



## L'arte del rilevamento tattile capacitivo

Mark Lee, Senior Application Engineer, Cypress Semiconductor Corp.

### Sommario

I sensori tattili sono in circolazione da anni ma i recenti progressi compiuti nel campo dei dispositivi programmabili mixed signal stanno rendendo queste soluzioni - basate sull'effetto capacitivo - una valida alternativa a valore aggiunto agli switch meccanici in un ampio spettro di prodotti consumer. Questo articolo prende in considerazione un esempio di progetto di tasto touch-sensitive comandato attraverso uno strato di copertura in vetro di un certo spessore.

### Introduzione

Normalmente, questi progetti, specificano una copertura di 3 mm o meno. Rilevare la presenza di un dito attraverso lo strato di copertura è tanto più difficile quanto più lo spessore cresce. In altre parole, quando lo spessore dello strato aumenta, il processo di taratura del sistema si sposta dal terreno della scienza a quello dell'arte. Per dimostrare dove sono stati spinti i limiti tecnologici dei sensori capacitivi, lo spessore dello strato di copertura previsto in questo esempio sarà pari a 10 mm

Lavorare col vetro è semplice: si tratta di un materiale disponibile ovunque ed è trasparente, caratteristica che permette di individuare facilmente i pad del sensore sottostante. Il vetro ha inoltre un'applicazione concreta e diretta nel campo dei prodotti "bianchi".

### La capacità delle dita

Cuore di qualsiasi sistema di rilevamento capacitivo è un set di conduttori che interagisce con i campi elettrici. Il tessuto del corpo umano è saturo di elettroliti conduttivi; questi sono coperti da uno strato di pelle, materiale che si può assimilare a un blando dielettrico. Ed è proprio la natura conduttiva delle dita che rende possibile il rilevamento tattile. Nella sua forma più semplice un condensatore a superfici parallele, prevede due conduttori separati da un dielettrico. La maggior parte dell'energia di questo sistema è concentrata direttamente tra le superfici. Una quota di energia però fuoriesce nell'area esterna e dà origine a delle linee di campo elettrico denominate "**fringing fields**", campi dispersi. Parte della sfida associata realizzazione pratica di un sensore capacitivo è di progettare, sul circuito stampato, un set di tracce che dirigono i loro campi dispersi all'interno dell'area di rilevamento tattile accessibile all'utilizzatore. Per questo tipo di schema, un condensatore a superfici parallele non rappresenta certo una buona scelta.

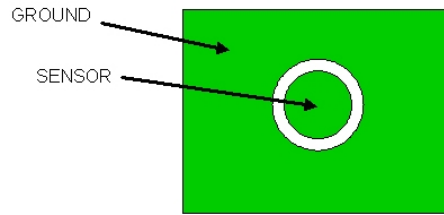
Posizionando il dito in prossimità dei campi elettrici dispersi ha come effetto quello di aggiungere un'area capacitiva all'intero sistema. La capacità di immagazzinamento della carica aggiunta dal dito è nota col termine "finger capacitance",  $C_F$ .

In questo articolo la capacità del sensore senza la presenza del dito è identificata dal valore  $C_P$ , cioè "capacità parassita".

Un malinteso comune sui sensori capacitivi riguarda la convinzione che per permettere al sistema di lavorare, le dita devono essere in qualche modo connesse a massa. In realtà il dito può essere rilevato in quanto trasportatore di carica, e questo accade sia se è flottante, sia se è a massa.

