CY3280-20x66

# Universal CapSense® Controller Kit Guide

Doc. No. 001-67447 Rev. *G

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone (USA): 800.858.1810
Phone (Intnl): 408.943.2600
www.cypress.com

## Copyrights

# Contents

# 1. Introduction

Thank you for your interest in the CY3280-20x66 Universal CapSense® Controller Kit. This kit is designed for easy prototyping and debug of the 20xx6A/S CapSense family designs using predefined control circuitry and plug-in hardware. The kit includes programming hardware and a CY3240-I2USB bridge for tuning and data acquisition. This module connects to any CY3280 Universal CapSense Module board.

This document describes the CY3280-20x66 Universal CapSense Controller Kit and describes the code examples provided with the kit. 'CY3280_20x66 CSD PD Project1' is the default project programmed on the CY3280-20x66 Universal CapSense Controller board. For more information on this project, see the Code Examples chapter on page 35.

## 1.1 Kit Contents

The CY3280-20x66 Universal CapSense Controller Kit contains:

- CY3280-20x66 Universal CapSense Controller board
- CY3240-I2USB bridge board
- MiniProg1 programmer
- USB A to Mini-B cable
- CY3280-20x66 Universal CapSense Controller Kit CD
  - 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

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

### 1.1.1 Prerequisites

The following are required for the functioning of this kit:

- CY3280 Universal CapSense Module board or custom board similar to CY3280 module boards (see CY3280 Universal CapSense Module Boards on page 6)
- 12-V DC adapter (optional)
- CY3215-DK (optional, required to debug programs in PSoC Designer)

Universal CapSense Module boards are available for purchase separately or as part of the combination kits. Visit www.cypress.com/shop for more information.

### 1.1.2 MiniProg1 Programmer

The PSoC MiniProg gives you the ability to program PSoC parts quickly and easily. It is small, compact, and connects to your PC using the provided USB 2.0 cable. During prototyping, the MiniProg can be used as an in-system serial programmer (ISSP) to program PSoC devices on your PCB.

When the MiniProg is connected, you can use PSoC Programmer to program. PSoC Programmer is a free software, which can either be launched from within PSoC Designer or run as a standalone program.

### 1.1.3 CY3240-I2USB Bridge

The I2USB bridge allows you to test, tune, and debug hardware and software of a PSoC application by bridging the USB port to I2C. Populated with the CY8C24894 PSoC device, the I2USB bridge can be connected through the ISSP pins on the controller board.

### 1.1.4 CY3280 Universal CapSense Module Boards

CY3280 Universal CapSense Module boards are available for purchase separately or as part of the combination kits. The CY3280-20x66 Universal CapSense Controller can be connected to CY3280-SLM, CY3280-SRM, CY3280-BMM, and CY3280-BSM CapSense boards.

## 1.2 PSoC Designer

PSoC Designer is the integrated development environment (IDE) used to customize your PSoC application. More information about PSoC Designer is available in the PSoC Designer IDE Guide; go to: <Install_directory>\PSoC Designer\<version>\Documentation.

Default <Install_directory> on Windows 32-bit OS is C:\Program Files\Cypress and on 64-bit OS it is C:\Program Files (x86)\Cypress.

## 1.3 PSoC Programmer

PSoC Programmer offers a simple GUI to configure and program PSoC devices.

## 1.4 Bridge Control Panel

The Bridge Control Panel GUI is used with the CY3240-I2USB bridge to enable communication with I2C slave devices; here, it is used with the CY3280-20x66 Universal CapSense Controller. This software is used to configure I2C devices as well as acquire and process data received from I2C slave devices. The Bridge Control Panel helps in optimizing, debugging, and tuning the target devices.

## 1.5 Additional Learning Resources

Cypress provides a wealth of data at 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® 1, PowerPSoC®, 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
- 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® 1 – AN75320
  - PSoC® 1 - Getting Started with GPIO – AN2094
  - PSoC® 1 Analog Structure and Configuration – AN74170
  - PSoC® 1 Switched Capacitor Analog Blocks – AN2041
  - Selecting Analog Ground and Reference – AN2219

**Note:** For application notes related to CY8C29X66 devices, 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 development kits related to CY8C29X66 devices, click here.

The MiniProg1 and MiniProg3 devices provide interfaces for flash programming and debug.

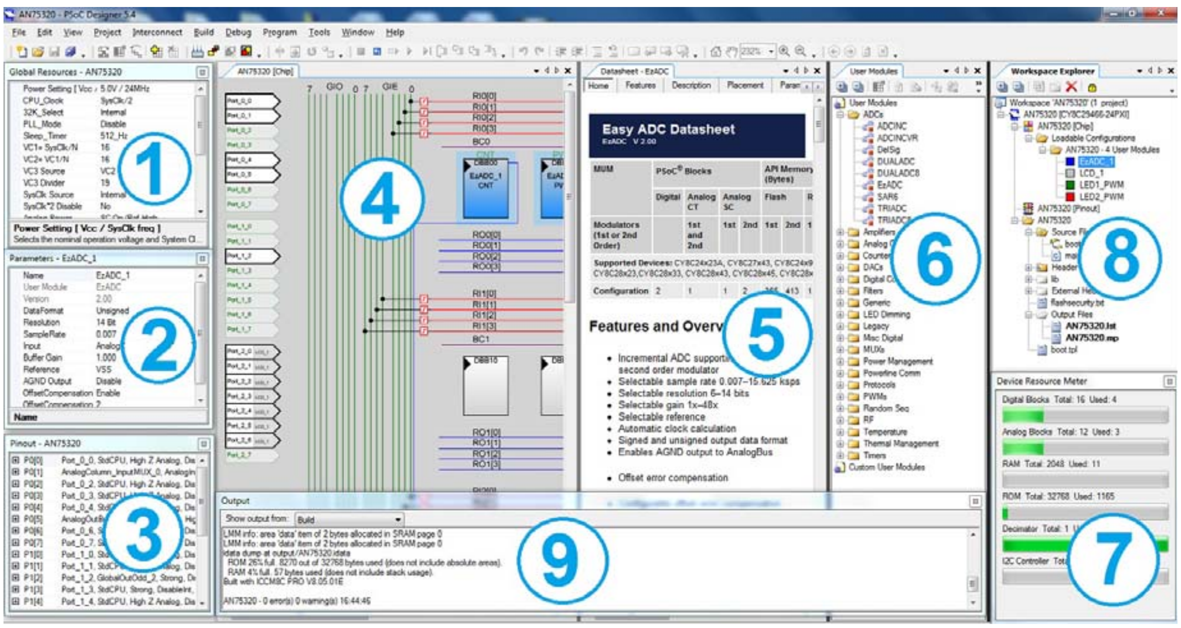
## 1.5.1 PSoC Designer

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



### 1.5.2 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.

www.cypress.com/go/CapSenseCodeExamples

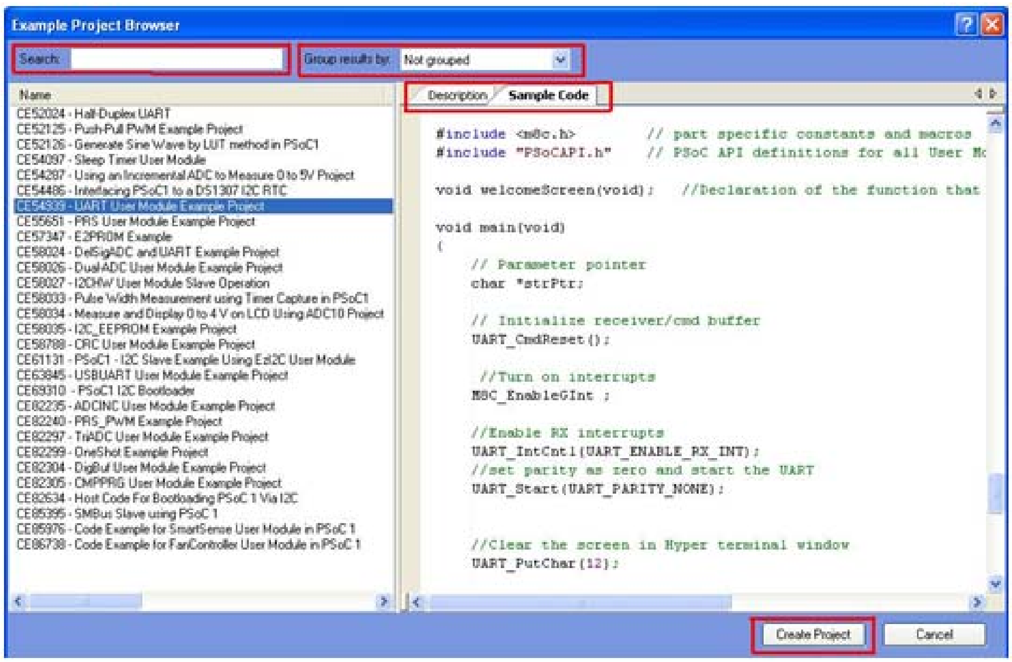
To access the Code Examples integrated with PSoC Designer, follow the path **Start Page** > **Design Catalog** > **Launch Example Browser** as shown in Figure 1-2.

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



### 1.5.3 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.

### 1.5.4 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

## 1.6 Documentation Conventions

Table 1-1. Document Conventions for Guides

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

# 2. Getting Started

This chapter describes the installation and configuration of the CY3280-20x66 Universal CapSense Controller Kit.

## 2.1 Kit Installation

To install the kit software, follow these steps:

1. Insert the kit CD into the CD 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 www.cypress.com/go/CY3280-20x66. Three different types of installers are available for download.

   a. CY3280-20X66 Universal CapSense Controller Kit (Rev *B): This executable file installs only the kit contents, which includes kit code examples, hardware files, and user documents.

   b. CY3280-20X66 KIt Installer (Single Package): This executable file installs PSoC Programmer, PSoC Designer, kit code examples, kit hardware files, and user documents.

   c. CY3280-20X66 Universal CapSense Controller Kit (Rev *B): 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. This includes the kit installer along with PSoC Programmer, PSoC Designer, and other mandatory non-cypress prerequisites.

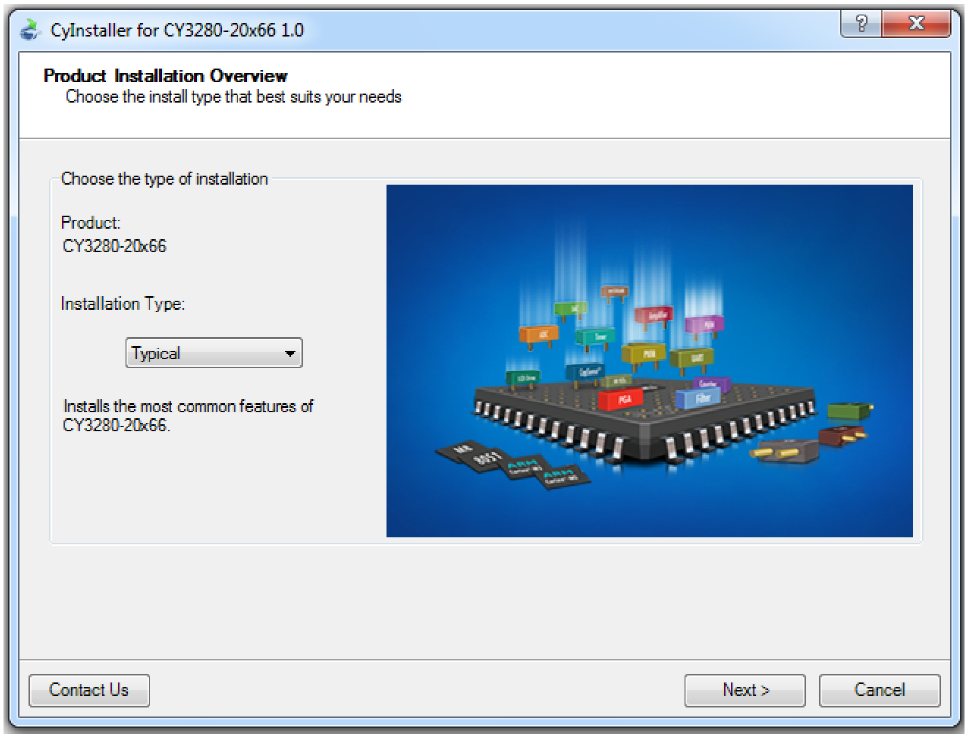
2. Click **Install the CY3280-20x66** to start the installation, as shown in Figure 2-1.

Figure 2-1. Kit Installer Startup Screen


**Note** If auto-run does not execute, double-click *cyautorun.exe* in the root directory of the CD.

3. The **Install Shield Wizard** screen appears. The default setup location is shown on the Install Shield Wizard screen. You can change the location using **Change**.
4. Click **Next** to launch the kit installer.
5. In 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 2-2.
6. Click **Next** to start the installation.

Figure 2-2. Installation Type Options




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

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

Figure 2-3. Installation Completion Page

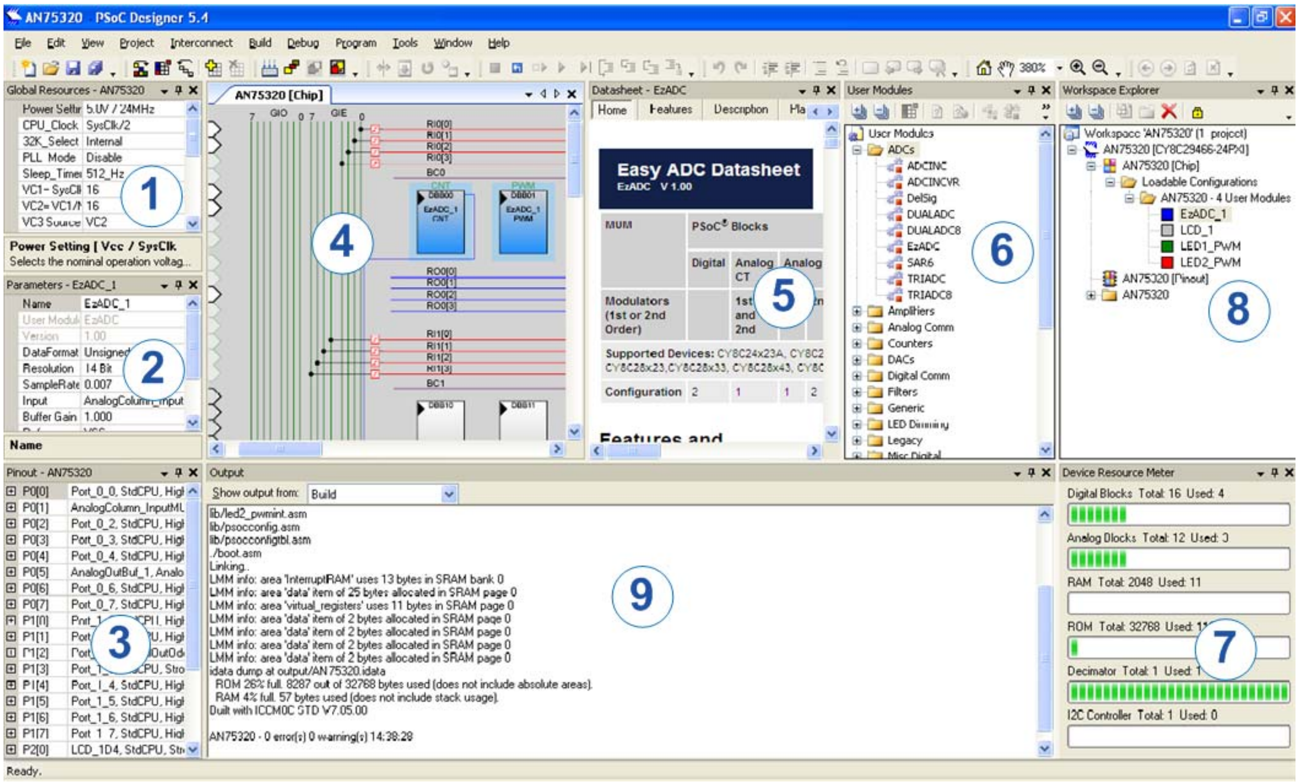

After software installation, verify your setup by opening PSoC Programmer with the MiniProg attached to the PC. Open Bridge Control Panel with the I2USB bridge attached to the PC to verify driver installation.

**Note** Advanced users can go to the Code Examples chapter on page 35.

## 2.2 PSoC Designer

1. Click **Start** > **All Programs** > **Cypress** > **PSoC Designer <version>** > **PSoC Designer <version>**.
2. Click **File** > **New Project** to create new project; click **File** > **Open Project** to work with an existing project.

Figure 2-4. PSoC Designer Interconnect View




3. To experiment with the code examples, go to the Code Examples chapter on page 35.

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

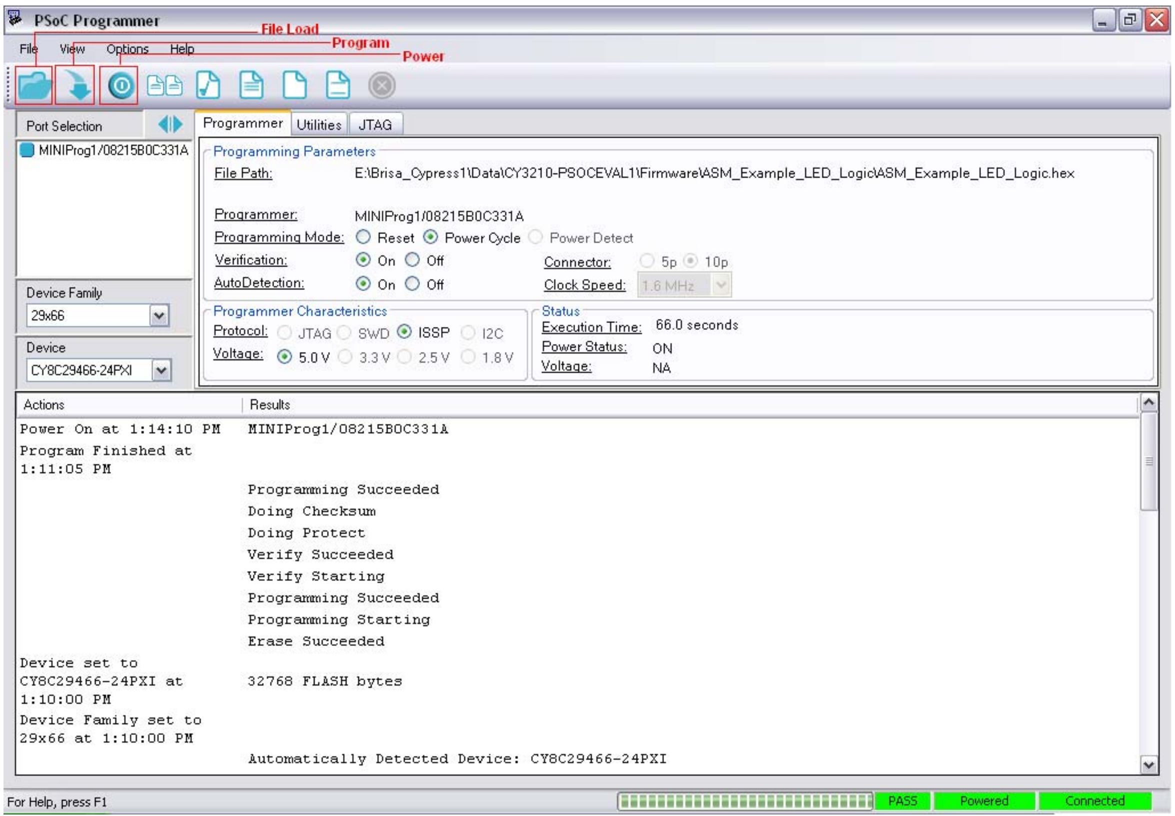
See Additional Learning Resources on page 7 for links to PSoC Designer training.

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

## 2.3 PSoC Programmer

1. Click **Start** > **All Programs** > **Cypress** > **PSoC Programmer <version>** > **PSoC Programmer <version>**.
2. Select the MiniProg from the port selection, as shown in Figure 2-5.

Figure 2-5. PSoC Programmer Window



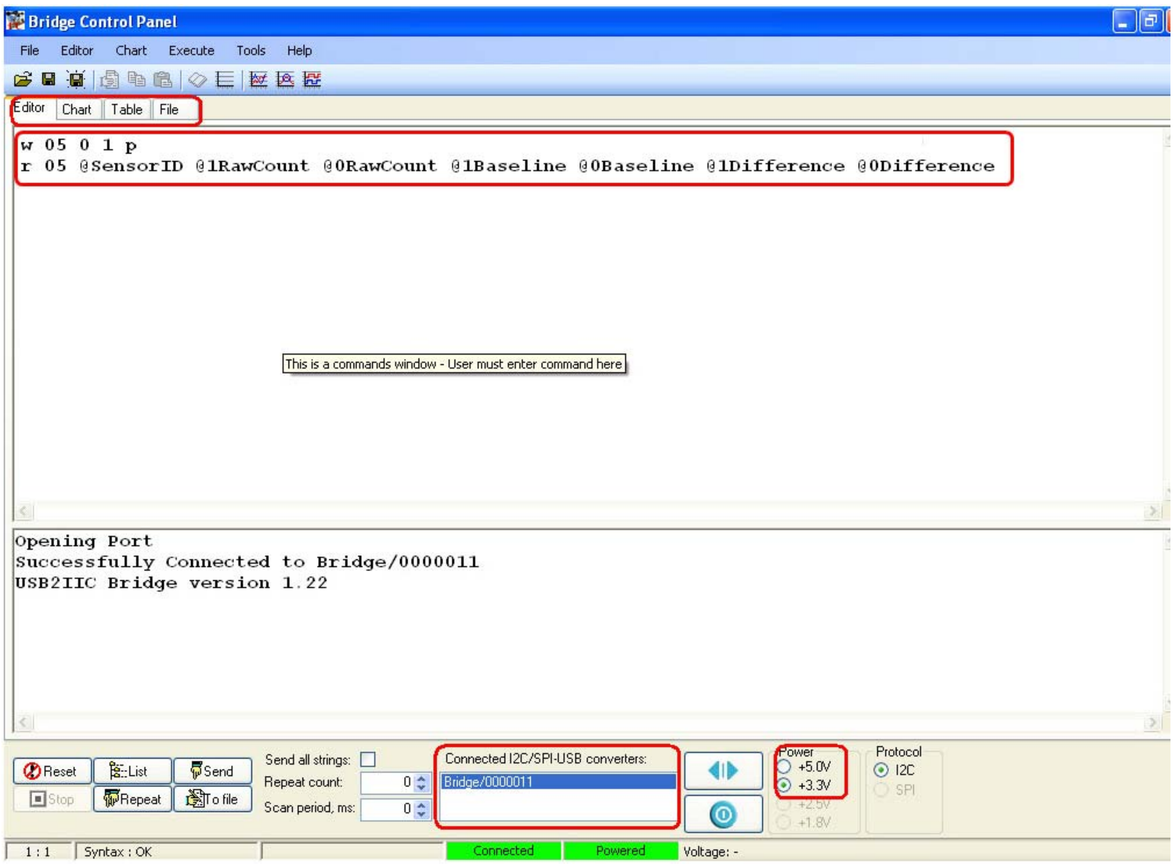

3. Click the **File Load** button from the Programmer menu bar, navigate and select the hex file.
4. Use **Program** button to program the hex file on to the chip.
5. When programming is successful, **Programming Succeeded** appears in the Actions pane.
6. Close PSoC Programmer.

**Note** For more details on PSoC Programmer, go to the Programmer user guide at: <Install_directory>\Cypress\Programmer\<version>\Documents.

## 2.4 Bridge Control Panel

1. Click **Start** > **All Programs** > **Cypress** > **Bridge Control Panel <version>** > **Bridge Control Panel <version>**.
2. Select **5.0V** from **Power Setting** box, as highlighted in Figure 2-6.
3. Select the device to be connected from the port window.
4. Click the **Variable Settings** option from the **Chart** menu, click **Load**, navigate to and open the *.ini file.
5. Select **File > Open**; navigate to and open the *.iic file. The iic file contents appear on the Editor pane of the Bridge Control Panel.

Figure 2-6. Selecting the Bridge




**Note** For more details on the Bridge Control Panel, view the help topics from the Bridge Control Panel Menu bar.

# 3. Kit Operation

## 3.1 Introduction

The CY3280-20x66 Universal CapSense Controller (UCC) connects to any CY3280 Universal CapSense Module board. This kit requires PSoC Designer for development, PSoC Programmer for programming, and Bridge Control Panel to test and tune the CapSense parameters.

