

PSoC® 4 Segment LCD Direct Drive

Author: Nidhin MS and Rahul Raj Sharma

Associated Project: Yes

Associated Part Family: **PSoC 4100** and **PSoC 4200**

Software Version: **PSoC Creator™ 4.0** or higher

Related Application Notes: **AN52927**, **AN56384**

More code examples? We heard you.

To access an ever-growing list of hundreds of PSoC code examples, please visit our [code examples web page](#).
 You can also explore the PSoC 4 video library [here](#).

AN87391 demonstrates how easy it is to drive segment LCD glass using the integrated LCD driver in PSoC® 4. PSoC 4 offers multiple LCD drive modes, very low power consumption, and provides a complete user interface and system solution. Using example projects, this application note explains low-power LCD design and a user interface solution that includes CapSense®.

Contents

1	Introduction.....	1	8.1	Project 1: Simple Segment LCD	14
2	PSoC Resources	2	8.2	Project 2: Low-Power Segment LCD	21
2.1	PSoC Creator	2	8.3	Project 3: Segment LCD and CapSense User Interface	24
2.2	Code Examples	4	8.4	Project 4: Alphanumeric Segment LCD	26
2.3	PSoC Creator Help	5	9	Summary	27
2.4	Technical Support.....	5		Document History.....	28
3	Features of Segment LCD Drive in PSoC 4	6		Worldwide Sales and Design Support.....	29
4	Basics of Segment LCD	6		Products	29
5	PSoC Creator Component: Segment LCD	8		PSoC® Solutions	29
6	PSoC 4 Segment LCD Direct Drive	8		Cypress Developer Community.....	29
6.1	PWM Drive Mode.....	8		Technical Support	29
6.2	Digital Correlation Drive Mode	9			
6.3	Low-Power Operation	10			
7	PSoC 4 User Interface Solution.....	12			
8	PSoC Creator Projects	14			

1 Introduction

Most low-power portable handheld devices such as glucose meters, multimeters, and blood pressure monitors use a segment LCD to display information. Segment LCDs require an external driver to interface with a microcontroller. PSoC 4 implements an integrated low-power LCD driver that can directly drive segment LCD glass.

PSoC 4 is a true programmable embedded system-on-chip integrating configurable analog and digital peripheral functions, capacitive touch sensing, memory, and a 32-bit ARM® Cortex®-M0 microcontroller on a single chip. PSoC 4 accelerates time-to-market, integrates critical system functions, and reduces overall system cost. The capacitive touch sensing capability of PSoC 4 (CapSense), together with LCD drive allows you to implement an entire user interface solution using only PSoC 4. PSoC 4 has flexible low-power modes to reduce overall power consumption. For example, the segment LCD current consumption is as low as 3.1 µA when the PSoC is in its Deep-Sleep low-power mode. The PSoC Creator™ IDE provides a Segment LCD Component that makes it easy to use different types of segment LCD modules with PSoC 4.

This application note explains the features offered by PSoC 4's segment LCD drive, shows best practices for low-power LCD design, and demonstrates user interface solutions with the help of example projects.

If you are new to PSoC 4, refer to [AN79953, Getting Started with PSoC® 4](#). If you are new to PSoC Creator, see the [PSoC Creator home page](#). If you are new to PSoC 4 power modes, refer to [AN86233, PSoC 4 Low-Power Modes and Power Reduction Techniques](#).

2 PSoC 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 [KBA86521, How to Design with PSoC 3, PSoC 4, and PSoC 5LP](#). Here is an abbreviated list for PSoC 4:

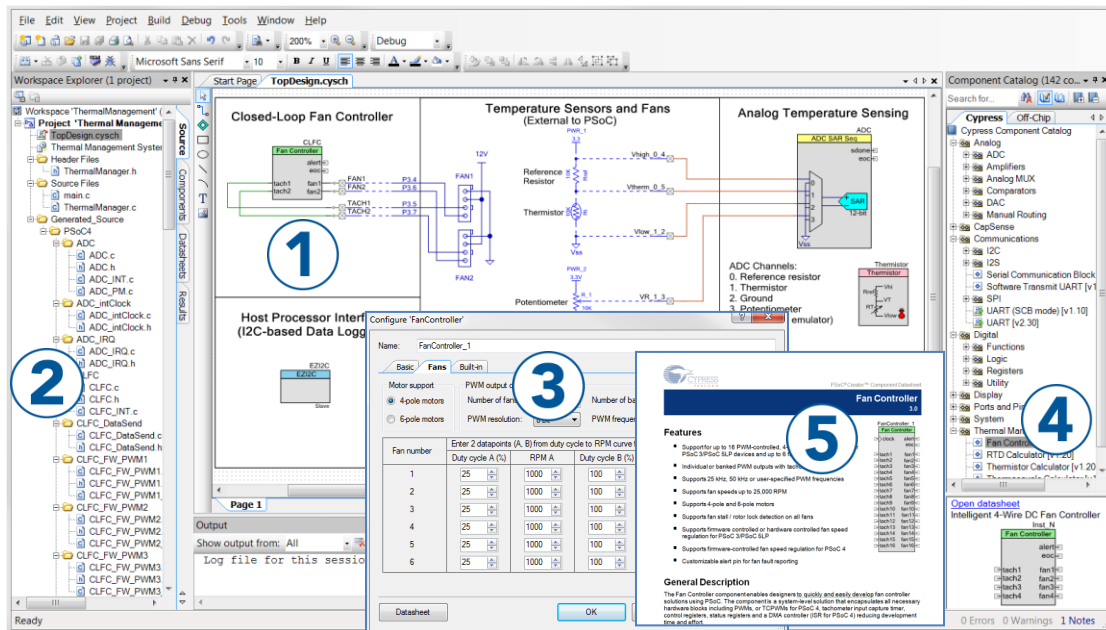
- **Overview:** [PSoC Portfolio](#), [PSoC Roadmap](#)
- **Product Selectors:** [PSoC 1](#), [PSoC 3](#), [PSoC 4](#), or [PSoC 5LP](#). In addition, [PSoC Creator](#) includes a device selection tool.
- **Datasheets** describe and provide electrical specifications for the PSoC 4 device families
- **CapSense Design Guide:** Learn how to design capacitive touch-sensing applications with the PSoC 4 family of devices.
- **Application Notes and Code Examples** cover a broad range of topics, from basic to advanced level. Many of the application notes include code examples. PSoC Creator provides additional code examples – see [Code Examples](#).
- **Technical Reference Manuals (TRM)** provide detailed descriptions of the architecture and registers in each PSoC 4 device family.
- **Development Kits:**
 - [CY8CKIT-040](#), [CY8CKIT-042](#), [CY8CKIT-044](#), and [CY8CKIT-046](#) are PSoC 4 Pioneer Kits are easy-to-use and inexpensive development platforms. These kits include connectors for Arduino™ compatible shields and Digilent® Pmod™ daughter cards.
 - [CY8CKIT-049](#), and [CY8CKIT-043](#) are very low-cost prototyping platforms for sampling PSoC 4 devices.
 - [CY8CKIT-001](#) is a common development platform for all PSoC family devices.
- The [MiniProg3](#) device provides an interface for flash programming and debug.

2.1 PSoC Creator

[PSoC Creator](#) is a free Windows-based Integrated Design Environment (IDE). It enables concurrent hardware and firmware design of systems based on PSoC 3, PSoC 4, and PSoC 5LP. See [Figure 1](#) – with PSoC Creator, you can:

1. Drag and drop Components to build your hardware system design in the main design workspace
2. Codesign your application firmware with the PSoC hardware
3. Configure Components using configuration tools
4. Explore the library of 100+ Components
5. Review Component datasheets

Figure 1. PSoC Creator Features



2.2 Code Examples

PSoC Creator includes a large number of code example projects. These projects are available from the PSoC Creator Start Page, as Figure 2 shows.

Example projects can speed up your design process by starting you off with a complete design, instead of a blank page. The example projects also show how PSoC Creator Components can be used for various applications. Code examples and datasheets are included, as Figure 3 shows.

In the Find Example Project dialog shown in Figure 3, you have several options:

- Filter for examples based on architecture or device family, i.e., PSoC 3, PSoC 4 or PSoC 5LP; category; or keyword.
- Select from the menu of examples offered based on the Filter Options.
- Review the datasheet for the selection (on the Documentation 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 2. Code Examples in PSoC Creator

