

AN2394

Author: Mark Lee

Associated Project: No

Associated Part Family: CY8C20xx6, CY8C20x34, CY8C21x4, CY8C24x94

[GET FREE SAMPLES HERE](#)

Software Version: PSoC Designer™ 5.0

Associated Application Notes: 2233a, 2292, 2360, 2318, 2403

Application Note Abstract

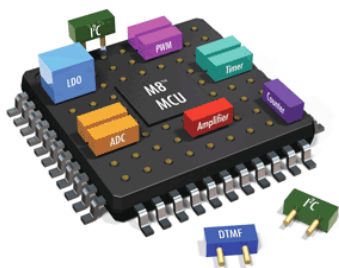
This application note presents best practices for designing CapSense™ systems. The topics covered include an overview of sensing methods, guidelines for layout and assembly, and CapSense tools and techniques.

Introduction

Adopting capacitive sensing as an interface technology in high volume, high visibility applications such as portable media players and mobile handsets has created demand for this technology in more conventional consumer electronics. This demand has led to significant innovation, and several competitive technologies are available. While these technologies each have their respective differences, the underlying principle is the measurement of capacitance between a plate (the sensor) and its environment.

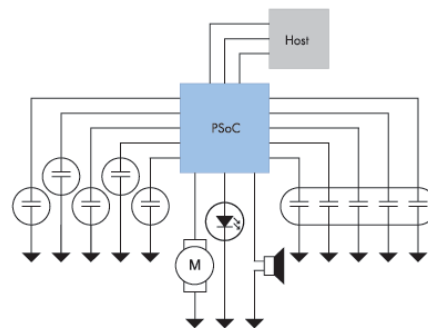
Compared to modules and fixed-function ICs, programmable ICs enable more flexibility in design as custom code is used to develop solutions. PSoC® CapSense combines a microcontroller, configurable digital and analog resources, on-board memory, and other features that enable flexibility in capacitive system design. This application note gives an overview of best practices for CapSense design.

Figure 1. PSoC Analog and Digital Blocks Configured for CapSensePlus



The PSoC architecture allows designers to incorporate multiple capacitive sensing design elements into an application. Buttons, sliders, touchpads, and proximity detectors are supported simultaneously with the same device in the same circuit. Use PSoC to scan capacitive sensors and use the activation status to drive LEDs, control a motor, drive a speaker, and so on, as shown in Figure 2. A concept called dynamic reconfiguration enables the CapSense application to use more than 100 percent of the system resources by reconfiguring as needed on-the-fly.

Figure 2. Sample Application with CapSensePlus: Motor, LED, and Speaker with a Single PSoC



How to Design CapSense Printed Circuit Boards

In the typical CapSense application, the capacitive sensors are formed by the traces of a printed circuit board (PCB). The following set of guidelines show how to design a CapSense PCB (see [AN2292](#)).

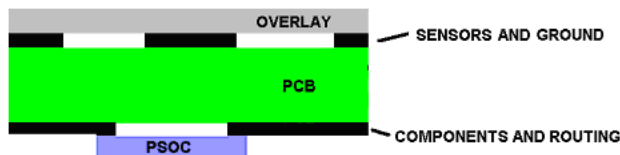
General Layout and Assembly Guidelines

Board Area: The area required for CapSense is only slightly larger than the sensing area itself. The electric fields around the sensors are very localized, especially if the ground plane and sensor pad are placed on the same PCB layer.

PSoC Placement: It is good practice to minimize the distance between the PSoC and the sensors. Typically, the PSoC is mounted on the bottom layer along with the other components and the CapSense sensor pads are placed on the top layer.

Board Layers: The most common PCB format is two layers with sensor pads and gridded ground plane on top and everything else at the bottom. Four layer boards are used when board area must be minimized. Do not route traces directly under sensor pads.

Figure 3. Two Layer Stack-Up for CapSense Boards

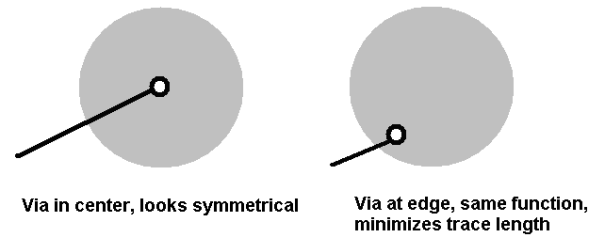


Board Thickness: FR4-based designs perform well with standard board thickness ranging from 0.020 inch (0.5 mm) to 0.063 inch (1.6 mm). See [Flex Circuits](#) for guidelines.

