第 48 章 存储器构成和权限

目录

本章包括下列主题：

48.1 简介 48-2
48.2 控制寄存器 48-3
48.3 存储器布局 48-13
48.4 系统总线 48-16
48.5 系统总线仲裁 48-18
48.6 访问权限 48-19
48.7 复位的影响 48-21
48.8 节能模式下的操作 48-21
48.9 调试模式 48-21
48.10 代码示例 48-22
48.11 相关应用笔记 48-25
48.12 版本历史 48-26

注： 本系列参考手册章节旨在用作对器件数据手册的补充。根据不同的器件型号，本手册章节可能并不适用于所有 PIC32 器件。

请参见当前器件数据手册中 **“存储器构成”** 章节开头部分的注，以检查本文档是否支持您所使用的器件。

器件数据手册和系列参考手册章节可从 Microchip 网站下载：http://www.microchip.com。

48.1 简介

PIC32MZ 系列单片机提供 4 GB 的统一虚拟存储器地址空间。所有存储器区域（包括程序存储器、数据存储器、外部存储器、SFR 和配置寄存器）都位于该地址空间中各自的惟一地址范围内。PIC32MZ 器件可以从引导闪存、程序闪存、数据存储器或外部存储器中执行。高度可配置的保护方案可以阻止用户软件或 DMA 对选定存储器区域进行访问，而安全内核或引导代码保持可供完全访问。

PIC32MZ 系统总线以多层结构的形式（交叉式）实现，通过将大量发起器与大量目标连接来支持并发事务。总线发起器包括 CPU、通用 DMA 和带有专用 DMA 的外设，而总线目标则包括程序存储器、数据存储器和外设。

PIC32MZ 存储器构成的主要特性包括：

- 32 位固有数据宽
- 集成存储器管理单元（Memory Management Unit，MMU）与固定映射，实现可安全配置的存储器和外设访问控制权限
- 总线仲裁方案使用最近最少处理（LRS）优先级实现，为 CPU、通用 DMA 和带有专用 DMA 的外设提供服务质量 （Quality of Service， QOS）
- 双闪存分区支持即时更新程序存储器
- 独立的双重引导闪存支持更新引导代码
- 双 RAM 存储区可以用于避免使用 DMA 时发生的总线仲裁
- 外部串行和 / 或并行存储器可以使用串行四通道接口（Serial Quad Interface，SQI）或外部总线接口 （External Bus Interface， EBI）映射到虚拟存储空间，用于进行数据访问或代码执行
- 可高速缓存的地址区和不可高速缓存的地址区

© 2014 Microchip Technology Inc.

48.2 控制寄存器

PIC32 MMU 具有以下 SFR：

- **SBFLAG：系统总线状态标志寄存器**
  该位编码寄存器指示（如果有）哪些总线目标正在报告权限组违例错误。
- **SBTxELOG1：系统总线目标 x 错误日志寄存器 1（x = 0-13）**
  该寄存器提供关于权限组违例错误的详细信息（如存在）。
- **SBTxELOG2：系统总线目标 x 错误日志寄存器 2（x = 0-13）**
  该寄存器提供关于权限组违例错误的详细信息（如存在）。
- **SBTxECON：系统总线目标 x 错误控制寄存器（x = 0-13）**
  该寄存器用于控制权限组违例错误报告。
- **SBTxECLRS：系统总线目标 x 单个错误清除寄存器（x = 0-13）**
  该寄存器用作清除单个权限组违例错误报告的机制。
- **SBTxECLRM：系统总线目标 x 多个错误清除寄存器（x = 0-13）**
  该寄存器用作清除多个权限组违例错误报告的机制。
- **SBTxREGy：系统总线目标 x 区域 y 寄存器（x = 0-13； y = 0-8）**
  该寄存器用于配置目标的存储器区域的基址、优先级和大小。目标具有多个存储器区域。
- **SBTxRDy：系统总线目标 x 区域 y 读权限寄存器（x = 0-13； y = 0-8）**
  该寄存器用于配置每个组对于每个目标的存储器区域的读权限。
- **SBTxWRy：系统总线目标 x 区域 y 写权限寄存器（x = 0-13； y = 0-8）**
  该寄存器用于配置每个组对于每个目标的存储器区域的写权限。

表 48-1 简要汇总了相关的存储器构成和权限寄存器。该汇总表之后列出了相应的寄存器，并且每个位均附有详细的说明。

表 48-1： 系统总线 SFR 汇总

名称		Bit 31/15	Bit 30/14	Bit 29/13	Bit 28/12	Bit 27/11	Bit 26/10	Bit 25/9	Bit 24/8	Bit 23/7	Bit 22/6	Bit 21/5	Bit 20/4	Bit 19/3	Bit 18/2	Bit 17/1	Bit 16/0
SBFLAG(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	T13PGV	T12PGV	T11PGV	T10PGV	T9PGV	T8PGV	T7PGV	T6PGV	T5PGV	T4PGV	T3PGV	T2PGV	T1PGV	T0PGV
SBTxELOG1(1)	31:16	MULTI	—	—	—	CODE<3:0>				—	—	—	—	—	—	—	—
	15:0	INITID<7:0>								REGION<3:0>				—	CMD<2:0>		
SBTxELOG2(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GROUP<1:0>	
SBTxECON(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERRP
SBTxECLRS(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
SBTxECLRM(1)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLEAR
SBTxREGy(1,2)	31:16	BASE<21:6>															
	15:0	BASE<5:0>						PRI	—	SIZE<4:0>					—	—	—
SBTxRDy(1,2)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0
SBTxWRy(1,2)	31:16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
	15:0	—	—	—	—	—	—	—	—	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

图注： — = 未实现，读为 0。

注 1： 关于可用目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”** 章节。

2： 对于一些目标区域，该寄存器中的某些位为只读，且具有预设值。更多信息，请参见具体器件数据手册中的 **“存储器构成”** 章节。

© 2014 Microchip Technology Inc.

寄存器 48-1： SBFLAG：系统总线状态标志寄存器

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	R-0	R-0	R-0	R-0	R-0	R-0
	—	—	T13PGV	T12PGV	T11PGV	T10PGV	T9PGV	T8PGV
7:0	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	T7PGV	T6PGV	T5PGV	T4PGV	T3PGV	T2PGV	T1PGV	T0PGV

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31-14 **未实现：**读为 0

bit 13-0 **TxPGV：**目标 x 权限组违例状态位 （x = 0-13）

关于可用目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。

1 = 目标正在报告权限组违例
0 = 目标未报告权限组违例

注： 清除源寄存器 （SBTxELOG1、 SBTxELOG2、 SBTxECLRS 或 SBTxECLRM 寄存器）中的所有错误。

寄存器 48-2： SBTxELOG1：系统总线目标 x 错误日志寄存器 1（x = 0-13）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W-0	U-0	U-0	U-0	R/W-0	R/W-0	R/W-0	R/W-0
	MULTI	—	—	—	CODE<3:0>			
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
	INITID<7:0>(1)							
7:0	R-0	R-0	R-0	R-0	U-0	R-x	R-x	R-x
	REGION<3:0>				—	CMD<2:0>		

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31 **MULTI：**多个权限违例状态位
1 = 已检测到多个错误
0 = 未检测到多个错误

bit 30-28 **未实现：**读为 0

bit 27-24 **CODE<3:0>：**错误代码位
指示所报告错误的类型。
1111 = 保留
•
•
•
0100 = 保留
0011 = 权限违例
0010 = 保留
0001 = 保留
0000 = 无错误

bit 23-16 **未实现：**读为 0

注 1： 选择会因器件而变化。要确定哪些选择可用，请参见具体器件数据手册中的 **“存储器构成”**章节。

注： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。

© 2014 Microchip Technology Inc.

