

第 3 章 TMS320C621x /C671x /C64x

二级内部存储器

TMS320C621x/C671x/C64x 为内部程序总线和数据总线提供一个两级存储器结构,这个两级存储器是由程序 Cache 和数据 Cache 组成的。

3.1 概 述

TMS320C621x/C671x/C64x 有一个用于程序和数据的两级存储器结构,由程序 Cache 和数据 Cache 组成。一级程序 Cache 记为 L1P,而一级数据 Cache 记为 L1D。程序和数据存储器共享第二级存储器,记为 L2。L2 配置为局部 Cache 和局部 SRAM。

图 3-1 和图 3-2 分别为 C621x/C671x 和 C64x 的框图。表 3-1 概述了这些器件的内部构造。图 3-3 说明了在 TMS320C6000 的 CPU、内存和增强 DMA 之间的总线连接。

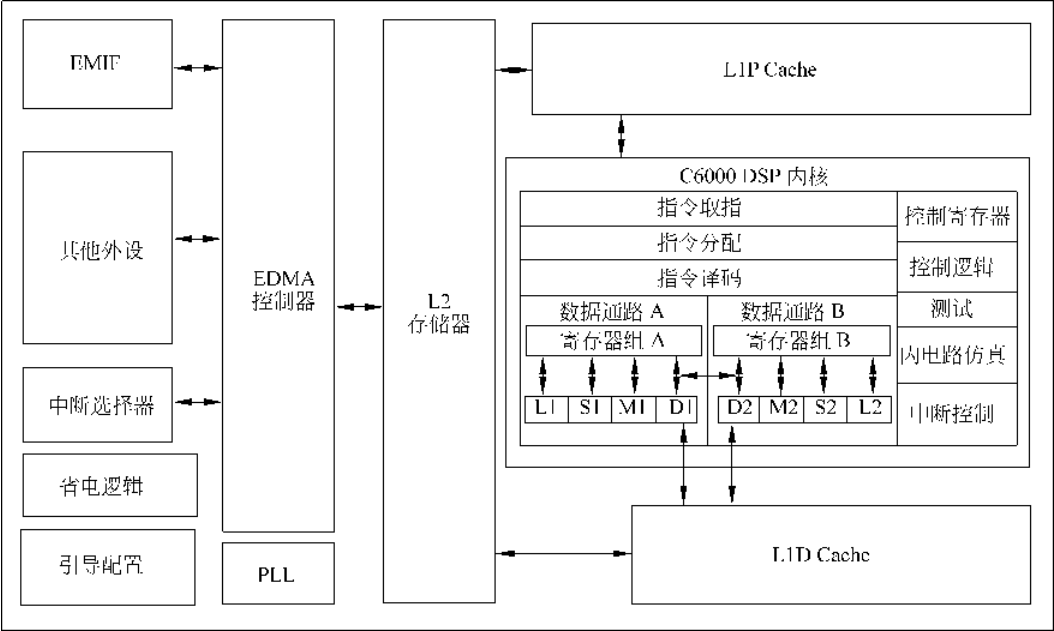


图 3-1 TMS320C621x/C671x 框图

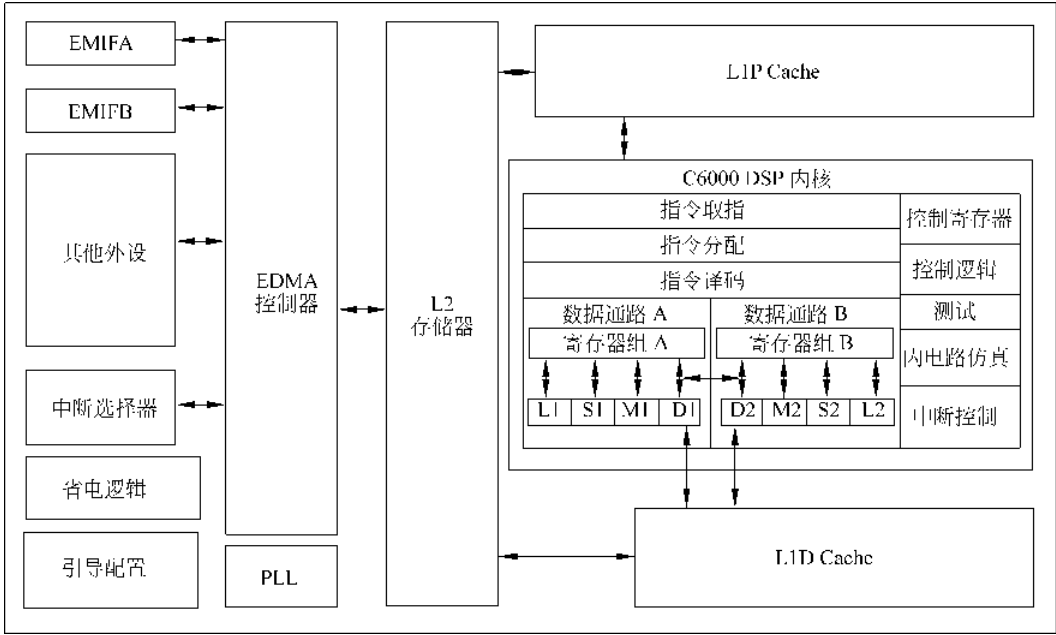


图 3-2 TMS320C64x 框图

表 3-1 TMS320C621x/C671x/C64x 内存

	TMS320C621x/C671x	TMS320C64x
内存结构	两级	两级
L1P 的大小	4KB	16KB
L1P 的结构	直接映射	直接映射
L1P 的 CPU 访问时间	单周期	单周期
L1P 行的大小	64B	32B
L1P 的分组	1×256 位的组	8×32 位的组
L1P 读缺失的分配	L1P 中分配一行	L1P 中分配一行
L1P 读命中的分配	从 L1P 中读数据	从 L1P 中读数据
L1P 写缺失的分配	不支持 L1P 写	不支持 L1P 写
L1P 写命中的分配	不支持 L1P 写	不支持 L1P 写
L1P→L2 请求的大小	2 次获取/L1P 行	1 次获取/L1P 行
L1P 协议	读分配	读分配；缺失流水
L1P 存储器	单周期 RAM	单周期 RAM
L1P→L2 请求的 CPU 阻塞	5 周期	8 周期
L1D 的大小	4KB	16KB
L1D 的结构	双路联合集	双路联合集
L1D 的 CPU 访问时间	单周期	单周期

续表

	TMS320C621x/C671x	TMS320C64x
