Segment LCD drive using PSOC™ Automotive 4

About this document

Scope and purpose

This document demonstrates how easy it is to drive segment LCD glass using the integrated LCD driver in Automotive PSOC™ 4. Automotive 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™.

Intended audience

This application note is intended for engineers who want to use Seven segment LCD for their application and those with experience in working with Automotive PSOC™ 4 microcontroller and ModusToolbox™ IDE.

Table of contents

1 Introduction

Most low-power portable hand-held 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. Automotive PSOC™ 4 implements an integrated low-power LCD driver that can directly drive segment LCD glass.

Automotive 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. Automotive PSOC™ 4 accelerates time-to-market, integrates critical system functions, and reduces overall system cost. The capacitive touch sensing capability of Automotive PSOC™ 4 (CAPSENSE™), together with LCD drive allows you to implement an entire user interface solution using only PSOC™ Automotive 4. PSOC™ Automotive 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 PSOC™ is in its Deep Sleep low-power mode. The ModusToolbox™ IDE provides a Segment LCD Configurator that makes it easy to use different types of segment LCD modules with Automotive PSOC™ 4.

This application note explains the features offered by Automotive 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 Automotive PSOC™ 4, refer to AN79953, Getting Started with PSOC™ Automotive 4
- If you are new to ModusToolbox™, see the ModusToolbox™ home page
- If you are new to Automotive PSOC™ 4 power modes, refer to AN86233, Automotive PSOC™ 4 low-power modes and power reduction techniques

2 PSOC™ resources

Infineon provides a wealth of data at www.infineon.com to help you to select the right PSOC™ Automotive device for your design, and to help you to quickly and effectively integrate the device into your design. Here is an abbreviated list for Automotive PSOC™ 4:

1. **Overview:** PSOC™ Automotive portfolio, PSOC™ Automotive roadmap
2. **Product selectors:** PSOC™ 1, PSOC™ 3, Automotive PSOC™ 4, or PSOC™ 5LP. In Addition, ModusToolbox™ includes a device selection option
3. Datasheets describe and provide electrical specifications for the Automotive PSOC™4 device families
4. CAPSENSE™ design guide**:** Learn how to design capacitive touch-sensing applications with the Automotive PSOC™ 4 family of devices
5. Application notes and code examples cover a broad range of topics, from basic to advanced level. Many of the application notes include code examples. ModusToolbox™ provides additional code examples – see Code examples
6. Technical Reference Manuals provide detailed descriptions of the architecture and registers in each Automotive PSOC™4 device family
7. Development Kits**:**
   - CY8CKIT-040, CY8CKIT-042, CY8CKIT-044, and CY8CKIT-046 are Automotive 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, CY8CKIT-043, CY8CKIT-149 and CY8CKIT-041S-MAX are very low cost prototyping platforms for sampling Automotive PSOC™4 devices
   - CY8CKIT-001 is a common development platform for all Automotive PSOC™ family devices
8. The MiniProg3 device provides an interface for flash programming and debug

2.1 ModusToolbox™

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

1. Configure components using configuration tools
2. Add and manage middleware and libraries using the Library Manager
3. Review component datasheets directly within the IDE
4. Co-design your application firmware with the Automotive PSOC™ hardware

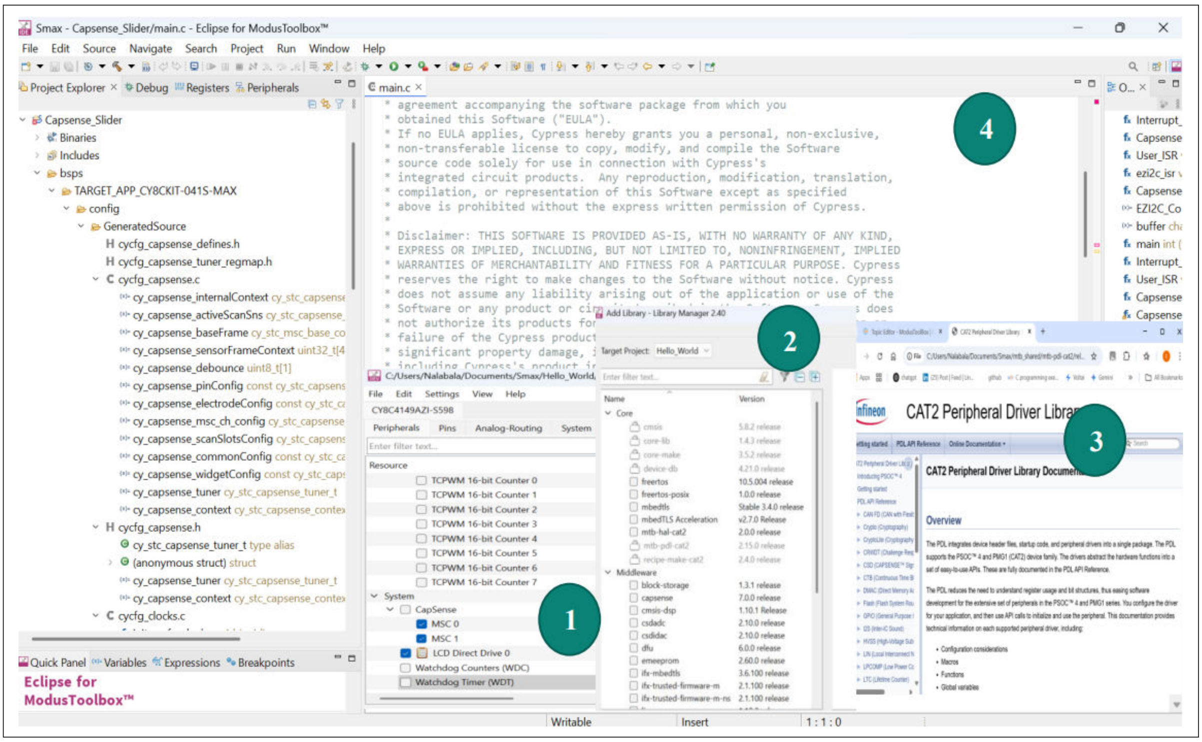

Figure 1 **ModusToolBox™ features**

2.2 Code examples

ModusToolbox™ includes a large number of code example projects . 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 ModusToolbox™ components can be used for various applications. Code examples are included, as Figure 3 shows.

Steps on how to open code examples in ModusToolbox™:

There are two main ways to open examples:

1. Go to **File** > **New** > **ModusToolbox™ New Application**
   - This opens the **Project Creation Wizard**
2. Select your target development kit (e.g., CY8CKIT -041S-MAX) as Figure 2 shows

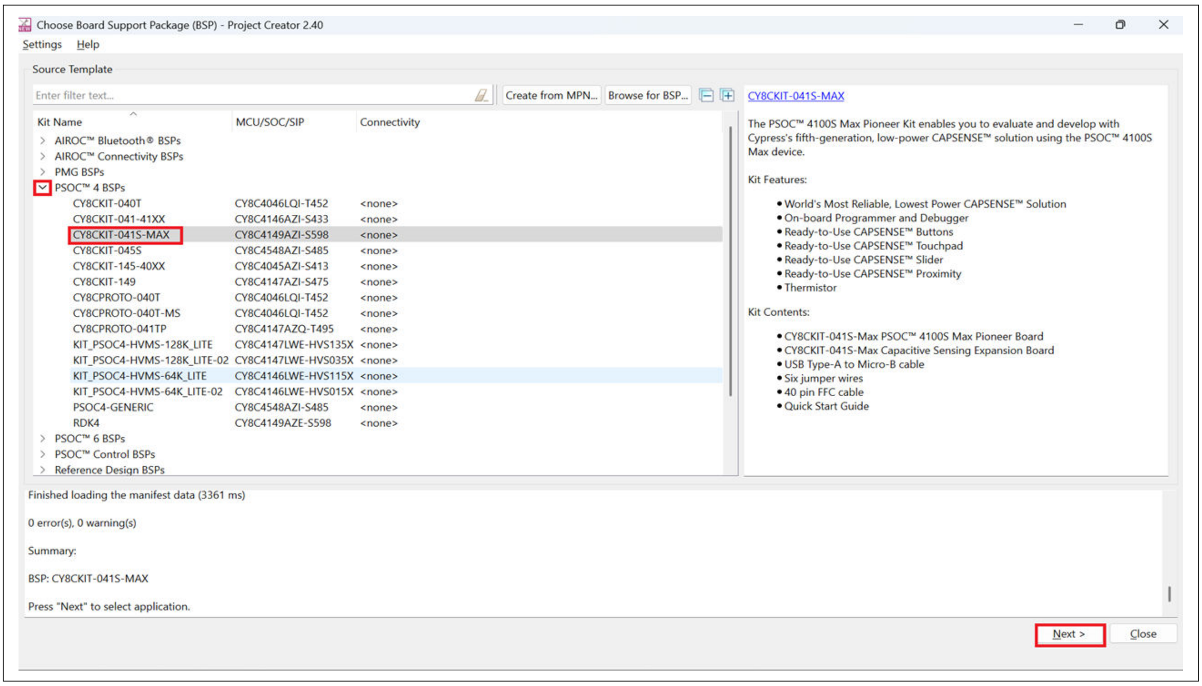

Figure 2 Target boards in ModusToolbox™

3. Choose a code example
   - After selecting the board, a list of available code examples appears

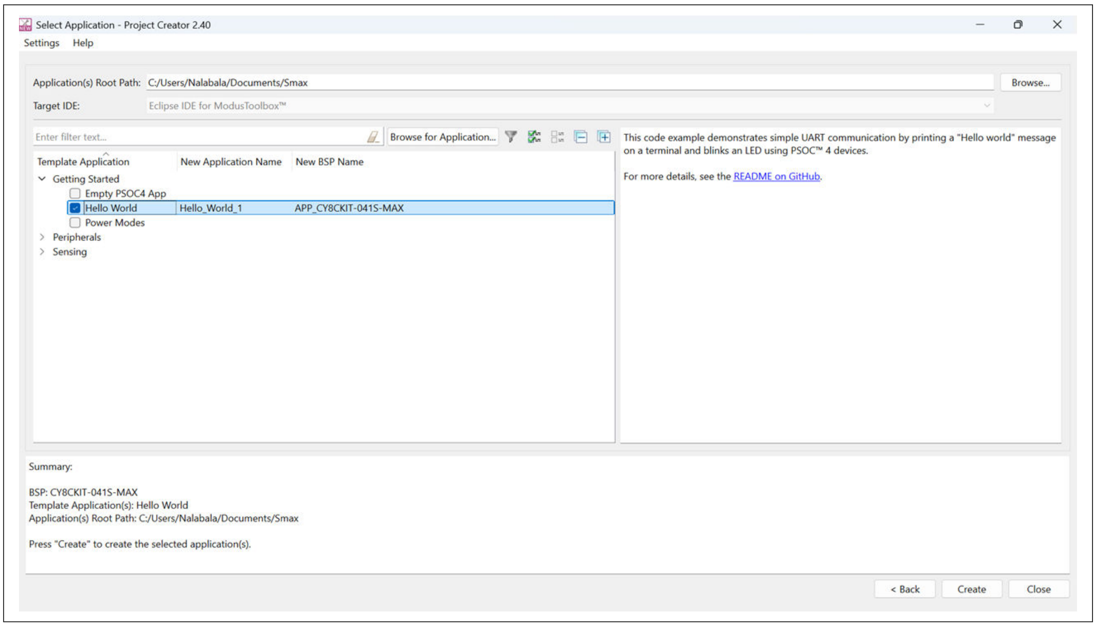

Figure 3 Code examples in ModusToolbox™

4. Select the Example Project
   - Click on the example you want (e.g., Hello World)
   - The right panel shows a description
5. Click **Next**, then **Finish**
   - The project will be imported into your workspace

2.3 ModusToolbox™ Help

Visit the ModusToolbox™ home page to download the latest version of ModusToolbox™™ and navigate to the following items:

- **User guide:** Choose **Help > Eclipse for ModusToolbox™ Documentation > User Guide**. Provides comprehensive, detailed documentation on all features, tools, and workflows in ModusToolbox™
- **Survival guide:** Choose **Help > ModusToolbox™ Eclipse for Documentation > Survival Guide**. Focuses on essential tasks like creating projects, building firmware, and programming devices
- **ModusToolbox™ Training material:** Choose **Help > Eclipse for ModusToolbox™ Documentation > ModusToolbox™ Training Material**. Contains tutorials, videos, labs, and example projects for hands-on learning
- **Component datasheets:** Navigate to the **<project name> API Documentation** section under your project. Click on **CAT2 Peripheral Driver Library: Main Page** to open the detailed component documentation
- **Code examples:** Click **New Application** in the **Start** section of the ModusToolbox™ home window. Select your target development board, then choose from a list of available **example projects** tailored to that board

2.4 Technical support

For more queries, reach out to Infineon Support by creating a request on the Infineon 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 for quick assistance.