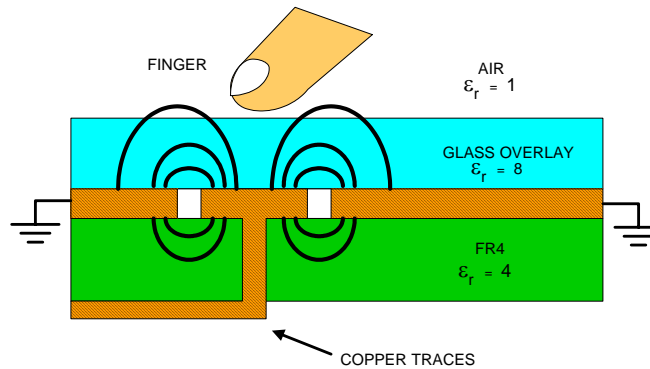
### Layout del PCB del sensore

La figura 1 riporta la vista dall'alto di un circuito stampato (PCB) che implementa uno dei pulsanti a sensore capacitivo del progetto preso ad esempio. Il diametro è di 10mm, pari alla dimensione media dell'impronta digitale di un adulto. Il PCB assemblato per questa dimostrazione contiene quattro pulsanti spazati di 20mm. Come illustrato in figura, sullo strato superiore è presente il piano di massa. L'area del sensore è isolata dal piano di massa da un gap uniforme. Le dimensioni del gap sono un parametro di progetto importante. Se il gap è troppo ridotto, verrà inviata a massa una quantità eccessiva di energia. Se è troppo ampio, viene a mancare il controllo su come è diretta l'energia attraverso la copertura. Le dimensioni scelte per il gap dell'esempio sono pari a 0.5mm e sono adeguate per dirigere i campi dispersi attraverso la copertura di vetro da 10mm.



**Figura 1. Vista dall'alto del PCB**

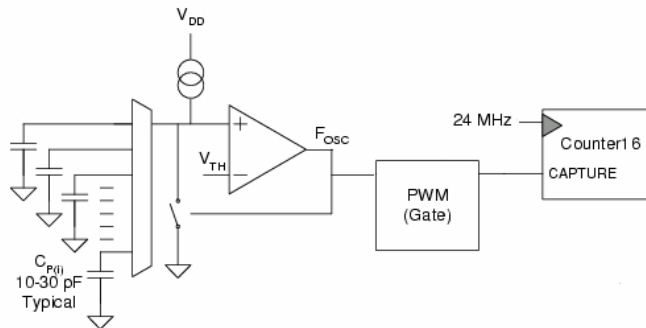
La figura 2 riporta la sezione trasversale del sensore. Come illustrato nella figura, una "via" nel PCB connette l'area del sensore con la traccia sul lato superiore della scheda. La costante dielettrica  $\epsilon_r$  influenza la quantità di carica che può essere immagazzinata nel materiale quando il campo tenta di trovare la strada più breve verso massa. Una finestra di vetro standard ha una costante  $\epsilon_r$  pari circa a 8, mentre il materiale FR4 del PCB ha una  $\epsilon_r$  di circa 4. Il vetro Pyrex®, utilizzato normalmente nei beni bianchi, ha una  $\epsilon_r$  intorno a 5. Nel nostro esempio è stato utilizzato un normale vetro da finestra. Notare che il foglio di vetro è montato sul PCB utilizzando il film adesivo non conduttivo 468-MP di 3M.



**Figura 2. Vista trasversale del PCB e della copertura**

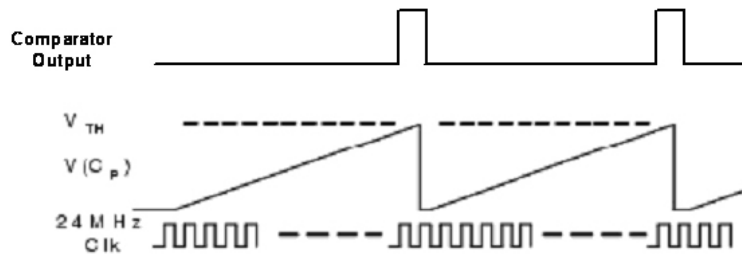
### CapSense 101

I componenti fondamentali del sistema di rilevamento capacitivo sono delle fonti di corrente programmabili, un comparatore analogico di precisione e un bus con mux analogico che permette di interrogare tutto l'array di sensori capacitivi. Nel sistema presentato in questo articolo un oscillatore a rilassamento opera come sensore di capacità. Uno schema semplificato dell'oscillatore è riportato in figura 3.



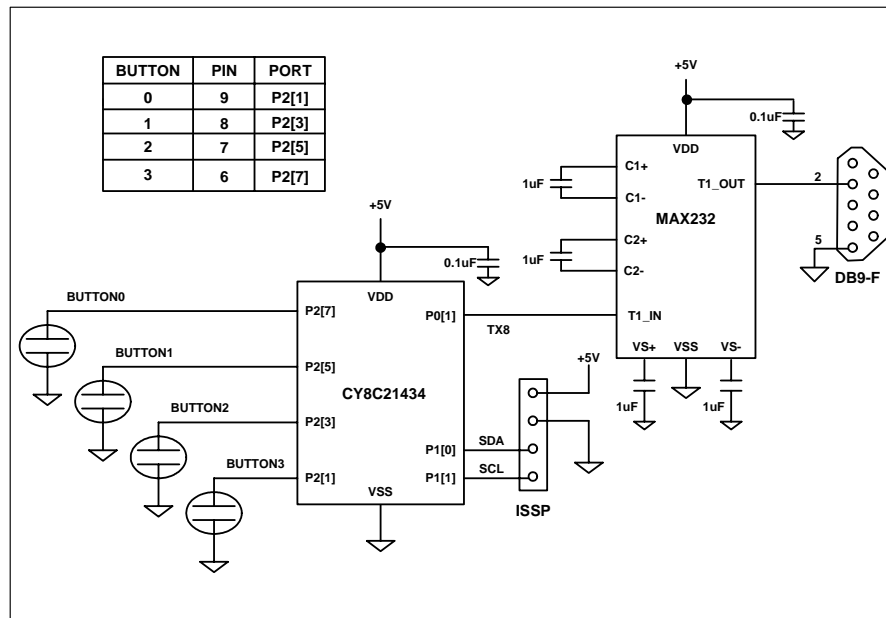
**Figura 3. Circuito oscillatore a rilassamento.**