寄存器 48-2： SBTxELOG1：系统总线目标 x 错误日志寄存器 1 （x = 0-13）（续）

bit 15-8 **INITID<7:0>：**请求方的发起器 ID 位

11111111 = 保留
•
•
•
00001111 = 保留
00001110 = 加密引擎
00001101 = 闪存控制器
00001100 = SQI1
00001011 = CAN2
00001010 = CAN1
00001001 = 以太网写
00001000 = 以太网读
00000111 = USB
00000110 = DMA 写 （DMAPRI （CFGCON<25>） = 1）
00000101 = DMA 写 （DMAPRI （CFGCON<25>） = 0）
00000100 = DMA 读 （DMAPRI （CFGCON<25>） = 1）
00000011 = DMA 读 （DMAPRI （CFGCON<25>） = 0）
00000010 = CPU （CPUPRI （CFGCON<24>） = 1）
00000001 = CPU （CPUPRI （CFGCON<25>） = 0）
00000000 = 保留

bit 7-4 **REGION<3:0>：**请求的区域编号位

1111 - 0000 = 目标中报告权限组违例的区域 （该编号对应于 SBTxREGy 区域定义寄存器名称中的 “y”）

bit 3 **未实现：**读为 0

bit 2-0 **CMD<2:0>：**请求方的事务命令位

111 = 保留
110 = 保留
101 = 写 （non-posted 写）
100 = 保留
011 = 读 （读 - 修改 - 写事务引起的锁定读）
010 = 读
001 = 写
000 = 空闲

注 1： 选择会因器件而变化。要确定哪些选择可用，请参见具体器件数据手册中的 **“存储器构成”**章节。

注： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。

寄存器 48-3： SBTxELOG2：系统总线目标 x 错误日志寄存器 2 （x = 0-13）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	R-0	R-0
	—	—	—	—	—	—	GROUP<1:0>	

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31-2 **未实现：**读为 0

bit 1-0 **GROUP<1:0>：**请求的权限组位

11 = 组 3
10 = 组 2
01 = 组 1
00 = 组 0

注： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。

寄存器 48-4： SBTxECON：系统总线目标 x 错误控制寄存器 （x = 0-13）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R/W-0
	—	—	—	—	—	—	—	ERRP
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31-25 **未实现：**读为 0

bit 24 **ERRP：**错误控制位

1 = 报告保护组违例错误
0 = 不报告保护组违例错误

bit 23-0 **未实现：**读为 0

注： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。

© 2014 Microchip Technology Inc.

寄存器 48-5： SBTxECLRS：系统总线目标 x 单个错误清除寄存器 （x = 0-13）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0
	—	—	—	—	—	—	—	CLEAR

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31-1 **未实现：**读为 0

bit 0 **CLEAR：**读操作时清除单个错误位

SBTxELOG1 和 SBTxELOG2 所报告的单个错误在读该寄存器时清除。如果错误日志已被清除，则会在 CLEAR 字段中返回值 1。

注： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。

寄存器 48-6： SBTxECLRM：系统总线目标 x 多个错误清除寄存器 （x = 0-13）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	U-0	U-0	U-0	R-0
	—	—	—	—	—	—	—	CLEAR

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31-1 **未实现：**读为 0

bit 0 **CLEAR：**读操作时清除多个错误位

SBTxELOG1 和 SBTxELOG2 所报告的多个错误在读该寄存器时清除。如果错误日志已被清除，则会在 CLEAR 字段中返回值 1。

注： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。

寄存器 48-7： SBTxREGy：系统总线目标 x 区域 y 寄存器（x = 0-13； y = 0-8）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	R/W0	R/W-0	R/W0	R/W-0	R/W0	R/W-0	R/W0	R/W-0
	BASE<21:14>							
23:16	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0
	BASE<13:6>							
15:8	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-x	U-0
	BASE<5:0>						PRI	—
7:0	R/W-0	R/W-0	R/W-0	R/W-0	R/W-0	U-0	U-0	U-0
	SIZE<4:0>					—	—	—

图注：			
R = 可读位	W = 可写位	U = 未实现位，读为 0	
-n = POR 时的值	1 = 置 1	0 = 清零	x = 未知

bit 31-10 **BASE<21:0>：**区域基址位
在物理存储空间中定义的该权限区域的基址位的 bit 31:10。这些位定义的地址必须与 SIZE<4:0> 位指定的大小对齐。

bit 9 **PRI：**区域优先级位
1 = 级别 2
0 = 级别 1

bit 8 **未实现：**读为 0

bit 7-3 **SIZE<4:0>：**区域大小位
区域的权限仅在大小为非零时有效。区域大小 = $2^{(大小-1)}$ x 1024 字节。
11111 = 保留
•
•
•
11000 = 保留
10111 = 4,294,967,296
10110 = 2,147,483,648
10101 = 1,073,741,824
10100 = 536,870,912
10011 = 268,435,456
10010 = 134,217,728
10001 = 67,108,864
10000 = 33,554,432
01111 = 16,777,216
01110 = 8,388,608
01101 = 4,194,304
01100 = 2,097,152
01011 = 1,048,576
01010 = 524,288
01001 = 262,144
01000 = 131,072
00111 = 65,536
00110 = 32,768
00101 = 16,384
00100 = 8,192
00011 = 4,096
00010 = 2,048
00001 = 1,024
00000 = 区域不存在

bit 2-0 **未实现：**读为 0

注 1： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。
2： 对于一些目标区域，该寄存器中的某些位为只读，且具有预设值。更多信息，请参见具体器件数据手册中的 **“存储器构成”**章节。

© 2014 Microchip Technology Inc.

寄存器 48-8： SBTxRDy：系统总线目标 x 区域 y 读权限寄存器 （x = 0-13 ； y = 0-8）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31-4 **未实现：**读为 0

bit 3 **GROUP3：**组 3 读权限位
1 = 权限组 3 具有读权限
0 = 权限组 3 不具有读权限

bit 2 **GROUP2：**组 2 读权限位
1 = 权限组 2 具有读权限
0 = 权限组 2 不具有读权限

bit 1 **GROUP1：**组 1 读权限位
1 = 权限组 1 具有读权限
0 = 权限组 1 不具有读权限

bit 0 **GROUP0：**组 0 读权限位
1 = 权限组 0 具有读权限
0 = 权限组 0 不具有读权限

注 1： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。
2： 对于一些目标区域，该寄存器中的某些位为只读，且具有预设值。更多信息，请参见具体器件数据手册中的 **“存储器构成”**章节。

寄存器 48-9： SBTxWRy：系统总线目标 x 区域 y 写权限寄存器 （x = 0-13 ； y = 0-8）

位范围	Bit 31/23/15/7	Bit 30/22/14/6	Bit 29/21/13/5	Bit 28/20/12/4	Bit 27/19/11/3	Bit 26/18/10/2	Bit 25/17/9/1	Bit 24/16/8/0
31:24	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
23:16	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
15:8	U-0	U-0	U-0	U-0	U-0	U-0	U-0	U-0
	—	—	—	—	—	—	—	—
7:0	U-0	U-0	U-0	U-0	R/W-1	R/W-1	R/W-1	R/W-1
	—	—	—	—	GROUP3	GROUP2	GROUP1	GROUP0