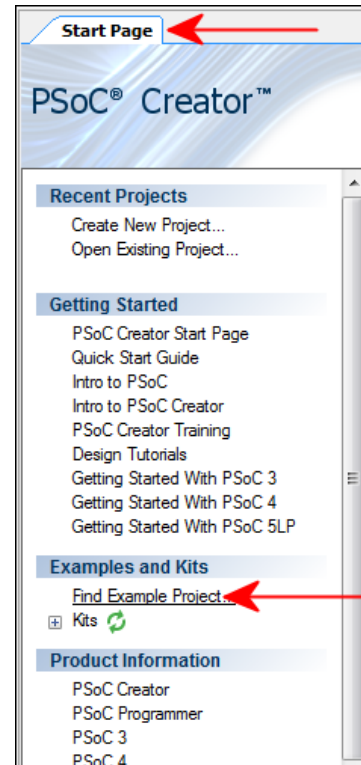
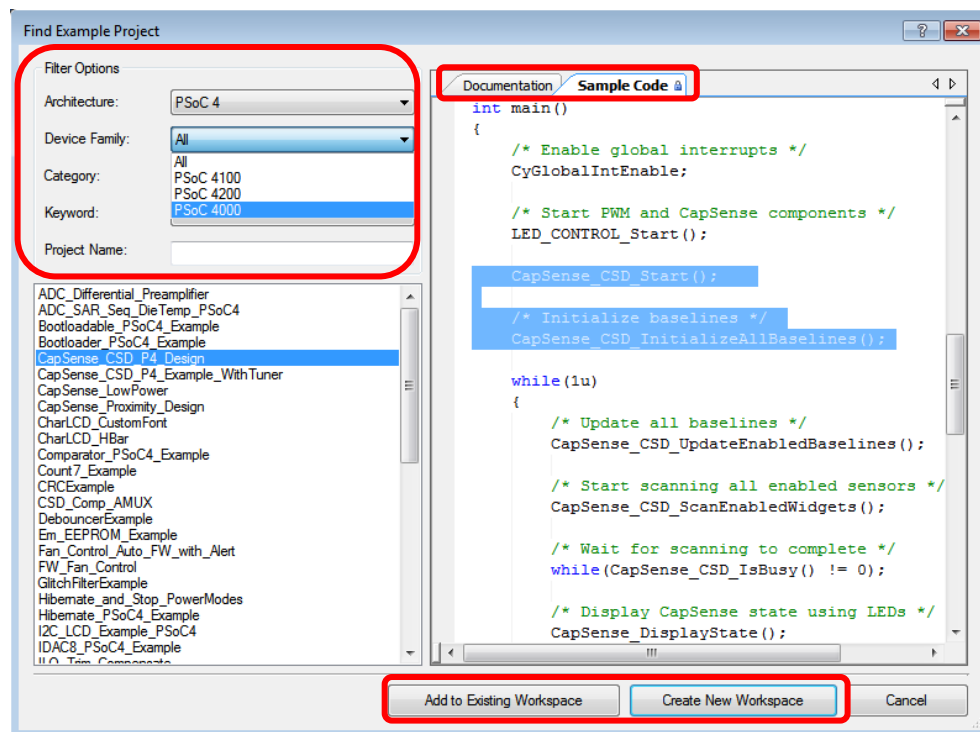


Figure 3. Code Example Projects, with Sample Code



2.3 PSoC Creator Help

Visit the [PSoC Creator home page](#) to download the latest version of PSoC Creator. Then, launch PSoC Creator and navigate to the following items:

- **Quick Start Guide:** Choose **Help > Documentation > Quick Start Guide**. This guide gives you the basics for developing PSoC Creator projects.
- **Simple Component example projects:** Choose **File > Open > Example projects**. These example projects demonstrate how to configure and use PSoC Creator Components.
- **Starter designs:** Choose **File > New > Project > PSoC 4 Starter Designs**. These starter designs demonstrate the unique features of PSoC 4.
- **System Reference Guide:** Choose **Help > System Reference > System Reference Guide**. This guide lists and describes the system functions provided by PSoC Creator.
- **Component datasheets:** Right-click a Component and select “Open Datasheet.” Visit the [PSoC 4 Component Datasheets](#) page for a list of all PSoC 4 Component datasheets.
- **Document Manager:** PSoC Creator provides a document manager to help you to easily find and review document resources. To open the document manager, choose the menu item **Help > Document Manager**.

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](#)

3 Features of Segment LCD Drive in PSoC 4

PSoC 4's segment LCD drive has the following features:

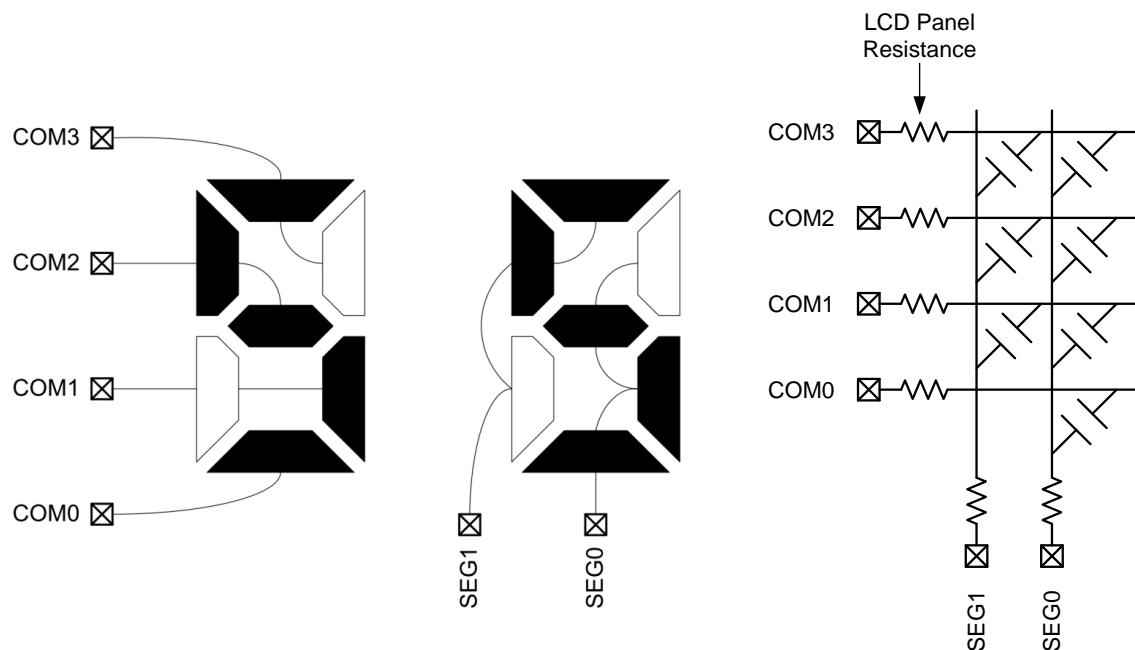
- Supports up to 8 common (COM) and 56 segment (SEG) electrodes
- Programmable PSoC 4 GPIOs provide flexible selection of COM and SEG electrodes.
- Supports 14-segment and 16-segment alphanumeric display, 7-segment numeric display, dot matrix, and special symbols
- Two drive modes: digital correlation and PWM
- Operates in PSoC 4's Active, Sleep, and Deep-Sleep system power modes.
- Can drive a 3-volt display from 1.8-volt V_{DD}
- Digital contrast control

Note: The number of commons and segments supported by a PSoC 4 device varies based on the device family and device package. See the respective device datasheets for details.

4 Basics of Segment LCD

A segment LCD panel has liquid crystal material between two sets of electrodes. The top and bottom electrodes of an individual LCD segment are called the common (COM) and the segment (SEG) electrodes, respectively. These electrodes form a matrix as Figure 4 shows.

Figure 4. COM and SEG Electrodes



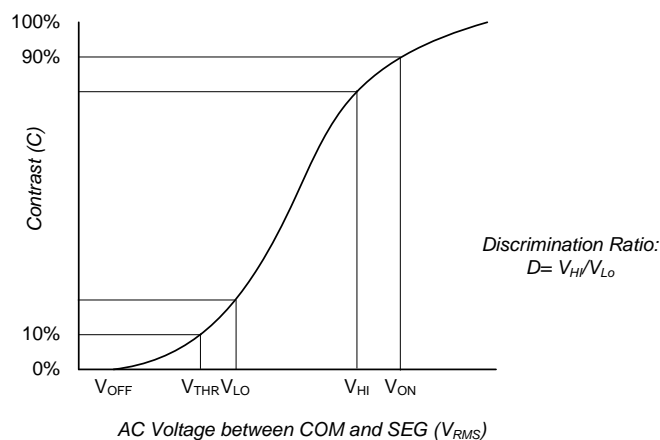
The lumped capacitors in Figure 4 represent the capacitive load of the ideal LCD segments. Each pair of COM and SEG electrodes is connected by only one LCD segment. The root-mean-square (RMS) voltage across a COM-SEG electrode pair controls the opacity of the LCD segment connected between the electrode pair.

Liquid crystal materials do not tolerate long-term exposure to DC voltage; therefore, waveforms applied between COM and SEG electrodes must produce a 0 V DC component on every LCD segment. The drive waveforms also contain time division multiplexing to simultaneously activate multiple LCD segments. A typical LCD driver generates AC waveforms with multiple voltage steps on the COM and SEG electrodes. The following terms are used to define these waveforms:

- **Duty:** A driver is said to operate in $1/M^{\text{th}}$ duty when it drives 'M' number of COM electrodes. Each COM electrode is effectively driven $1/M^{\text{th}}$ of the time. PSoC 4100 and PSoC 4200 devices support $1/2^{\text{nd}}$, $1/3^{\text{rd}}$, and $1/4^{\text{th}}$ duties. PSoC 4100M, 4200M, 4100L and 4200L devices support $1/2^{\text{nd}}$, $1/3^{\text{rd}}$, $1/4^{\text{th}}$ and $1/8^{\text{th}}$ duties.
- **Bias:** A driver is said to use $1/B^{\text{th}}$ bias when its waveforms use voltage steps of $(1/B) \times V_{\text{DRV}}$. V_{DRV} is the highest drive voltage in the system (equal to V_{DD} in PSoC 4). PSoC 4100 and PSoC 4200 devices supports $1/2^{\text{nd}}$ and $1/3^{\text{rd}}$ biases in PWM drive modes. PSoC 4100M, 4200M, 4100L, and 4200L devices support $1/2^{\text{nd}}$, $1/3^{\text{rd}}$, $1/4^{\text{th}}$, and $1/5^{\text{th}}$ duties in PWM drive modes. For more information, see [Table 1](#) on page 10.
- **Frame:** A frame is the length of time required to drive all the segments. During a frame, the driver cycles through the COM electrodes in sequence. All segments receive 0 V DC (but non-zero RMS voltage) when measured over the entire frame.