Trace Length and Width: The parasitic capacitance of the traces and sensor pad, C_p , is minimized to make the dynamic range of the system as large as possible. C_p is minimized by short and narrow traces. The longest traces in a successful CapSense product are 9 inches (230 mm) for a slider, and 12 inches (300 mm) for a button. (This extreme example requires large sensing pads and a thin overlay to maximize the signal from the sensor.) Trace width adds to the sensor C_p and increases coupling to elements on other layers. Trace widths of 0.0065 inch to 0.008 inch (0.17 to 0.20 mm) suffice for most applications.

Vias: Use the minimum number of vias consistent with routing of the CapSense inputs to minimize C_p . The placement of the via is done at any location on the sensor pad (see Figure 4).

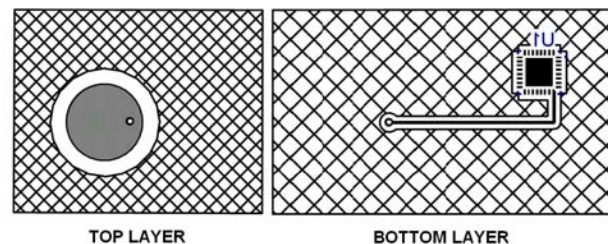
Figure 4 Via to Sensor Pad can be Anywhere on the Pad



Communication Lines: Do not run capacitive sensing traces in close proximity with and parallel to high frequency communication lines, such as I2C or SPI. If it is necessary to cross communication lines with sensor pins, ensure the intersection is orthogonal. One effective method of reducing the interaction between communication traces and sensor traces is to isolate each by port assignment. Port pins P1[0] and P1[1] are used for programming and I2C. If they are not used for these purposes, then they are assigned to CapSense. Series resistors are recommended in line with all CapSense inputs (560 ohms) and series with communication lines such as I2C and SPI (300 ohms) to prevent RF interference.

Ground Fill: When ground fill is added near a CapSense sensor pad, there is a trade-off between maintaining a high level of CapSense signal and increasing noise immunity of the system. Typical hatching for the ground fill is 15 percent on the top layer (7 mil line, 45 mil spacing), and 10 percent on the bottom layer (7 mil line, 70 mil spacing), as shown in Figure 5.

Figure 5. Partial Ground Fill to Minimize C_p



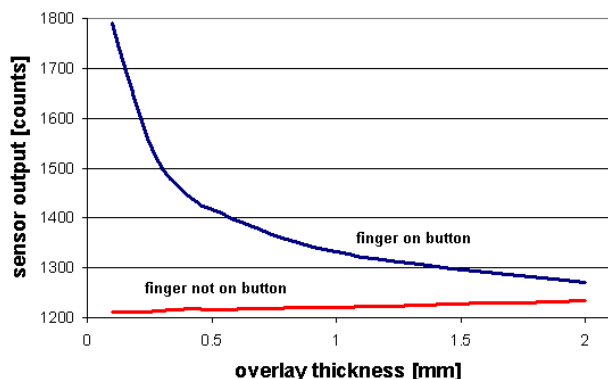
Overlay Thickness: Table 1 lists the recommended maximum overlay thicknesses for PSoC CapSense applications (plastic overlay). The dielectric constant plays a role in the thickness of the overlay. Common glass has a dielectric constant around $\epsilon_r = 8$, while plastic is around $\epsilon_r = 2.5$. The ratio of $\epsilon_r / 2.5$ is an estimate of how thick the overlay is, relative to plastic for the same level of sensitivity. Using this rule of thumb, a common glass overlay is about three times as thick as a plastic overlay for the same sensitivity.

Table 1. Recommended Plastic Overlay Thickness for CapSense

Design Element	Overlay Thickness
Button	< 5 mm
Slider	< 2 mm
Touchpad	< 0.5 mm

Both signal and noise are affected by the overlay properties. As the thickness of the overlay increases, signal and noise both decrease. Signal is defined as the difference in average sensor output between the Finger Absent and Finger Present states. Noise is defined in the peak-to-peak deviation in sensor output in the Finger Absent state. A representative plot of CapSense signal versus overlay thickness is shown in Figure 6.

