design process by starting you off with a complete, basic design. You can then adapt that design to your application.

Figure 1-3. Code Example Projects, with Sample Codes



### 1.2.3 PSoC Designer Help

Visit the PSoC Designer home page to download the latest version of PSoC Designer. Then, launch PSoC Designer and navigate to the following items:

- **IDE User Guide:** Choose **Help > Documentation > Designer Specific Documents > IDE User Guide.pdf**. This guide gives you the basics for developing PSoC Creator projects.
- **Simple User module Code Examples:** Choose **Start Page > Design Catalog > Launch Example Browser**. These code examples demonstrate how to configure and use PSoC Designer User modules.
- **Technical Reference Manual:** Choose **Help > Documentation > Technical Reference Manuals**. This guide lists and describes the system functions of PSoC devices.
- **User module datasheets:** Right-click a User module and select "Datasheet." This datasheet explains the parameters and APIs of the selected user module.
- **Device Datasheet:** Choose **Help > Documentation > Device Datasheets** to pick the datasheet of a particular PSoC device.
- **Imagecraft Compiler Guide:** Choose **Help > Documentation > Compiler and Programming Documents > C Language Compiler User Guide.pdf**. This guide provides the details about the Imagecraft compiler specific directives and Functions.

### 1.2.4 Technical Support

If you have any questions, our technical support team is happy to assist you. You can create a support request on the [Cypress Technical Support page](#).

If you are in the United States, you can talk to our technical support team by calling our toll-free number: +1-800-541-4736. Select option 8 at the prompt.

You can also use the following support resources if you need quick assistance.

- [Self-help](#)
- [Local Sales Office Locations](#)

Visit <http://www.cypress.com> for additional learning resources in the form of datasheets, technical reference manual, and application notes. For the latest information about this kit, visit <http://www.cypress.com/go/CY3210-PSoCEval1>.

- *PSoC Designer: PSoC Designer Overview*  
<http://www.cypress.com/go/psocdesigner>
- *PSoC Designer Training: PSoC Designer On-Demand Training Series and Videos*  
<http://www.cypress.com/psoctraining>
- *PSoC Programmer, COM Hardware Layer Supported Languages*  
<http://www.cypress.com/go/psocprogrammer>
- *PSoC Programmable System-on-Chip™ Datasheets*  
<http://www.cypress.com/?mpn=CY8C29466-24PXI>  
<http://www.cypress.com/?mpn=CY8C27443-24PXI>
- [AN75320 - Getting Started with PSoC 1](#)  
This application note describes the capabilities of PSoC 1 devices and the PSoC Designer development environment used to configure and program these devices. An introductory project is included to help you develop PSoC 1 applications.
- [AN73212 - Debugging with PSoC 1](#)  
This application note introduces the elements of the PSoC 1 debugger system and explains how to configure and use them effectively.
- [AN2010 - PSoC 1 Best Practices and Recommendation](#)  
This application note provides introductory guidelines and best practices for developing PSoC 1 systems; it exposes some common mistakes designers make.
- [AN74170 - PSoC 1 Analog Structure and Configuration with PSoC Designer](#)  
This application note explains the analog structure of standard PSoC 1 devices and how the global analog parameters affect many of the analog user modules.
- [AN2027 - Using the PSoC Microcontroller External Crystal Oscillator](#)  
The external crystal oscillator in the PSoC microcontroller has specific requirements for correct operation in different configurations. This application note details these requirements.
- [AN2014\\_Introduction to PSoC 1 Programming](#)  
This application note discusses how to design an application to enable ISSP with PSoC Designer using either the device reset or power cycle programming mode.

## 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\
<i>Italics</i>	Displays file names and reference documentation: Read about the <i>sourcefile.hex</i> file in the <i>PSoC Designer User Guide</i> .
<b>[Bracketed, Bold]</b>	Displays keyboard commands in procedures: <b>[Enter]</b> or <b>[Ctrl] [C]</b>
File > Open	Represents menu paths: File > Open > New Project
<b>Bold</b>	Displays commands, menu paths, and icon names in procedures: Click the <b>File</b> icon and then click <b>Open</b> .
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 and configure the CY3210-PSoCEVAL1 kit.

### 2.1 Kit Installation

To install the kit software, follow these steps:

1. Insert the kit DVD into the DVD drive of your PC. The DVD 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/CY3210-PSoCEval1>. Three different types of installers are available for download.

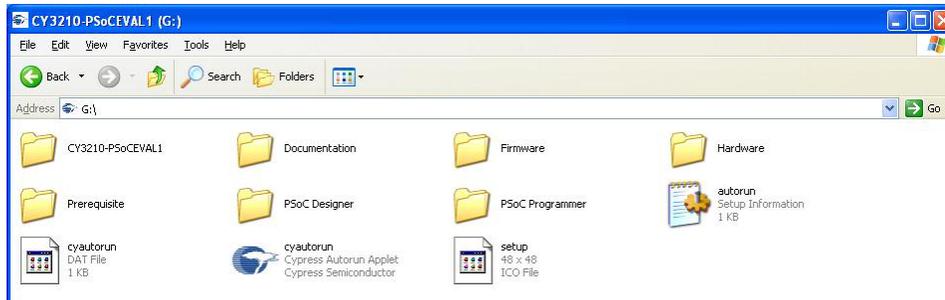
- a. CY3210-PSoCEval1\_ISO: This file (ISO image) is an archive file of the optical disc provided with the kit. You can use this to create an installer DVD or extract information using WinRar or similar tools.
  - b. CY3210-PSoCEval1\_Single Package: This executable file installs the contents of the kit DVD, which includes PSoC Programmer, PSoC Designer, kit code examples, kit hardware files, and user documents.
  - c. CY3210-PSoCEval1\_Single Package (without prerequisites): This executable file installs only the kit contents, which includes kit code examples, hardware files, and user documents.
2. Click **Install the CY3210-PSoCEVAL1** to start the kit installation, as shown in [Figure 2-1](#).

