
第 4 章 预取高速缓存模块

目录

本章包括下列主题：

4.1	简介	4-2
4.2	高速缓存概述	4-3
4.3	控制寄存器	4-7
4.4	高速缓存工作方式	4-27
4.5	高速缓存配置	4-27
4.6	一致性支持	4-30
4.7	复位的影响	4-31
4.8	设计技巧	4-31
4.9	节能模式下的操作	4-32
4.10	相关应用笔记	4-33
4.11	版本历史	4-34

4.1 简介

注： 预取高速缓存仅在一些选定器件上提供。关于特定器件上是否提供预取高速缓存模块的信息，请参见相应的数据手册。

本节介绍 PIC32MX 器件系列中的预取高速缓存模块的功能和工作方式。预取高速缓存功能可以提高大多数应用程序的系统性能。

PFM 高速缓存和预取高速缓存模块实现了以下功能，这些功能提高了在可高速缓存的程序闪存存储器（Program Flash Memory, PFM）区域之外执行的应用程序的性能：

- 指令高速缓存
16 线高速缓存可以每个时钟提供一条指令，最长可循环 256 个字节。
- 数据高速缓存
预取高速缓存还可以为数据存储最多分配 4 条高速缓存线，提高对于闪存存储的常量数据的访问性能。
- 预测性指令预取
对于线性代码，预取高速缓存模块可以在程序计数器之前预取指令，即使不进行高速缓存也可以实现每个时钟提供一条指令的速率，隐去闪存的访问时间。

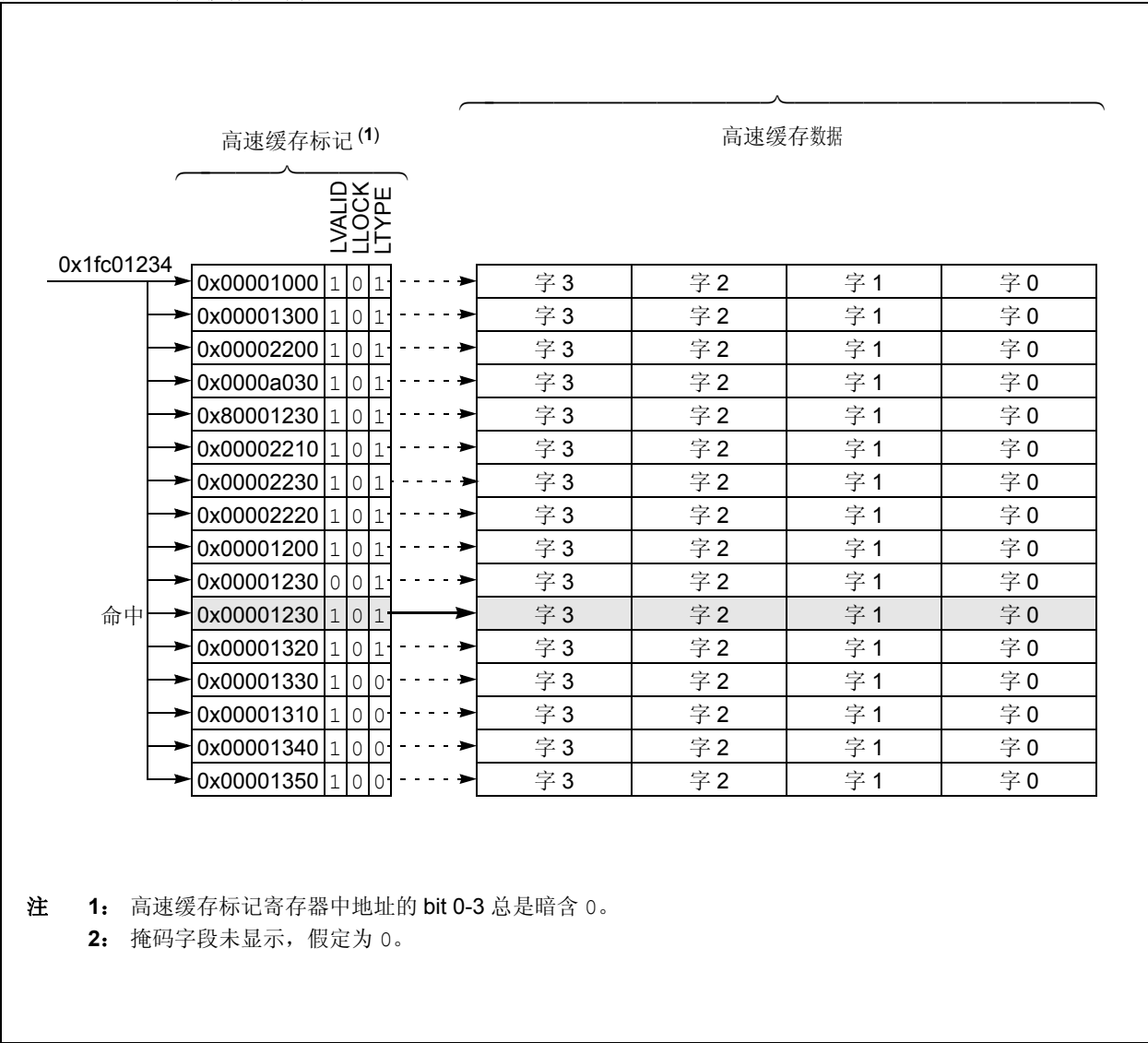
4.1.1 其他预取高速缓存模块功能

预取高速缓存模块还包含以下功能：

- 16 条完全关联的可锁定高速缓存线
- 16 字节高速缓存线
- 最多可为数据分配 4 条高速缓存线
- 2 条带有地址掩码的高速缓存线，用于保存重复的指令
- 伪最近最少使用算法（Least-Recently-Used, LRU）替换策略
- 可用软件写所有高速缓存线
- 16 字节并行存储器取操作
- 预测性指令预取高速缓存

为了说明预取高速缓存的基本工作方式，图 4-2 给出了一个 CPU 从物理地址 0x1FC01234 处请求数据的示例。预取高速缓存会将该地址与所有标为“有效”的标记同时进行比较。由于下面的阴影条目具有该地址，并标为“有效”，因此发生高速缓存命中。然后，数据阵列中相应的数据字将在单个时钟周期中送到 CPU 中。

