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.

AN205508

F²MC-8FX 家族 MB95200 系列使用 AD 的键盘开发

本文档描述了如何使用 AD 转换设计键盘，同时也以实例介绍了 AD 转换的实现理论。

目录

1 概要 1
2 AD 功能介绍 1
2.1 AD 转换功能 1
2.2 键盘抖动消除 1
3 硬件设计 2
3.1 硬件电路 2
3.2 键状态和输入电压的关系 2
4 资源使用 3
4.1 8/10 位 A/D 转换器控制寄存器 1（ADC1） 3
4.2 8/10 位 A/D 转换器控制寄存器 2（ADC2） 4
4.3 8/10 位 A/D 转换器数据寄存器上 / 下（ADDH、ADDL） 5
4.4 AD 转换过程 5
5 软件设计 6
6 代码示例 7
7 附加信息 9
文档修改记录 10

1 概要

使用 MB95200 系列 MCU 设计键盘的方法一共有三种：外部中断、AD 转换和 matrix。

本文档描述了如何使用 AD 转换设计键盘。

本文档同时也以实例介绍了 AD 转换的实现理论。

2 AD 功能介绍

本章介绍了使用 AD 转换设计键盘的功能。

2.1 AD 转换功能

AD 转换器的功能是把模拟输入引脚的模拟电压值（输入电压）转换为一个 10 位的数值。MB95200 系列 MCU 至少有 2 个模拟输入引脚。启用 AD 转换器后，所选通道上的模拟输入信号将在 AD 转换器中被取样和比较，最终转换成一个数值，存储在 A/D 转换器数据寄存器（ADDH、ADDL）。

2.2 键盘抖动消除

抖动消除是键盘设计中最重要的问题之一。对于硬件电路设计，在模拟输入引脚中增加一个电容可以减弱抖动。对于软件设计，反复确认数据可以有效避免键盘抖动。

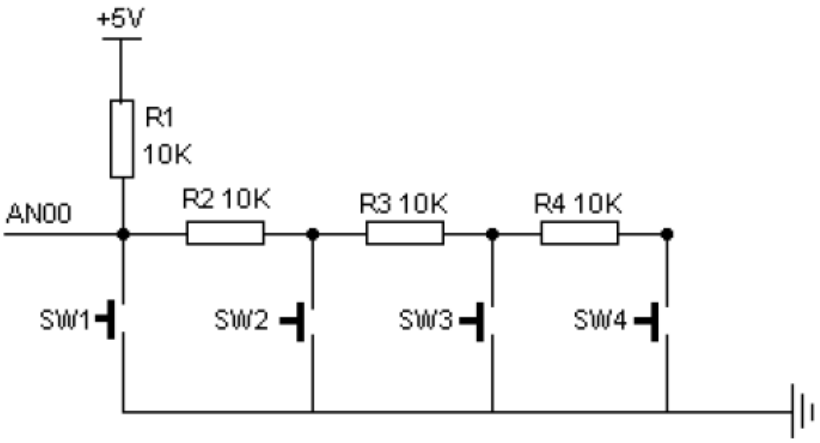
3 硬件设计

本章介绍了使用 AD 转换进行键盘开发的硬件设计。

3.1 硬件电路

以下硬件电路的电阻用于分压。模拟输入引脚将从电路获得一个不同的电压。下图中的电路供用户参考：

图 1. 使用 AD 进行键盘硬件设计



3.2 键状态和输入电压的关系

表 1. 键状态和输入电压的关系

按键	AN00 电压	ADDL 值	有效值区域
SW1	0 V	0x00	0x00>=ADDL<=0x05
SW2	2.5 V	0x7F	0x79>=ADDL<=0x85
SW3	3.3 V	0xAA	0xA1>=ADDL<=0xB2
SW4	3.75 V	0xBF	0xB5>=ADDL<=0xC8