Figure 2-1. Kit Installer Startup Screen



**Note** If auto-run does not execute, double-click *cyautorun.exe* file on the root directory of the DVD, as shown in [Figure 2-2](#). To access the root directory, click **Start > My Computer > CY3210-PSoCEVAL1 <drive:>**.

Figure 2-2. Root Directory of DVD



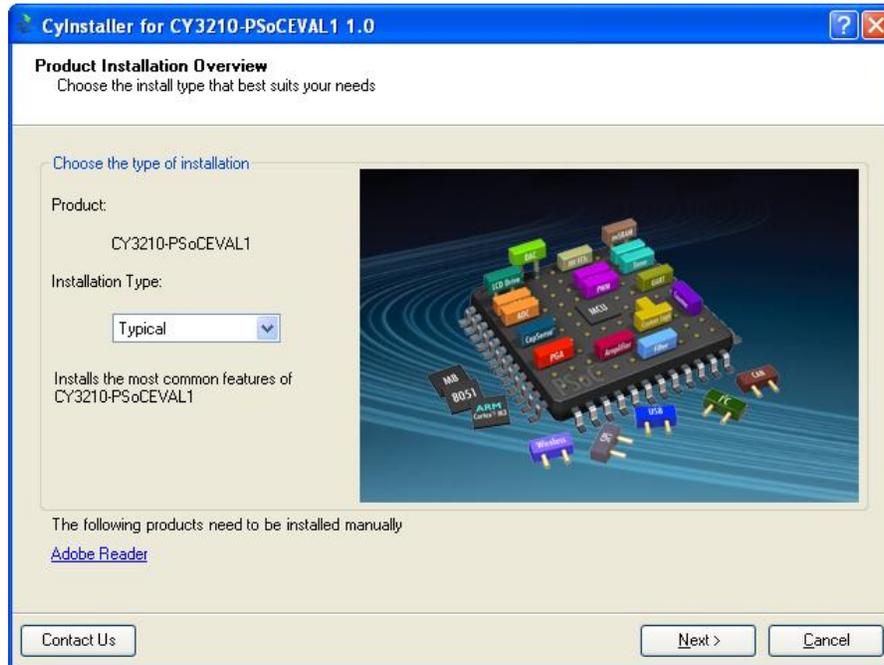
3. On the startup screen, click **Next** to start the installer.
4. The **InstallShield Wizard** screen appears. On this screen, choose the folder location to install the setup files. You can change the folder location for setup files using **Change**, as shown in [Figure 2-3](#).
5. Click **Next** to launch the kit installer.

Figure 2-3. InstallShield Wizard



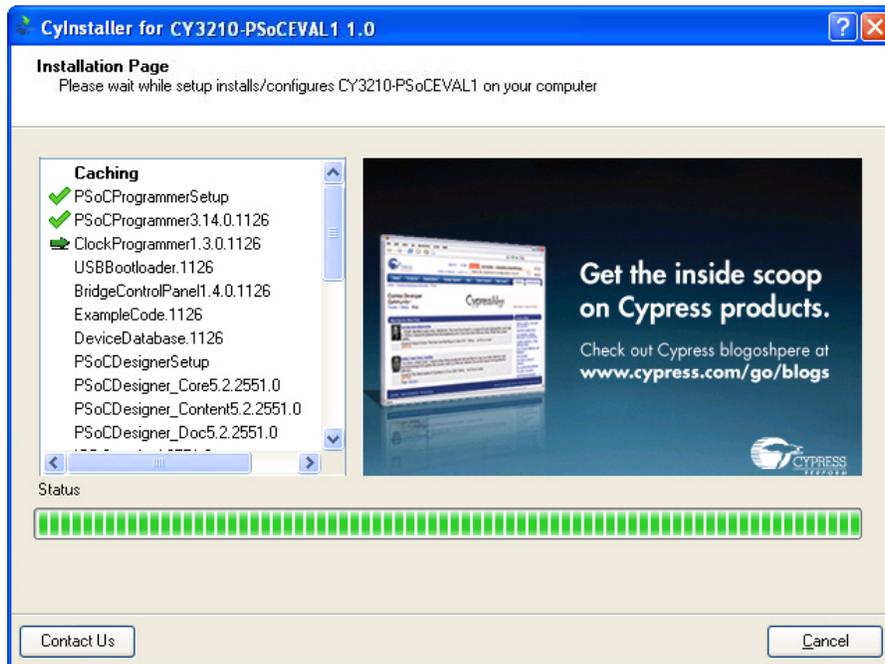
6. On the **Product Installation Overview** screen, select the installation type that best suits your requirement. The drop-down menu has three options: **Typical**, **Complete**, and **Custom**, as shown in [Figure 2-4](#). If you are uncertain, proceed with the default setting (Typical).
7. Click **Next** to start the installation.