图 4-2: 高速缓存查询示例⁽²⁾



注 1: 高速缓存标记寄存器中地址的 bit 0-3 总是暗含 0。
2: 掩码字段未显示，假定为 0。

4.2.1 高速缓存构成

高速缓存包含两个阵列：标记和数据。数据阵列可以包含程序指令或程序数据。高速缓存会进行物理标记，并且地址基于物理地址而不是虚拟地址进行匹配。

标记阵列中的每条线包含以下信息：

- 掩码——地址掩码值
- 标记——用于进行匹配的标记地址
- 有效位
- 锁定位
- 类型——指令和 / 或数据类型指示位

数据阵列中的每条线包含 16 字节的程序指令或程序数据，具体取决于类型指示位的值。

图 4-3 显示了每条线的构成。请注意，并不是每条线的 LMASK（CHEMSK<15:5>）和 LTYPE（CHETAG<1>）字段都是可编程的。LTAG（CHETAG<23:4>）字段仅实现了完全映射 PFM 大小所需的位数；例如，如果闪存大小为 512 KB，则 LTAG（CHETAG<23:4>）字段仅实现 bit 18 至 bit 4。

图 4-3: 掩码线

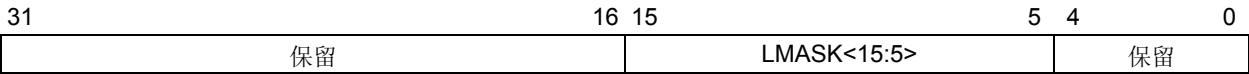


图 4-4: 标记线

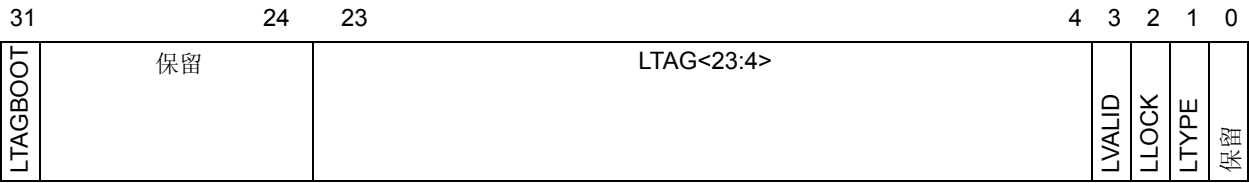


图 4-5: 数据线

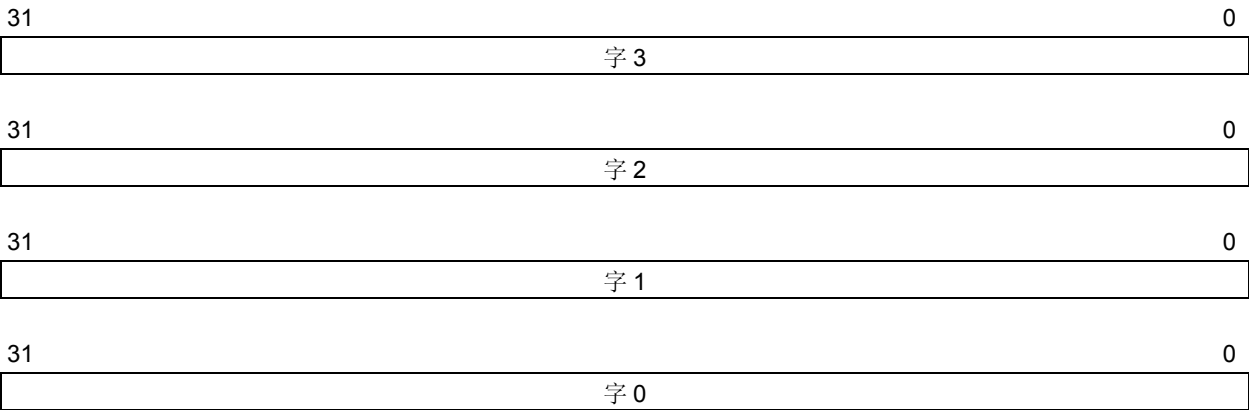


表 4-1 显示了高速缓存阵列。软件可以修改高速缓存标记线和数据线中的值。配置寄存器字段 CHEIDX (CHEACC<3:0>) 可用于选择要访问的线。然后，可以通过 CHETAG、CHEMSK、CHEW0、CHEW1、CHEW2 和 CHEW3 寄存器修改该线的值。

表 4-1: 高速缓存阵列

高速缓存阵列						数据阵列 (2)			
线编号	标记阵列								
0	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
1	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
2	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
3	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
4	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
5	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
6	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
7	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
8	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
9	000h ⁽¹⁾	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
A	掩码	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
B	掩码	标记	V	L	T ⁽³⁾	字 3	字 2	字 1	字 0
C	000h ⁽¹⁾	标记	V	L	T	字 3	字 2	字 1	字 0
D	000h ⁽¹⁾	标记	V	L	T	字 3	字 2	字 1	字 0
E	000h ⁽¹⁾	标记	V	L	T	字 3	字 2	字 1	字 0
F	000h ⁽¹⁾	标记	V	L	T	字 3	字 2	字 1	字 0

- 注 **1:** 只读字段。
 2: 在器件受代码保护时读为 0。否则，可进行读 / 写。
 3: 类型固定为指令。

建议在从不可高速缓存地址执行代码时修改高速缓存线，因为从可高速缓存地址执行代码时，高速缓存控制器并不会防止修改高速缓存。

并不是所有字段都可写。LMASK (CHEMSK<15:5>) 字段仅对于线 10 和 11 是可写的，LTYPE (CHETAG<1>) 字段对于线 0 至 11 固定为“指令”设置。