Figure 6. Signal Level Drops Off as Overlay Thickness Increases



Overlay Adhesive: Overlay materials must have good mechanical contact with the sensing PCB. Two widely used non-conductive adhesives for overlays are 467MP and 468MP, made by 3M.

Gloves: If the sensors work with a gloved hand, then add the thickness of the glove material to the total overlay stack-up when sizing the buttons. Dry leather and rubber are similar to plastic with a dielectric constant of 2.5 to 3.5. Ski gloves have a dielectric constant of 2 or less, depending on the air content of glove's thermal insulation.

LED Backlighting: CapSense works well with LED backlighting. Just cut a hole in the sensor pad. Keep LED traces at the bottom of the board.

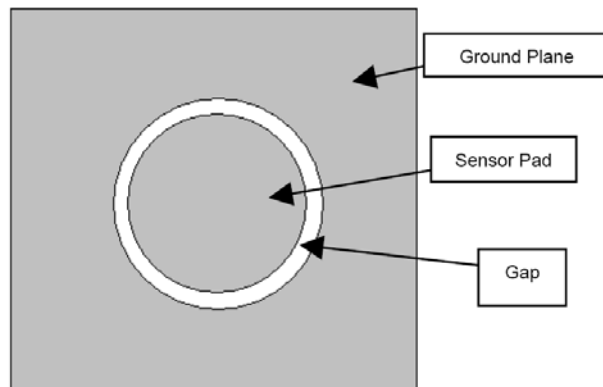
Multiple PSoCs on One PCB: For systems with many buttons, such as a keyboard, the system design may require two or more PSoCs dedicated to CapSense. If this is the case, partition buttons so that a ground fill area separates the traces of each button group. This prevents coupling between the independent CapSense groups.

Buttons

The function of a button is to determine the presence or absence of a conductive object. A typical application of a CapSense button is to sense the presence of a finger.

Shape: The recommended shape for sensing a finger press is a solid round pattern as shown in Figure 7.

Figure 7. Recommended Button Shape is Solid Round Pattern



The capacitance, C_P , decreases as the clearance surrounding the button is increased. An example of this dependence of C_P on the gap is shown in Figure 8 for three button sizes (5 mm, 10 mm, and 15 mm diameter).

Figure 8. C_P as a Function of Button-Ground Clearance and Button Diameter (0.062 inch Thick, FR4)

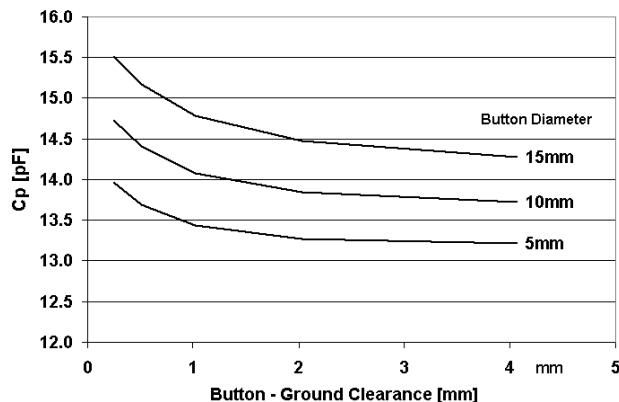


Figure 9 gives an overview of shape recommendations for buttons. A square or rectangular button works if the layout does not support a round shape. Buttons should not be triangular or include other pointed features with angles less than 90 degrees. Interdigitated sensor traces do not work well with CSD or CSA.

Figure 9. Recommendations for Button Shape



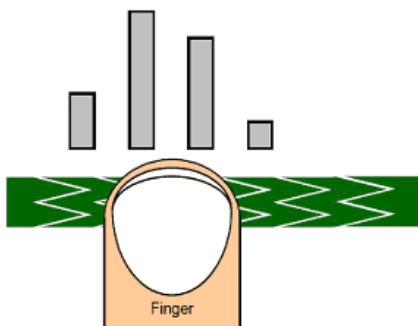
Round buttons must be at least 5 mm in diameter, and rectangular buttons must have at least one dimension greater than 6 mm. The size of the button is selected to meet the minimum signal-to-noise ratio (SNR) requirement of 5:1 (see AN2403). The thicker the protective overlay, the larger the button diameter should be.

Sliders

A slider is a sensor array. Changes between adjacent capacitive elements are used to determine the position of a conductive object. Position is determined in firmware using a centroid (center of mass) calculation.