Figure 2-4. Installation Type Options



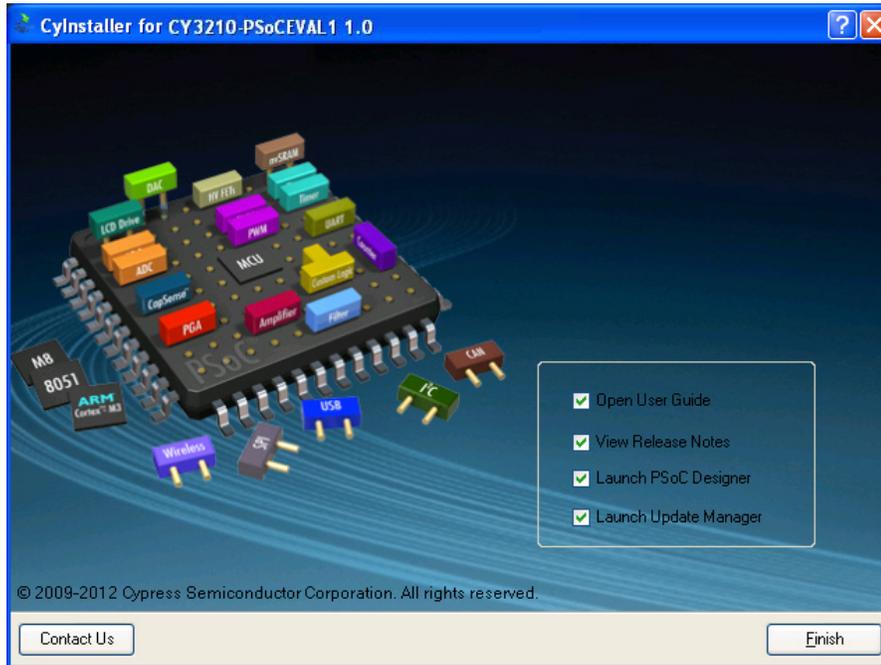
8. When the installation begins, a list of all packages appear on the **Installation Page**. A green check mark appears against every package that is downloaded and installed, as shown in Figure 2-5.
9. Wait until all the packages are downloaded and installed successfully.

Figure 2-5. Installation Page



10. Click **Finish** to complete the installation.

Figure 2-6. Installation Complete



After software installation, drivers are installed when MiniProg1 is connected to the PC for the first time. Verify driver installation by opening PSoC Programmer with the MiniProg connected to the USB port of the PC. The connected device will be listed under the Port Selection window in PSoC Programmer.

**Note** Advanced users can skip to [Code Examples on page 31](#).

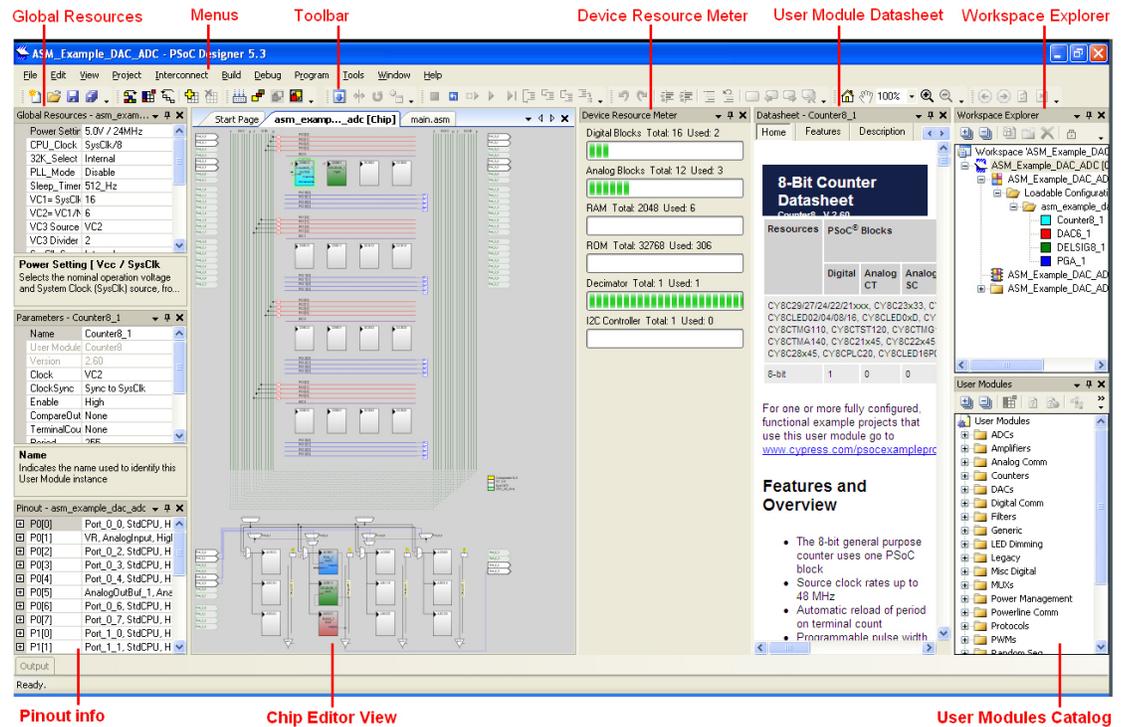
## 2.2 PSoC Designer

PSoC Designer is the integrated design environment (IDE) that you can use to customize your PSoC application. The latest version of PSoC Designer has several new features, bug fixes, and support for new PSoC devices.