- Self-help
- Local sales office locations

3 Features of segment LCD drive in Automotive PSOC™ 4

Automotive PSOC™ 4's segment LCD drive has the following features:

- Supports up to 8 common (COM) and 56 segment (SEG) electrodes
- Programmable Automotive 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 Automotive PSOC™ 4's Active, Sleep, and Deep Sleep system power modes
- Can drive a 3 V display from 1.8 V VDD
- Digital contrast control

***Note**: The number of commons and segments supported by a Automotive 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.

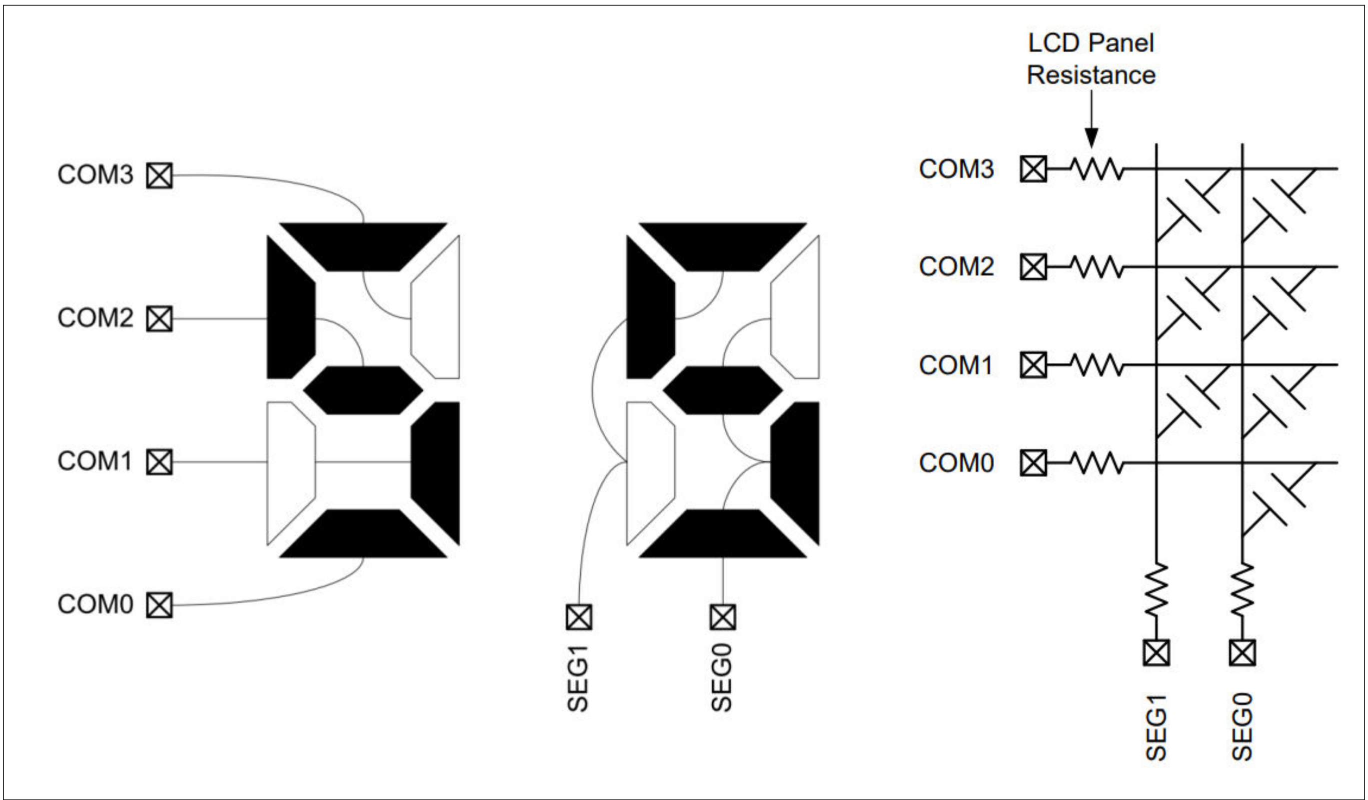

Figure 4 COM and SEG electrodes

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/Mth duty when it drives 'M' number of COM electrodes. Each COM electrode is effectively driven 1/Mth of the time. AutomotivePSOC™ 4100 and 4200 devices support 1/2nd, 1/3rd , and 1/4th duties. Automotive PSOC™4100M, 4200M, 4100L, and 4200L devices support 1/2nd, 1/3rd, 1/4th, and 1/8 th duties
- **Bias:** A driver is said to use 1/Bth bias when its waveforms use voltage steps of (1/B) × VDRV. VDRV is the highest drive voltage in the system (equal to VDD in Automotive PSOC™4). Automotive PSOC™ 4100 4200 devices supports 1/2nd and 1/3rd biases in PWM drive modes. Automotive PSOC™ 4100M, 4200M, 4100L, and 4200L devices support 1/2nd , 1/3rd , 1/4th, and 1/5th duties in PWM drive modes. For more information, see Table-1
- **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.

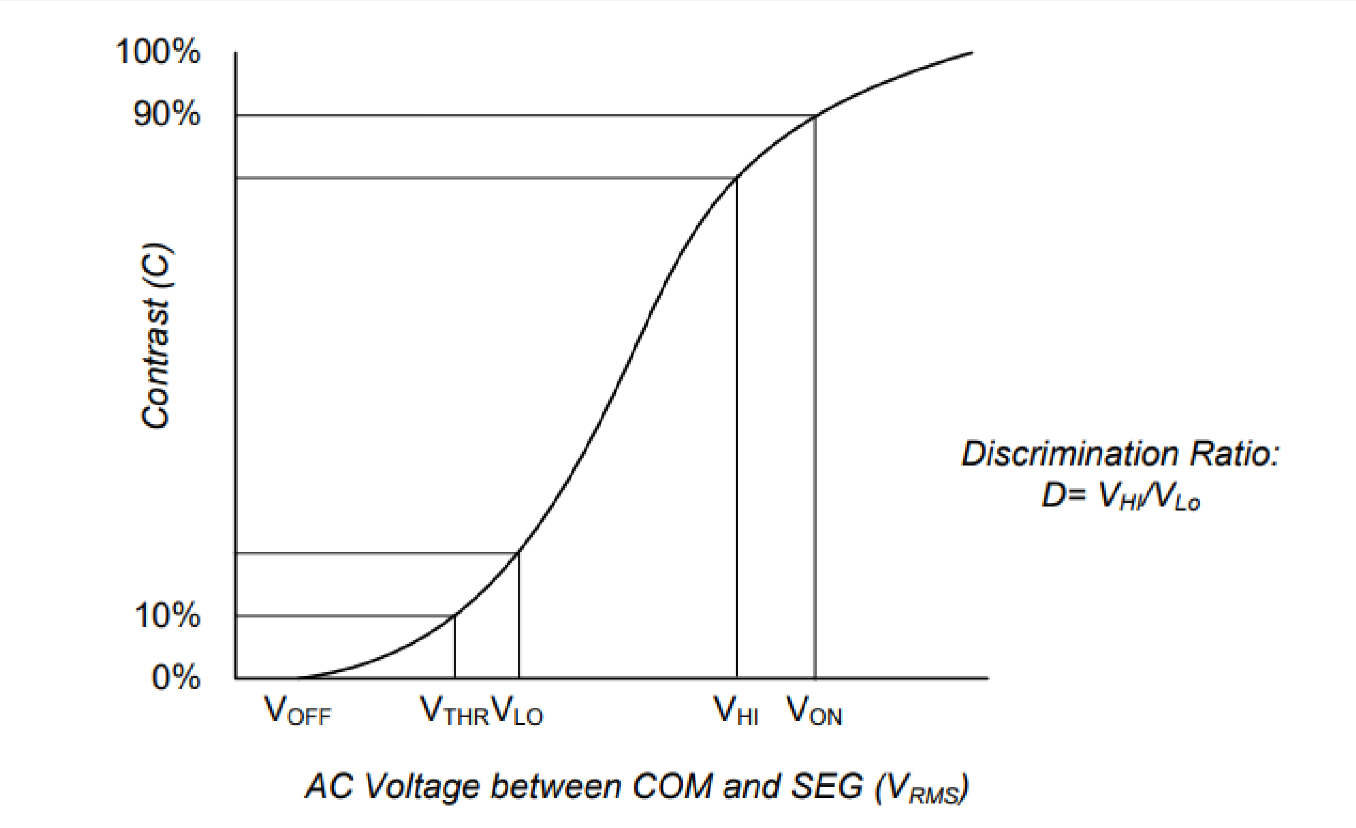

Figure 5 **Relationship between RMS voltage and contrast**

The terms in Figure 5 are defined as follows:

- **VOFF:** The RMS voltage across the pixel below which no contrast change is visible on the LCD
- **VTHR:** The RMS voltage across the pixel at which 10% contrast is achieved
- **VON:** The RMS voltage across the pixel at which 90% contrast is achieved
- **VLO:** The lowest RMS voltage which the LCD driver can generate
- **VHI:** The highest RMS voltage which the LCD driver can generate
- **Discrimination Ratio (D):** The ratio between VHI and VLO 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 Architecture Reference Manual

5 ModusToolbox™ Component: Segment LCD

ModusToolbox™ provides a Segment LCD Component which allows you to easily interface different types of segment LCD display modules with Automotive PSOC™ 4. Navigate to BSP Configurator in ModusToolbox™, enable the Segment LCD Configurator, and configure it according to your LCD module's specifications. The Component also provides an easy-to-use application programming interface (API). For more information, see the Automotive PSOC™ 4 Segment LCD Component APIs.

Additionally, see the project "Example 1: Simple segment LCD". This project shows you how to interface a 4-digit 7-segment LCD module with Automotive PSOC™ 4 by using the Segment LCD Component and APIs.

6 Automotive PSOC™ 4 segment LCD Direct drive

Automotive 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 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 Hz 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.