The slider segments must be small enough so that multiple segments couple with each finger position, yet large enough to produce the required signal level through the overlay. A sawtooth pattern works well for sliders, with a minimum of five segments. Each finger position on the slider produces a usable signal on at least three segments. The maximum length of the slider is only limited by the available IO pins of the PSoC and the required report rate for the system. A typical slider pattern is shown in Figure 10. The sensor output is represented by a bar graph above each slider segment.

Figure 10. Sawtooth Pattern Used With Slider Sensor Pattern



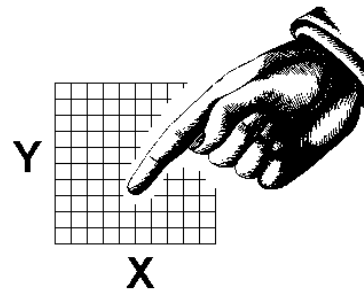
Slider Diplexing: If IO pins are at a premium, connecting two slider segments to a single PSoC pin increases the number of slider segments that are sensed by the PSoC by two-fold. The CapSense User Module Wizard enables the user to select this as an option for pin assignment, and the User Module API determines the correct half of the

slider that the finger is touching. Connecting each CapSense input pin to two slider elements doubles C_P without any increase in signal.

Touchpads

The CapSense User Module does not directly support touchpads. Touchpads are implemented as two independent sliders. All the guidelines that apply to sliders also apply to touchpads.

Figure 11. Touchpads with CapSense Sliders X and Y

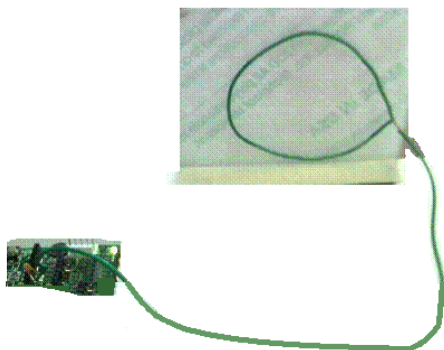


An example of a good CapSense touchpad is a commercially successful design with a 20-position column slider (X) and a 10-position row slider (Y). A total of 30 pins are assigned as CapSense inputs. The dimensions of the active area are 3.9 inches x 1.9 inches (99 mm x 47 mm). The overlay is 0.010-inch (0.25 mm) ABS plastic. The row and column sensors are spaced with a pitch of 0.2 inch (5 mm). The baseline noise level is a single count in the Finger Absent state. A finger on the touchpad produces a difference signal of 15 counts, which results in an SNR of 15:1. Setting the centroid algorithm to resolve 20 positions between each row pair and each column pair, this touchpad system has a resolution of 100 counts per inch.

Proximity Sensors

The CapSense User Module does not directly support proximity sensors. A proximity sensor is implemented as a CapSense button with large C_P and small difference counts. A dedicated proximity sensor is best implemented as a single length of wire, as demonstrated in Figure 12. Connecting button and slider sensors already on the CapSense PCB into a single large sensor is another technique for implementing a proximity sensor. CSD is the best method to use for proximity sensing. CSD performs better than CSA with large C_P values, and the shield feature of CSD is used to extend the detection distance of the sense wire.

Figure 12. Rear View of a Proximity Sensor Prototype



Flex Circuits

Flex circuits work well with CapSense. All the guidelines presented for printed circuit boards also apply to flex. A flex circuit is typically much thinner than a PCB. Limit C_P by making the flex circuit no thinner than 0.01 inch (0.25 mm), and limiting trace lengths to a few inches. One good feature of flex is the high breakdown voltage provided by the Kapton material (290 KV/mm).

ITO Touch Screens

ITO is an acronym for Indium Tin Oxide. Films of this ceramic material are both electrically conductive and visually transparent. An example ITO touch screen is shown in Figure 13. Sheet resistivity of ITO films range from 0.25 to 1000 ohms/square, with the typical value of 100 to 500 ohms/square. Film thickness determines the sheet resistivity. Thinner film passes more light and has higher resistance.

Touch screens operate in either resistive or capacitive modes. Both modes have their niche markets. Resistive mode requires pressure for contact between conductive layers, and is prone to wear, tear, and breakage. Resistive mode is a four-layer solution with poor transparency (less than 75 percent). Capacitive mode makes use of the finger as a conductive object. Capacitive mode is a one-layer or two-layer solution with high transparency (greater than 90 percent). Cypress supports both touch screen technologies.