The intensity, or contrast, of a segment LCD pixel depends on the RMS voltage applied across the pixel. [Figure 5](#) shows the relationship between the contrast and the RMS voltage on an LCD segment.

Figure 5. Relationship Between RMS Voltage and Contrast



The terms in [Figure 5](#) are defined as follows:

- V_{OFF} : The RMS voltage across the pixel below which no contrast change is visible on the LCD
- V_{THR} : The RMS voltage across the pixel at which 10 percent contrast is achieved
- V_{ON} : The RMS voltage across the pixel at which 90 percent contrast is achieved
- V_{LO} : The lowest RMS voltage which the LCD driver can generate
- V_{HI} : The highest RMS voltage which the LCD driver can generate
- **Discrimination Ratio (D):** The ratio between V_{HI} and V_{LO} that the driver can realize. A higher discrimination ratio results in better contrast.

Twisted Nematic (TN) and Super Twisted Nematic (STN) displays are the most commonly available types of segment LCDs. TN displays require a higher change in RMS voltage to produce a noticeable difference in opacity, compared to the STN displays. Therefore, for a given discrimination ratio, STN displays provide better contrast than TN displays. STN displays also have superior viewing angles. However, TN displays are usually cheaper than STN displays.

For more information on the basics of segment LCD, refer to the [PSoC 4 Technical Reference Manual](#).

5 PSoC Creator Component: Segment LCD

PSoC Creator provides a Segment LCD Component which allows you to easily interface different types of segment LCD display modules with PSoC 4. Simply drag and drop the Component onto your PSoC Creator project schematic and configure it for your LCD module. The Component also provides an easy-to-use application programming interface (API). For more information, see the [PSoC 4 Segment LCD Component datasheet](#).

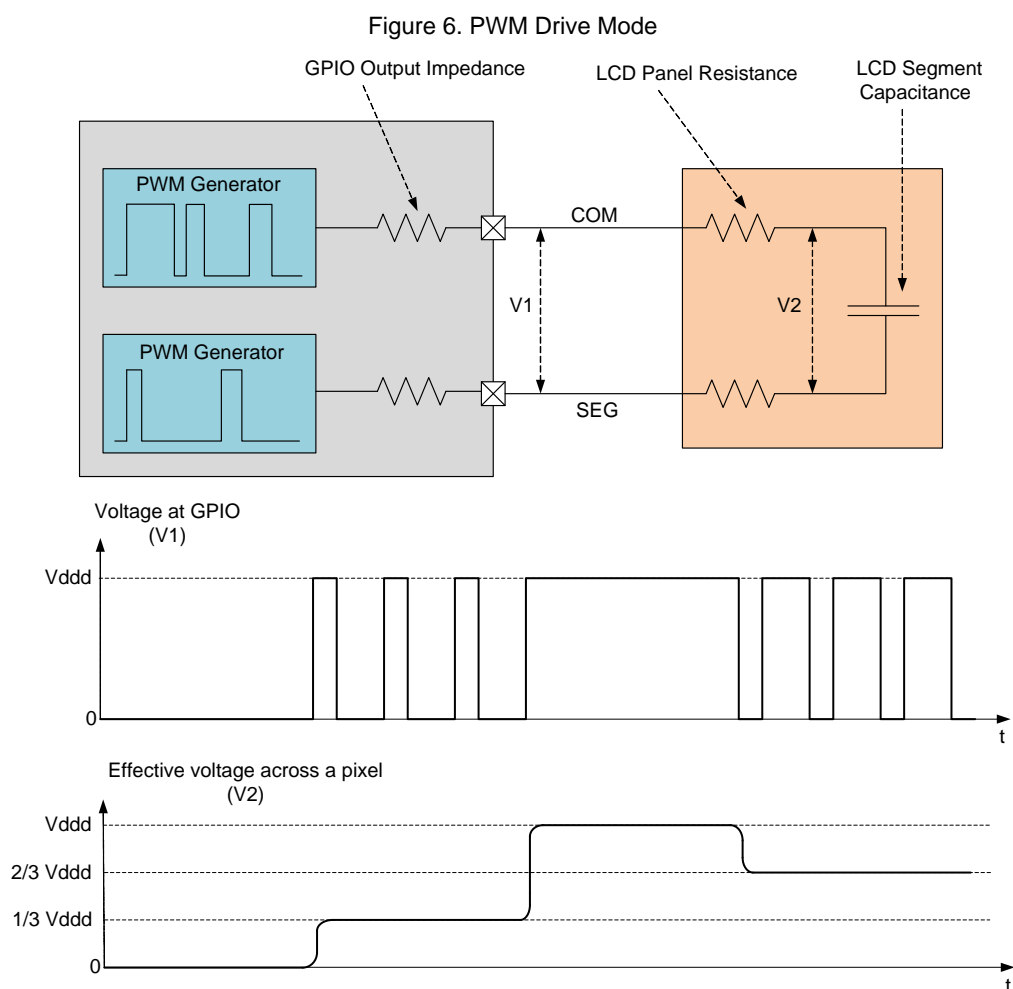
Also see the project “[Simple Segment LCD](#)”. This project shows you how to interface a 4-digit 7-segment LCD module with PSoC 4 by using the Segment LCD Component and APIs.

6 PSoC 4 Segment LCD Direct Drive

PSoC 4 supports two segment LCD drive modes, PWM and digital correlation. They use different techniques to generate the AC waveform required for the LCD glass, and they have different contrast and power consumption characteristics. The following sections explain in detail the differences between the two drive modes.

6.1 PWM Drive Mode

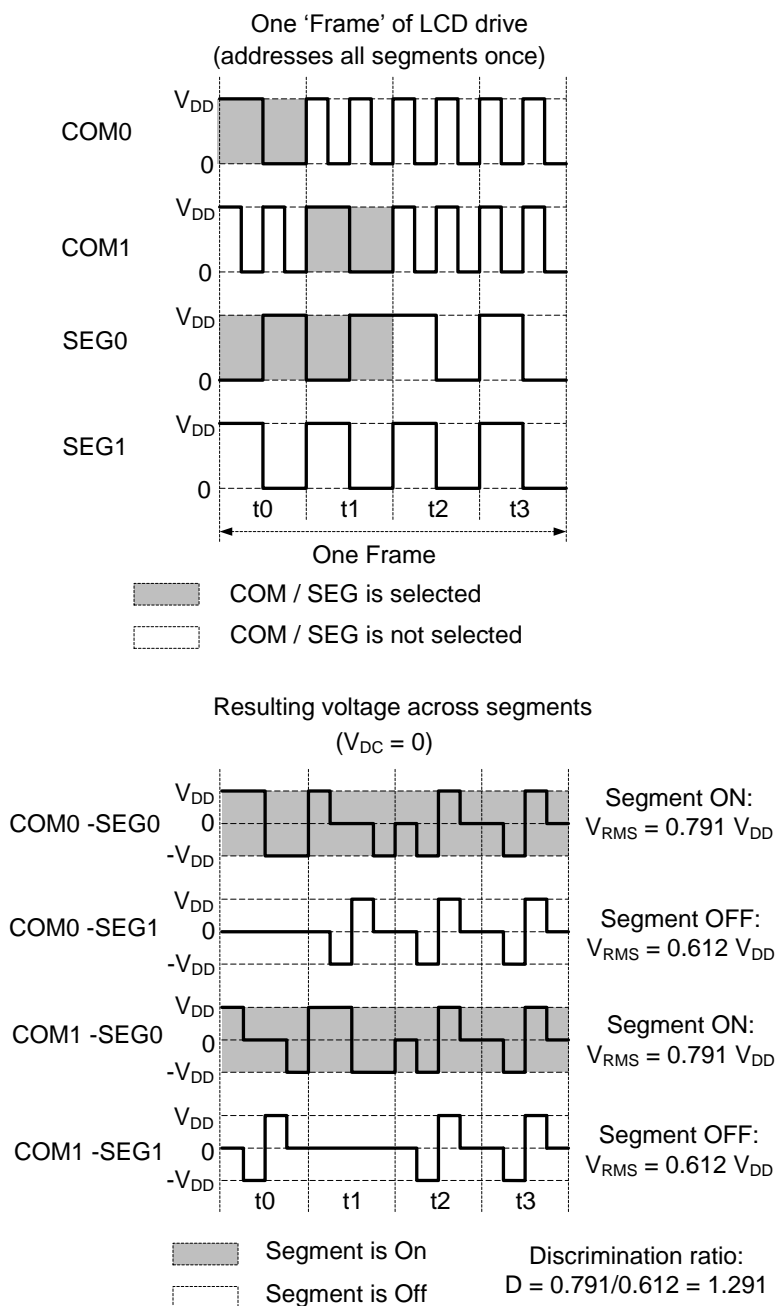
The PWM drive mode uses high-frequency (32 kHz to 48 MHz) PWM digital signals to generate the low-frequency (30 to 150 Hz) analog drive waveforms across the LCD segments. The capacitive nature of the LCD segments and the panel resistance filters the high-frequency PWM signals to generate variable voltage levels across the LCD segments, as [Figure 6](#) shows.



6.2 Digital Correlation Drive Mode

The digital correlation drive mode is a patent-pending Cypress invention that is ideal for low-power and battery-operated applications. It works by doubling the drive frequency of the COM signals in their inactive frame intervals. This changes the V_{RMS} across the segments, which in turn affects the segment's contrast (Figure 5). Figure 7 shows an example of digital correlation mode waveforms.

Figure 7. Digital Correlation Example



See the [PSoC 4 Technical Reference Manual](#) for an in-depth explanation of each drive mode and its corresponding waveform.

The segment LCD glass has the characteristics of an RC series circuit; therefore, its power consumption depends on the frequency of operation. Because of the very low frequency of operation, the digital correlation drive mode consumes considerably less current than the PWM drive mode.