L1D 行的大小	32B	64B
L1D 的替代策略	双路最近最少使用(LRU)策略	双路最近最少使用(LRU)策略
L1D 的分组	64 位宽的双口 RAM	8×32 位的组
L1D 读缺失的分配	L1D 中分配一行	L1D 中分配一行
L1D 读命中的分配	从 L1D 中读数据	从 L1D 中读数据
L1D 写缺失的分配	L1D 中不分配,数据送到 L2	L1D 中不分配,数据送到 L2
L1D 写命中的分配	数据写入到命中的 L1D 位置	数据写入到命中的 L1D 位置
L1D 协议	读分配	读分配; 缺失流水
L1P→L2 请求的大小	2 次获取/L1D 行	2 次获取/L1D 行
L1D→L2 请求的 CPU 阻塞	4 周期	6 周期/SRAM 8 周期/L2 Cache 命中
L2 的大小	64KB	1024KB
L2 高速缓存的大小	0/16/32/48/64KB	0/32/64/128/256KB
L2 SRAM 的大小	64/48/32/16/0 KB	1024/992/960/896/768 KB
L2 的结构	1/2/3/4 路联合集	4 路联合集 Cache
L2 行的大小	128B	128B
L2 的替代策略	1/2/3/4 路最近最少使用(LRU)策略	4 路最近最少使用(LRU)策略
L2 的分组	4×64 位的组	8×64 位的组
L2-L1P 协议	一致性侦听使无效	一致性侦听使无效
L2-L1D 协议	一致性侦听	一致性侦听
L2 协议	读分配和写分配	读分配和写分配
L2 读缺失的分配	通过 EDMA 读数据,存入 L2	通过 EDMA 读数据,存入 L2
L2 读命中的分配	从 L2 读数据	从 L2 读数据
L2 写缺失的分配	数据通过 EDMA 读入到符合 LRU 策略的 L2 行,且被新数据修改	数据通过 EDMA 读入到符合 LRU 策略的 L2 行,且被新数据修改
L2 写命中的分配	数据写入到命中的 L2 位置	数据写入到命中的 L2 位置
L2→L1P 读通道宽度	256 位	256 位
L2→L1D 读通道宽度	128 位	256 位
L1D→L2 写通道宽度	32 位	64 位
L1D→L2 牺牲通道宽度	128 位	256 位
L2→EDMA 读通道宽度	64 位	64 位
L2→EDMA 写通道宽度	64 位	64 位

