



电容式触摸感应的技巧

By (赛普拉斯半导体公司高级应用工程师 Mark Lee)

触摸传感器被广泛使用已经有很多年了，不过，混合信号可编程器件的近期发展使得电容式触摸传感器在众多消费类产品中都成为了机械式开关的一种实用、增值型替代方案。本文将粗略地介绍一种可透过一层厚玻璃覆盖物来激活的触摸感应式按钮的设计实例。

典型的电容式传感器设计所规定的覆盖物厚度为 **3mm** 或更薄。随着覆盖物厚度的增加，透过一层覆盖物来检测手指的触摸将变得越来越困难。换句话说，伴随着覆盖物厚度的增加，系统调整的过程将从“科学”走向“技巧”。为了说明如何制作一个能够提升当今技术极限的电容式传感器，在本文所述的实例中，玻璃覆盖物的厚度被设定为 **10mm**。玻璃易于使用，购买方便，而且是透明的（因此您可以看到位于其下方的金属感应垫）。玻璃覆盖物还被直接应用于白色家电。

手指电容

所有电容式触摸感应系统的核心部分都是一组与电场相互作用的导体。人体组织中充满了覆盖着一层皮肤（这是一种有损电介质）的导电电极。正是手指的导电特性使得电容式触摸感应成为可能。

简单的平行片电容器具有两个导体，其间隔着一层电介质。该系统中的大部分能量直接聚集在电容器极板之间。少许能量会溢出至电容器极板以外的区域中，而与该效应相关的电场线被称为“边缘场”。制作实用电容式传感器的部分难题是：需要设计一组印刷电路走线，以便将边缘场引导至一个用户可以够得到的有效感应区域。对于这样一种传感器模式来说，平行片电容器并非上佳之选。

把手指放在边缘电场的附近将增加电容式系统的导电表面积。由手指所产生的额外电荷积聚电容被称为手指电容 C_F 。在本文中，无手指触摸时的传感器电容用 C_P 来表示，它代表寄生电容。

关于电容式传感器的一个常见的误解是：为了使系统正常工作，手指必需接地。手指是可以检测到的，因为它会保存电荷（而在手指浮置或接地时都将产生这种现象）。

传感器的 PCB 布局

图 1 示出了一块印刷电路板（PCB）的顶视图，在本设计实例中，该 PCB 实现了其中的一个电容式传感器按钮。该按钮的直径为 10mm，这是一个成人指尖的平均大小。为该演示电路而组装的 PCB 包含 4 个按钮，它们的中心相隔 20mm。如图所示，接地平面也位于顶层。金属感应垫和接地平面之间设置了一个均匀的隔离间隙。该间隙的尺寸是一个重要的设计参数。如果间隙设置得过小，则过多的电场能量将直接传递至地。而如果间隙设置得过大，则将无法控制能量穿越覆盖物的方式。选择 0.5mm 的间隙尺寸可以很好地使边缘场透过 10mm 厚的玻璃覆盖物。