4 资源使用

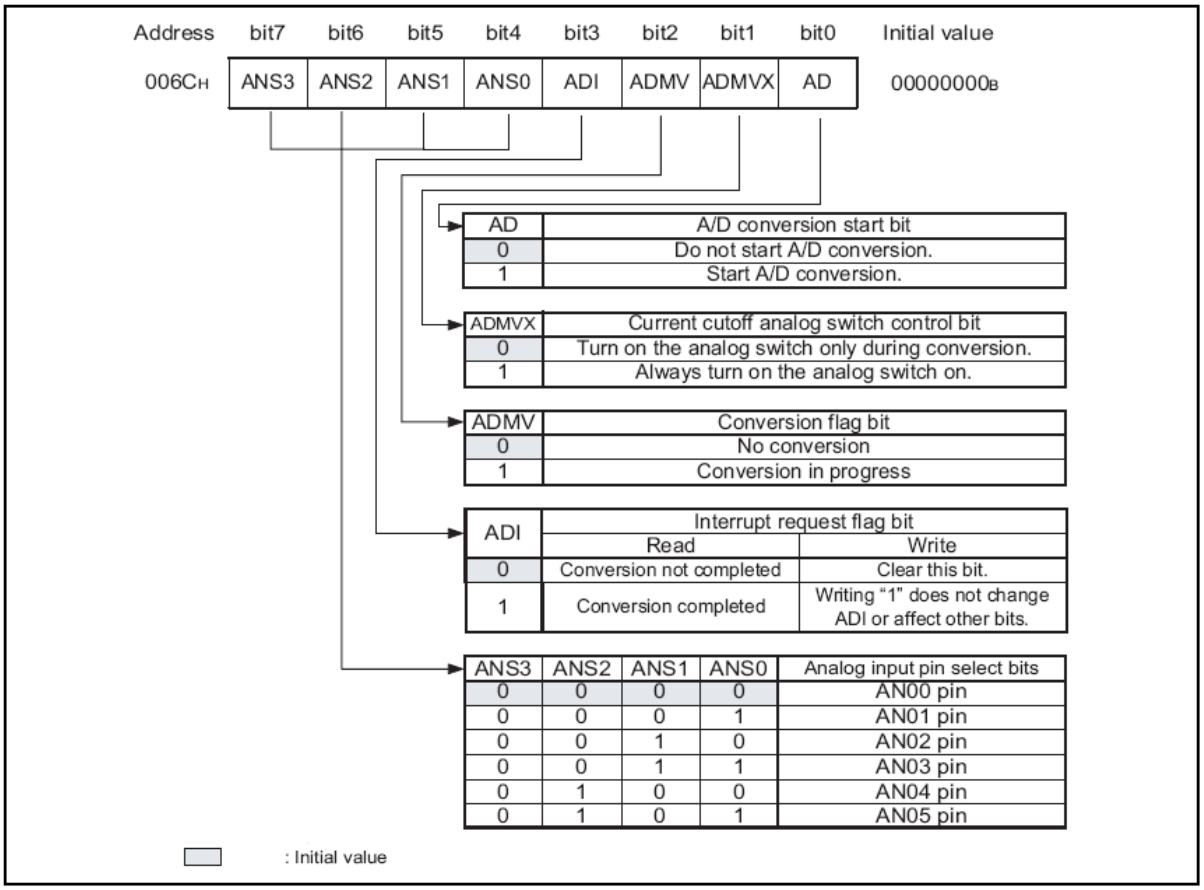
本章介绍了 AD 转换的资源使用。

参见 MB95200 系列硬件手册的第 17 章了解更多关于寄存器设置的详细信息。

4.1 8/10 位 A/D 转换器控制寄存器 1（ADC1）

8/10 位 A/D 转换器控制寄存器 1（ADC1）用于启用和禁用 8/10 位 A/D 转换器的个别功能：选择模拟输入引脚，以及检查转换器的状态。

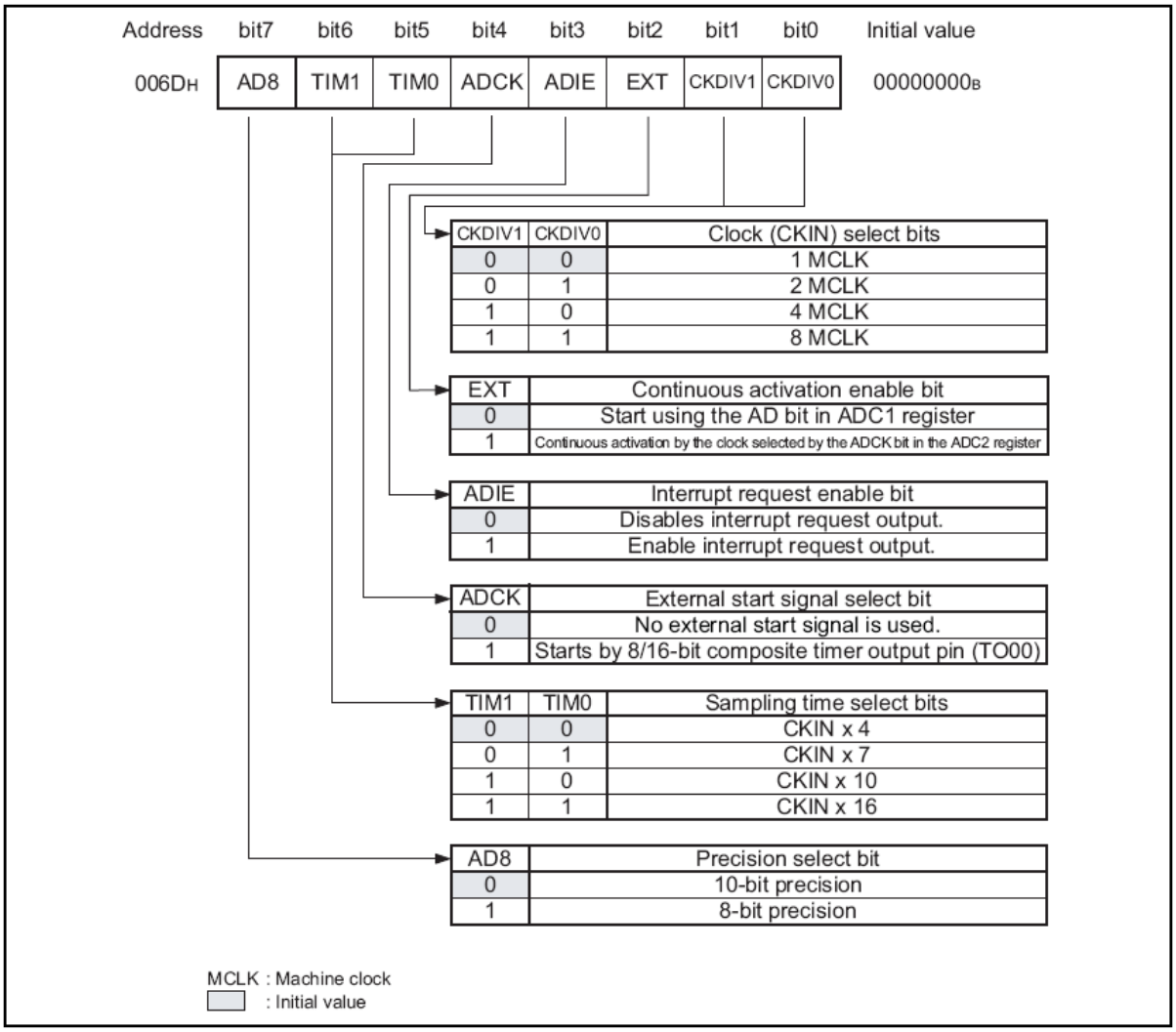
图 2. ADC1 寄存器



4.2 8/10 位 A/D 转换器控制寄存器 2（ADC2）

8/10 位 A/D 转换器控制寄存器 2（ADC2）用于控制 8/10 位 A/D 转换器的不同功能：选择输入时钟，启用和禁用中断。

图 3. ADC2 寄存器



4.3 8/10 位 A/D 转换器数据寄存器上/下（ADDH、ADDL）

8/10 位 A/D 转换器数据寄存器上/下（ADDH、ADDL）用于在 10 位 A/D 转换期间存储 10 位 A/D 转换的结果。10 位数据的上面二位存储于 ADDH 寄存器，下面的八位存储于 ADDL 寄存器。