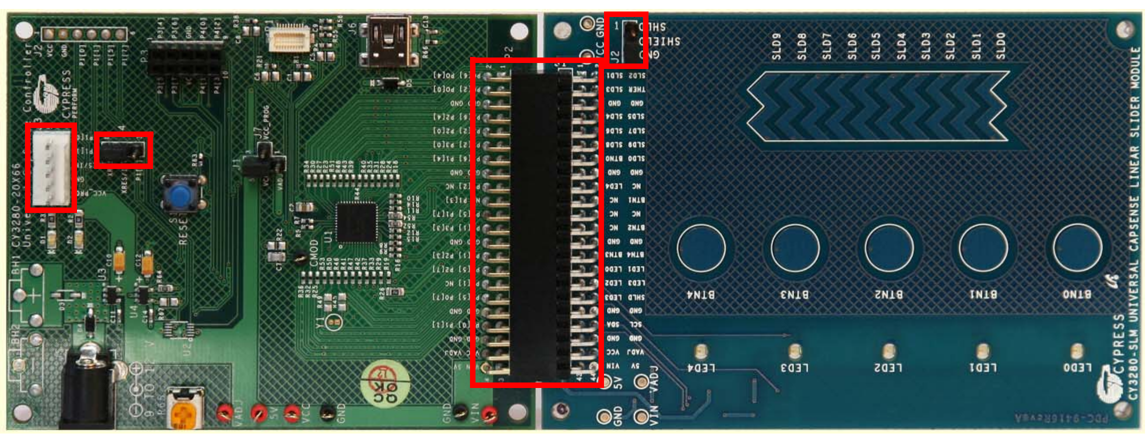
## 3.2 Hardware Requirement

- CY3280-20x66 Universal CapSense Controller board (PSoC Device ID: CY8C20666A-24LTXI)
- CY3280-Universal CapSense Linear Slider board or a similar CapSense module board
- MiniProg1
- CY3240-I2USB bridge
- USB A to Mini-B cable

## 3.3 Connecting CapSense Module Board

The CY3280-Universal CapSense Linear Slider Module Board can be connected to the UCC through Port P2 of the UCC board and J1 of the CY3280 Universal CapSense Linear Slider Module. The connection is as shown in the following figure.

Figure 3-1. Connect CapSense Module Board to Universal CapSense Controller

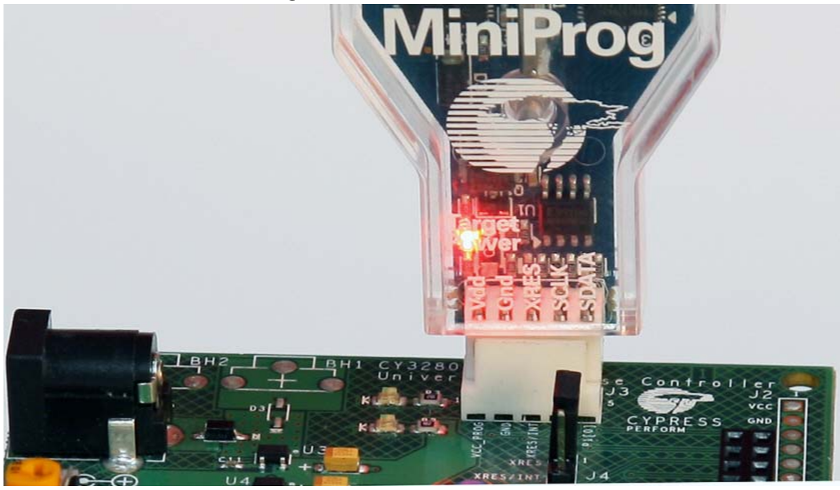


On the CY3280-20x66 Universal CapSense Controller board, place shunts on pins 1 and 2 of J4 (XRES select) to connect the XRES pin of PSoC to the XRES on programming header. On the CY3280-SLM Universal CapSense Linear Slider Module, place a shunt on pins 2 and 3 of J2 to connect the shield electrode to ground.

## 3.4 Programming PSoC with New Design

The CY3280-20x66 Universal CapSense Controller is programmed using a MiniProg1 provided with the kit.

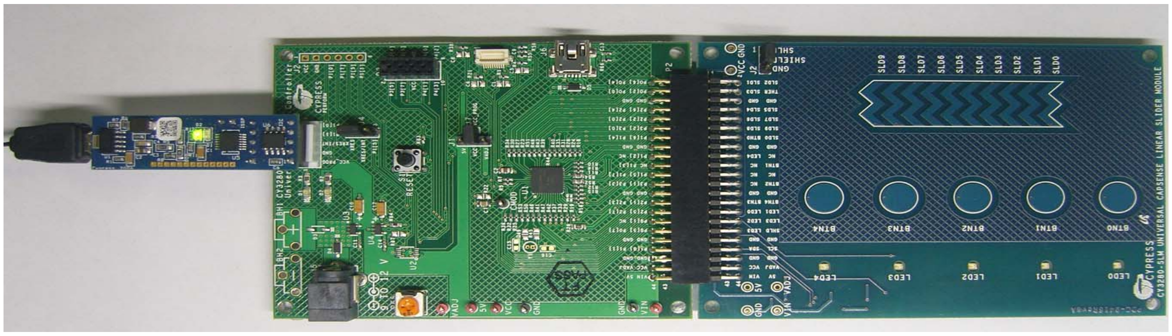
Figure 3-2. Connect MiniProg to Controller




1. Connect the USB port of your PC to the CY3280-20x66 ISSP connector (J3) using the PSoC MiniProg and a USB cable (A to Mini-B). Programming can be done using PSoC Programmer.
2. On the UCC, connect J7 to pin 2 of J1. This enables power supply by the MiniProg1.
3. Toggle the **Power** button on PSoC Programmer.
4. LED **D1** on the CY3280-20x66 Universal CapSense Controller is ON if powered with Vcc_Prog. Otherwise, both LEDs **D1** and **D2** light up on powering the device with other power sources.
5. Program the hex file onto the CY3280-20x66 Universal CapSense Controller board using the MiniProg. While programming is in progress, the **Target Power** LED on the MiniProg is on, as shown in Figure 3-2.

   **Note** Hex files are available in the installed directory and in the kit CD.
6. When **Programming Succeeded** appears in the Actions pane, detach the MiniProg and connect an I2USB bridge to the ISSP connector.
7. Connect your computer to the CY3280-20x66's ISSP connector using the I2USB bridge and a USB cable, as shown in Figure 3-3.

Figure 3-3. Connect I2USB Bridge to Controller



## 3.5 Bridge Control Panel

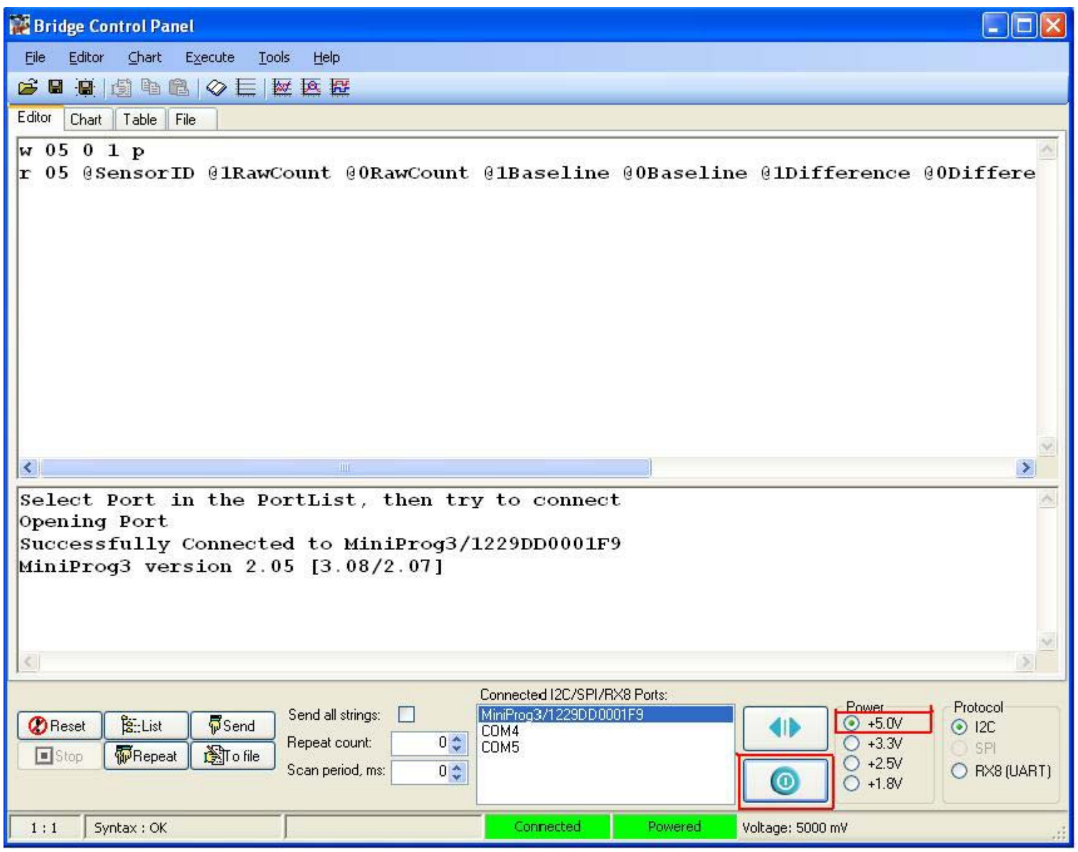
The I2USB bridge is used to read the CapSense parameters from the controller board. These parameters can be viewed using the Bridge Control Panel software. Follow these steps to use the Bridge Control Panel software with the CY3280-20x66 Universal CapSense Controller board.

1. Click **Start > All Programs > Cypress > Bridge Control Panel <version>**.
2. Select the device to be connected from the port selection window.
3. Select **Variable Settings** option from the **Chart** menu. Load the .ini file, by clicking the **Load** button.

   The *CY3280_SLM_Project1.ini* file is the variable setting file, available in the kit CD or at the following location: `<Install_Directory>\CY3280-20X66\<version>\Firmware\I2C-USBBridgeSoftwareConfig`.
4. Load the *CY3280_SLM_Project1.iic* file for iic commands that can be sent to the board.

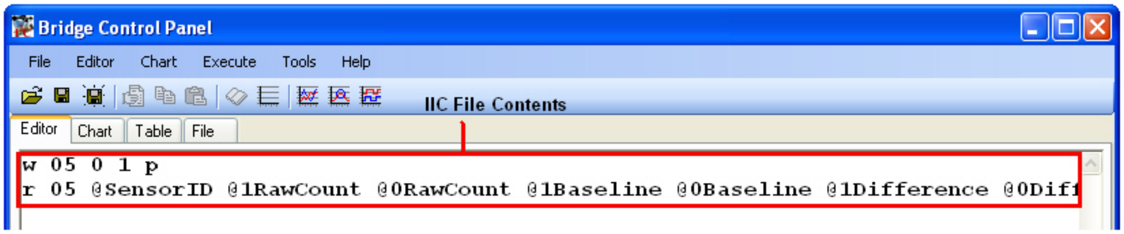
   The .iic file is the configuration setting file, available in the kit CD or at the following location: `<Install_Directory>\CY3280-20X66\<version>\Firmware\I2C-USBBridgeSoftwareConfig`.
5. Go to **File** > **Open File > CY3280_SLM_Project1.iic** to select the file for iic commands.
6. Select **+5.0V** in the **Power** box. See Figure 3-4.
7. Click **Toggle Power** to power the I2USB bridge; the red LED **D1** glows, as shown in Figure 3-3.

Figure 3-4. Toggle Power Button



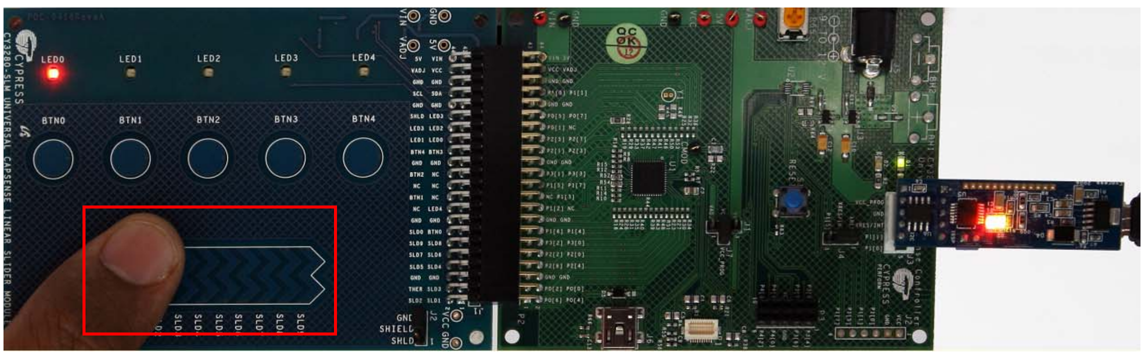

**Note** Close PSoC Designer and PSoC Programmer before opening Bridge Control Panel.

Figure 3-5. Bridge Control Panel Editor View



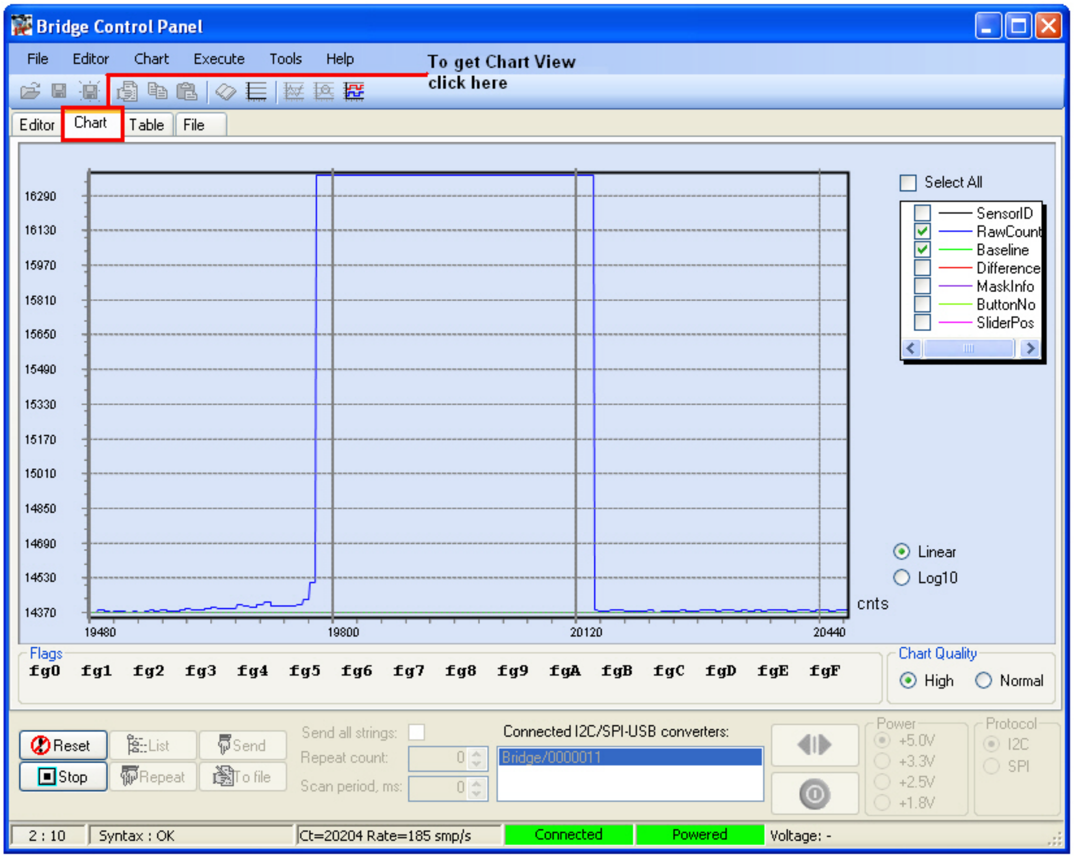

8. Place the cursor on the first command (W 05 0 1 P) and click **Send** to send the sensor ID to slave. In this command, 05 is the slave address. 0 and 1 are the offset address and value of the sensor ID, respectively.
9. Place the cursor on the second command and click **Repeat** to get the parameters continuously from the controller.
10. Touch a button or slider. Each touch lights up the associated LED on the module board, representing where your finger is on the slider.

Figure 3-6. LED2 Glows on Touching Sensor SLD4



11. Click the **Chart** tab to switch to Chart view and see the respective waveforms of CapSense parameters.

Figure 3-7. Bridge Control Panel Chart View



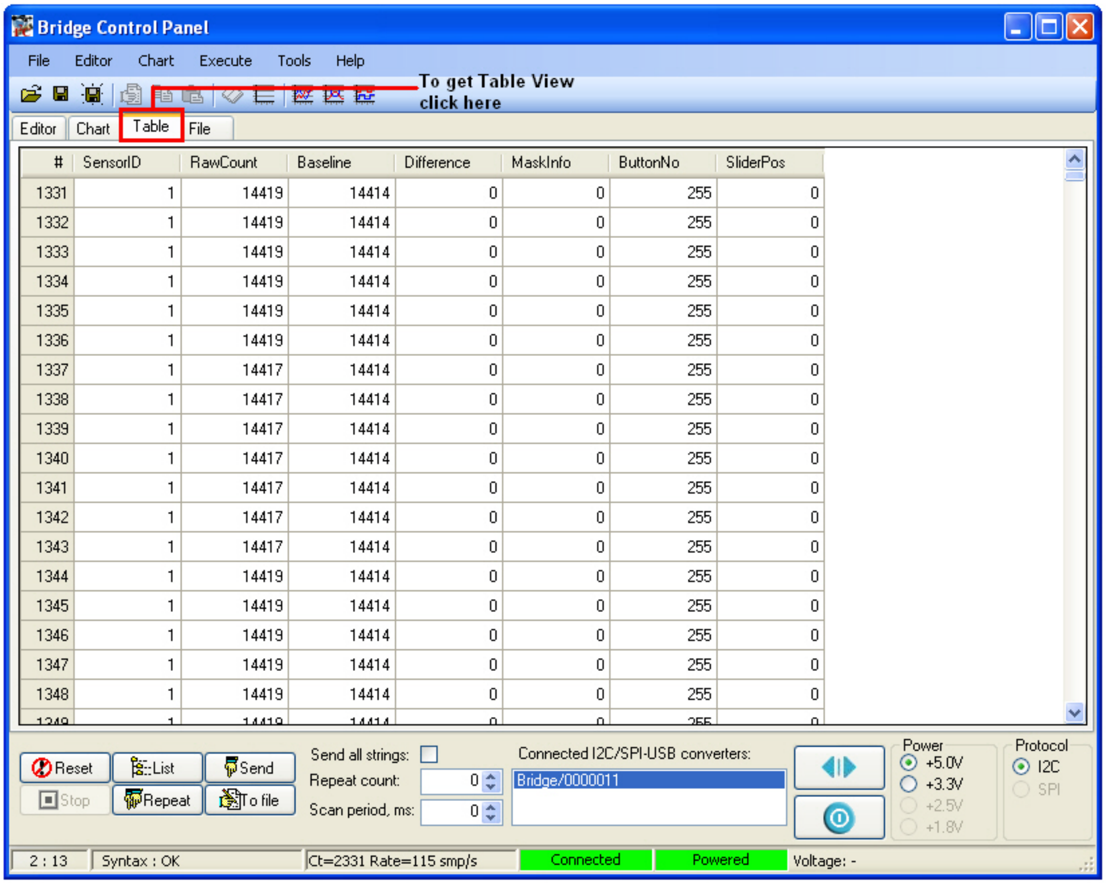

**Note** In the figure, the brown line represents the axis; the blue line indicates RawCount; and the green line indicates the Baseline.

12. View the required parameters by selecting or clearing the checkboxes to the right of the Chart view.

13. Click **Stop** to stop scanning.

14. Click the **Table** tab to view the values of the demonstration board variables, as shown in Figure 3-8.

Figure 3-8. Bridge Control Panel Table View.



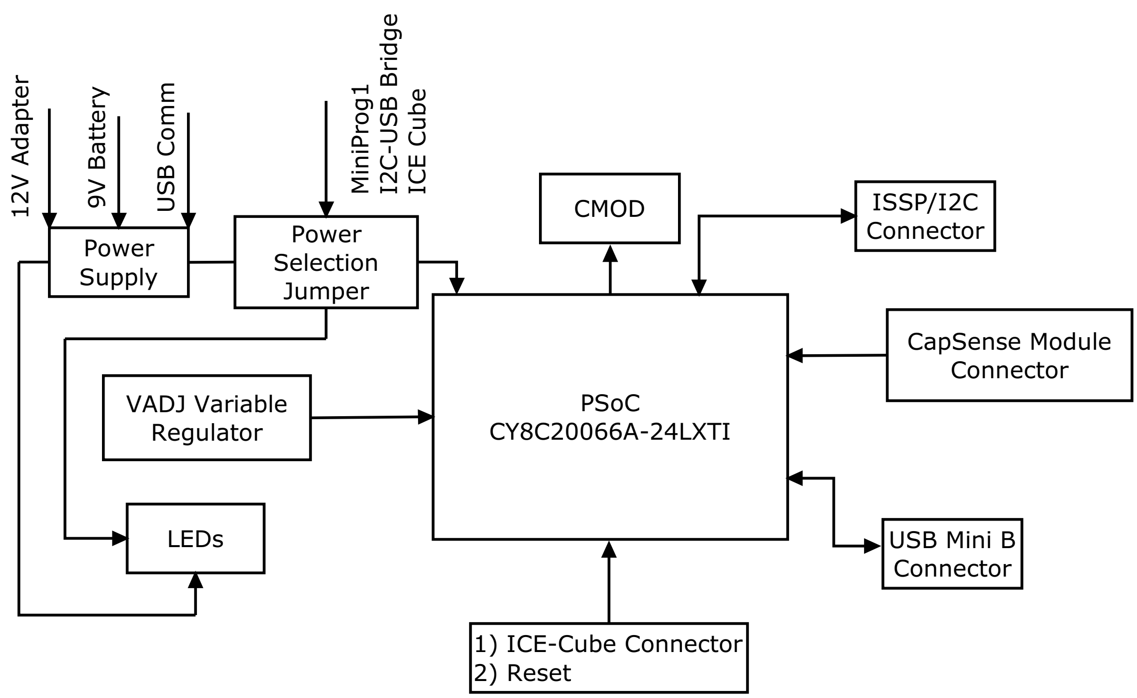
# 4. Hardware

## 4.1 System Block Diagram

The CY3280-20x66 Universal CapSense Controller board has the following sections:

- PSoC CY8C20066A-24 LTXI
- Power supply system
- USB Mini-B connector
- CapSense module connector
- ISSP/I2C connector
- ICE-Cube debug connector
- VADJ variable regulator control
- LEDs
- Reset switch
- CMOD

Figure 4-1. System Block Diagram

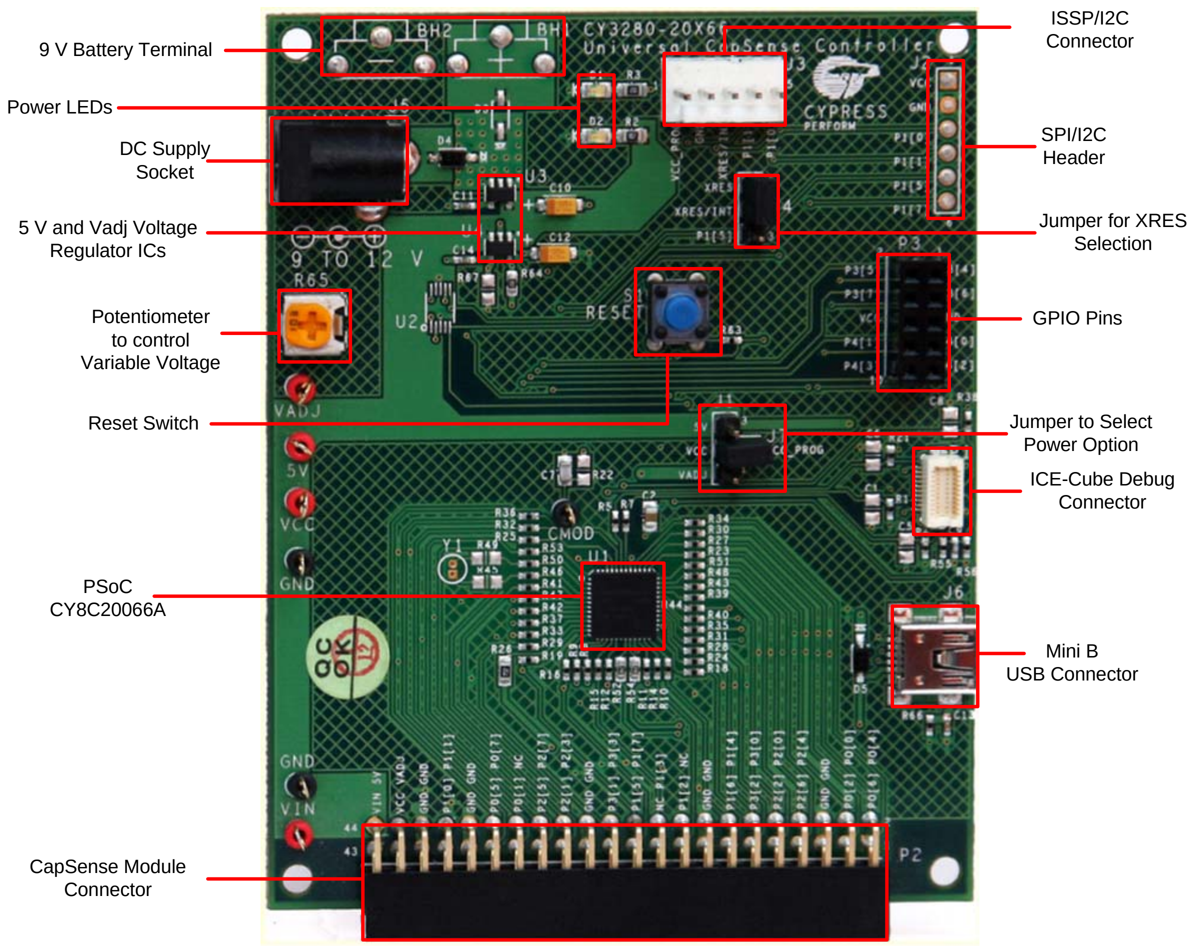


## 4.2 Functional Description

The CY3280-20x66 Universal CapSense Controller includes PSoC CY8C20066A, ISSP connector, CapSense module connector, USB Mini-B connector, DC supply socket, reset button, ICE-Cube debug connector.

