AN73034 - CY8CMBR2016

CapSense® Design Guide

Doc. No. 001-73034 Rev. *F

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
www.cypress.com

© Cypress Semiconductor Corporation, 2012-2017. 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.

Contents

1. Introduction 5

1.1 Abstract 5
1.2 Cypress's CapSense Documentation Ecosystem 6
1.3 CY8CMBR2016 CapSense® Express Device Features 7
1.4 Document Conventions 8

2. CapSense Technology 9

2.1 CapSense Fundamentals 9
2.2 Capacitive Sensing Method 10
2.2.1 CapSense Sigma-Delta (CSD) 10
2.3 SmartSense Auto-Tuning 12
2.3.1 Process Variation 12
2.3.2 Reduced Design Cycle Time 13

3. CapSense Schematic Design 14

3.1 CY8CMBR2016 Configuration Options 14
3.1.1 CapSense Buttons (CSx Pins) 14
3.1.2 Modulation Capacitor (C_{MOD} Pin) 14
3.1.3 Flanking Sensor Suppression (FSS Pin) 15
3.1.4 Output Select (OUT_SEL Pin) 15
3.1.5 Key Scan Interface 15
3.1.6 Truth Table Output 17
3.1.7 Encoded 4-bit Output 18
3.1.8 Buzzer Signal Output (Buzzer Pin) 18
3.1.9 Interrupt Line (INT Pin) 18
3.1.10 Button Auto Reset (ARST Pin) 18
3.1.11 Button Scan Rate (SCAN Pin) 19
3.1.12 Sensitivity Control (SENSITIVITY Pin) 19
3.1.13 System Diagnostics 19
3.1.14 Serial Debug Data Out (DEBUG Pin) 22
3.2 Design Toolbox 25
3.2.1 General Layout Guidelines 25
3.2.2 Layout Estimator 26
3.2.3 C_P, Power Consumption and Response Time Calculator 27
3.2.4 Design Validation 28

4. Electrical and Mechanical Design Considerations 30
4.1 Overlay Selection 30
4.1.1 Bonding Overlay to PCB 31
4.2 ESD Protection 31
4.2.1 Prevent 31
4.2.2 Redirect 31
4.2.3 Clamp 31
4.3 Electromagnetic Compatibility (EMC) Considerations 32
4.3.1 Radiated Interference 32
4.3.2 Conducted Immunity and Emissions 32
4.4 Scan Line Waveform Requirements (Key Scan Interface) 32
4.5 PCB Layout Guidelines 33

5. Low-Power Design Considerations 34
5.1 System Design Considerations 34
5.2 Calculating Average Power 34
5.2.1 Button Scan Rate (T_R) 34
5.2.2 Scan Time (T_S) 35
5.2.3 Average Current in NO TOUCH state (I_{AVE_NT}) 35
5.2.4 Average Current in TOUCH state (I_{AVE_T}) 35
5.2.5 Average Use Current (I_{AVE_U}) 36
5.2.6 Average Current (I_{AVE}) 36
5.2.7 Average Power (P_{AVE}) 36
5.2.8 Example Calculation 36
5.3 Sleep Modes 37
5.3.1 Low-Power Sleep Mode 37
5.3.2 Deep Sleep Mode 37

6. Resources 38
6.1 Website 38
6.2 Datasheet 38
6.3 Kits 38
6.4 Design Toolbox 38
6.5 Sample Board Files 38
6.6 MultiChart 38
6.7 Design Support 38

7. Appendix 39
7.1 Schematic Examples 39
7.1.1 Schematic 1: 16 Buttons with Key Scan Output Mode 39
7.1.2 Schematic 2: 16 Buttons with Truth Table Output Mode 41

Glossary 43

Revision History 49

1. Introduction

1.1 Abstract

This document describes how to implement capacitive sensing functionality using Cypress's CapSense® Express™ CY8CMBR2016 device. The following topics are covered in this guide:

- Features of the CY8CMBR2016
- CapSense principles of operation
- Configuration options of the CY8CMBR2016 device
- Using the Design Toolbox with the CY8CMBR2016
- System electrical and mechanical design considerations for the CY8CMBR2016
- Low Power Design Considerations for the CY8CMBR2016
- Additional resources and support for designing CapSense into your system

1.2 Cypress's CapSense Documentation Ecosystem

Figure 1-1 and Table 1-1 summarize the Cypress CapSense documentation ecosystem. These resources allow implementers to quickly access the information needed to successfully complete a CapSense product design. Figure 1-1 shows the typical flow of a product design cycle with capacitive sensing; the information in this guide is most pertinent to the topics highlighted in green. Table 1-1 provides links to the supporting documents for each of the numbered tasks in Figure 1-1.

Figure 1-1. Typical CapSense Product Design Flow

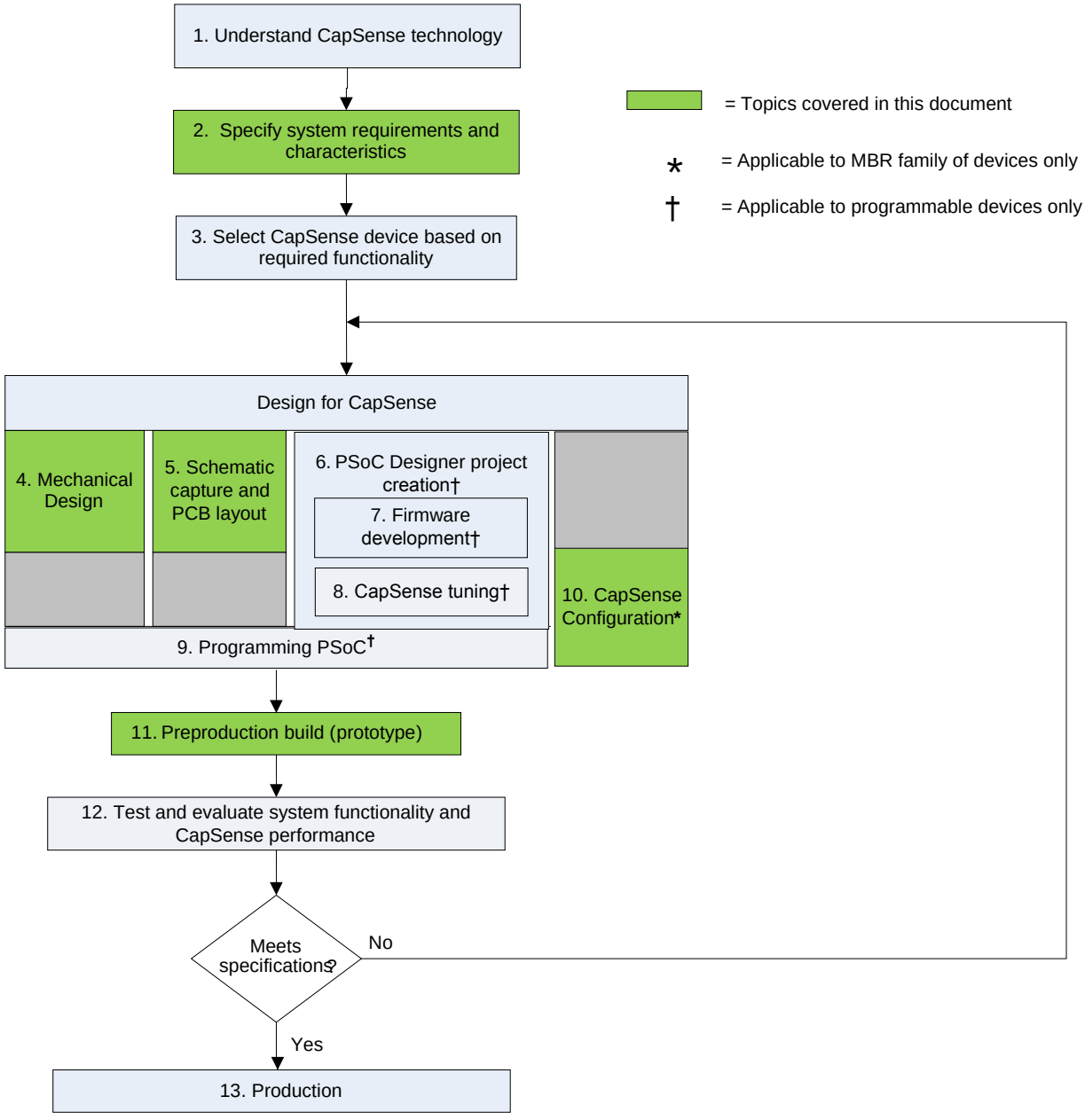


Table 1-1. Cypress Documents Supporting Numbered Design Tasks of Figure 1-1

Numbered Design Task of Figure 1-1	Supporting Cypress CapSense®Documentation
1	Getting Started with CapSense
2	CY8CMBR2016 Device Datasheet
3	Getting Started with CapSense
4	This document
5	This document
6	Not applicable for CY8CMBR2016
7	Not applicable for CY8CMBR2016
8	Not applicable for CY8CMBR2016
9	Not applicable for CY8CMBR2016
10	This document
11	This document

1.3 CY8CMBR2016 CapSense® Express Device Features

Cypress's low-power CapSense controller can easily add capacitive touch sensing to your user interface. The device's features include:

- Hardware Configurable Matrix CapSense Controller
  - Does not require software tools or programming
  - 16 buttons can be configured individually or as a matrix
  - Supports 3x4 and 4x4 matrix configurations
- Matrix Host Interface Communication
  - Industry standard host interface protocols reuse existing host processor firmware
    - Key Scan Interface
    - Truth Table Interface
  - Encoded GPO Interface – minimizes number of pins required
- SmartSense™ Auto-Tuning
  - Maintains optimal button performance even in noisy environment
  - CapSense parameters dynamically set in runtime
  - Wide parasitic capacitance (C_P) range (5—40 pF)
  - Saves time and effort in device tuning
- Noise Immunity
  - High sensitivity, low noise capacitive sensing algorithm
  - Strong immunity to RF and AC noise
  - Low radiated noise emission
- System Diagnostics
  - Button shorts
  - Improper value of modulating capacitor (C_{MOD})
  - Parasitic capacitance (C_P) out of range

- Advanced Features
  - Flanking Sensor Suppression (FSS) provides robust sensing even with closely spaced buttons
  - Buzzer Signal Output
  - Configurable sensitivity for all buttons
  - Interrupt line to host to indicate any CapSense button status change
  - Serial debug data output
    - Simplifies production-line testing and system debug
- Wide operating range
  - 1.71—5.5 V
  - Ideal for both regulated and unregulated battery applications
- Low power consumption
  - Supply current in run mode as low as 20 µA[1] per button
  - Deep sleep current: 100 nA
- Industrial temperature range: –40 °C to + 85 °C
- 48-pin QFN package (6 × 6 × 0.6 mm)

1.4 Document Conventions

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

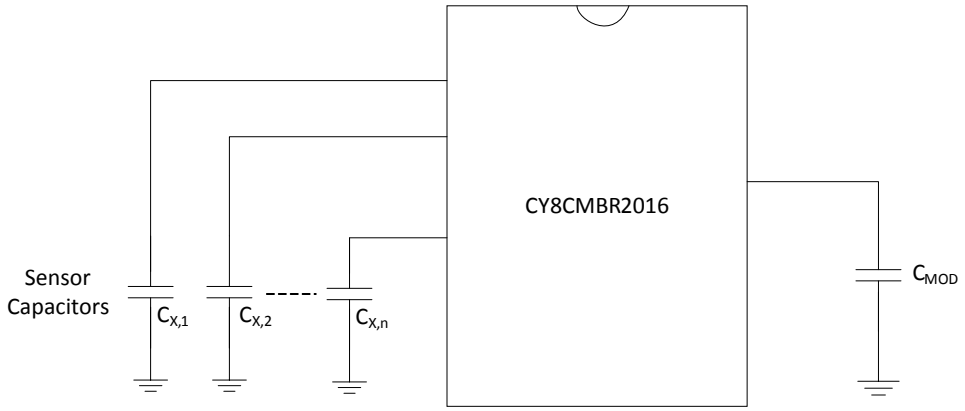
[1] Power consumption calculated with 250 ms scan time, 2% touch time and Cp of each button < 19 pF

2. CapSense Technology

2.1 CapSense Fundamentals

CapSense is a touch sensing technology that works by measuring the capacitance of each sensor input pin on the CapSense controller. The total capacitance on each of the sensor pins can be modeled as equivalent lumped capacitors with values of $C_{X,1}$ through $C_{X,n}$ as shown in Figure 2-1. Circuitry internal to the CY8CMBR2016 device converts the magnitude of each C_X into a digital code that is stored for post processing. A modulator capacitor, C_{MOD}, is used by the CapSense controller's internal circuitry and will be discussed in more detail in Capacitive Sensing Method.