However, the contrast and viewing angle of the PWM mode is superior to that of digital correlation mode. This is due to the PWM mode's higher discrimination ratio (Figure 5). Table 1 shows a comparison of discrimination ratios.

Table 1. Comparison of Discrimination Ratios

COM	Discrimination Ratio				
	Digital Correlation	PWM 1/2 Bias	PWM 1/3 Bias	PWM 1/4 Bias	PWM 1/5 Bias
2	1.732	2.236	2.236	1.844	1.612
3	1.414	1.732	1.915	1.732	1.567
4	1.291	1.528	1.732	1.648	1.528
8	1.134	1.254	1.414	1.581	1.494

The actual difference in contrast and viewing angle between these drive modes depends on the characteristics of the LCD module.

6.3 Low-Power Operation

Follow these steps to reduce power consumption while using PSoC 4 segment LCD drive:

1. See [AN86233, PSoC 4 Low-Power Modes and Power Reduction Techniques](#) and implement the techniques explained in it to reduce the overall system power consumption.
2. Enter Deep-Sleep mode whenever possible. Deep-Sleep power mode is PSoC 4's lowest-power mode that retains the segment LCD drive capability.
3. Use the [digital correlation drive mode](#) of the segment LCD when the device is in Deep-Sleep mode.

Compared to the PWM drive mode, the digital correlation mode produces reduced but acceptable contrast and viewing angles on TN displays, and no noticeable difference in contrast or viewing angle on STN displays. However, the digital correlation mode takes significantly less current than the PWM mode. Table 2 compares the current consumption of the two drive modes and Table 3 shows the recommended usage of the drive modes.

Note: Table 2 shows the approximate current consumption for a 3 COM, 12 SEG LCD with a PSoC 4200 device. The actual power consumption depends on the device family, type of LCD used, and the firmware.

Table 2. Typical Device Current Consumption in Deep-Sleep Mode with Segment LCD Enabled

Frame Rate, Hz	Current Consumption, μ A	
	Digital Correlation Mode	PWM Mode
30	3.1	65.9
50	3.8	66.2
70	4.5	66.5
90	5.1	66.7
110	5.8	67.1
130	6.5	67.3
150	7.4	67.6

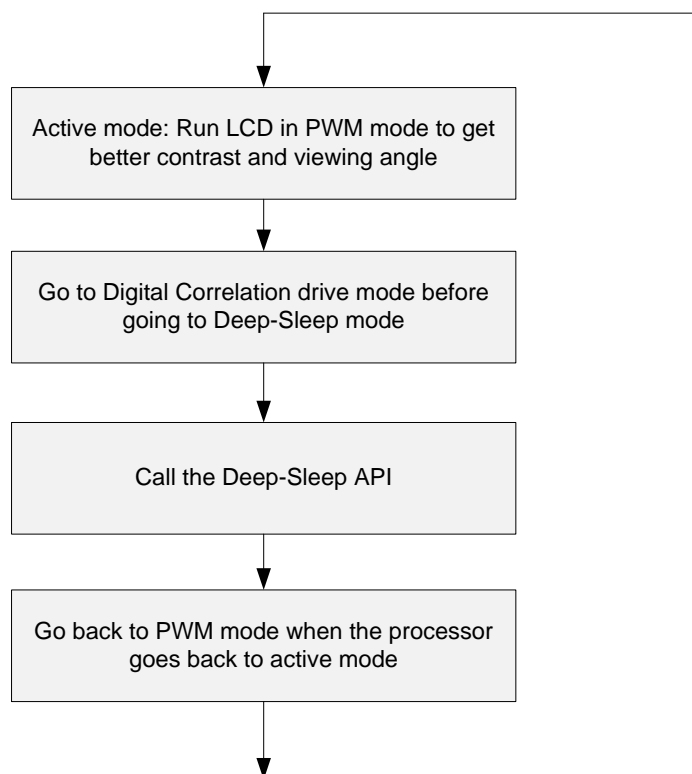
Table 3. Recommended Usage of Drive Modes for Low-Power Operation

Display Type	Deep-Sleep Mode	Active and Sleep Modes	Notes
TN LCD	Digital Correlation	PWM	Firmware must dynamically switch between LCD drive modes
STN LCD	Digital Correlation		No contrast advantage for PWM drive with STN LCD

You can change the LCD drive modes on the fly to give the optimum balance between power consumption and display contrast. [Figure 8](#) shows the firmware flow for dynamically switching LCD drive modes, for good contrast and viewing angle in the Active mode, and reduced power consumption in the Deep-Sleep mode.

See the “[Low-Power Segment LCD](#)” project for more details.

Figure 8. Firmware Flow of Low-Power LCD Operation



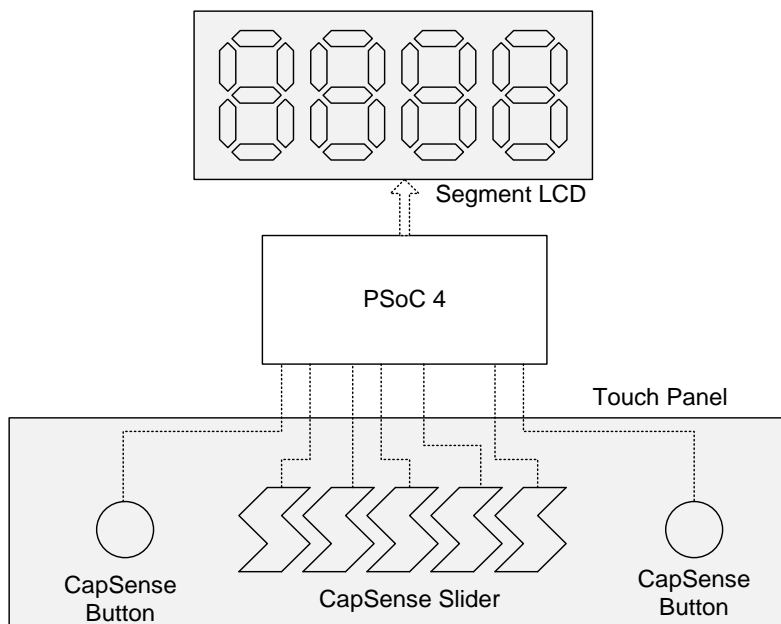
7 PSoC 4 User Interface Solution

This section shows how to create complete user interface (UI) solutions using just PSoC 4. A basic knowledge of PSoC 4 CapSense is a prerequisite for this section. See the [PSoC 4 CapSense Design Guide](#) to understand the basics of CapSense.

The CapSense feature in PSoC 4 offers unprecedented signal-to-noise ratio (SNR), best in class waterproofing, and a wide variety of sensors such as buttons, sliders, track pads, and proximity sensors. You can use the segment LCD drive together with CapSense and low-power modes to create elegant, reliable, power-efficient, and easy-to-use UI solutions.

[Figure 9](#) shows an example UI in which PSoC drives a segment LCD and scans a touch panel. The touch panel consists of two CapSense buttons and a slider.

Figure 9. Example of PSoC 4 User Interface



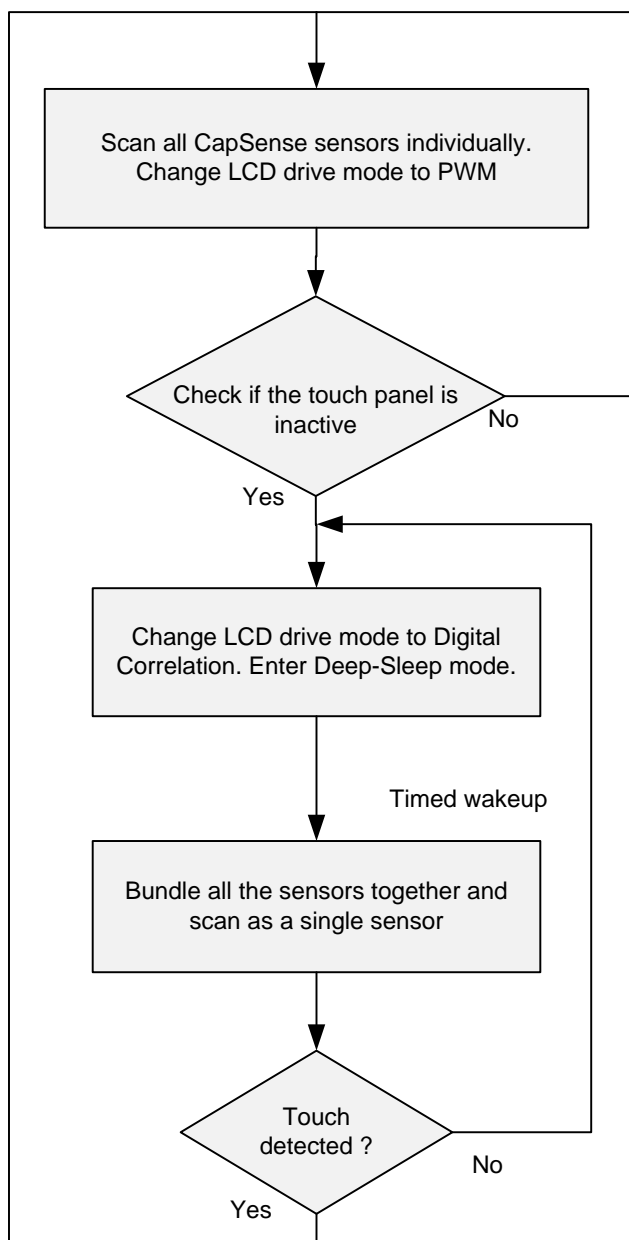
CapSense cannot scan sensors in Deep-Sleep mode, so the device must wake up periodically to scan the sensors. PSoC 4 must remain in the Active mode while scanning the sensors. To save power, the device should spend more time in Deep-Sleep mode and less in the Active mode.