请注意，为锁定和数据分配的高速缓存线会影响在未命中时的高速缓存选择。但是，它们不会影响使用次序或伪 LRU 值。

4.3 控制寄存器

注： PIC32MX 系列中的一些器件不包含预取高速缓存模块。对于这些器件，所有预取高速缓存寄存器单元均为保留，不应访问它们。

预取高速缓存模块包含以下特殊功能寄存器（Special Function Register，SFR）：

- **CHECON：**预取高速缓存控制寄存器
用于管理预取高速缓存的配置和控制等待状态。
- **CHECONCLR、CHECONSET 和 CHECONINV：**CHECON 的原子级位操作只写寄存器
- **CHEACC：**预取高速缓存访问寄存器
指向 16 条高速缓存线中的一条，以使用 CHETAG、CHEMSK、CHEW0、CHEW1、CHEW2 和 CHEW3 寄存器进行访问。
- **CHEACCCLR、CHEACCSET 和 CHEACCINV：**CHEACC 的原子级位操作只写寄存器
- **CHETAG：**预取高速缓存标记寄存器
包含高速缓存线中存储的信息的地址和类型。
- **CHETAGCLR、CHETAGSET 和 CHETAGINV：**CHETAG 的原子级位操作只写寄存器
- **CHEMSK：**预取高速缓存标记掩码寄存器
提供一种机制来忽略 CHETAG 中的标记位。
- **CHEMSKCLR、CHEMSKSET 和 CHEMSKINV：**CHEMSK 的原子级位操作只写寄存器
- **CHEW0：**高速缓存字 0 寄存器
用于访问预取高速缓存数据阵列
- **CHEW1：**高速缓存字 1 寄存器
用于访问预取高速缓存数据阵列
- **CHEW2：**高速缓存字 2 寄存器
用于访问预取高速缓存数据阵列
- **CHEW3：**高速缓存字 3 寄存器
用于访问预取高速缓存数据阵列
- **CHELRU：**高速缓存 LRU 寄存器
- **CHEHIT：**高速缓存命中统计寄存器
- **CHEMIS：**高速缓存未命中统计寄存器
- **PFABT：**预取高速缓存中止统计寄存器
包含已中止预取高速缓存操作数量的统计寄存器。

下表简要汇总了与预取高速缓存相关的寄存器。该汇总表之后列出了相应的寄存器，并且每个寄存器均附有详细的说明。

表 4-2: 预取高速缓存 SFR 汇总

名称		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
CHECON	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	CHECOH
	15:8	—	—	—	—	—	—	DCSZ<1:0>	
	7:0	—	—	PREFEN<1:0>		—	PFMWS<2:0>		
CHECONCLR	31:0	将 CHECON 中的选定位清零，读取时获得的值未定义							
CHECONSET	31:0	将 CHECON 中的选定位置 1，读取时获得的值未定义							
CHECONINV	31:0	将 CHECON 中的选定位置取反，读取时获得的值未定义							
CHEACC	31:24	CHEWEN	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	—	—	—	—	—	—	—	—
	7:0	—	—	—	—	CHEIDX<3:0>			
CHEACCCLR	31:0	将 CHEACC 中的选定位清零，读取时获得的值未定义							
CHEACCSET	31:0	将 CHEACC 中的选定位置 1，读取时获得的值未定义							
CHEACCINV	31:0	将 CHEACC 中的选定位置取反，读取时获得的值未定义							
CHETAG	31:24	LTAGBOOT	—	—	—	—	—	—	—
	23:16	LTAG<23:16>							
	15:8	LTAG<15:8>							
	7:0	LTAG<7:4>				LVALID	LLOCK	LTYPE	—
CHETAGCLR	31:0	将 CHETAG 中的选定位清零，读取时获得的值未定义							
CHETAGSET	31:0	将 CHETAG 中的选定位置 1，读取时获得的值未定义							
CHETAGINV	31:0	将 CHETAG 中的选定位置取反，读取时获得的值未定义							
CHEMSK	31:24	—	—	—	—	—	—	—	—
	23:16	—	—	—	—	—	—	—	—
	15:8	LMASK<15:8>							
	7:0	LMASK<7:5>			—	—	—	—	—
CHEMSKCLR	31:0	将 CHEMSK 中的选定位清零，读取时获得的值未定义							
CHEMSKSET	31:0	将 CHEMSK 中的选定位置 1，读取时获得的值未定义							
CHEMSKINV	31:0	将 CHEMSK 中的选定位取反，读取时获得的值未定义							
CHEW0	31:24	CHEW0<31:24>							
	23:16	CHEW0<23:16>							
	15:8	CHEW0<15:8>							
	7:0	CHEW0<7:0>							
CHEW1	31:24	CHEW1<31:24>							
	23:16	CHEW1<23:16>							
	15:8	CHEW1<15:8>							
	7:0	CHEW1<7:0>							
CHEW2	31:24	CHEW2<31:24>							
	23:16	CHEW2<23:16>							
	15:8	CHEW2<15:8>							
	7:0	CHEW2<7:0>							
CHEW3	31:24	CHEW3<31:24>							
	23:16	CHEW3<23:16>							
	15:8	CHEW3<15:8>							
	7:0	CHEW3<7:0>							

表 4-2: 预取高速缓存 SFR 汇总 (续)

名称		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
CHELRU	31:24	—	—	—	—	—	—	—	CHELRU<24>
	23:16	CHELRU<23:16>							
	15:8	CHELRU<15:8>							
	7:0	CHELRU<7:0>>							
CHEHIT	31:24	CHEHIT<31:24>							
	23:16	CHEHIT<23:16>							
	15:8	CHEHIT<15:8>							
	7:0	CHENIT<7:0>							
CHEMIS	31:24	CHEMIS<31:24>							
	23:16	CHEMIS<23:16>							
	15:8	CHEMIS<15:8>							
	7:0	CHEMIS<7:0>							
PFABT	31:24	PFABT<31:24>							
	23:16	PFABT<23:16>							
	15:8	PFABT<15:8>							
	7:0	PFABT<7:0>							

寄存器 4-1: CHECON: 高速缓存控制寄存器

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 31							bit 24

r-X	r-X	r-X	r-X	r-X	r-X	r-X	R/W-0
—	—	—	—	—	—	—	CHECOH
bit 23							bit 16

r-X	r-X	r-0	r-0	r-X	r-X	R/W-0	R/W-0
—	—	—	—	—	—	DCSZ<1:0>	
bit 15							bit 8

r-x	r-x	R/W-0	R/W-0	r-x	R/W-1	R/W-1	R/W-1
—	—	PREFEN<1:0>		—	PFMWS<2:0>		
bit 7							bit 0

