The following document contains information on Cypress products. The document has the series name, product name, and ordering part numbering with the prefix “MB”. However, Cypress will offer these products to new and existing customers with the series name, product name, and ordering part number with the prefix “CY”.

How to Check the Ordering Part Number

1. Go to www.cypress.com/pcn.
2. Enter the keyword (for example, ordering part number) in the **SEARCH PCNS** field and click **Apply**.
3. Click the corresponding title from the search results.
4. Download the Affected Parts List file, which has details of all changes

For More Information

Please contact your local sales office for additional information about Cypress products and solutions.

About Cypress

Cypress is the leader in advanced embedded system solutions for the world's most innovative automotive, industrial, smart home appliances, consumer electronics and medical products. Cypress' microcontrollers, analog ICs, wireless and USB-based connectivity solutions and reliable, high-performance memories help engineers design differentiated products and get them to market first. Cypress is committed to providing customers with the best support and development resources on the planet enabling them to disrupt markets by creating new product categories in record time. To learn more, go to www.cypress.com.

AN204902

F²MC-8FX 家族 MB95310/370 系列 8 位微型控制器红外遥控函数 API

相关器件系列：MB95310/370 系列

本文档介绍了红外遥控函数的 API。

目录

1 概要 1
2 背景 1
3 红外遥控的原理 2
4 MB95F310 红外遥控寄存器 3
5 红外遥控库的函数列表.....4
6 红外遥控函数详情.....4
6.1 Initial_Remot 函数 4
6.2 中断函数.....4
6.3 Remote_detect 函数.....5
7 用法演示 5
7.1 硬件设计 5
7.2 软件设计的步骤.....6
7.3 添加遥控库的步骤 6
8 调试 9
9 更多信息 10
文档修改记录.....11

1 概要

本文档介绍了红外遥控函数的 API。

红外遥控函数一般应用于电视、音频和空调系统。以下章节将介绍红外遥控的原理和库。第三章描述了红外遥控的原理，第四章描述了红外遥控的函数库。函数库中设置有三个函数用于控制红外遥控，初始化遥控中断，遥控中断函数用于接收红外代码，按键判断函数用于判断哪一个键被按下。

2 背景

本章介绍了红外遥控的背景。

红外遥控简单易用，购买方便，价格便宜，因此被广泛应用于 TV、VCD、DVD、以及空调系统。

红外遥控出现至今大约有二十五年。它编码按键并把键值转换为“0”和“1”，然后发出。因为每个按键的代码唯一，用户可以根据代码判断键值，而不需要使用键盘解码，省去了一个复杂电路和输入/输出端口。

3 红外遥控的原理

本章介绍了红外遥控的原理。

根据遥控芯片的不同，红外遥控可分为多种类型。以下部分描述了 uPD6121G。

图 1 显示了遥控代码帧的配置。

图 1. 遥控代码

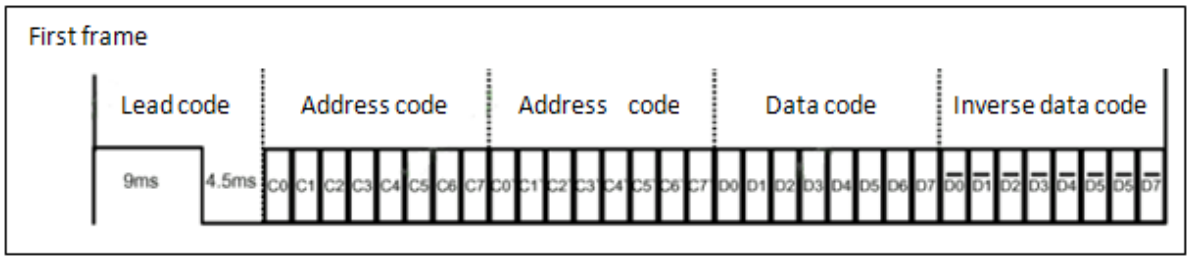
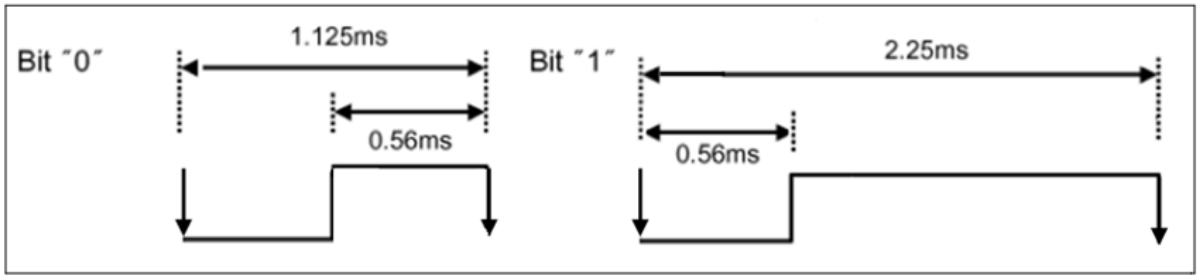