Figure 2-1. CapSense Implementation in a CY8CMBR2016 Device



Each sensor input pin is connected to a sensor pad by traces, vias, or both as necessary. A nonconductive overlay is required to cover the sensor pad and constitutes the product's touch interface. When a finger comes into contact with the overlay, the conductivity and mass of the body effectively introduces a grounded conductive plane parallel to the sensor pad. This is represented in Figure 2-2. This arrangement constitutes a parallel plate capacitor, whose capacitance is given by:

$$C_F = \frac{\varepsilon_0 \varepsilon_r A}{D}$$ Equation 1

Where:

C_F = The capacitance affected by a finger in contact with the overlay over a sensor

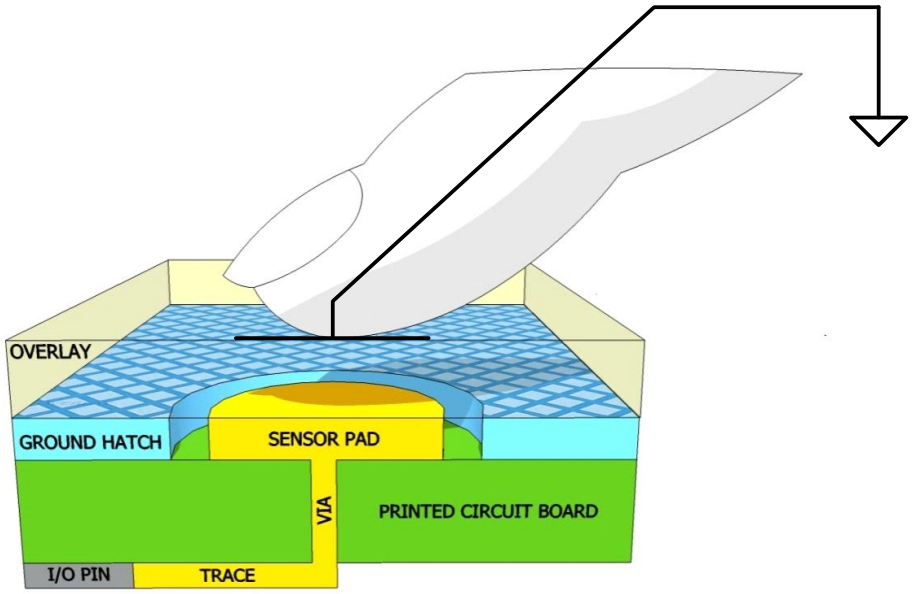
ε_0 = Free space permittivity

ε_r = Dielectric constant (relative permittivity) of overlay

A = Area of finger and sensor pad overlap

D = Overlay thickness

Figure 2-2. Section of Typical CapSense PCB with the Sensor Being Activated by a Finger



In addition to the parallel plate capacitance, a finger in contact with the overlay causes electric field fringing between itself and other conductors in the immediate vicinity. Typically, the effect of these fringing fields is minor, and it can usually be ignored.

Even without a finger touching the overlay, the sensor input pin has some parasitic capacitance (C_P). C_P results from the combination of the CapSense controller internal parasitic and electric field coupling among the sensor pad, traces, and vias, and other conductors in the system such as ground plane, other traces, any metal in the product's chassis or enclosure, and so on. The CapSense controller measures the total capacitance (C_X) connected to a sensor pin.

When a finger is not touching a sensor:

$$C_X = C_P \quad \text{Equation 2}$$

With a finger on the sensor, C_X equals the sum of C_P and C_F:

$$C_X = C_P + C_F \quad \text{Equation 3}$$

In general, C_P is an order of magnitude greater than C_F. C_P usually ranges from 10—20 pF, but in extreme cases can be as high as 40 pF. C_F usually ranges from 0.1—0.4 pF.

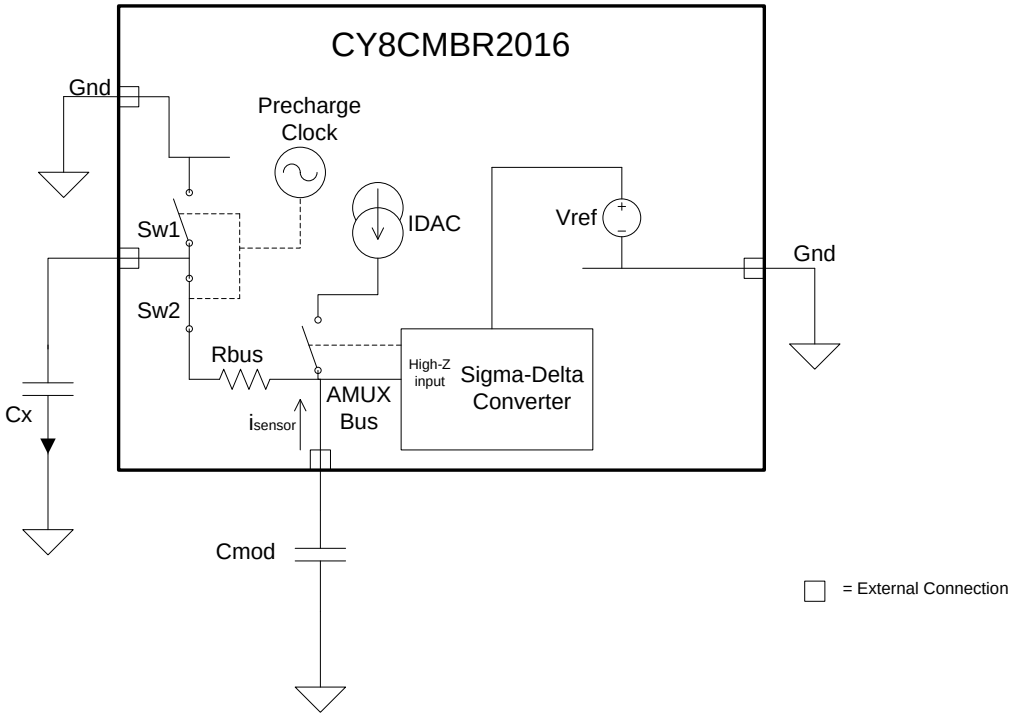
2.2 Capacitive Sensing Method

CY8CMBR2016 device supports CapSense Sigma Delta (CSD) with SmartSense Auto-Tuning for converting sensor capacitance (C_X) into digital counts. The method is described in the following sections.

2.2.1 CapSense Sigma-Delta (CSD)

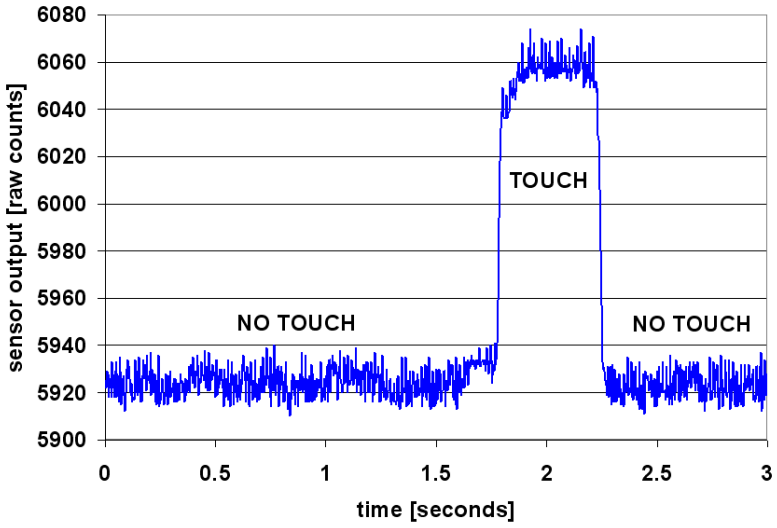
The CSD method in the CY8CMBR2016 device incorporates C_X into a switched capacitor circuit as shown in Figure 2-3. C_X is alternatively connected to Gnd and the AMUX bus by the non-overlapping switches Sw1 and Sw2. Sw1 and Sw2 are driven by the Precharge clock to bleed a current, i_{sensor} from the AMUX bus. The magnitude of i_{sensor} is directly proportional to the magnitude of C_X. The sigma-delta converter samples the AMUX bus voltage and generates a modulating bit stream that controls the constant current source, IDAC. The IDAC charges AMUX such that the average AMUX bus voltage is maintained at Vref. The sensor bleeds off the charge i_{sensor} from C_{MOD}, which, in combination with Rbus, forms a low-pass filter that attenuates precharge switching transients at the sigma-delta converter input.

Figure 2-3. CSD Block Diagram



To maintain the AMUX bus voltage at Vref, the sigma-delta converter matches IDAC to i_{sensor} by controlling the bit stream duty cycle. The sigma-delta converter stores the bit stream over the duration of a sensor scan, and the accumulated result is a digital output, raw count, which is directly proportional to C_X. This raw count is interpreted by high-level algorithms to resolve the sensor state. Figure 2-4 plots the CSD raw counts from a number of consecutive scans during which the sensor is touched and then released by a finger. As explained in CapSense Fundamentals, the finger touch causes C_X to increase by C_F, which in turn causes raw counts to increase proportionally. By comparing the shift in steady state raw count level to a predetermined threshold, the high-level algorithms can determine whether the sensor is in an ON (Touch) or OFF (No Touch) state. To learn more about Raw Counts, Finger Threshold, and Signal-to-Noise Ratio (SNR), refer to Getting Started with CapSense.

Figure 2-4. CSD Raw Counts during a Finger Touch



2.3 SmartSense Auto-Tuning

Tuning the touch-sensing user interface is critical for proper system operation and a pleasant user experience. Unfortunately, tuning is time-consuming because it is an iterative process. In a typical development cycle, the interface is tuned in the initial design phase, during system integration, and before production ramp. SmartSense Auto-Tuning was developed to simplify the user interface development cycle. It is easy to use and reduces design cycle time by eliminating manual tuning during the prototype and manufacturing stages. SmartSense Auto-Tuning tunes each CapSense button automatically at power up and maintains optimum button performance during runtime. SmartSense Auto-Tuning adapts for manufacturing variation in PCBs and overlays and automatically tunes out noise from sources such as LCD inverters, AC lines, and switch-mode power supplies.

2.3.1 Process Variation

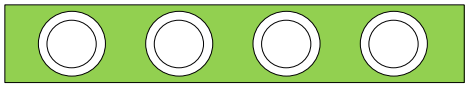
The CY8CMBR2010 device's SmartSense Auto-Tuning is designed to work with C_P values in the range of 5—40 pF. The sensitivity parameter for each button is set automatically, based on its characteristics. This parameter improves yield in mass production because every button maintains a consistent response regardless of C_P variation between the buttons. C_P can vary due to PCB layout and trace length, PCB manufacturing process variation, or vendor-to-vendor PCB variation within a multi-sourced supply chain. The sensitivity of a button depends on C_P; higher C_P values decrease sensitivity, resulting in decreased finger touch signal amplitude. A change in C_P can result in a button becoming too sensitive, not sensitive enough, or non-operational. When this happens, you must retune the system and, in some cases, re-qualify the user interface subsystem. The CY8CMBR2016 device solves these issues by incorporating SmartSense and configurable sensitivity based on sensor size and C_p.

SmartSense Auto-Tuning makes platform designs possible. For example, consider the capacitive touch sensing multimedia keys on a laptop computer. The parasitic capacitance of the CapSense buttons can vary in different models of the same platform design depending on the size of the laptop and the keyboard layout. In this example, a wide-screen laptop model would have larger spaces between the buttons than a standard-screen model. Therefore, a wide-screen model would have longer traces between each button and the CapSense controller, which would result in higher C_P values. Though the buttons' functionality is identical for all of the laptop models, the buttons must be tuned for each model. SmartSense Auto-Tuning lets you do platform designs using the recommended practices shown in the PCB Layout in Getting Started with CapSense.

Figure 2-5 Design of Laptop Multimedia Keys for a 21-Inch Model



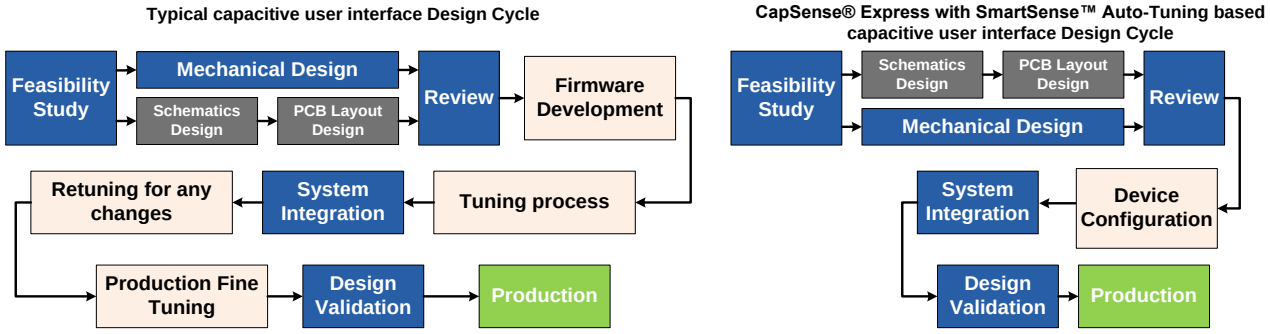
Figure 2-6 Design of Laptop Multimedia Keys for a 15-Inch Model with Identical Functionality and Button Size



2.3.2 Reduced Design Cycle Time

When you design a capacitive button interface, the most time-consuming tasks are firmware development, layout, and button tuning. With a typical touch-sensing controller, the button must be retuned when the same design is ported to different models or when the mechanical dimensions change in the PCB or the button PCB layout. A design with SmartSense Auto-Tuning meets these challenges because it does not require firmware development, manual tuning, or retuning. In addition, SmartSense Auto-Tuning speeds up a typical design cycle. Figure 2-7 compares the design cycles of a typical touch-sensing controller and a SmartSense Auto-Tuning-based design.