Figure 4-2 shows the different functional blocks on the CY3280-20x66 Universal CapSense Controller board.

Figure 4-2. CY3280-20x66 Universal CapSense Controller Functional Blocks



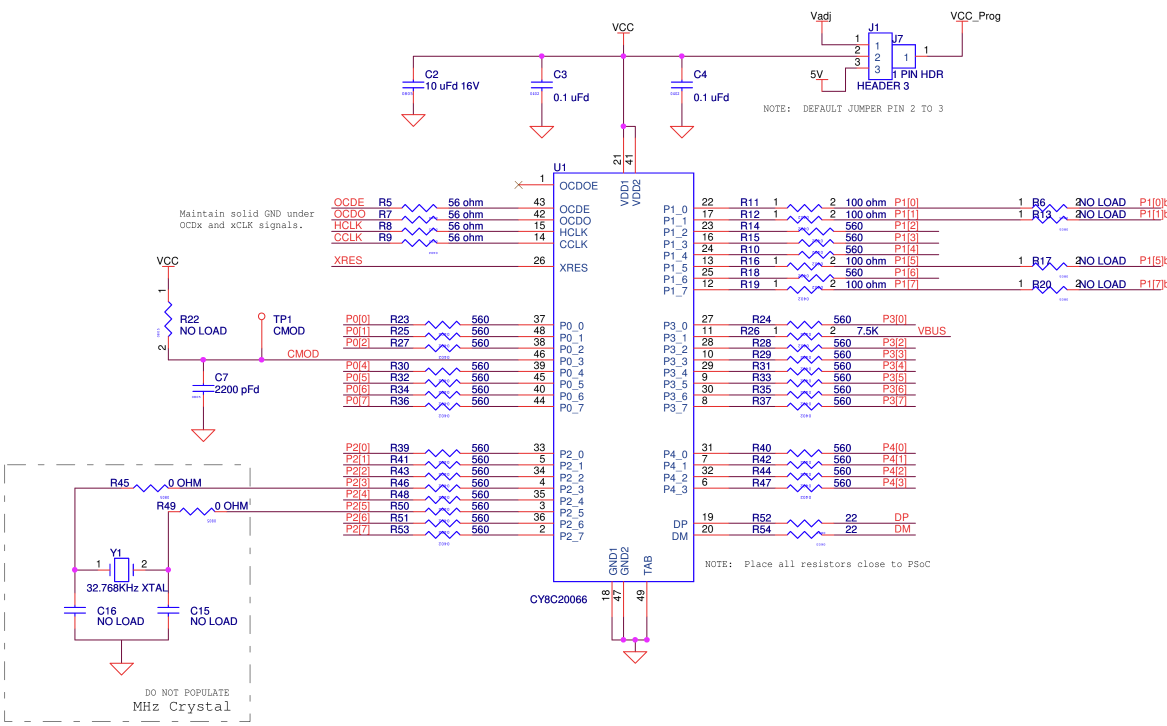
### 4.2.1 PSoC CY8C20066A-24 LTXI

The PSoC CY8C20066A is initially factory programmed as a CapSense Controller with the control circuitry to work with the CY3280-20x66 Universal CapSense Controller Kit. The PSoC CY8C20066A along with CSA/CSD technology demonstrates the use of CapSense buttons and linear sliders; the module board connected to the CY3280-20x66 Universal CapSense Controller has sensors and LEDs. The configuration number varies with the modules.

A CapSense module board is connected to the PSoC through Port P2. The list of pins connected to different ports is shown in Pin Description of CY8C20066A-24LTXI on page 28.

PSoC CY8C20066A is programmed through ISSP using a MiniProg. Data acquisition and output checking is done using I2USB bridge.

Figure 4-3. Schematic View of PSoC CY8C20066A



#### 4.2.1.1 Pin Description of CY8C20066A-24LTXI

| Pin No | Name | Description | Connected To |
|---|---|---|---|
| 1 | OCDOE | OCD mode direction pin | |
| 2 | P2[7] | GPIO, LED1 | P2 |
| 3 | P2[5] | Crystal output (XOut), LED0 | P2 |
| 4 | P2[3] | Crystal input (XIn), CapSense | P2 |
| 5 | P2[1] | CapSense | P2 |
| 6 | P4[3] | GPIO | P3 |
| 7 | P4[1] | GPIO | P3 |
| 8 | P3[7] | GPIO | P3 |
| 9 | P3[5] | GPIO | P3 |
| 10 | P3[3] | CapSense | P2 |
| 11 | P3[1] | GPIO | P2 |
| 12 | P1[7] | I2C SCL, SPI SS | P2, J2 |
| 13 | P1[5] | I2C SDA, SPI MISO | P2, J4, J2 |
| 14 | CCLK | OCD CPU clock output | P1 |
| 15 | HCLK | OCD high-speed clock output | P1 |
| 16 | P1[3] | SPI CLK, CapSense | P2 |
| 17 | P1[1] | ISSP CLK[3], I2C SCL, SPI MOSI | P2, J3, J2 |
| 18 | VSS | Ground connection | |
| 19 | D+ | USB D+ | J6 |
| 20 | D– | USB D– | J6 |
| 21 | VDD | Supply voltage | |
| 22 | P1[0] | ISSP DATA[3], I2C SDA, SPI CLK[4] | P2, J3, J2 |
| 23 | P1[2] | GPIO, LED4 | P2 |
| 24 | P1[4] | Optional external clock input (EXTCLK), CapSense | P2 |
| 25 | P1[6] | CapSense | P2 |
| 26 | XRES | Active high external reset with internal pull down | J4 |
| 27 | P3[0] | CapSense | P2 |
| 28 | P3[2] | CapSense | P2 |
| 29 | P3[4] | GPIO | P3 |
| 30 | P3[6] | GPIO | P3 |
| 31 | P4[0] | GPIO | P3 |
| 32 | P4[2] | GPIO | P3 |
| 33 | P2[0] | CapSense | P2 |
| 34 | P2[2] | CapSense | P2 |
| 35 | P2[4] | CapSense | P2 |
| 36 | P2[6] | CapSense | P2 |
| 37 | P0[0] | GPIO | P2 |
| 38 | P0[2] | CapSense | P2 |
| 39 | P0[4] | CapSense | P2 |
| 40 | P0[6] | CapSense | P2 |
| 41 | VDD | Supply voltage | |
| 42 | OCDO | OCD even data I/O | P1 |
| 43 | OCDE | OCD odd data output | P1 |
| 44 | P0[7] | GPIO | P2 |
| 45 | P0[5] | GPIO, LED3 | P2 |
| 46 | P0[3] | Integrating input | P2 |
| 47 | VSS | Ground connection | |
| 48 | P0[1] | GPIO, LED2 | P2 |

### 4.2.2 Power Supply System

The power supply system on this board is versatile; it takes input supply from the following sources:

- 12-V DC supply using connector J5
- 9-V battery connector using connectors BH1 and BH2
- USB power (5 V) from communications section using connector J6
- Power from MiniProg and I2USB bridge (Vcc_Prog) connected at J3
- Power from ICE-Cube (Vcc_Prog) connected at P1

The board power domain is split into:

- **Vin rail:** This is the rail where the input of on-board regulators are connected. This domain is powered through protection diodes (by 12-V DC supply and 9-V battery terminal).
- **5V rail:** This is the output of the 5-V regulator IC U3. The rail has a fixed 5-V output regardless of jumper settings. The voltage in this rail can be lesser than 5 V only when the board is powered by the USB.
- **Vadj rail:** This is the output of the variable voltage regulator control.
- **VCC_PROG:** This is power from devices such as MiniProg, I2USB bridge, and ICE-Cube debugger.

The following block diagram shows the structure of the power system on the board.

Figure 4-4. Power Supply System Structure

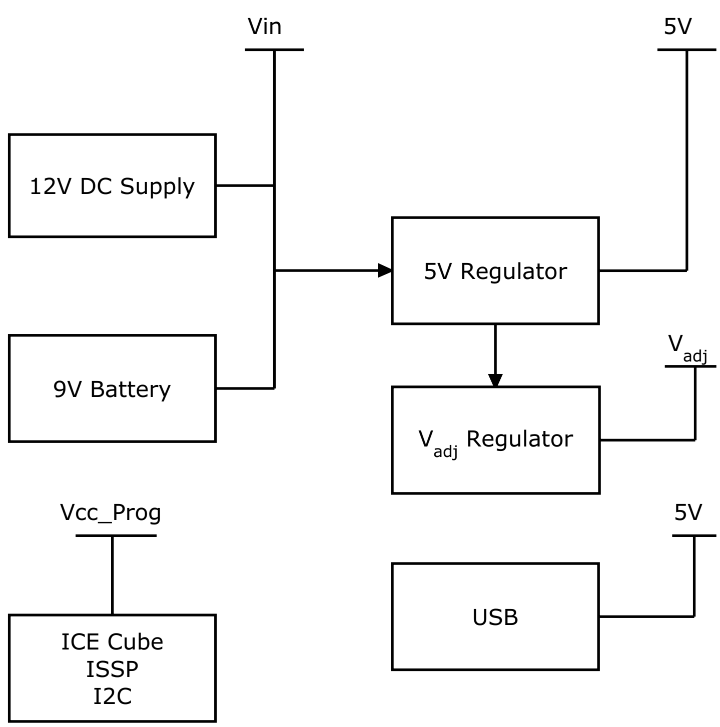
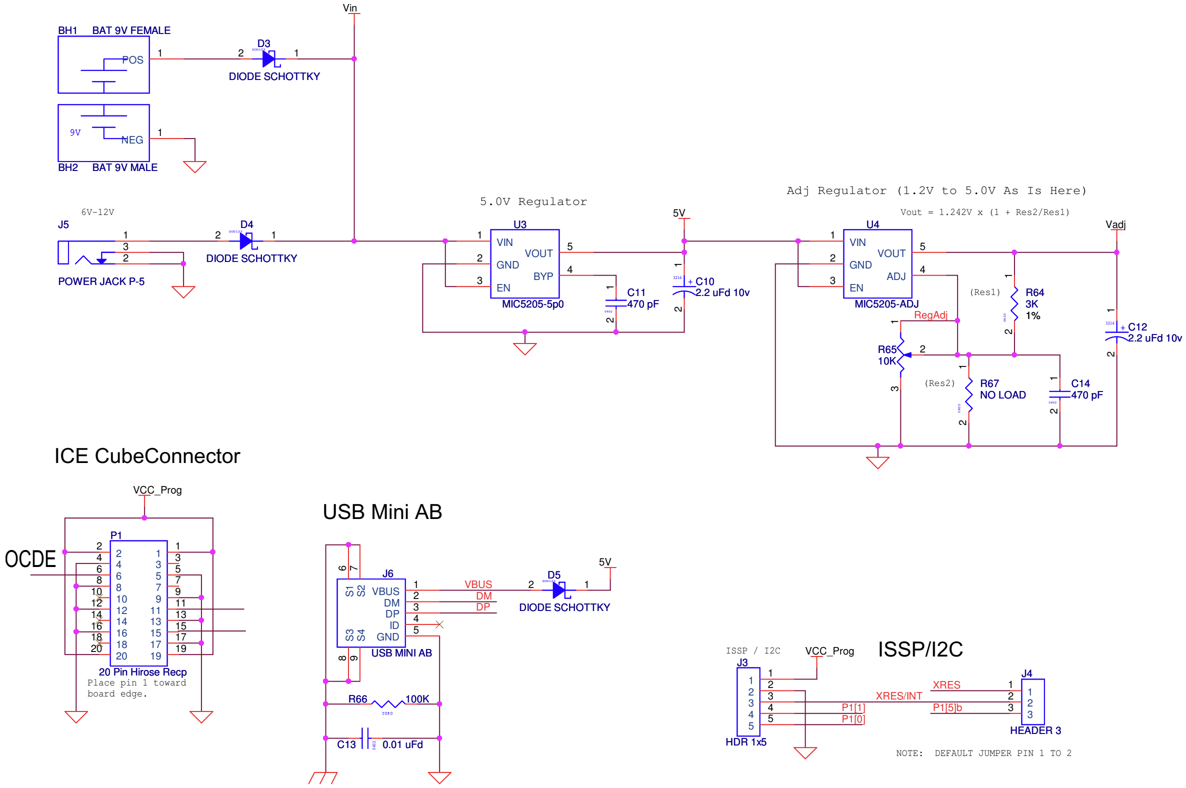


Figure 4-5. Schematic View of Power Supply System Structure




### 4.2.2.1 *Power Supply Jumper Setting*

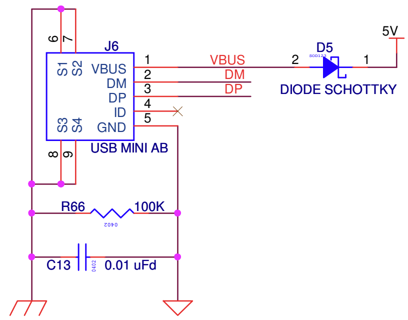
The jumper settings for each power setting are as follows

- To allow power to be supplied by the MiniProg, I2USB bridge, and ICE-Cube, connect J7 to pin 2 of J1 with a jumper.
- To allow power from the 12-V DC supply or 9-V battery terminal or USB Mini-B connector, connect pin 2 and 3 of J1 with a jumper.
- To allow power from the variable regulator control (Vadj), connect pins 1 and 2 of J1 with a jumper.

### 4.2.3 USB Mini-B Connector

The USB Mini-B connector is a mini port to communicate between the PC and the board. It is also used to power up the controller, supplying a voltage of 5 V. Protection diode D5 is present so that the 5 V from the board does not flow to the USB connector.

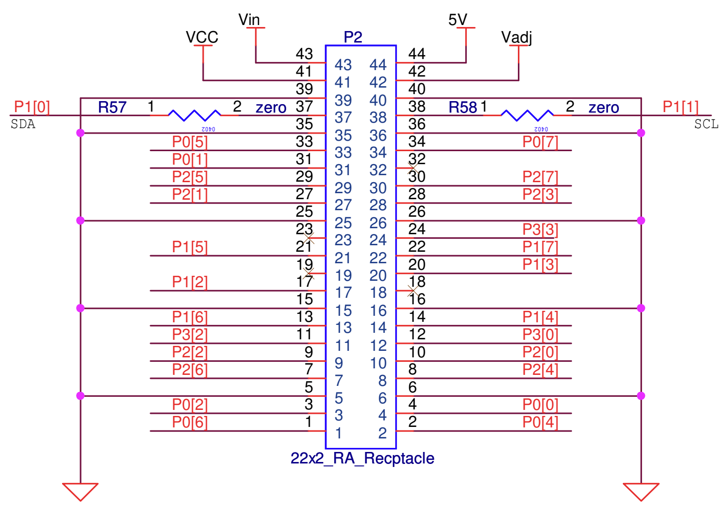
Figure 4-6. Schematic View of USB Mini B Port




### 4.2.4 CapSense Module Connector

The CY3280-20x66 has an expansion port, P2. It is designed to connect CapSense module boards. The CY3280-20x66 controller board can be used with any of the Universal CapSense module boards, such as the CY3280-SLM. Universal CapSense module boards can be interfaced to the CY3280-20x66 controller via the 44-pin connector P2. The pin mapping for the port P2 is shown in the following figure.

Figure 4-7. Schematic View of Port P2

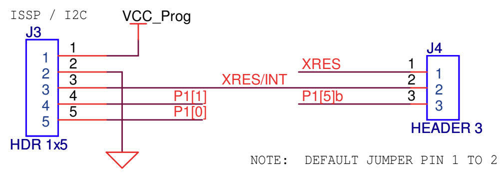


### 4.2.5 ISSP/I2C Connector

In-system serial programming (ISSP) is used to program the PSoC device using MiniProg1 and the USB cable. Plug in the MiniProg device to the ISSP header J3.

The ISSP connector is also used to connect the I2USB bridge to communicate between the PC and the controller board. See Power Supply Jumper Setting on page 30. The pin mapping for the ISSP connector is shown in the following figure.

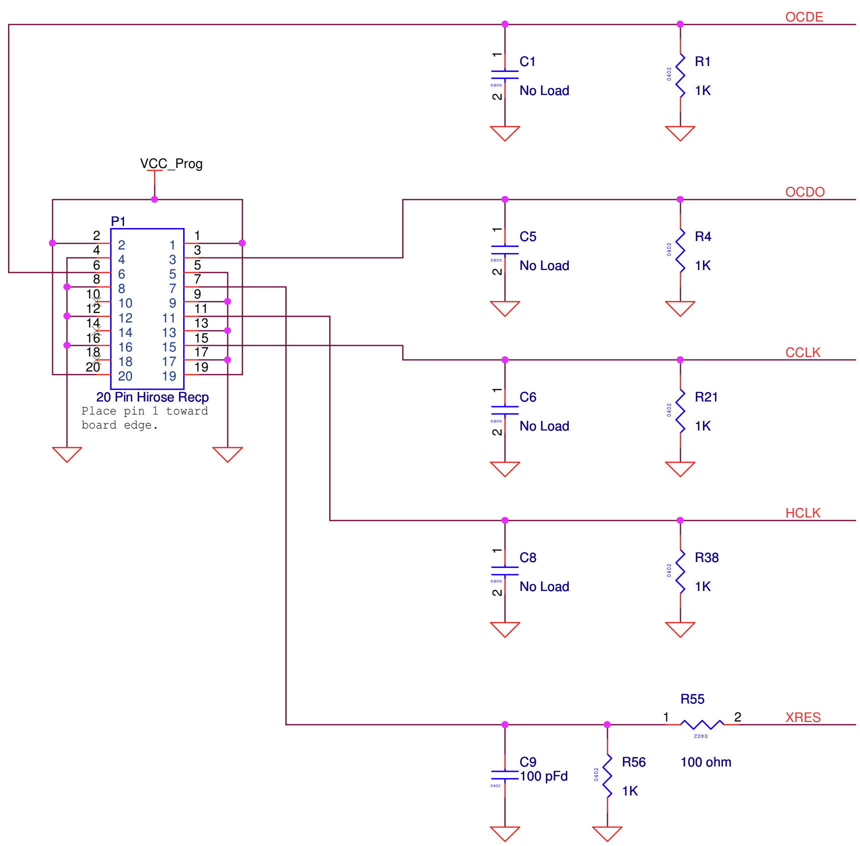
Figure 4-8. Schematic View of ISSP/I2C Connector.




### 4.2.6 ICE-Cube Debug Connector

The CY3215-DK ICE-Cube in-circuit debugger allows you to debug and view the content of specific memory locations. The ICE-Cube debugger can be connected to the CY3280-20x66 controller through port P1. See Power Supply Jumper Setting on page 30. The following figure shows the schematic view of the ICE-Cube debug connector.

Figure 4-9. Schematic View of ICE-Cube Connector.



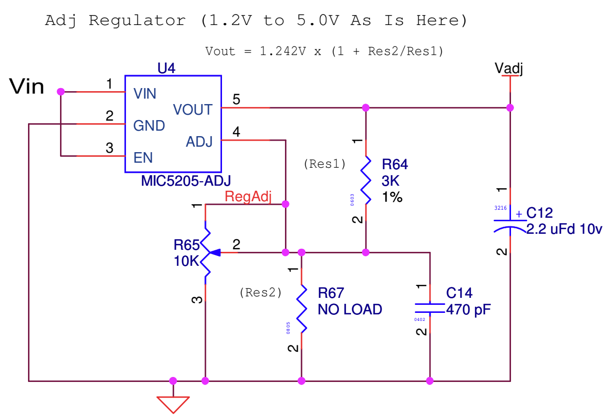
### 4.2.7 VADJ Variable Regulator Control

The CY3280-20x66 controller has a variable regulator control used to vary input voltage using the voltage regulator IC U4. The input for the variable regulator control is the output of the 5-V regulator. Vadj is used to demonstrate CapSense at several voltages.

The minimum Vout from IC U4 is 1.242 V when the resistance at R67 is 0. This load at R67 can be varied using the potentiometer. If the load at R67 is 3 K, the output voltage is 2.484 V; if the load at R67 is 9 K, the output voltage is 4.968 V.

See Power Supply Jumper Setting on page 30 for the settings to use input source Vadj.

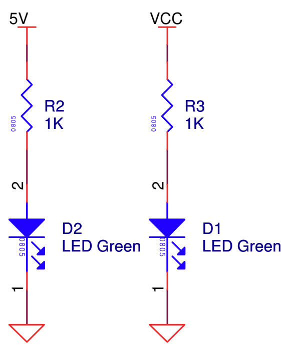
Figure 4-10. Schematic View of Vadj Regulator.




### 4.2.8 LEDs

The LEDs are used to show the status of the controller board. LED D2 lights up on connecting power supply from 12-V DC supply or USB or 9-V battery. LED D1 lights up when the board is powered by any of the following power sources: 12-V DC supply, USB, 9-V battery, MiniProg, I2USB bridge, or ICE-Cube.

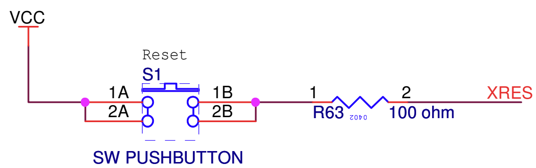
Figure 4-11. Schematic View of LEDs D1 and D2.



### 4.2.9 Reset Switch

The Reset switch resets PSoC to start of the program (code examples). On reset, after XRES de-asserts, the SDA and SCL lines drive resistive low for eight sleep clock cycles and transition to high impedance state.

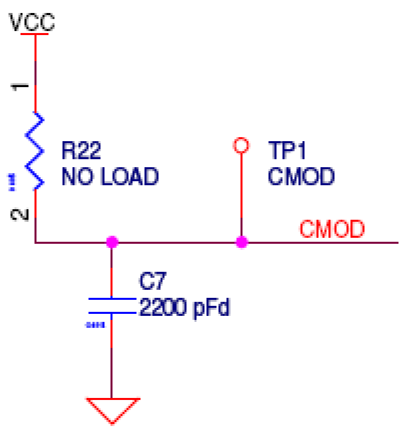
Figure 4-12. Schematic View of RESET Switch




### 4.2.10 CMOD

CMOD is the test point provided on the Universal CapSense Controller for accessibility of charge and discharge waveforms of the measured capacitance. This test point may increase the noise sensitivity by acting as an antenna.

Figure 4-13. Schematic View of CMOD



# 5. Code Examples

All code examples are available in the Firmware folder of the kit CD or at the following location: <Install_Directory>\CY3280-20x66\<version>\Firmware\

## 5.1 My First Code Example (CY3280_20x66_CSD_PD_Project1)

### 5.1.1 Project Description

This project demonstrates the use of CapSense buttons and linear slider on the CY3280-SLM board using CSD technology and CY8C20x66A. The EzI2Cs User Module is used to transfer CapSense parameters related to a sensor from the board to the PC for monitoring.

This project scans five CapSense buttons and a 10-segment slider using the CSD User Module. There are five LEDs on the board, which illuminate when a CapSense button or slider is touched. The EzI2Cs User Module is used to provide a register-based I2C slave communications protocol. The status of CapSense sensors (both button and slider) and their parameters are updated in the I2C register, which can be accessed by any I2C master, similar to the I2USB bridge.