图注：		
R = 可读位	W = 可写位	U = 未实现位，读为 0
-n = POR 时的值	1 = 置 1	0 = 清零

bit 31-4 **未实现：**读为 0

bit 3 **GROUP3：**组 3 写权限位
1 = 权限组 3 具有写权限
0 = 权限组 3 不具有写权限

bit 2 **GROUP2：**组 2 写权限位
1 = 权限组 2 具有写权限
0 = 权限组 2 不具有写权限

bit 1 **GROUP1：**组 1 写权限位
1 = 权限组 1 具有写权限
0 = 权限组 1 不具有写权限

bit 0 **GROUP0：**组 0 写权限位
1 = 权限组 0 具有写权限
0 = 权限组 0 不具有写权限

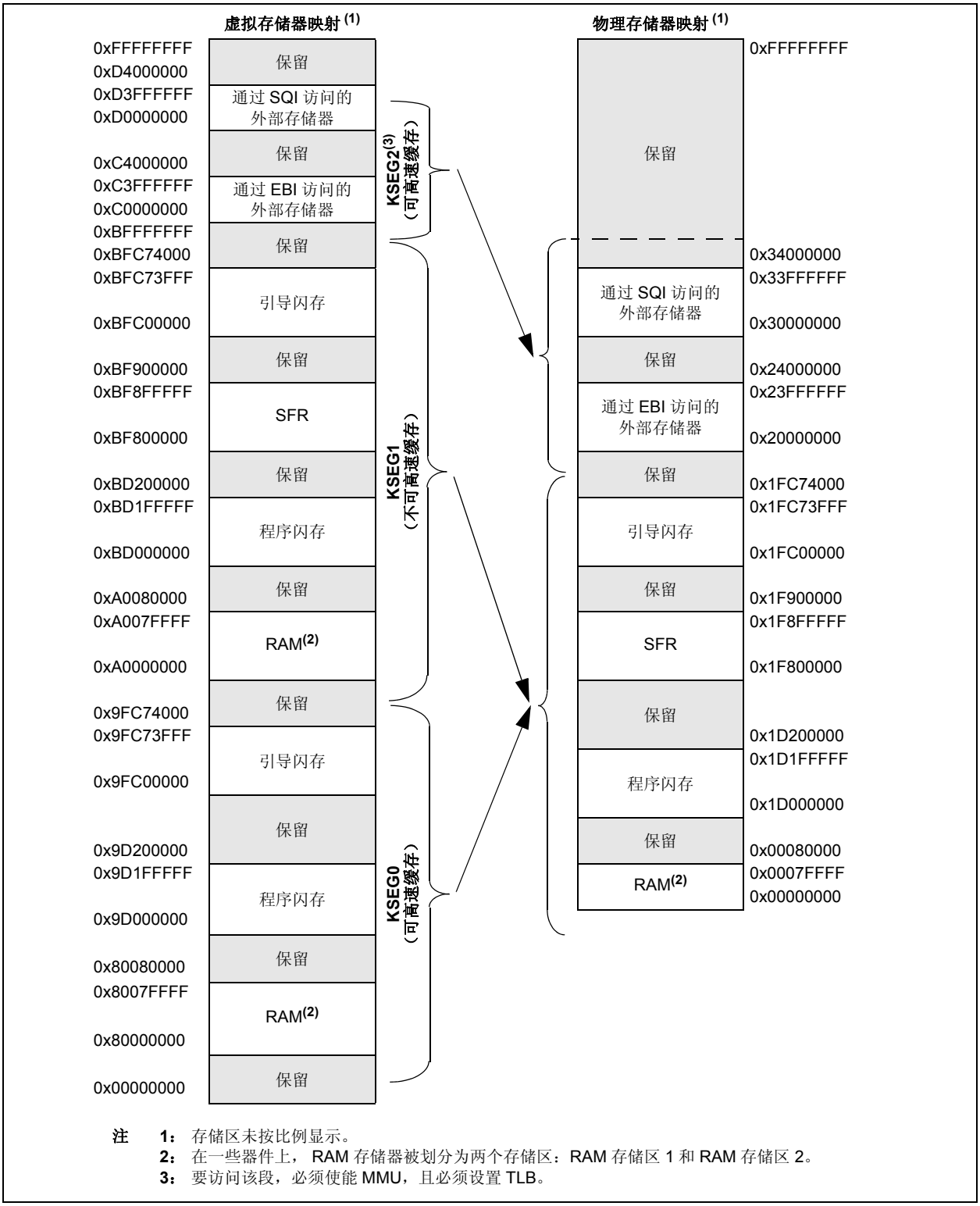
注 1： 关于可用系统总线目标的列表以及它们的说明，请参见具体器件数据手册中的 **“存储器构成”**章节。
2： 对于一些目标区域，该寄存器中的某些位为只读，且具有预设值。更多信息，请参见具体器件数据手册中的 **“存储器构成”**章节。

© 2014 Microchip Technology Inc.

48.3 存储器布局

图 48-1 给出了一个存储器映射示例，关于具有 2048 KB 程序存储器的器件在内核模式下运行典型应用程序。右侧所示的物理地址由 DMA、闪存控制器 （对闪存进行编程时）使用，以及在定义受保护存储器区域时使用。虚拟地址由 CPU 使用，用于获取和读 / 写数据或外设 SFR。

图 48-1： 具有 2048 KB 程序存储器的器件的存储器映射



注 **1：** 存储区未按比例显示。
2： 在一些器件上， RAM 存储器被划分为两个存储区：RAM 存储区 1 和 RAM 存储区 2。
3： 要访问该段，必须使能 MMU，且必须设置 TLB。

对于在内核模式下运行的典型应用程序，虚拟存储器划分为3个段，名为KSEG0、KSEG1和KSEG2。段KSEG0和KSEG1均转换为物理地址0x0，并包括所有程序闪存和数据存储器；但是，KSEG0可高速缓存，KSEG1不可高速缓存。这种安排使CPU可以通过虚拟段KSEG0和KSEG1访问相同的物理地址空间，从而使应用程序可以选择通过跳转或调用高速缓存区域或非高速缓存区域中的函数，以高速缓存或非高速缓存的形式执行任意或所有代码。非高速缓存区域KSEG1提供转换为PIC32MZ系列器件特殊功能寄存器的虚拟地址空间转换。KSEG2映射到外部存储器，外部存储器通过SQI和EBI模块使用页表缓冲区（Translation Lookaside Buffer，TLB）与器件连接。

此外还有两个额外的虚拟存储器段：用户段KUSEG，它占用虚拟存储器的低2 GB空间；KSEG3，它占用存储器的高512 MB区块。关于使用这些段和使用TLB配置它们的信息，请参见**第50章“采用 microAptiv™ 内核的器件的 CPU”**（DS60001192）。