Figure 2-7. Typical Capacitive Interface Design Cycle Comparison



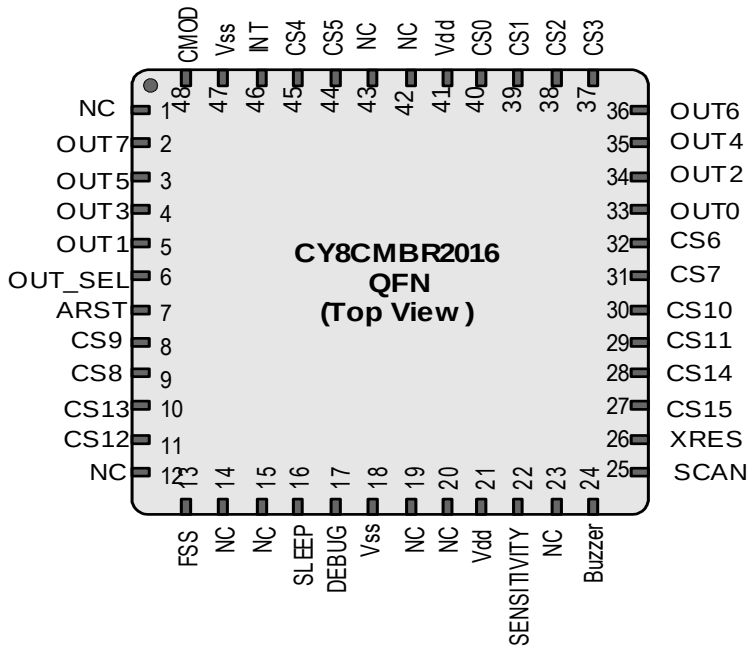
3. CapSense Schematic Design

3.1 CY8CMBR2016 Configuration Options

Cypress's CY8CMBR2016 allows you to implement capacitive touch sensing using only hardware. The CY8CMBR2016 provides three different host interface communication modes. Industry standard Key Scan Interface and Truth Table Output reuse existing host processor firmware, allowing you to easily convert mechanical buttons to CapSense buttons. The Encoded GPO Interface with a 4-bit output minimizes the number of pins required for a button output. These three outputs are configurable, providing a wide variety of uses for our device across multiple applications.

This section provides an overview of the CapSense controller pins and how to configure them.

Figure 3-1. CY8CMBR2016 Pin Diagram



3.1.1 CapSense Buttons (CSx Pins)

The CY8CMBR2016 controller has 16 capacitive sense inputs, CS0—CS15. Each capacitive button requires a connection to one of the capacitive sense inputs. You must ground all unused CapSense (CSx) inputs pins.

3.1.2 Modulation Capacitor (C_{MOD} Pin)

Connect a 2.2 nF (±10%) capacitor to the C_{MOD} pin.

3.1.3 Flanking Sensor Suppression (FSS Pin)

FSS allows only one CSx to be in the TOUCH state at a time. This means you can distinguish TOUCH states for closely spaced buttons. If a finger contacts multiple CSx buttons, only the first one to sense a TOUCH state will turn ON.

FSS also is useful when nearby buttons can produce opposite effects such as an interface with two buttons for brightness control (UP or DOWN).

To enable the feature, connect the FSS pin to V_{DD}. Connect the pin to ground to disable it.

Figure 3-2. FSS When Only One Button is Touched



No button is ON prior to the touch

CS1 is reported as ON upon touch

Figure 3-3. FSS When Multiple Buttons are Touched With One Button ON Previously



CS1 is touched; reported ON

CS2 also touched along with CS1; only CS1 is reported ON

3.1.4 Output Select (OUT_SEL Pin)

CY8CMBR2016 provides three different host interface communication modes:

1. Key Scan Interface
2. Truth Table Output
3. Encoded 4-bit Output

The OUT_SEL pin selects the host interface mode as described in Table 3-1

Table 3-1. OUT_Select Configuration

OUT_SEL pin	Host Interface Mode
Ground	Truth Table Output
1.5 KΩ (±5%) resistor to ground	Encoded 4-bit Output
V_{DD} / Floating	Key Scan Interface

3.1.5 Key Scan Interface

Refer to Table 3-1 for how to select the host interface mode.

Key Scan Interface mimics legacy mechanical keypads with four scan lines (I/P) and four read lines (O/P). The host drives the device scan lines SCAN0 to SCAN3 in a time slot manner. Each scan line is driven for a time period, T, where:

T = 30 µs, if debug is disabled

T = 130 µs, if debug is enabled

Refer to Figure 3-5 for scan line waveforms.

The device reads the scan lines and updates the read lines based on the button status. The host reads the device read lines to know the status of corresponding buttons.

Key Scan Interface is a ‘Plug' n 'Play' replacement for mechanical keypads. Refer to Figure 3-4 for an example Key Scan Interface.

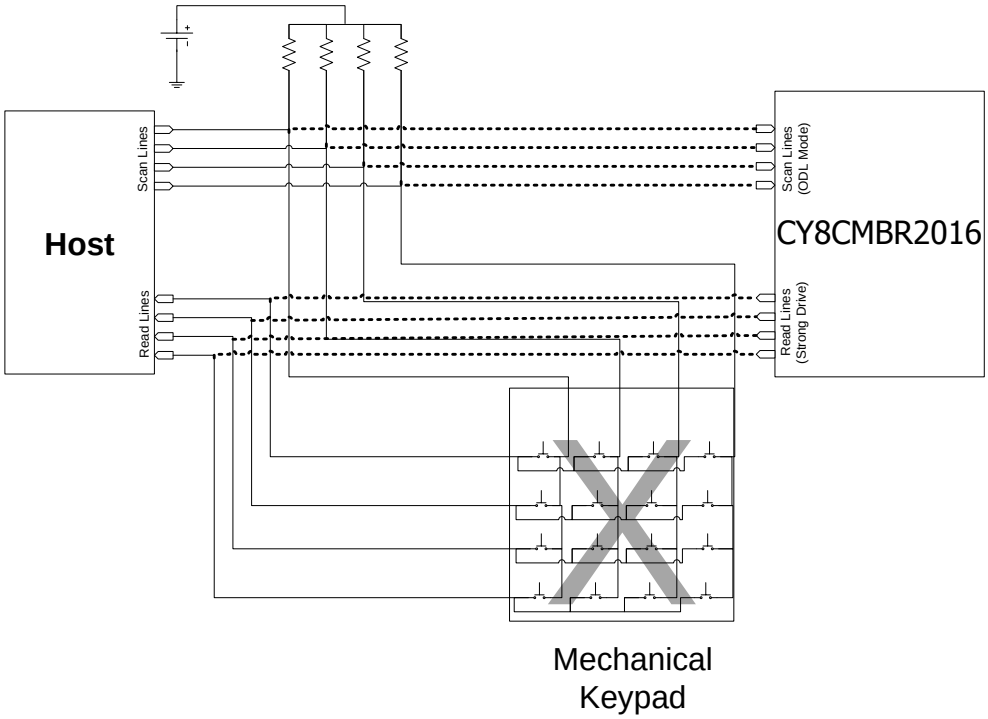
When buttons are disabled or found to be invalid, Key Scan Interface helps identify the disabled or invalid buttons using the scan and read lines. When the scan lines are not used, they should be connected to VDD.

In this output mode, OUT0 to OUT3 act as the scan lines and OUT4 to OUT7 act as the read lines.

Table 3-2. Key Scan Interface Selection Based on Number of Buttons

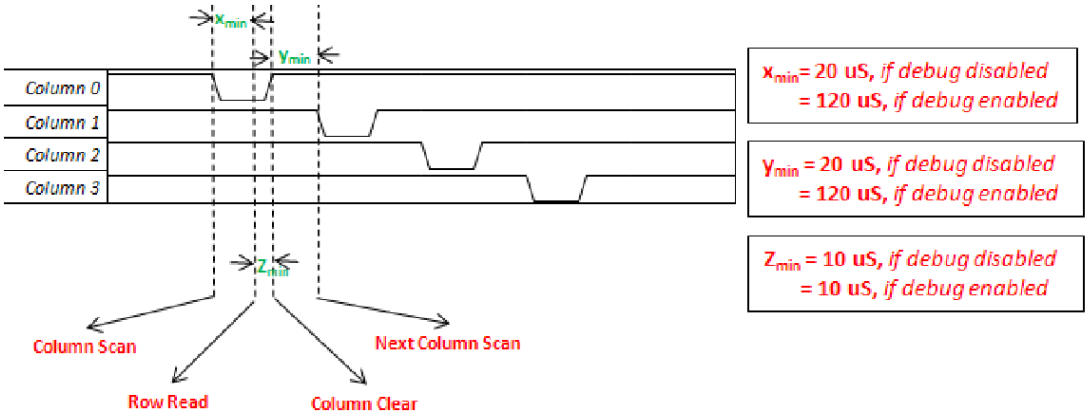
	SCAN x READ Lines	SCAN Lines
Buttons >12	4x4	OUT0 to OUT3
12 ≥ Buttons > 8	3x4	OUT0 to OUT2
8 ≥ Buttons > 4	2x4	OUT0 to OUT1
Buttons ≤ 4	1x4	OUT0

Figure 3-4. Example of Key Scan Interface (4x4 Matrix)[1]



[1] Scan Lines in Open Drain Low (ODL) Mode

Figure 3-5. Key Scan Line Waveforms



3.1.6 Truth Table Output

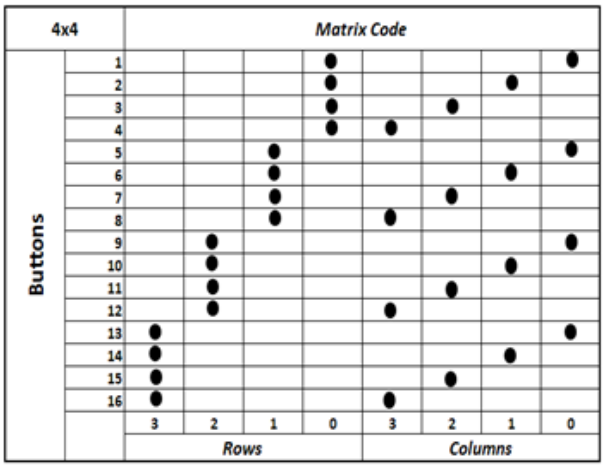
Refer to Table 3-1 for how to select the host interface mode.

Truth Table Output is a matrix output interface. All of the output pins are divided into ROW and COLUMN lines. For this type of output interface, only one button can be reported at a time, therefore, FSS should be enabled when using Truth Table Output.

Button status is reported in an encoded ROW/COLUMN manner as shown in Figure 3-6. Each button has its own ROW-COLUMN code. OUT_4 to OUT_7 form the ROW lines and OUT_0 to OUT_3 form the COLUMN lines.

Truth Table Output is easy to integrate into a system requiring a simple interface with single key press requirement.

Figure 3-6. Button Status Reporting



3.1.7 Encoded 4-bit Output

Refer to Table 3-1 for how to select the host interface mode.

In an Encoded 4-bit Output, only four pins are needed to report a button touch, even for 16 buttons. Each button has a unique code which is seen on the output pins, when the button is touched. Only one button can be reported at a time using this interface, therefore, FSS should be enabled when using Encoded 4-bit Output. Table 3-3 defines the output given.

Table 3-3. Encoded Output

Key Press Detected	Encoded Output [3:0]	Interrupt Time
Key #1	0000	1
Key #2	0001	1
Key #3	0010	1
Key #4	0011	1
…	…	1
Key #16	1111	1
No keys pressed	XXXX	0

3.1.8 Buzzer Signal Output (Buzzer Pin)

Buzzer Signal Output can be used to drive a p-type transistor that drives a buzzer, or to directly drive a DC buzzer with sink current up to 10 mA.

3.1.9 Interrupt Line (INT Pin)

An interrupt line to the host controller is provided. The line goes high when one or more buttons are touched, and remains high until those buttons are released.

The Interrupt Line can be used as a latch input at the host side to read the OUT lines. It can also be used as an interrupt line for the host controller to read the OUT lines.

3.1.10 Button Auto Reset (ARST Pin)

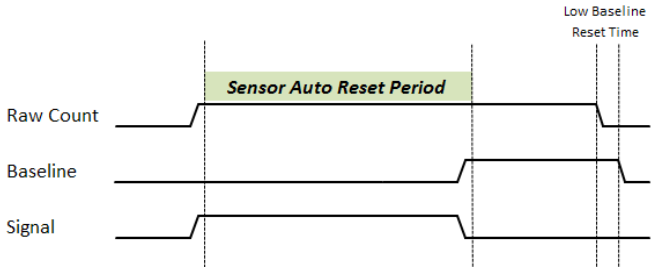
Button Auto Reset determines the maximum time a button is considered to be ON when CSx is continuously touched. This feature prevents a button from getting stuck if a conducting object is placed too close to it. The Button Auto Reset period is defined in Table 3-4 and shown Figure 3-7

After the output is turned off due to Button Auto Reset and the button is released, you should not touch the button for 440 ms.