图注:

R = 可读位 W = 可写位 P = 可编程位 r = 保留位
U = 未实现位 -n = POR 时的值: (0, 1, x = 未知)

- bit 31-17 **保留:** 写入 0; 忽略读操作
- bit 16 **CHECOH:** PFM 编程周期的高速缓存一致性设置位
1 = 使所有数据线和指令线失效
0 = 使所有未锁定的数据线和指令线失效
- bit 15-14 **保留:** 写入 0; 忽略读操作
- bit 13-12 **保留:** 必须写入 0
- bit 11-10 **保留:** 写入 0; 忽略读操作
- bit 9-8 **DCSZ<1:0>:** 数据高速缓存容量 (线数) 位
11 = 使能数据高速缓存, 并且高速缓存的大小为 4 条线
10 = 使能数据高速缓存, 并且高速缓存的大小为 2 条线
01 = 使能数据高速缓存, 并且高速缓存的大小为 1 条线
00 = 禁止数据高速缓存
更改该字段会导致所有数据高速缓存线重新初始化为 “无效” 状态。
- bit 7-6 **保留:** 写入 0; 忽略读操作
- bit 5-4 **PREFEN<1:0>:** 预测性预取高速缓存使能位
11 = 对于可高速缓存和不可高速缓存区域均使能预测性预取高速缓存
10 = 仅对于不可高速缓存区域使能预测性预取高速缓存
01 = 仅对于可高速缓存区域使能预测性预取高速缓存
00 = 禁止预测性预取高速缓存
- bit 3 **保留:** 写入 0; 忽略读操作

寄存器 4-1: CHECON: 高速缓存控制寄存器 (续)

bit 2-0 **PFMWS<2:0>**: 以 SYSLK 等待状态数定义的 PFM 访问时间位

111	= 7 个等待状态
110	= 6 个等待状态
101	= 5 个等待状态
100	= 4 个等待状态
011	= 3 个等待状态
010	= 2 个等待状态
001	= 1 个等待状态
000	= 0 个等待状态

寄存器 4-2: CHECONCLR: CHECON 清零寄存器

写入时会将 CHECON 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHECON 中的选定位清零**
在一个或多个位位置中写入 1 会将 CHECON 寄存器中的相应位清零，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHECONCLR = 0x00010020 时，会将 CHECON 寄存器中的 bit 16 和 bit 5 清零。

寄存器 4-3: CHECONSET: CHECON 置 1 寄存器

写入时会将 CHECON 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHECON 中的选定位置 1**
在一个或多个位位置中写入 1 会将 CHECON 寄存器中的相应位置 1，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHECONSET = 0x00010020 时，会将 CHECON 寄存器中的 bit 16 和 bit 5 置 1。

寄存器 4-4: CHECONINV: CHECON 取反寄存器

写入时会将 CHECON 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHECON 中的选定位取反**
在一个或多个位位置中写入 1 会将 CHECON 寄存器中的相应位取反，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHECONINV = 0x00010020 时，会将 CHECON 寄存器中的 bit 16 和 bit 5 取反。

寄存器 4-5: **CHEACC: 高速缓存访问寄存器**

R/W-0	r-x	r-x	r-x	r-x	r-x	r-x	r-x
CHEWEN	—	—	—	—	—	—	—
bit 31							bit 24
r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 23							bit 16
r-x	r-x	r-x	r-x	r-x	r-x	r-x	r-x
—	—	—	—	—	—	—	—
bit 15							bit 8
r-x	r-x	r-x	r-x	R/W-0	R/W-0	R/W-0	R/W-0
—	—	—	—	CHEIDX<3:0>			
bit 7							bit 0

图注：

R = 可读位

W = 可写位

P = 可编程位

r = 保留位

U = 未实现位

-n = POR 时的值：（0， 1， x = 未知）

- bit 31 **CHEWEN:** 寄存器 CHETAG、CHEMSK、CHEW0、CHEW1、CHEW2 和 CHEW3 高速缓存访问使能位
1 = 通过 CHEIDX 选择的高速缓存线可写
0 = 通过 CHEIDX 选择的高速缓存线不可写
- bit 30-4 **保留:** 写入 0；忽略读操作
- bit 3-0 **CHEIDX<3:0>:** 高速缓存线索引位
该值选择用于读操作或写操作的高速缓存线。

寄存器 4-6: CHEACCCLR: CHEACC 清零寄存器

写入时会将 CHEACC 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHEACC 中的选定位清零**
在一个或多个位位置中写入 1 会将 CHEACC 寄存器中的相应位清零，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHEACCCLR = 0x80000000 时，会将 CHEACC 寄存器中的 bit 31 清零。

寄存器 4-7: CHEACCSET: CHEACC 置 1 寄存器

写入时会将 CHEACC 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHEACC 中的选定位置 1**
在一个或多个位位置中写入 1 会将 CHEACC 寄存器中的相应位置 1，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHEACCSET = 0x80000000 时，会将 CHEACC 寄存器中的 bit 31 置 1。

寄存器 4-8: CHEACCINV: CHEACC 取反寄存器

写入时会将 CHEACC 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHEACC 中的选定位取反**
在一个或多个位位置中写入 1 会将 CHEACC 寄存器中的相应位取反，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHEACCINV = 0x80000000 时，会将 CHEACC 寄存器中的 bit 31 取反。

寄存器 4-9: CHETAG⁽¹⁾: 高速缓存标记 (TAG) 寄存器

R/W-0	r-x	r-x	r-x	r-x	r-x	r-x	r-x
LTAGBOOT	—	—	—	—	—	—	—
bit 31							bit 24
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
LTAG<23:16>							
bit 23							bit 16
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
LTAG<15:8>							
bit 15							bit 8
R/W-x	R/W-x	R/W-x	R/W-x	R/W-0	R/W-0	R/W-1	r-0
LTAG<7:4>				LVALID	LLOCK	LTYPE	—
bit 7							bit 0
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