48.3.1 虚拟地址与物理地址相互转换的计算

要将 KSEG0 或 KSEG1 中的虚拟地址转换为物理地址，需要使用 0x1FFFFFFF 对虚拟地址执行“位与”运算：

- 物理地址 = 虚拟地址 & 0x1FFFFFFF

要将物理地址转换为 KSEG0 虚拟地址，需要使用 0x80000000 对物理地址执行 “位或”运算：

- KSEG0 虚拟地址 = 物理地址 | 0x80000000

要将物理地址转换为 KSEG1 虚拟地址，需要使用 0xA0000000 对物理地址执行 “位或”运算：

- KSEG1 虚拟地址 = 物理地址 | 0xA0000000

要从 KSEG0 虚拟地址转换为 KSEG1 虚拟地址，需要使用 0x20000000 对 KSEG0 虚拟地址执行“位或”运算：

- KSEG1 虚拟地址 = KSEG0 虚拟地址 | 0x20000000

（由于 KSEG2 使用 TLB 进行映射，所以与其相关的虚拟地址转换为物理地址的计算取决于具体配置）

48.3.2 引导闪存

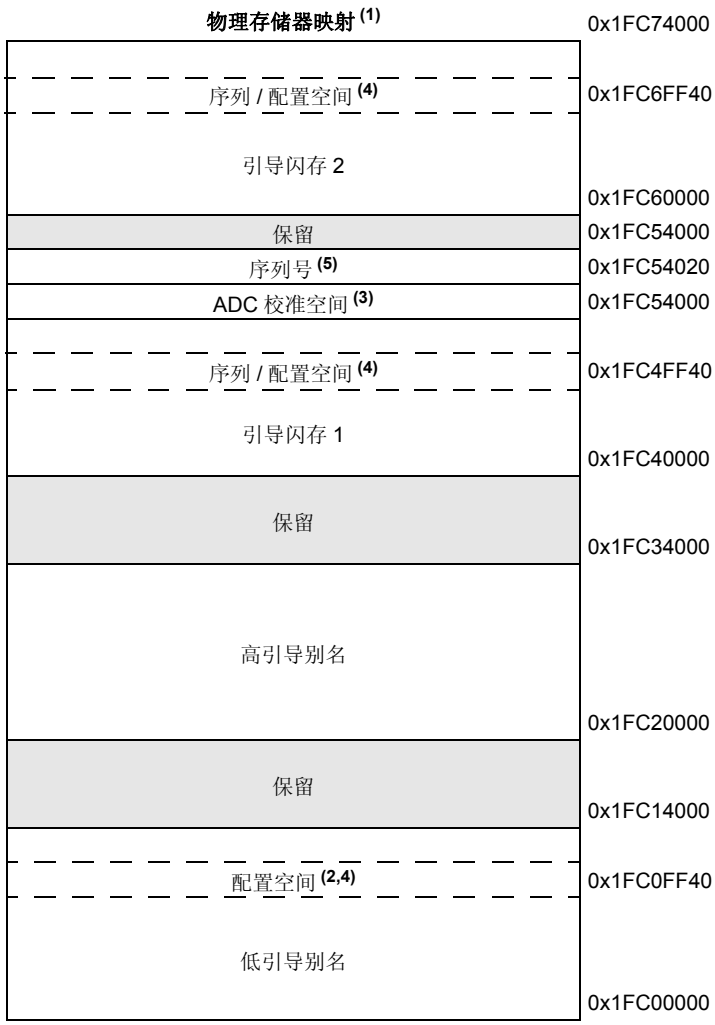
引导闪存（Boot Flash Memory， BFM）区域具有可方便双重引导实现的特殊功能。器件实现了两个相同的引导闪存区域，一个处于活动状态（在启动时使用），另一个可用于现场升级。每个BFM区域都存在于固定地址空间（引导闪存1或引导闪存2）和低或高引导别名地址空间中，如图 48-2 所示。

在上电复位（Power-on Reset，POR）时，器件将读取序列字，并将引导闪存1和引导闪存2区域中具有较大序列号的区域映射到低引导别名区域，具有较小序列号的区域映射到高引导别名区域。然后，从低引导别名区域读取配置数据，并最终在位于低引导别名区域起始处的复位向量处（在图 48-2 的示例中为 0x1FC00000）开始代码执行。更多信息，请参见**第 52 章 “支持即时更新的闪存”**（DS60001193）。

序列号存储在序列字的低 16 位。高 16 位设置为低 16 位的补码。这使系统可以知道序列字已设定了序列号。在不使用双重引导但需要将两个引导区域都用于应用程序引导代码的系统中，建议保留序列字，不用于代码或数据存储，以确保启动时的别名操作正确。

© 2014 Microchip Technology Inc.

图 48-2：　引导和别名存储器映射



注　**1：** 存储区未按比例显示。

2： 存储单元 0x1FC0FF40 至 0x1FC0FFFC 用于初始化配置寄存器（请参见具体器件数据手册中的 **“特殊功能”** 章节）。

3： 存储单元 0x1FC54000 至 0x1FC5410 用于初始化 ADC 校准寄存器（请参见具体器件数据手册中的 **“特殊功能”** 章节）。

4： 更多信息，请参见具体器件数据手册中的 **“存储器构成”** 章节。

5： 存储单元 0x1FC54020 和 0x1FC54024 包含惟一的器件序列号（请参见具体器件数据手册中的 **“特殊功能”** 章节）。

48.3.3　双 RAM 存储区

在一些器件上，RAM 存储器被划分为两个存储区，每个存储区都有自己的目标总线接口。两个总线接口使两个不同的总线发起器可以无需仲裁而同时访问RAM。这使用户可以实现这样的系统设计，即仅供 CPU 专用的 RAM 存储器（堆栈和通用数据存储）分配在一个 RAM 存储区中，而供 DMA 专用的 RAM 存储器（数据缓冲区）分配在另一个存储区中。高速外设可以在 CPU RAM 访问期间向 RAM 缓冲区存储或获取数据，而无需进行会影响其性能的仲裁。关于为项目指定和配置 RAM 使用的信息，请参见《MPLAB® XC32 C/C++ 编译器用户指南》（DS50001686G_CN）文档中的**第 17 章 “链接程序”**。

48.4 系统总线

PIC32MZ 系列器件包含了系统总线，它以多层结构的形式实现，通过将多个发起器（总线主器件）与多个目标（总线从器件）连接来支持并发事务。除非两个发起器在同一时刻尝试访问同一目标，否则不会出现仲裁延时。发起器包括 CPU、通用 DMA 和带有专用 DMA 访问的总线主控外设。

以下是典型器件的发起器列表：

- CPU
- 通用 DMA
- 加密引擎
- SQI
- 高速 USB
- CAN
- 以太网
- 闪存控制器（用于运行时自编程（Run-Time Self-Programming， RTSP））

目标包括闪存和 RAM 存储器，以及器件的所有外设，例如 UART、SPI 和 I^2C 等。一些外设组合在一起作为单个目标，构成一条外设总线，它们除了共用目标接口之外，还共用公共外设总线时钟源（PBCLK）。更多信息，请参见具体器件数据手册中的**“器件概述”**和**“振荡器配置”**章节。

带有集成 DMA 的所有总线主控外设除了充当发起器之外，也充当目标。这是访问其配置 SFR 所必需的。图 48-3 给出了 PIC32MZ 系统总线的图示。所有数据总线均为 32 位宽。发起器只能访问目标，如图所示。

SRAM 是惟一一种每个发起器均可访问的目标。在大多数应用中， SRAM 将是 CPU 操作以及通信外设和数据缓冲区之间的 DMA 传输的频繁访问目标。为了避免在这些情况下发生仲裁，许多 PIC32MZ 实现提供了两个 SRAM 存储区。建议通过非高速缓存的存储器区域（KSEG1）来访问 DMA 处理的缓冲区，否则会产生高速缓存一致性问题。

© 2014 Microchip Technology Inc.