图 2 描述了编码“0”和“1”。

图 2. 编码波形



不同遥控类型的代码和配置是不同的。

一般来说，遥控生成高电平，代码状态例外，它生成低和高电平。

4 MB95F310 红外遥控寄存器

本章描述了 MB95F310 红外遥控寄存器。

MB95F310 系列 MCU 的遥控函数由 8/16 位多功能定时器输入捕捉模块实现。表 1 列出了多功能定时器输入捕捉的寄存器。

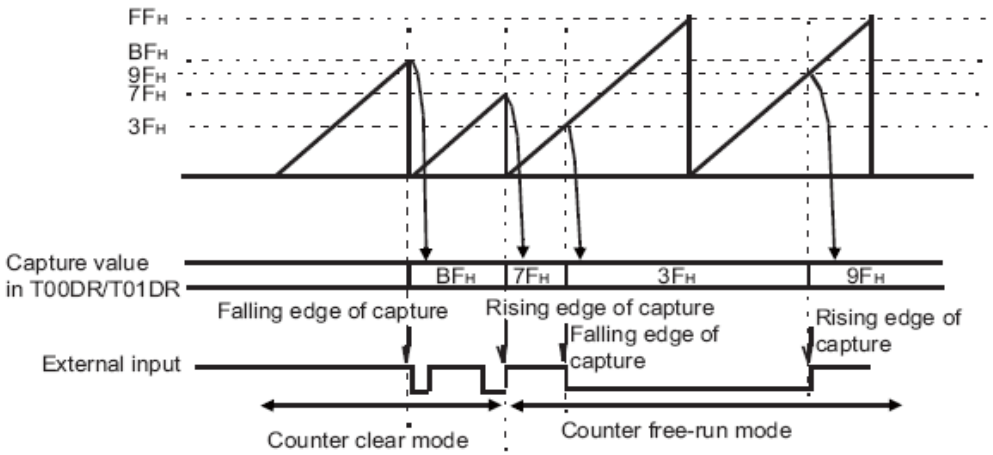
表 1. 寄存器列表

寄存器		说明
Timer0	T00CR0	状态控制寄存器 0 的低 8 位
	T01CR0	状态控制寄存器 0 的高 8 位
Timer0	T00CR1	状态控制寄存器 1 的低 8 位
	T01CR1	状态控制寄存器 1 的高 8 位
Timer0	T00DR	数据寄存器的低 8 位
	T01DR	数据寄存器的高 8 位
Timer0	TMCR0	定时器模式控制寄存器
Timer1	T10CR0	状态控制寄存器 0 的低 8 位
	T11CR0	状态控制寄存器 0 的高 8 位
Timer1	T10CR1	状态控制寄存器 1 的低 8 位
	T11CR1	状态控制寄存器 1 的高 8 位
Timer1	T10DR	数据寄存器的低 8 位
	T11DR	数据寄存器的高 8 位
Timer1	TMCR1	定时器模式控制寄存器

参见 MB95F310 硬件手册的第 18 章了解关于寄存器的更多详细信息。

图 3 详细描述了这些寄存器的工作条件。

图 3. 寄存器工作条件



5 红外遥控库的函数列表

本章介绍了工程 remote.prj 的红外遥控库的所有函数，其 MCU 为 MB95F310。

表 2 列出了红外遥控函数。

表 2. 控捕捉函数

函数名	说明
void initial_Remot(void)	初始化遥控捕捉寄存器
__interrupt void inputcapture0 (void)	接收遥控代码的中断
unsigned char Remote_detect(void)	根据遥控代码判断哪一个键被按下