图 1: PCB 的顶视图

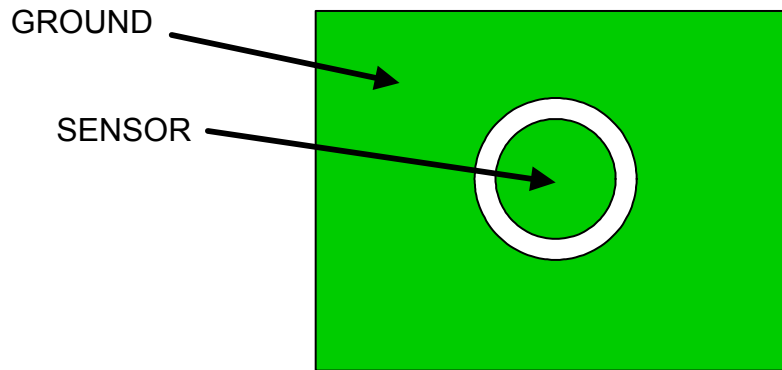
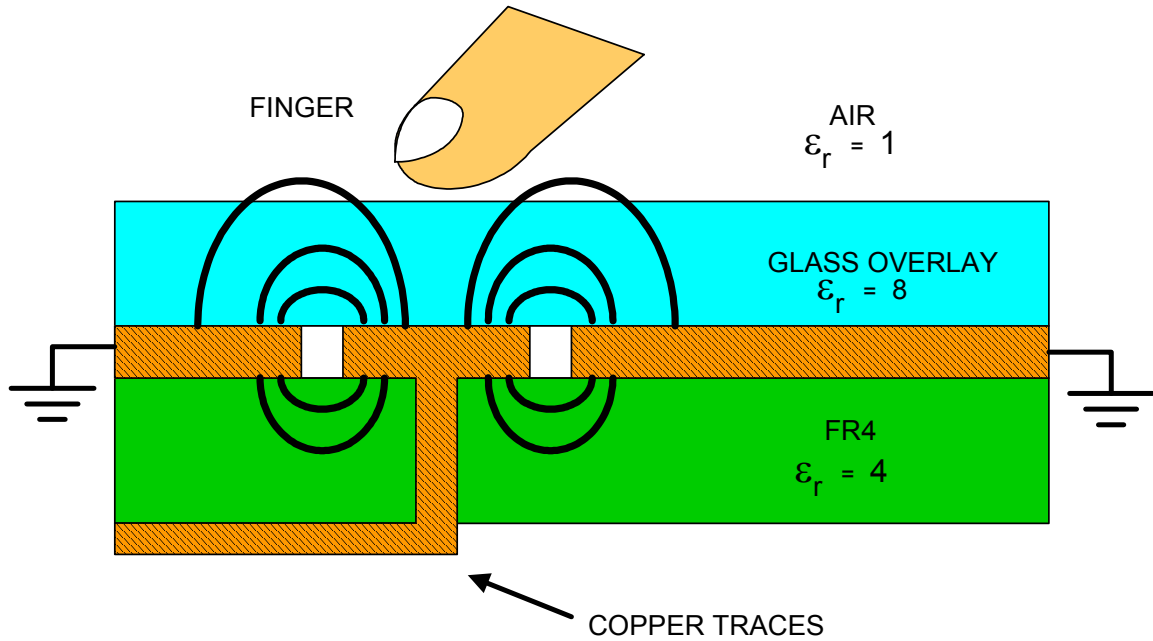


图 2 示出了同一种传感器模式的截面图。如该图所示，PCB 中的一条通路将金属感应垫与电路板底面上的走线相连。当电场试图找到最短的接地路径时，介电常数 ϵ_r 将对材料中的电场能量充填密度产生影响。标准窗户玻璃的 ϵ_r 约为 8，而采用 FR4 材料制成的 PCB 的 ϵ_r 则在 4 左右。白色家电中常用的 Pyrex[®]（派莱克斯）玻璃具有数值大约为 5 的 ϵ_r 。在本设计实例中，采用的是标准的窗户玻璃。需要注意的是，玻璃片是采用 3M 公司的 468-MP 绝缘胶膜安装在 PCB 上的。