L'uscita del comparatore è inviata all'ingresso del clock di un PWM che alimenta un contatore a 16-bit temporizzato a 24MHz. Un dito sul sensore incrementa la capacità, incrementando in tal modo i conteggi. Questa è la modalità di rilevamento del dito. Le forme d'onda tipiche di questo sistema sono riportate in figura 4.



**Figura 4. Forme d'onda del circuito oscillatore a rilassamento di CapSense**

Lo schema dell'implementazione di questo progetto è riportata in figura 5. Per il rilevamento capacitivo e per le comunicazioni seriali il circuito utilizza il chip PSoC Cypress della serie CY8C21x34, costituito da un set di blocchi funzionali analogici e digitali che vengono configurati dal firmware memorizzato nella memoria flash on-board. Un secondo chip gestisce il trasferimento a livello RS232 che mette a disposizione un link di comunicazione verso il computer host, consentendo l'acquisizione dei dati del rilevamento capacitivo a 115,200 baud. L'assegnazione dei pin per i quattro tasti CapSense è riportata nella tabella di figura 5. Il PSoC è programmato attraverso l'header ISSP che contiene alimentazione, massa e i pin di programmazione SCL e SDA. Il PC host si collega alla scheda di rilevamento capacitivo attraverso un connettore DB9.



**Figura 5. Schema del circuito di rilevamento capacitivo**

Il PSoC è configurato a livello firmware per operare con un'alimentazione a 5V e con un clock di sistema a 24MHz generato internamente. Il clock a 24MHz è diviso per 26 per fornire una temporizzazione al modulo TX8 da 115,200 baud. Il CapSense User Module è predisposto per operare in "Period Method", una modalità di funzionamento in cui i conteggi sono accumulati nell'arco di un numero fisso di cicli dell'oscillatore a rilassamento. In altri termini, il valore del contatore a 16-bit rappresenta un periodo proporzionale alla capacità del sensore.

Il listato 1 riporta il codice del firmware di sistema. La maggior parte del lavoro relativo al set up del sistema di rilevamento capacitivo è stato codificato in un set di routine CSR standard che vengono chiamate da un programma in C. Per esempio, CSR\_1\_Start() configura il routing interno del PSoC in modo che la fonte di corrente DAC sia connessa al mux analogico e che il comparatore sia connesso al PWM appropriatamente inizializzato e al contatore a 16 bit.

### Listing 1: Firmware per un sistema di rilevamento capacitivo

```
//-----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();

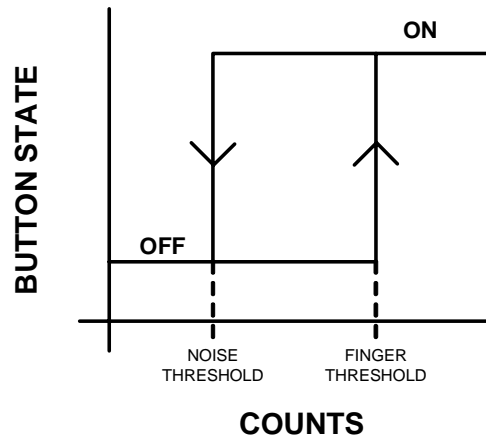
    }
}
//-----end of listing-----
```

## Regolazione del sensore

Ogni volta che viene chiamata la funzione `CSR_1_StartScan( )` dal programma listato sopra, viene misurata la capacità del tasto 1 (Button1). I valori assoluti conteggiati sono memorizzati nell'array `CSR_1_iaSwResult[]`. Il modulo utente traccia anche un valore di partenza per i conteggi assoluti. Il valore della linea di partenza di ciascun tasto è un livello medio di conteggio assoluto elaborato periodicamente da un filtro IIR implementato nel software. La velocità di aggiornamento del filtro IIR è programmabile. Il valore di partenza permette al sistema di adattare le derive dovute alla temperatura e ad altri effetti ambientali.