6 红外遥控函数详情

本章详细介绍了红外遥控函数。

6.1 Initial_Remot 函数

表 3 描述了 initial_Remot 函数。

表 3. initial_Remot 函数

函数名	initial_I2C
函数原型	Void initial_Remot(void)
行为描述	初始化遥控捕捉条件
输入参数	无
返回值	无
示例	库函数设置时钟为内部时钟，计数器时间间隔为 0.64μs，下降沿触发器计数器： initial_Remot ();

参考寄存器 T00CR0、T00CR1 和 TMCR0 更改捕捉条件。

6.2 中断函数

表 4 描述了_interrupt 函数。

表 4. 中断函数描述

函数原型	_ interrupt void inputcapture0 (void)
行为描述	捕捉到一个下降沿时，产生中断，接收脉冲，并计算脉冲宽度
输入参数	无
返回值	无
示例	_interrupt void inputcapture0 ();

不同遥控类型的配置不同，因此代码也不同，可能是 16 位代码，也可能是 32 位代码。用户可以通过示波器查看代码。工程 remote.prj 为 16 位遥控代码。

6.3 Remote_detect 函数

表 5 描述了 Remote_detect 函数。

表 5. Read_I²C 函数

函数名	AD_Read
函数原型	unsigned char Remote_detect(void)
行为描述	判断代码
输入参数	无
返回值	遥控的键值
示例	[Variable] = Remote_detect ();

键值由用户的遥控编码决定。

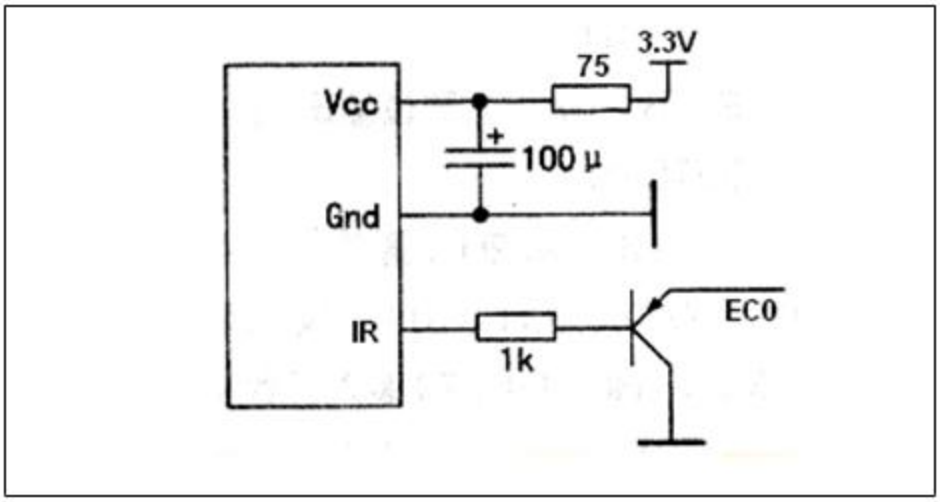
7 用法演示

本章说明了使用时的注意事项。

7.1 硬件设计

下图为红外遥控的硬件设计。

图 4. 遥控硬件设计



7.2 软件设计的步骤

- 红外遥控的软件设计的第一步是初始化，打开中断，设置间隔时间和触发器条件。

  参见以下设置。

图 5. 初始化设计

```
Void initial_Remot(void)
{
    TMCR0 = 0x00;
    T00CR0 = 0x4b;
    T00CR1 = 0xa0;
}
```