The application starts by executing boot.asm. The boot.asm initializes the hardware and invokes the 'main' function. The main function initializes the EzI2Cs Slave and CapSense user modules. After initialization, the main function enters into a loop, which does the following:

- Scans all sensors
- Reads the sensor ID sent by the I2C master
- Stores CapSense data in the I2C registers
- Updates the LED status for the On/Off sensors

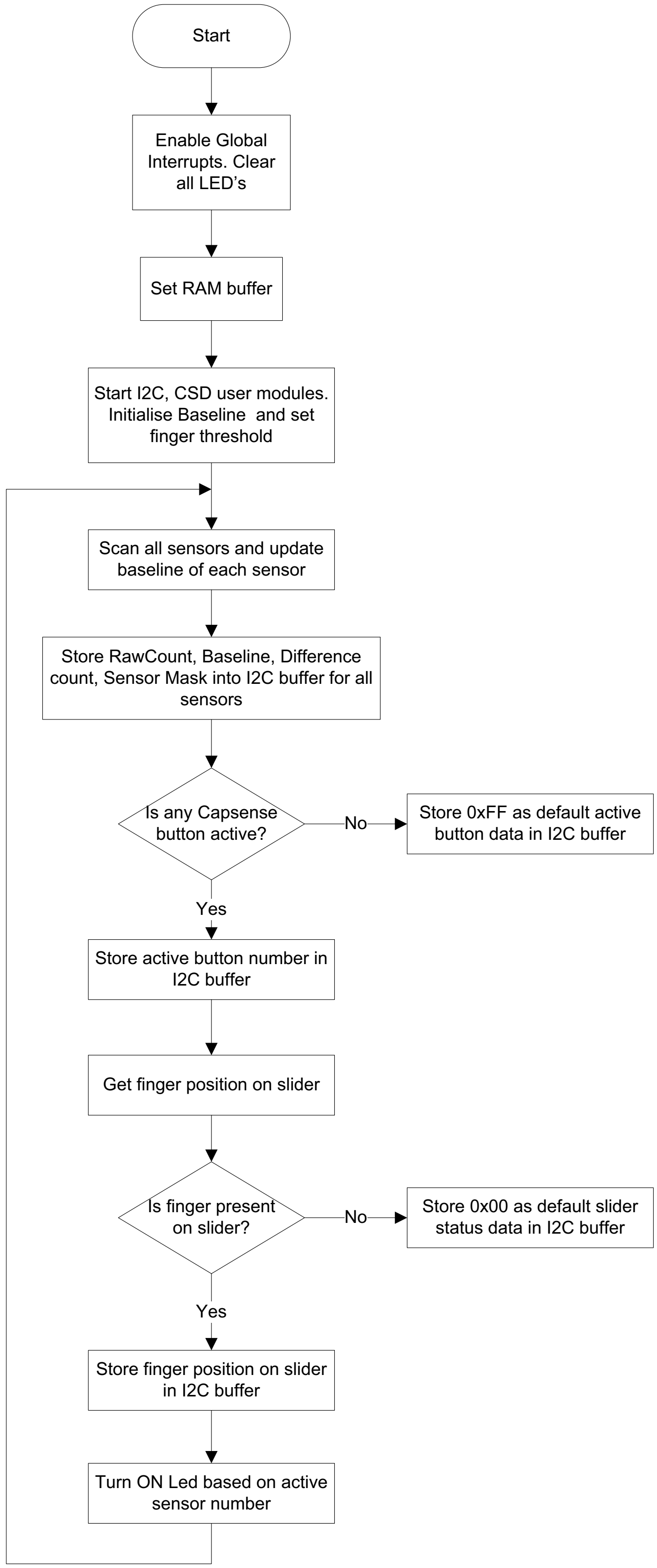
The following user modules are used in this project:

**CSD:** The CSD provides capacitance sensing using the switched capacitor technique with a sigma-delta modulator to convert the sensed switching capacitor current to digital code.

**EzI2Cs:** The EzI2Cs User Module implements an I2C register based slave device. This user module does not require any digital or analog PSoC blocks. It is used to transfer all CapSense parameters related to a sensor to the PC for monitoring

**Note:** To open the project from PSoC Designer, move the Firmware folder to a writable directory and then open it. The firmware folder is located at <Install_Directory>\CY3280-BK1\<version>\Firmware.

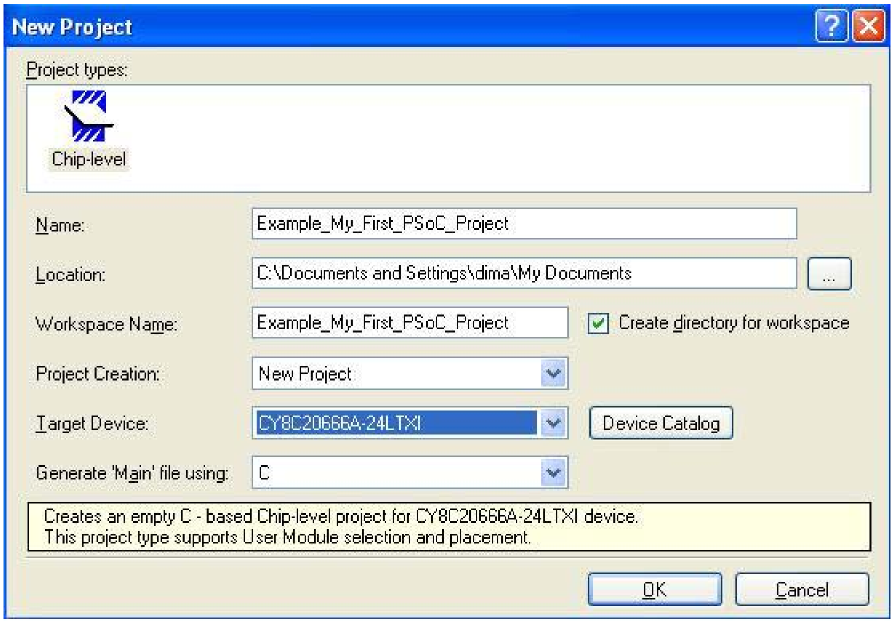
### 5.1.2 Flowchart



### 5.1.3 Creating My First PSoC 1 Project

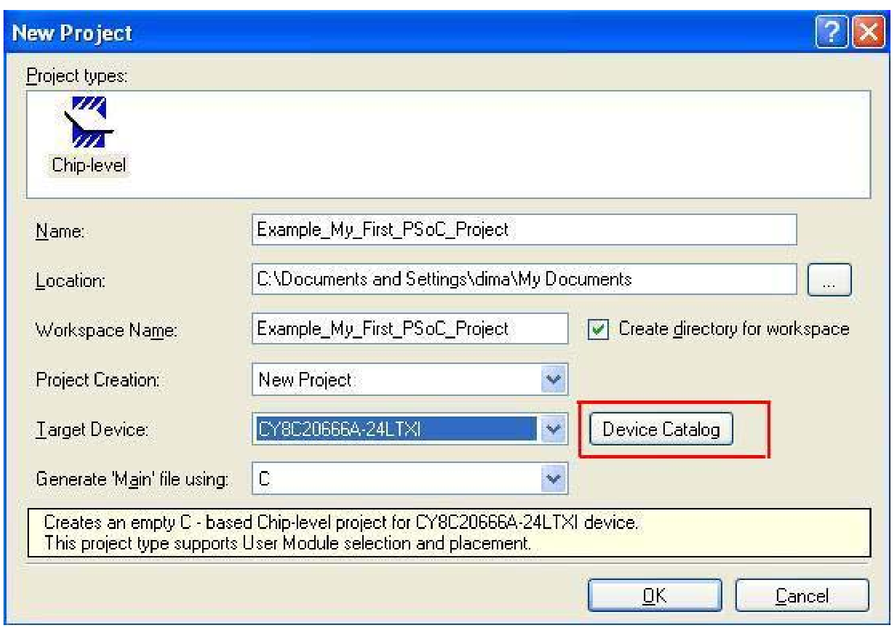
1. Open PSoC Designer.
2. To create a new project, click **File> New Project**.
3. In the New Project window, select the **Chip-level** icon. Name the project **Example_My_First_PSoC_Project**, as shown in Figure 5-1.
4. Click **Browse** and navigate to the directory in which the project is created.

Figure 5-1. New Project



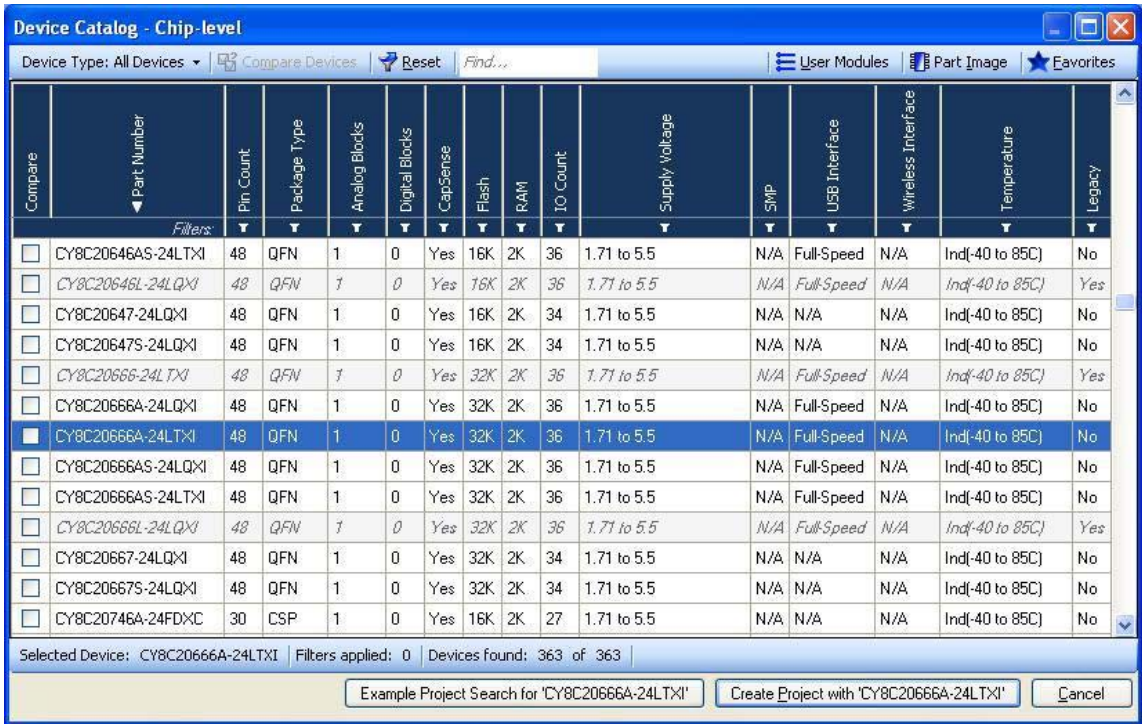

5. Click the **Device Catalog** button, as shown in Figure 5-2, to select the target device.

Figure 5-2. Select Target Device



6. The Device Catalog window opens. Select the **CY8C20666A-24LTXI** device and click **Create Project with 'CY8C20666A-24LTXI'** as shown in Figure 5-3.

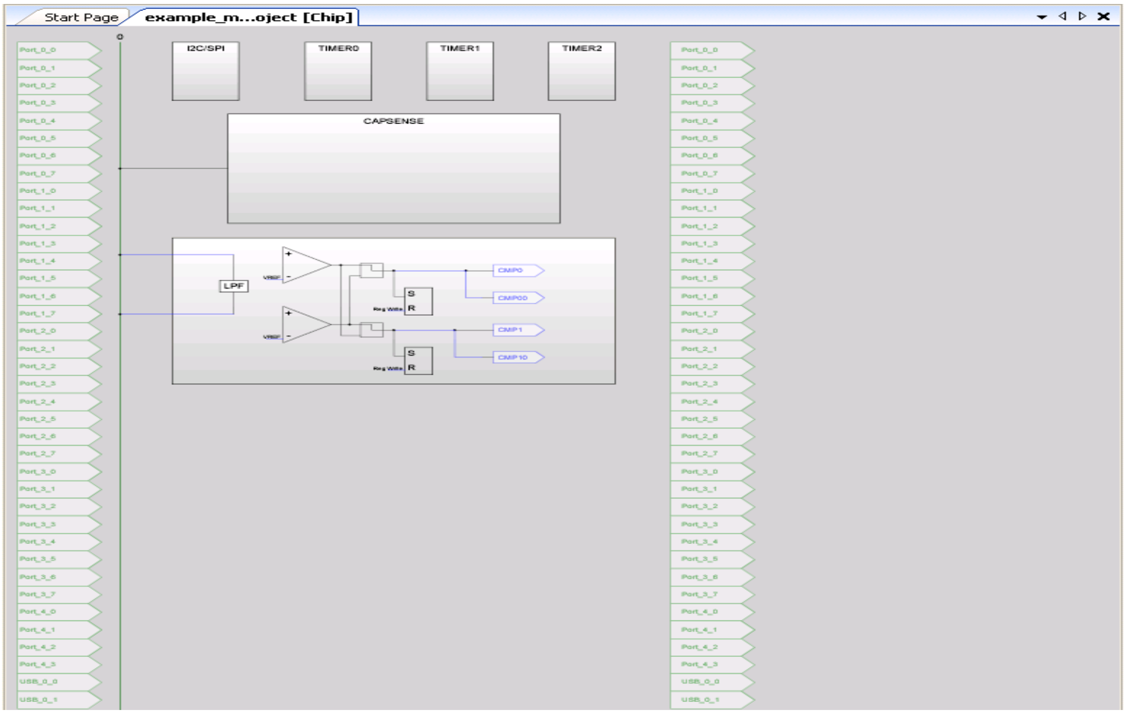
Figure 5-3. Device Catalog




7. Under Generate 'Main' File Using: select **C** and then click **OK**.

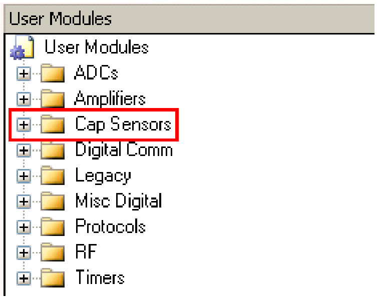
8. By default, the project opens in Chip view, as shown in Figure 5-4.

Figure 5-4. Default View




9. The next step is to place and configure the modules required for this design, as well as connect the modules together and to the pins of the PSoC. In the User Modules window, expand the **Cap Sensors** folder.

Figure 5-5. Cap Sensors Folder



10. In this folder, right-click on **CSD** and select **Place**.

Figure 5-6. User Modules Window - CSD Select

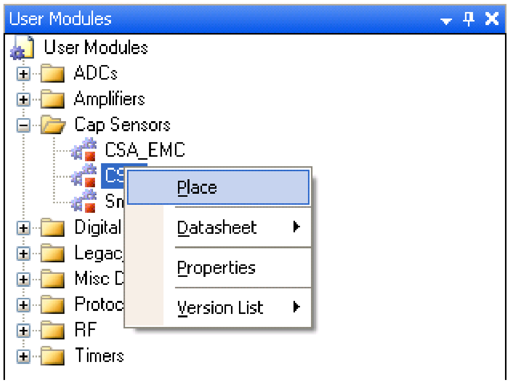
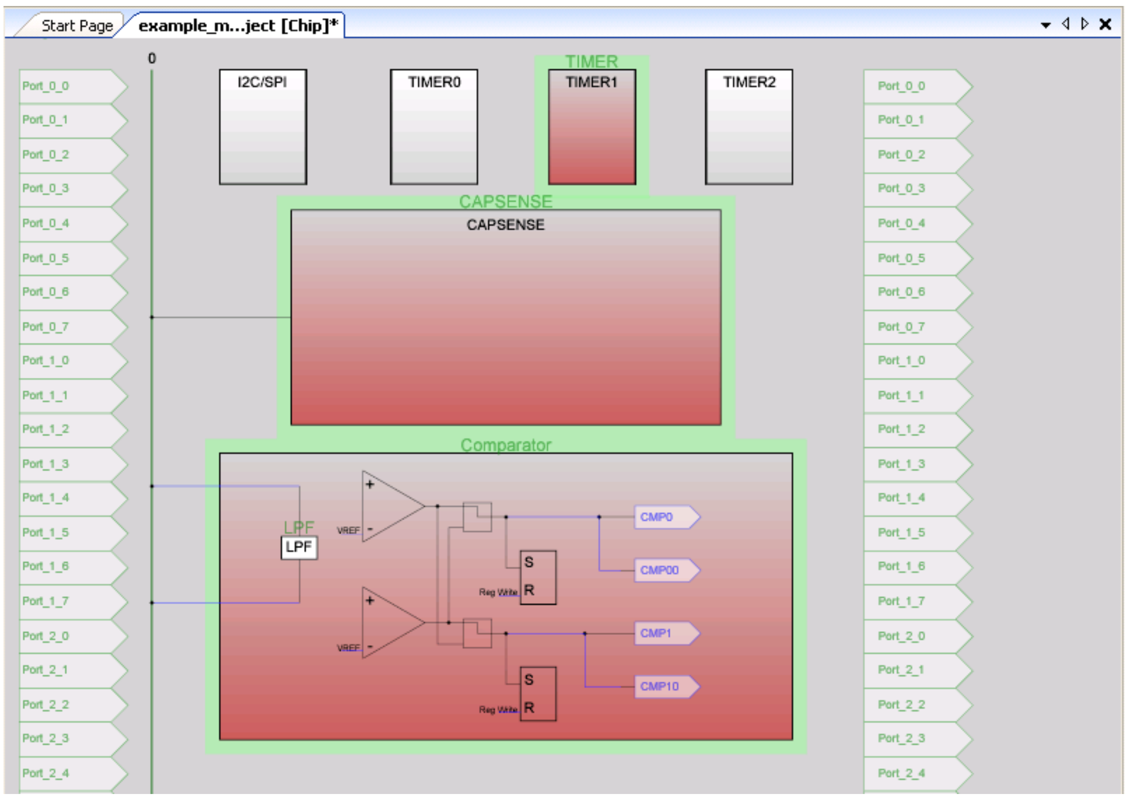
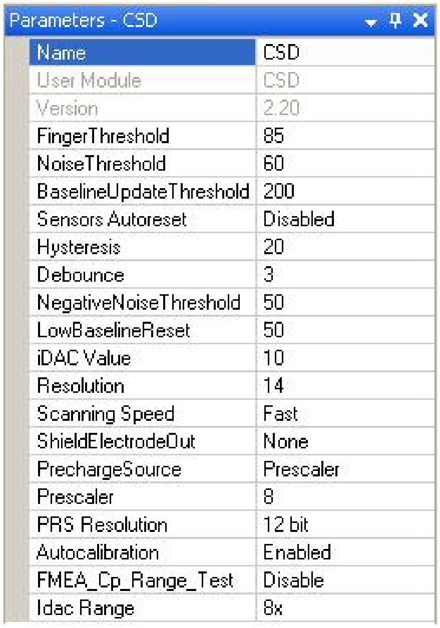



Figure 5-7. CSD User Module Placement



11. Configure the CSD_1 properties, as shown in Figure 5-8:

Figure 5-8. CSD User Module Properties



Parameters - CSD

| | |
|---|---|
| Name | CSD |
| User Module | CSD |
| Version | 2.20 |
| FingerThreshold | 85 |
| NoiseThreshold | 60 |
| BaselineUpdateThreshold | 200 |
| Sensors Autoreset | Disabled |
| Hysteresis | 20 |
| Debounce | 3 |
| NegativeNoiseThreshold | 50 |
| LowBaselineReset | 50 |
| iDAC Value | 10 |
| Resolution | 14 |
| Scanning Speed | Fast |
| ShieldElectrodeOut | None |
| PrechargeSource | Prescaler |
| Prescaler | 8 |
| PRS Resolution | 12 bit |
| Autocalibration | Enabled |
| FMEA_Cp_Range_Test | Disable |
| Idac Range | 8x |

12. Right-click on the **CSD** user module and select the **CSD Wizard** option to assign pins to the sensors.

Figure 5-9. Open CSD Wizard

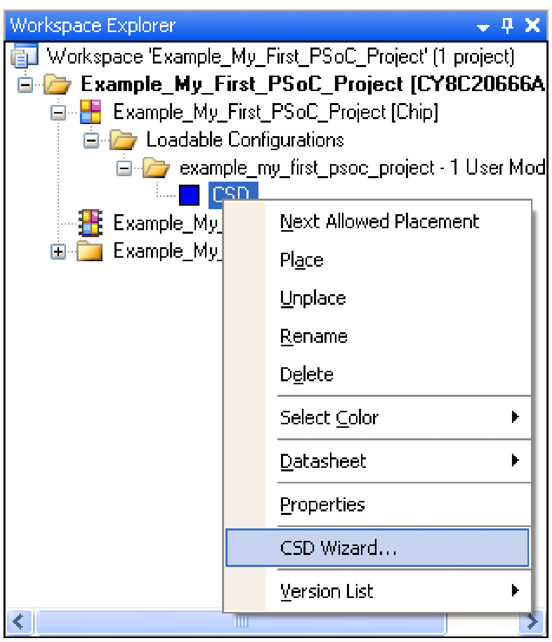
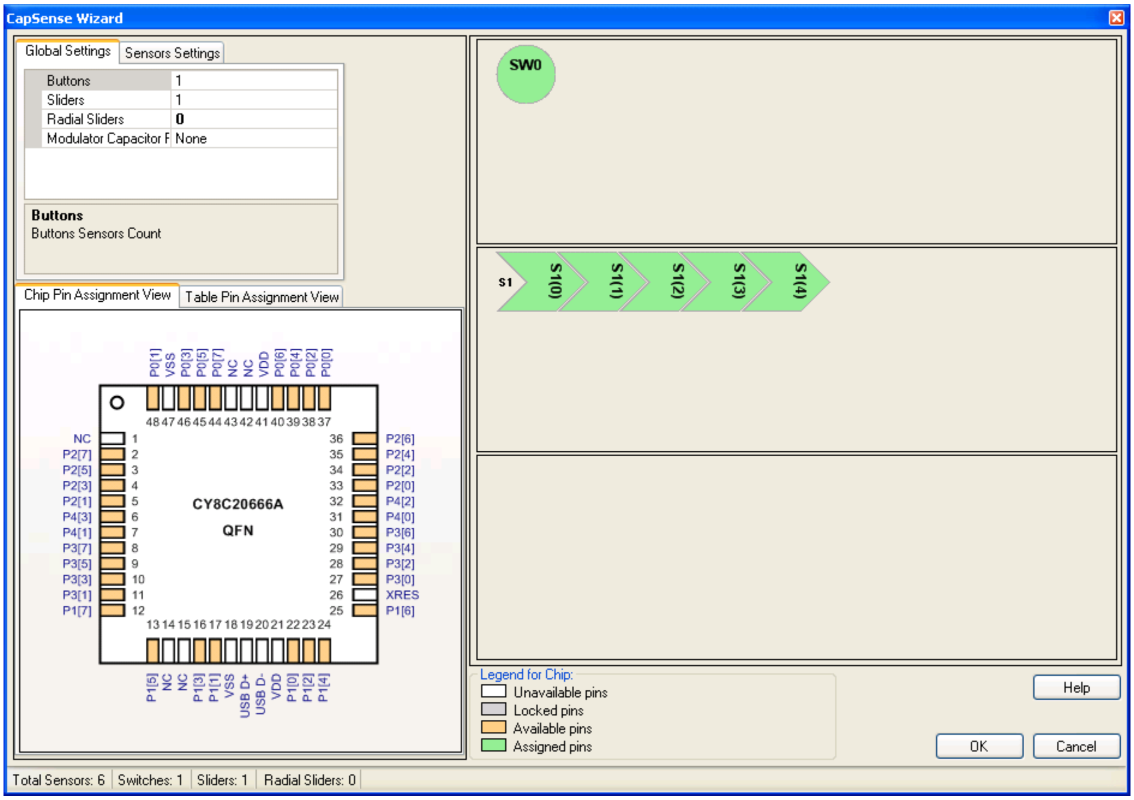
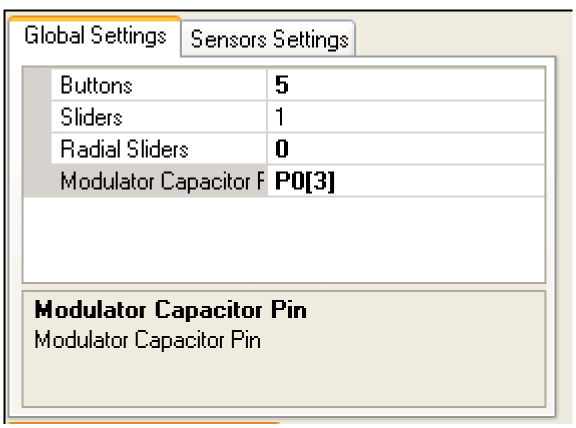


Figure 5-10. CSD Wizard



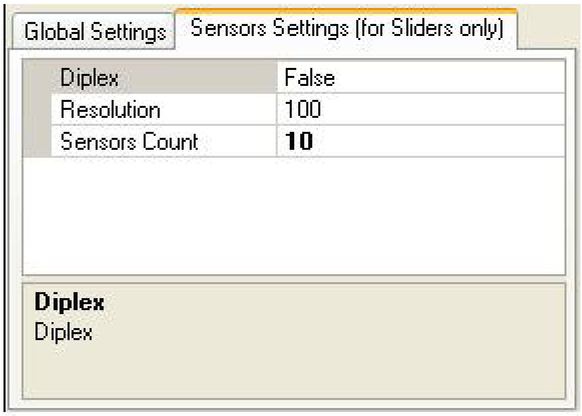

13.Configure the **Global Settings** in the CSD Wizard window.

Figure 5-11. Global Settings - CSD Wizard



14. Click on the slider in the CSD Wizard to view sensor settings. Configure the **Sensor Settings**, as shown in Figure 5-12.

Figure 5-12. Sensors Settings - CSD Wizard




15. To assign the sensor on a particular pin, click and drag from the sensor block to the required pin in the Pin Assignment window. Drag and drop **SW0** to pin **P1 [6]**. Sensor pin assignment can be done in Table Pin Assignment View (Figure 5-14) or Chip Pin Assignment View (Figure 5-13).