Figure 13. ITO Touch Screen



From Concept to Production: CapSense Tools and Techniques

Evaluation Boards and Example Firmware

The CY3213 board, shown in Figure 14, is an evaluation board for development of CapSense applications. Application firmware is written in 'C'. A library of common functions makes development of projects as simple as writing a few lines of code.

Figure 14. CY3213—CapSense Training Evaluation Board



Here is an example of the code required to scan two sensors in an array of buttons and save the result in an I2C array.

Code 1. Code to Scan Sensors

```
// starting with sensor0, scan 2 sensors,
single scan mode,
SensorArray_StartScan(0,2,0);

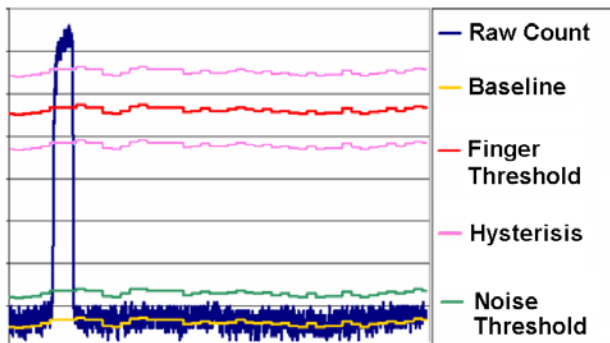
//scan complete?
while (!(SensorArray_GetScanStatus() &
SensorArray_SCAN_SET_COMPLETE));

//save sensor0 and sensor1results in i2c
array
info.iRowCount[0] =
SensorArray_waSnsResult[0];
info.iRowCount[1] =
SensorArray_iwaSnsResult[1];
```

Baseline Technique

The baseline is the reference for CapSense measurements. Each capacitive sensor has its own baseline. The baseline is a trend line for the capacitive sensor data that is computed by the Baseline function of the CapSense User Module. The raw count data is processed using an Infinite Impulse Response (IIR) low pass filter as shown in Figure 15. High level decisions, such as Finger Present and Finger Absent states, are based on the reference level established by the baseline.

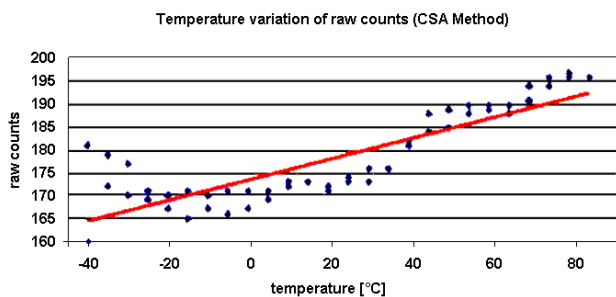
Figure 15. Baseline in CapSense Data is Continuously Updated



Environmental Effects

Temperature and Humidity: Temperature and humidity both cause the baseline counts to drift over time. The CapSense User Module is characterized from -40°C to +85°C, as shown in Figure 16. The trend that is tracked by the baseline automatically compensates for the effects of temperature and humidity.

Figure 16. Temperature Variation



Water: Use the CSD method if the sensing surface is exposed to a water stream or droplets. In this case, the shield electrode is enabled in the CSD User Module. When water film deposits on the overlay surface, the coupling between the shielding and sensing electrodes is increased. The shield electrode allows you to reduce the influence of parasitic capacitance, which gives more dynamic range to process sense capacitance changes.

Power Consumption and Sleep

Battery life is determined by mA-hours. The lower the average current, the longer CapSense operates between recharges (see AN2360). The PSoC is programmed such that it has different modes of power consumption:

- A fast response mode during intervals of constant button presses.
- A power saving, slow response mode after a period of inactivity.
- A deep sleep mode after a longer period of inactivity.

One of the strengths of PSoC, as compared to other capacitive sensing solutions, is its programmability. You can make the power saving modes of CapSense as sophisticated as required. CapSense buttons are fast, taking as little as 200 microseconds for each button scanned. This high scan speed is combined with a low sleep current to achieve very low average currents. One example of the real CapSense system is power saving, slow response mode using a three button scan every 100 milliseconds, while consuming less than 50 μ A of average current.