图 4. AD 数据寄存器

ADDH	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	Initial value
Address 006E$_H$	-	-	-	-	-	-	SAR9	SAR8	00000000$_B$
ADDL	bit15	bit14	bit13	bit12	bit11	bit10	bit9	bit8	Initial value
Address 006F$_H$	SAR7	SAR6	SAR5	SAR4	SAR3	SAR2	SAR1	SAR0	00000000$_B$

10 位 A/D 数据的上面二位对应 ADDH 寄存器中的 bit1 和 bit0，下面八位对应 ADDL 寄存器的 bit15-bit8。

如果 ADC2 寄存器的 AD8 位设置为“1”，则选择 8 位精确度。读 ADDL 寄存器可以得到一个 8 位数据。

4.4 AD 转换过程

根据寄存器设置进行转换后，转换结果存储于 ADDH 和 ADDL 寄存器。在 A/D 转换完成后，并且下一次 A/D 转换开始前，需要读取 A/D 数据寄存器（转换结果）并在 ADC1 寄存器中清除 ADI 标志位（bit3）。

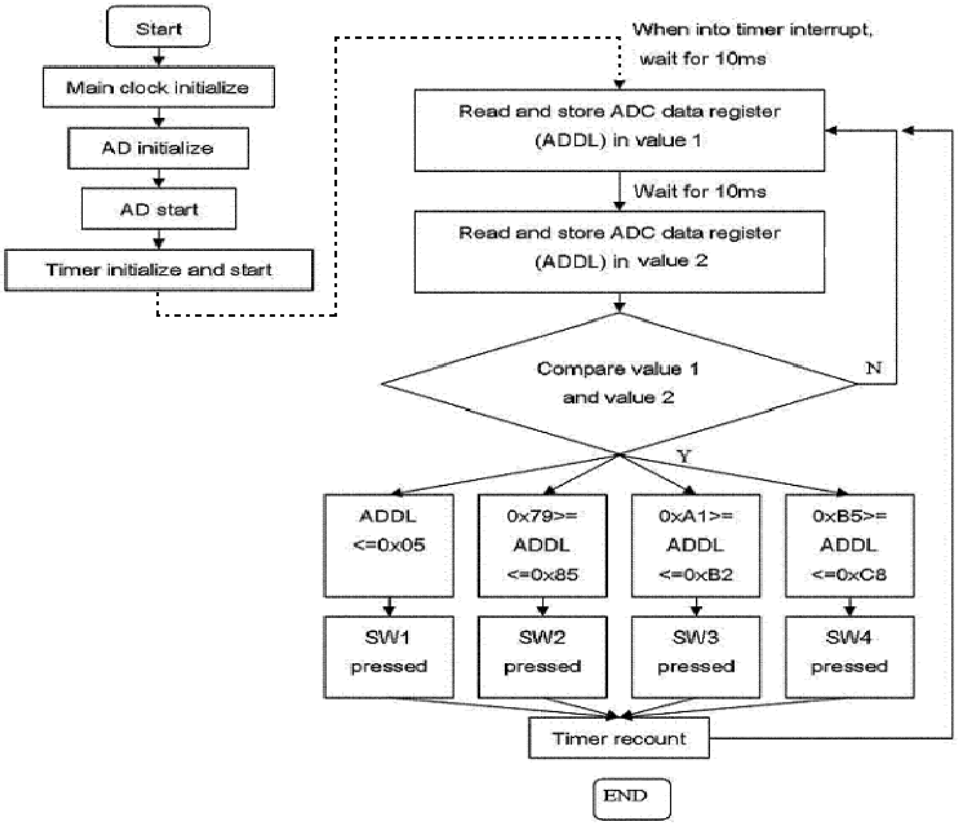
5 软件设计

本章介绍了使用 AD 转换进行键盘开发的软件设计。

一旦启用 AD 转换器，它将获取键盘的状态信息。如果该值（Distem0）属于键值，恢复它；如果不是，AD 转换器继续采样。然后，停止 AD 转换器，让系统等待约 20ms 后，重新启动 AD 转换器并再次获取采样值（Distem1）。最后，判断该值是否等于 Distem0。如果与 Distem0 不同，视为抖动发生；否则，表示按键发生。