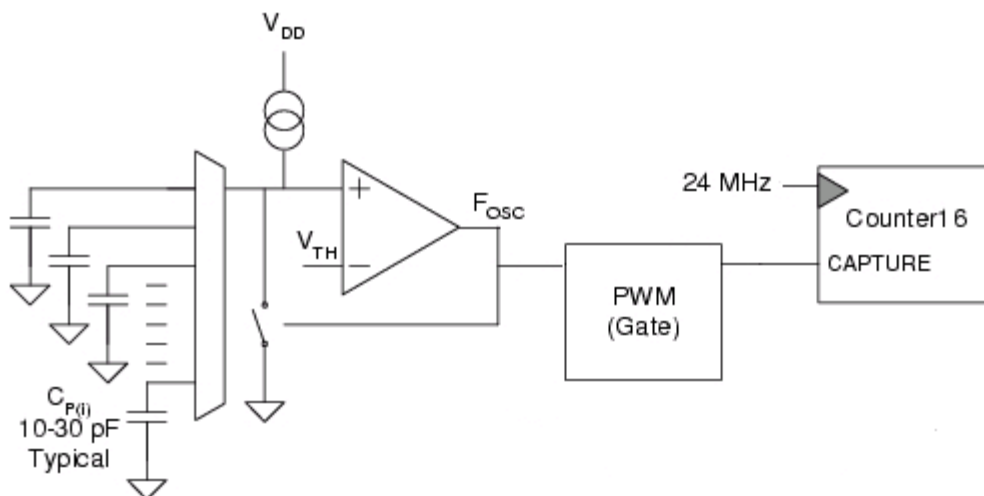
图 2: PCB 和覆盖物的截面图



CapSense 101

电容式触摸感应系统的基本元件是：一个可编程电流源、一个精密型模拟比较器和一根可通过一组电容式传感器进行排序的模拟多路复用器总线。在本文给出的系统中，一个弛张振荡器起着电容传感器的作用。该振荡器的简化电路示意图见图 3。

图 3: 弛张振荡器电路



比较器的输出被馈入一个 PWM 的时钟输入，该 PWM 负责对一个时钟频率为 24MHz 的 16 位计数器进行选通。触摸按键的手指使电容增大，从而导致计数值增加。手指就是以这样的方式来检测的。该系统的典型波形示于图 4。