- bit 31
- LTAGBOOT:** 高速缓存线标记地址引导位
1 = 该线位于存储器的 0x1D000000 (物理) 区域
0 = 该线位于存储器的 0x1FC00000 (物理) 区域
- bit 30-24
- 保留:** 写入 0; 忽略读操作
- bit 23-4
- LTAG<23:4>:** 高速缓存线标记地址位
LTAG 位用于与物理地址 <23:4> 进行比较, 以确定是否命中。由于在内核空间 and 用户空间中 LTAG 位所处的闪存地址范围和位置的原因, 所以 LTAG 闪存地址对于虚拟地址、(系统) 物理地址和闪存物理地址是相同的。
- bit 3
- LVALID:** 高速缓存线有效位
1 = 该线有效, 并与物理地址进行比较来检测是否命中
0 = 该线无效, 不与物理地址进行比较来检测是否命中
- bit 2
- LLOCK:** 高速缓存线锁定位
1 = 该线已锁定, 将不会被替换
0 = 该线未锁定, 可以被替换
- bit 1
- LTYPE:** 高速缓存线类型位
1 = 该线对指令字进行高速缓存
0 = 该线对数据字进行高速缓存
- bit 0
- 保留:** 写入 0; 忽略读操作
- 注 1:
- CHEIDX (CHEACC<3:0>) 指向的高速缓存线的标记和状态。

寄存器 4-10: CHETAGCLR: CHETAG 清零寄存器

写入时会将 CHETAG 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHETAG 中的选定位清零**
在一个或多个位位置中写入 1 会将 CHETAG 寄存器中的相应位清零，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHETAGCLR = 0x0000000C 时，会将 CHETAG 寄存器中的 bit 2 和 bit 3 清零。

寄存器 4-11: CHETAGSET: CHETAG 置 1 寄存器

写入时会将 CHETAG 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHETAG 中的选定位置 1**
在一个或多个位位置中写入 1 会将 CHETAG 寄存器中的相应位置 1，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHETAGSET = 0x00000004 时，会将 CHETAG 寄存器中的 bit 2 置 1。

寄存器 4-12: CHETAGINV: CHETAG 取反寄存器

写入时会将 CHETAG 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHETAG 中的选定位取反**
在一个或多个位位置中写入 1 会将 CHETAG 寄存器中的相应位取反，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例: CHETAGINV = 0x00000010 时，会将 CHETAG 寄存器中的 bit 4 取反。

寄存器 4-13: CHEMSK⁽¹⁾: 高速缓存标记掩码寄存器

r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 31							bit 24
r-X	r-X	r-X	r-X	r-X	r-X	r-X	r-X
—	—	—	—	—	—	—	—
bit 23							bit 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
LMASK<15:8>							
bit 15							bit 8
R/W-0	R/W-0	R/W-0	r-x	r-x	r-x	r-x	r-x
LMASK<7:5>			—	—	—	—	—
bit 7							bit 0
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

- bit 31-16 保留: 写入 0; 忽略读操作
- bit 15-5 LMASK<15:5>: 高速缓存线掩码位
1 = 使能掩码逻辑, 以强制使 LTAG (CHETAG<23:4>) 和物理地址中的相应位位置匹配。
0 = 只有 CHEIDX (CHEACC<3:0>) 的值等于 0x0A 和 0x0B 时才可写。
禁止掩码逻辑。
- bit 4-0 保留: 写入 0; 忽略读操作
- 注 1: CHEIDX (CHEACC<3:07>) 指向的高速缓存线的标记掩码。

寄存器 4-14: CHEMSKCLR: CHEMSK 清零寄存器

写入时会将 CHEMSK 中的选定位清零，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHEMSK 中的选定位清零**
在一个或多个位位置中写入 1 会将 CHEMSK 寄存器中的相应位清零，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例：CHEMSKCLR = 0x00008020 时，会将 CHEMSK 寄存器中的 bit 15 和 bit 5 清零。

寄存器 4-15: CHEMSKSET: CHEMSK 置 1 寄存器

写入时会将 CHEMSK 中的选定位置 1，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHEMSK 中的选定位置 1**
在一个或多个位位置中写入 1 会将 CHEMSK 寄存器中的相应位置 1，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例：CHEMSKSET = 0x00008020 时，会将 CHEMSK 寄存器中的 bit 15 和 bit 5 置 1。

寄存器 4-16: CHEMSKINV: CHEMSK 取反寄存器

写入时会将 CHEMSK 中的选定位取反，读取时获得的值未定义	
bit 31	bit 0

bit 31-0 **将 CHEMSK 中的选定位取反**
在一个或多个位位置中写入 1 会将 CHEMSK 寄存器中的相应位取反，但不会影响未实现位或只读位。
写入 0 不会影响该寄存器。
示例：CHEMSKINV = 0x00008020 时，会将 CHEMSK 寄存器中的 bit 15 和 bit 5 取反。

寄存器 4-17: **CHEW0: 高速缓存字 0**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW0<31:24>							
bit 31				bit 24			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW0<23:16>							
bit 23				bit 16			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW0<15:8>							
bit 15				bit 8			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW0<7:0>							
bit 7				bit 0			
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

bit 31-0 **CHEW0<31:0>**: 通过 CHEACC.CHEIDX 选择高速缓存线的字 0
只有器件不受代码保护时才可读。

寄存器 4-18: CHEW1: 高速缓存字 1

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW1<31:24>							
bit 31				bit 24			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW1<23:16>							
bit 23				bit 16			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW1<15:8>							
bit 15				bit 8			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW1<7:0>							
bit 7				bit 0			
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

bit 31-0 **CHEW1<31:0>**: 通过 CHEACC.CHEIDX 选择高速缓存线的字 1
只有器件不受代码保护时才可读。

寄存器 4-19: CHEW2: 高速缓存字 2

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW2<31:24>							
bit 31				bit 24			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW2<23:16>							
bit 23				bit 16			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW2<15:8>							
bit 15				bit 8			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW2<7:0>							
bit 7				bit 0			
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

bit 31-0 **CHEW2<31:0>**: 通过 CHEACC.CHEIDX 选择高速缓存线的字 2
只有器件不受代码保护时才可读。

寄存器 4-20: **CHEW3⁽¹⁾: 高速缓存字 3**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW3<31:24>							
bit 31				bit 24			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW3<23:16>							
bit 23				bit 16			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW3<15:8>							
bit 15				bit 8			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEW3<7:0>							
bit 7				bit 0			