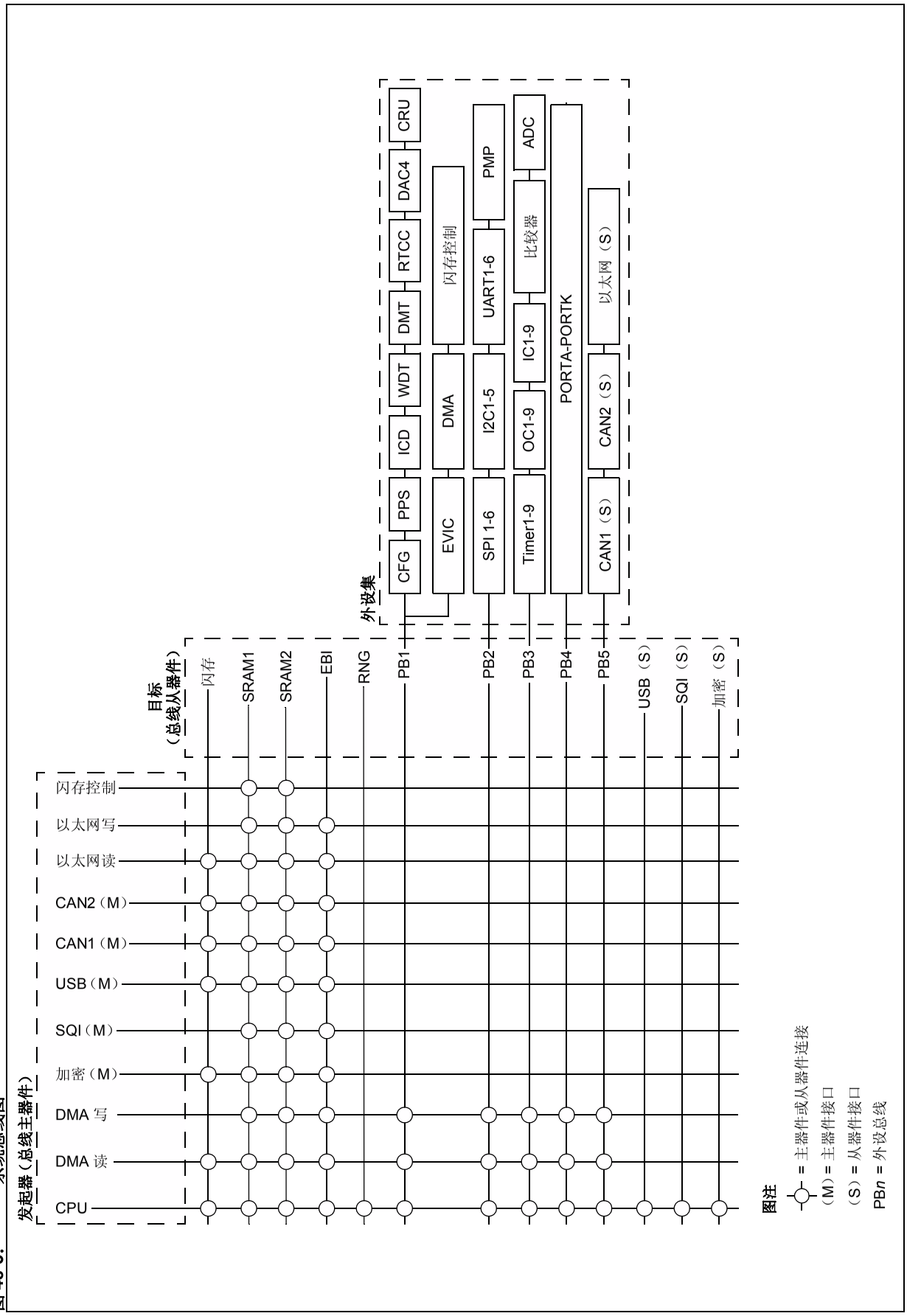

发起器（总线主器件）
CPU
DMA 读
DMA 写
加密（M）
SQI（M）
USB（M）
CAN1（M）
CAN2（M）
以太网读
以太网写
闪存控制
目标
（总线从器件）
闪存
SRAM1
SRAM2
EBI
RNG
PB1
PB2
PB3
PB4
PB5
USB（S）
SQI（S）
加密（S）
外设集
CFG
PPS
ICD
WDT
DMT
RTCC
DAC4
CRU
EVIC
DMA
闪存控制
SPI 1-6
I2C1-5
UART1-6
PMP
Timer1-9
OC1-9
IC1-9
比较器
ADC
PORTA-PORTK
CAN1（S）
CAN2（S）
以太网（S）
图注
= 主器件或从器件连接
（M）= 主器件接口
（S）= 从器件接口
PBn = 外设总线

图 48-3：　系统总线图

48.5 系统总线仲裁

两个发起器不能在同一时刻访问同一目标。如果发生这种情况，则会进行仲裁，即判定两个或多个发起器的访问优先级，并将访问权限授予其中一个发起器，并推迟所有其他发起器的访问。在系统设计中，可以通过将 CPU 和 DMA 发起器的 SRAM 分配划分为两个 SRAM 存储区来最大程度减少或避免仲裁。这也适用于在不同外设总线上提供相同外设的器件。

在需要仲裁时，使用最近最少处理（LRS）优先级的方式进行处理，为大多数发起器提供服务质量（QOS）。闪存控制器发起器是一种带宽极低的发起器，总是使用固定高优先级来确保用于闪存行编程操作的数据。

使用 CPUPRI（CFGCON<24>）配置位，可以为 CPU 分配对于 SRAM 访问的固定高优先级，用于进行中断处理。使用 DMAPRI（CFGCON<25>）配置位，可以为通用 DMA 分配对于 SRAM 访问的固定高优先级。关于 CFGCON 寄存器的信息，请参见具体器件数据手册中的**“特殊功能”**章节。

注：	在更改 DMAPRI 位设置时，必须总是禁止 DMA。

不当使用固定优先级仲裁会对其他发起器产生严重的消极影响。对于使用高带宽的发起器，建议不要使用固定高优先级设置。它旨在用于需要低延时的低带宽应用。设置为固定优先级的发起器将与具有固定高优先级的其他发起器共同进行仲裁。但是，可以通过利用双 RAM 存储区，使具有固定高优先级的 DMA 发起器可以维持低延时，同时对于高带宽总线主控 DMA 发起器仍然维持足够的服务质量。这种情况的一个示例是在具有以太网或高速 USB 的系统中，使用 DMA 来实现低成本无控制器 （Low-cost Controllerless， LCC）图形功能 （要求低延时）。用于图形功能的 DMA 缓冲区放置在一个 RAM 存储区中，而以太网或 USB 缓冲区则放置在另一个 RAM 存储区中。

表 48-2 给出了具有发起器标识符 （ID）和服务质量 （QOS）优先级选项的器件的典型发起器列表的示例。ID 用于标识权限访问违例，如**第 48.6 节“访问权限”**所述。请注意，具有可选 QOS 的发起器对于 HIGH 和 LRS 优先级具有惟一 ID。

表 48-2： 发起器 ID 和 QOS

名称	ID	QOS
CPU	1	LRS
CPU	2	HIGH
DMA 读	3	LRS
DMA 读	4	HIGH
DMA 写	5	LRS
DMA 写	6	HIGH
USB	7	LRS
以太网读	8	LRS
以太网写	9	LRS
CAN1	10	LRS
CAN2	11	LRS
SQI1	12	LRS
闪存控制器	13	HIGH
加密引擎	14	LRS

© 2014 Microchip Technology Inc.

48.6 访问权限

PIC32MZ系列器件提供了可配置存储器保护功能，可用于限制总线发起器对总线目标或总线目标区域的访问。访问违例会被记录下来，并且可以选择产生中断。这些功能通常由操作系统和自举程序使用，用于防止一个任务或应用程序影响另一个任务或应用程序的执行，并且还可以用于帮助阻止不可信应用程序访问受保护的存储器区域，防止知识产权侵权。

每个总线发起器可以分配到 4 个权限组之一。每个总线目标包含一个或多个存储器区域，这些区域可以配置为拒绝或允许每个权限组进行访问。关于典型总线目标和总线发起器的图示，请参见图 48-3。

可以将目标权限违例配置为产生中断，使操作系统可以截获并处理这些事件。这通过将目标的系统总线错误控制寄存器 SBTxECON 的 ERRP 位置 1 来实现，使得在发生权限违例时，该目标在 SBFLAG 寄存器中的位置 1。假定在中断控制器中配置并允许中断，目标在 SBFLAG 寄存器中的位置 1 将导致产生系统总线保护违例中断。更多信息，请参见具体器件数据手册中的**“CPU 异常和中断控制器”**章节。

该中断的服务程序可以查询系统总线的 SBFLAG，以确定是哪个目标产生了违例。此外，它还可以查询系统总线目标寄存器 SBTxELOG1 和 SBTxELOG2，了解关于违例的具体详细信息，包括发生错误的目标区域。最后，可以使用系统总线目标寄存器 SBTxECLRS 和 SBTxECLRM 清除错误。