This section gives a brief introduction on the PSoC Designer Interconnect View. Additional details on the PSoC Designer software is given in the [Code Examples chapter on page 31](#).

1. Click **Start > All Programs > Cypress > PSoC Designer <version> > PSoC Designer <version>**.

Figure 2-7. PSoC Designer Interconnect View



**Note** The **Datasheet** and **Resource Meter** windows are hidden by default. To include them in the view, click on **View** and select the required windows.

**Note** For more details on PSoC Designer, see the PSoC Designer IDE Guide at the following location: <Install\_directory>:\PSoC Designer\<version>\Documentation. You can also access this document via the Help menu (**Help > Documentation**).

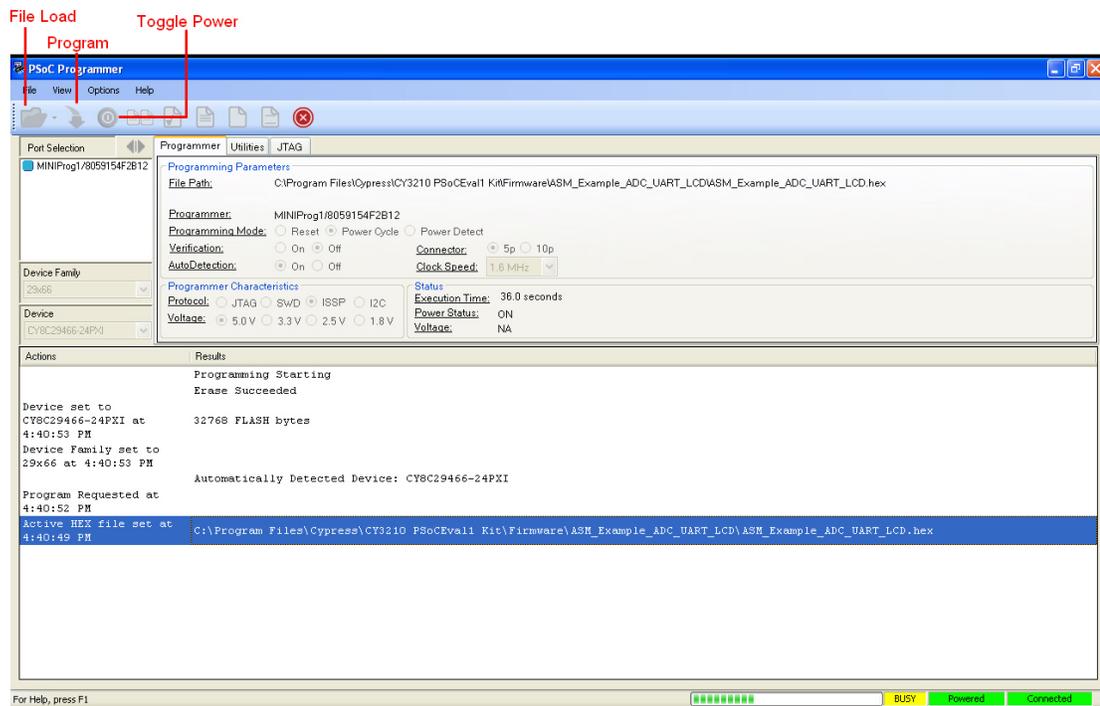
See [Additional Learning Resources on page 6](#) for links to PSoC Designer training. The PSoC Designer quick start guide is available at: <http://www.cypress.com/?rID=47954>

## 2.3 PSoC Programmer

This section gives a brief introduction on programming PSoC; for kit specific programming instruction, see [Programming Specifications and Connections on page 20](#).

1. Click **Start > All Programs > Cypress > PSoC Programmer <version> > PSoC Programmer <version>**.
2. Select the MiniProg from **Port Selection**.

Figure 2-8. PSoC Programmer Window



3. Click the **File Load** button to load the hex file.
4. Use the **Program** button to program the hex file on to the chip.
5. When programming is successful, the “Programming Succeeded” message appears in the Action pane.
6. 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. Kit Operation



The CY3210-PSoCEVAL1 kit helps in evaluating applications using the PSoC 1 family of devices. This kit can be used to work with the different user modules provided in the PSoC Designer software and explore hardware features that are integrated into the PSoC 1 device.

### 3.1 Evaluating the PSoC 1 Device

The CY3210-PSoCEVAL1 board is populated with a preprogrammed CY8C29466-24PXI part. To evaluate the default project programmed on the board, make the following hardware connections:

- Connect the jumper (shunt) at JP1 and JP2.
- Connect the potentiometer (VR on connector J5) and port P0[1] on connector J6 using one of the single-strand jumper wires (see [Figure 3-1](#)). This connects one of the PSoC pins to the potentiometer.