Noise Filtering

Noise is introduced into the CapSense system through both conductive and radiated sources. Conductive noise enters the system through power and signal lines. Radiated sources, such as cell phones or fluorescent lamp ballasts introduce noise through the air. Filtering techniques in firmware are used to increase the signal-to-noise ratio of the CapSense system in the presence of both types of noise. PSoC implements Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) digital filters with only a few lines of code.

FIR Filter: The frequency of finger pressing events is low compared to the frequency of power line noise. A low pass filter (LPF) is an effective noise filtering solution for this situation. An FIR LPF is defined by:

$$y = (x_1 + x_2 + \dots + x_N) / N \quad \text{Equation 1}$$

Raw counts are sampled N times per cycle of the noise. The N samples are combined together per Equation 1. With 50 Hz noise, the sample period must be 18 ms/N. The performance of the FIR filter increases with N, so make N as large as the system allows.

IIR Filter: An FIR filter has a disadvantage in that it needs to be a much higher order than an IIR filter to get the same result. It may also be difficult to adjust the sample rate to fit the period of the noise. So, in some cases, an IIR filter is a more appropriate choice for the LPF. Table 2 compares the FIR and IIR filters.

Table 2. Low Pass Filter Comparison—FIR versus IIR

Filter Type (order=N)	RAM	Response Time	Always Stable
FIR LPF	1	N*T	Yes
IIR LPF	N*2	T	No

RF Immunity Considerations

RF interferes with the operation of any capacitive sensing system, including CapSense (see AN2318). Where the electric field strength is high enough, RF interference causes false button presses or prevents real button presses from being sensed. A cell phone is a good example of where an RF transmitter and buttons are purposely combined in close proximity.

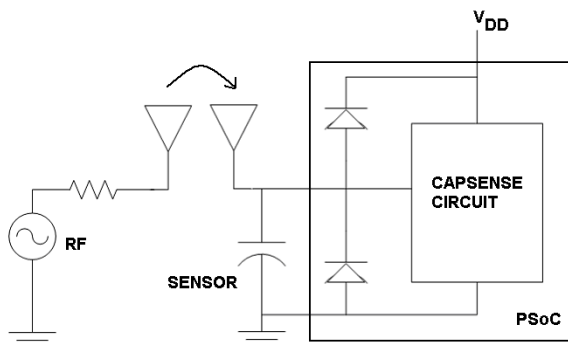
The electric field strength at a distance greater than 1/6 of a wavelength from a transmitter is approximated by Equation 2.

$$E = 6.85 * \frac{\sqrt{10^{(P/10)}}}{D} \quad \text{Equation 2}$$

- E [volts/meter] is the electric field.
- P [dBm] is the RF power fed to antenna.
- D [inches] is the distance from antenna to field sensor.

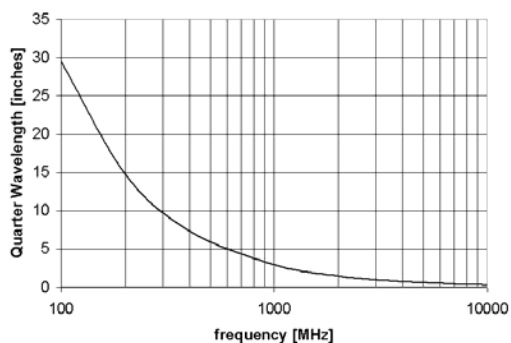
For an 800 MHz cell phone transmitting at +28 dBm (0.6W), the electric field 3 inches from the antenna is estimated to be around 60 V/m. Figure 17 shows the equivalent circuit of the RF interference scenario, with the PSoC configured to run CapSense. Internal diodes protect the PSoC from ESD events up to 2 KV.

Figure 17. Diodes at Input of PSoC Provide ESD Protection



The resonant effects of a circuit trace form the receiver antenna. A quarter wavelength trace is an efficient antenna. Figure 18 shows the length of a quarter wavelength antenna versus frequency.

Figure 18. Quarter Wavelength Trace is an Efficient Antenna



For a low level RF signal, the CapSense circuit operates with no effect on the digital output of the system, because low levels of RF look similar to background noise and are ignored by the system. When the RF power increases, the CapSense counts are offset a constant amount that is set by the power level of the interference. The RF signal is AC but the effect on CapSense counts is DC due to the diodes on the CapSense input. A positive shift in counts

causes false button presses. A negative shift prevents real button presses from being sensed. The finger and noise thresholds of the CapSense User Module enable normal operation in the presence of small shifts in the counts. For higher levels of RF interference, other measurements need to be taken. Following are two solutions.