Figure 5-13. Assign Sensors to Pins - Chip Pin Assignment View

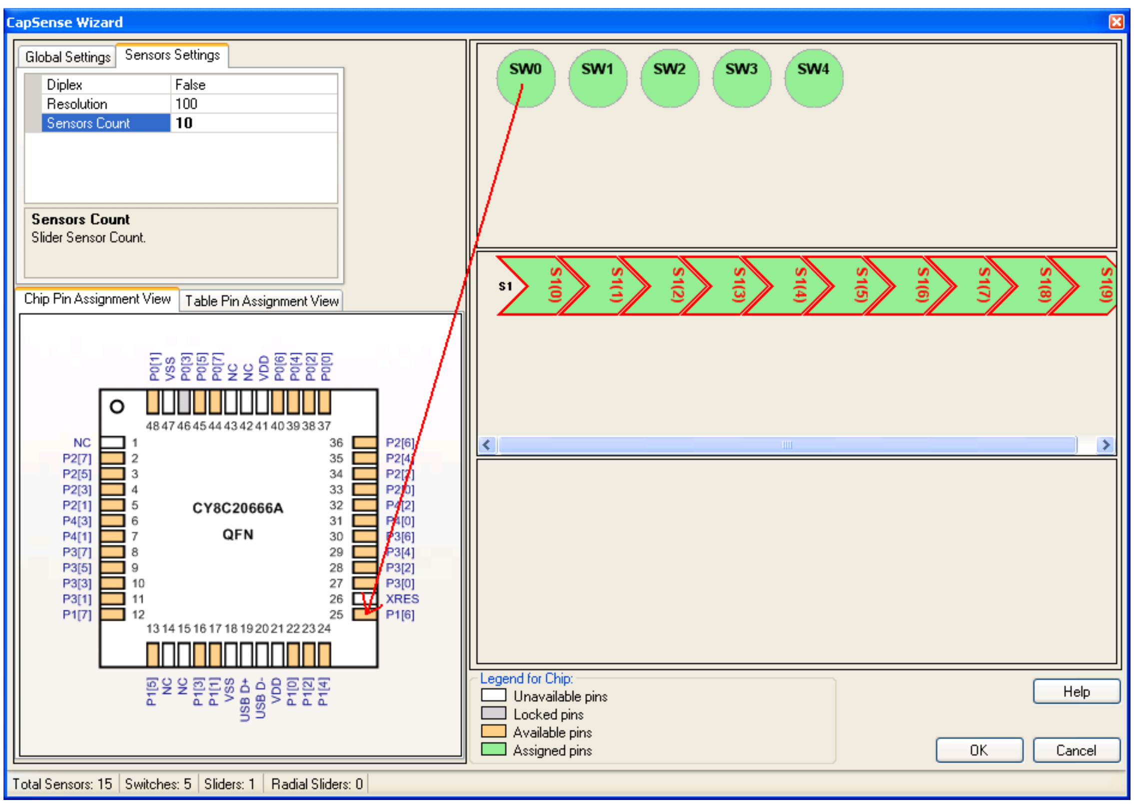
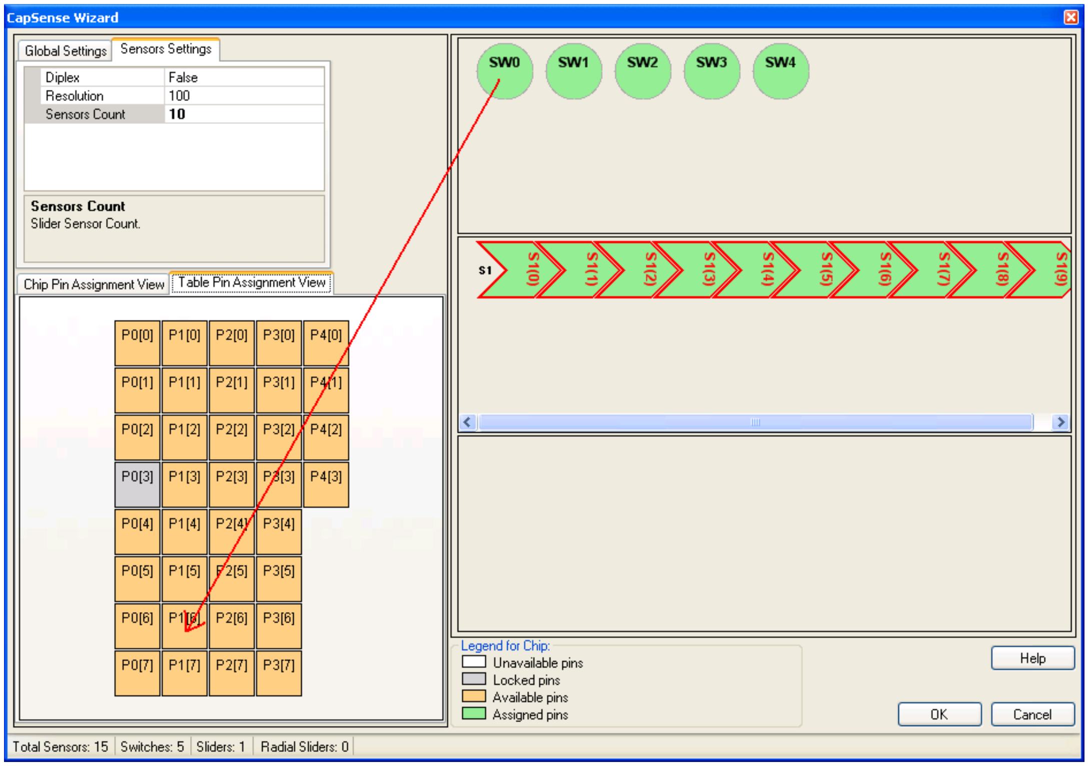
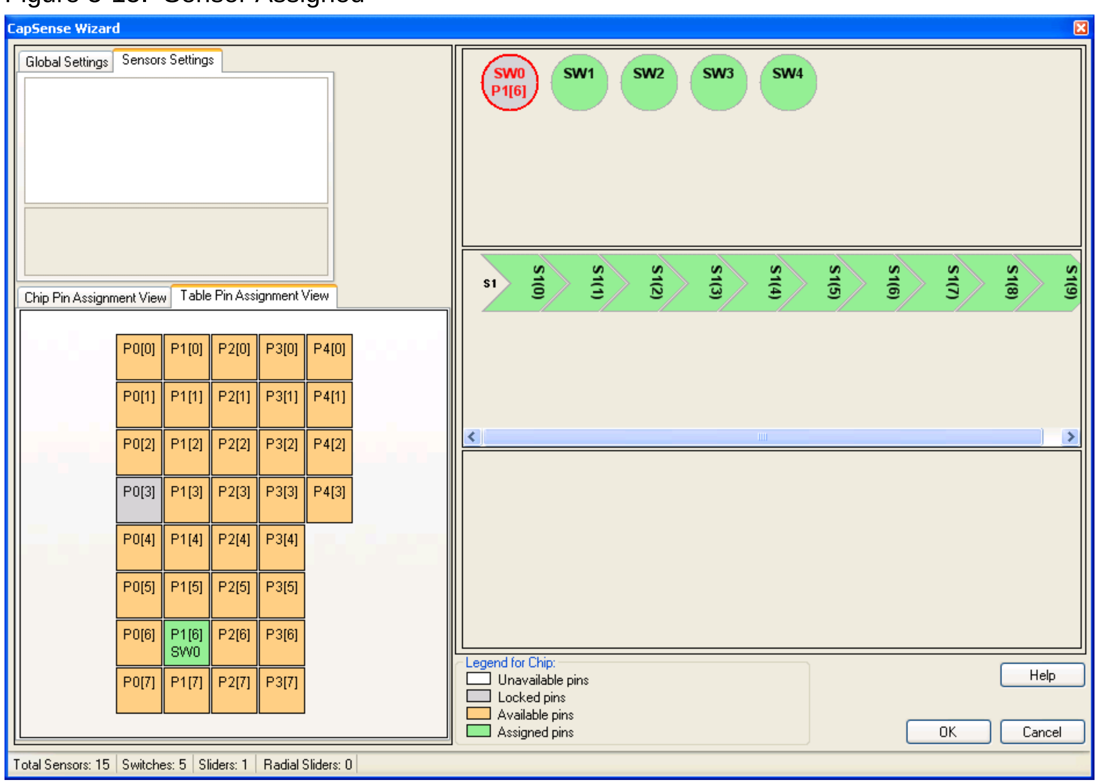


Figure 5-14. Assign Sensors to Pins - Table Pin Assignment View




16.After assigning a sensor, the CSD Wizard looks as shown in Figure 5-15.

Figure 5-15. Sensor Assigned



17. Similarly, assign all the sensors according to the following table

Table 5-1. Sensor Assignment

| Sensor | Port Pin |
|---|---|
| SW0 | P1[6] |
| SW1 | P1[3] |
| SW2 | P3[3] |
| SW3 | P2[1] |
| SW4 | P2[3] |
| S1(0) | P1[4] |
| S1(1) | P0[6] |
| S1(2) | P0[4] |
| S1(3) | P0[2] |
| S1(4) | P2[6] |
| S1(5) | P2[4] |
| S1(6) | P2[2] |
| S1(7) | P2[0] |
| S1(8) | P3[2] |
| S1(9) | P3[0] |

18. After assigning all the sensors successfully, the CSD Wizard appears as follows:

Figure 5-16. All Sensors Assigned - Table Pin Assignment View

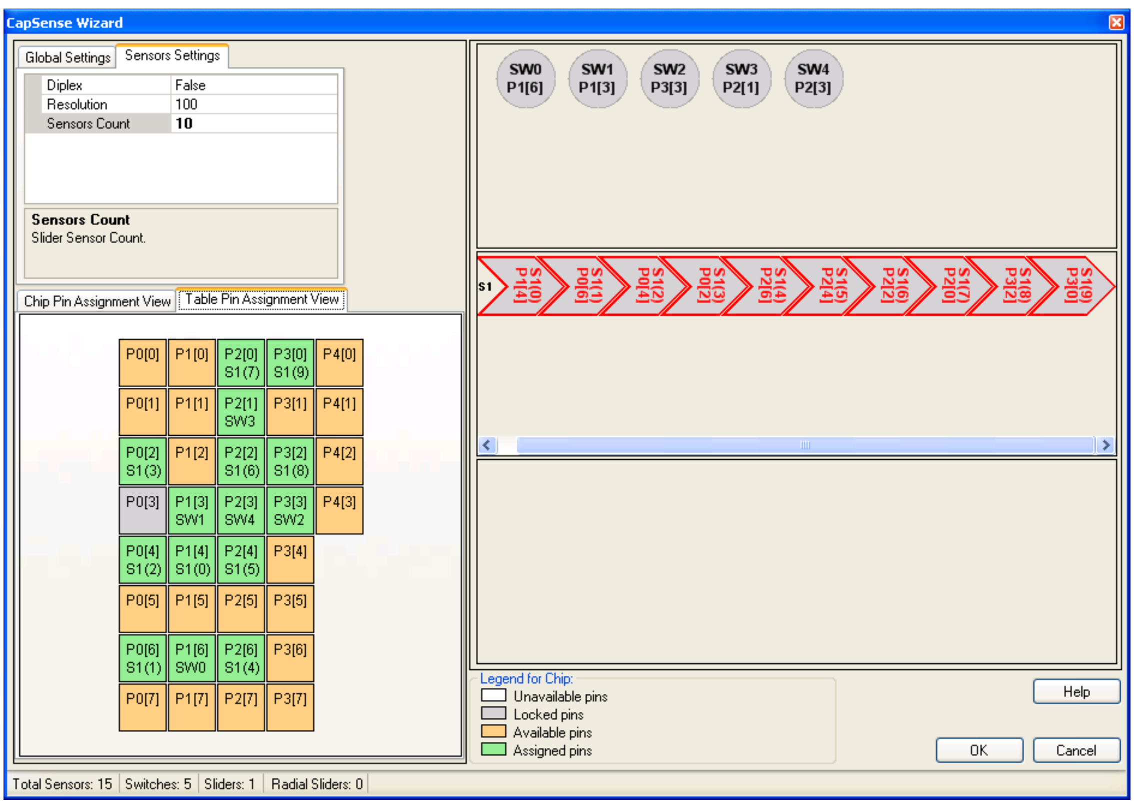
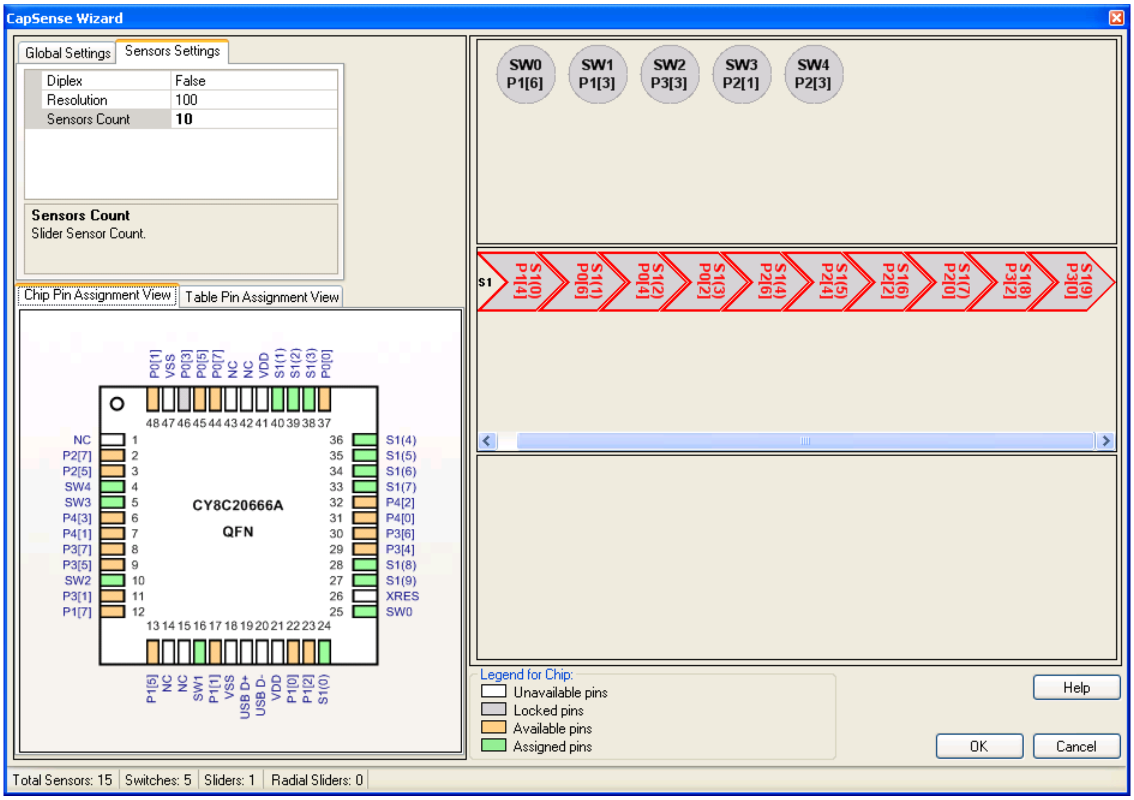
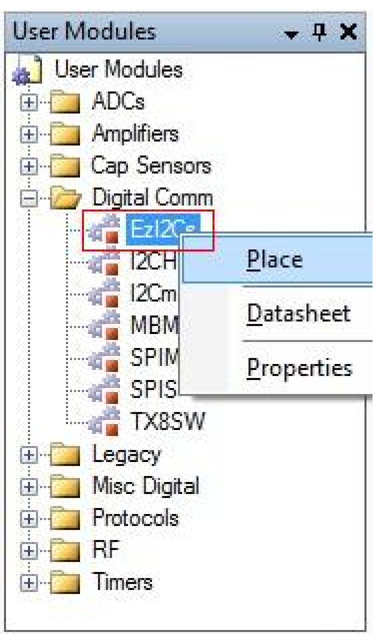


Figure 5-17. All Sensors Assigned - Chip Pin Assignment View



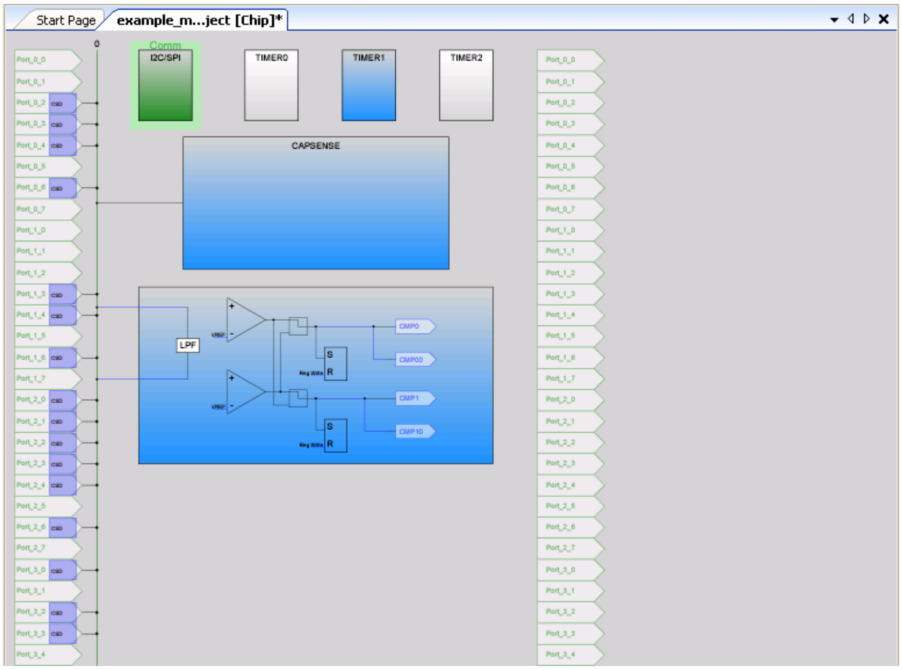

19. In the User Modules window, expand the **Digital Comm** folder, select **EzI2Cs**, right-click and select **Place** to place an EzI2Cs in the design.

Figure 5-18. User Module Window - EzI2Cs Select



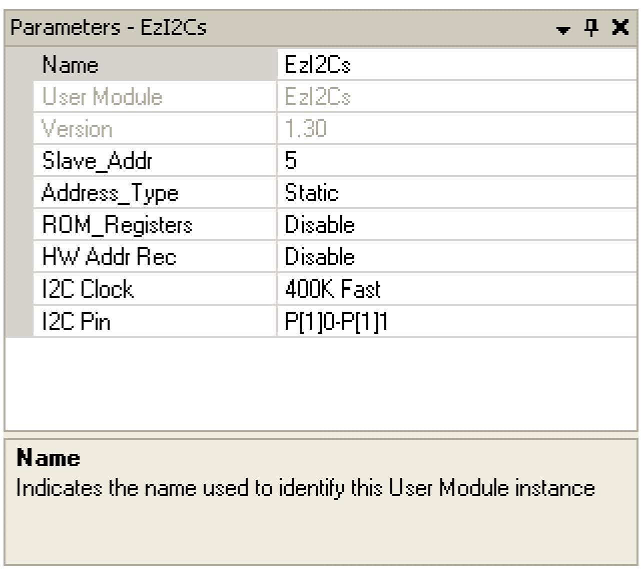
20. By default, the EzI2Cs is placed at the I2C/SPI block. The placement of EzI2Cs module is shown in Figure 5-19.

Figure 5-19. EzI2Cs User Module Placement



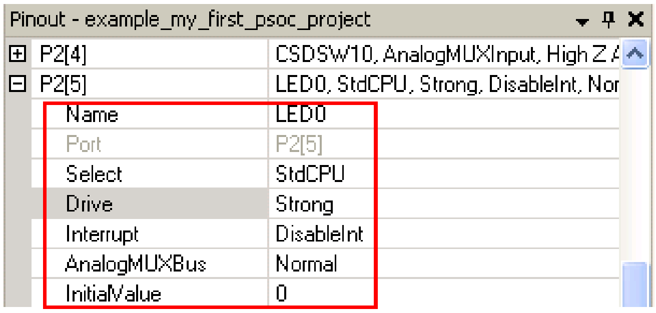

21. Configure the properties of EzI2Cs_1 as shown in Figure 5-20.

Figure 5-20. EzI2Cs User Module Properties



22. Configure the properties of port pin P2[5] in the Pinout window as shown in Figure 5-21.

Figure 5-21. P2[5] Port Pin Properties




23. Configure the properties of P2[7], P0[1], P0[5] and P1[2] as shown in the following figures.

Figure 5-22. P2[7] Port Pin Properties

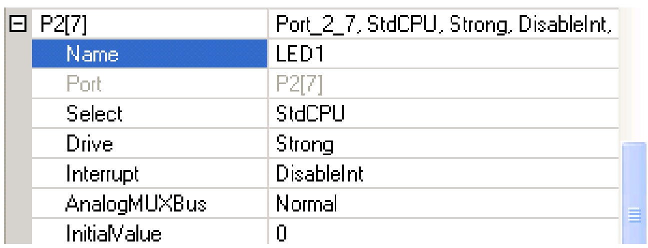



Figure 5-23. P0[1] Port Pin Properties

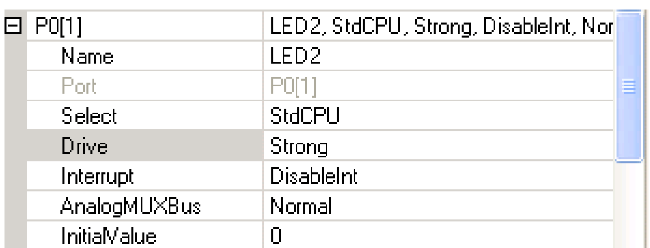



Figure 5-24. P0[5] Port Pin Properties

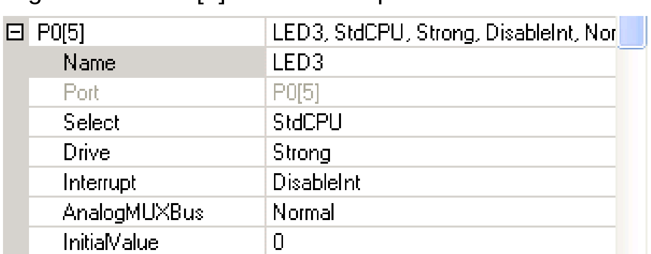
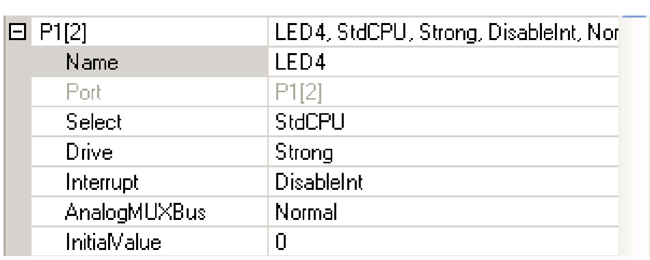
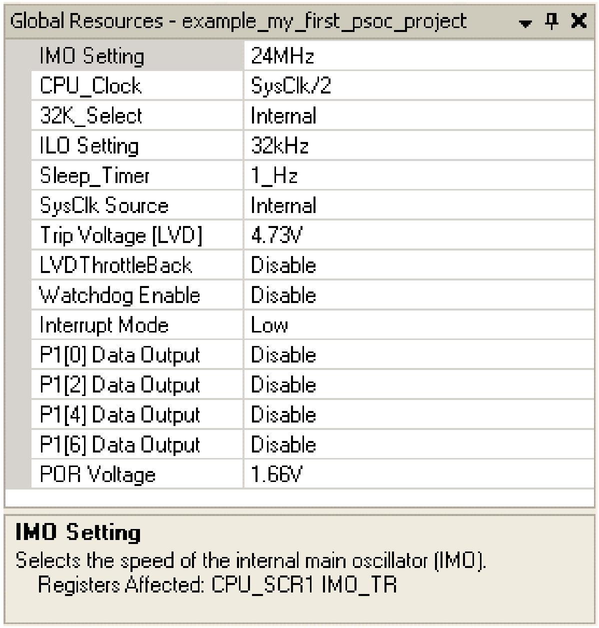



Figure 5-25. P1[2] Port Pin Properties



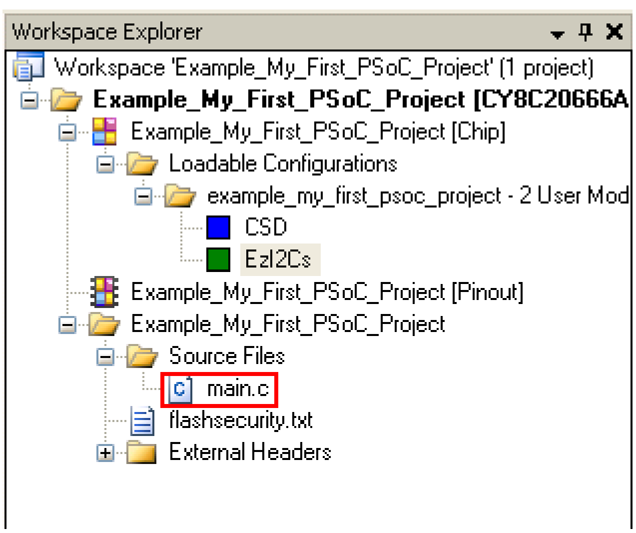
24. Configure the Global Resources window as follows.