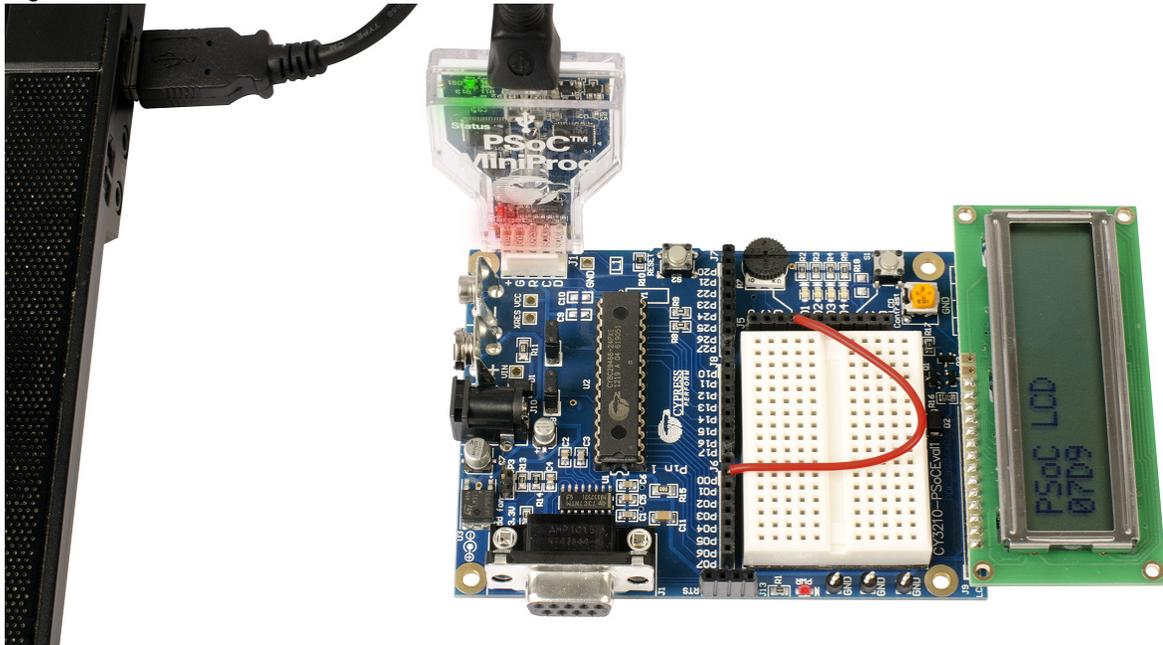
After the connections are made, plug in the MiniProg to the ISSP header. Connect the MiniProg to the PC using the USB cable provided. Power the kit using PSoC Programmer. For more information on using PSoC Programmer, see [Programming Specifications and Connections on page 20](#). The CY3210-PSoCEVAL1 can be powered from a DC supply jack, battery terminals, or using the MiniProg. The PWR LED5 lights up (red) when the board is powered.

You can vary the analog input by varying the potentiometer (R7). The output is displayed on the LCD module. LCD contrast control potentiometer (R6) can be used to vary the LCD contrast.

The PSoC MiniProg gives you the ability to program PSoC parts quickly and easily. It is small and compact, and connects to your PC using the USB cable. During prototyping, the MiniProg can be used as in-system serial programming (ISSP) to program PSoC devices, as shown in [Figure 3-2](#).

MiniProg1 does not support I2C communication. The [CY3240-I2USB](#) and [MiniProg3](#) can be used as a USB-I2C Bridge for debugging I2C serial connections and communicating to PSoC devices.

Figure 3-1. PSoCEVAL1 Evaluation Board



### 3.1.1 Programming Specifications and Connections

When the MiniProg is connected, you can use PSoC Programmer to program the CY3210-PSoCEVAL1 Evaluation kit. Plug in the USB cable into the MiniProg before attaching it to the ISSP header on the board. When using a USB cable with MiniProg, keep the length under six feet to avoid signal integrity issues.

When using MiniProg, the LEDs blink at a variable rate to track connection status. The green LED near the USB connector turns on after MiniProg is plugged into the computer and is configured by the operating system. If MiniProg cannot find the correct driver in the system, this LED does not turn on. After the device is configured, the LED stays on at about a 4-Hz blink rate. This changes during programming, where the blink duty cycle increases.

The red LED (Figure 3-2) at the bottom turns on when the MiniProg powers the part. The LED is off when power is provided by the target board.