- RF Solution 1 - Coordinate RF and CapSense: If the source of the interference is part of the same system in which CapSense is embedded, then disable CapSense when RF is transmitting. One pin on the PSoC is assigned to monitor a Transmit_Enable signal. CapSense counts continue to be affected by the high power RF, but counts are only valid with the transmitter turned off.
- RF Solution 2 - Damp the Resonance: Resistors placed in series with the CapSense input damps the resonance of each trace. The recommended series resistance added to the CapSense inputs is 560 ohms. Communication lines, I2C and SPI, benefit from 300 ohms in series.

ESD Considerations

The electrostatic voltage on the human body reaches 15 KV when the humidity is low. The type of clothes worn by the CapSense user makes a difference on how this voltage goes, as shown in Figure 19.

Figure 19. Electrostatic Voltage on a Human Body vs. Relative Humidity and Material Type

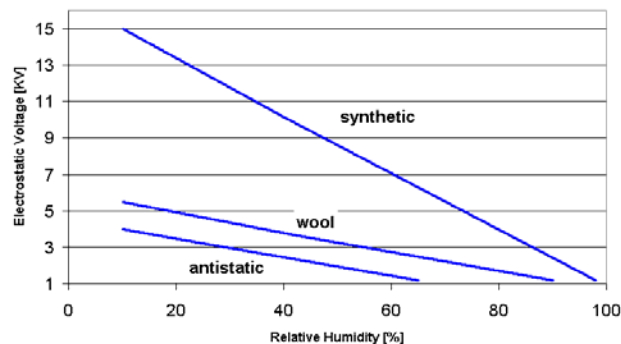


Table 3 shows the minimum thickness required to withstand 12 KV for common overlay materials. The overlay in the CapSense system protects the PSoC from permanent damage when the thickness guidelines of the table are followed. A layer of Kapton tape works well in applications needing extra ESD protection.

Table 3. Breakdown Voltage of Overlay Materials and Minimum Thickness to Prevent Breakdown

Material	Breakdown Voltage [V/mm]	Min. Overlay Thickness at 12 KV [mm]
Air	1200 – 2800	10
Glass – Common	7900	1.5
Glass - Borosilicate (Pyrex)	13,000	0.9
Formica	18,000	0.7
ABS	16,000	0.8
Acrylic (Plexiglass)	13,000	0.9
Polycarbonate (Lexan)	16,000	0.8
PET Film (Mylar)	280,000	0.04
Polyimide Film (Kapton)	290,000	0.04
FR-4	28,000	0.4
Wood – Dry	3900	3

Summary

The best practices for CapSense designs presented in this application note enable engineers to successfully add capacitive sensing features to their products.

About the Author

Name: Mark Lee

Title: Principal Application Engineer

Background: Ph.D.E.E. 1992

University of Washington

Computer Aided Modeling of the Dielectric Properties of Materials

Experienced in aerospace, medical, and computer industries, focusing on new product development. Holder of several patents.

Cypress employee since 2005.

Contact: olr@cypress.com

Document History

Document Title: CapSense™ Best Practices

Document Number: 001-41450

Revision	ECN	Orig. of Change	Submission Date	Description of Change
**	1541809	XSG	10/04/07	New publication of existing application note.
*A	1664045	OLR	10/22/07	Figures 7, 9, and 13 revised Section on Signal-to-Noise Ratio removed Edited entire document as per Cypress styles Title has been changed to CapSense (TM) Best Practices
*B	1940787	OLR	01/16/08	The artwork in "Figure 9. Partial Ground Fill to Minimize CP" has been changed to reflect the correct grid spacing for the Top and Bottom layers Updated spec title to the right format in DMS
*C	2640992	OLR	01/20/09	Added part number CY8C20xx6 and updated the software version to PSoC Designer 5.0. Made document more general purpose by removing some user module references

In March of 2007, Cypress recataloged all of its application notes using a new documentation number and revision code. This new documentation number and revision code (001-xxxxx, beginning with rev. **), located in the footer of the document, will be used in all subsequent revisions.

PSoC is a registered trademark of Cypress Semiconductor Corp. "Programmable system-on-Chip," PSoC Designer, and PSoC Express are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein may be the property of their respective owners.

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
Phone: 408-943-2600
Fax: 408-943-4730
<http://www.cypress.com/>

© Cypress Semiconductor Corporation, 2006-2009. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

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

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

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