Figure 5-26. Global Resources



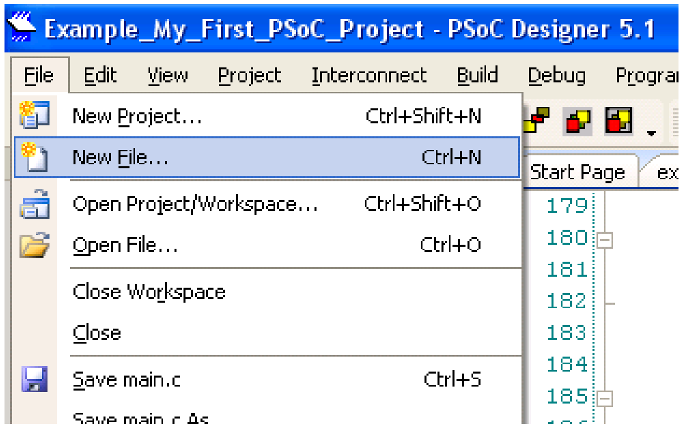

25. Open the existing *main.c* file in Workspace Explorer. Replace the existing *main.c* content with the content of the embedded *Example_My_First_PSoC_Project_Main.c* file, which is attached to this PDF.

Figure 5-27. Workspace Explorer



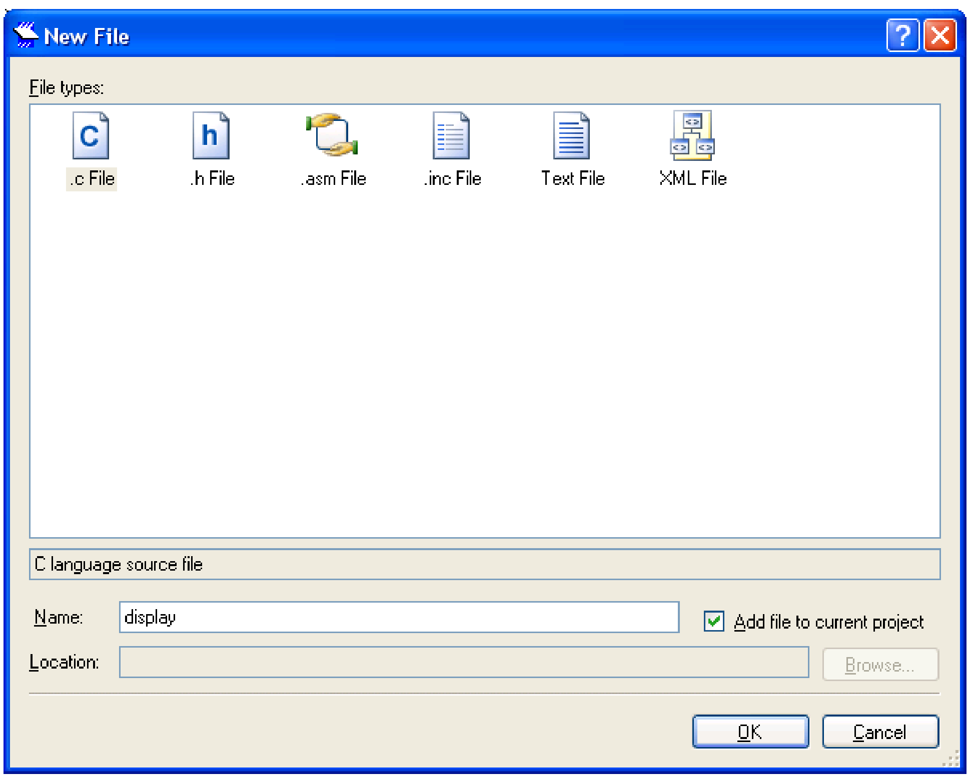
26. Add a new file to the project by clicking **File > New File**.

Figure 5-28. Add New File



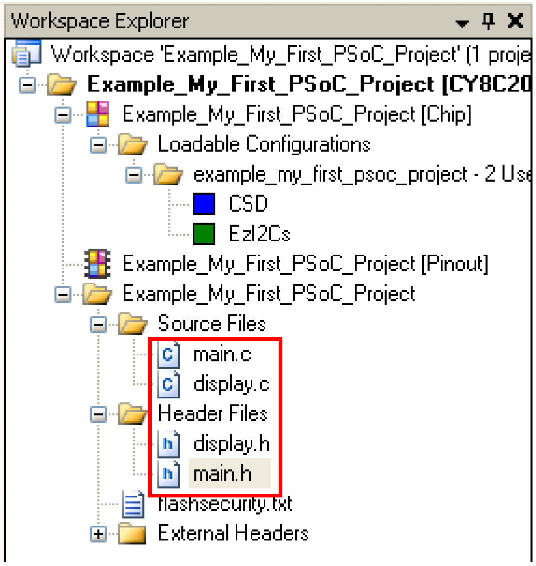

27. Select the **C** file type and name the file.

Figure 5-29. File Types



28. Similarly, create two .h file types and name them as **main.h** and **display.h**, respectively.

Figure 5-30. New Files in Project




29. Copy content of the *display.C*, *display.h,* and *main.h* files that is attached to this PDF to the respective files in the project.
30. Click **Build > Generate/Build 'Example_My_First_PSoC_Project'**.
31. Connect the CY3280-20X66 board to the PC using a MiniProg1.

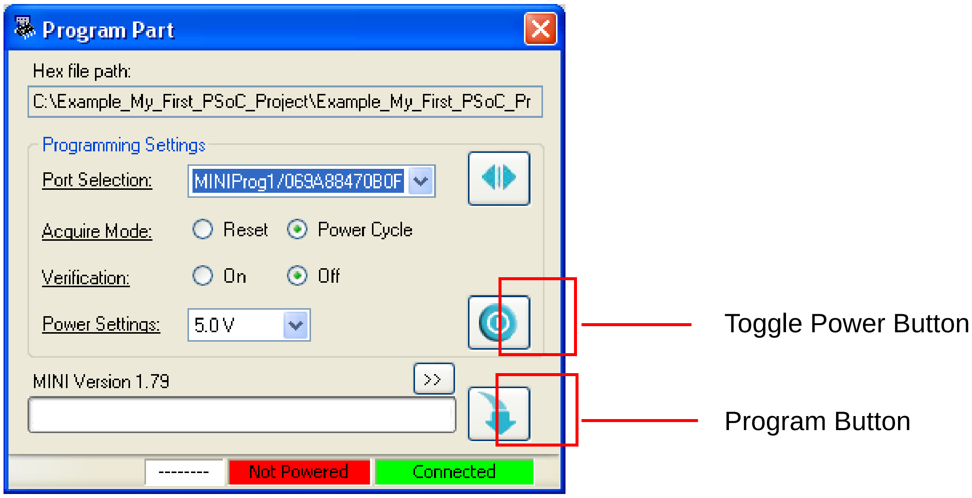
Figure 5-31. Connect MiniProg1 to Board


32. The board can be programmed either through the PSoC Designer IDE or PSoC Programmer. To program the board using PSoC Programmer, see Programming PSoC with New Design on page 20. To program the board through PSoC Designer, follow these steps.

    **Note** While programming the board with PSoC Designer, close any open instance of PSoC Programmer.

a. Click **Program > Program Part**.

Figure 5-32. Program Part Window




b. In Program Part window, make sure the following settings:

Port Selection drop-down, select **MiniProg1/xxxxxxxxxxx** and **Connected**

Acquire Mode: **Power Cycle**

Verification: **Off**

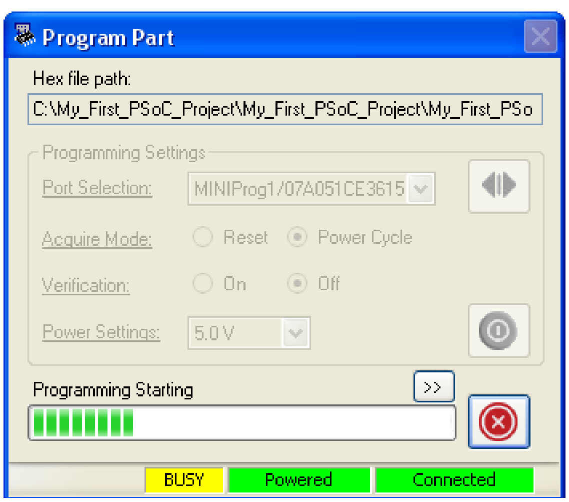
Power Settings: **5.0 V**

c. Click the **Program** button to start programming the board.

**Note** Ensure that the shorting jumpers is placed on pin 2 of J1 and pin 1 of J7.

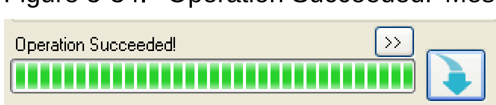
d. The board starts programming and the status is shown on the progress bar.

Figure 5-33. Programming Status




33.When the programming is done successfully, the 'Operation Succeeded!' Message is shown.

Figure 5-34. 'Operation Succeeded!' Message

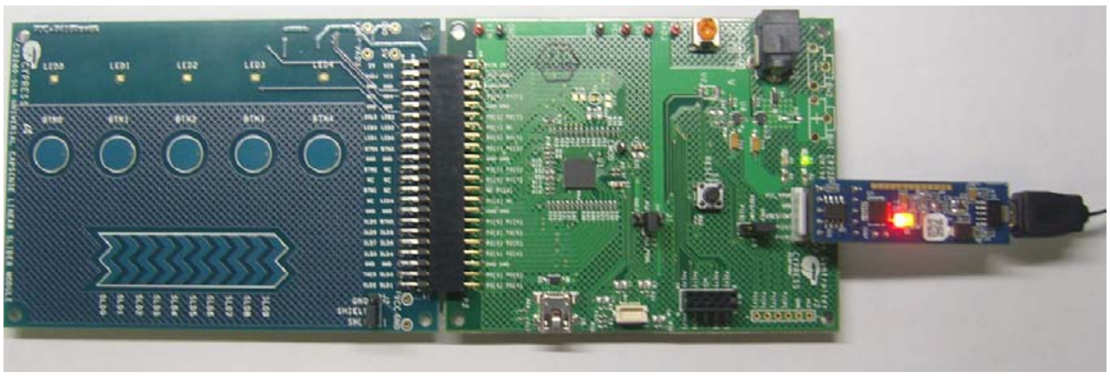


### 5.1.4 Verify Output

Follow these steps to verify the output:

1. Connect a Linear Slider Module (SLM) on connector P2 of the board.
2. Disconnect the MiniProg1 from the header J3 and connect a CY3240-I2USB bridge board in its place.
3. Connect a USB cable from the CY3240-I2USB bridge board to a free USB port on a PC.
4. Click **Start > Programs > Cypress > Bridge Control Panel <version> > Bridge Control Panel <version>**. **Note** Make sure that PSoC Designer and PSoC Programmer are closed before opening the Bridge Control Panel.
5. Select **Variable Settings** from the Chart menu.
6. Click **Load**, navigate to open the *CY3280_SLM_Project1.ini* file and click **OK**.
7. Click **Open File** from the File menu; navigate to and open the *CY3280_SLM_Project1.iic* file.
8. Select **+5.0V** in the Power box. Click **Toggle Power** to power the CY3240-I2USB bridge, LED (red) D1 glows.

Figure 5-35. SLM Module and CY3240-I2USB Bridge Connected to CapSense Controller



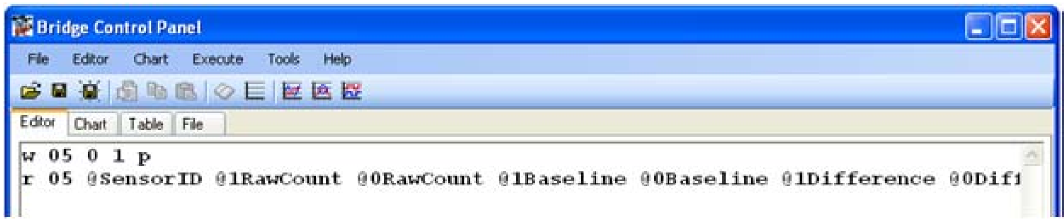
9. On the CapSense slider board, touch the button. Each button touch lights up the associated LED on the module board.
10. Touch the linear slider. The associated LED on the module board lights up, representing where your finger is on the slider.
11. The CapSense parameters such as the RawCount, Baseline, Difference Count and Mask Info (refer to the CSD User Module datasheet for more details of each parameter) for a particular sensor can be seen on the Bridge Control Panel. The current active button number and finger position on slider are also output on the Bridge Control Panel.
12. The syntax of the first command line, is as follows:

| | | | | |
|---|---|---|---|---|
| W | 05 | 0 | 1 | p |
| Write command | Slave Id (constant) | Address Offset (constant) | Sensor number (In hexadecimal, Valid range: 0x1 - 0xF) | Stop |

13. The first command writes to the Universal CapSense Controller (UCC) board, the sensor number for which the monitoring is required.
14. The second command line reads the CapSense parameters from the UCC board.

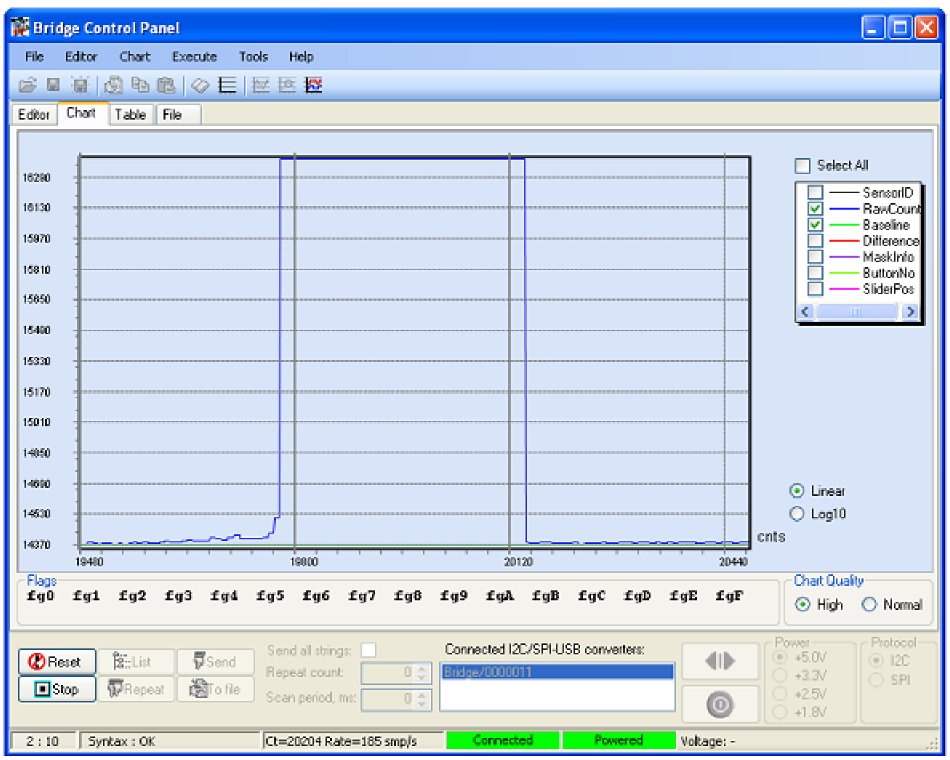
15. Change the sensor ID for which parameter monitoring is required and click **Send** to write the sensor ID to the Universal CapSense Controller board.

Figure 5-36. Command Line View



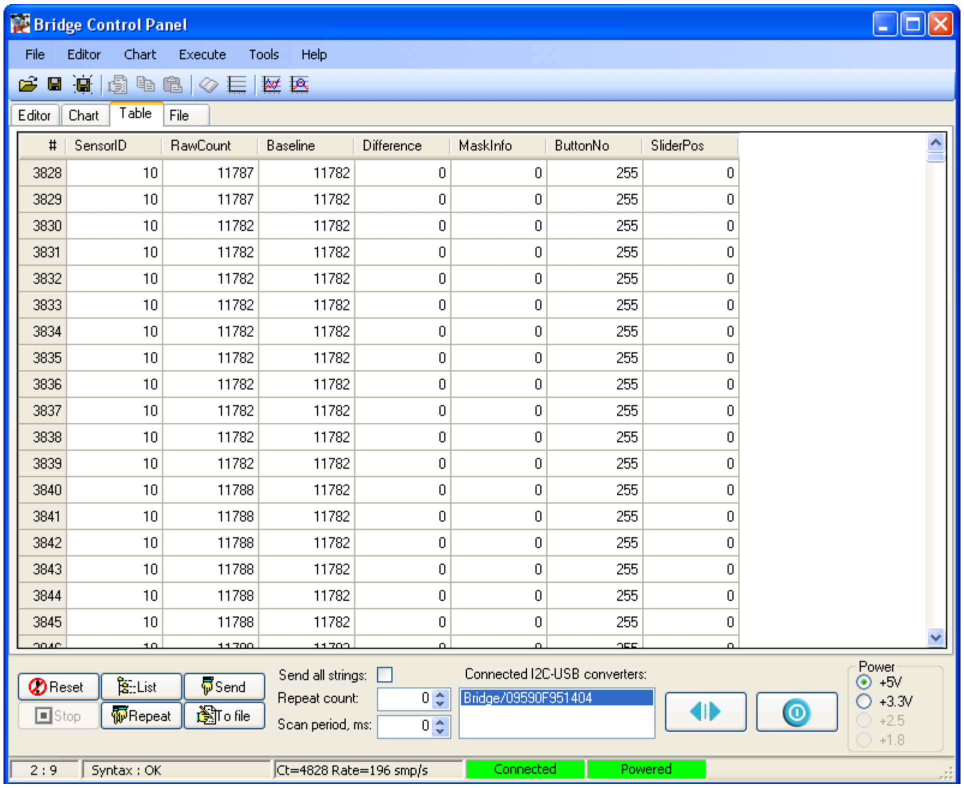

16. Click the second command line and then click **Repeat** to read I2Cdata received from the Universal CapSense Controller board.
17. Switch to the **Chart** tab to view the respective wave forms of CapSense parameters.

Figure 5-37. Bridge Control Panel Chart View



18. The various parameter values received by the UCC is displayed in the **Table** tab.

Figure 5-38. Bridge Control Panel Table View




## 5.2 CY3280_20x66_CSD_PD_Project2

### 5.2.1 Project Description

This project demonstrates the use of CapSense buttons and linear sliders using CSD technology and CY8C20x66A. The EzI2Cs User Module is used to transfer the raw count of CapSense parameters related to all the sensors to PC for monitoring.

This project scans five CapSense buttons and a 10-segment slider using the CSD User Module. There are five LEDs on the board, which illuminate when a CapSense button or slider is touched. The EzI2Cs User Module is used to provide a register-based I2C slave communications protocol. The status of CapSense sensors (both button and slider) and their parameters are updated in the I2C register, which can be accessed by any I2C master, similar to the I2USB bridge.

The application starts by executing *boot.asm*. The *boot.asm* initializes the hardware and invokes the 'main' function. The main function initializes the EzI2Cs slave and CapSense user modules. After initialization, the main function enters into a loop, which does the following:

- Scans all sensors
- Reads the sensor ID sent by the I2C master
- Stores CapSense data in the I2C registers
- Updates the LED status for the On/Off sensors

The following user modules are used in this project:

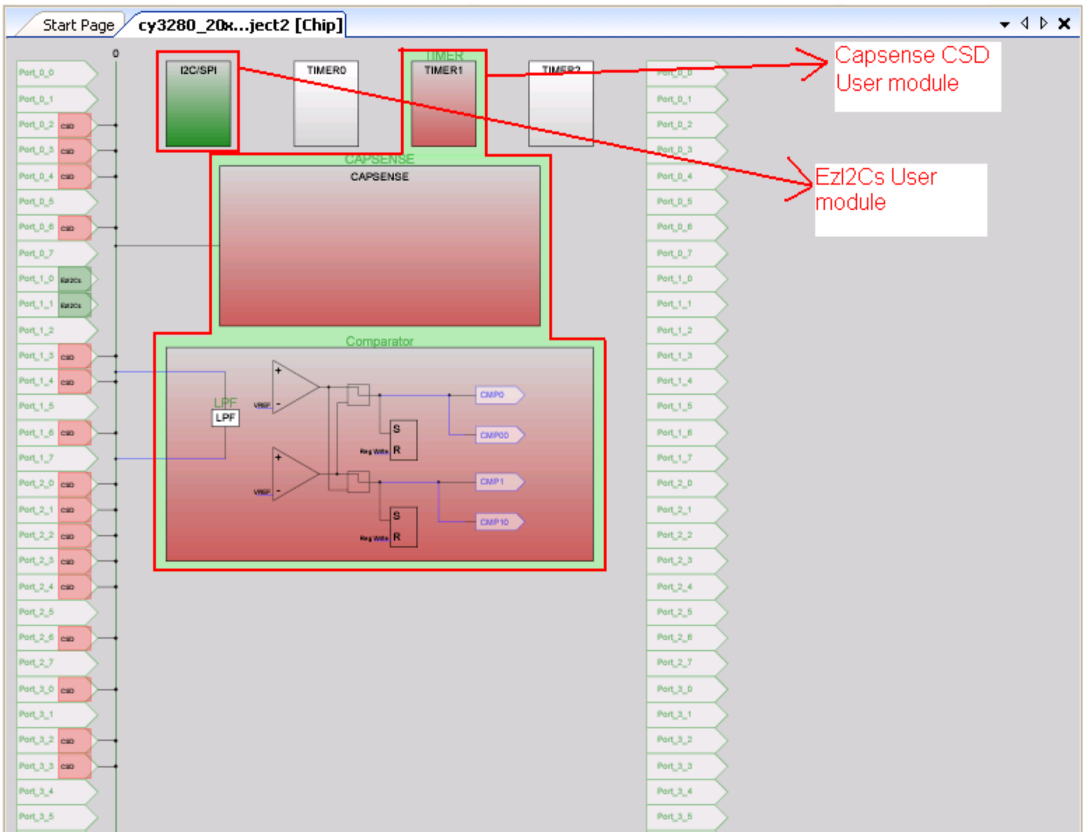
**CSD:** The CSD User Module provides capacitance sensing using the switched capacitor technique with a sigma-delta modulator to convert the sensed switching capacitor current to digital code.

**EzI2Cs:** The EzI2Cs User Module implements an I2C register based slave device. This user module does not require any digital or analog PSoC blocks. It is used to transfer all CapSense parameters related to a sensor to the PC for monitoring.

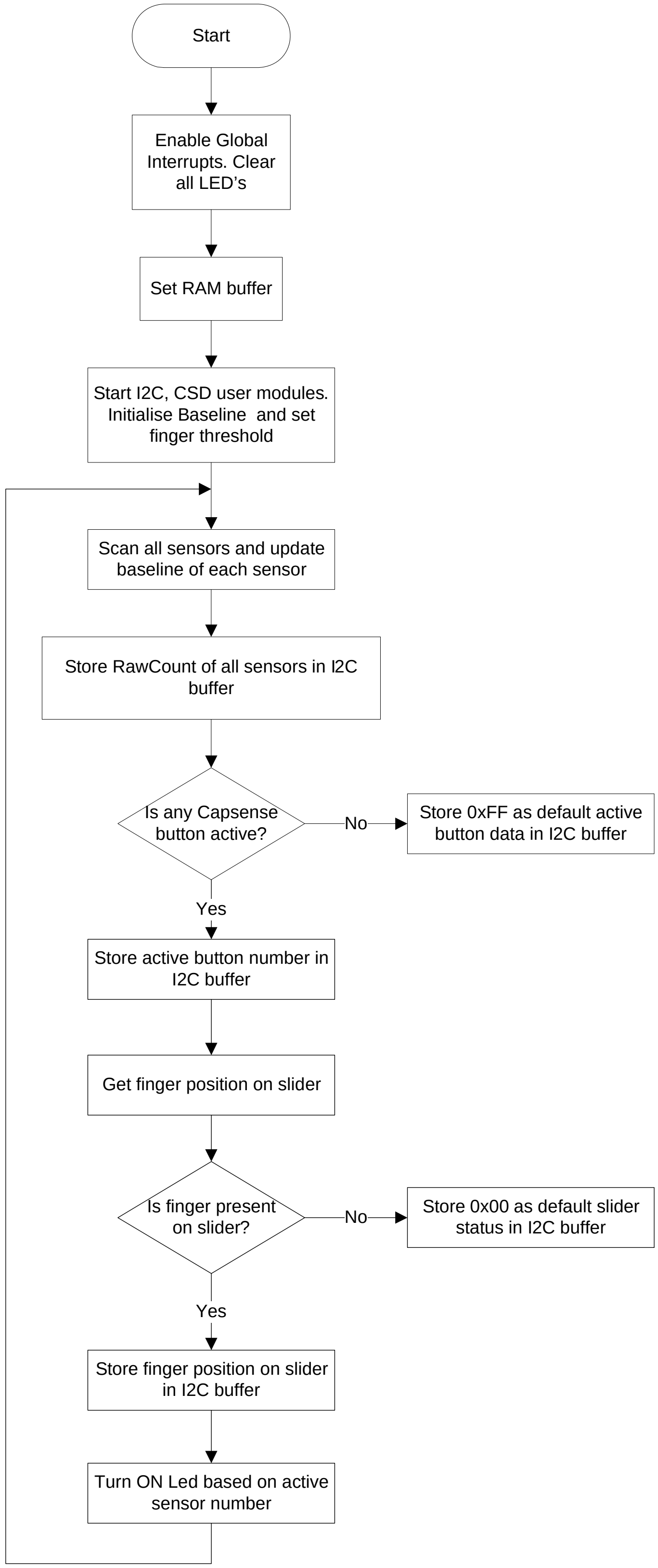
**Note:** To open the project from PSoC Designer, move the Firmware folder to a writable directory and then open it. The firmware folder is located at `<Install_Directory>\CY3280-BK1\<version>\Firmware`.