Figure 3-2. Programming PSoC Device

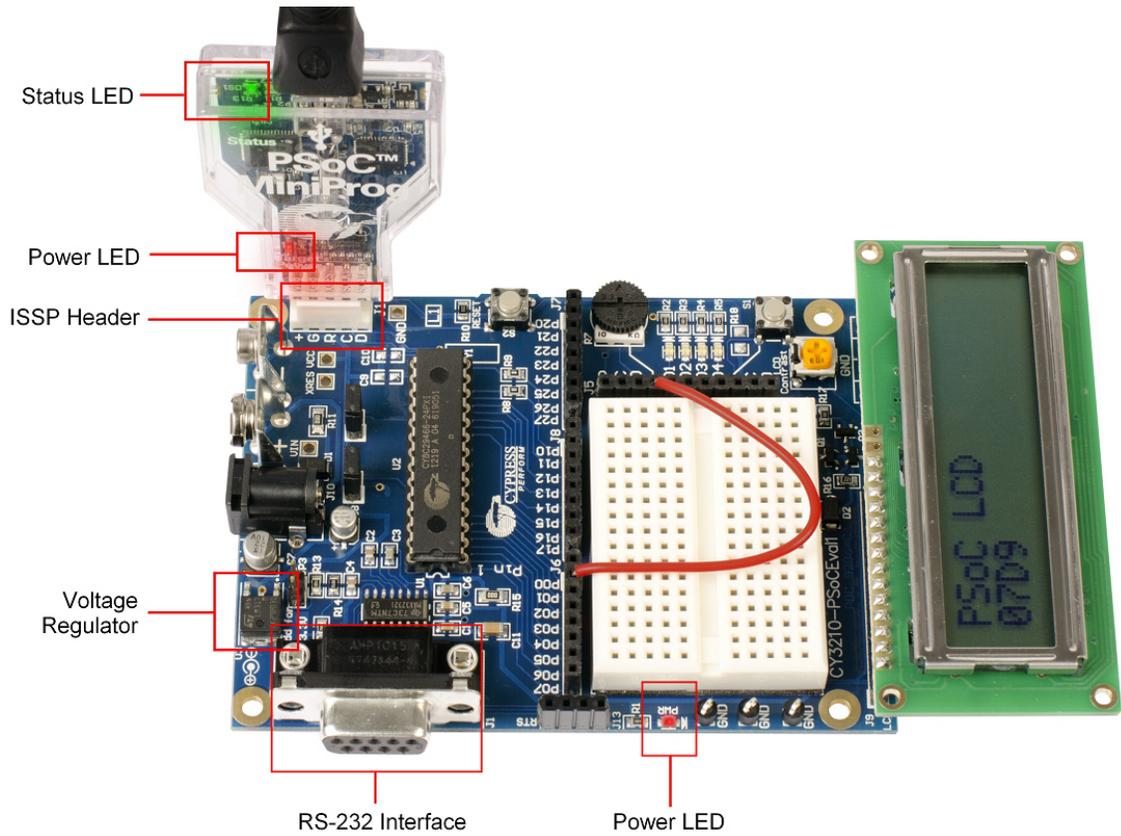
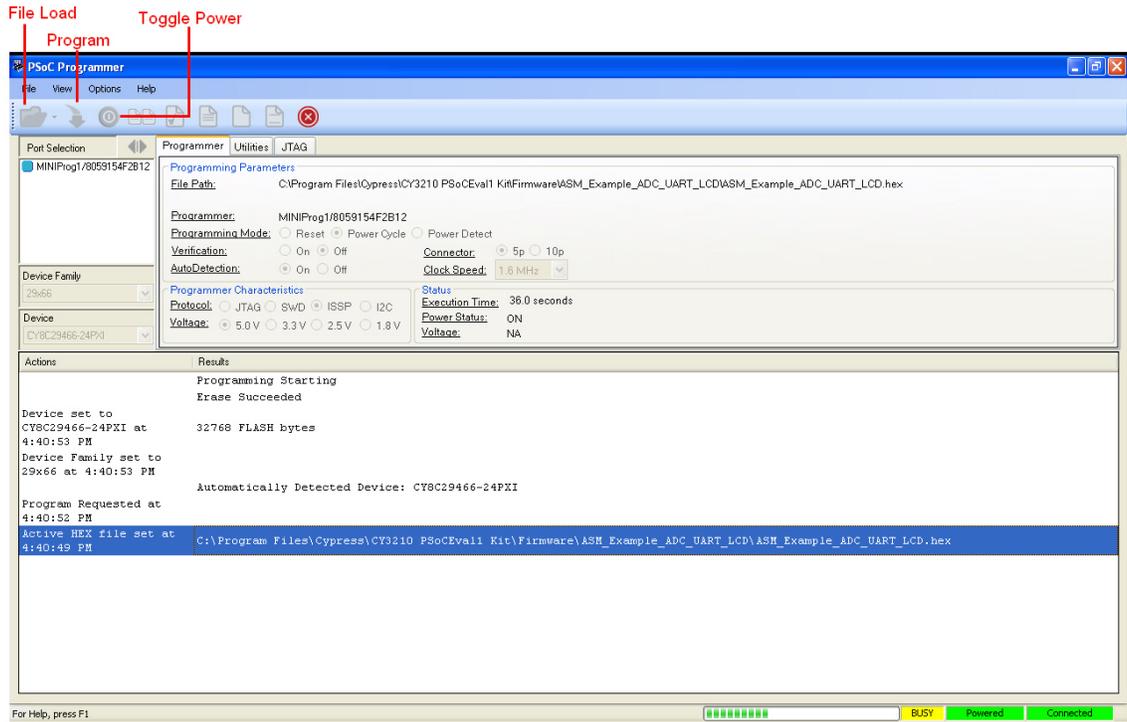


Figure 3-3. PSoC Programmer Screen



Follow these steps to program using MiniProg1:

1. Connect the MiniProg1 to the PC using the USB cable.
  2. Plug in the MiniProg1 to the ISSP header on the CY3210-PSoCEVAL1 board.
  3. When USB is connected to the MiniProg1, LED (green) glows in the MiniProg1.
  4. Open PSoC Programmer.
  5. Click the **Load File** button and browse to the hex file location  
(<Install\_directory>:\Cypress\CY3210-PSoCEVAL1\<version>\Firmware\  
ASM\_Example\_ADC\_UART\_LCD\ASM\_Example\_ADC\_UART\_LCD.hex). Click **Open** to select  
the hex file.
  6. Click **Connect** or double-click on the respective MiniProg under Port Selection to select or connect to MiniProg.
  7. Make sure the **Power Cycle** radio button is selected to power the kit using USB power.
  8. Click **Program** or press **[F5]** to initiate programming.
  9. The green LED on the MiniProg1 blinks to indicate the progress of programming.
  10. After successful programming, the red LED on MiniProg1 is powered off.
  11. Select the **Toggle Power** button in PSoC Programmer to power the board and verify output.
- See the section [Replace CY8C29466-24PXI with CY8C27443-24PXI on page 74](#) for instructions on how to replace the CY8C29466-24PXI PSoC part with CY8C27443-24PXI.