If CapSense detects a touch, PSoC 4 stays in the Active mode and scans each sensor individually to read the touch panel input. The device returns to Deep-Sleep mode when the panel becomes inactive again.

PSoC 4 CapSense can bundle several sensors together and scan as a single sensor. This reduces the total time required in Active mode.

Figure 10 shows an example firmware flow of a low-power user interface (UI) solution.

Figure 10. Firmware Flow of a Low-Power UI Solution



See the [“Segment LCD and CapSense User Interface”](#) project for more details. This project demonstrates a user interface with a 4-digit, 7-segment LCD, and a 5-element CapSense slider.

8 PSoC Creator Projects

This application note provides four example projects:

1. **Simple Segment LCD:** This is a simple project to demonstrate basic usage of the PSoC Creator PSoC 4 Segment LCD Component.
2. **Low-Power Segment LCD:** This project shows how to use the Segment LCD Component with very low power consumption.
3. **Segment LCD and CapSense User Interface:** This project shows how to create a complete user interface solution using the Segment LCD and CapSense Components.

You can use any PSoC 4 kit or your own hardware to evaluate projects 1 and 2. Project 3 requires the Cypress [CY8CKIT-042](#).

All three projects require a [VIM 404 segment LCD module](#), which has four 7-segment digits, three COM electrodes, and 12 SEG electrodes. See [Figure 17](#) on page 19 for a wiring schematic.

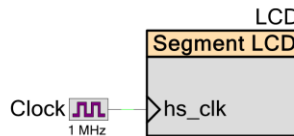
4. **Alphanumeric Segment LCD:** This project takes advantages of the large number of GPIOs available in the PSoC 4100M / 4200M devices to drive a [VIM-828 segment LCD module](#), which has eight 14-segment digits, four COM electrodes, and 32 SEG electrodes. This project requires Cypress [CY8CKIT-043](#) or [CY8CKIT-046](#) and a [VIM-828 Segment LCD module](#).

8.1 Project 1: Simple Segment LCD

To create this project, do the following:

1. Create a new PSoC 4 project. If you are new to PSoC Creator, see the [PSoC Creator home page](#).
2. Drag and drop a Segment LCD Component and a Clock Component into the TopDesign schematic. Connect them together using a wire, as [Figure 11](#) shows.

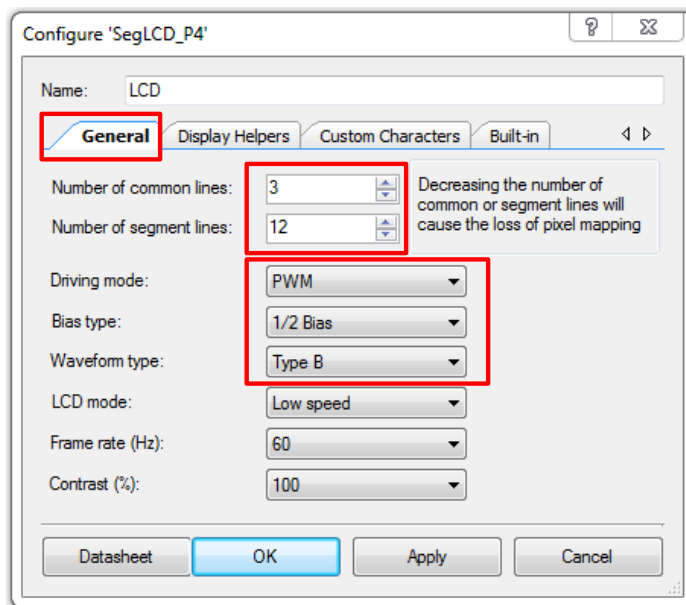
Figure 11. Connection of LCD and Clock



3. Double-click the Segment LCD Component to open the configuration window.

4. Configure the **General** tab in the Component configuration dialog. Choose the LCD drive settings as [Figure 12](#) shows.

Figure 12. Segment LCD Configuration, General Tab



- Set the number of common and segment lines to 3 and 12 respectively, to match the VIM 404 segment LCD module.
- Use PWM drive mode; see [PSoC 4 Segment LCD Direct Drive](#) on page 8.
- Select 1/2 bias because it provides good contrast on the VIM 404 segment LCD module.
- Choose Type B waveform. Type B waveform consumes slightly less power than the type A waveform. For more information on these waveforms, see the [PSoC 4 Technical Reference Manual](#).
- Leave the remaining settings at their default values.

- Configure the **Display Helpers** tab. This tab maps SEG and COM electrodes to the display module. Use the LCD module datasheet to configure this tab (see [VIM-404](#)).


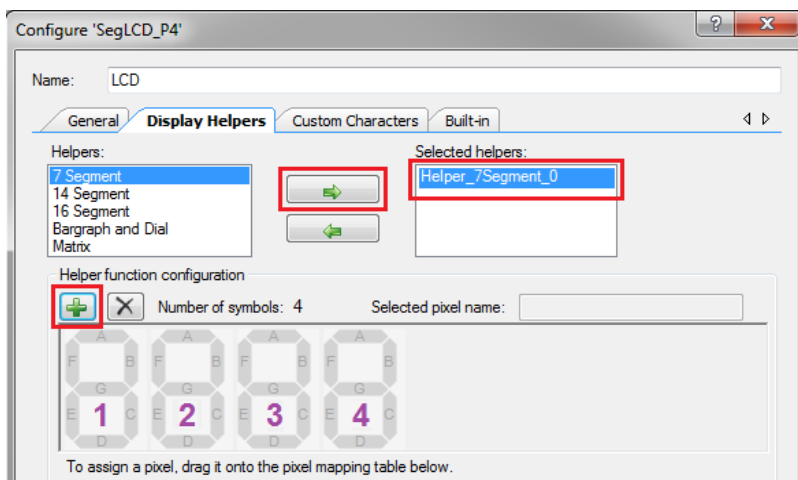

The LCD module has four 7-segment digits. To add 7-segment display helpers, select **7 Segment** in the **Helpers** list box, and click the  button. The selected display feature is visible in the **Selected helpers** list box, as [Figure 13](#) shows.

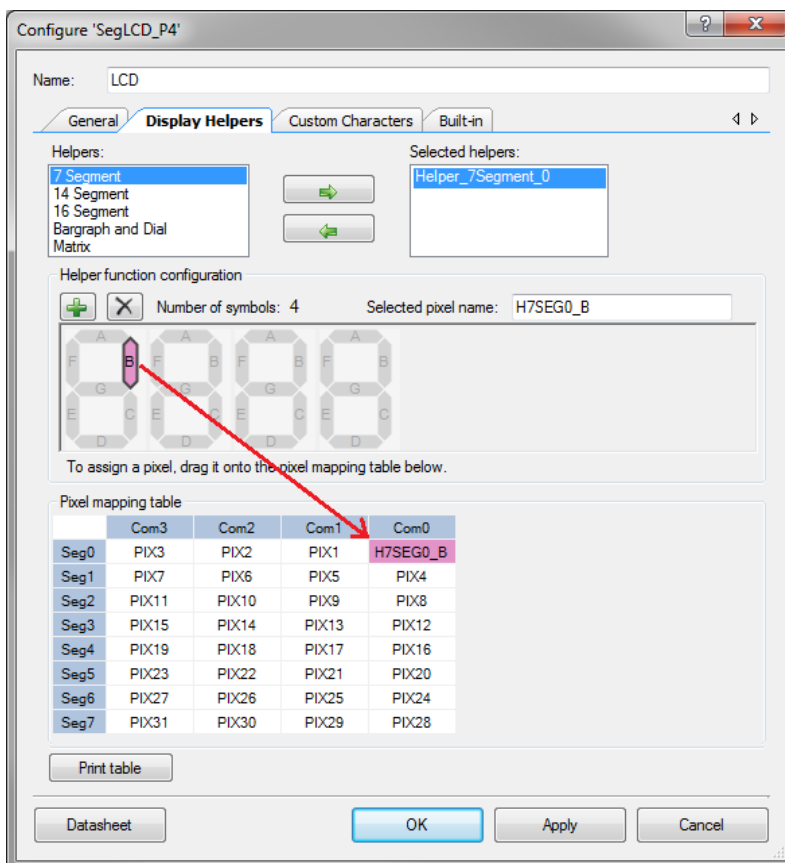
Figure 13. Adding Display Helpers



The LCD module has four 7-segment digits, so add three more 7-segment digits by clicking the  button.