如果发起器尝试访问受保护的区域，则无论中断允许或报告状态如何，都不会发生写操作。产生违例的读指令将返回值 0x0。

每个发起器的权限组使用 CFGPG 寄存器进行分配（关于该寄存器的信息，请参见具体器件数据手册中的 **“特殊功能”**章节）。在复位时，所有发起器均分配到权限组 0。

发起器对系统总线目标的访问权限使用 SBTxREGy、SBTxRDy 和 SBTxWRy 寄存器进行配置。这 3 个寄存器用于定义存储器区域，以及对于该区域的读权限和写权限。每个目标均包含一组这样的寄存器，对应于默认的存储空间区域，即，整个目标地址空间的大小。此外，大多数目标至少具有一个可定义的（基址和大小）权限存储器区域。基址的高22位通过BASE<21:0>（SBTxREGy<31:10>）位指定，低 10 位固定为 0。基址必须与 SIZE<4:0>（SBTxREGy<7:3>）位指定的大小对齐。例如，基址为 9216（0x2400）将仅允许大小为 1024（0x400），而基址为 8192（0x2000）将允许大小为 1024（0x400）、2048（0x800）、4096（0x1000）和 8192（0x2000）。包含 CFGPG、SBTxREGy、SBTxRDy 和 SBTxWRy 寄存器的存储空间本身就是一个具有可定义权限的目标区域（目标 0 - 系统总线）。

当发起器尝试访问在两个目标区域中定义的存储器区域时，优先级将决定需要哪个权限级别。级别 0 优先级最低，级别 3 优先级最高。将总是使用所定义地址空间的最高级别的权限。对于任何目标，区域 0 （默认空间为整个目标的大小）总是为级别 0，而区域 1 总是为级别 3。如果目标具有 0 和 1 以外的区域，则该区域的级别在 SBTxREGy 寄存器的只读位 LEVEL 中定义。在这些情况下， LEVEL 位值为 0 指示优先级为 1，而 LEVEL 位值为 1 指示优先级为 2。永远不要配置具有相同优先级的重叠区域。

初始化时在引导软件中进行权限配置，此时将为每个发起器分配一个权限组，并为所有目标配置权限区域。将假定引导软件是可信代码。在复位时，CPU 将设置为组 0 权限。完成配置之后，引导代码将在跳转到应用程序之前将 CPU 权限设置为适当的组 （禁止对安全区域的访问）。

注：	引导代码必须驻留在这样的页中：对组 0 以及对在 CPUPG<1:0>（CFGPG<1:0>）配置位的设置发生更改时设置的组具有权限。

权限组 0 旨在用作安全权限组，配置为惟一对目标 0 系统总线区域具有权限的组，该区域可用于更改发起器权限组和目标权限，以及应用程序希望保护的任何存储器区域。这样配置的原因是为了在 CPU 权限发生更改之后，阻止对安全目标区域进行访问，惟一可以用于更改 CPU 权限的运行时机制是不可屏蔽中断（Non-Maskable Interrupt，NMI）。在发生 NMI 时，CPU 权限将更改为组 0，之后再通过向量跳转到中断服务程序（Interrupt Service Routine，ISR）。NMI 服务程序必须驻留在允许组 0 权限的存储器中。在 NMI 之后，CPU 具有对安全存储器区域的访问权限。在完成安全操作之后，软件必须将 CPU 权限更改为适当的组，采用在引导代码中启动时所用的方式阻止对安全区域的访问。

权限组 3 是在处理器处于调试模式时使用的特殊权限组。在调试模式下，处理器处于一种由外部调试器控制的状态，从而可以检查或更改存储器或 SFR。DBGPER<2:0>（DEVCFG0<14:12>）位用于选择组 0、1 和 2 在调试模式下的权限。当处理器处于调试模式时，由 CPU1PG<1:0>（CFGPG<1:0>）位定义的 CPU 权限设置为由 DBGPER<2:0> 位指定的被拒绝权限组之一，事务请求将被分配组 3 权限。系统总线目标区域权限寄存器 SBTxRDy 和 SBTxWRy 将复位为允许访问组 0 至 2 并拒绝访问组 3 的权限设置。通过使用 DBGPER<2:0> 位，可以阻止调试器进行访问，直到受保护的引导代码定义了指定哪些存储器区域可由调试器访问的权限为止。这些功能需要设计为支持它们的调试工具。此外，EJTAGBEN（DEVCFG0<30>）位应设置为 0，并且应使能代码保护。

48.6.1 中断向量和服务程序

PIC32MZ 系列器件允许将 ISR 和异常处理程序放置在系统存储器中的任意位置。必须小心确保包含 ISR 和异常处理代码的目标区域的读权限包含了软件可能用于 CPU 的所有权限。 NMI 中断属于特殊情况，它总是向量跳转到通常用于引导代码的固定复位或 NMI 位置。

48.6.2 外设权限

将外设寄存器与中断配合使用时，还必须为外设寄存器分配读权限和写权限，允许可能在目标应用程序中为 CPU 设置的任何权限组进行访问。如果需要对外设的访问进行保护，阻止 CPU 进行访问，则必须使用 DMA（通用 DMA 或与总线主器件关联的 DMA）。在这种情况下，可以为 DMA 和目标外设分配一个权限组，而不考虑 CPU 使用的权限组。

48.6.3 高速缓存

在发起器访问目标存储器时，将会检查权限。如果目标存储器为 SRAM 或闪存，并且使用了高速缓存，则对目标的访问在指令或数据在目标和高速缓存之间传送时发生。从高速缓存中访问内容时，不会检查 CPU 的权限。如果高速缓存中包含高权限的权限组数据，而 CPU 权限更改为较低的权限组，则必须清空高速缓存，以防止权限较低的 CPU 设置进行访问。清空高速缓存应在更改 CPU 权限设置之前完成。

48.6.4 别名和存储区交换存储器

在对存储器进行别名操作的器件上，如果同时存在相同存储器的固定和别名区域，则必须对这两个区域都进行保护。此外，可能还需要保护闪存控制器的 SWAP （NVMCON<7>）位，仅允许安全权限组进行访问。

© 2014 Microchip Technology Inc.

48.7 复位的影响

48.7.1 复位时

RAM 的内容保持不变， CPU 设置为权限组 0。所有目标区域均设置为允许所有 4 个权限组进行访问。可编程目标区域和大小均设置为其复位值。

48.7.2 上电或欠压复位时

RAM 的内容是未定义的， CPU 设置为权限组 0。所有目标区域均设置为允许所有 4 个权限组进行访问。可编程目标区域和大小均设置为其复位值。

48.8 节能模式下的操作

48.8.1 休眠模式

当器件从休眠模式唤醒时， RAM 的内容保持不变， CPU 权限组保持不变。目标区域的设置保持不变。

48.8.2 空闲模式

当器件退出空闲模式时，CPU 权限组保持不变。目标区域的设置保持不变。退出空闲模式不会导致 RAM 内容发生更改；但是，可以在空闲模式下使能运行的 DMA 外设可能会在 CPU 处于空闲模式时修改 RAM 内容。

48.9 调试模式

为了防止调试器在设置引导代码保护方案之前访问受保护区域，可以通过将EJTAGBEN（DEVCFG0<30>）位设置为 0 来限制 EJTAG 功能，防止探针能够从 EJTAG 进行引导。

此外，器件配置字 DEVCFG0 包含 DBGPER<2:0> （DEVCFG0<6:4>）位，这些位与器件调试工具配合使用来控制调试模式期间的访问。更多信息，请参见具体器件数据手册中的**“特殊功能”**章节和相关的开发工具文档。此外，关于在进行调试时如何使用权限组 3，请参见**第 48.6 节 “访问权限”**。