Table 3-4. ARST Pin Configuration

ARST pin	Button Auto Reset Period
Ground	5 sec
1.5 KΩ (±5%) resistor to ground	20 sec
5 KΩ (±5%) resistor to ground	40 sec
V_{DD} / Floating	No limit

Figure 3-7.Button Auto Reset



3.1.11 Button Scan Rate (SCAN Pin)

Button Scan Rate defines the rate at which the device scans all the buttons and then sleeps, in the Low Power Sleep mode. Button Scan Rate is defined in Table 3-5.

Device power consumption is dependent on the Button Scan Rate. A higher Button Scan Rate reduces power consumption. For more details on Low Power Sleep, refer to Sleep Modes.

Table 3-5. SCAN Pin Configuration

SCAN pin	Button Scan rate
Ground	Low, 250 ms
1.5 kΩ (±5%) resistor to ground	Medium, 150 ms
5 kΩ (±5%) resistor to ground	High, 40 ms
V_{DD} / Floating	Continuous

3.1.12 Sensitivity Control (SENSITIVITY Pin)

A button's sensitivity setting determines the minimum finger capacitance (C_F) required to turn ON a button. The following factors affect the button's sensitivity:

1. Overlay thickness: For a thicker overlay, the button needs to be more sensitive to a touch. Therefore, the thicker the overlay, the higher the sensitivity requirement.
2. System noise: As system noise increases, sensitivity needs to be reduced to avoid false button triggers.
3. Form factor of the design: A relatively large button size is required to support low sensitivity (Higher C_F). For a small button diameter, the sensitivity needs to be high.
4. Power consumption: Power consumption increases for high sensitivity buttons. For low power consumption needs, the sensitivity needs to be low.

The different sensitivity settings available are "High", "Medium" and "Low". Sensitivity is defined in Table 3-6.

Table 3-6. SENSITIVITY Pin Configuration

SENSITIVITY pin	Sensitivity
Ground	Low
1.5 kΩ (±5%) resistor to ground	Medium
V_{DD} / Floating	High

3.1.13 System Diagnostics

A built-in Power-on Self Test (POST) mechanism performs five tests at power-on reset (POR), which can be useful in production testing.

If any button fails any of the five tests, the corresponding bit is set in the button mask and the System Diagnostics data is sent out serially on OUT6, synchronized with a 2 kHz clock on OUT7.

If no clock is sensed on OUT7 for 300 ms after power ON, all of the buttons have passed System Diagnostics.

If a clock is sensed, then beginning with the first falling edge of the clock, each button takes up one clock slot. A HIGH output on OUT6 during a falling edge on OUT7 indicates the button being reported during that clock slot has failed one or more tests. The clock stops after the last failure has been reported.

For example, if CS1, CS3, and CS5 are shorted to ground, the System Diagnostics data is transmitted as shown in Figure 3-8. CS1's failure is marked by a HIGH on OUT6 in the 0.5—1 ms slot. CS3's failure is marked by a HIGH on OUT6 in the 1.5—2 ms slot. CS5's failure is marked in the 2.5—3 ms slot. The clock output stops after report CS5.

In another example, Figure 3-9 shows a scenario where CS1, CS3, and CS15 fail.

Figure 3-8. System Diagnostics Output for CS1, CS3, and CS5

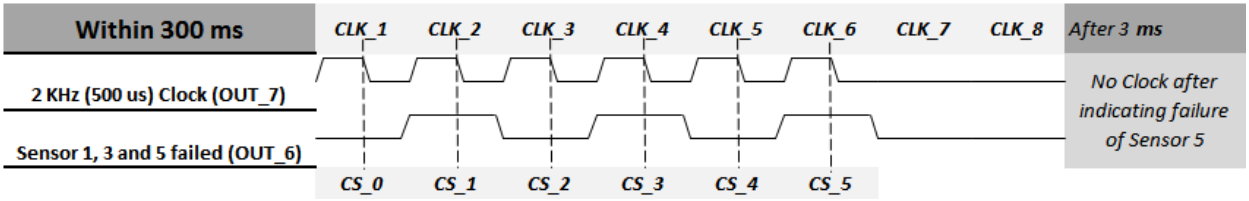
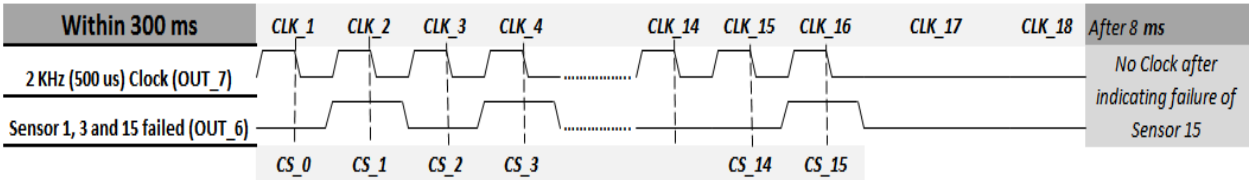


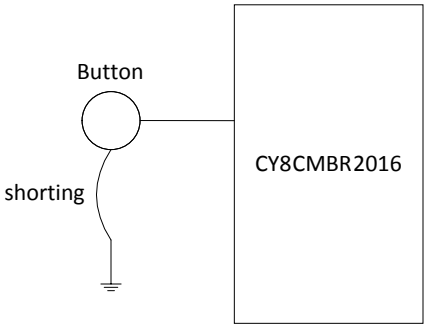
Figure 3-9. System Diagnostics Output for CS1, CS3, and CS15



3.1.13.1 Button Shorted to Ground

If a button is disabled/shorted to ground the corresponding bit in the button mask is set and the System Diagnostics data is sent.

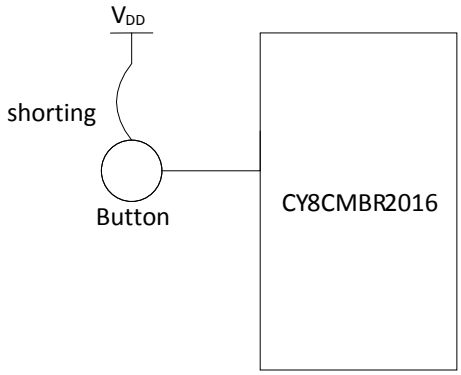
Figure 3-10. Button Short to Ground



3.1.13.2 Button Shorted to V_{DD}

If any button is shorted to V_{DD} it is disabled and the corresponding bit in the button mask is set, and the System Diagnostics signal is sent.

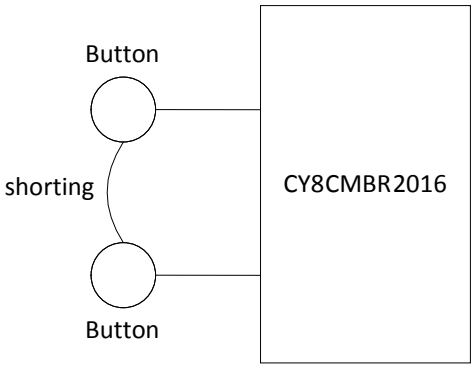
Figure 3-11. Button Short to V_{DD}



3.1.13.3 Button to Button Short

Any buttons that are shorted together are disabled and the corresponding bit in the button mask is set and the System Diagnostics signal is sent.

Figure 3-12. Button to Button Short



3.1.13.4 Improper Value of C_{MOD}

Recommended value of C_{MOD} is 2—2.4 nF.

If the value of C_{MOD} is found to be less than 1 nF or greater than 4 nF, all buttons are disabled and the status output is logic high for all clock slots.

3.1.13.5 Button C_P > 40 pF

If any button's C_P exceeds 40 pF, it is disabled and the corresponding bit in button mask is set and the System Diagnostics signal is sent.

3.1.14 Serial Debug Data Out (DEBUG Pin)

Serial Debug Data reports firmware revision, CapSense status, raw count, baseline, difference count, and parasitic capacitance for all buttons. To enable this feature, the DEBUG pin is pulled down with a 5.6 KΩ resistor.

The Cypress MultiChart Tool can be used to view the debug data for each button. The Serial Debug Data is sent by the device in the order according to Table 3-7. The MultiChart tool arranges the data in the format as shown in Table 3-8.

Table 3-7. Serial Debug Data Output sent by CY8CMBR2016

Byte	Data	Notes
0	0x0D	Dummy data for MultiChart
1	0x0A	
2	CS0_RawCount	CS0 Raw Counts, unsigned 16-bit integer
3		
4	CS1_RawCount	CS1 Raw Counts, unsigned 16-bit integer
5		
6	CS2_RawCount	CS2 Raw Counts, unsigned 16-bit integer
7		
---	---	---
32	CS15_RawCount	CS15 Raw Counts, unsigned 16-bit integer
33		
34	0x00	–-
35	FW_REV	Firmware Revision
36	0x00	–-
37	CS0_Cp	CS0 parasitic capacitance (pF) in Hex
38	0x00	–-
39	CS1_Cp	CS1 parasitic capacitance (pF) in Hex
40	0x00	–-
41	CS2_Cp	CS2 parasitic capacitance (pF) in Hex
42	0x00	–-
43	CS3_Cp	CS3 parasitic capacitance (pF) in Hex
44	0x00	–-
45	CS4_Cp	CS4 parasitic capacitance (pF) in Hex
46	CS0_Baseline	CS0 Baseline, unsigned 16-bit integer
47		
48	CS1_Baseline	CS1 Baseline, unsigned 16-bit integer
49		
50	CS2_Baseline	CS2 Baseline, unsigned 16-bit integer
51		
---	---	---
76	CS15_Baseline	CS15 Baseline, unsigned 16-bit integer
77		

Byte	Data	Notes
78	CS_Status	CapSense status, unsigned 16-bit integer
79		
80	0x00	–
81	CS5_Cp	CS5 parasitic capacitance (pF) in Hex
82	0x00	–
83	CS6_Cp	CS6 parasitic capacitance (pF) in Hex
84	0x00	–
85	CS7_Cp	CS7 parasitic capacitance (pF) in Hex
86	0x00	–
87	CS8_Cp	CS8 parasitic capacitance (pF) in Hex
88	0x00	–
89	CS9_Cp	CS9 parasitic capacitance (pF) in Hex
90	CS0_DiffCount	CS0 Difference Counts, unsigned 16-bit integer
91		
92	CS1_DiffCount	CS1 Difference Counts, unsigned 16-bit integer
93		
94	CS2_DiffCount	CS2 Difference Counts, unsigned 16-bit integer
95		
---	---	---
122	CS15_DiffCount	CS15 Difference Counts, unsigned 16-bit integer
123		
124	0x00	–
125	CS10_Cp	CS10 parasitic capacitance (pF) in Hex
126	0x00	–
127	CS11_Cp	CS11 parasitic capacitance (pF) in Hex
128	0x00	–
129	CS12_Cp	CS12 parasitic capacitance (pF) in Hex
130	0x00	–
131	CS13_Cp	CS13 parasitic capacitance (pF) in Hex
132	0x00	–
133	CS14_Cp	CS14 parasitic capacitance (pF) in Hex
134	0x00	–
135	CS15_Cp	CS15 parasitic capacitance (pF) in Hex
136	0x00	Dummy data for MultiChart
137	0xFF	
138	0xFF	

Table 3-8. Serial Debug Data Arranged in MultiChart

#	Raw Count Array		Baseline Array		Signal array	
	MSB	LSB	MSB	LSB	MSB	LSB
0	CS0_RawCount		CS0_Baseline		CS0_DiffCount	
1	CS1_RawCount		CS1_Baseline		CS1_DiffCount	
2	CS2_RawCount		CS2_Baseline		CS2_DiffCount	
3	CS3_RawCount		CS3_Baseline		CS3_DiffCount	
4	CS4_RawCount		CS4_Baseline		CS4_DiffCount	
5	CS5_RawCount		CS5_Baseline		CS5_DiffCount	
6	CS6_RawCount		CS6_Baseline		CS6_DiffCount	
7	CS7_RawCount		CS7_Baseline		CS7_DiffCount	
8	CS8_RawCount		CS8_Baseline		CS8_DiffCount	
9	CS9_RawCount		CS9_Baseline		CS9_DiffCount	
10	CS10_RawCount		CS10_Baseline		CS10_DiffCount	
11	CS11_RawCount		CS11_Baseline		CS11_DiffCount	
12	CS12_RawCount		CS12_Baseline		CS12_DiffCount	
13	CS13_RawCount		CS13_Baseline		CS13_DiffCount	
14	CS14_RawCount		CS14_Baseline		CS14_DiffCount	
15	CS15_RawCount		CS15_Baseline		CS15_DiffCount	
16	0x00	FW_Rev	CS_Status		0x00	CS10_Cp
17	0x00	CS0_Cp	0x00	CS5_Cp	0x00	CS11_Cp
18	0x00	CS1_Cp	0x00	CS6_Cp	0x00	CS12_Cp
19	0x00	CS2_Cp	0x00	CS7_Cp	0x00	CS13_Cp
20	0x00	CS3_Cp	0x00	CS8_Cp	0x00	CS14_Cp
21	0x00	CS4_Cp	0x00	CS9_Cp	0x00	CS15_Cp

3.2 Design Toolbox

The Design Toolbox helps you to design a CY8CMBR2016 CapSense solution. It offers basic information about the board layout and feature settings and recommends whether the design is fit for mass production.