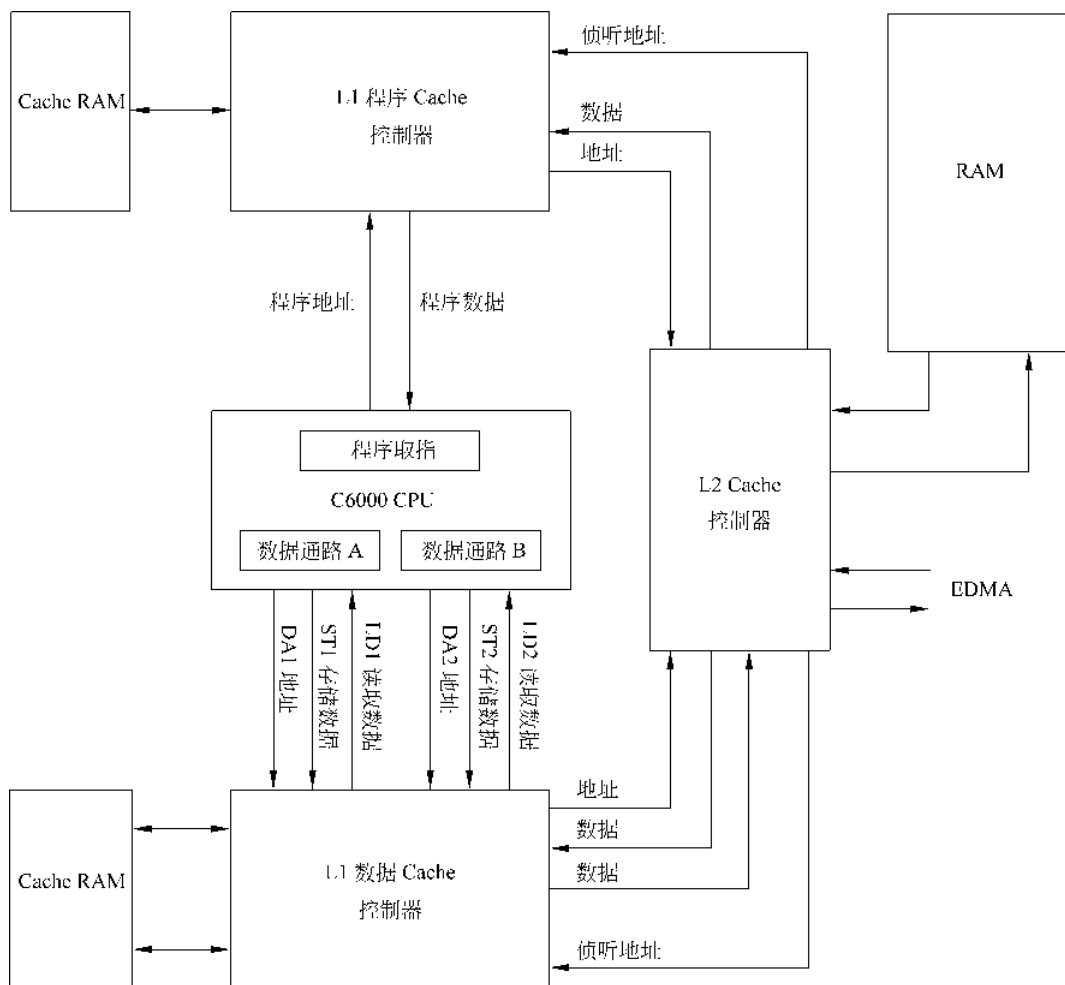


图 3-3 TMS320C6000 二级内存框图

3.2 TMS320C621x/C671x/C64x 高速缓存定义

以下是涉及 TMS320C621x/C671x/C64x 二级 Cache 操作的一些术语：

- **分配(allocate)**。是在 Cache 中寻找一个位置来存储新的、未缓存的数据的过程。
- **联合性(associativity)**。是在每一集(set)中行帧的数目。
- **清除(clean)**。当 Cache 的一行保持的数据有效且未被修改,则该行是要被清除的。
- **一致性**。当系统使用两级存储体系时,有可能同时将相同的地址分别存储到两个级别的存储器中。如果低级别存储器中的数据被更改,而高一级别存储器中的数据并没有随之更新,则两级存储器中的数据是不一致的。如果在两级存储器中驻

留相同数据,就说两级存储器是一致的。

- 直接映射 Cache。直接映射 Cache 包含能缓存特定的高级存储器的唯一 Cache 行。这并不意味着该 Cache 像它所缓存的存储器一样包含很多行,而是指一个存储器地址只能由专门的行来缓存。例如,一个存储器地址被缓存在直接映射 Cache 位置 A,清除 Cache 之后再缓存,它还是只能被缓存在位置 A。
- DMA。直接存储器访问,这是数据块从一个存储器空间到另外一个存储空间的转移。它在 EDMA 引擎中执行。
- 重写。若 Cache 的一行保存的数据有效且被 CPU 修改过,但是这种修改没有在下—更高级内存中更新,就是该 Cache 行重写。
- 逐出。当牺牲行包含重写数据时,数据必须写出到下一级存储器来维持一致性。这个向下一级存储器重写数据的过程就是逐出,这是由匹配行的寻址缺失和被送入 Cache 造成的。
- 执行包。执行包由单周期内并行执行的指令块组成。