## 5.2.2 Device Configurations

Figure 5-39. Device Configuration for CY3280_20x66 CSD PD Project2.



### 5.2.3 Firmware Architecture



### 5.2.4 Verify Output

Load *CY3280_SLM_Project2.ini* and *CY3280_SLM_Project2.iic* file from the Bridge Control Panel, as explained in Bridge Control Panel on page 21. This file is available in the following location: <Install_Directory>\CY3280-20X66\<version>\Firmware\I2C-USBBridgeSoftwareConfig. Connect the CY3240-I2USB bridge board to the J3 header and to a free USB port on a PC through a USB cable, as explained in Programming PSoC with New Design on page 20.

1. Touch one or more buttons; the associated LEDs light up.
2. Touch the linear slider; the associated LEDs light up.
3. Touch the linear slider and buttons simultaneously. The associated LEDs light up corresponding to the buttons and the sliders being pressed.
4. Click the command line and then click **Repeat** to read I2C data received from the Universal CapSense Controller board.
5. Switch to the Chart tab to view respective wave forms of CapSense parameters; see Figure 5-41.

**Note** The character ‘r’ in Figure 5-40 defines the start of "read data" command.

Figure 5-40. Command Line View

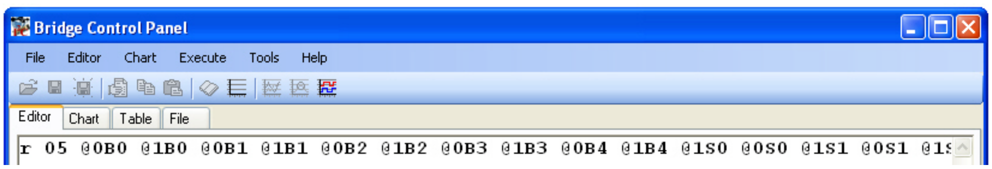
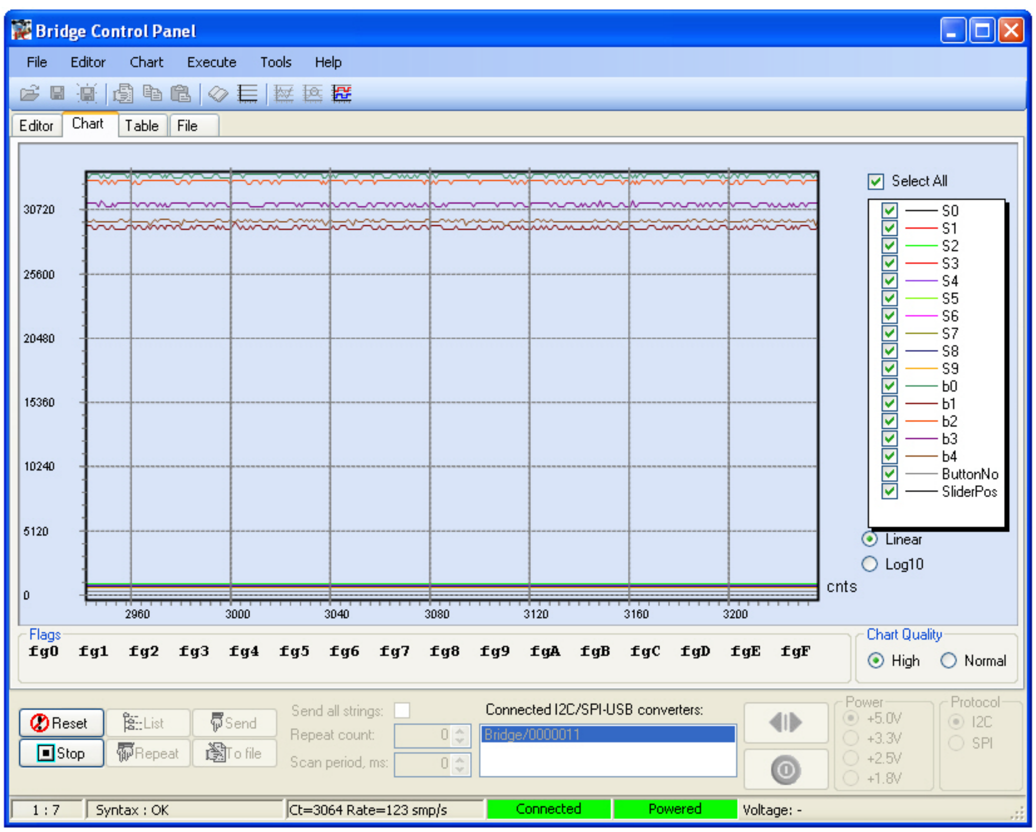



Figure 5-41. Bridge Control Panel Chart View




**Note** The brown line in the figure represents the axis.

## 5.3 CY3280_ 20x66_CSA_PD_Project1

### 5.3.1 Project Description

This project demonstrates the use of CapSense buttons and linear sliders using CSA technology and CY8C20x66A. The EzI2Cs User Module is used to transfer CapSense parameters related to a sensor to a PC for monitoring.

This project scans five CapSense buttons and a 10-segment slider using the CSA User Module. There are five LEDs on the board, which illuminate when a CapSense button or slider is touched. The EzI2Cs User Module is used to provide a register-based I2C slave communications protocol. The status of CapSense sensors (both button and slider) and their parameters are updated in the I2C register, which can be accessed by any I2C master, similar to the I2USB bridge.

The application starts by executing *boot.asm*. The *boot.asm* initializes the hardware and invokes the 'main' function. The main function initializes the EzI2Cs User Module and CapSense User Module. After initialization, the main function enters into a loop, which does the following:

- Scans all sensors
- Reads the sensor ID sent by the I2C master
- Stores CapSense data in the I2C registers
- Updates the LED status for the On/Off sensors

To make the CapSense sensors more sensitive, decrease the IDAC setting in the CSA User Module properties in PSoC Designer. However, this also increases the time it takes to scan each sensor. For more information on these or other parameters and the user module in general, see the CSA User Module datasheet.

The following user modules are used in this project:

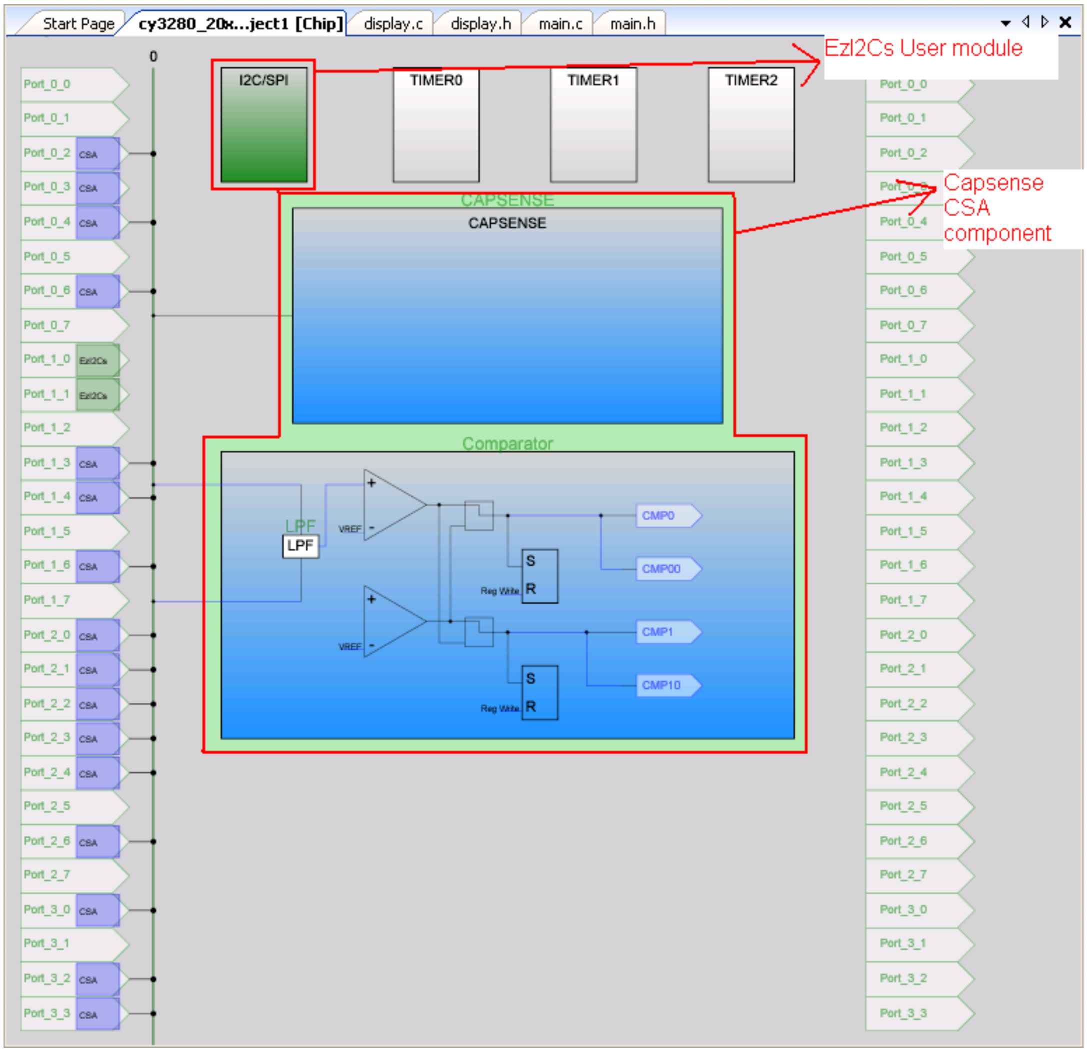
**CSA:** The CSA User Module implements an array of capacitive touch sensors using switched capacitor circuitry, an analog multiplexer, digital counting functions, and high-level software routines to compensate for environmental and physical sensor variations.

**EzI2Cs:** The EzI2Cs User Module implements an I2C register based slave device.This user module does not require any digital or analog PSoC blocks. The EzI2Cs User Module is used to transfer all CapSense parameters related to a sensor to the PC for monitoring.

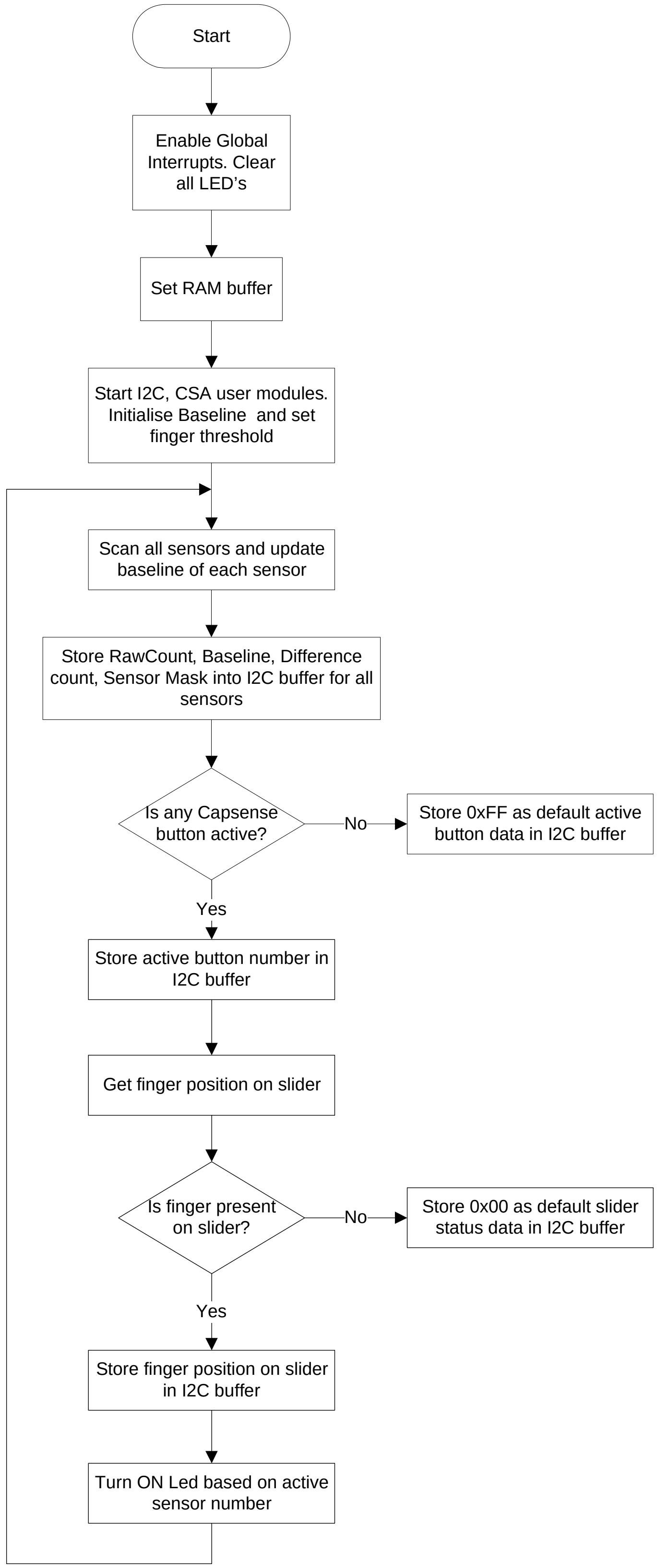
**Note:** To open the project from PSoC Designer, move the Firmware folder to a writable directory and then open it. The firmware folder is located at <Install_Directory>\CY3280-BK1\<version>\Firmware.

### 5.3.2 Device Configurations

Figure 5-42. Device Configuration for CY3280_20x66 CSA PD Project1.



### 5.3.3 Firmware Architecture



### 5.3.4 Verify Output

Load *CY3280_SLM_Project1.ini* and *CY3280_SLM_Project1.iic* file from the Bridge Control Panel, as explained in Bridge Control Panel on page 21. This file is available in the following location: `<Install_Directory>\CY3280-20X66\<version>\Firmware\I2C-USBBridgeSoftwareConfig`

1. Touch one or more buttons; the associated LEDs light up.
2. Touch the linear slider; the associated LEDs light up.
3. Touch the linear slider and buttons simultaneously. The associated LEDs light up corresponding to the buttons and the sliders being pressed.
4. The CapSense parameters such as the RawCount, Baseline, Difference Count and Mask Info (refer CSD User Module datasheet for more details of each parameter) for a particular sensor is displayed on the Bridge Control Panel. The current active button number and finger position on slider are also output on the Bridge Control Panel.
5. The syntax of the first command line, is as follows:

| | | | | |
|---|---|---|---|---|
| W | 05 | 0 | 1 | p |
| Write command | Slave Id | Address Offset | Sensor number | Stop |
| | (constant) | (constant) | (In hexadecimal, Valid range: 0x1 - 0xF) | |

6. The first command writes to the Universal CapSense Controller (UCC) board, the sensor number for which the monitoring is required.
7. The second command line reads the CapSense parameters from the UCC board.
8. Change the sensor ID for which parameter monitoring is required and click **Send** to write the sensor ID to the Universal CapSense Controller board.
9. Switch to **Chart** tab to view the respective waveforms of CapSense parameters; see Figure 5-44.

**Note** The character ‘w’ in Figure 5-43 defines the start of "write data" command. Similarly, the character ‘p’ generates stop condition on the I2C bus and ‘r’ defines start of "read data"command.

Figure 5-43. Command Line View

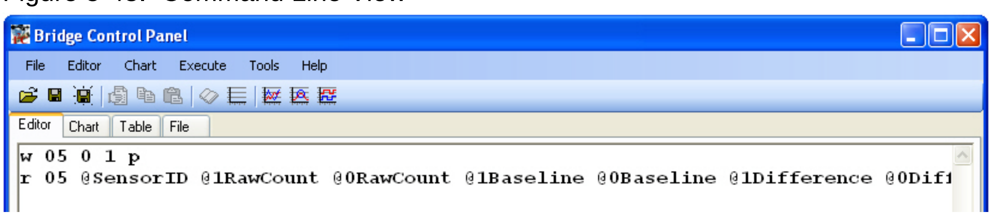
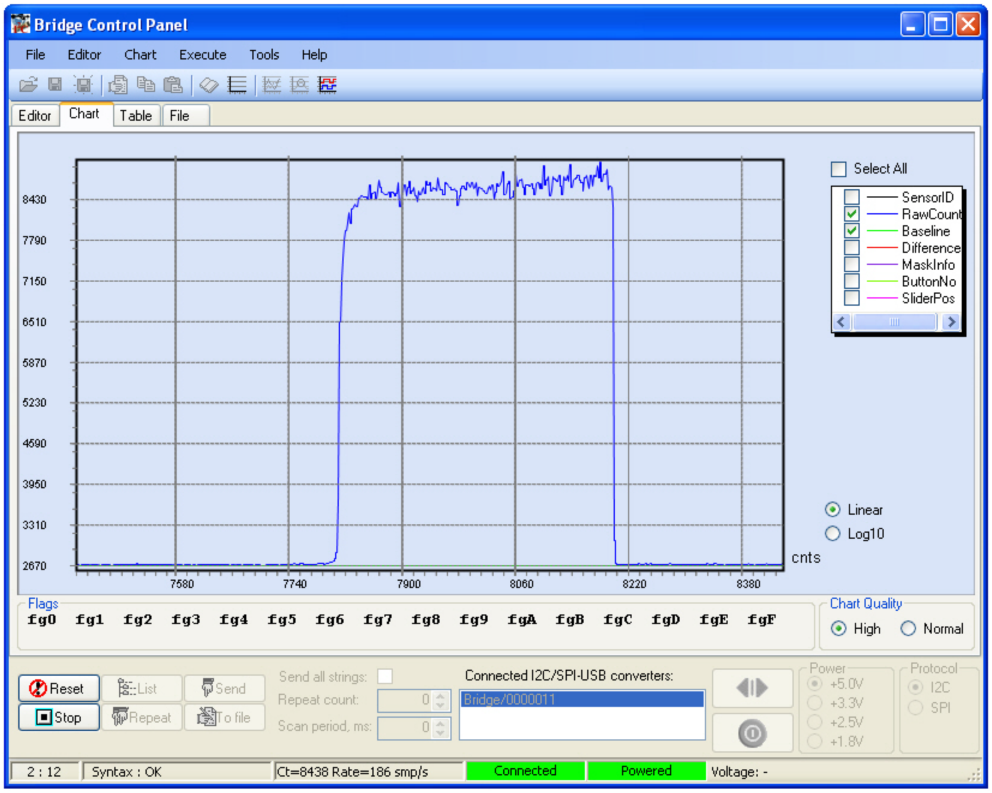


Figure 5-44. Bridge Control Panel Chart View




**Note** In the figure, the grey line represents the axis, the blue line indicates RawCount, and the green line indicates the Baseline.

## 5.4 CY3280_20x66_CSA_PD_Project2

### 5.4.1 Project Description

This project demonstrates the use of CapSense buttons and linear sliders using CSA technology and CY8C20x66A. The EzI2Cs User Module is used to transfer the raw count of CapSense parameters related to all the sensors to PC for monitoring.

This project scans five CapSense buttons and a 10-segment slider using the CSA User Module. There are five LEDs on the board, which illuminate when either a CapSense button or slider is touched. The EzI2Cs User Module is used to provide a register-based I2C slave communications protocol. The status of CapSense sensors (both button and slider) and their parameters are updated in the I2C register, which can be accessed by any I2C master, similar to the I2USB bridge.

The application starts by executing *boot.asm*. The *boot.asm* initializes the hardware and invokes the ‘main’ function. The main function initializes the EzI2Cs slave and CapSense user modules. After initialization, the main function enters into a loop, which does the following:

- Scans all sensors
- Reads the sensor ID sent by the I2C master
- Stores CapSense data in the I2C registers
- Updates the LED status for the On/Off sensors

To make the CapSense sensors more sensitive, you can decrease the IDAC setting in the CSA User Module properties within PSoC Designer. However, this also increases the time it takes to scan each sensor. For more information on these or other parameters, and the user module in general, see the CSA User Module datasheet.

The following user modules are used in this project:

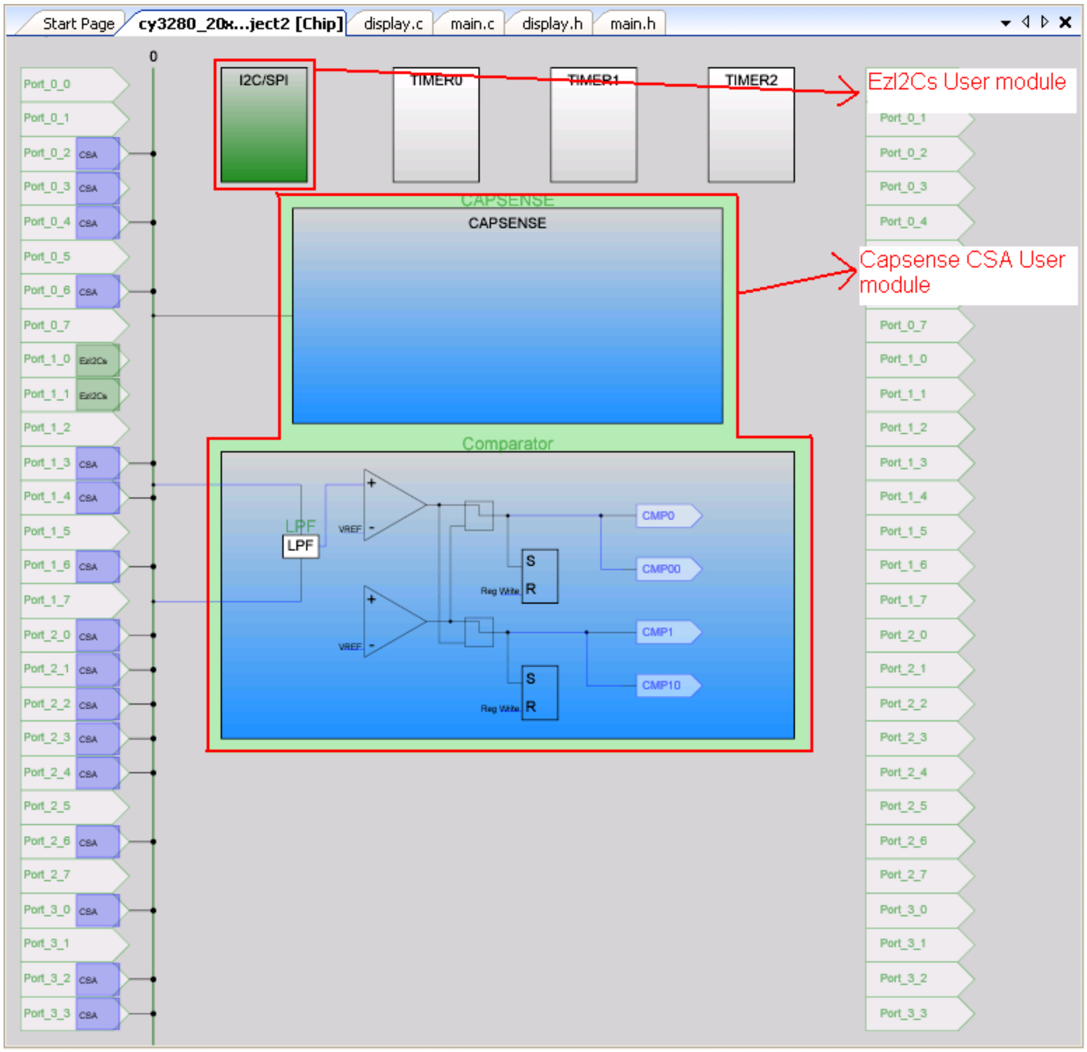
**CSA:** The CSA User Module implements an array of capacitive touch sensors using switched capacitor circuitry, an analog multiplexer, digital counting functions, and high-level software routines to compensate for environmental and physical sensor variations.

**EzI2Cs:** The EzI2Cs User Module implements an I2C register based slave device. This user module does not require any digital or analog PSoC blocks. The EzI2Cs User Module is used to transfer all CapSense parameters related to a sensor to the PC for monitoring.