图注:			
R = 可读位	W = 可写位	P = 可编程位	r = 保留位
U = 未实现位	-n = POR 时的值: (0, 1, x = 未知)		

bit 31-0 **CHEW3<31:0>**: 通过 CHEACC.CHEIDX 选择高速缓存线的字 3
只有器件不受代码保护时才可读。

注 1: 该寄存器是用于访问高速缓存数据阵列的窗口，只有器件不受代码保护时才可读。

寄存器 4-21: CHELRU: 高速缓存 LRU 寄存器

r-X	r-X	r-X	r-X	r-X	r-X	r-X	R-0
—	—	—	—	—	—	—	CHELRU<24>
bit 31							bit 24
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
CHELRU<23-16>							
bit 23							bit 16
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
CHELRU<15-8>							
bit 15							bit 8
R-0	R-0	R-0	R-0	R-0	R-0	R-0	R-0
CHELRU<7-0>							
bit 7							bit 0
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

bit 31-25 保留: 写入 0 ; 忽略读操作

bit 24-0 **CHELRU<24:0>**: 高速缓存最近最少使用 (LRU) 状态编码位
CHELRU 指示高速缓存的伪 LRU 状态。

寄存器 4-22: CHEHIT: 高速缓存命中统计寄存器

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEHIT<31:24>							
bit 31				bit 24			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEHIT<23:16>							
bit 23				bit 16			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEHIT<15:8>							
bit 15				bit 8			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEHIT<7:0>							
bit 7				bit 0			
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

bit 31-0 **CHEHIT<31:0>**: 高速缓存命中计数位

每次处理器从可高速缓存区域发出的取指或装载请求命中预取高速缓存时，它会递增。对于不可高速缓存区域的访问不会修改该值。

寄存器 4-23: **CHEMIS: 高速缓存未命中统计寄存器**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEMIS<31:24>							
bit 31				bit 24			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEMIS<23:16>							
bit 23				bit 16			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEMIS<15:8>							
bit 15				bit 8			
R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
CHEMIS<7:0>							
bit 7				bit 0			
图注：							
R = 可读位		W = 可写位		P = 可编程位		r = 保留位	
U = 未实现位		-n = POR 时的值：（0， 1， x = 未知）					

bit 31-0

CHEMIS<31:0>: 高速缓存未命中计数位
每次处理器从可高速缓存区域发出的取指请求未命中预取高速缓存时，它会递增。对于不可高速缓存区域的访问不会修改该值。

寄存器 4-24: **PFABT: 预取高速缓存中止统计寄存器**

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PFABT<31:24>							
bit 31				bit 24			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PFABT<23:16>							
bit 23				bit 16			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PFABT<15:8>							
bit 15				bit 8			

R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x	R/W-x
PFABT<7:0>							
bit 7				bit 0			

图注:			
R = 可读位	W = 可写位	P = 可编程位	r = 保留位
U = 未实现位	-n = POR 时的值: (0, 1, x = 未知)		

bit 31-0 **PFABT<31:0>:** 预取高速缓存中止计数位
每次自动预取高速缓存由于非顺序取指、装载或存储而被中止操作时，它会递增。

4.4 高速缓存工作方式

高速缓存和预取高速缓存模块实现了完全关联的 16 线高速缓存。每条线都包含 128 位（16 字节）。高速缓存和预取高速缓存模块仅从 PFM 中请求 16 字节对齐的指令数据。如果 CPU 请求地址未与 16 字节边界对齐，则模块会通过丢弃地址位 <3:0> 来对齐地址。配置为仅高速缓存模式时，模块会在未命中时向高速缓存线中装入多条指令。它使用伪 LRU 算法来选择接收新一组指令的高速缓存线。高速缓存控制器使用来自 PFMWS（CHECON<2:0>）的等待状态状态值来确定在检测到未命中时，对于闪存访问必须等待多长时间。在命中时，高速缓存会在 0 个等待状态内返回数据。如果代码 100% 线性，则在仅高速缓存模式下，只有对于高速缓存线中的第一条指令，CPU 取指时才会出现等待状态。对于 32 位线性代码，每隔 4 条指令会出现等待状态。对于 16 位线性代码，每执行 8 条指令才会出现等待状态。

4.5 高速缓存配置

CHECON 寄存器控制 PFM 指令和数据高速缓存的可用配置。通过控制两个参数将高速缓存线分配给特定功能。

DCSZ（CHECON<9:8>）字段控制分配到程序数据高速缓存的线数。表 4-3 列出了对应于 DCSZ（CHECON<9:8>）值的高速缓存线之间的关系。数据高速缓存功能仅用于不进行修改的只读数据，例如常量、参数和表数据等。

表 4-3: 程序数据高速缓存

DCSZ<1:0>	为程序数据分配的高速缓存线
00	无
01	高速缓存线编号 15
10	高速缓存线编号 14 和 15
11	高速缓存线编号 12 至 15

PREFEN（CHECON<5:4>）字段控制预测性预取，该功能让高速缓存控制器可以推测性地预取下一组 16 字节对齐指令。

4.5.1 高速缓存线锁定

高速缓存线中的每条线都可以进行锁定，以保持其内容。如果 LVALID（CHETAG<3>）= 1 且 LLOCK（CHETAG<2>）= 1，则相应的线被锁定。如果 LVALID = 0 且 LLOCK = 1，则高速缓存控制器会发出预装载请求（见第 4.5.3 节“预装载行为”）。锁定高速缓存线可能会降低一般程序流的性能。但是，如果由于一个或两个函数调用而消耗相当大比例的总体处理能力的话，则锁定它们的地址可以提高性能。

虽然可以锁定任意数量的高速缓存线，但是锁定 1 或 4 条线时，高速缓存的工作效率较高。如果锁定 4 条线，请选择编号除以 4 之后商相同的那些高速缓存线。这样一来可以锁定整个 LRU 组，为 LRU 算法带来好处。例如，线 8、9、A 和 B 除以 4 之后的商均为 2。