- 获取包。获取包由 8 个指令组成,这些指令在单周期中获取,但部分或全部指令并不需要在这一个周期内执行。
- 高级存储器。与系统存储器更接近的存储器。按此定义 L2 是高级存储器,它与处理器外部的物理存储器更接近。
- 命中。请求存储器地址的数据刚好在 Cache 中时发生 Cache 命中。Cache 命中可使阻塞时间最小化,因为从 Cache 中获取数据比从源存储器中快得多。
- LRU(LEAST RECENTLY USED)分配。对于联合集 Cache,LRU 分配是指选择哪一行帧来分配空间的方法。当寻址映射集中的全部行包含有效数据时,就分配哪个最近最少被读或被写的行来存储新的数据;或者说最近最多使用的行被保留。
- 行。Cache 行是 Cache 缺失时从高一级存储器获取的数据总和。Cache 行可以申请比那些导致 Cache 缺失的数据长度更长。例如,读取字节指令可能导致 L1D 缺失,但是 Cache 将从 L2 获取一整行,而不仅仅是申请的字节。
- 行帧。行帧是 Cache 中保持缓存数据(一行)、联合寻址标记和行状态数据的一个位置。状态数据指明该行有效或重写。
- 穿越读取。当 CPU 请求在一级和二级均缺失时,从外存获取数据并同时存储到一级和二级 Cache 中。一个 Cache 存储数据,并同时将其发送到低级 Cache,就是穿越读取 Cache。与那种先存储数据然后再将其送入低级 Cache 的 Cache 相比,使用穿越读取 Cache 可以减少阻塞时间。
- 远距离访问。由 CPU 发起的、针对不可缓存的存储器空间的访问是远距离访问,可以认为是直接针对外存的访问。当访问映射为控制寄存器的存储器时也使用远距离访问。
- 低级存储器。就是最接近 CPU 而离物理存储器系统最远的存储器。按这样的定义 L1D 和 L1P 是低级存储器。
- 存储器排序。这是决定将要在存储器中处理数据的顺序的过程。有两种类型,强