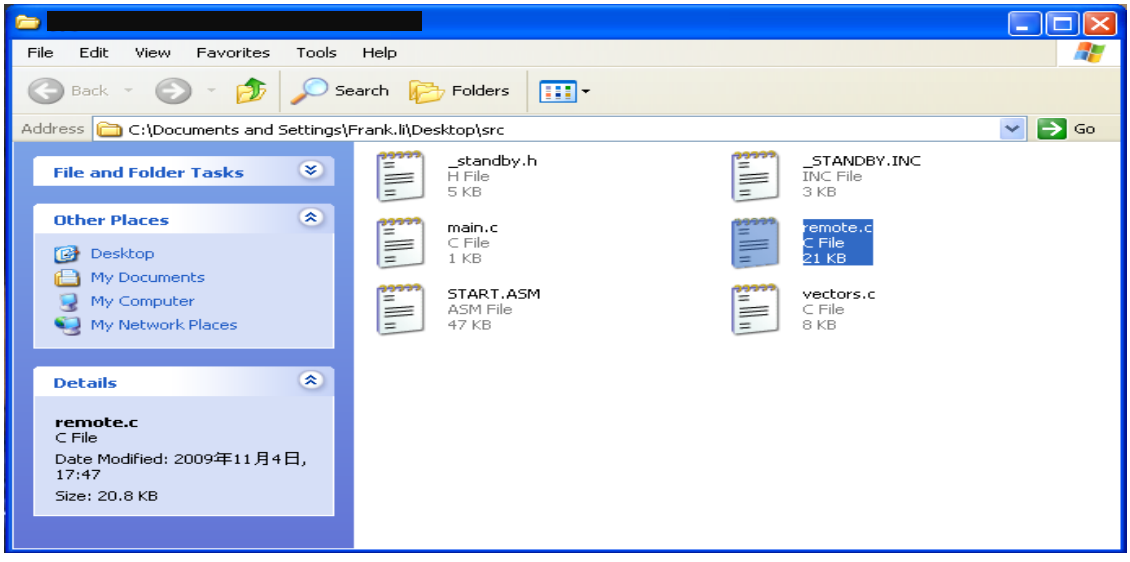
- 第二步是检测遥控代码。每个脉冲都是一位代码。每次中断进入，检测一位代码。如果遥控代码为 16 位，在第十六次进入中断后，生成按键代码。
- 最后一步是键值判断。不同按键的代码不同，因此用户可以根据步骤二中生成的代码判断键值。

7.3 添加遥控库的步骤

参见以下步骤使用该工程。

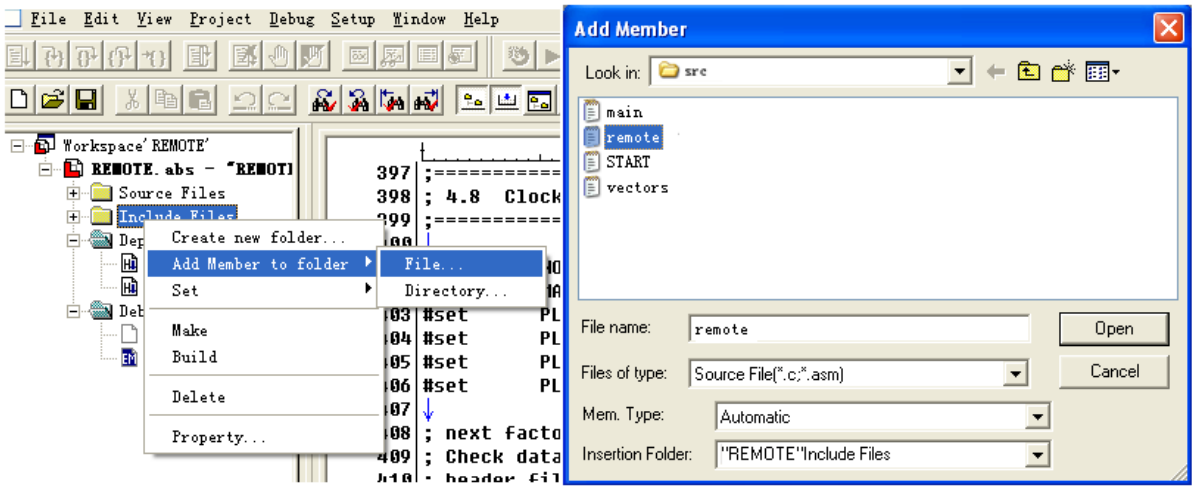
- 第一步：添加库至文件，图 6 描述了此步骤。

图 6. 使用库的第一步



- 第二步：添加函数至工程，图 7 描述了此步骤。

图 7. 用库的第二步



- 第三步：添加初始函数至 main.c，图 8 描述了此步骤。

图 8. 使用库的第三步

```
void main(void)
{
    unsigned char KeyVA;
    InitIrqLevels();
    __EI();

    Initial_Inter();

    while(1)
    {
        KeyVA = Remote_detect();
    }
}
```