3.2.1 General Layout Guidelines

Table 3-9 summarizes the layout guidelines for the CY8CMBR2016. These guidelines are discussed in Electrical and Mechanical Design Considerations of this guide. For a thorough treatment of this material, see Getting Started with CapSense, Section 3.7.

Table 3-9. Design Layout Recommendations

Sl. No.	Category	Min	Max	Recommendations/Remarks
1.	Button Shape			Solid round pattern, Round with LED hole, rectangle with round corners
2.	Button Size	5mm	15mm	Given in Layout Estimator sheet
3.	Button-Button spacing	equal to button ground clearance		8 mm
4.	Button Ground Clearance	0.5mm	2mm	Given in Layout Estimator sheet
5.	Ground Flood - Top layer			Hatched ground 7 mil trace and 45 mil grid (15% filling)
6.	Ground Flood - Bottom layer			Hatched ground 7 mil trace and 70 mil grid (10% filling)
7.	Trace Length from Sensor pad to device pin		450	Given in Layout Estimator sheet
8.	Trace Width	0.17 mm	0.20 mm	0.17 mm (7 mil)
9.	Trace Routing			Traces should be routed on the non sensor side. If any non CapSense trace crosses CapSense trace, ensure that intersection is orthogonal.
10.	Via Position for the sensors			Via should be placed near the edge of the button to reduce trace length thereby increasing sensitivity.
11.	Via Hole Size for sensor traces			10 mil
12.	No. of via on sensor trace	1	2	1
13.	CapSense series resistor placement		10mm	Place CapSense series resistors close to the device for noise suppression .CapSense resistors have highest priority compared to LED ressitors place them first.
14.	Distance between any CapSense trace to Ground Flood	10 mil	20 mil	20 mil
15.	Device placement			Mount the Device on the layer opposite to sensor. The CapSense trace length between the Device and sensors should be minimum (see trace length above)
16.	Placement of Components in two layer PCB			Top layer-Sensors and bottom layer-device, other components and traces.
17.	Placement of Components in four layer PCB			Top layer-Sensors, 2nd Layer – CapSense traces & Vdd and avoid the Vdd traces below the sensors, 3rd Layer-hatched ground, Bottom layer- device other components and non CapSense traces
18.	Overlay Thickness	0 mm	5mm	Use layout Estimator sheet to decide on overlay, given maximum limit is for plastic overlay.
19.	Overlay material			Should be non-conductive material. Glass, ABS Plastic, Formica, wood etc. No air gap should be there between PCB and overlay. Use adhesive to stick the PCB and overlay.
20.	Overlay Adhesives			Adhesive should be non conductive and dielectrically homogenous. 467MP and 468MP adhesives made by 3M are recommended.
21.	LED Back Lighting			Cut a hole in the sensor pad and use rear mountable LEDs.
22.	Board Thickness			Standard board thickness for CapSense FR4 based designs is 1.6 mm.

3.2.2 Layout Estimator

The Layout Estimator provides the minimum button size and maximum trace length recommendation based on the intended end-system requirements and industrial design. The inputs include the overlay material, overlay thickness, and trace capacitance of circuit board material. Figure 3-13, Table B, lists the dielectric constants for different overlay materials as well as the trace capacitance per unit length for different PCBs. Table A calculates the minimum button diameter and maximum trace length for the design, based on three system noise conditions. “Low”, “Medium”, and “High” noise conditions are relative figures of merit to help you with button development. Noise conditions can vary from button to button based on the end-system environment. If the noise conditions are unknown, use “Medium” as a starting point. The actual noise seen at each sensor will be determined in the design validation step described in Design Validation.

Use the outputs of this sheet to guide the button board layout process, and then check the design prior to prototyping using the C_P, Power Consumption and Response Time Calculator.

Figure 3-13. Layout Estimator

TableA- Estimator

Input Parameters	Value	Units	Comments
Overlay Thickness	5	mm	
Overlay - Dielectric constant	8	farad/m	
Capacitance of trace per inch	2	pF	
Minimum Recommended Button Diameter (based on minimum 0.25pF Finger response)			
Noise Conditions - Low (0.05 pF Noise)	7	mm	
Noise Conditions - Medium (0.075 pF Noise)	8	mm	
Noise Conditions - High (0.1 pF Noise)	10	mm	
Maximum Trace length			
Noise Conditions - Low (0.05 pF Noise)	400	mm	
Noise Conditions - Medium (0.075 pF Noise)	393	mm	
Noise Conditions - High (0.1 pF Noise)	381	mm	
Button to Ground clearance	2	mm	

	input cells, edit with actuals
	output cells, based on inputs

Note: Button diameter of all the buttons CS0 to CS3 will be same with respect to overlay thickness, but can differ with respect to noisy conditions

TableB - Reference values

Overlay Material	Dielectric constant
Plastic	2.8
Plexi glass	8
Formica	4.6-4.9
Glass (Standard)	7.6-8.0
Glass (Ceramic)	6
Mylar	3.2
ABS Plastic	3.8 – 4.5
Wood	1.2-2.5
Trace and board type	**Capacitance per inch in pF**
Copper trace , PCB, 2 layer, 64mil, FR4	2
Copper trace, flex PCB, 2 layer	8

Inputs:

- Overlay thickness
- Overlay dielectric constant
- Capacitance of trace per inch of board

Outputs:

- Recommended minimum button diameter and maximum trace length for different noise conditions
- Button to ground clearance

The button diameter of each button can be different in the design based on whether the noise varies button to button.

3.2.3 C_P, Power Consumption and Response Time Calculator

After board layout is complete, use the Power Consumption and Response Time Calculator to check your design before building a prototype, as shown in Figure 3-14. To verify the C_P value of each button, insert the button diameters and trace lengths into Table A. After you enter the information, the toolbox confirms whether each button is within the specified C_P range of 5—40 pF.

The power calculator in Table B is used to optimize power consumption. Power consumption is a function of the button scan rate, noise immunity level, and the percentage of time the finger is on the button. Enter these details from the drop down menu.

Table C outputs the button response time based on the inputs in Tables A and B.

Figure 3-14. C_P, Power Consumption and Response Time Calculator

Table A: Cp Calculator

Sensor	Button diameter		Trace length		Parasitic capacitance (Cp) of sensors (Approx)		Comments
CS0		mm		mm	0	pF	
CS1		mm		mm	0	pF	
CS2		mm		mm	0	pF	
CS3		mm		mm	0	pF	
CS4		mm		mm	0	pF	
CS5		mm		mm	0	pF	
CS6		mm		mm	0	pF	
CS7		mm		mm	0	pF	
CS8		mm		mm	0	pF	
CS9		mm		mm	0	pF	
CS10		mm		mm	0	pF	
CS11		mm		mm	0	pF	
CS12		mm		mm	0	pF	
CS13		mm		mm	0	pF	
CS14		mm		mm	0	pF	
CS15		mm		mm	0	pF	
Total No of buttons	0	Buttons					

Table B: Power calculator

Select Output Interface	Truth Table	
Select Button Scan Rate	Low	
Approximate Button Scan Rate value	250	ms
% of time finger is on the sensors	20	%
Max Sleep Current	0.0015	mA
Max Active Current	4	mA
Average Current without Finger	0	mA
Average Current with Finger	0	mA
Actual average current consumption	0	mA
Actual average current consumption per butto	0	mA

Table C: Response time calculator

First button press	290	ms
consecutive button press	40	ms

	input cells, edit with actuals
	output cells, based on inputs

Note: The power values given here are worse case, the actual power values will be lower.

Inputs:

- Button diameter and trace length of CS0–CS15 as designed in layout.
- Output interface
- Button Scan Rate used in the design
- The percentage of time a finger is on the buttons.

Outputs:

- C_P for each button. Confirms whether the C_P values are within the specified range of 5—40 pF
- Power consumption per button
- Button response time

3.2.4 Design Validation

After you have built and tested your prototype board, you can use Serial Debug Data Out test mode to capture the raw count, baseline, difference count, and C_P for each of the buttons. You can then enter this information into Table C of the Design Validation Sheet shown in Figure 3-15 to validate your design.

Figure 3-15. Design Validation

Table A: Actual Design values

Input Parameters	Initial value	New value	Units
Overlay Thickness (in mm)	2		mm
Dielectric constant, overlay	2.8		farad/m
Capacitance of trace per inch in	2		pF
Button Scan Rate	Low		
Button Scan Rate Value	250		ms
No of buttons	0	0	Nos
% of time finger is on the sensor	20		%
CS0 Button diameter actual			mm
CS1 Button diameter actual			mm
CS2 Button diameter actual			mm
CS3 Button diameter actual			mm
CS4 Button diameter actual			mm
CS5 Button diameter actual			mm
CS6 Button diameter actual			mm
CS7 Button diameter actual			mm
CS8 Button diameter actual			mm
CS9 Button diameter actual			mm
CS10 Button diameter actual			mm
CS11 Button diameter actual			mm
CS12 Button diameter actual			mm
CS13 Button diameter actual			mm
CS14 Button diameter actual			mm
CS15 Button diameter actual			mm

Table B: Reference values

Overlay Material	Dielectric constant
Plastic	2.8
Plexi glass	8
Formica	4.6-4.9
Glass (Standard)	7.6-8.0
Glass (Ceramic)	6
Mylar	3.2
ABS Plastic	3.8 - 4.5
Wood	1.2-2.5
Trace and board type	**Capacitance per inch in p**
copper trace , PCB, 2 layer, 64mil,FR4	2
copper trace , flex, 2 layer	8

For Table A: The initial values of "Input Parameters" are the ones you have entered in the previous sheets. If your design passes, leave the "New Values" column blank. If your design fails, enter the New values for the corresponding parameter to re-configure your design.

	input cells, edit with actuals
	output cells, based on inputs
	constant cells, same as previous inputs

TableC: Power, Button diameter actuals

Sensor	Values taken from Debug Data out						verage Curre		Improvement Recommendations			
	Noise		Cp		Raw Count				Minimum Button diameter			Maximum Trace Length
CS0	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS1	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS2	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS3	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS4	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS5	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS6	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS7	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS8	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS9	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS10	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS11	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS12	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS13	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS14	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
CS15	0	counts	0	pF	0	counts	0	mA	0	mm		0 mm
			Actual average current consun				0	mA				

Note While logging debug data for this sheet, make sure there is no finger present on the sensors for the log duration

To capture and enter the data use the following steps:

1. Enable the Serial Debug Data Out feature by connecting a 5.2 kΩ resistor to Ground on the DEBUG pin.
2. Power on the device and connect the DEBUG pin to the computer through a COM/RS232 port.
3. Open MultiChart (see AN2397, CapSense Data Viewing Tools) and configure the following:
   a. Select PORT - <Specify Port number>
   b. Port speed – 115200
   c. Visible points – 1000
   d. Log file name – “1.csv” (to be saved in the same folder as the Design Toolbox)
4. Click on **Enable/disable log-file**. This will automatically store the data to the log file. Log this data for atleast 300 samples.
5. Click on **Enable/disable log-file**. This will stop logging the data.
6. Open the log file once. The toolbox will automatically be updated with the relevant data.

Table A shows the various design parameter values, taken up from the previous sheets, so that you need not enter any data in this sheet. This sheet provides a pass/fail grade for the prototype board. If your design fails, you can redesign your system by entering new values in Table A and further recommendations/results will be given. If your design passes, leave the **New value** column in Table A blank.

Inputs:

- Raw Counts
- Noise Counts
- Button C_P
- If the design fails, note the following:
  - ☐ New overlay thickness, overlay material permittivity, button diameter for each individual button, and trace capacitance
  - ☐ Button Scan Rate used in design
  - ☐ Percentage of time a finger is on the buttons

Outputs:

- Power consumption per button
- Design change recommendations: The Design Toolbox provides recommendations based on the actual values from the design if the button size or trace lengths are outside of best design practices.

If the button board does not pass, the Design Toolbox will offer recommendations to guide you to a passing outcome. You can change four areas to remedy a failing design: button size, trace length, overlay material, and overlay thickness. Changing the button size or trace length requires a board spin, while changing the overlay material, thickness, or both, may result in a passing design. The best solution depends on where your design is in the development cycle as well as your end-system requirements.

4. Electrical and Mechanical Design Considerations

When designing a capacitive touch sense technology into your application, it is crucial to remember that the CapSense device exists within a larger framework. Careful attention to every detail, including PCB layout, user interface, and end-user operating environment, leads to robust and reliable system performance. For in-depth information, refer to Getting Started with CapSense.