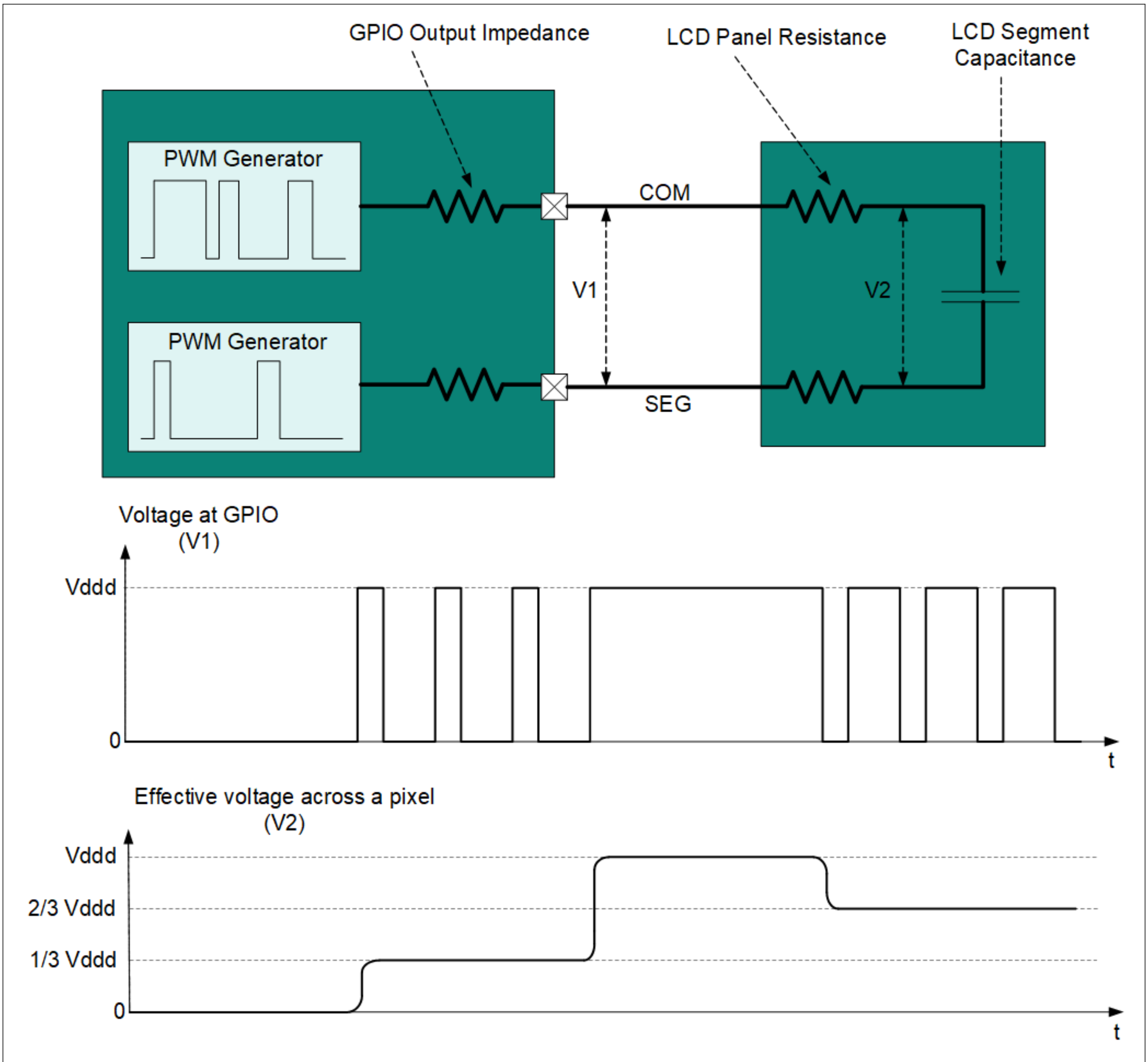

Figure 6 **PWM Drive mode**

6.2 Digital correlation drive mode

The digital correlation drive mode is a patent-pending Infineon 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 VRMS across the segments, which in turn affects the segment's contrast (Figure 5). Figure 7 shows an example of digital correlation mode waveforms.

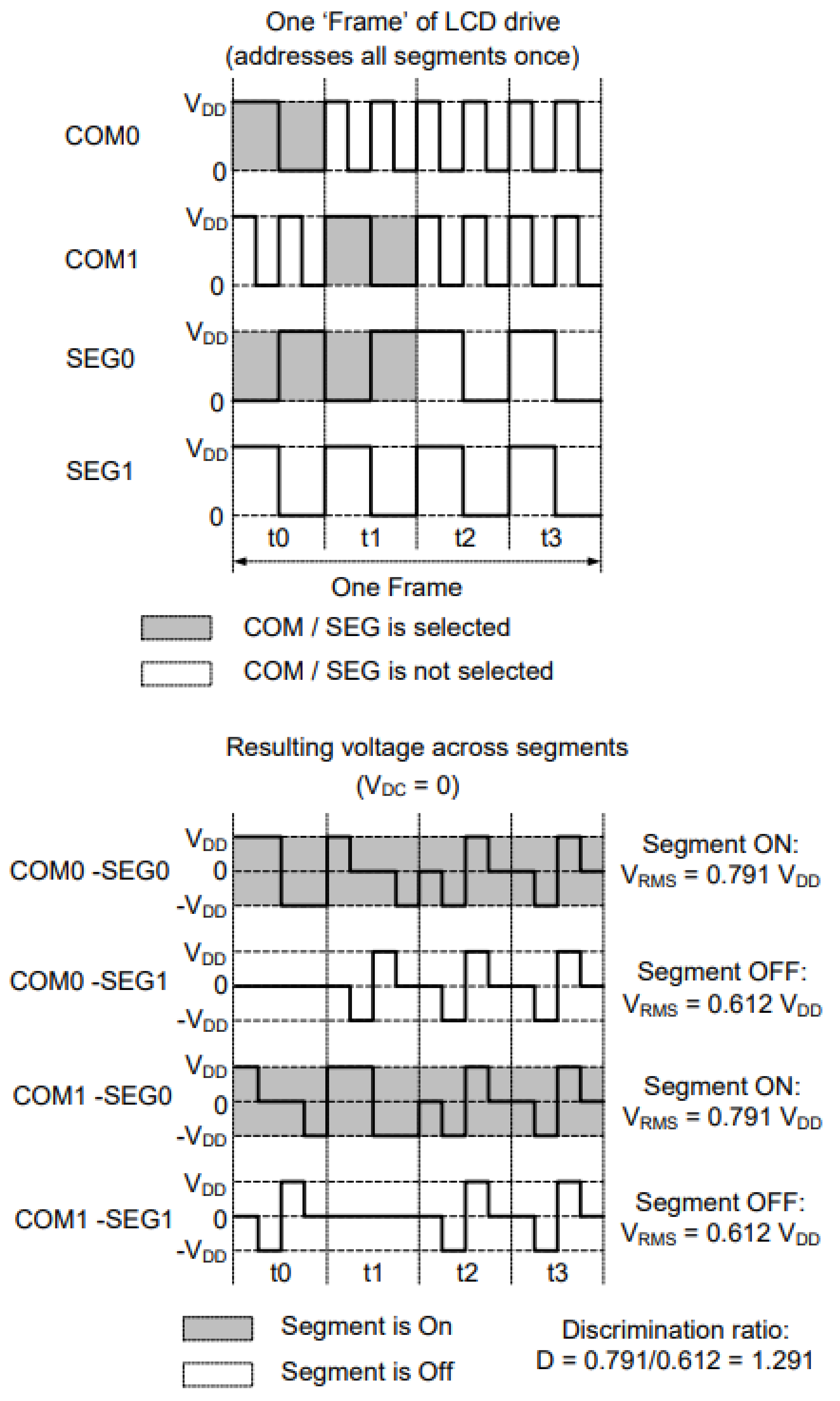

Figure 7 Digital correlation example

See the Automotive 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 AutomotivePSOC™ 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 Automotive 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 Automotive 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, µA	
	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

(table continues...)

Table 2 (continued) Typical device current consumption in Deep Sleep mode with Segment LCD enabled

Frame rate, Hz	Current consumption, µA	
	Digital correlation mode	PWM mode
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 "Example 2: Low-power segment LCD" project for more details.

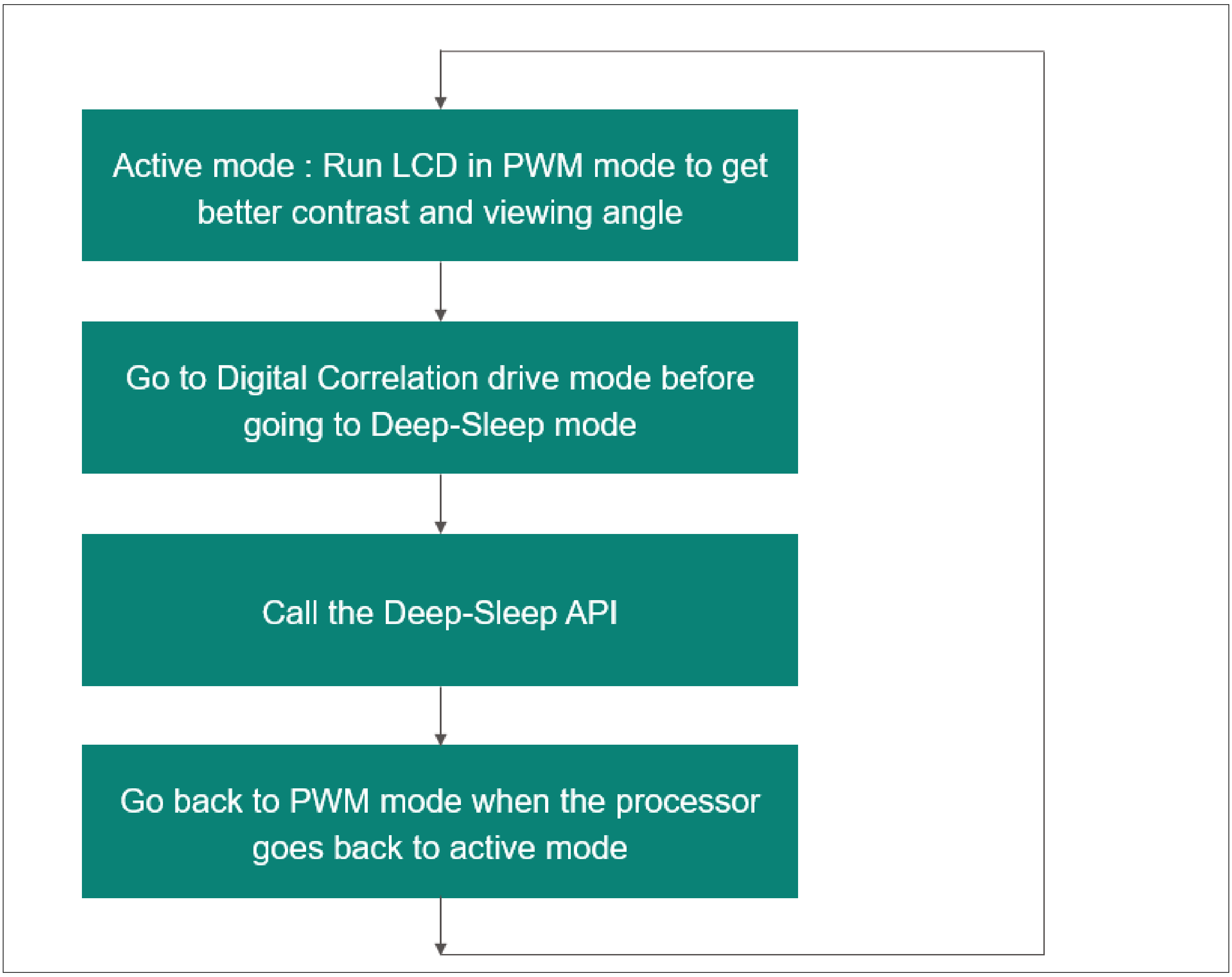

Figure 8 **Firmware flow of low-power LCD operation**

7 PSOC™ Automotive 4 user interface solution

This section shows how to create a complete user interface (UI) solutions using just Automotive PSOC™ 4. A basic knowledge of PSOC™ Automotive 4 CAPSENSE™ is a prerequisite for this section. See the PSOC™ Automotive 4 CAPSENSE™ design guide to understand the basics of CAPSENSE™.