下图为软件设计的流程图。

图 5. 键盘开发的软件设计流程图



6 代码示例

本章举例说明了如何使用 AD 转换进行键盘开发。

下面的例子显示了如何使用 AD 转换输入引脚 AN00 设计一个 4 键的键盘。

如第 5 章所述，如果使用 AD 转换设计键盘系统，首先初始化 AD 转换，然后 AD 转换开始采样，获得采样值，读取该值并判断其是否属于键值。如果不是，系统继续采样；如果是，停止 AD 转换，让系统等待大约 20ms 后，重新启动 AD 转换，获取另一个采样值。如果这两个采样值相同，表示一个键被按下，否则视为抖动发生。

```
#include "../MB95200_IO/mb95200.h". . .
void clock_Select(void)       // Main clock initialize
{
SYCC = 0x00;
WATR = 0x00;
STBC = 0x01;
SYCC2= 0xF4;
}
void syinit(void)
{
DDR0_P00 = 0;  //P00 input enable
ADC1 = 0x01;           //AN00 as input
ADC2 = 0x80;           //8-bit resolution,1×MCLK
ADC1_AD = 1;           //start A/D converter
DDR0 = 0xF0;           //P4~P7 as output pin
PUL0 = 0x00;
AIDRL = 0xFE;
}
void timer_init(void)
{
TMCR0 = 0x10;  // 16-bit
T00CR0 = 0x81;         // interval timer with continuous mode
T00CR1 = 0xA0;         // disable output, start timer
T01DR = 0x27;  // 10000us
T00DR = 0x10;
while(STBC_MRDY);
}
__interrupt void Timer_Interrupt (void)
{
T00CR1_IF = 0;
T00CR1_IE = 0;
timer_counter++;
if(timer_counter==1) // Read and store ADDL in value 1
{
AD_Sample();
}
```

```
if(timer_counter==2)     // Read and store ADDL in value 2 and
compare value and value 2
{
key_judge();
}
if(timer_counter>=3)     /Timer_counter restart
{
timer_counter=0;
}
}
void AD_Sample(void)
{
if(ADC1_ADI==1)
{
while(!ADC1_ADI);
Distem0 = ADD_ADDL; // Read AD Sample value 1
}
ADC1_AD=1;               // Start AD again
}
void key_judge(void)
{
if(ADC1_ADI==1)   // Read and store ADDL in value 2
{
while(!ADC1_ADI);
Distem1 = ADD_ADDL;
}
if(Distem1==Distem0)     // Judge if value 1 and value 2 equality
{
if(Distem1 == Range_value0) // Key process
{
DDR0_P04 = 1;
PDR0_P04 = 0;
}
if((Distem1 >= Range_value1)&&(Distem1 <= Range_value2))
{
DDR0_P05 = 1;
PDR0_P05 = 0;
}
if((Distem1 >= Range_value3)&&(Distem1 <= Range_value4))
{
DDR0_P06 = 1;
PDR0_P06 = 0;
}
if((Distem1 >= Range_value5)&&(Distem1 <= Range_value6))
{
DDR0_P07 = 1;
PDR0_P07 = 0;
}
}
}
void main(void)
{
InitIrqLevels(); //initialise Interrupt level and IRQ vector table
__EI();          // global interrupt enable
clock_Select();
while(1)
{
syinit();
timer_init();
}
}
```

注意：该工程基于 starter-kit MB2146-410A。目标 MCU 为 MB95F304K。AN0 用作模拟输入引脚，LED1-LED4 用作指示器。如果没有提供 3.1 节中的键盘硬件电路，停止跳线并把 VR3 连接到 MCU，用作模拟输入引脚。然后通过调整 VR3 观察 LED 的变化。

7 附加信息

如欲了解有关 Cypress MB95200 产品的更多信息，敬请访问以下网址：

http://www.cypress.com/8fx-mb95200

文档修改记录

文档标题：AN205508 - F²MC-8FX 家族 MB95200 系列使用 AD 的键盘开发

文档编号：002-05775

修订版	ECN	变更者	提交日期	变更说明
**	—	HUAL	02/24/2009	初稿
*A	5348878	HUAL	07/13/2016	已将 Spansion 应用手册《MCU-AN-500037-Z-10》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。