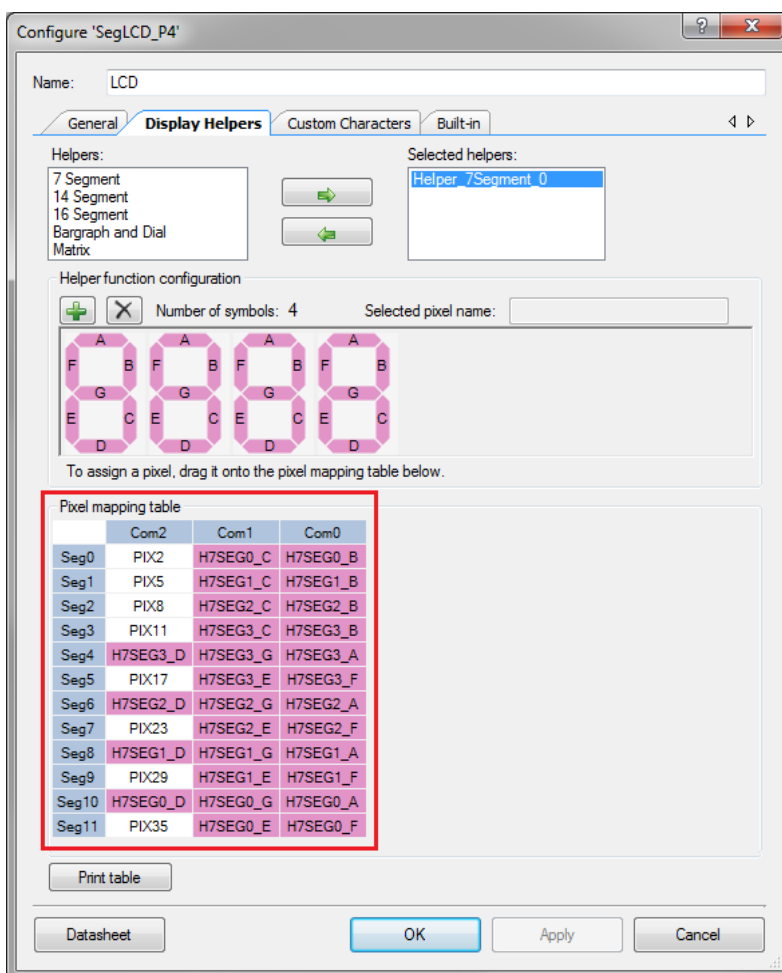
6. Drag a segment from a digit and drop it onto the pixel mapping table, as [Figure 14](#) shows. See [VIM-404](#) datasheet for the pixel mapping of the VIM 404 LCD module.

Figure 14. Pixel Mapping Procedure



7. Repeat the process for each segment in the display. After mapping all of the pixels, the completed table appears as shown in Figure 15.

Figure 15. Completed Pixel Mapping Table



- Click **OK**; the Segment LCD Component configuration is now complete. To assign the pins for the commons and segments, open the .cydwr (design wide resources) file and assign pins as [Figure 16](#) shows.

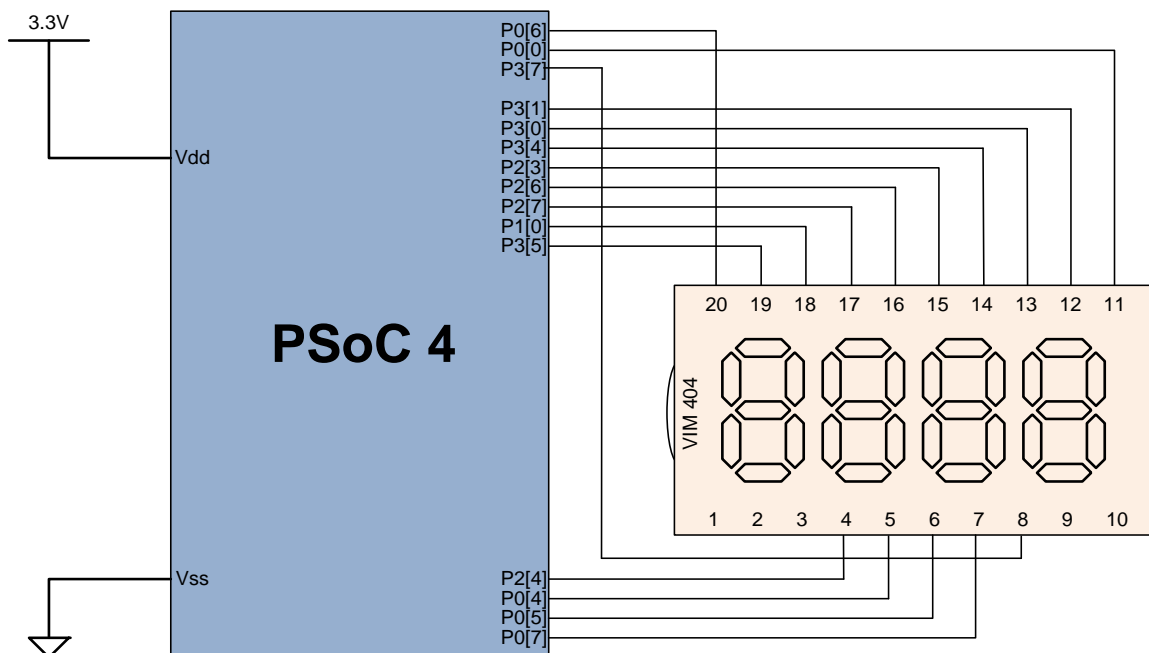
Note: The pin assignment shown here is for the CY8CKIT-042 development kit. If you're using a different kit, refer to the kit user guide and select the suitable pins that aren't connected to any other peripherals on the kit.

Figure 16. Pin Assignment

Name /	Port
\LCD:Com[0]\	P0[0] SCB0:SPI:SS1
\LCD:Com[1]\	P0[6] SCB1:SPI:SCLK, EXTCLK
\LCD:Com[2]\	P3[7] TCPWM3:N
\LCD:Seg[0]\	P2[4] TCPWM0:P
\LCD:Seg[1]\	P0[4] SCB1:I2C:SCL, SCB1:SPI:MOSI, SCB1:UART:RX
\LCD:Seg[2]\	P0[5] SCB1:I2C:SDA, SCB1:SPI:MISO, SCB1:UART:TX
\LCD:Seg[3]\	P0[7] SCB1:SPI:SS0, WAKEUP
\LCD:Seg[4]\	P3[1] SCB1:I2C:SDA, SCB1:SPI:MISO, TCPWM0:N, SCB1:UART:TX
\LCD:Seg[5]\	P3[0] SCB1:I2C:SCL, SCB1:SPI:MOSI, TCPWM0:P, SCB1:UART:RX
\LCD:Seg[6]\	P3[4] SCB1:SPI:SS1, TCPWM2:P
\LCD:Seg[7]\	P2[3]
\LCD:Seg[8]\	P2[6] TCPWM1:P
\LCD:Seg[9]\	P2[7] TCPWM1:N
\LCD:Seg[10]\	P1[0] TCPWM2:P
\LCD:Seg[11]\	P3[5] SCB1:SPI:SS2, TCPWM2:N

- Wire the VIM 404 LCD glass to PSoC 4, as [Figure 17](#) shows. This can be done on your own PCB, or you can use the [CY8CKIT-042](#) kit and an Arduino ProtoShield board.

Figure 17. PSoC 4 to LCD Module External Connections



10. Add firmware to the project. To create a scrolling display of numbers 1, 2, 3, ..., 9, 0, add the code in [Code 1](#) to *main.c*.

The Segment LCD Component offers a ready to use API. For details on the API functions, see the [PSoC 4 Segment LCD Component datasheet](#).

Code 1. Scrolling Numbers on the LCD

```
#include <device.h>

/* Macro for a black space on the 7-segment LCD */
#define BLANK (16u)

/* Macro for scrolling length */
#define SCROLL_LENGTH (14u)

/* Macro for initial scrolling value */
#define INITIAL_VALUE (0u)

/* Define the numbers used for scrolling the 4-digit display */
#define SCROLL0 (0u)
#define SCROLL1 (1u)
#define SCROLL2 (2u)
#define SCROLL3 (3u)
#define DIGIT0 (0u)
#define DIGIT1 (1u)
#define DIGIT2 (2u)
#define DIGIT3 (3u)

/* Define the loop delay */
#define LOOP_DELAY (512u)

void main()
{
    /* This variable is used to index the character array */
    uint8 index = 0;

    /* Array of scrolling numbers including blank spaces to improve visibility */
    uint8 numbers[] = {BLANK, BLANK, BLANK, BLANK,
                      1, 2, 3, 4, 5, 6, 7, 8, 9, 0, BLANK, BLANK, BLANK};

    /* Starts the Segment LCD Component */
    LCD_Start();

    for(;;) /* do forever */
    {
        /* Scroll the blanks and numbers stored in the array, from right to left */
        for(index = INITIAL_VALUE; index < SCROLL_LENGTH; index++)
        {
            /* Print the numbers on the LCD glass */
            LCD_Write7SegDigit_0(numbers[index + SCROLL3], DIGIT0);
            LCD_Write7SegDigit_0(numbers[index + SCROLL2], DIGIT1);
            LCD_Write7SegDigit_0(numbers[index + SCROLL1], DIGIT2);
            LCD_Write7SegDigit_0(numbers[index + SCROLL0], DIGIT3);

            /* Give delay for a visible scrolling */
            CyDelay(LOOP_DELAY);
        }
    }
}
```

11. Make sure that PSoC 4 is operating from 3.3 V V_{DD} supply. Program the device and view the LCD display to verify the output. You can see numbers 1, 2, 3, ..., 9, 0, scrolling from right to left as [Figure 18](#) shows.

Figure 18. LCD Display Module Scrolling Numbers



If the contrast of the display is too high, reduce the contrast in the Segment LCD Component and program the device again.

8.2 Project 2: Low-Power Segment LCD

This project is a modification of [Project 1](#), for lower average power.

To reduce power consumption, PSoC 4 stays in Active mode just long enough to update the display and scroll the numbers one place, which takes approximately 720 μ s. The device then goes into the Deep-Sleep power mode for the next 512 ms. [Figure 19](#) shows the timing diagram and [Figure 20](#) shows the firmware flow.

With this technique, PSoC 4 consumes an average current less than 11 μ A. [Table 4](#) shows a comparison of approximate power consumption between the two projects.