The CAPSENSE™ feature in Automotive 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™ Automotive drives a segment LCD and scans a touch panel. The touch panel consists of two CAPSENSE™ buttons and a slider.

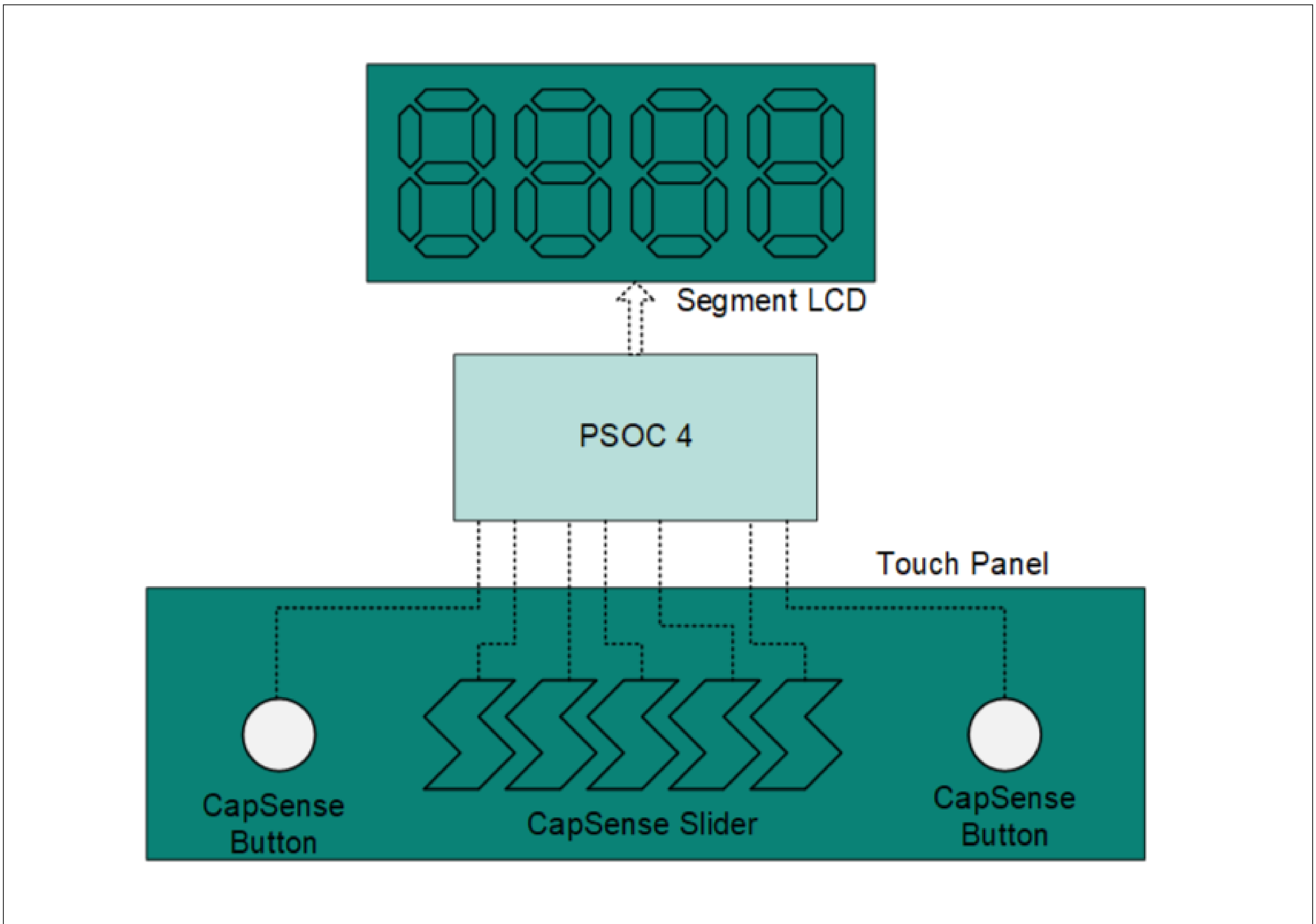

Figure 9 Example of Automotive PSOC™ 4 user interface

CAPSENSE™ cannot scan sensors in Deep Sleep mode, so the device must wake up periodically to scan the sensors. AutomotivePSOC™ 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, Automotive 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.

Automotive 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 UI solution.

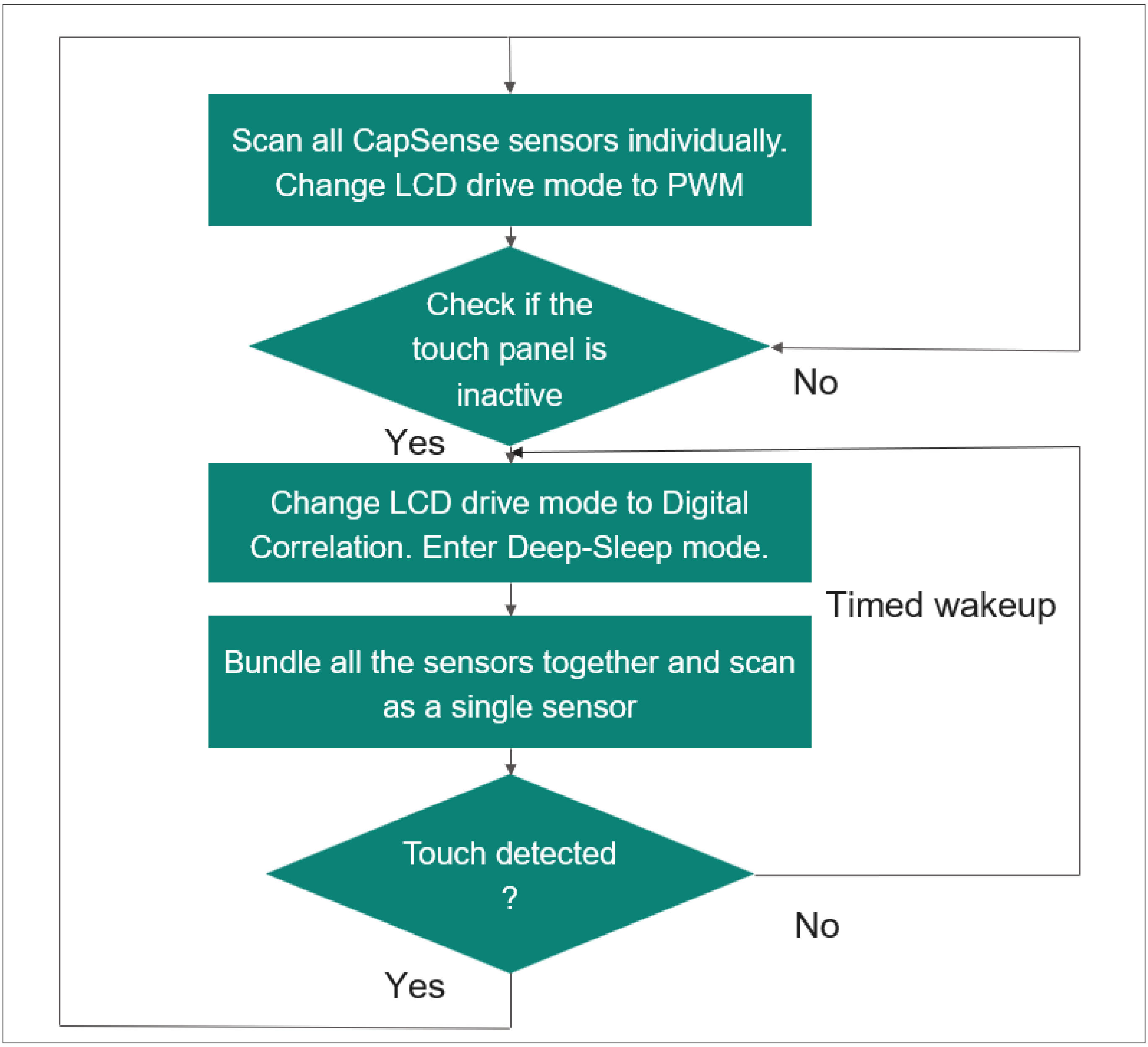

Figure 10 **Firmware flow of a low-power UI solution**

See the "Example 3: Segment LCD and CAPSENSE™ user interface" project for more details. This project demonstrates a user interface with a four-digit, seven-segment LCD, and a five-element CAPSENSE™ slider.

8 ModusToolbox™ projects

This application note provides four example projects:

1. **Simple Segment LCD:** This is a simple project to demonstrate basic usage of the ModusToolbox™AutomotivePSOC™ 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 Automotive PSOC™ 4 kit or your own hardware to evaluate projects 1 and 2. Project 3 requires the Infineon CY8CKIT-149 or CY8CKIT-041S-MAXor CY8CKIT-042kit, because for that project both CAPSENSE™ buttons and segment LCD interface are required

   All three projects require a VIM 404 segment LCD module, which has four seven-segment digits, three COM electrodes, and 12 SEG electrodes. See Figure 16 for a wiring schematic
4. **Alphanumeric Segment LCD:** This project takes advantages of the large number of GPIOs available in the Automotive PSOC™ 4100M/4200M/4100S Plus/4100S Max 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 Infineon CY8CKIT-043 or CY8CKIT-046, or CY8CKIT-149 and a VIM-828 Segment LCD module

8.1 Example 1: Simple segment LCD

To create this project, do the following:

1. To create a new project in ModusToolbox™, click on **New Application** under the **Start** section.

   The **Project Creator** window will open.
2. Click the arrow next to "PSOC 4 BSPs," then select **CY8CKIT-041S-MAX**

   A description of the selected kit will appear on the right side.
3. Click **Next** as Figure 11 shows
4. Click the arrow next to "Getting Started," check the box beside "Empty PSOC4 App," and enter a project name of your choice — for example, "Scrolling_Numbers_LCD"—then click **Create**. The project will now be created
5. After creating the project, open the **Device Configurator** by clicking on it under the **Quick Panel**to begin configuring components. In the configurator, check the box next to **LCD Direct Drive 0** to enable it as Figure 12 shows.