- 第四步：添加中断函数至 vector.c，图 9 描述了此步骤。

图 9. 使用库的第四步

```
/*-----------------------------------------------------------------------
   Prototypes
   Add your own prototypes here. Each vector definition needs is proto-
   type. Either do it here or include a header file containing them.
-----------------------------------------------------------------------*/

__interrupt void DefaultIRQHandler (void);
__interrupt void inputcapture0 (void);
/*-----------------------------------------------------------------------
```

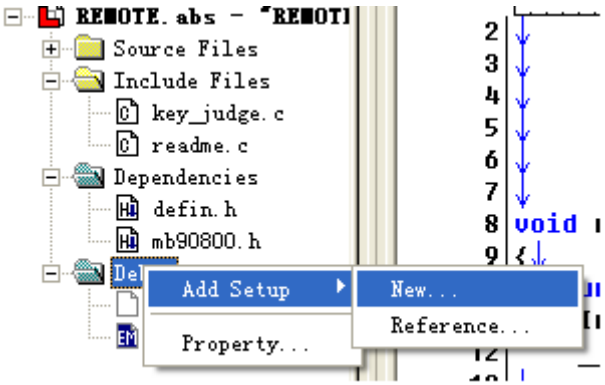
- 第五步：添加 remote_detect 函数至 main.c，图 10 描述了此步骤。

图 10. 使用库的第五步

```
 8 void main(void)
 9 {
10     unsigned char KeyVA;
11     InitIrqLevels();
12     __EI();
13
14     Initial_Inter();
15
16     while(1)
17     {
18         KeyVA = Remote_detect();
19     }
20 }
```