# 4. Hardware



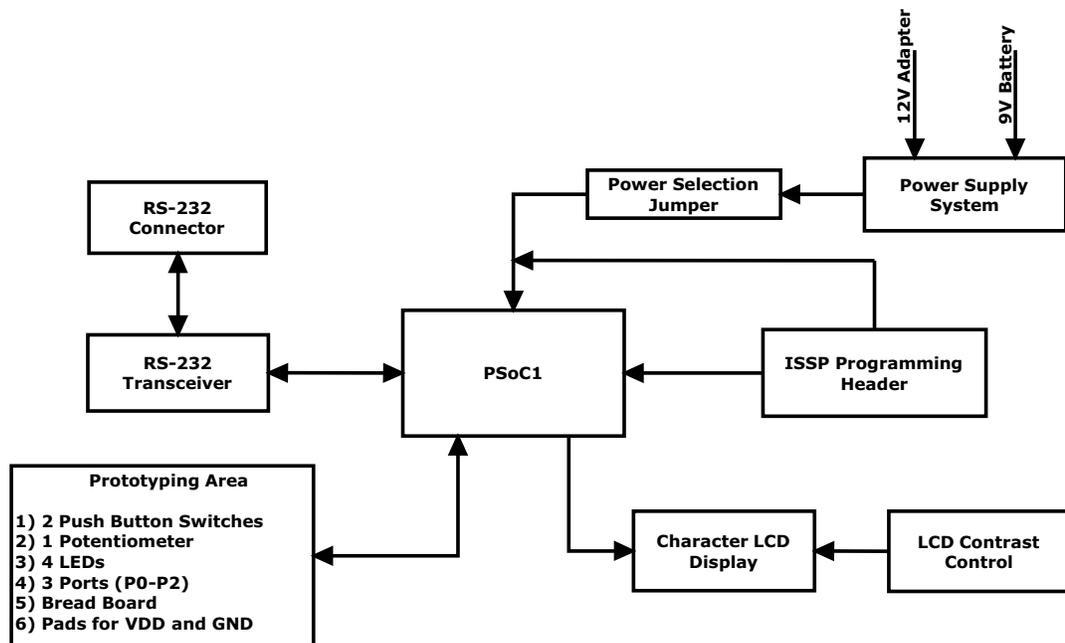
This section provides an overview of the development kit hardware, including power system, jumper setting, and programming interface. To start using the board, go to [Code Examples on page 31](#).

## 4.1 CY3210-PSoCEVAL1 System Block Diagram

The CY3210-PSoCEVAL1 Evaluation Kit consists of:

- Power supply system
- Programming interface
- PSoC 1
- RS-232 interface
- Prototyping area
- Character LCD interface