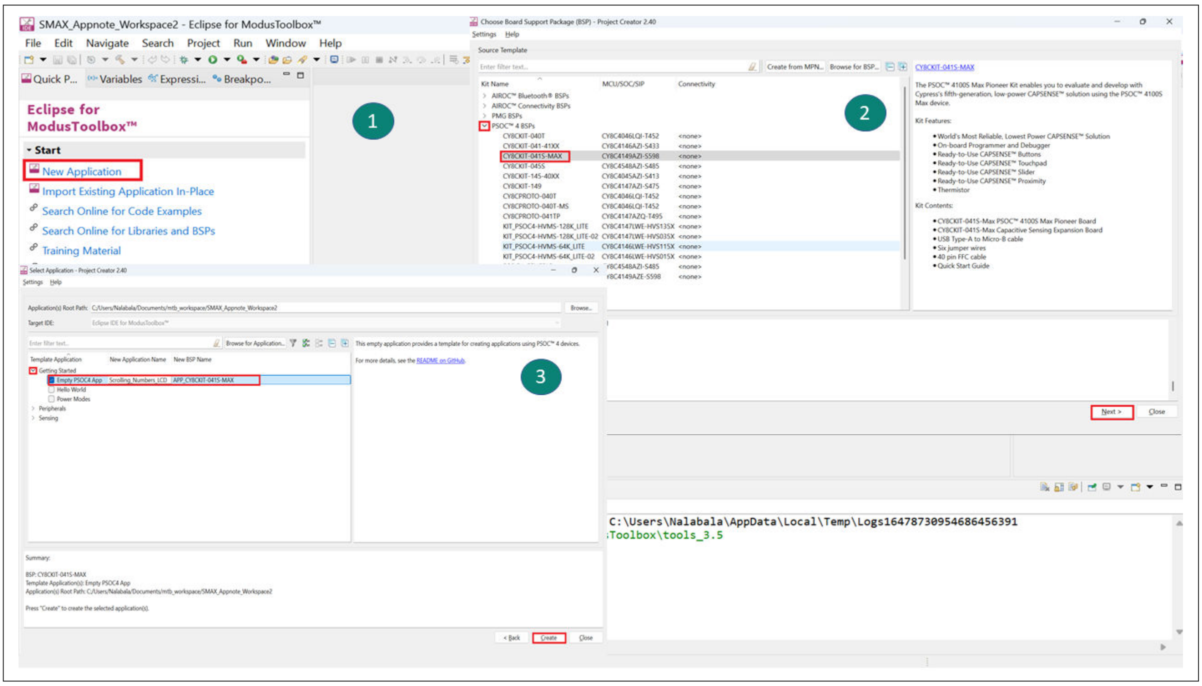

Figure 11 Creating a project and selecting Device Configurator

6. Navigate to the **Peripherals** tab. Locate the "Segment LCD (SegLCD)" peripheral and enable it. And in the **General** tab
   - Set the Drive Mode to "PWM"
   - Select "1/2 Bias" for optimal contrast on the VIM-404 module
   - Choose "Type B" Waveform to reduce power consumption
   - Configure the Clock Source appropriately
   - Leave all other settings at their default values
7. In the same **Quick Panel** tab under the BSP configurators section click **Launch SegLCD Configurator** as Figure 13 shows

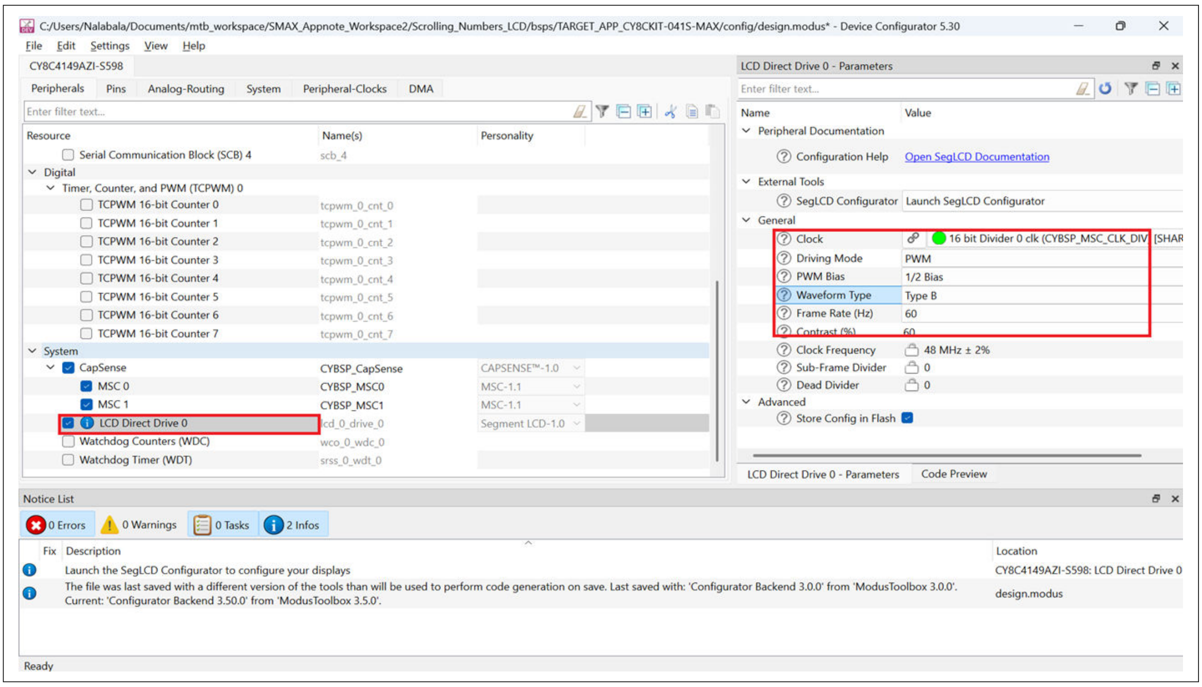

Figure 12 Device Configurator

8. Set the number of Commons (COM) to "3" and Segments (SEG) to "12", to match the VIM 404 segment LCD module

9. Click the **Commons Edit Displays** option

10. Click the **"+"** (Add) button to create a new display block

11. Select **7-Segment** in **Type Section** and select "4" in **Symbols** section to get four 7-Segments

12. Drag a segment from a digit and drop it onto the pixel mapping table, as Figure 13 shows. See the VIM-404 datasheet for the pixel mapping of the VIM 404 LCD module

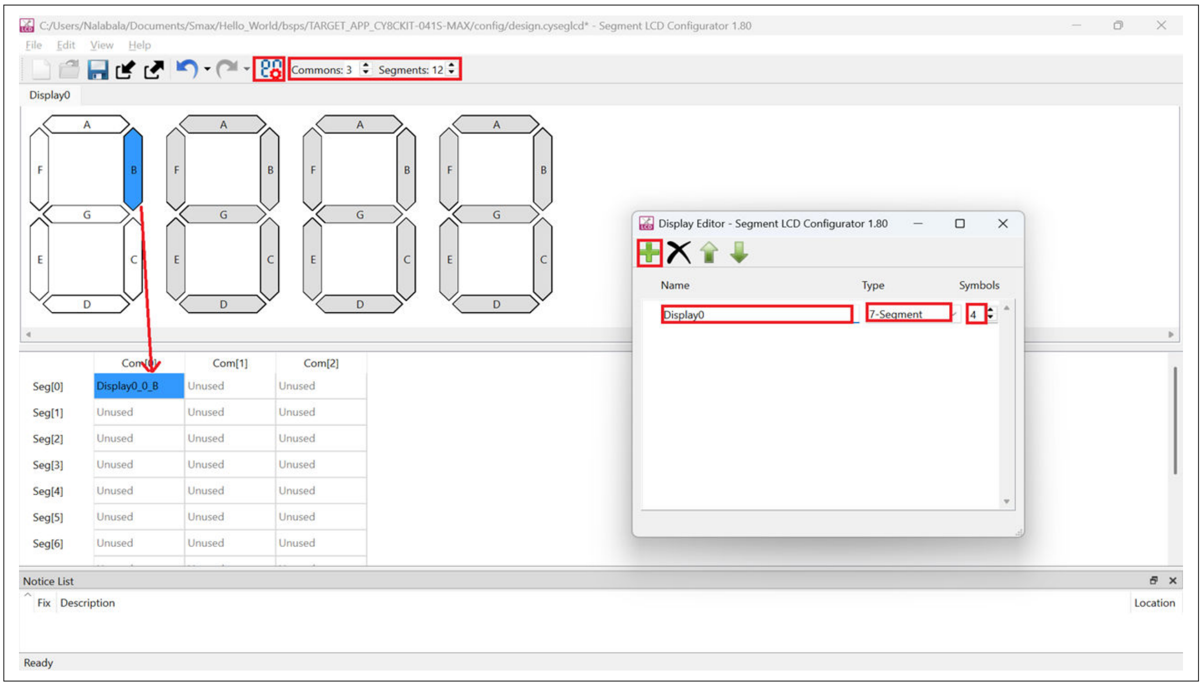

Figure 13 **Segment LCD configuration**

13. Repeat the process for each segment in the display. After mapping all of the pixels, the completed table appears as shown in Figure 14

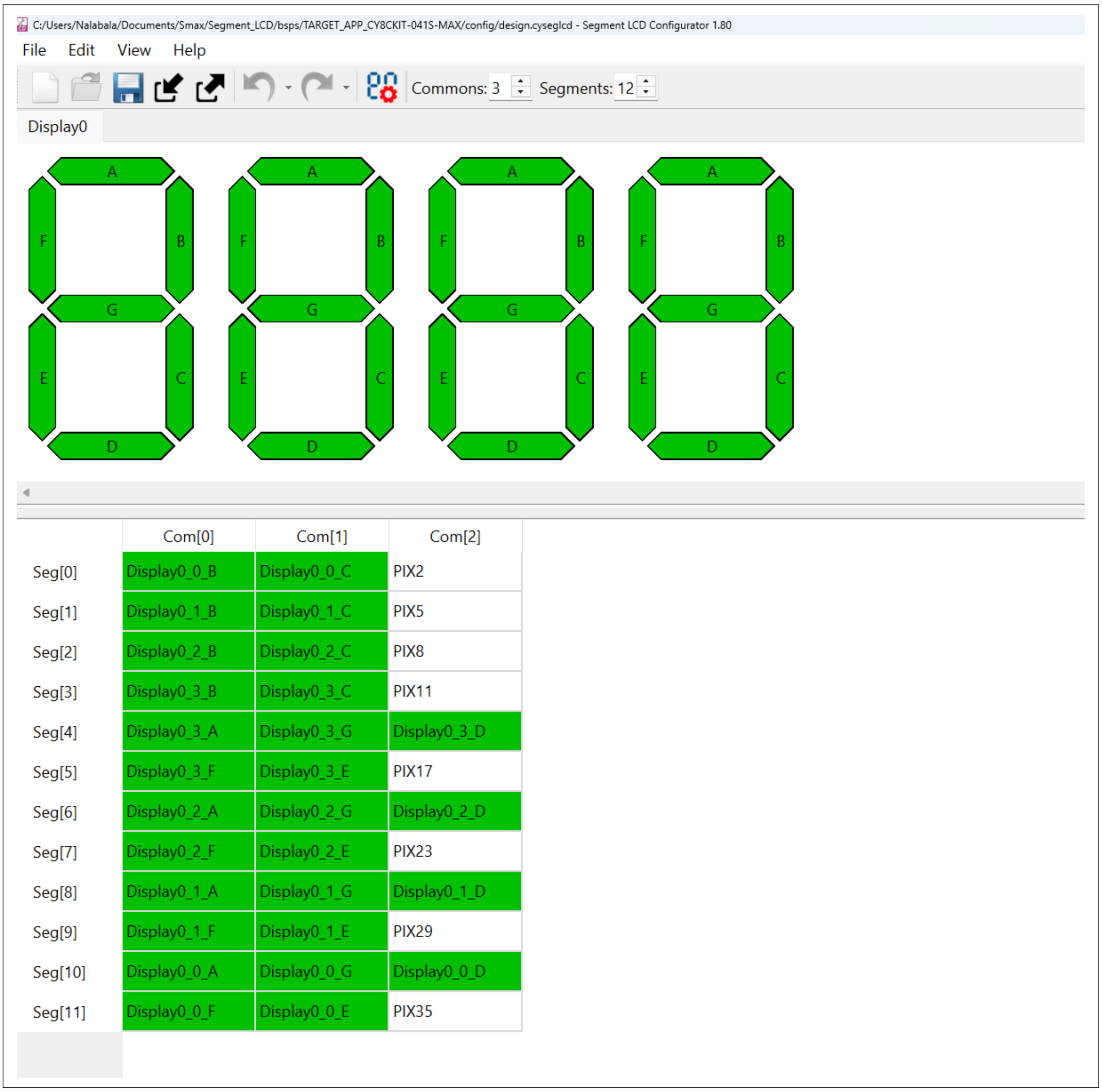

Figure 14 Completed pixel mapping table

14. Click **OK**; the Segment LCD Component configuration is now complete. To assign pins for the commons and segments, open the Device Configurator and in pins tab assign pins as shown in Figure 15