- 第六步：调试。在调试前设置调试环境，图 11 描述了此步骤。

图 11. 使用库的第六步



参见第 8 章调试了解关于条件的更多信息。

8 调试

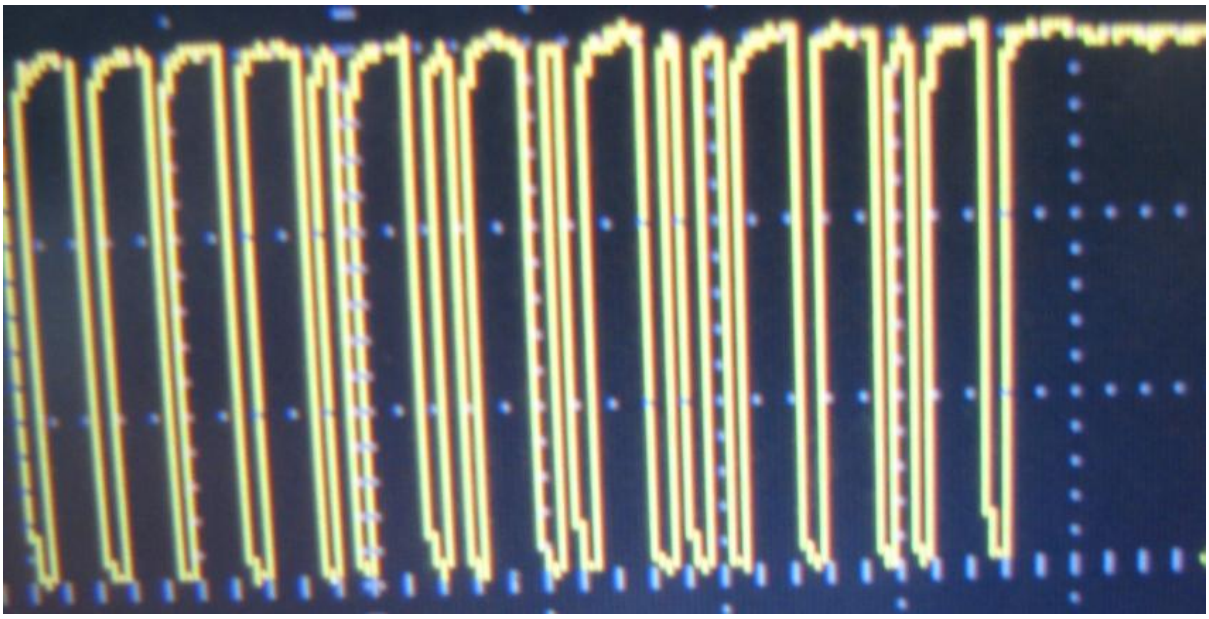
本章描述了如何在 EV 板上调试范例代码，以及代码运行时会发生什么。

工程 remote.prj 是一个简单用于调试的工程。该工程基于 EV 板 MB2146-450-E，目标 MCU 为 MB95F310。

图 4 显示了调试时的硬件连接。

按下键后，示波器检测到波形，如图 12 所示。

图 12. 按键代码

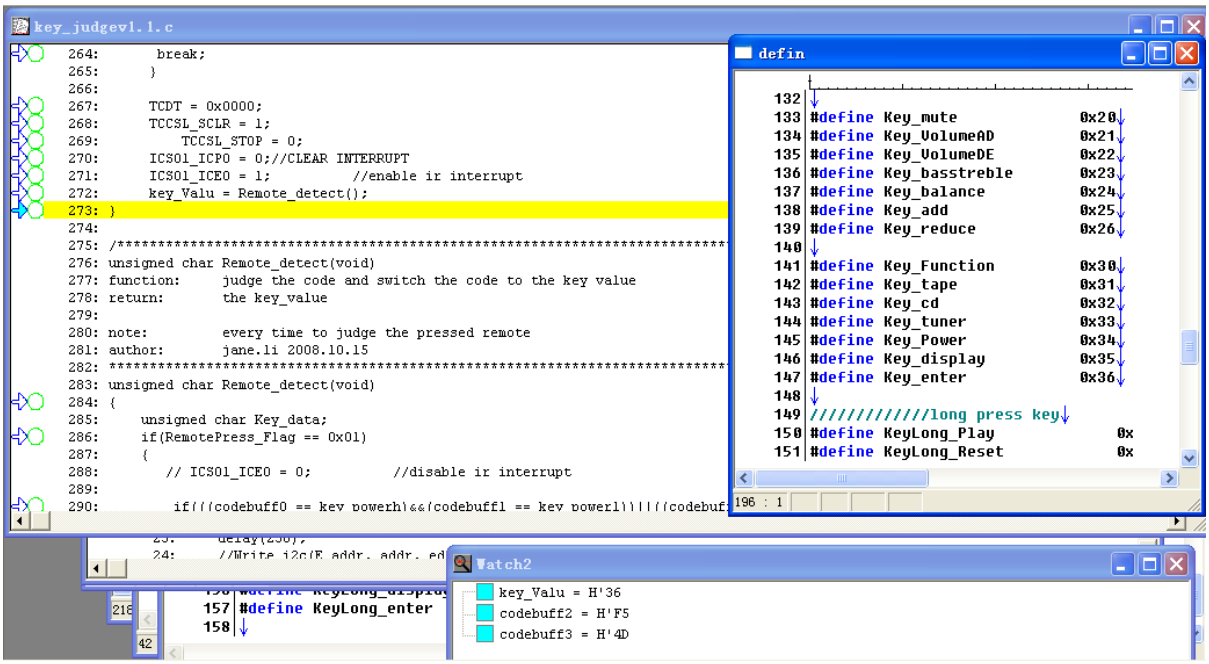


宽波为二进制位“1”，短波为二进制位“0”。

图片中的遥控代码在二进制中为“1111 0101 0100 1101”。

运行工程，Remote_detect 函数将读取键值。键值将由全局变量“key_Valu”读出。在此范例中，Enter 键被按下。参见图 13 了解更多详情。

图 13. 调试描述



9 更多信息

如欲了解有关如何使用 MB95310 EV-board、BGM Adaptor 和 SOFTUNE 的更多详情，敬请参见 EV-Board MB2146-450-E 用户手册，或者访问以下网址：

http://www.cypress.com/documentation/application-notes/mb95310370-mb2146-450-e-lcd-evb-user-manual

文档修改记录

文档标题：AN204902 - F²MC-8FX 家族 MB95310/370 系列 8 位微型控制器红外遥控函数 API

文档编号：002-05699

修订版	ECN	变更者	提交日期	变更说明
**	—	HUAL	11/06/2009	初稿.
			03/17/2011	修改范例代码
*A	5333065	HUAL	07/01/2016	已将 Spansion 应用手册《MCU-AN-500066-Z-11》转换成 Cypress 格式。