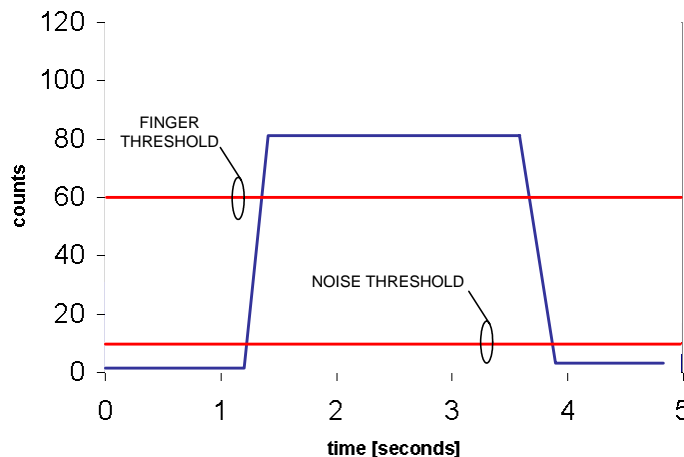
L'array "switch difference" `CSR_1_iaSwDiff[]` contiene i valori di conteggio assoluti meno l'offset di partenza. Lo stato ON/OFF dei tasti è determinato utilizzando i valori delle differenze. Ciò consente di mantenere costanti le prestazioni del sistema spostando nel tempo il valore della linea di partenza.

La figura 6 riporta la funzione di trasferimento tra i conteggi delle differenze e lo stato del tasto, implementata nel firmware. L'isteresi di questa funzione di trasferimento offre delle transizioni chiare tra gli stati di ON e OFF, anche in presenza di conteggi rumorosi. Questo garantisce anche la funzione antirimbazzo dei tasti. La soglia inferiore è chiamata Noise Threshold mentre la soglia superiore è chiamata Finger Threshold (soglia di presenza del dito). La predisposizione dei livelli di soglia determina le prestazioni del sistema. Con coperture molto spesse, il rapporto segnale-rumore è basso. Predisporre i livelli di soglia in questo tipo di sistema è impegnativo e fa parte dell'arte del rilevamento capacitivo.



**Figura 6. Funzione di trasferimento tra conteggio differenze e stato del tasto**

La forma d'onda ideale dei conteggi relativi a una pressione del tasto di 3 secondi è presentata in figura 7. In figura sono riportati i livelli delle soglie di questo progetto. La Noise Threshold è predisposta a 10 conteggi; la Finger Threshold è predisposta a 60 conteggi. La componente di rumore sempre presente nel dato del conteggio reale non è riportata nella figura 8: ciò consente di vedere chiaramente i livelli delle soglie.

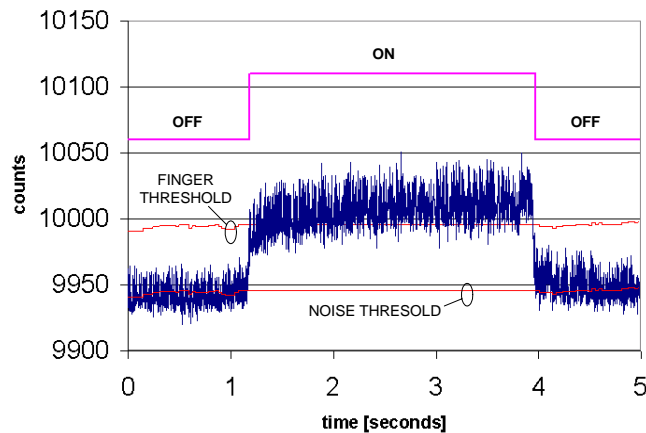


**Figure 7. Livelli di soglia posizionati su una curva di conteggi assoluti con linea di partenza rimossa.**

Parte del processo di regolazione include la selezione del livello della fonte di corrente DAC e la predisposizione del numero di cicli di oscillazione per accumulare i conteggi. Nel firmware, la funzione `CSR_1_SetDacCurrent(200,0)` predispose la fonte di corrente a un livello pari a 200 su 255, circa 14uA. La funzione `CSR_1_SetScanSpeed(255)` predispose il numero di cicli di oscillazione a 253 (255-2). L'analisi dei conteggi assoluti e dei conteggi di differenza mostra che il sistema ha una capacità parassita dovuta alle tracce -  $C_P$  - di circa 15pF, e una capacità "finger" -  $C_F$  - di circa 0.5pF. Il dito apporta una variazione alla capacità totale di circa il 3%. L'acquisizione di ciascun valore di conteggio assoluto richiede 500 microsecondi per tasto.