***Note**:* *The pin assignment shown here is for the CY8CKIT-041-SMAX 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.*

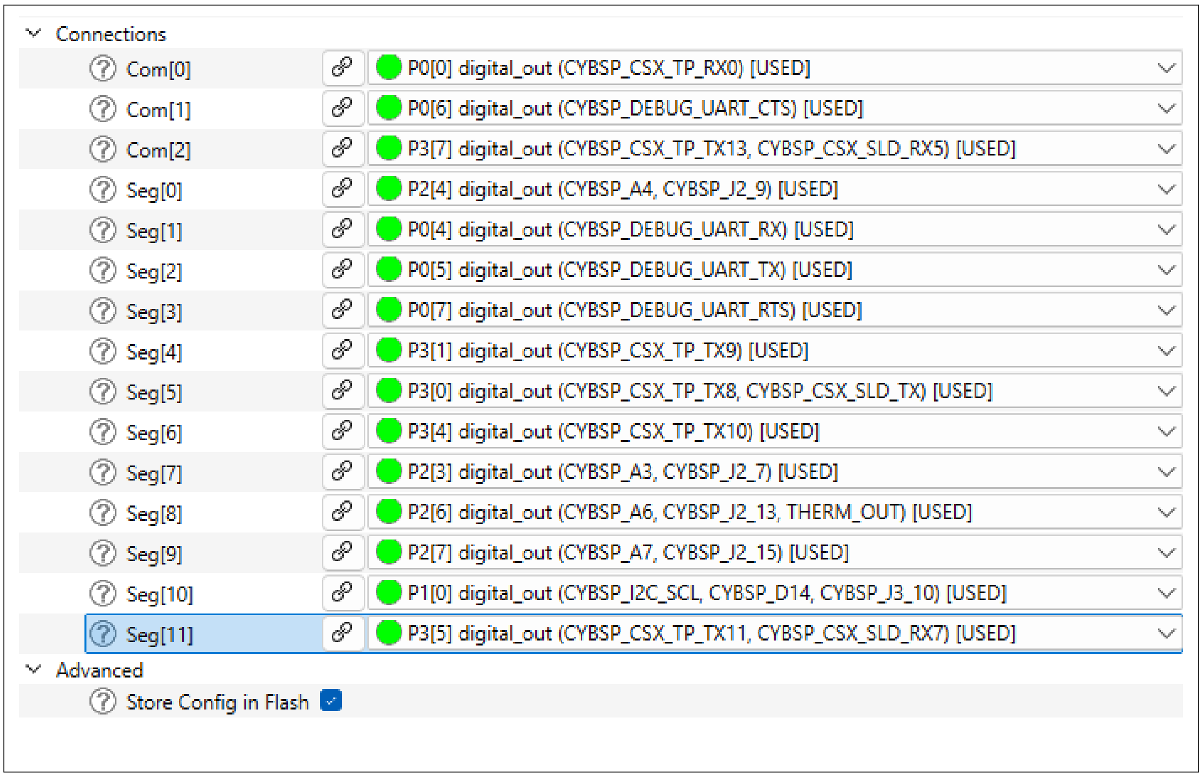

Figure 15 **Figure 15. Pin assignment**

15. Wire the VIM 404 LCD glass to Automotive PSOC™ 4, as Figure 16 shows. This can be done on your own PCB, or you can use the CY8CKIT-041-SMAX kit and an Arduino Protoshield board

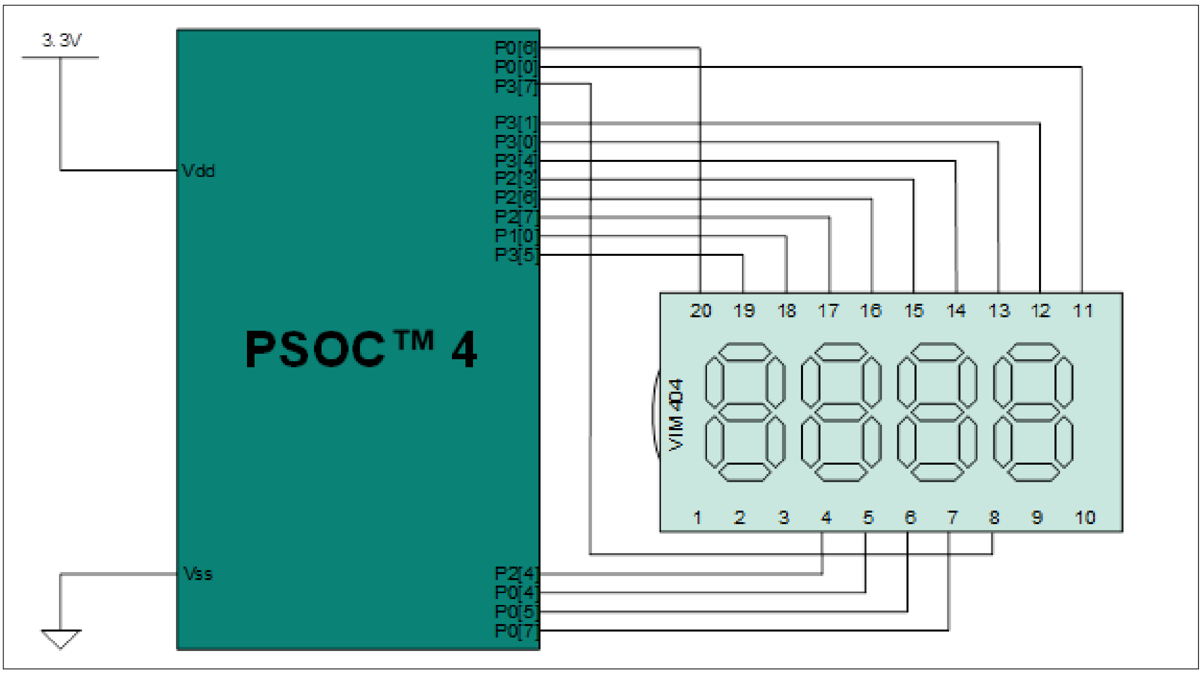

Figure 16 **Automotive PSOC™ 4 to LCD module external connections**

16. Add the below 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 Automotive PSOC™ 4 Peripheral Driver Library.

```
#include "cybsp.h"
#include "cycfg_seglcd.h"

#define SCROLL_LENGTH (10u)     /* Macro for scrolling length */
#define INITIAL_VALUE (0u)      /* Macro for initial scrolling value */
#define SCROLL0 (0u)            /* Define the numbers used for scrolling the 4-digit
display */
#define SCROLL1 (1u)
#define SCROLL2 (2u)
#define SCROLL3 (3u)
#define DIGIT0 (0u)
#define DIGIT1 (1u)
#define DIGIT2 (2u)
#define DIGIT3 (3u)

#define LOOP_DELAY (1000u)      /* Define the loop delay */

int main(void) {
     cy_en_seglcd_status_t segLCD_status;
     cy_rslt_t result = cybsp_init();
         if (result != CY_RSLT_SUCCESS) {
             CY_ASSERT(0);
         }
     uint8_t index = 0;
     uint8_t numbers[] = {0, 1, 2, 3, 4, 5, 6, 7, 8, 9,0,1,2};
     segLCD_status = Cy_SegLCD_Init(LCD, &LCD_DISP_config);                  /*Segment
LCD Initialization */
     if(segLCD_status == CY_SEGLCD_SUCCESS) {
        segLCD_status |= Cy_SegLCD_ClrFrame(LCD, LCD_DISP_commons);        /*Clearing
Segment LCD Initially*/
     }
     if(segLCD_status == CY_SEGLCD_SUCCESS) {
        Cy_SegLCD_Enable(LCD);                                                      /
*Enabling the segment LCD */
     }
     for(;;) {
         for(index = INITIAL_VALUE; index < SCROLL_LENGTH; index++) {
            Cy_SegLCD_WriteNumber(LCD, numbers[index + SCROLL3],
DIGIT3,&LCD_DISP_Display0, false, false);
            Cy_SegLCD_WriteNumber(LCD, numbers[index + SCROLL2],
DIGIT2,&LCD_DISP_Display0, false, false);
            Cy_SegLCD_WriteNumber(LCD, numbers[index + SCROLL1],
DIGIT1,&LCD_DISP_Display0,false, false);
            Cy_SegLCD_WriteNumber(LCD, numbers[index + SCROLL0],
DIGIT0,&LCD_DISP_Display0,false, false);

            Cy_SysLib_Delay(LOOP_DELAY);
         }
```

```
        }
    }
```

17. Make sure that Automotive PSOC™ 4 is operating from 3.3 V VDD 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 17 shows

Figure 17 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 Example 2: Low-power segment LCD

This project is a modification of Project 1, for lower average power.

To reduce power consumption, Automotive PSOC™ 4 stays in Active mode just long enough to update the display and scroll the numbers one place, which takes approximately 720 µs. The device then goes into the Deep Sleep power mode for the next 512 ms. Figure 18 shows the timing diagram and Figure 19 shows the firmware flow.

With this technique, Automotive PSOC™ 4 consumes an average current less than 11 µA. 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 µA
Average current	2.7 mA	10.9 µA