4.1 Overlay Selection

In CapSense Fundamentals, Equation 1 was presented for finger capacitance

$$C_F = \frac{\varepsilon_0 \varepsilon_r A}{D}$$

Where:

ε_0 = Free space permittivity

ε_r = Dielectric constant of overlay

A = Area of finger and sensor pad overlap

D = Overlay thickness

To increase the CapSense signal strength, choose an overlay material with a higher dielectric constant, decrease the overlay thickness, and increase the button diameter. The Design Toolbox helps you to design a robust and reliable CY8CMBR2016 solution, as discussed in CapSense Schematic Design.

Table 4-1.Overlay Material Dielectric Strength

Material	Breakdown Voltage (V/mm)	Min. Overlay Thickness at 12 kV (mm)
Air	1200–2800	10
Wood – dry	3900	3
Glass – common	7900	1.5
Glass – Borosilicate (Pyrex®)	13,000	0.9
PMMA Plastic (Plexiglas®)	13,000	0.9
ABS	16,000	0.8
Polycarbonate (Lexan®)	16,000	0.8
Formica	18,000	0.7
FR-4	28,000	0.4
PET Film – (Mylar®)	280,000	0.04
Polymide film – (Kapton®)	290,000	0.04

Conductive material cannot be used as an overlay because it interferes with the electric field pattern. Therefore, do not use paints containing metal particles.

4.1.1 Bonding Overlay to PCB

Because the dielectric constant of air is very low, an air gap between the overlay and the button degrades the performance of the button. To eliminate the gap, use a non-conductive adhesive to bond the overlay to the CapSense PCB. A transparent acrylic adhesive film from 3M™ called 200MP is qualified for use in CapSense applications. This special adhesive is dispensed from paper-backed tape rolls (3M™ product numbers 467MP and 468MP).

4.2 ESD Protection

Robust ESD tolerance is a natural byproduct of thoughtful system design. By considering how the contact discharge occurs in your end product, particularly in your user interface, you can withstand an 18-kV discharge event without incurring any damage to the CapSense controller.

CapSense controller pins can withstand a direct 12-kV event. In most cases, the overlay material provides sufficient ESD protection for the controller pins. Table 4-1 lists the thickness of various overlay materials required to protect the CapSense sensors from a 12-kV discharge as specified in IEC 61000-4-2. If the overlay material does not provide sufficient protection, apply ESD countermeasures in the following order: Prevent, Redirect, Clamp.

4.2.1 Prevent

Make sure all paths on the touch surface have a breakdown voltage greater than potential high voltage contacts. In addition, design your system to maintain an appropriate distance between the CapSense controller and possible sources of ESD. If it is not possible to maintain adequate distance, place a protective layer of a high-breakdown-voltage material between the ESD source and CapSense controller. One layer of 5-mil-thick Kapton® tape can withstand 18 kV.

4.2.2 Redirect

If your product is densely packed, it may not be possible to prevent the discharge event. In this case you can protect the CapSense controller by controlling where the discharge occurs. Place a guard ring on the perimeter of the circuit board that is connected to chassis ground. As recommended in PCB Layout Guidelines, providing a hatched ground plane around the button can redirect the ESD event away from the button and CapSense controller.

4.2.3 Clamp

Because CapSense sensors are purposely placed in close proximity to the touch surface, it may not be practical to redirect the discharge path. In this case, consider including series resistors or special-purpose ESD protection devices. The recommended series resistance value is 560 Ω.

A more effective method is to provide special-purpose ESD protection devices on the vulnerable traces. Note that ESD protection devices for CapSense need to be low in capacitance. Table 4-2 lists devices recommended for use with CapSense controllers.

Table 4-2. Low-Capacitance ESD Protection Devices Recommended for CapSense.

ESD Protection device		Input Capacitance	Leakage Current	Contact Discharge maximum limit	Air Discharge maximum limit
Manufacturer	Part Number				
Littlefuse	SP723	5 pF	2 nA	8 kV	15 kV
Vishay	VBUS05L1-DD1	0.3 pF	0.1 µA <	±15 kV	±16 kV
NXP	NUP1301	0.75 pF	30 nA	8 kV	V

4.3 Electromagnetic Compatibility (EMC) Considerations

4.3.1 Radiated Interference

Radiated electrical energy can influence system measurements and potentially influence the operation of the processor core. The interference enters the CY8CMBR2016 CapSense Express chip at the PCB level, through CapSense button traces and any other digital or analog inputs. The Layout guidelines for minimizing the effects of RF interference include:

- **Ground Plane**: Provide a ground plane on the PCB.
- **Series Resistor**: Place series resistors within 10 mm of the CapSense controller pins
- The recommended series resistance for CapSense input lines is 560 ohms.
- **Trace Length**: Minimize trace length whenever possible.
- **Current Loop Area**: Minimize the return path for current. Hatched ground, instead of solid fill, should be provided within 1 cm of the buttons and traces to reduce the impact of parasitic capacitance.
- **RF Source Location**: Partition systems with noise sources such as LCD inverters and switched-mode power supplies (SMPS) to keep them separated from CapSense inputs. Shielding the power supply is another common technique for preventing interference.

4.3.2 Conducted Immunity and Emissions

Noise entering a system through interconnections with other systems is referred to as conducted noise. These interconnections include power and communication lines. Because CapSense controllers are low-power devices, conducted emissions must be avoided. The following guidelines will help reduce conducted emission and immunity:

- Use decoupling capacitors as recommended by the datasheet.
- Add a bidirectional filter on the input to the system power supply. This is effective for both conducted emissions and immunity. A pi-filter can prevent power supply noise from effecting sensitive parts, while also preventing the switching noise of the part itself from coupling back to the power planes.
- If the CapSense controller PCB is connected to the power supply by a cable, minimize the cable length and consider using a shielded cable.
- A ferrite bead can be placed around the power supply or communication lines to filter out high frequency noise.

4.4 Scan Line Waveform Requirements (Key Scan Interface)

You should complete a feasibility study when designing a mechanical keypad system with the CY8CMBR2016 device.

Figure 4-1 shows the basic SCAN line outputs in a typical keypad scan system. Figure 4-2 shows CY8CMBR2016's specific timing details.

Figure 4-1. Scan Outputs

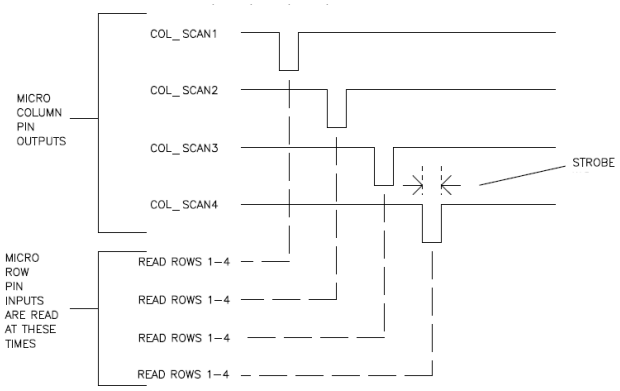
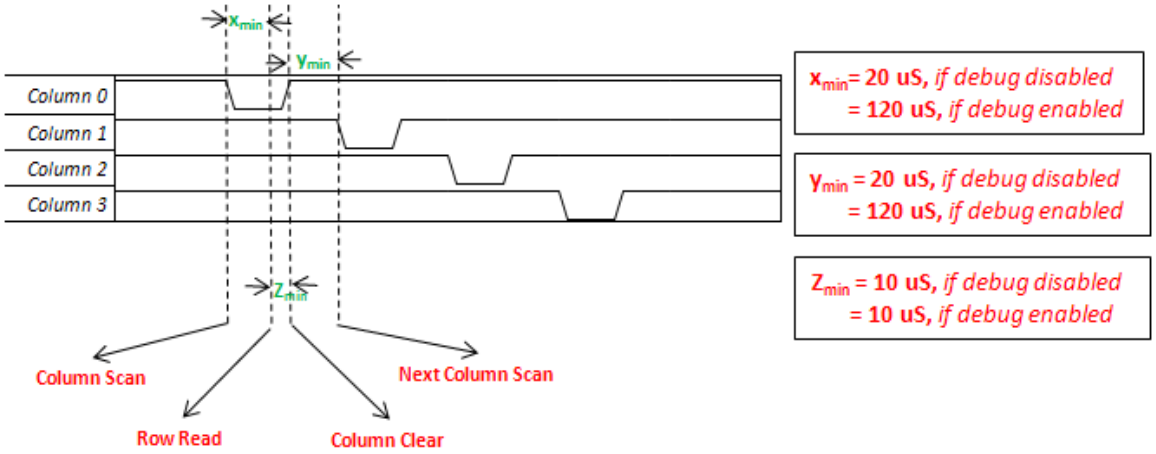


Figure 4-2. Scan/Column Waveform Details



The first phase of the design feasibility study is to check that your design conforms to the polling waveforms shown in Figure 4-1 and Figure 4-2satisfies the timing details mentioned in Figure 4-2.

The next phase of the study involves checking your system's scanning method. If you use a polling-based method, your design is straight forward, and you can move forward. If you use an interrupt based method, the CY8CMBR2016's interrupt line needs to trigger polling on the SCAN lines. The interrupt line is an active high line, therefore, if there are any active buttons, the line goes high and remains high until all buttons are released. The line should be configured to be an active high interrupt in the host and whenever the interrupt is triggered, the host should poll the scan lines and read the button status.

4.5 PCB Layout Guidelines

Cypress has created an interactive spreadsheet, Design Toolbox to help you design a robust CY8CMBR2016 CapSense PCB layout, as discussed in Layout Guidelines.

Detailed PCB layout guidelines are available in Getting Started with CapSense, Section 3.7.

5. Low-Power Design Considerations

5.1 System Design Considerations

Cypress's CY8CMBR2016 is designed to meet the low-power requirements of battery powered applications. To minimize power consumption, take these steps:

- Ground all unused CapSense inputs (CSx)
- Minimize C_P using the design guidelines in Getting Started with CapSense, Section 3.7.1
- Reduce supply voltage (valid range is 1.71—5.5 V)
- Use a higher Button Scan Rate or Deep Sleep operating mode, refer to Button Scan Rate
- Connect un-used scan lines to V_{DD}

5.2 Calculating Average Power

The design toolbox, which is the tool of choice, automates the power optimization calculations described in the following sections. The average power consumed by the CY8CMBR2016 is determined by calculating the parameters below:

1. Button Scan Rate, T_R
2. Scan Time, T_S
3. Average current in a NO TOUCH state, I_{AVE_NT}
4. Average current in a TOUCH state, I_{AVE_T}
5. Average use current, I_{AVE_U}
6. Average current, I_{AVE}
7. Average power, P_{AVE}

5.2.1 Button Scan Rate (T_R)

The Button Scan Rate is set using an external resistor as defined in Table 3-5.

5.2.1.1 Response Time

Response Time is the minimum amount of time a button CSx should be touched for the device to detect as valid button touch and produce an output.

Equation 4

$$RT_{FBT} = Button\ Scan\ Rate + 40\ ms$$

$$RT_{CBT} = 40\ ms$$

Where:

RT_{FBT} is Response Time for first button touch

RT_{CBT} is Response Time for consecutive button touch after first button touch

5.2.2 Scan Time (T_S)

Approximate scan time can be calculated using the following equation:

$$T_S = [0.325\ ms\ \times (K_{CS0} + K_{CS1} + K_{CS2} + \cdots + K_{CS15})] + T_{FW}$$ Equation 5

Where:

K_{CSx} = Button Sensitivity Constant for CSx, from Table 5-1.

T_{FW} = Firmware execution time, from Table 5-2.