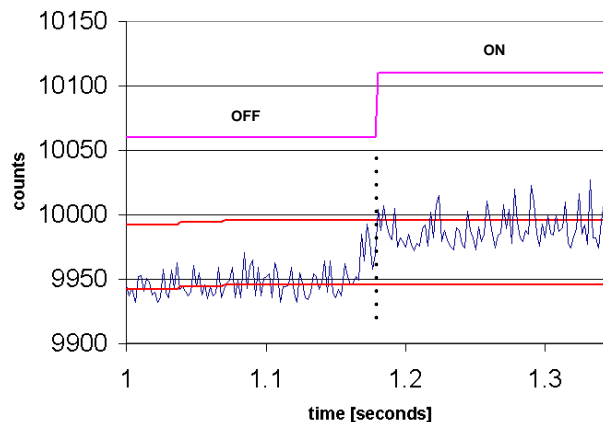
### Prestazioni misurate

Le prestazioni misurate del sistema di rilevamento capacitivo sono riportate in figura 8. I conteggi delle differenze sono stati catturati sul PC host attraverso un programma di emulazione di terminale e quindi disegnati con l'aiuto di un foglio di calcolo. Il dito è presente sulla copertura in vetro da 10mm per 3 secondi. Lo stato ON/OFF dei tasti è sovrapposto ai conteggi assoluti. I tasti passano in modo pulito tra gli stati anche con conteggi assoluti relativamente rumorosi prodotti dal rilevamento attraverso lo strato di vetro spesso. È necessario notare come la soglia del tasto e del dito venga aggiustata periodicamente con il variare della linea di partenza. Quando viene rilevata la pressione di un dito, il valore della linea di partenza si blocca fino a quando il dito viene rimosso.

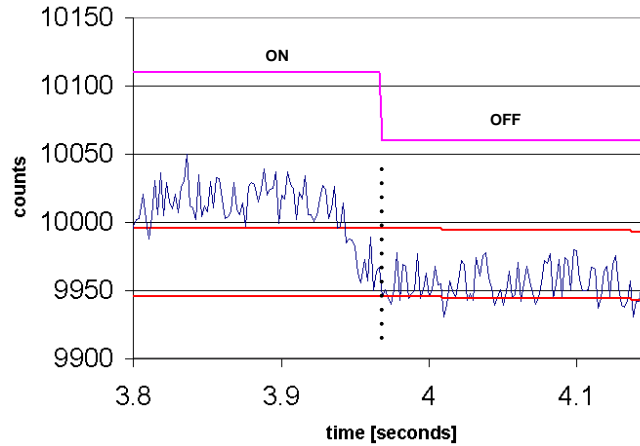


**Figure 8. prestazioni misurate del sensore attraverso una vetro di 10mm**

Le figure 9 e 10 riportano i dettagli di ciascuna transizione di stato. In figura 9 lo stato iniziale del tasto è OFF. Il primo campione del conteggio della differenza è superiore alla soglia del dito porta il tasto nello stato di ON. In figura 10, il tasto esegue una transizione nello stato di OFF con il primo campione del conteggio della differenza inferiore alla soglia di rumore.



**Figura 9. Contesto di transizione nello stato di ON**



**Figura 10. Contesto di transizione nello stato di OFF**

Il vantaggio principale dei sensori tattili a effetto capacitivo rispetto agli switch meccanici è che i primi non si deteriorano nel tempo. I recenti progressi nelle tecnologie mixed signal hanno permesso non solo di ridurre i costi a un punto tale da rendere queste soluzioni convenienti per un ampio spettro di prodotti consumer ma anche di aumentare la sensibilità e l'affidabilità del circuito di rilevazione per consentire di incrementare la robustezza e lo spessore dello strato di copertura. Utilizzando le tecniche di progettazione qui illustrate, è possibile rilevare la pressione di un dito attraverso un vetro di 10mm, usufruendo sempre di transizioni chiare dello stato dei tasti tra ON e OFF utilizzando un metodo antirimbalzo basato su soglie di rumore e di dito-presente. Il tutto rende i sensori tattili capacitivi un'alternativa efficace ai componenti meccanici.



## References

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.