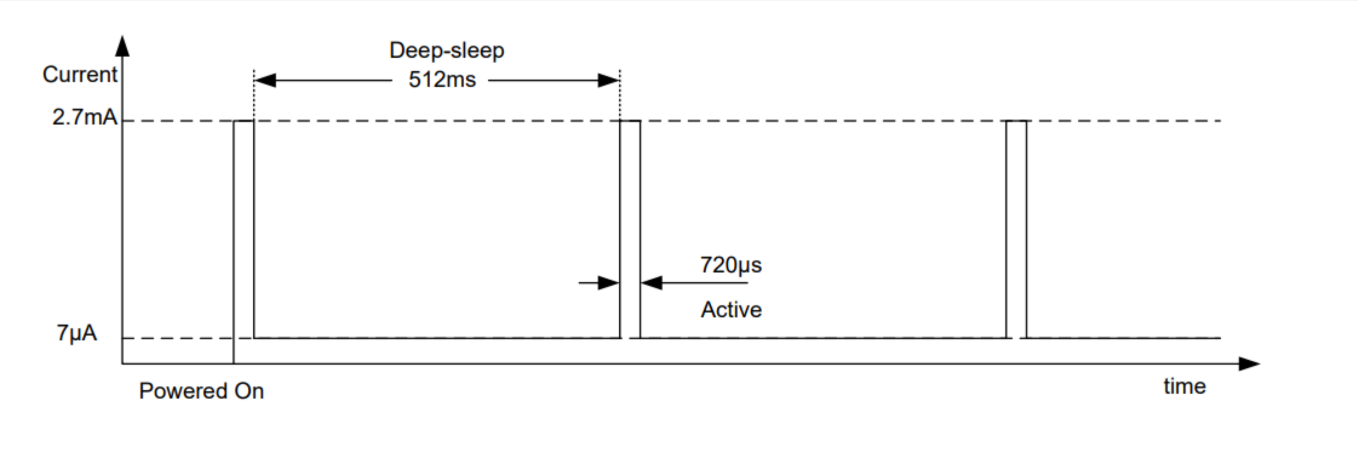

Figure 18 **Project 2 timing diagram**

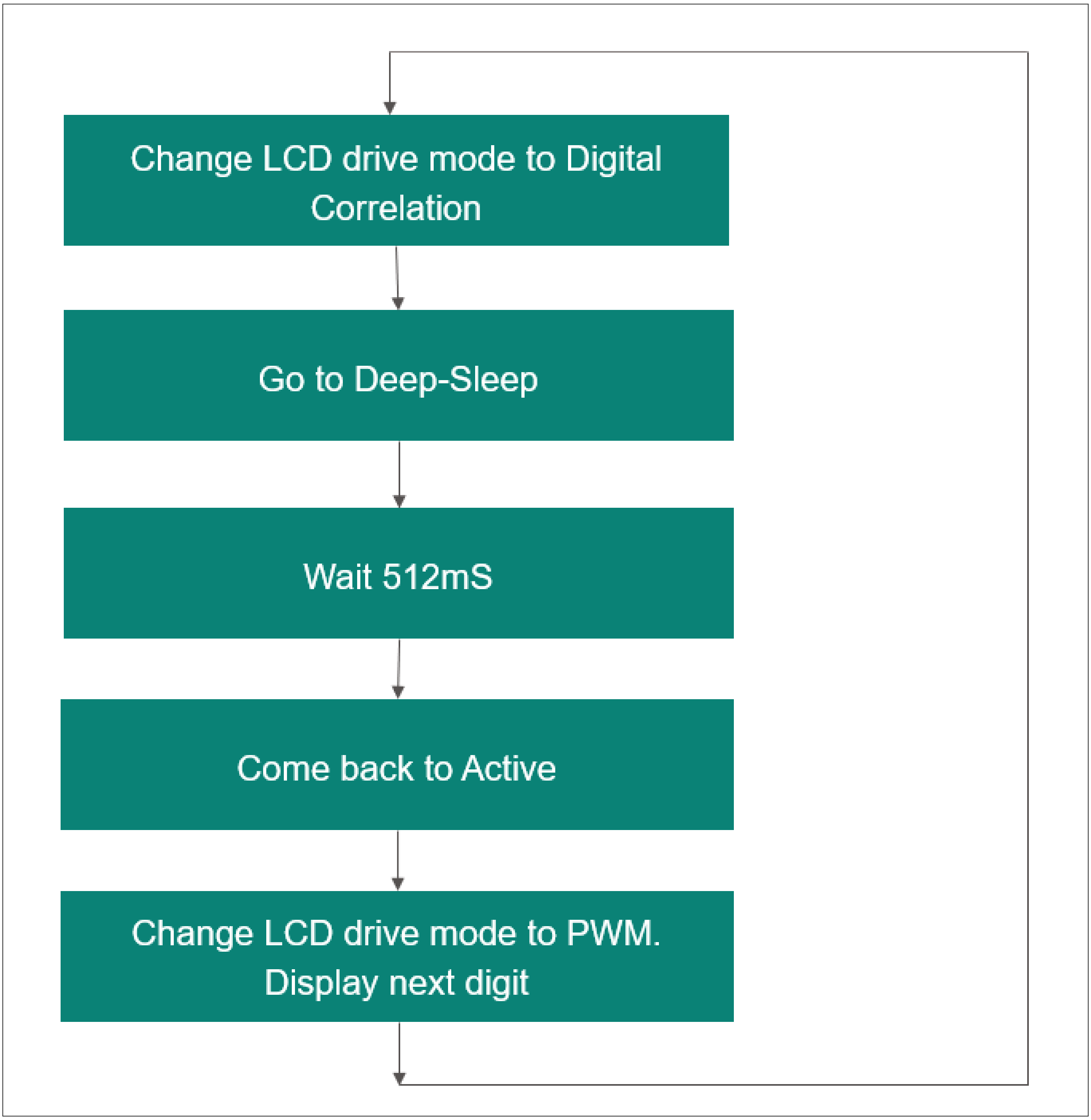

Figure 19 **Project 2 firmware flow**

This project uses watchdog timer (WDT) interrupt. Configure the Component as Figure 20 shows. The WDT is configured to generate an interrupt every 512 ms. Figure 21 shows the TopDesign schematic of this project.

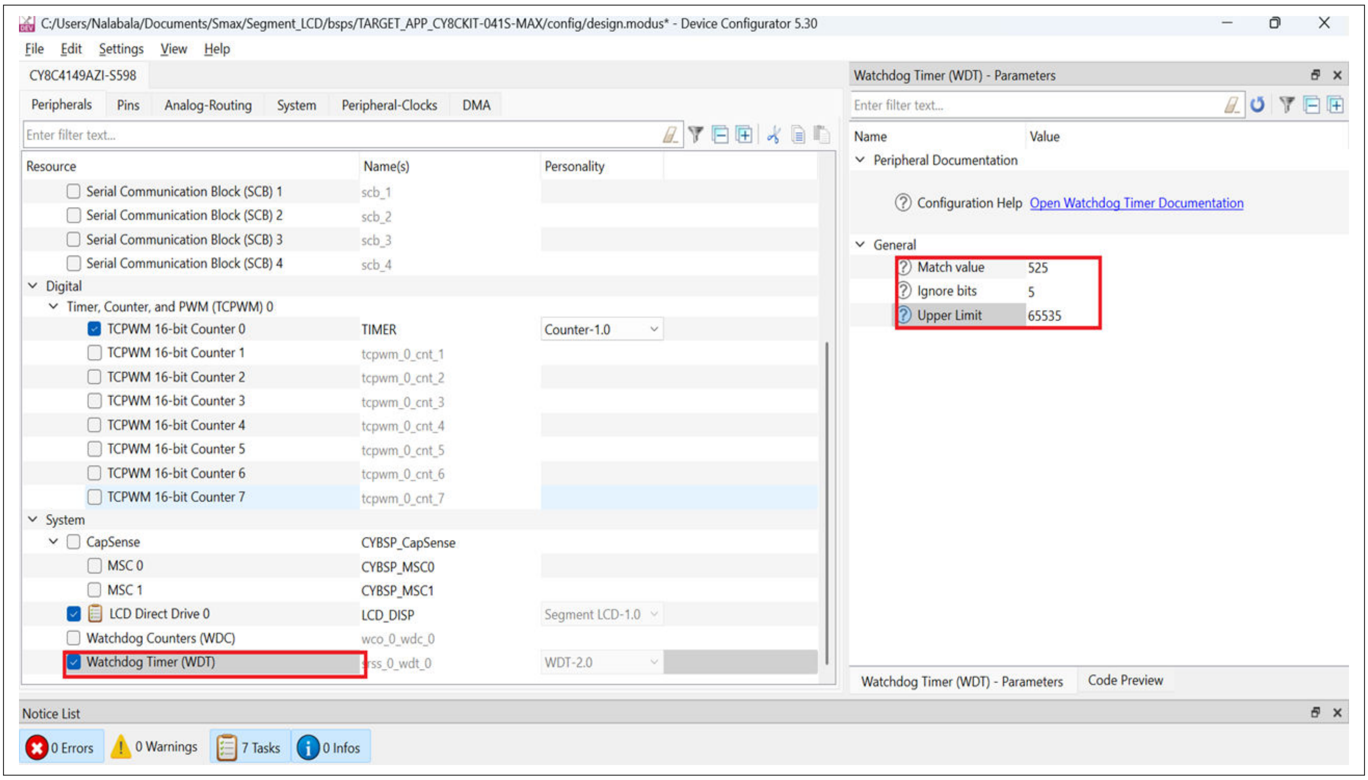

Figure 20 **WDT configuration**

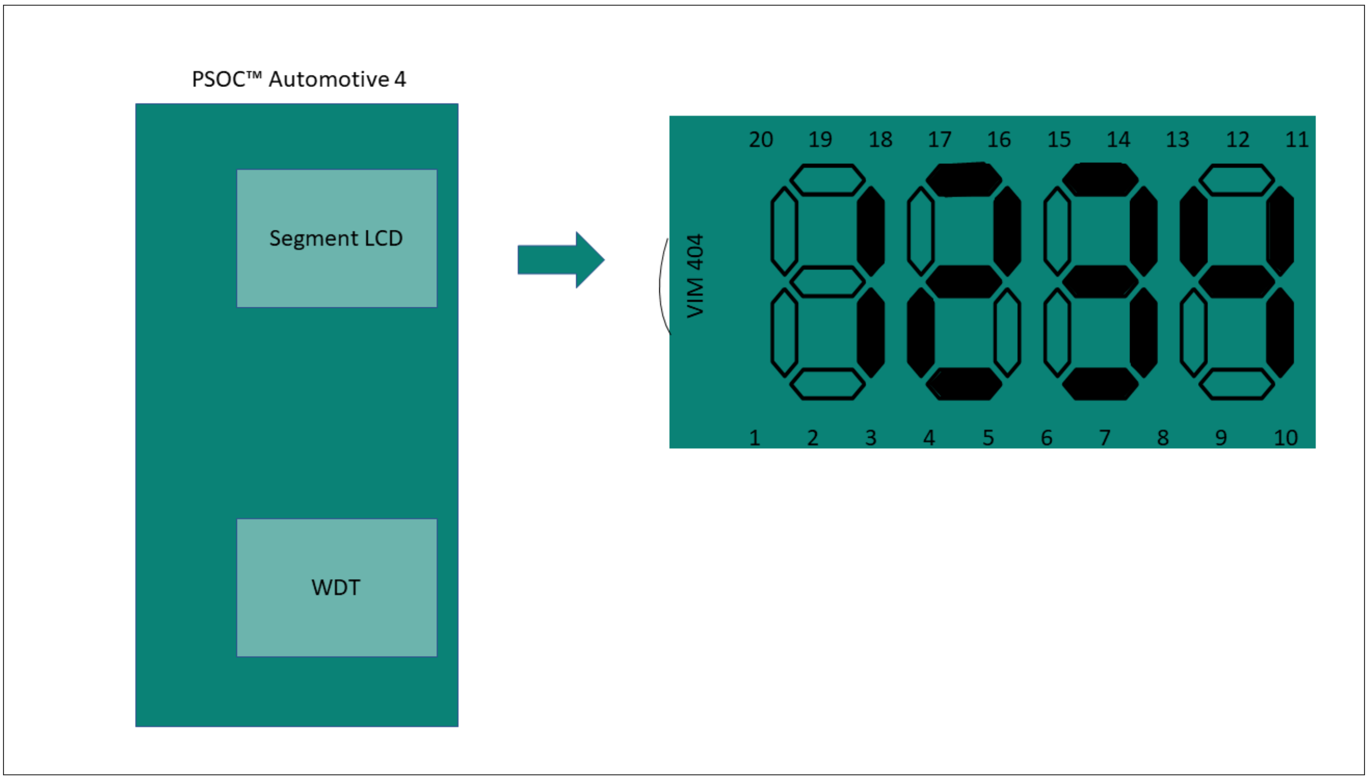

Figure 21 **Project 2 LCD TopDesign schematic**

To test the project, program the device, connect the display as in Project 1 (Figure 16) and view the output. You will still see a scrolling display; however, power consumption for this project is significantly reduced.

8.3 Example 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 Automotive 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* Automotive PSOC™ 4 CAPSENSE™ design guide *to understand manual tuning.*

This project is a modification of Project 2 to add CAPSENSE™ capability. Figure 22 shows the timing diagram, Figure 23 shows the Top Design schematic for this project, and Figure 24 shows the firmware flow.