全球销售和设计支持

赛普拉斯公司拥有一个由办事处、解决方案中心、厂商代表和经销商组成的全球性网络。如果想要查找离您最近的办事处，请访问赛普拉斯所在地。

产品

ARM® Cortex® 微控制器	cypress.com/arm
汽车级	cypress.com/automotive
时钟与缓冲器	cypress.com/clocks
接口	cypress.com/interface
照明和电源控制	cypress.com/powerpsoc
存储器	cypress.com/memory
PSoC	cypress.com/psoc
触摸感应	cypress.com/touch
USB 控制器	cypress.com/usb
无线/射频	cypress.com/wireless

PSoC® 解决方案

PSoC 1 | PSoC 3 | PSoC 4 | PSoC 5LP

赛普拉斯开发者社区

论坛 | 项目 | 视频 | 博客 | 培训 | 组件

技术支持

cypress.com/support

PSoC 是赛普拉斯半导体公司的注册商标。PSoC Creator 是赛普拉斯半导体公司的商标。 此处引用的所有其他商标或注册商标都归其各自所有者所有。

赛普拉斯半导体
198 Champion Court
San Jose, CA 95134-1709

电话 : 408-943-2600
传真 : 408-943-4730
网站地址 : www.cypress.com

©赛普拉斯半导体公司，2009-2016 年。本文件是赛普拉斯半导体公司及其子公司，包括 Spansion LLC（“赛普拉斯”）的财产。本文件，包括其包含或引用的任何软件或固件（“软件”），根据全球范围内的知识产权法律以及美国与其他国家签署条约由赛普拉斯所有。除非在本款中另有明确规定，赛普拉斯保留在该等法律和条约下的所有权利，且未就其专利、版权、商标或其他知识产权授予任何许可。如果软件并不附随有一份许可协议且贵方未以其他方式与赛普拉斯签署关于使用软件的书面协议，赛普拉斯特此授予贵方属个人性质的、非独家且不可转让的如下许可（无再许可权）（1）在赛普拉斯特软件著作权项下的下列许可权（一）对以源代码形式提供的软件，仅出于在赛普拉斯硬件产品上使用之目的且仅在贵方集团内部修改和复制软件，和（二）仅限于在有关赛普拉斯硬件产品上使用之目的将软件以二进制代码的形式向外部最终用户提供（无论直接提供或通过经销商和分销商间接提供），和（2）在被软件（由赛普拉斯公司提供，且未经修改）侵犯的赛普拉斯专利的权利主张项下，仅出于在赛普拉斯硬件产品上使用之目的制造、使用、提供和进口软件的许可。禁止对软件的任何其他使用、复制、修改、翻译或汇编。
在适用法律允许的限度内，赛普拉斯未对本文件或任何软件作出任何明示或暗示的担保，包括但不限于关于适销性和特定用途的默示保证。在适用法律允许的限度内，赛普拉斯保留更改本文件的权利，届时将不另行通知。赛普拉斯不对因应用或使用本文件所述任何产品或电路引起的任何后果负责。本文件，包括任何样本设计信息或程序代码信息，仅为参考之目的提供。文件使用人应负责正确设计、计划和测试信息应用和由此生产的任何产品的功能和安全性。赛普拉斯产品不应被设计为、设定为或授权用作武器操作、武器系统、核设施、生命支持设备或系统、其他医疗设备或系统（包括急救设备和手术植入物）、污染控制或有害物质管理系统中的关键部件，或产品植入之设备或系统故障可能导致人身伤害、死亡或财产损失的其他用途（“非预期用途”）。关键部件指，若该部件发生故障，经合理预期会导致设备或系统故障或会影响设备或系统安全性和有效性的部件。针对由赛普拉斯产品非预期用途产生或相关的任何索赔、费用、损失和其他责任，赛普拉斯不承担全部或部分责任且贵方不应追究赛普拉斯之责任。贵方应赔偿赛普拉斯因赛普拉斯产品任何非预期用途产生或相关的所有索赔、费用、损失和其他责任，包括因人身伤害或死亡引起的索赔，并使之免受损失。
赛普拉斯、赛普拉斯徽标、Spansion、Spansion 徽标，及上述项目的组合，及 PSoC、CapSense、EZ-USB、F-RAM 和 Traveo 应视为赛普拉斯在美国和其他国家的商标或注册商标。敬请访问 cypress.com 获取赛普拉斯商标的完整列表。其他名称和品牌可能由其各自所有者主张为该方财产。