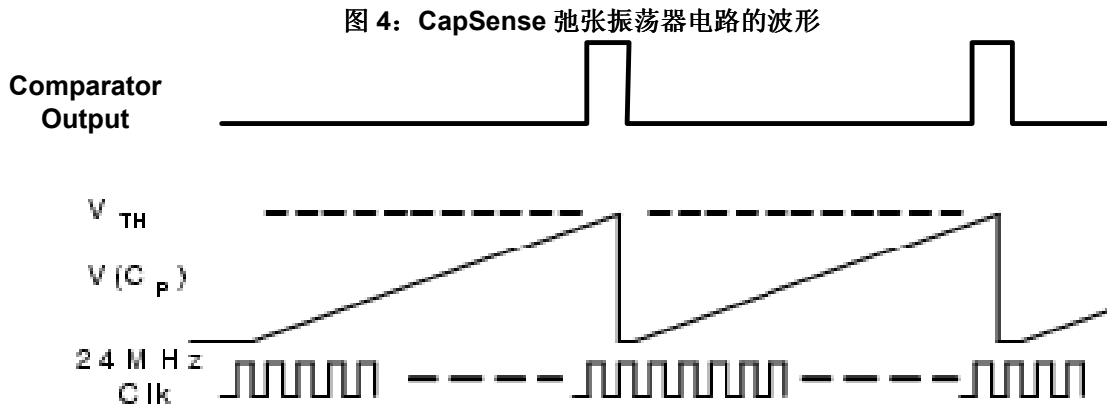
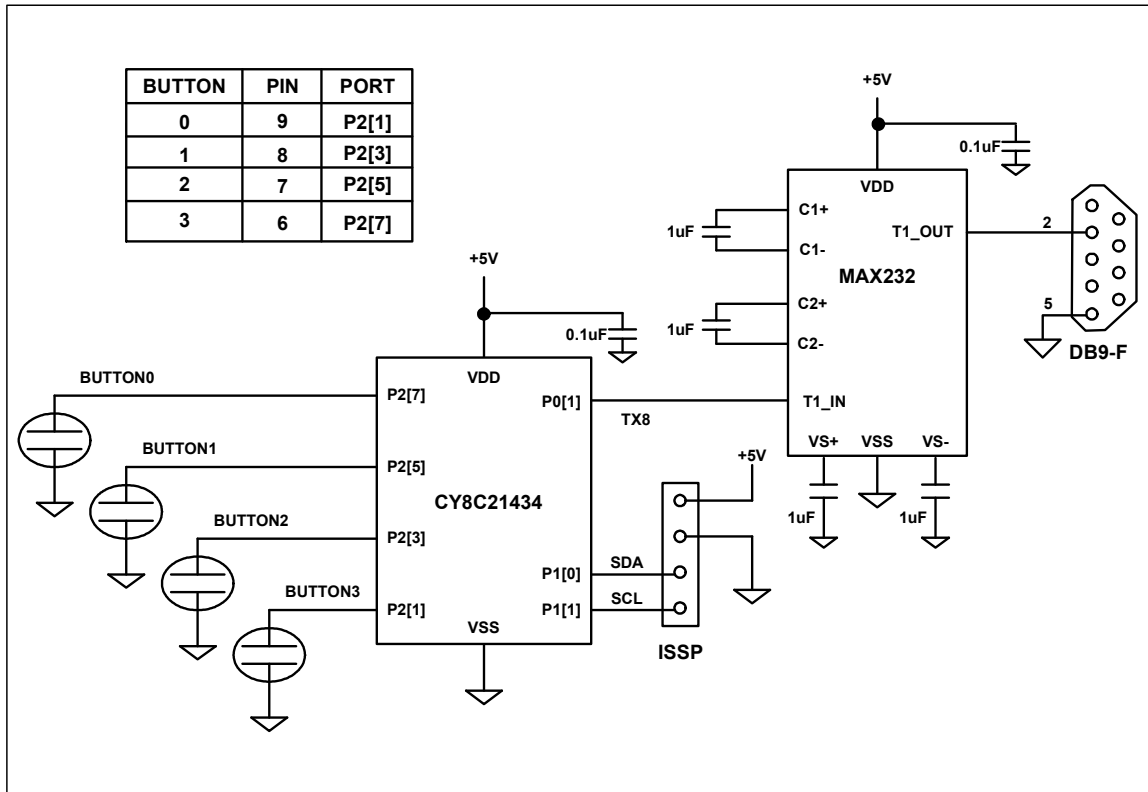


图 5 给出了该项目的一种可实现方案的示意图。为了实现电容式触摸感应和串行通信，该电路采用了赛普拉斯的 CY8C21x34 系列 PSoC 芯片，该芯片包含一组模拟和数字功能块，这些功能块可由存储于板上闪存中的固件来配置。另一颗芯片负责处理 RS232 电平移动，旨在提供至主机的通信链接，并实现 115,200 波特的电容式触摸感应数据记录。PSoC 是通过 ISSP 头（包含电源、地）以及编程引脚 SCL 和 SDA 来编程的。主 PC 通过一个 DB9 连接器与电容式触摸感应电路板相连。

图 5：电容式触摸感应电路示意图



PSoC 采用固件来配置，以采用一个 5V 工作电源和一个内部生成的 24MHz 系统时钟。对该 24MHz 时钟进行 1:26 分频，以提供一个用于 115,200 波特 TX8 模块的时钟。CapSense 用户模块选择以“周期法”（Period Method）来运行，在该工作模式中，计数在固定数量的弛张振荡器周期中累加。换言之，16 位计数器值代表了一个与传感器电容成正比的周期。

清单 1 罗列了系统固件。与设立电容式触摸感应系统相关的大部分工作都已被编码为一组由 C 程序来调用的标准 CSR 例行程序。例如：CSR_1_Start()负责配置 PSoC 的内部布线，以使电流源 DAC 与模拟多路复用器相连，而比较器与经过正确初始化的 PWM 和 16 位计数器相连。