In this project, Automotive 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 Automotive PSOC™ 4 is less than 21 µA.

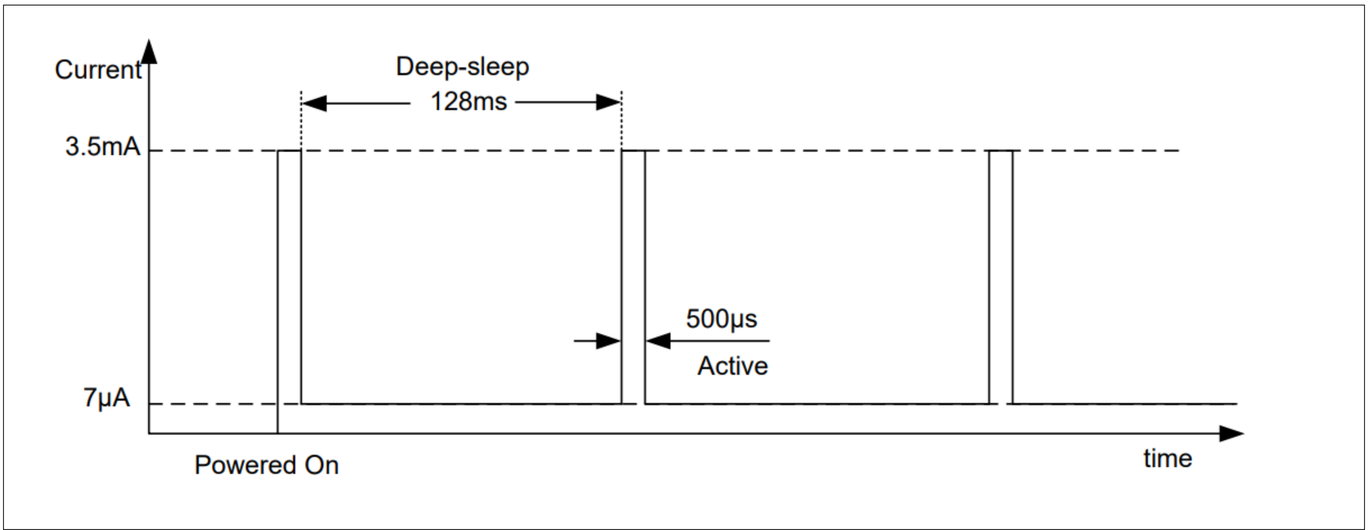

Figure 22 Project 3 timing diagram

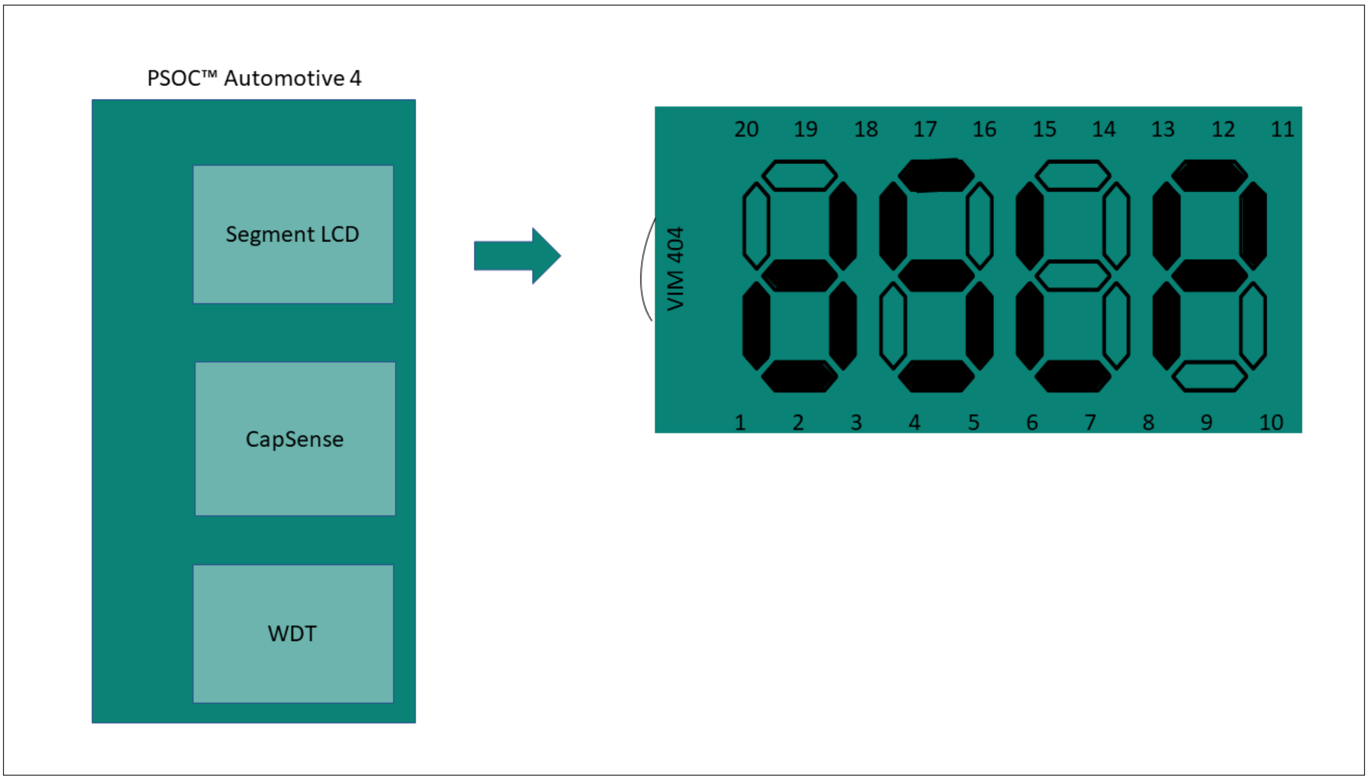

Figure 23 **Project 3 top design schematic**

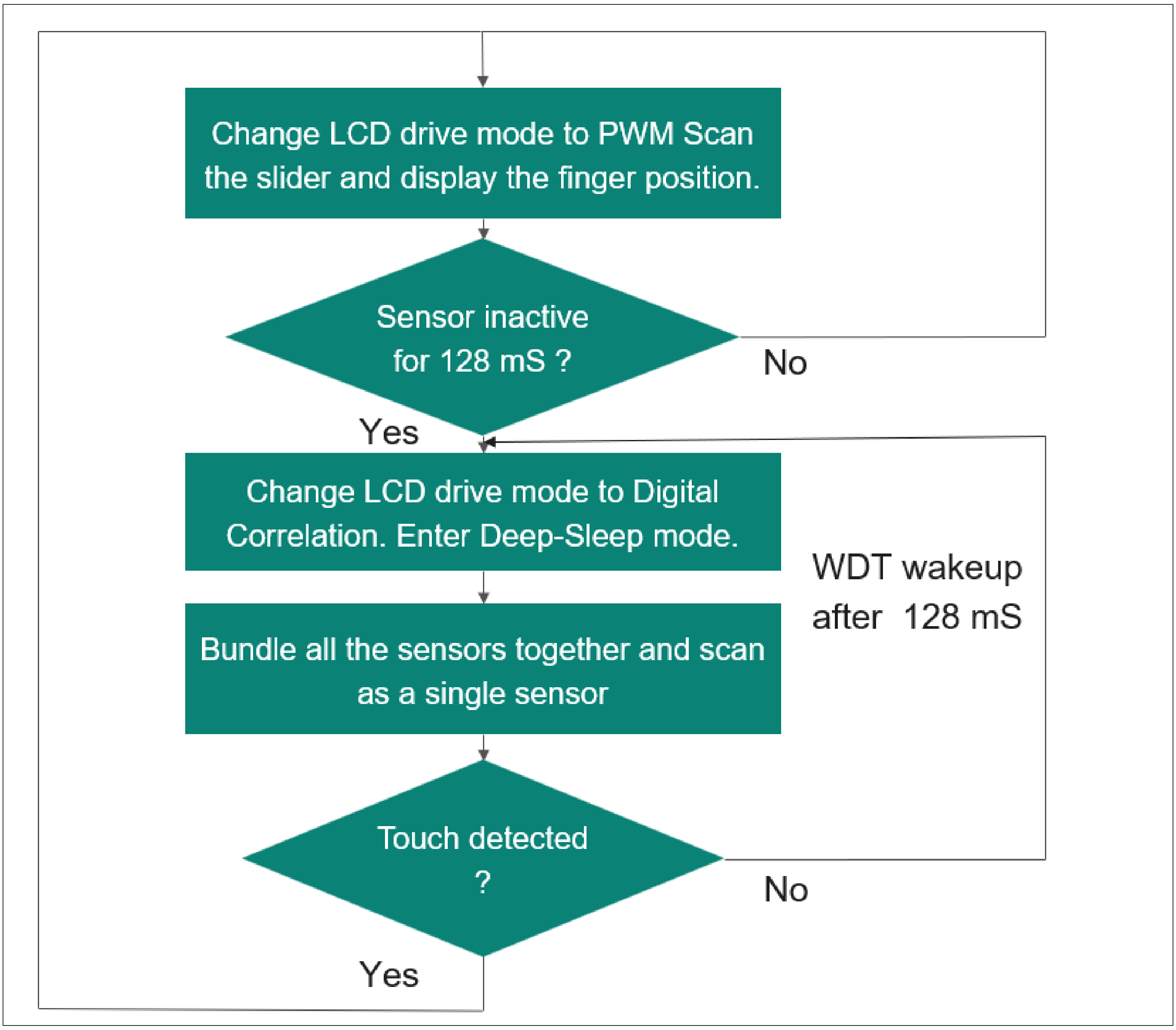

Figure 24 **Project 3 firmware flow**

To test the project, connect the LCD as Figure 16 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-041-SMAX 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 Automotive 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 µA.

8.4 Example 4: Alphanumeric Segment LCD

This project takes advantage of the large number of GPIOs available in the Automotive PSOC™ 4100M/4200M/4100L/4200L/4100S Plus 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 Infineon 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 "PSOC™ Automotive 4200" displayed on VIM-828. See the ModusToolbox™ 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]

(table continues...)

Table 5 (continued) Wiring information

LCD pin number	LCD pin function	CY8CKIT-43 connection	CY8CKIT-046 connection
32	SEG28	P5[5]	P8[3]
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 Automotive PSOC™ 4. The combination of robust capacitive touch sensing and best-in-class low-power capabilities make Automotive 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 Automotive PSOC™ 4 segment LCD, showing low-power operation, and demonstrate a complete user interface.

Note that Automotive 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 Automotive PSOC™ 4.

Revision history

Document version	Date of release	Description of changes
**	2025-09-11	Initial release

Trademarks

PSOC™, formerly known as PSoC™, is a trademark of Infineon Technologies. Any references to PSoC™ in this document or others shall be deemed to refer to PSOC™.

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2025-09-11

Published by

Infineon Technologies AG

81726 Munich, Germany

© 2025 Infineon Technologies AG

All Rights Reserved.

Do you have a question about any aspect of this document?

Email: erratum@infineon.com

Document reference
IFX-spy1748938399655

Important notice

The information contained in this application note is given as a hint for the implementation of the product only and shall in no event be regarded as a description or warranty of a certain functionality, condition or quality of the product. Before implementation of the product, the recipient of this application note must verify any function and other technical information given herein in the real application. Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind (including without limitation warranties of non-infringement of intellectual property rights of any third party) with respect to any and all information given in this application note.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

Warnings

Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.