注：	权限组不会阻止调试代码在受保护区域中设置跟踪。如果未通过清零 TRCEN（DEVCFG0<5>）位来禁止跟踪，并且如果需要保护，则必须禁止EJTAG串行执行。

48.10 代码示例

例 48-1 说明了如何为支持两个应用程序代码空间的器件设置闪存权限。应用程序 1（其中包含自举软件）是可信代码。应用程序 2 不是可信代码，将阻止它访问应用程序 1 闪存区域。

在该示例中，该器件包含具有固定和别名区域的双重引导存储区。应用程序仅使用低别名引导区域。低别名引导闪存包含 5 个 16 KB 存储器页，总共 80 KB。页 0 至页 3 被选择作为应用程序 1 的引导代码的空间（64 KB）。必须使用页 0，因为它包含复位和 NMU 向量。引导闪存页 4（16 KB）被指定为中断向量表（Interrupt Vector Table，IVT）和将 CPU 权限从应用程序 1 更改为应用程序 2 的代码所需的双重权限空间。在这两种情况下，在 CPU 处于应用程序 1 或应用程序 2 的权限组中时，该代码都必须能够执行。

在该示例中，器件包含 2 MB 的程序闪存，它在两个应用程序之间平均分配，上半部分分配给应用程序 2，下半部分分配给应用程序 1。

注： 由于进行双重引导，所以必须使用相同的方案来保护两个区域（别名和固定区域）。

在复位时，CPU 设置为组 0 的权限，它将用作应用程序 1 的组权限。应用程序 1 将进行初始化，这包括设置如代码示例中所定义的目标区域和权限。当应用程序 1 准备好将控制权交给应用程序 2 时，它会将 CPU 权限组从 0 更改为应用程序 2 的权限组，也就是该示例中使用的权限组 1。CPU 权限组更改发生在过渡区域中执行的代码中，然后再跳转到应用程序 2 中的启动位置。

应用程序 2 可以通过在软件中产生 NMI 而协作式地将控制权交回给应用程序 1，或者也可以使用看门狗定时器 （Watchdog Timer， WDT）来抢占式地执行应用程序 2，并将控制权返回给应用程序 1。

© 2014 Microchip Technology Inc.

例 48-1：　两个应用程序代码空间的闪存权限

为了实现前面所述的权限要求，必须定义区域和设置 IVT/ 过渡区域和应用程序 2 代码空间的权限。然后，包括所有闪存的闪存默认区域将设置为仅允许由组 0 访问，该组是应用程序 1 的权限组。最后，定义系统总线目标权限，仅允许由应用程序 1 访问。

IVT/ 过渡区域和应用程序 2 代码空间的权限将优先于默认权限，因为它们是在具有较高优先级的寄存器中定义的。例 48-2 给出了可用于为两个应用程序空间设置权限的代码。

注：　例 48-2 给出的代码仅处理闪存的权限设置。它并不包括 DMA 发起器或 RAM、外部存储器和外设目标在其他方面的权限控制。

例 48-2：　　为两个应用程序空间设置权限的代码

```
// Transitory Region, last page of Lower Aliased Boot Memory
// Must be configured for the aliased and fixed regions
    SBT1REG3bits.BASE = 0x1FC10000 >> 10;// Lower aliased region, starting address
    SBT1REG3bits.SIZE = 0x5; // Lower aliased size is one 16 KB page
    SBT1REG4bits.BASE = 0x1FC50000 >> 10; // Lower fixed region, starting address
    SBT1REG4bits.SIZE = 0x5; // Size is identical to alias

// Set permissions for both aliased and fixed regions
    SBT1RD3 = 0x3; // Allow access by Group 0 and 1
                   // permissions and block access to Group 2 and 3 permissions
    SBT1RD4 = 0x3; // Duplicate for the fixed region

// Application 2 Region, upper half of PFM
    SBT1REG7bits.BASE = 0x1D100000 >> 10; // PFM upper region, starting address
    SBT1REG7bits.SIZE = 0xB; // Upper PFM, 1 MB size
    SBT1RD7 = 0x2; // Allow access only for Group 1 permissions while blocking
                   // access by groups 0, 2 and 3

// Set default region to allow access only for Group 0
    SBT1RD0 = 0x1;

// Set the System Bus to allow access to only group 0
    SBT0RD0 = SBT0WR0 = 0x01;
    SBT0RD1 = SBT0WR1 = 0x01;
```


© 2014 Microchip Technology Inc.

48.11 相关应用笔记

本节列出了与手册本章内容相关的应用笔记。这些应用笔记可能并不是专为 PIC32 器件系列而编写的，但其概念是相近的，通过适当修改并受到一定限制即可使用。当前与存储器构成和权限相关的应用笔记有：

标题	应用笔记编号
目前没有相关的应用笔记。	N/A

注： 如需获取更多 PIC32 系列器件的应用笔记和代码示例，请访问 Microchip 网站（www.microchip.com）。

48.12 版本历史

版本 A （2013 年 11 月）

这是本文档的初始版本。

© 2014 Microchip Technology Inc.

请注意以下有关 Microchip 器件代码保护功能的要点：

- Microchip 的产品均达到 Microchip 数据手册中所述的技术指标。
- Microchip 确信：在正常使用的情况下， Microchip 系列产品是当今市场上同类产品中最安全的产品之一。
- 目前，仍存在着恶意、甚至是非法破坏代码保护功能的行为。就我们所知，所有这些行为都不是以 Microchip 数据手册中规定的操作规范来使用 Microchip 产品的。这样做的人极可能侵犯了知识产权。
- Microchip 愿与那些注重代码完整性的客户合作。
- Microchip 或任何其他半导体厂商均无法保证其代码的安全性。代码保护并不意味着我们保证产品是 “牢不可破”的。

代码保护功能处于持续发展中。 Microchip 承诺将不断改进产品的代码保护功能。任何试图破坏 Microchip 代码保护功能的行为均可视为违反了《数字器件千年版权法案 （Digital Millennium Copyright Act）》。如果这种行为导致他人在未经授权的情况下，能访问您的软件或其他受版权保护的成果，您有权依据该法案提起诉讼，从而制止这种行为。

提供本文档的中文版本仅为了便于理解。请勿忽视文档中包含的英文部分，因为其中提供了有关 Microchip 产品性能和使用情况的有用信息。Microchip Technology Inc. 及其分公司和相关公司、各级主管与员工及事务代理机构对译文中可能存在的任何差错不承担任何责任。建议参考 Microchip Technology Inc. 的英文原版文档。

本出版物中所述的器件应用信息及其他类似内容仅为您提供便利，它们可能由更新之信息所替代。确保应用符合技术规范，是您自身应负的责任。**Microchip 对这些信息不作任何明示或暗示、书面或口头、法定或其他形式的声明或担保，包括但不限于针对其使用情况、质量、性能、适销性或特定用途的适用性的声明或担保。** Microchip 对因这些信息及使用这些信息而引起的后果不承担任何责任。如果将 Microchip 器件用于生命维持和 / 或生命安全应用，一切风险由买方自负。买方同意在由此引发任何一切伤害、索赔、诉讼或费用时，会维护和保障 Microchip 免于承担法律责任，并加以赔偿。在 Microchip 知识产权保护下，不得暗中或以其他方式转让任何许可证。

商标

Microchip 的名称和徽标组合、 Microchip 徽标、 dsPIC、 FlashFlex、 KEELOQ、 KEELOQ 徽标、 MPLAB、 PIC、 PICmicro、 PICSTART、 PIC32 徽标、 rfPIC、 SST、 SST 徽标、 SuperFlash 和 UNI/O 均为 Microchip Technology Inc. 在美国和其他国家或地区的注册商标。