清单 1：用于电容式触摸感应系统的固件

```
//-----start of listing-----
//-----
```



```
// main.c, a CapSense program in C
//   A demonstration of Capacitive Sensing with PSoC
//   with a 10mm glass overlay
//-----
#include <m8c.h>          // part specific constants and macros
#include "PSoCAPI.h"     // PSoC API definitions for all User Modules

void main()
{
    //a flag that is set when a finger is on any buttons
    int bBaselineButtonFlag;

    CSR_1_Start(); //initialize CapSense user module
    TX8_1_Start(TX8_1_PARITY_NONE); //initialize TX8 module
    M8C_EnableGInt; //enable global interrupts

    CSR_1_SetDacCurrent(200,0); //set current source to 200 out of 255
                                //use low range of current source
    CSR_1_SetScanSpeed(255); //set number of osc cycles to 255-2=253

    while(1)
    {
        CSR_1_StartScan(1,1,0); //scan one button only, button 1 on P2[3]
        //wait for scanning of button to complete
        while (!(CSR_1_GetScanStatus() & CSR_1_SCAN_SET_COMPLETE));

        //update baseline if required, set flag if any button pressed
        bBaselineButtonFlag = CSR_1_bUpdateBaseline(0);

        //data log the raw counts on button 1
        TX8_1_PutSHexInt(CSR_1_iaSwResult[1]);
        TX8_1_PutChar(',');

        //data log switch mask... which switch is on?
        TX8_1_PutSHexInt(CSR_1_baSwOnMask[0]);
        TX8_1_CPutString(",");

        //data log switch difference = raw counts - baseline
        TX8_1_PutSHexInt(CSR_1_iaSwDiff[1]);
        TX8_1_PutChar(',');

        //data log update timer as a teaching aid
        TX8_1_PutSHexInt(CSR_1_bBaselineUpdateTimer);
        TX8_1_PutChar(',');

        //data log the baseline counts for button 1
        TX8_1_PutSHexInt(CSR_1_iaSwBaseline[1]/4);

        TX8_1_PutCRLF();
    }
}
```

```
    }  
}  
//-----end of listing-----
```

调整传感器

每次在上列程序中调用函数 `CSR_1_StartScan()` 时，均对 `Button1` 的电容进行测量。原始计数值被存储于 `CSR_1_iaSwResult[]` 阵列中。用户模块还跟踪一个用于原始计数的基线。每个按钮的基线值均为一个由 IIR 滤波器采用软件进行周期性计算的平均原始计数值。IIR 滤波器的更新速率是可编程的。基线使得系统能够适应由于温度和其他环境影响而在系统中引起的漂移。

开关差分阵列 `CSR_1_iaSwDiff[]` 包含消除了基线偏移的原始计数值。按钮目前的 ON/OFF 状态采用开关差值来决定。这可使系统的性能保持恒定，即便在基线有可能随着时间的推移而发生漂移的情况下也是如此。

图 6 示出了采用固件来实现的差分计数与按钮状态之间的转移函数。该转移函数中的迟滞提供了 ON 和 OFF 状态之间的干净转换，即使计数是有噪的也不例外。这提供了一种针对按钮的除跳功能。较低的阈值被称为“噪声阈值”，而较高的阈值则被称为“手指阈值”。阈值水平的设定决定了系统的性能。在覆盖物非常厚的场合，信噪比很低。在此类系统中设定阈值水平是一项具有挑战性的工作，而这恰好是电容式触摸感应技巧的一部分。