Table 4. Power Comparison Between Projects

Power Mode	Project 1	This Project
Active current	2.7 mA	2.7 mA
Deep-Sleep current	n/a	7 μ A
Average current	2.7 mA	10.9 μ A

Figure 19. Project 2 Timing Diagram

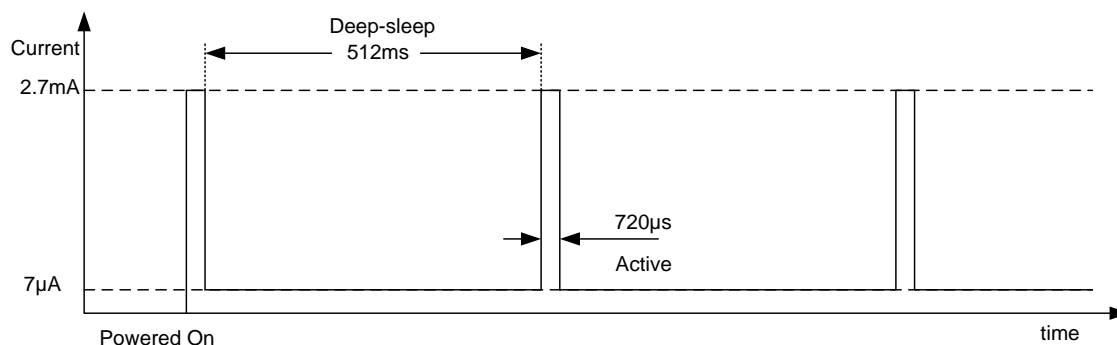
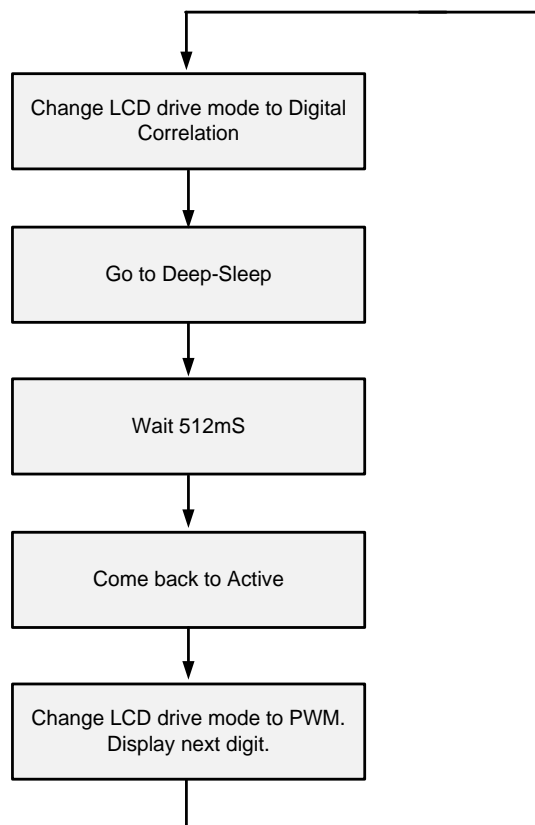


Figure 20. Project 2 Firmware Flow



This project uses the Global Signal Component to generate a watchdog timer (WDT) interrupt. Configure the Component as [Figure 21](#) shows. The WDT is configured to generate an interrupt every 512 ms. [Figure 22](#) shows the TopDesign schematic of this project.

Figure 21. Global Signal Component Configuration

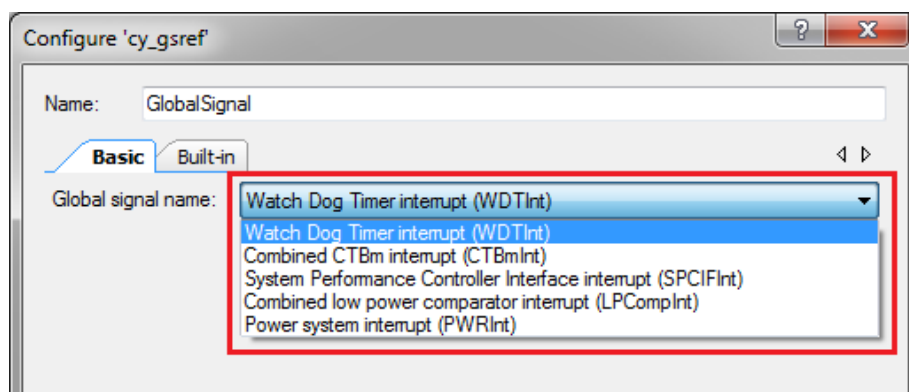
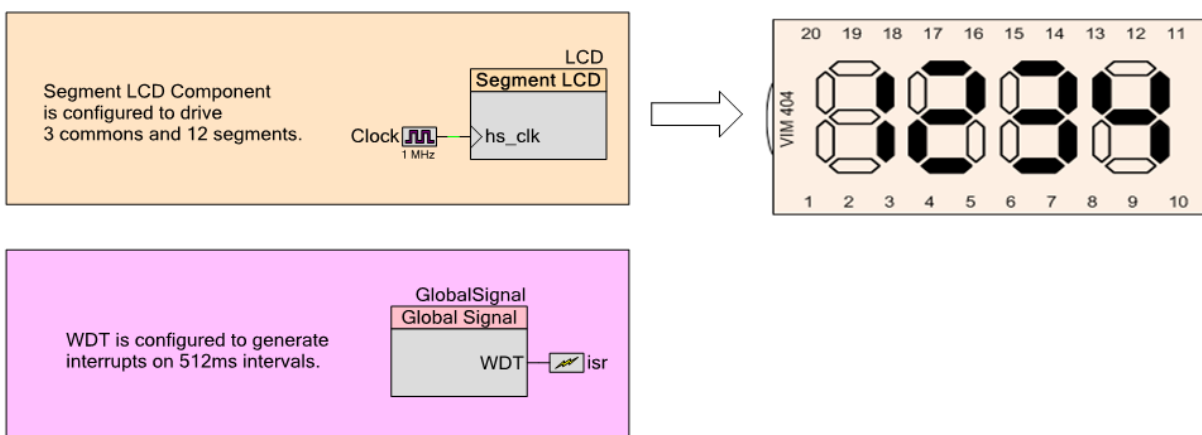


Figure 22. Project 2 LCD TopDesign Schematic



To test the project, program the device, connect the display as in [Project 1 \(Figure 17\)](#) and view the output. You will still see a scrolling display; however, power consumption for this project is significantly reduced.

8.3 Project 3: Segment LCD and CapSense User Interface

This project demonstrates a complete user interface (UI) consisting of CapSense touch sensing and a segment LCD for display. UI controllers spend most of their time, and hence power, waiting for user input. To implement a power efficient design, the PSoC 4 periodically enters Deep-Sleep low-power mode.

Note: This project uses manual tuning of CapSense widgets to achieve very low scan time and power consumption. Knowledge of CapSense manual tuning is a prerequisite for this project. See the [PSoC 4 CapSense Design Guide](#) to understand manual tuning.

This project is a modification of [Project 2](#) to add CapSense capability. [Figure 23](#) shows the timing diagram, [Figure 24](#) shows the TopDesign schematic for this project, and [Figure 25](#) shows the firmware flow.

In this project, PSoC 4 stays in the Deep-Sleep mode for only 128 ms, so that CapSense scans the sensors frequently enough to detect fast touches. Upon wakeup, CapSense bundles together all slider elements and scans them as a single proximity sensor. If any slider element is touched, the proximity sensor detects a touch and then each element is scanned individually to determine slider touch position. The slider position is then displayed on the LCD.

If all the CapSense widgets are inactive for 128 ms after a touch is detected, the device returns to Deep-Sleep mode. In the absence of touch, the average current consumed by PSoC 4 is less than 21 μ A.