4.5.2 地址掩码

高速缓存线 10 和 11 允许对 CPU 地址和标记地址设置掩码，强制使相应位匹配。LMASK (CHEMSK<15:5>) 字段可设置用于为 CPU 中的中断向量间距字段提供补充。通过该功能，引导代码可以将某个向量的前 4 条指令锁定在高速缓存中。如果所有向量在前 4 个单元中都包含相同的指令，那么将 LMASK (CHEMSK<15:5>) 设置为与向量间距匹配，将 LTAG (CHETAG<23:4>) 设置为与向量基址匹配，可以使所有向量地址都命中高速缓存。高速缓存可以在 0 个等待状态内进行响应，并且如果使能了预取高速缓存，则会立即发出下一组 4 条指令的取指请求。

LMASK (CHEMSK<15:5>) 的使用仅限于对齐的地址范围。其大小支持的最大范围为 32 KB，最小间距为 32 B。两条线组合使用时，可以提供不同的范围和不同的间距。

如果设置的地址掩码使多条线与某个地址匹配，则会产生未定义的结果。因此，强烈建议先设置掩码，然后再进入可高速缓存代码。

4.5.3 预装载行为

应用程序代码可以指示高速缓存控制器对高速缓存线执行预装载操作，并将它锁定为来自闪存指令或数据。预装载功能使用 CHEACC.CHEIDX 寄存器字段来选择高速缓存线，装载数据送到该高速缓存线中。将 CHEACC.CHEWEN 设置为 1 时，可以使能对 CHETAG 寄存器的写操作。

写入 LVALID (CHETAG<3>) = 0 和 LLOCK (CHETAG<2>) = 1 时，会对高速缓存控制器产生预装载请求。如果可行，控制器会在写操作之后的周期中确认请求，停止所有未完成的闪存访问，并暂停对高速缓存或闪存的任何 CPU 装载操作。

当控制器完成或暂停先前事务时，它会启动闪存读操作，使用 LTAG (CHETAG<23:4>) 中的地址来请求获取指令或数据。经过设定的等待状态数 (通过 PFMWS (CHECON<2:0>) 定义) 之后，控制器会使用从闪存读取的值更新数据阵列。在更新时，它会设置 LVALID (CHETAG<3>) = 1。高速缓存线的 LRU 状态不受影响。

在控制器完成高速缓存更新之后，CPU 可以请求完成操作。如果该请求未命中高速缓存，则控制器会启动闪存读操作，这需要花费一个完整的闪存访问时间。

4.5.4 旁路行为

处理器进行访问时，如果高速缓存的一致性属性指示所访问的地址属于不可高速缓存地址，则模块会旁路高速缓存。在旁路时，对于每条指令，模块都需要访问 PFM，需花费 PFMWS (CHECON<2:0>) 所定义的闪存访问时间。

4.5.5 预测性预取高速缓存行为

如果对可高速缓存地址配置了预测性预取高速缓存，则模块会预测下一条高速缓存线地址，并将它返回到高速缓存的伪 LRU 线中。如果使能，预取高速缓存功能会根据第一次 CPU 取指进行预测。第一条线放入高速缓存中后，模块会将地址递增为下一个 16 字节对齐地址，并开始闪存访问。如果运行的是线性代码（即，无任何跳转），则在上一条高速缓存线中的所有指令执行时或在此之前，闪存会将下一组指令返回到预取高速缓存缓冲区中。

如果在进行预测闪存访问期间的任何时刻，新的 CPU 地址与预测地址不匹配，则闪存访问将更改为访问正确的地址。这种行为不会导致 CPU 访问时间长于不进行预测时的访问时间。

如果访问未命中高速缓存，但命中预取高速缓存缓冲区，则指令将与其地址标记一起被放入伪 LRU 线。伪 LRU 值会被标记为最近最常使用的线，其他线也进行相应更新。如果访问既未命中高速缓存，也未命中预取高速缓存缓冲区，则访问会传递到闪存，并且返回的指令会被放入伪 LRU 线。

如果对不可高速缓存地址配置了预测性预取高速缓存，则控制器将仅使用预取高速缓存缓冲区。对于命中或填充，都不会更新 LRU 高速缓存线，所以高速缓存保持原样。对于线性代码，对不可高速缓存地址使能预测性预取高速缓存时，CPU 可以在 0 个等待状态内完成取指。

当闪存访问的等待状态数设置为 0 时，对不可高速缓存地址使用预测性预取没有任何好处。控制器最多会将预取指令在闪存输出上保持 3 个时钟周期（当 CPU 从缓冲区获取数据时）。对于等待状态数为 0 的闪存访问，这会产生更多功耗，没有任何好处。

预测性数据预取不受支持。但是，在预测性取指操作中途中进行数据访问，会导致高速缓存控制器停止取指操作的闪存访问，并开始从闪存装载数据。预测性预取高速缓存不会重新开始，而是等待另一次取指。此时，它或者由于未命中而填充缓冲区，或者由于命中而开始预取高速缓存。

4.5.6 高速缓存替换策略

对于由于读取未命中而导致的高速缓存线填充，高速缓存控制器使用伪 LRU 替换策略。该策略允许替换 LRU 高速缓存线中末四分之一的任意高速缓存线。使能锁定和数据高速缓存会影响要替换的线，但不会影响伪 LRU 的实际值。

4.6 一致性支持

对闪存进行编程时，无法执行高速缓存操作。在编程序列期间，闪存控制器会停用高速缓存。因此，启动编程序列的用户代码不应位于可高速缓存地址区域。

在编程操作期间，可以通过使全部或部分高速缓存线失效而清空预取高速缓存。

如果 CHECOH (CHECON<16>) 置 1，则在闪存程序存储器写操作期间，每条高速缓存线会被置为失效和解锁。所有高速缓存线的高速缓存标记和掩码也会被清零。

如果 CHECOH 未置 1，则只有未被锁定的线会被强制置为失效。锁定的线保持不变。

4.7 复位的影响

4.7.1 复位时

- 所有高速缓存线均置为无效
- 所有高速缓存线均恢复为指令设置
- 所有高速缓存线均解锁
- LRU 次序是顺序性的，线 0 作为最近最少使用的线
- 所有掩码位均清零
- 所有寄存器均恢复为复位状态