图 6: 差分计数与按钮状态之间的转移函数

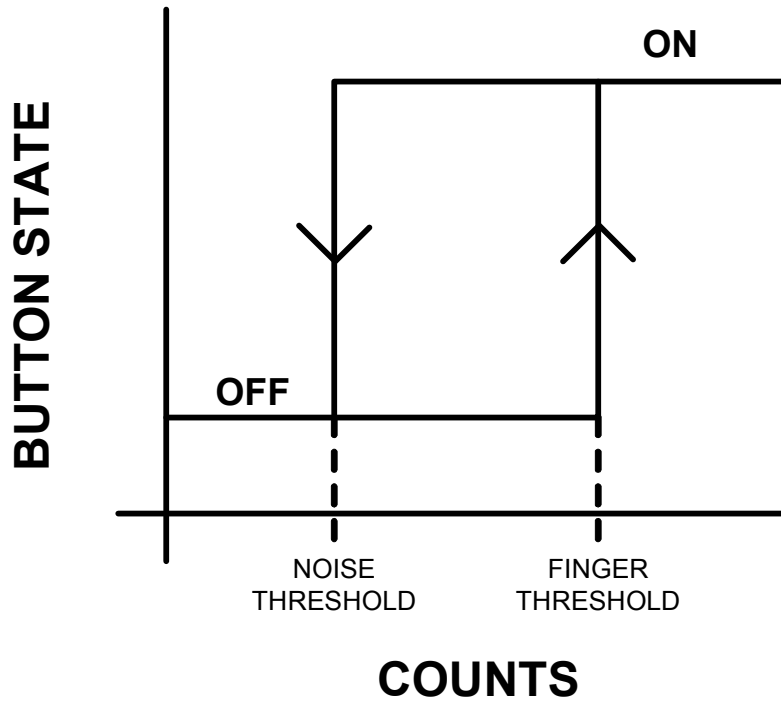
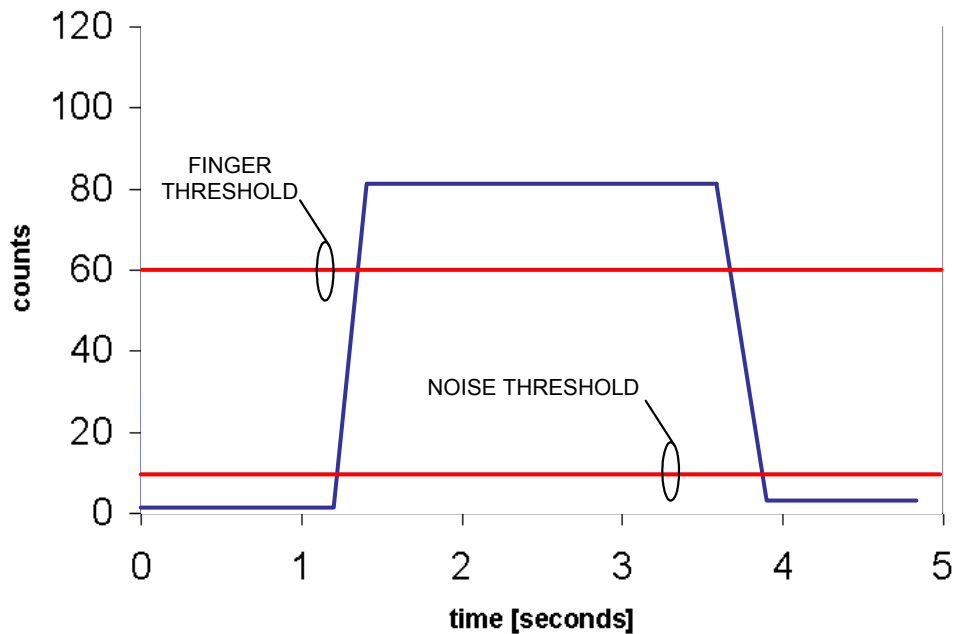


图 7 示出了一个持续时间为 3 秒的按钮触压操作的理想化原始计数波形。针对本项目的阈值水平示于此图。噪声阈值被设定为 10 个计数，而手指阈值被设定为 60 个计数。为了清晰地显示阈值水平，图 8 并未示出始终存在于实际计数数据之中的噪声分量。