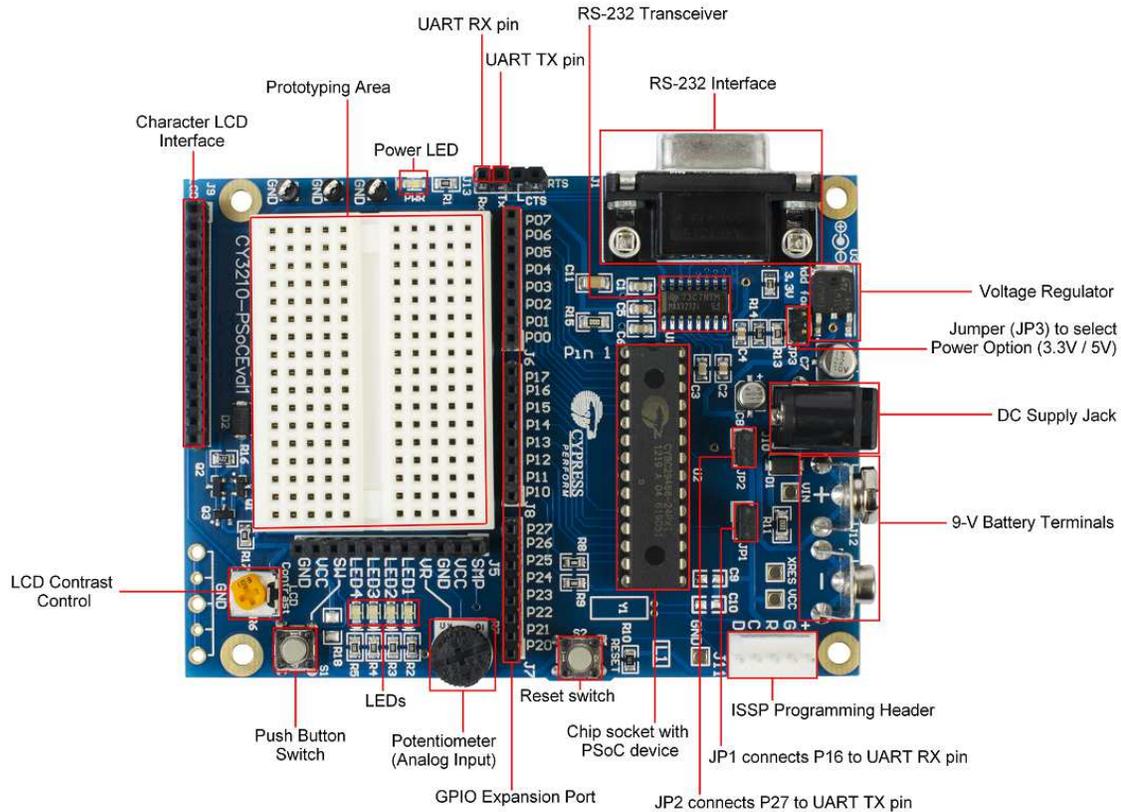
Figure 4-1. System Block Diagram



## 4.2 Functional Description

The CY3210-PSoCEVAL1 Evaluation Kit demonstrates the function of PSoC 1 devices. Connect the device to onboard peripherals such as potentiometer, LEDs, LCD, and RS-232. The board also has additional features such as a general prototyping area (bread board) and an ISSP programming header. It provides different voltage domains; see [4.2.1 Power Supply System](#).

Figure 4-2. CY3210-PSoCEVAL1 Evaluation Board



### 4.2.1 Power Supply System

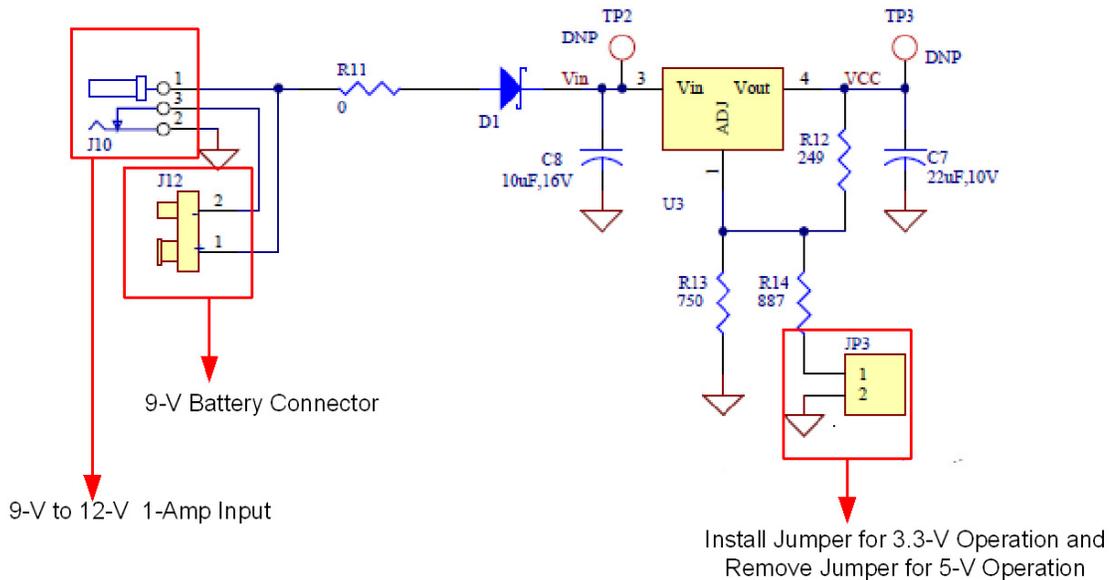
The power supply system on this board is versatile. The kit can be powered in three ways: using a MiniProg unit, DC wall adaptor, or a 9-V battery. The DC wall adaptor must be of 9 V to 12 V, 1-A rating unit. The onboard voltage regulator converts input 9 V/12 V into 3.3 V/ 5 V. Selection between 3.3 V and 5 V is done using jumper JP3. When the jumper (shunt) is not connected, the regulator gives the output voltage of 5 V; installing the jumper gives a 3.3-V output.

**Note** Use only one power supply at a time. Do not use a power supply that is less than 7 V or exceeds 12 V.

If the board needs to be powered via external 3.3-V or 5-V supply, the power input should be connected to the VCC sockets of the J5 connector ([Figure 4-9](#)). The external voltage must be less than 5.5 V.

**Warning** Any voltage input greater than 12 V connected on these headers will permanently damage the board components.

Figure 4-3. Power Supply Schematic



#### 4.2.1.1 Protection Circuit

A reverse-voltage and over-voltage protection circuit is added at the expansion port on VCC lines. The protection circuit consists of two P-channel MOSFETs on the power line allowing the power/current to flow from input to output depending on the voltages applied at the external board connector. Figure 4-4 is the protection circuit placed between the VCC domain near the prototype board (on J5 connector) and the onboard components. This circuit disconnects or behaves similar to an open switch if an external voltage above 5.6 V is applied on the VCC pins of the J5 connector.

**Warning** Any voltage input greater than 12 V connected on these headers will permanently damage the board components.

Figure 4-4. Protection Circuit