Table 5-1. Button Sensitivity Constant

C_P (pF)	Button Sensitivity Constant (K)
Button connected to Ground	0
5 pF ≤ C_P ≤ 11 pF	1
11 pF < C_P ≤ 22 pF	2
22 pF < C_P ≤ 40 pF	4

Table 5-2. Average Current Parameters

Parameter	Typ	Max
T_{FW}	-	5.00 ms
I_{SLEEP}	1.07 µA	1.5 µA
T_S	Equation 5	+5% from TYP value
T_R	Table 3-5	±10 from Value
I_{ACTIVE}	3.4 mA	4.0 mA

5.2.3 Average Current in NO TOUCH state (I_{AVE_NT})

$$I_{AVE_NT} = \left(\frac{T_R - T_S}{T_R} \times I_{SLEEP}\right) + \left(\frac{T_S}{T_R} \times I_{ACTIVE}\right)$$ Equation 6

Where:

T_R = Button Scan Rate

T_S = Scan time

I_{SLEEP} = current consumed by CY8CMBR2016 during low power sleep mode, from Table 5-2

I_{ACTIVE} = current consumed by CY8CMBR2016 during active operation, from Table 5-2

5.2.4 Average Current in TOUCH state (I_{AVE_T})

Equation 7

$$I_{AVE_T} = \left(\frac{40\ mS - T_S}{40\ mS} \times I_{SLEEP}\right) + \left(\frac{T_S}{40\ mS} \times I_{ACTIVE}\right) \quad Output\ mode \neq Key\ Scan\ Interface$$

$$I_{AVE_T} = 4\ \mu A \quad Output\ mode = Key\ Scan\ Interface$$

Where:

T_S = Scan time

I_{SLEEP} = current consumed by CY8CMBR2016 during low power sleep mode, from Table 5-2

I_{ACTIVE} = current consumed by CY8CMBR2016 during active operation, from Table 5-2

5.2.5 Average Use Current (I_{AVE_U})

$$I_{AVE_U} = \left(\frac{100-P}{100} \times I_{AVE_NT}\right) + \left(\frac{P}{100} \times I_{AVE_T}\right)$$ Equation 8

Where:

P = percentage of time CapSense button CSx is in the TOUCH state compared to total time CY8CMBR2016 is on

I_{AVG_NT} = average current in the NO TOUCH state

I_{AVG_T} = average current in the TOUCH state

5.2.6 Average Current (I_{AVE})

$$I_{AVE} = \left[I_{AVE_U} \times \left(\frac{T_{SA}}{T_{DS}+T_{SA}}\right)\right] + 0.1\ \mu A$$ Equation 9

Where:

T_{SA} = time device is not in deep sleep mode

T_{DS} = time device is in deep sleep mode

5.2.7 Average Power (P_{AVE})

$$P_{AVE} = V_{DD} \times I_{AVE}$$ Equation 10

Where:

I_{AVE} = average current

V_{DD} = supply voltage

5.2.8 Example Calculation

As an example of how to calculate average power, consider a CapSense user interface with nine well-designed buttons. The C_P for all nine buttons is 10—20 pF. The Encoded 4-bit Output Interface is selected. The Button Scan Rate is set to 150 ms (Medium).

The scan time can be calculated from Equation 5, with the Button sensitivity constant obtained from Table 5-1, and the typical value for firmware execution time from Table 5-2.

$$T_S = [0.325\ ms \times (9 \times 2)] + 5.00 = 10.85\ ms$$

The average current in NO TOUCH state is calculated as follows using Equation 6 and the maximum values for I_{SLEEP} and I_{ACTIVE} from Table 5-2.

$$I_{AVE_NT} = \left(\frac{150-10.85}{150} \times 1.50\ \mu A\right) + \left(\frac{10.85}{150} \times 4.0\ mA\right) = 290.72\ \mu A$$

The average current in TOUCH state is calculated as follows using Equation 7:

$$I_{AVE_T} = \left(\frac{40-10.85}{40} \times 1.50\ \mu A\right) + \left(\frac{10.85}{40} \times 4.0\ mA\right) = 1086.1\ \mu A$$

To calculate the average current, assume the user touches a CapSense button for three seconds every minute. This means your design operates in a TOUCH state about five percent of the day during which the device is turned on. The average current consumption of the design is calculated as follows using Equation 8:

$$I_{AVE_U} = (0.05 \times 1086.1\ \mu A) + (0.95 \times 290.72\ \mu A) = 330.5\ \mu A$$

Assuming this design does not utilize deep sleep mode and that it operates at 1.71 V the average power is calculated as follows using Equation 10:

$$P_{AVE} = 1.71 \times 330.5\ \mu A = 565.2\ \mu W$$

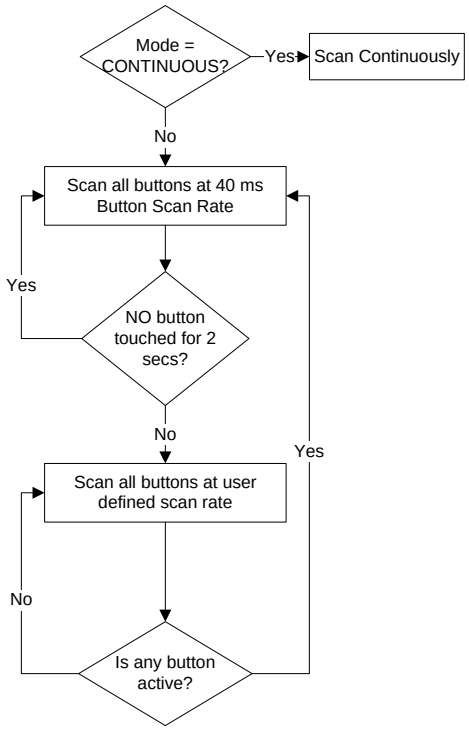
5.3 Sleep Modes

Cypress's CY8CMBR2016 can be configured to operate in either low-power sleep mode or deep sleep mode. These modes reduce the power consumption of the device.

5.3.1 Low-Power Sleep Mode

When the CY8CMBR2016 device operates in low power sleep mode, the device draws 1 µA when not scanning the CS inputs. The button scan rate determines the amount of time the device sleeps. The behavior of the CY8CMBR2016 controller in low-power sleep mode is described in Figure 5-1.

Figure 5-1. Low-Power Sleep Mode



Because low-power sleep mode does not require a host processor to operate, it is also referred to as standalone mode.

5.3.2 Deep Sleep Mode

If you use the CY8CMBR2016 in a system with a host processor, the Sleep pin can be used to operate the device in deep sleep mode.

When the host processor provides logic high to the Sleep pin, the CY8CMBR2016 enters deep sleep mode. In this mode all ongoing communication is suspended and the device consumes ~0.1 µA current.

When the host processor drives a GND signal to the Sleep pin the CY8CMBR2010 wakes up from deep sleep and operates in low-power sleep mode. When the device comes out of deep sleep mode, the CapSense system is reinitialized. Typical time for re-initialization is 8 ms. Any button press within this time is not reported.

6. Resources

6.1 Website

Visit Cypress's CapSense Express Controllers website to access all of the reference material discussed in this section.

Find a variety of technical resources for the CY8CMBR2016 CapSense Express device on the CY8CMBR2016 web page.

6.2 Datasheet

The datasheet for the CapSense CY8CMBR2016 device is available here.

6.3 Kits

Visit our partner website for supported evaluation kit for CY8CMBR2016.

6.4 Design Toolbox

Cypress has created an interactive spreadsheet, the Design Toolbox, to help you rapidly design a robust and reliable CY8CMBR2016 CapSense solution.

6.5 Sample Board Files

Cypress offers sample schematic and board files, which can be used as a reference to quickly complete your PCB design process.

6.6 MultiChart

MultiChart is a simple PC tool for real-time CapSense data viewing and logging. The application allows you to view data from up to 48 sensors, save and print charts, and save data for later analysis in a spreadsheet.

6.7 Design Support

Cypress has a variety of design support channels to ensure the success of your CapSense solutions.

Knowledge Base Articles – Refer to technical articles by product family or perform a search on various CapSense topics.

CapSense Application Notes – Refer to a wide variety of application notes built on information presented in this document.

White Papers – Learn about advanced capacitive-touch interface topics.

Cypress Developer Community – Connect with the Cypress technical community and exchange information.

CapSense Product Selector Guide – See the complete product offering of Cypress's CapSense product line.

Video Library – Quickly get up to speed with tutorial videos

Quality & Reliability – Cypress is committed to complete customer satisfaction. At our Quality website you can find reliability and product qualification reports.

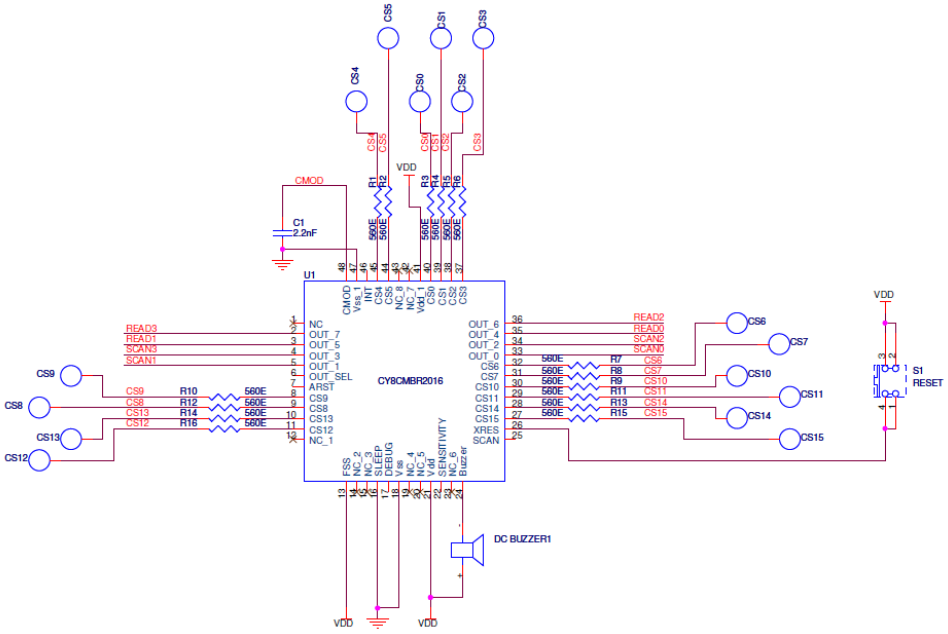
Technical Support – World class technical support is available online.

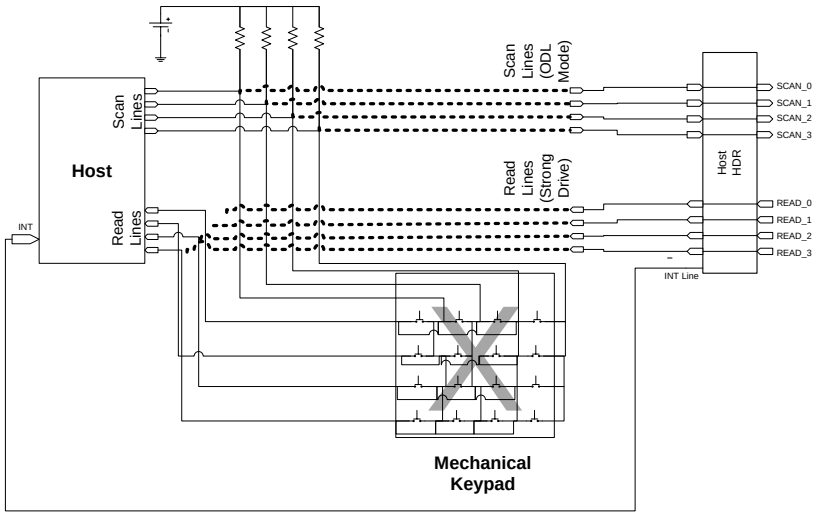
7. Appendix

7.1 Schematic Examples

7.1.1 Schematic 1: 16 Buttons with Key Scan Output Mode

Figure 7-1. Schematic 1



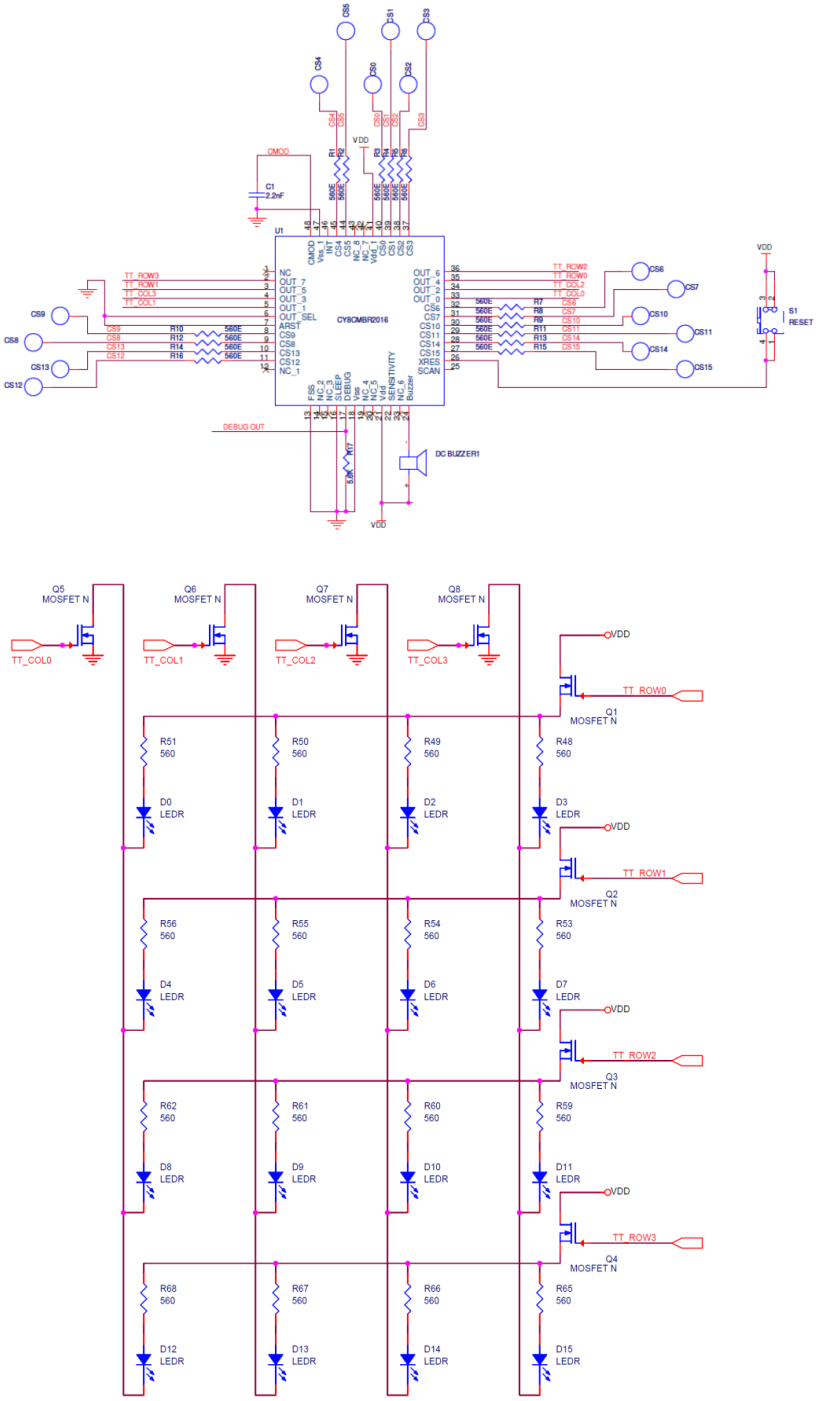


In Schematic 1, CY8CMBR2016 is configured as follows:

- 16 CapSense buttons
- Key Scan Interface
- Continuous scan mode
- High sensitivity for all buttons
- FSS enabled
- Button Auto Reset disabled
- Serial Debug Data Out disabled
- DC Buzzer output
- Reset button
- Interrupt line output

7.1.2 Schematic 2: 16 Buttons with Truth Table Output Mode

Figure 7-2. Schematic 2



In Schematic 2, CY8CMBR2016 is configured as follows:

- 16 CapSense buttons
- Truth Table Output configured to drive LEDs
- Continuous Scan mode
- High sensitivity for all buttons
- FSS disabled
- Button Auto Reset enabled, with a period of 5 seconds
- Serial Debug Data Out enabled
- DC Buzzer output
- Reset button
- Interrupt line output

Glossary

AMUXBUS

Analog multiplexer bus available inside PSoC that helps to connect I/O pins with multiple internal analog signals.

SmartSense™ Auto-Tuning

A CapSense algorithm that automatically sets sensing parameters for optimal performance after the design phase and continuously compensates for system, manufacturing, and environmental changes.

Baseline

A value resulting from a firmware algorithm that estimates a trend in the Raw Count when there is no human finger present on the sensor. The Baseline is less sensitive to sudden changes in the Raw Count and provides a reference point for computing the Difference Count.

Button or Button Widget

A widget with an associated sensor that can report the active or inactive state (that is, only two states) of the sensor. For example, it can detect the touch or no-touch state of a finger on the sensor.

Difference Count

The difference between Raw Count and Baseline. If the difference is negative, or if it is below Noise Threshold, the Difference Count is always set to zero.

Capacitive Sensor

A conductor and substrate, such as a copper button on a printed circuit board (PCB), which reacts to a touch or an approaching object with a change in capacitance.

CapSense®

Cypress's touch-sensing user interface solution. The industry's No. 1 solution in sales by 4x over No. 2.

CapSense Mechanical Button Replacement (MBR)

Cypress's configurable solution to upgrade mechanical buttons to capacitive buttons, requires minimal engineering effort to configure the sensor parameters and does not require firmware development. These devices include the CY8CMBR3XXX and CY8CMBR2XXX families.

Centroid or Centroid Position

A number indicating the finger position on a slider within the range given by the Slider Resolution. This number is calculated by the CapSense centroid calculation algorithm.

Compensation IDAC

A programmable constant current source, which is used by CSD to compensate for excess sensor C_P. This IDAC is not controlled by the Sigma-Delta Modulator in the CSD block unlike the Modulation IDAC.

CSD

CapSense Sigma Delta (CSD) is a Cypress-patented method of performing self-capacitance (also called self-cap) measurements for capacitive sensing applications.

In CSD mode, the sensing system measures the self-capacitance of an electrode, and a change in the self-capacitance is detected to identify the presence or absence of a finger.

Debounce

A parameter that defines the number of consecutive scan samples for which the touch should be present for it to become valid. This parameter helps to reject spurious touch signals.

A finger touch is reported only if the Difference Count is greater than Finger Threshold + Hysteresis for a consecutive Debounce number of scan samples.

Driven-Shield

A technique used by CSD for enabling liquid tolerance in which the Shield Electrode is driven by a signal that is equal to the sensor switching signal in phase and amplitude.

Electrode

A conductive material such as a pad or a layer on PCB, ITO, or FPCB. The electrode is connected to a port pin on a CapSense device and is used as a CapSense sensor or to drive specific signals associated with CapSense functionality.

Finger Threshold

A parameter used with Hysteresis to determine the state of the sensor. Sensor state is reported ON if the Difference Count is higher than Finger Threshold + Hysteresis, and it is reported OFF if the Difference Count is below Finger Threshold – Hysteresis.

Ganged Sensors

The method of connecting multiple sensors together and scanning them as a single sensor. Used for increasing the sensor area for proximity sensing and to reduce power consumption.

To reduce power when the system is in low-power mode, all the sensors can be ganged together and scanned as a single sensor taking less time instead of scanning all the sensors individually. When the user touches any of the sensors, the system can transition into active mode where it scans all the sensors individually to detect which sensor is activated.

PSoC supports sensor-ganging in firmware, that is, multiple sensors can be connected simultaneously to AMUXBUS for scanning.

Gesture

Gesture is an action, such as swiping and pinch-zoom, performed by the user. CapSense has a gesture detection feature that identifies the different gestures based on predefined touch patterns. In the CapSense component, the Gesture feature is supported only by the Touchpad Widget.

Guard Sensor

Copper trace that surrounds all the sensors on the PCB, similar to a button sensor and is used to detect a liquid stream. When the Guard Sensor is triggered, firmware can disable scanning of all other sensors to prevent false touches.

Hatch Fill or Hatch Ground or Hatched Ground

While designing a PCB for capacitive sensing, a grounded copper plane should be placed surrounding the sensors for good noise immunity. But a solid ground increases the parasitic capacitance of the sensor which is not desired. Therefore, the ground should be filled in a special hatch pattern. A hatch pattern has closely-placed, crisscrossed lines looking like a mesh and the line width and the spacing between two lines determine the fill percentage. In case of liquid tolerance, this hatch fill referred as a shield electrode is driven with a shield signal instead of ground.

Hysteresis

A parameter used to prevent the sensor status output from random toggling due to system noise, used in conjunction with the Finger Threshold to determine the sensor state. See Finger Threshold.

IDAC (Current-Output Digital-to-Analog Converter)

Programmable constant current source available inside PSoC, used for CapSense and ADC operations.

Liquid Tolerance

The ability of a capacitive sensing system to work reliably in the presence of liquid droplets, streaming liquids or mist.

Linear Slider

A widget consisting of more than one sensor arranged in a specific linear fashion to detect the physical position (in single axis) of a finger.

Low Baseline Reset

A parameter that represents the maximum number of scan samples where the Raw Count is abnormally below the Negative Noise Threshold. If the Low Baseline Reset value is exceeded, the Baseline is reset to the current Raw Count.

Manual-Tuning

The manual process of setting (or tuning) the CapSense parameters.

Matrix Buttons

A widget consisting of more than two sensors arranged in a matrix fashion, used to detect the presence or absence of a human finger (a touch) on the intersections of vertically and horizontally arranged sensors.

If M is the number of sensors on the horizontal axis and N is the number of sensors on the vertical axis, the Matrix Buttons Widget can monitor a total of M x N intersections using ONLY M + N port pins.

When using the CSD sensing method (self-capacitance), this Widget can detect a valid touch on only one intersection position at a time.

Modulation Capacitor (CMOD)

An external capacitor required for the operation of a CSD block in Self-Capacitance sensing mode.

Modulator Clock

A clock source that is used to sample the modulator output from a CSD block during a sensor scan. This clock is also fed to the Raw Count counter. The scan time (excluding pre and post processing times) is given by
$(2^N - 1)$/Modulator Clock Frequency, where N is the Scan Resolution.

Modulation IDAC

Modulation IDAC is a programmable constant current source, whose output is controlled (ON/OFF) by the sigma-delta modulator output in a CSD block to maintain the AMUXBUS voltage at V_{REF}. The average current supplied by this IDAC is equal to the average current drawn out by the sensor capacitor.

Mutual-Capacitance

Capacitance associated with an electrode (say TX) with respect to another electrode (say RX) is known as mutual capacitance.

Negative Noise Threshold

A threshold used to differentiate usual noise from the spurious signals appearing in negative direction. This parameter is used in conjunction with the Low Baseline Reset parameter.

Baseline is updated to track the change in the Raw Count as long as the Raw Count stays within Negative Noise Threshold, that is, the difference between Baseline and Raw count (Baseline – Raw count) is less than Negative Noise Threshold.

Scenarios that may trigger such spurious signals in a negative direction include: a finger on the sensor on power-up, removal of a metal object placed near the sensor, removing a liquid-tolerant CapSense-enabled product from the water; and other sudden environmental changes.

Noise (CapSense Noise)

The variation in the Raw Count when a sensor is in the OFF state (no touch), measured as peak-to-peak counts.

Noise Threshold

A parameter used to differentiate signal from noise for a sensor. If Raw Count – Baseline is greater than Noise Threshold, it indicates a likely valid signal. If the difference is less than Noise Threshold, Raw Count contains nothing but noise.

Overlay

A non-conductive material, such as plastic and glass, which covers the capacitive sensors and acts as a touch-surface. The PCB with the sensors is directly placed under the overlay or is connected through springs. The casing for a product often becomes the overlay.

Parasitic Capacitance (C_P)

Parasitic capacitance is the intrinsic capacitance of the sensor electrode contributed by PCB trace, sensor pad, vias, and air gap. It is unwanted because it reduces the sensitivity of CSD.

Proximity Sensor

A sensor that can detect the presence of nearby objects without any physical contact.

Radial Slider

A widget consisting of more than one sensor arranged in a specific circular fashion to detect the physical position of a finger.

Raw Count

The unprocessed digital count output of the CapSense hardware block that represents the physical capacitance of the sensor.

Refresh Interval

The time between two consecutive scans of a sensor.

Scan Resolution

Resolution (in bits) of the Raw Count produced by the CSD block.

Scan Time

Time taken for completing the scan of a sensor.

Self-Capacitance

The capacitance associated with an electrode with respect to circuit ground.

Sensitivity

The change in Raw Count corresponding to the change in sensor capacitance, expressed in counts/pF. Sensitivity of a sensor is dependent on the board layout, overlay properties, sensing method, and tuning parameters.

Sense Clock

A clock source used to implement a switched-capacitor front-end for the CSD sensing method.

Sensor

See Capacitive Sensor.

Sensor Auto Reset

A setting to prevent a sensor from reporting false touch status indefinitely due to system failure, or when a metal object is continuously present near the sensor.

When Sensor Auto Reset is enabled, the Baseline is always updated even if the Difference Count is greater than the Noise Threshold. This prevents the sensor from reporting the ON status for an indefinite period of time. When Sensor Auto Reset is disabled, the Baseline is updated only when the Difference Count is less than the Noise Threshold.

Sensor Ganging

See Ganged Sensors.

Shield Electrode

Copper fill around sensors to prevent false touches due to the presence of water or other liquids. Shield Electrode is driven by the shield signal output from the CSD block. See Driven-Shield.

Shield Tank Capacitor (C_{SH})

An optional external capacitor (C_{SH} Tank Capacitor) used to enhance the drive capability of the CSD shield, when there is a large shield layer with high parasitic capacitance.

Signal (CapSense Signal)

Difference Count is also called Signal. See Difference Count.

Signal-to-Noise Ratio (SNR)

The ratio of the sensor signal, when touched, to the noise signal of an untouched sensor.

Slider Resolution

A parameter indicating the total number of finger positions to be resolved on a slider.

Touchpad

A Widget consisting of multiple sensors arranged in a specific horizontal and vertical fashion to detect the X and Y position of a touch.

Trackpad

See Touchpad.

Tuning

The process of finding the optimum values for various hardware and software or threshold parameters required for CapSense operation.

V_{REF}

Programmable reference voltage block available inside PSoC used for CapSense and ADC operation.

Widget

A user-interface element in the CapSense component that consists of one sensor or a group of similar sensors. Button, proximity sensor, linear slider, radial slider, matrix buttons, and touchpad are the supported widgets.

Revision History

Document Revision History

Document Title: AN73034 - CY8CMBR2016 CapSense® Design Guide Document Number: 001-73034			
Revision	**Issue Date**	**Origin of Change**	**Description of Change**
**	01/02/2012	MSUR	New design guide.
*A	10/17/2012	UDYG	Multiple updates to design guide.
*B	09/02/2014	SSHH	Updated Figure 7-2.
*C	02/04/2015	SSHH	Sunset review; no content update.
*D	04/22/2015	MQJ	Added AN73034 in the spec title
*E	01/21/2016	VAIR	Added Glossary.
*F	07/14/2017	AESATMP8	Updated logo and Copyright.