图 7: 把阈值水平放置在一个去除了基线的原始计数曲线图上



电流源 DAC 的电流水平选择和用于计数累加的振荡器周期数的设定是调整过程的一部分。在固件中，函数 `CSR_1_SetDacCurrent(200,0)` 把电流源设定在其低电流水平范围内，数值为 200（最高 255），大约对应于 $14\ \mu\text{A}$ 。函数 `CSR_1_SetScanSpeed(255)` 把振荡器周期数设定为 253（ $255-2$ ）。原始计数和差分计数的分析表明：该系统具有一个约 15pF 的寄生走线电容 C_P 和一个 0.5pF 左右的手指电容 C_F 。手指使总电容产生了约 3% 的变化。每个原始计数值的采集仅需 $500\ \mu\text{s}$ 的时间（每个按钮）。

测量性能

电容式触摸感应系统的测量性能示于图 8。差分计数通过一个终端仿真程序在主 PC 上捕获，然后借助电子制表软件加以绘制。手指放置在 10mm 厚的玻璃覆盖物上，并持续 3 秒的时间。按钮的 ON/OFF 状态被叠加在原始计数上。按钮在这两种状态之间干净地转换，即使存在因通过厚玻璃进行检测而产生的比较嘈杂的原始计数信号时也是如此。请注意手指和按钮阈值是如何随着基线的漂移而进行周期性调整的。当检测到手指的触压动作时，基线值将锁定其数值，直到手指移开为止。

图 8：通过 10mm 厚的玻璃进行检测时传感器的测量性能

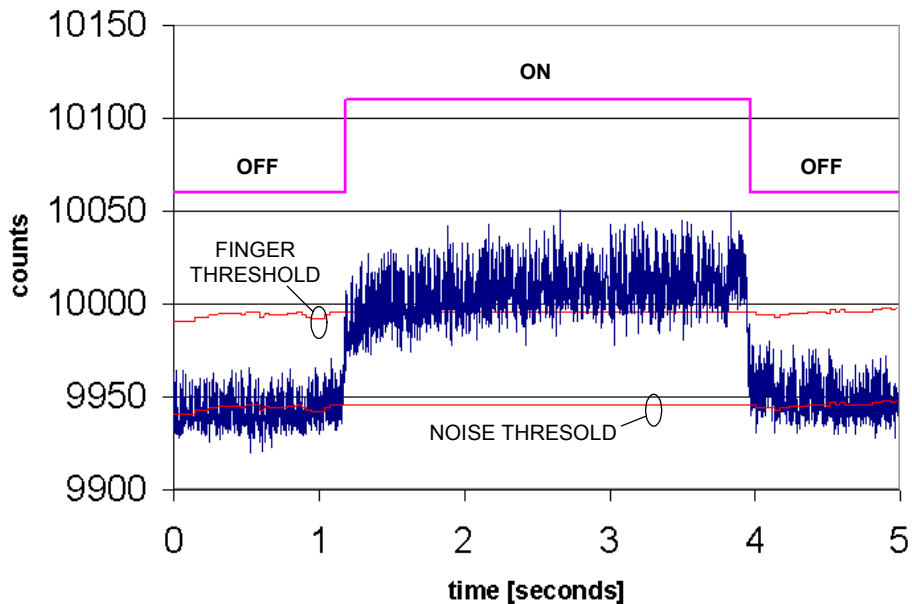


图 9 和图 10 示出了每种状态转换的细部视图。在图 9 中，按钮状态一开始为“OFF”（关闭）。超过手指阈值的差分计数的第一个采样把按钮状态转换至“ON”（接通）。在图 10 中，利用低于噪声阈值的差分计数的第一个采样将按钮转换至 OFF 状态。