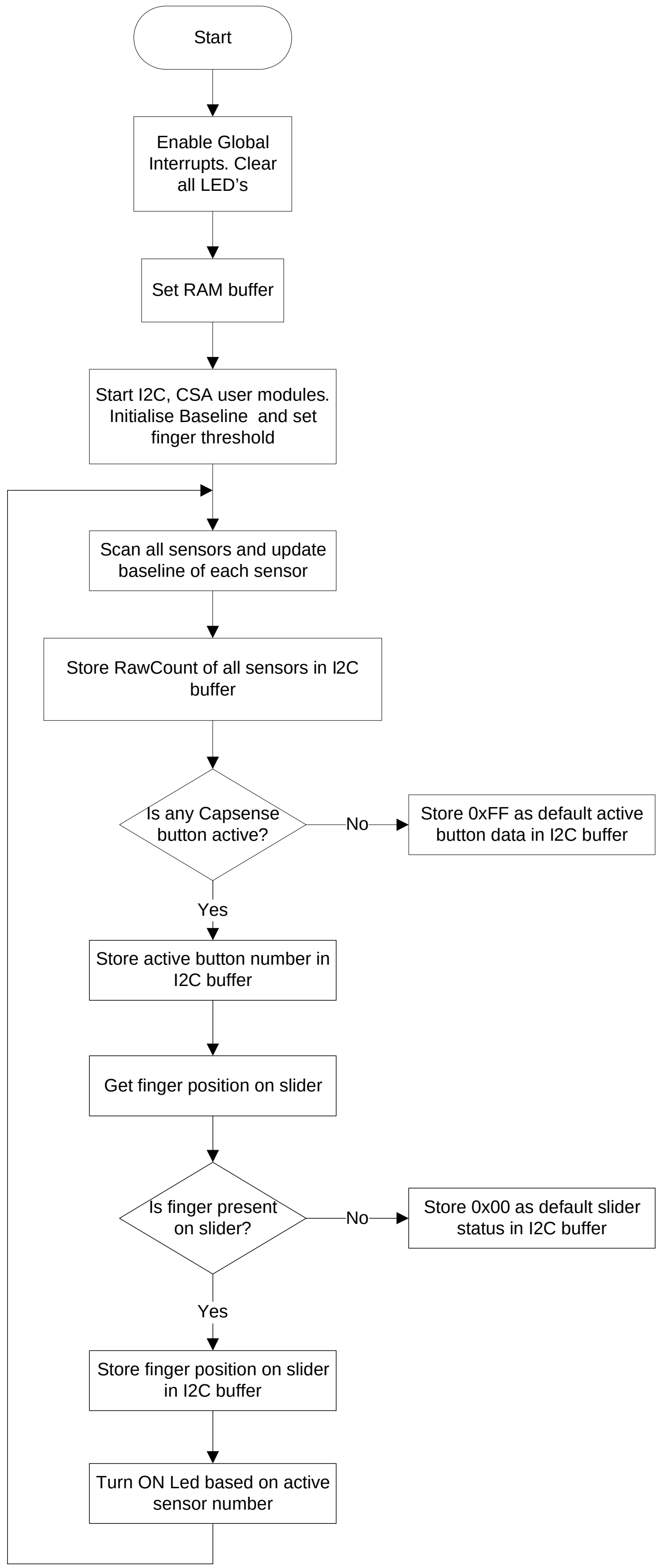
**Note:** To open the project from PSoC Designer, move the Firmware folder to a writable directory and then open it. The firmware folder is located at <Install_Directory>\CY3280-BK1\<version>\Firmware.

### 5.4.2 Device Configurations

Figure 5-45. Device Configuration for CY3280_20x66 CSA PD Project2



### 5.4.3 Firmware Architecture



### 5.4.4 Verify Output

Load *CY3280_SLM_Project2.ini* and *CY3280_SLM_Project2.iic* file from the Bridge Control Panel, as explained in Bridge Control Panel on page 21. This file is available in the following location: <Install_Directory>\CY3280-20X66\<version>\Firmware\I2C-USBBridgeSoft-wareConfig

1. Touch one or more buttons; the associated LEDs light up.
2. Touch the linear slider: the associated LEDs light up.
3. Touch the linear slider and buttons simultaneously. The associated LEDs light up corresponding to the buttons and the sliders being pressed.
4. Click the command line and then click **Repeat** to read I2C data received from the Universal CapSense Controller board.
5. Switch to the **Chart** tab to view respective waveforms of CapSense parameters; see Figure 5-47.

Figure 5-46. Command Line View

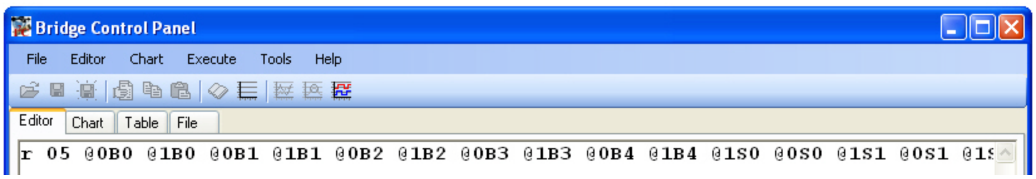


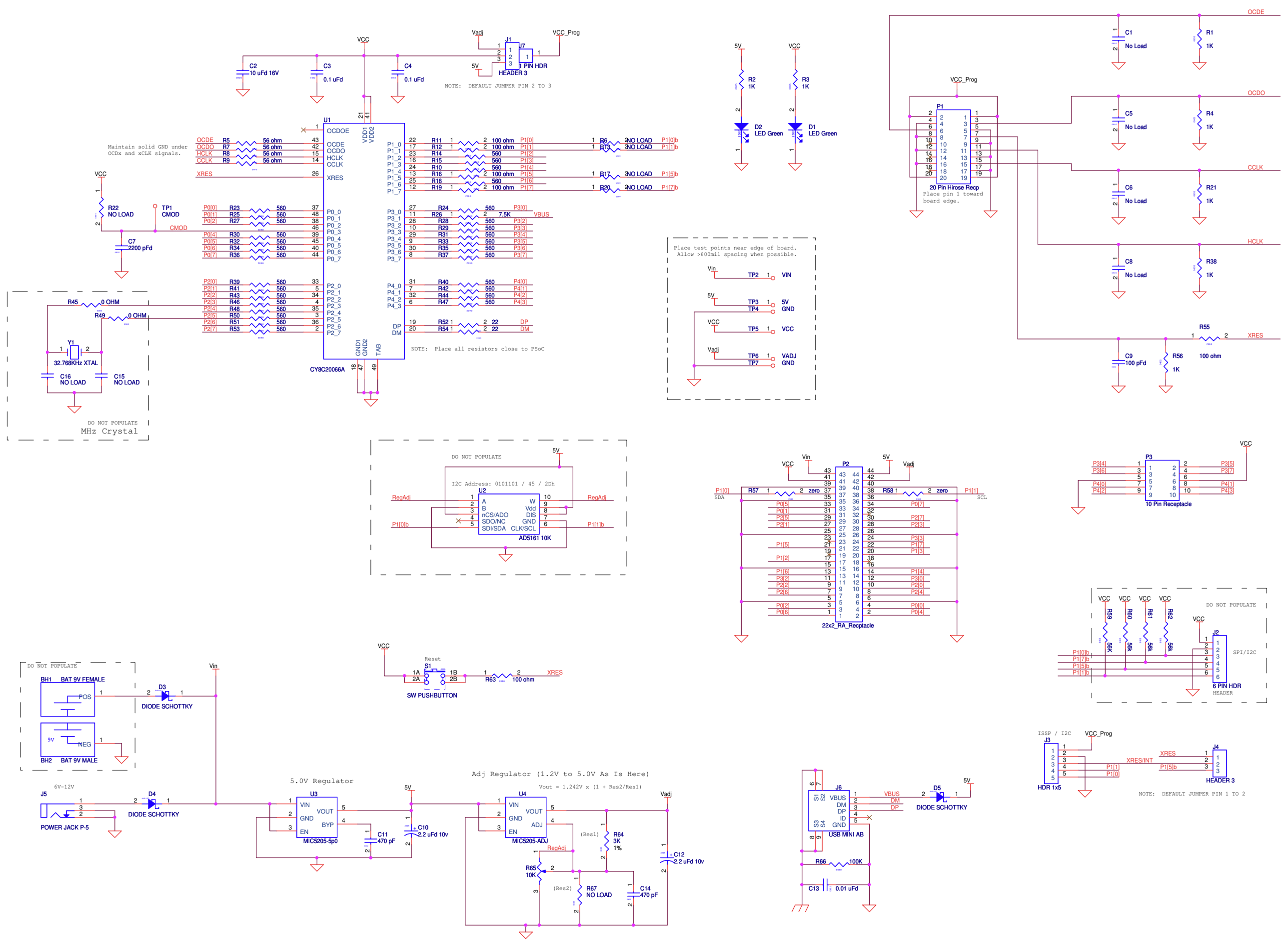

Figure 5-47. Bridge Control Panel Chart View


**Note** The brown line in the figure represents the axis.

# A. Appendix

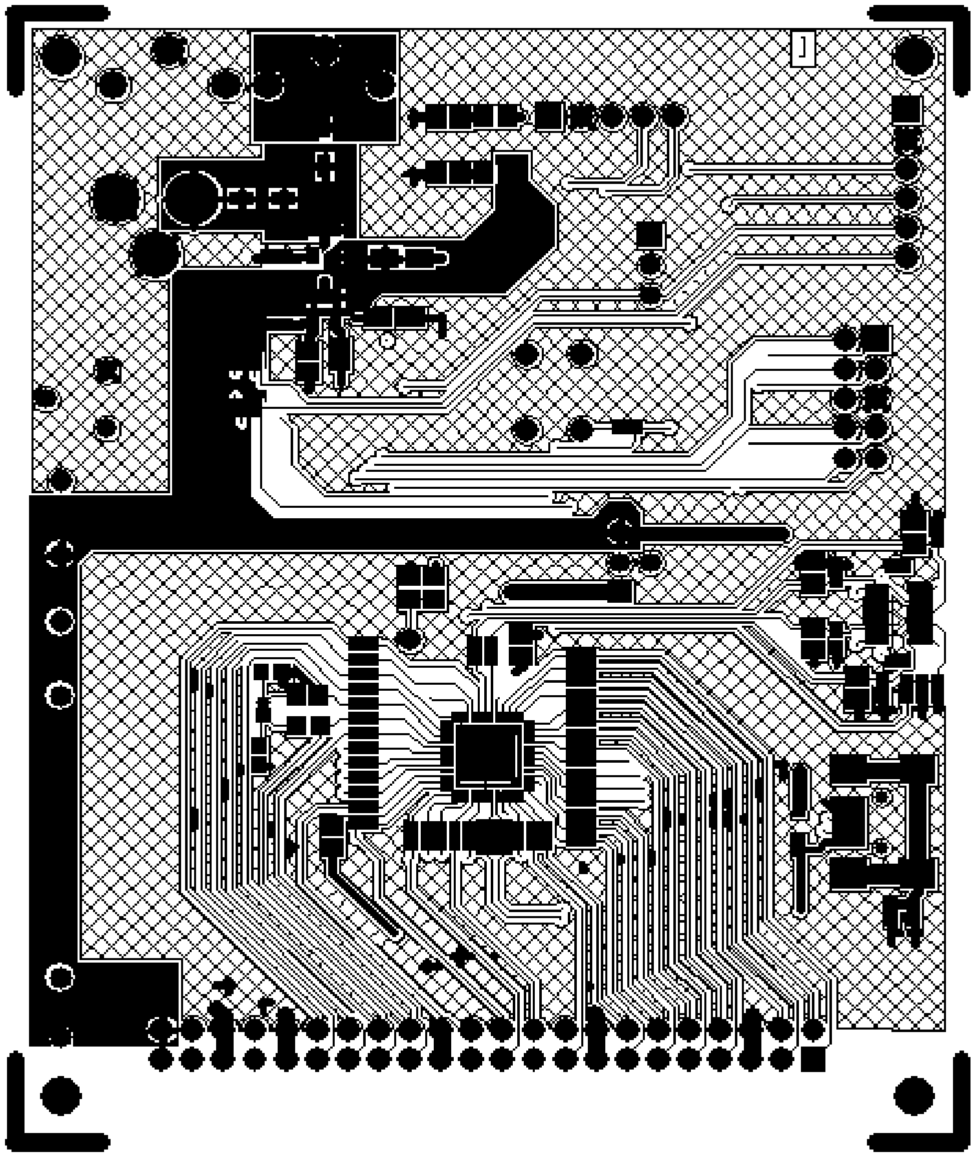
The schematic, board layouts, and BOM are available on the CY3280-20x66 kit CD or at: <Install_directory>\CY3280-20x66\<version>\Hardware.

## A.1 Schematic

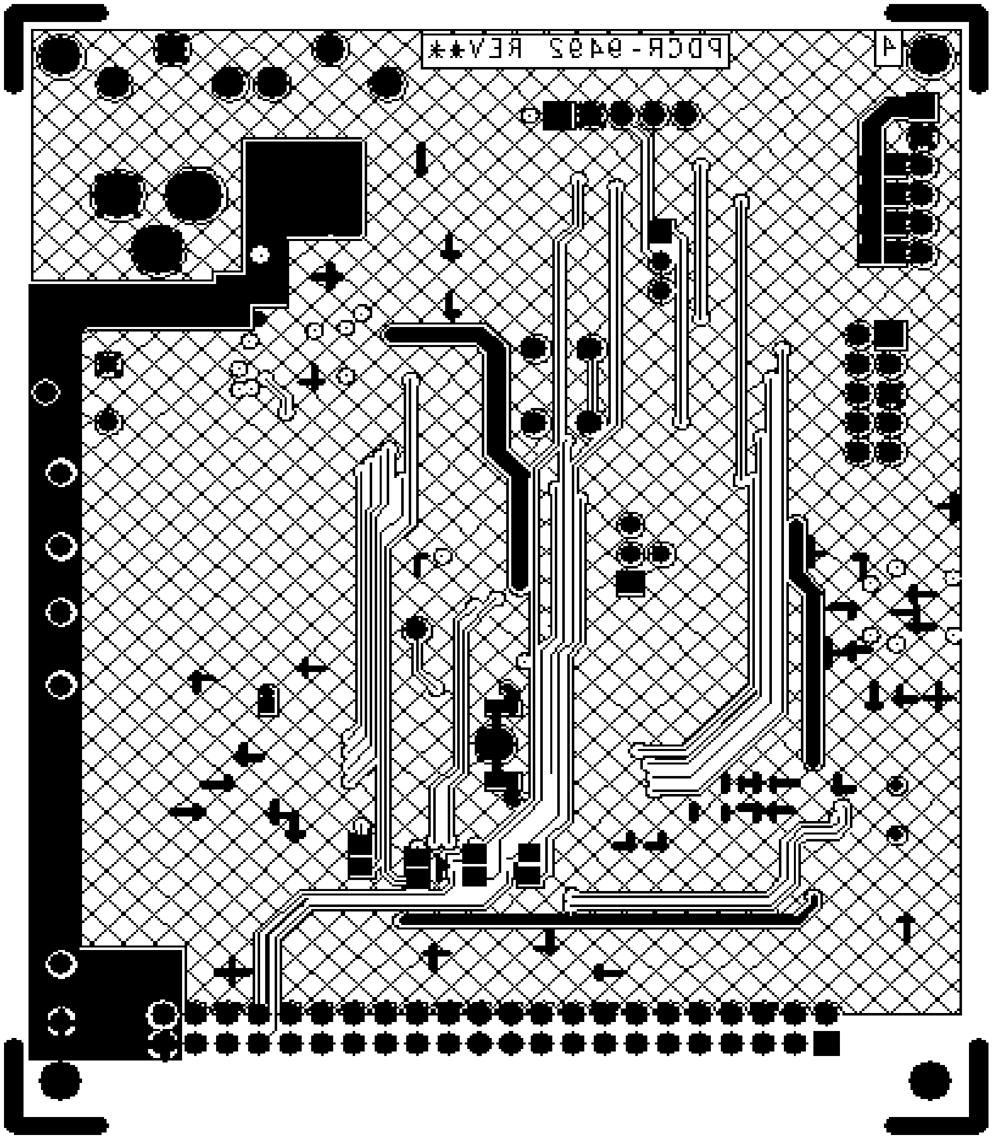


## A.2 Board Layout

### A.2.1 PDCR-9492 Top



### A.2.2 PDCR-9492 Bottom



## A.3 BOM

| No | Qty | Reference | Part | Manufacturer | Manuf. Part# |
|---|---|---|---|---|---|
| 1 | 1 | C2 | CAP CER 10UF 16V X5R 0805 | Murata Electronics North America | GRM21BR61C106KE15L |
| 2 | 2 | C3,C4 | CAP .1UF 16V CERAMIC Y5V 0402 | Panasonic - ECG | ECJ-0EF1C104Z |
| 3 | 1 | C7 | CAP CER 2200PF 50V 5% C0G 0805 | Murata Electronics North America | GRM2165C1H222JA01D |
| 4 | 1 | C9 | CAP 100PF 50V CERAMIC 0402 SMD | Panasonic - ECG | ECJ-0EC1H101J |
| 5 | 2 | C10,C12 | CAP 2.2UF 10V TANTALUM 10% 3216 | AVX | TPSA225K010R1800 |
| 6 | 2 | C11,C14 | CAP 470PF 50V CERAMIC 0402 SMD | Panasonic - ECG | ECJ-0EB1H471K |
| 7 | 1 | C13 | CAP 10000PF 16V CERAMIC 0402 SMD | Panasonic - ECG | ECJ-0EB1C103K |
| 8 | 2 | D1,D2 | LED GREEN CLEAR 0805 SMD | LITE-ON | LTST-C170GKT |
| 9 | 2 | D4,D5 | DIODE SCHOTTKY 0.5A 20V SOD-123 | Fairchild Semiconductor | MBR0520L |
| 10 | 2 | J1,J4 | CONN HEADER VERT 3POS .100 30AU | AMP Division of TYCO | 87220-3 |
| 11 | 1 | J3 | CONN HEADER 5POS 0.1 VERT KEYED | Molex | 22-23-2051 |
| 12 | 1 | J5 | CONN 2.1MM PWRJACK RT ANGLE PCB | Switchcraft | RAPC722X |
| 13 | 1 | J6 | CONN USB MINI AB SMT RIGHT ANGLE | TYCO | 1734035-2 |
| 14 | 1 | J7 | CONN HEADER VERT 1POS .100 | TYCO | 9-146280-0-01 |
| 15 | 1 | P1 | RECP VERT 20POS HIROSE | Hirose | DF12-5.0-20DP-0.5V-81 |
| 16 | 1 | P2 | CONN FMALE 44POS DL .1" R/A GOLD | Sullins Electronics Corp. | PPPC222LJBN-RC |
| 17 | 1 | P3 | CONN RCPT .100 DUAL STR 10POS | 3M | 929852-01-05-RA |
| 18 | 5 | R1,R4,R21,R38,R56 | RES 1.0K OHM 1/16W 5% 0402 SMD | Phycomp USA Inc | 9C1A04021001JLHF3 |
| 19 | 2 | R2,R3 | RES 1K OHM 1/10W 5% 0805 SMD | Panasonic - ECG | ERJ-6GEYJ102V |
| 20 | 4 | R5,R7,R8,R9 | RES 56 OHM 1/16W 5% 0402 SMD | Panasonic - ECG | ERJ-2GEJ560X |
| 21 | 30 | R10,R14,R15,R18,R23, R24,R25,R27,R28,R29, R30,R31,R32,R33,R34, R35,R36,R37,R39,R40, R41,R42,R43,R44,R46, R47,R48,R50,R51,R53 | RES 560 OHM 1/16W 5% 0402 SMD | Yageo Corporation | RC0402JR-07560RL |
| 22 | 6 | R11,R12,R16,R19,R55, R63 | RES 100 OHM 1/16W 5% 0402 SMD | Rohm | MCR01MZPJ101 |
| 23 | 1 | R26 | RES 7.5K OHM 1/8W 5% 0805 SMD | Panasonic - ECG | ERJ-6GEYJ752V |
| 24 | 2 | R52,R54 | RES 22 OHM 1/16W 5% 0603 SMD | Panasonic - ECG | ERJ-3EKF22R0V |
| 25 | 2 | R57,R58 | RES ZERO OHM 1/16W 0402 SMD | Panasonic - ECG | ERJ-2GE0R00X |
| 26 | 1 | R64 | RES 3.00K OHM 1/16W 1% 0603 SMD | Yageo America | 9C06031A3001FKHFT |
| 27 | 1 | R65 | POT 10K CARBON LAYDOWN (103) | Panasonic - ECG | EVN-D8AA03B14 |
| 28 | 1 | R66 | RES 100K OHM 1/16W 5% 0402 SMD | Panasonic - ECG | ERJ-2GEJ104X |
| 29 | 1 | S1 | LT SWITCH 6MM 100GF H=7MM TH | Panasonic - ECG | EVQ-PAC07K |
| 30 | 3 | TP1,TP4,TP7 | TEST POINT 43 HOLE 65 PLATED BLACK | Keystone Electronics | 5001 |
| 31 | 4 | TP2,TP3,TP5,TP6 | TEST POINT 43 HOLE 65 PLATED RED | Keystone Electronics | 5000 |
| 32 | 1 | U1 | IC, 48QFN PSoC Device w/ OCD | Cypress Semiconductor | CY8C20066A-24LTXI |
| 33 | 1 | U3 | IC REG LDO 150MA 5.0V 1% SOT23-5 | Micrel | MIC5205-5.0YM5 |
| 34 | 1 | U4 | IC REG LDO 150MA ADJ 1% SOT23-5 | Micrel | MIC5205YM5 |

| No | Qty | Reference | Part | Manufacturer | Manuf. Part# |
|---|---|---|---|---|---|
| Do Not Load | | | | | |
| 35 | 1 | BH1 | BATTERY HOLDER 9V Female PC MT | Keystone Electronics | 594 |
| 36 | 1 | BH2 | BATTERY HOLDER 9V Male PC MT | Keystone Electronics | 593 |
| 37 | 1 | D3 | DIODE SCHOTTKY 0.5A 20V SOD-123 | Fairchild Semiconductor | MBR0520L |
| 38 | 1 | J2 | CONN HEADER VERT 6POS .100 TIN | Molex/Waldom Electronics | 22-28-4060 |
| 39 | 3 | R59,R60,R61,R62 | RES 560 OHM 1/16W 5% 0402 SMD | Panasonic - ECG | ERJ-2GEJ563X |
| 40 | 6 | R6,R13,R17,R20,R22, R67 | RES NO LOAD 0805 SMD | NA | NA |
| 41 | 1 | U2 | IC DGTL POT SPI 10K 10-MSOP | Analog Devices Inc | AD5161BRMZ10 |
| 42 | 2 | R45,R49 | RES NO LOAD 0805 SMD | NA | NA |
| 43 | 1 | Y1 | CRYSTAL 32.768 KHZ CYL 12.5PF CFS308 | Citizen America Corpora-tion | CFS308-32.768KDZF-UB |
| 44 | 4 | C1,C5,C6,C8 | CAP NO LOAD 0805 | NA | NA |
| 45 | 4 | C15,C16 | CAP NO LOAD 0603 | NA | NA |
| **Additional Assembly Instructions** | | | | | |
| 46 | Place jumper (0.100" pitch) across pins 2 and 3 of J1 | | | | |
| 47 | Place jumper (0.100" pitch) across pins 1 and 2 of J4. | | | | |
| **Install On Bottom of PCB As Close To Corners As Possible** | | | | | |
| 48 | 4 | n/a | BUMPER CLEAR.370 x 19" CYLINDER | Richco Plastic Co | RBS-35 |

# Revision History

## Document Revision History

| Document Title: CY3280-20x66 Universal CapSense® Controller Kit Guide<br>Document Number: 001-67447 | | | | |
|---|---|---|---|---|
| **Revision** | **ECN#** | **Issue Date** | **Origin of Change** | **Description of Change** |
| ** | 3175811 | 02/17/2011 | RKPM | Initial version of kit guide. |
| *A | 3279588 | 06/10/2011 | SASH | Updated Code Examples chapter on page 35:<br>Added "My First Code Example (CY3280_20x66_CSD_PD_Project1)" on page 35.<br>Text and image updates throughout the document. |
| *B | 3983114 | 05/26/2013 | ZINE | Updated Introduction chapter on page 5:<br>Updated "Additional Learning Resources" on page 7:<br>Added "Code Examples". |
| *C | 4309193 | 03/14/2014 | SSHH | No technical updates.<br>Completing Sunset Review. |
| *D | 4497435 | 09/08/2014 | DIMA | Updated the kit code examples and user guide to support the latest version of PSoC Designer (v5.4). |
| *E | 4737229 | 04/23/2015 | DCHE | Updated Introduction chapter on page 5:<br>Updated "Additional Learning Resources" on page 7:<br>Updated description.<br>Removed "Code Examples".<br>Added "PSoC Designer" on page 8.<br>Added "Code Examples" on page 9.<br>Added "PSoC Designer Help" on page 11.<br>Added "Technical Support" on page 11.<br>Updated Code Examples chapter on page 35:<br>Updated "My First Code Example (CY3280_20x66_CSD_PD_Project1)" on page 35:<br>Updated "Flowchart" on page 36:<br>Updated diagram.<br>Updated to new template. |

| Document Title: CY3280-20x66 Universal CapSense® Controller Kit Guide<br>Document Number: 001-67447 | | | | |
|---|---|---|---|---|
| **Revision** | **ECN#** | **Issue Date** | **Origin of Change** | **Description of Change** |
| *F | 5204633 | 04/13/2016 | ASRI | Updated hyperlinks across the document.<br>Updated Getting Started chapter on page 13:<br>Updated “Kit Installation” on page 13:<br>Updated description.<br>Removed figure “Root Directory of the CD”.<br>Removed figure “Install Shield Wizard”.<br>Updated Figure 2-2.<br>Removed figure “Installation Page”.<br>Updated Figure 2-3.<br>Updated “PSoC Designer” on page 16:<br>Updated Figure 2-4.<br>Updated to new template. |
| *G | 5318990 | 06/22/2016 | ASRI | Updated cross-references<br>Modified project names in Code Examples chapter on page 35 |
| **Distribution**: External<br>**Posting**: None | | | | |