FilterLab、 Hampshire、 HI-TECH C、 Linear Active Thermistor、 MTP、 SEEVAL 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Silicon Storage Technology 为 Microchip Technology Inc. 在除美国外的国家或地区的注册商标。

Analog-for-the-Digital Age、 Application Maestro、 BodyCom、 chipKIT、 chipKIT 徽标、 CodeGuard、 dsPICDEM、 dsPICDEM.net、 dsPICworks、 dsSPEAK、 ECAN、 ECONOMONITOR、 FanSense、 HI-TIDE、 In-Circuit Serial Programming、 ICSP、 Mindi、 MiWi、 MPASM、 MPF、 MPLAB Certified 徽标、 MPLIB、 MPLINK、 mTouch、 Omniscient Code Generation、 PICC、 PICC-18、 PICDEM、 PICDEM.net、 PICkit、 PICtail、 REAL ICE、 rfLAB、 Select Mode、 SQI、 Serial Quad I/O、 Total Endurance、 TSHARC、 UniWinDriver、 WiperLock、 ZENA 和 Z-Scale 均为 Microchip Technology Inc. 在美国和其他国家或地区的商标。

SQTP 是 Microchip Technology Inc. 在美国的服务标记。

GestIC 和 ULPP 为 Microchip Technology Inc. 的子公司 Microchip Technology Germany II GmbH & Co. & KG 在除美国外的国家或地区的注册商标。

在此提及的所有其他商标均为各持有公司所有。

© 2014, Microchip Technology Inc. 版权所有。

ISBN：978-1-62077-838-8

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
═ ISO/TS 16949═

Microchip 位于美国亚利桑那州 Chandler 和 Tempe 与位于俄勒冈州 Gresham 的全球总部、设计和晶圆生产厂及位于美国加利福尼亚州和印度的设计中心均通过了 ISO/TS-16949:2009 认证。 Microchip 的 PIC® MCU 与 dsPIC® DSC、KEELOQ® 跳码器件、串行 EEPROM、单片机外设、非易失性存储器和模拟产品严格遵守公司的质量体系流程。此外， Microchip 在开发系统的设计和生产方面的质量体系也已通过了 ISO 9001:2000 认证。

全球销售及服务网点

美洲

公司总部 Corporate Office
2355 West Chandler Blvd.
Chandler, AZ 85224-6199
Tel: 1-480-792-7200
Fax: 1-480-792-7277
技术支持：
http://www.microchip.com/support
网址：www.microchip.com

亚特兰大 Atlanta
Duluth, GA
Tel: 1-678-957-9614
Fax: 1-678-957-1455

奥斯汀 Austin, TX
Tel: 1-512-257-3370

波士顿 Boston
Westborough, MA
Tel: 1-774-760-0087
Fax: 1-774-760-0088

芝加哥 Chicago
Itasca, IL
Tel: 1-630-285-0071
Fax: 1-630-285-0075

克里夫兰 Cleveland
Independence, OH
Tel: 1-216-447-0464
Fax: 1-216-447-0643

达拉斯 Dallas
Addison, TX
Tel: 1-972-818-7423
Fax: 1-972-818-2924

底特律 Detroit
Novi, MI
Tel: 1-248-848-4000

休斯敦 Houston, TX
Tel: 1-281-894-5983

印第安纳波利斯 Indianapolis
Noblesville, IN
Tel: 1-317-773-8323
Fax: 1-317-773-5453

洛杉矶 Los Angeles
Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

纽约 New York, NY
Tel: 1-631-435-6000

圣何塞 San Jose, CA
Tel: 1-408-735-9110

加拿大多伦多 Toronto
Tel: 1-905-673-0699
Fax: 1-905-673-6509

亚太地区

亚太总部 Asia Pacific Office
Suites 3707-14, 37th Floor
Tower 6, The Gateway
Harbour City, Kowloon
Hong Kong
Tel: 852-2401-1200
Fax: 852-2401-3431

中国 - 北京
Tel: 86-10-8569-7000
Fax: 86-10-8528-2104

中国 - 成都
Tel: 86-28-8665-5511
Fax: 86-28-8665-7889

中国 - 重庆
Tel: 86-23-8980-9588
Fax: 86-23-8980-9500

中国 - 杭州
Tel: 86-571-2819-3187
Fax: 86-571-2819-3189

中国 - 香港特别行政区
Tel: 852-2943-5100
Fax: 852-2401-3431

中国 - 南京
Tel: 86-25-8473-2460
Fax: 86-25-8473-2470

中国 - 青岛
Tel: 86-532-8502-7355
Fax: 86-532-8502-7205

中国 - 上海
Tel: 86-21-5407-5533
Fax: 86-21-5407-5066

中国 - 沈阳
Tel: 86-24-2334-2829
Fax: 86-24-2334-2393

中国 - 深圳
Tel: 86-755-8864-2200
Fax: 86-755-8203-1760

中国 - 武汉
Tel: 86-27-5980-5300
Fax: 86-27-5980-5118

中国 - 西安
Tel: 86-29-8833-7252
Fax: 86-29-8833-7256

中国 - 厦门
Tel: 86-592-238-8138
Fax: 86-592-238-8130

中国 - 珠海
Tel: 86-756-321-0040
Fax: 86-756-321-0049

亚太地区

台湾地区 - 高雄
Tel: 886-7-213-7830

台湾地区 - 台北
Tel: 886-2-2508-8600
Fax: 886-2-2508-0102

台湾地区 - 新竹
Tel: 886-3-5778-366
Fax: 886-3-5770-955

澳大利亚 Australia - Sydney
Tel: 61-2-9868-6733
Fax: 61-2-9868-6755

印度 India - Bangalore
Tel: 91-80-3090-4444
Fax: 91-80-3090-4123

印度 India - New Delhi
Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune
Tel: 91-20-3019-1500

日本 Japan - Osaka
Tel: 81-6-6152-7160
Fax: 81-6-6152-9310

日本 Japan - Tokyo
Tel: 81-3-6880-3770
Fax: 81-3-6880-3771

韩国 Korea - Daegu
Tel: 82-53-744-4301
Fax: 82-53-744-4302

韩国 Korea - Seoul
Tel: 82-2-554-7200
Fax: 82-2-558-5932 或
82-2-558-5934

马来西亚 Malaysia - Kuala Lumpur
Tel: 60-3-6201-9857
Fax: 60-3-6201-9859

马来西亚 Malaysia - Penang
Tel: 60-4-227-8870
Fax: 60-4-227-4068

菲律宾 Philippines - Manila
Tel: 63-2-634-9065
Fax: 63-2-634-9069

新加坡 Singapore
Tel: 65-6334-8870
Fax: 65-6334-8850

泰国 Thailand - Bangkok
Tel: 66-2-694-1351
Fax: 66-2-694-1350

欧洲

奥地利 Austria - Wels
Tel: 43-7242-2244-39
Fax: 43-7242-2244-393

丹麦 Denmark-Copenhagen
Tel: 45-4450-2828
Fax: 45-4485-2829

法国 France - Paris
Tel: 33-1-69-53-63-20
Fax: 33-1-69-30-90-79

德国 Germany - Dusseldorf
Tel: 49-2129-3766400

德国 Germany - Munich
Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

德国 Germany - Pforzheim
Tel: 49-7231-424750

意大利 Italy - Milan
Tel: 39-0331-742611
Fax: 39-0331-466781

意大利 Italy - Venice
Tel: 39-049-7625286

荷兰 Netherlands - Drunen
Tel: 31-416-690399
Fax: 31-416-690340

波兰 Poland - Warsaw
Tel: 48-22-3325737

西班牙 Spain - Madrid
Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

瑞典 Sweden - Stockholm
Tel: 46-8-5090-4654

英国 UK - Wokingham
Tel: 44-118-921-5800
Fax: 44-118-921-5820

10/28/13

© 2014 Microchip Technology Inc.