图 9: 至 ON 状态的转换的特写图

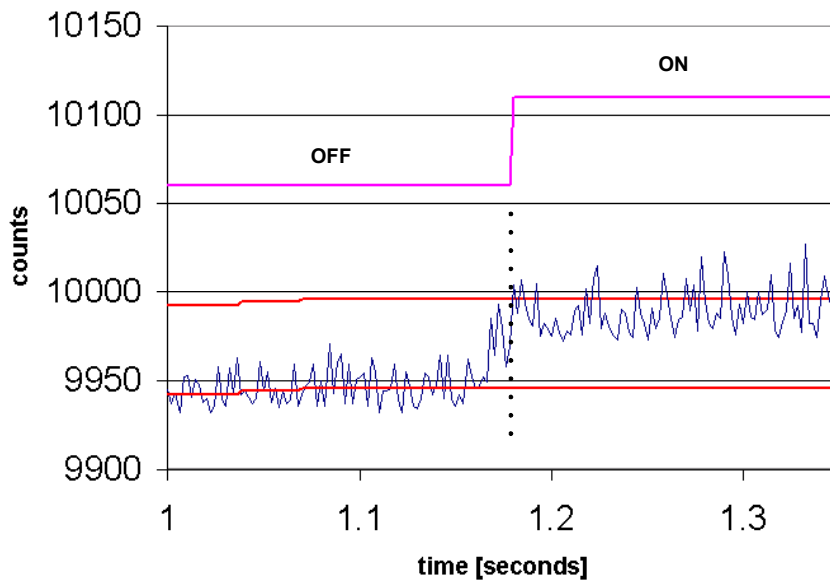
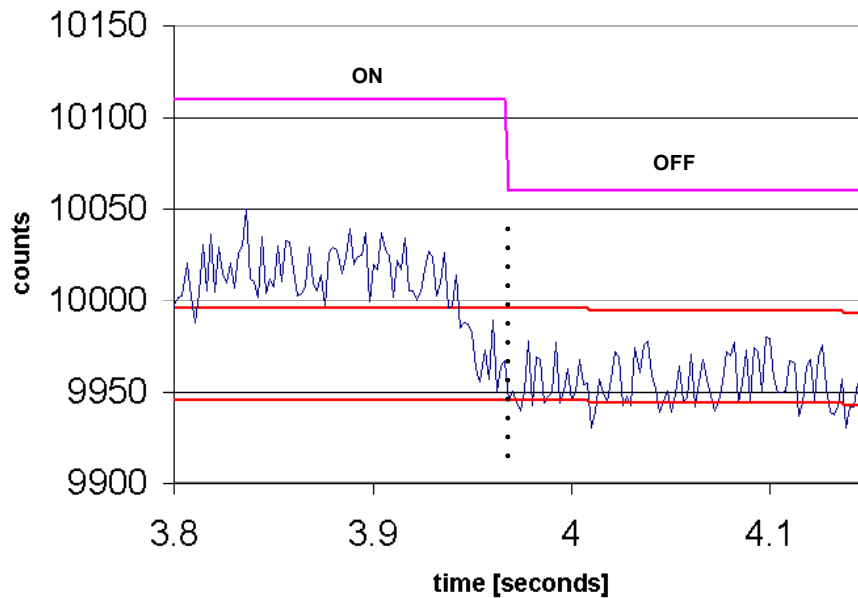


图 10: 至 OFF 状态的转换的特写图





与机械式开关相比，基于电容的触摸传感器的主要优点是耐用性好，不易损坏。混合信号技术的近期发展不仅使得触摸式传感器的费用降到了一个经济划算的水平（故可在各种消费类产品中实现），而且还提高了检测电路的灵敏度和可靠性（因而增加了覆盖物的厚度和耐用性）。利用本文介绍的设计方法，即可通过一个 10mm 的玻璃来检测手指对按键的触压，并借助基于噪声阈值和手指阈值的除跳法实现了 ON 和 OFF 按钮状态之间的干净转换，从而令电容式触摸传感器成为机械式开关元件的一种实用型替代方案。



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, 2007. 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.

PSoC Designer™, Programmable System-on-Chip™, and PSoC Express™ are trademarks and PSoC® is a registered trademark of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

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.