4.7.2 复位之后

- 模块按照 CHECON 寄存器中的值工作
- 高速缓存服从内核的高速缓存一致性属性

4.8 设计技巧

即使在以可实现 0 等待状态的时钟频率运行时，高速缓存功能也可以作为一种很有用的节能技术。访问闪存消耗的功耗会高于访问高速缓存。

4.9 节能模式下的操作

注： 在本手册中，对于特定模块中使用的功耗模式和器件使用的功耗模式进行了区分；例如，比较器的 **Sleep**（休眠）模式和 CPU 的 **SLEEP**（休眠）模式。为了指示所期望功耗模式的类型，模块功耗模式使用大写字母加小写字母（**Sleep**, **Idle**, **Debug**）（休眠、空闲和调试）来表示，器件功耗模式使用全大写字母（**SLEEP**, **IDLE**, **DEBUG**）（休眠、空闲和调试）来表示。

4.9.1 SLEEP（休眠）模式

当器件进入 **SLEEP**（休眠）模式时，预取高速缓存会被禁止，并进入低功耗状态，在该状态下预取高速缓存模块中不会产生时钟。

4.9.2 IDLE（空闲）模式

当器件进入 **IDLE**（空闲）模式时，高速缓存和预取高速缓存时钟源保持工作，但 CPU 会停止执行代码。所有未完成的预取高速缓存操作会先完成，然后模块通过自动时钟门控停止其时钟。

4.9.3 DEBUG（调试）模式

DEBUG（调试）模式不会改变预取高速缓存的行为。在 **DEBUG**（调试）模式执行期间使用软件断点时，必须小心确保高速缓存的连贯性。如果调试器将软件断点指令放入高速缓存中，则在将控制权返还给应用程序之前，应先锁定高速缓存线。当锁定的软件断点被移除时，应将该线解锁并置为无效，从而使得在执行时从 **PFM** 中重新装载原有指令。

4.10 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 PIC32MX 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。当前与预取高速缓存模块相关的应用笔记有：

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 PIC32MX 系列器件的应用笔记和代码示例，请访问 Microchip 网站（www.microchip.com）。

4.11 版本历史

版本 A（2007 年 10 月）

这是本文档的初始版本。

版本 B（2007 年 10 月）

更新了文档（删除了“机密”状态）。

版本 C（2008 年 4 月）

将状态修改为“初稿”；将 U-0 修改为 r-x。

版本 D（2008 年 6 月）

将保留位从“保持为”更改为“写入”。

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、KEELOQ、KEELOQ 徽标、MPLAB、PIC、PICmicro、PICSTART、PIC³² 徽标、rfPIC 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、MXDEV、MXLAB、SEEVAL 和 The Embedded Control Solutions Company 均为 Microchip Technology Inc. 在美国的注册商标。

Analog-for-the-Digital Age、Application Maestro、CodeGuard、dsPICDEM、dsPICDEM.net、dsPICworks、dsSPEAK、ECAN、ECONOMONITOR、FanSense、HI-TIDE、In-Circuit Serial Programming、ICSP、Mindi、MiWi、MPASM、MPLAB Certified 徽标、MPLIB、MPLINK、mTouch、Octopus、Omniscient Code Generation、PICC、PICC-18、PICDEM、PICDEM.net、PICKit、PICKtail、REAL ICE、rfLAB、Select Mode、Total Endurance、TSHARC、UniWinDriver、WiperLock 和 ZENA 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。

在此提及的所有其他商标均为各持有公司所有。

© 2010, Microchip Technology Inc. 版权所有。

ISBN: 978-1-60932-077-5

QUALITY MANAGEMENT SYSTEM
CERTIFIED BY DNV
== ISO/TS 16949:2002 ==

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-16949:2002 认证。公司在 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方面的质量体系流程均符合 ISO/TS-16949:2002。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。

全球销售及服务中心

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://support.microchip.com>
网址: www.microchip.com

亚特兰大 Atlanta

Duluth, GA
Tel: 678-957-9614
Fax: 678-957-1455

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: 216-447-0464
Fax: 216-447-0643

达拉斯 Dallas

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

底特律 Detroit

Farmington Hills, MI
Tel: 1-248-538-2250
Fax: 1-248-538-2260

科科莫 Kokomo

Kokomo, IN
Tel: 1-765-864-8360
Fax: 1-765-864-8387

洛杉矶 Los Angeles

Mission Viejo, CA
Tel: 1-949-462-9523
Fax: 1-949-462-9608

圣克拉拉 Santa Clara

Santa Clara, CA
Tel: 408-961-6444
Fax: 408-961-6445

加拿大多伦多 Toronto

Mississauga, Ontario,
Canada
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: 86-10-8528-2100
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 - 香港特别行政区

Tel: 852-2401-1200
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: 86-755-8203-2660
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

台湾地区 - 高雄

Tel: 886-7-536-4818
Fax: 886-7-536-4803

台湾地区 - 台北

Tel: 886-2-2500-6610
Fax: 886-2-2508-0102

亚太地区

台湾地区 - 新竹

Tel: 886-3-6578-300
Fax: 886-3-6578-370

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: 91-80-3090-4080

印度 India - New Delhi

Tel: 91-11-4160-8631
Fax: 91-11-4160-8632

印度 India - Pune

Tel: 91-20-2566-1512
Fax: 91-20-2566-1513

日本 Japan - Yokohama

Tel: 81-45-471- 6166
Fax: 81-45-471-6122

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 - Munich

Tel: 49-89-627-144-0
Fax: 49-89-627-144-44

意大利 Italy - Milan

Tel: 39-0331-742611
Fax: 39-0331-466781

荷兰 Netherlands - Druenen

Tel: 31-416-690399
Fax: 31-416-690340

西班牙 Spain - Madrid

Tel: 34-91-708-08-90
Fax: 34-91-708-08-91

英国 UK - Wokingham

Tel: 44-118-921-5869
Fax: 44-118-921-5820

12/30/09