Figure 23. Project 3 Timing Diagram

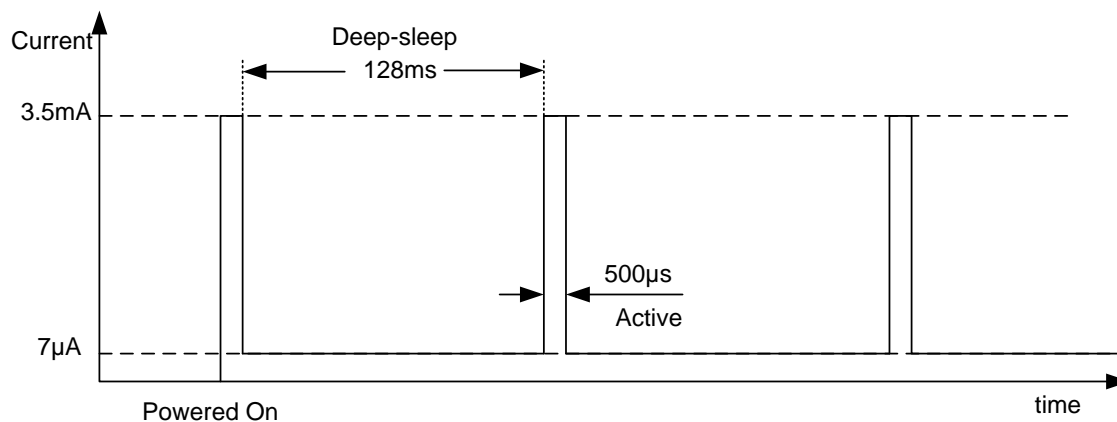


Figure 24. Project 3 TopDesign Schematic

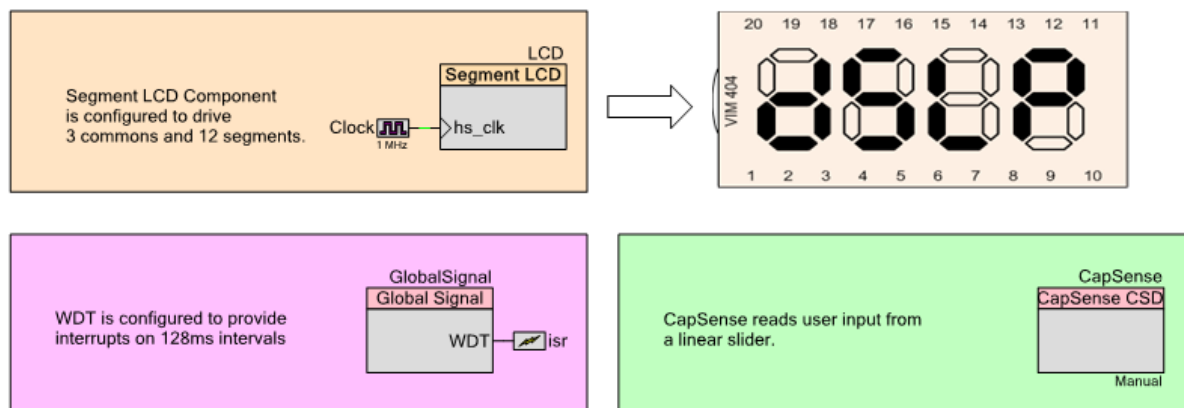
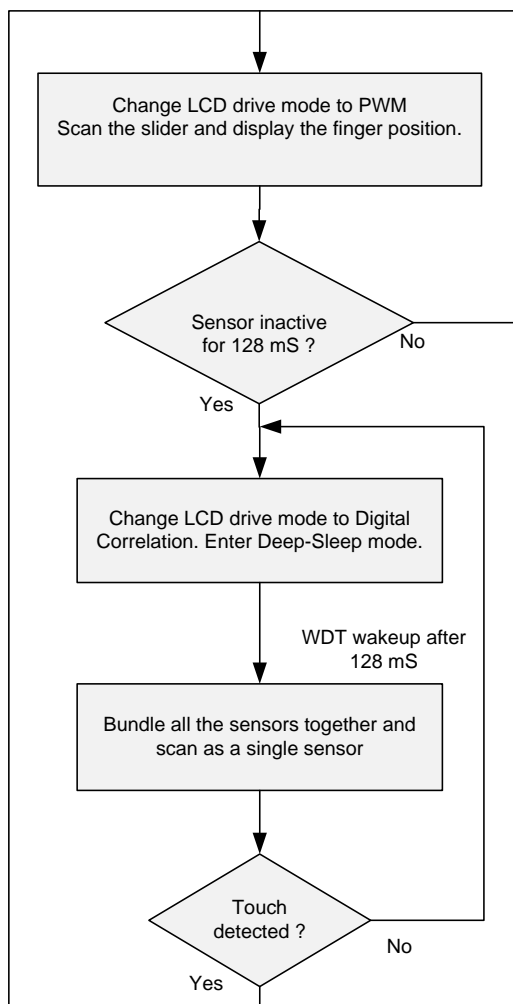


Figure 25. Project 3 Firmware Flow



To test the project, connect the LCD as [Figure 17](#) shows, touch the slider, and confirm that the LCD displays the slider position. When the slider is inactive, the LCD displays "dSLP" to indicate the Deep-Sleep mode of the device. Note that the default CapSense parameters are manually tuned for [CY8CKIT-042](#) under a given set of conditions. You should manually tune the CapSense again if the touch response is poor. The other alternative is to use SmartSense™, a firmware algorithm that automatically sets all parameters to optimum values. See the [PSoC 4 CapSense Design Guide](#) for details. Note that using SmartSense increases power consumption. The approximate average power consumption for this project with SmartSense is 40 μ A.

8.4 Project 4: Alphanumeric Segment LCD

This project takes advantage of the large number of GPIOs available in the PSoC 4100M/4200M/4100L/4200L devices to drive a [VIM-828 segment LCD module](#), which has eight 14-segment digits, four COM electrodes, and 32 SEG electrodes. This project requires Cypress [CY8CKIT-043](#) or [CY8CKIT-046](#) and a [VIM-828 LCD](#).

The segment LCD Component provides rich APIs to display strings on 14-segment alphanumeric displays. This project shows a static text “PSoC4200” displayed on VIM-828. See the PSoC Creator project for more details. [Table 5](#) gives the pin mapping information for wiring VIM-828 to CY8CKIT-43 and CY8CKIT-46.

Table 5. Wiring Information

LCD Pin Number	LCD Pin Function	CY8CKIT-43 Connection	CY8CKIT-046 Connection
1	COM3	P0[7]	P2[0]
2	SEG0	P0[6]	P2[1]
3	SEG1	P0[5]	P2[2]
4	SEG2	P0[4]	P2[3]
5	SEG3	P0[3]	P2[4]
6	SEG4	P0[2]	P2[5]
7	SEG5	P0[1]	P3[6]
8	SEG6	P0[0]	P3[7]
9	SEG7	P7[1]	P9[0]
10	SEG8	P4[0]	P9[1]
11	SEG9	P3[7]	P9[2]
12	SEG10	P3[6]	P9[3]
13	SEG11	P3[5]	P6[1]
14	SEG12	P3[4]	P6[0]
15	SEG13	P3[3]	P6[3]
16	SEG14	P3[2]	P4[0]
17	SEG15	P3[1]	P4[1]
18	COM2	P3[0]	P3[4]
19	COM0	P6[5]	P6[5]
20	SEG16	P6[4]	P5[5]
21	SEG17	P6[2]	P5[6]
22	SEG18	P6[1]	P4[4]
23	SEG19	P6[0]	P4[5]
24	SEG20	P2[7]	P4[6]
25	SEG21	P2[6]	P4[7]
26	SEG22	P2[5]	P3[1]
27	SEG23	P2[4]	P3[0]
28	SEG24	P1[3]	P8[7]
29	SEG25	P1[2]	P8[6]
30	SEG26	P1[1]	P8[5]
31	SEG27	P1[0]	P8[4]
32	SEG28	P5[5]	P8[3]

LCD Pin Number	LCD Pin Function	CY8CKIT-43 Connection	CY8CKIT-046 Connection
33	SEG29	P5[3]	P8[2]
34	SEG30	P5[2]	P8[1]
35	SEG31	P5[1]	P8[0]
36	COM1	P5[0]	P5[6]

9 Summary

This application note demonstrated the segment LCD drive in PSoC 4. The combination of robust capacitive touch sensing and best-in-class low-power capabilities make PSoC 4 an ideal choice for a low-power, single chip, user interface solution.

This application note also gives four example projects to get started with the PSoC 4 segment LCD, show low-power operation, and demonstrate a complete user interface.

Note that PSoC 4 has many additional features such as programmable analog and digital, 32-bit ARM Cortex-M0 microcontroller, and serial communication blocks that allow you to create an entire system solution around PSoC 4.

About the Authors

Name: Nidhin MS

Title: Applications Engineer Sr.

Background: Nidhin graduated from GEC Thrissur, with a Bachelor's degree in Electronics and Communication Engineering. His technical interests are analog signal processing, low-power design, and capacitive touch sensing.

Name: Rahul Raj Sharma

Title: Applications Engineer

Background: Rahul graduated from RNSIT Bangalore, with a BE in Electronics and Communication Engineering.

Document History

Document Title: AN87391 - PSoC® 4 Segment LCD Direct Drive

Document Number: 001-87391

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	4129400	NIDH	09/18/2013	New application note
*A	4602278	NIDH	12/19/2014	Updated the example project. Added mention of SmartSense for project 3 with power numbers.
*B	4797445	NIDH	06/24/2015	Updated Feature List , Basics of segment LCD, and Table -1 for PSoC 4A-M Added Project 4 Updated template Added PSoC Resources section
*C	5068119	NIDH	02/09/2016	Added PSoC 4 L series, CY8CKIT-046 information Added CY8CKIT-046 to pin mapping table Updated projects to PSoC Creator 3.3 CP1 Linked PSoC 4 low power AN in the introduction Added a note for Table-2 indicating that these are approximate power numbers that may vary with device family, LCD and firmware Added step-by-step instructions for Low-Power Operation to section 6.3
*D	5475276	NIDH	10/14/2016	Added links to code examples Added links to PSoC 4100 and PSoC 4200 Updated associated projects Updated template
*E	5700428	AESATP12	04/20/2017	Updated logo and copyright.

Worldwide Sales and Design Support

Cypress maintains a worldwide network of offices, solution centers, manufacturer's representatives, and distributors. To find the office closest to you, visit us at [Cypress Locations](#).

Products

ARM® Cortex® Microcontrollers	cypress.com/arm
Automotive	cypress.com/automotive
Clocks & Buffers	cypress.com/clocks
Interface	cypress.com/interface
Internet of Things	cypress.com/iot
Memory	cypress.com/memory
Microcontrollers	cypress.com/mcu
PSoC	cypress.com/psoc
Power Management ICs	cypress.com/pmic
Touch Sensing	cypress.com/touch
USB Controllers	cypress.com/usb
Wireless Connectivity	cypress.com/wireless

All other trademarks or registered trademarks referenced herein are the property of their respective owners.

PSoC® Solutions

[PSoC 1](#) | [PSoC 3](#) | [PSoC 4](#) | [PSoC 5LP](#) | [PSoC 6](#)

Cypress Developer Community

[Forums](#) | [WICED IOT Forums](#) | [Projects](#) | [Videos](#) | [Blogs](#) | [Training](#) | [Components](#)

Technical Support

cypress.com/support



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

© Cypress Semiconductor Corporation, 2013-2017. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC ("Cypress"). This document, including any software or firmware included or referenced in this document ("Software"), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress's patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage ("Unintended Uses"). A critical component is any component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.