制型和随意型。在强制型中,全部读操作和写操作严格按照编程顺序。当访问并行发生时,写总是发生在读前。如果有多个读或多个写,从 A 数据通路来的数据要先读还是先写,取决于执行包是由并行读取指令还是存储指令构成。强制型存储器排序与标准软件数据流直接对应。

- 缺失。所请求存储器地址的数据不在 Cache 中时将发生 Cache 缺失。当数据是从高一级 Cache 获取时,缺失会导致 CPU 阻塞。
- 缺失流水线。发生单个缺失时,会发生很多事件以获取缺失行进入低级存储器。这包括获取数据、写标记存储器和回写逐出行。发生多缺失时,这一总开销被所有缺失分摊,只有第一个缺失的开销不变,接下来的每个缺失开销要小得多。
- 读分配。读分配 Cache 仅对读缺失在 Cache 中分配空间。写缺失并不会导致空间分配的发生,而是数据将被传递到高一级 Cache 中。
- 集(set)。一个集是指其中能够驻留行的很多行帧的集合。直接映射 Cache 的每个集包含一行, n 路联合集 Cache 的每个集包含 n 行。
- 联合集 Cache。包含很多可以缓存高级存储器地址的行。例如,存储器地址 X 能在联合集 Cache 中的 A、B 或 C 行缓存。地址 X 被获取并存储于 A 行,然后它从 Cache 中被清除。当重新获取 X 时,它可以被缓存在 A、B 或 C 行中,这取决于哪一行被分配。A、B 和 C 行在 Cache 中组成一集。
- 侦听。侦听是一种高级存储器查询低级存储器以确定两者是否含有同一地址的数据的方法。在两级存储器系统中,当数据从高级存储器中移出时,必须侦听低级存储器来确定逐出的地址是否与低级存储器中的同一地址不一致。
- 标记。标记是包含行地址高位的一个存储单元。它们被存储在标记存储器中。查询标记存储器能确定一定范围的地址现在是否在 Cache 中。这会产生一个命中或缺失。
- 逆操作。当 CPU 对映射到同一 Cache 行的多个地址做交替访问时,每个访问都能使前一个访问退出 Cache。对第一个地址的下一次访问将从高一级存储器中获取数据,因为数据已从 Cache 中被逐出。这个读取数据进入 Cache、覆盖数据、从下一级存储器再次读取数据的过程称为 Cache 逆操作。
- 有效。当 Cache 的一行保持有从下一级存储器获取的数据时,就说该行是有效的。
- 牺牲。当 Cache 中分配了空间,但地址映射集中所有的行都包含有效数据,那么最近最少使用的行上的数据就会被新的数据所覆盖。被覆盖的行就是牺牲行。
- 回写(writeback)。回写 Cache 在写命中时仅更改它自己的数据。它不通知高一级存储器,因而 Cache 和高级存储器是不一致的,但是 Cache 中将保持正确的数据。
- 写分配。写分配 Cache 为读缺失或写缺失分配空间。对写缺失,在 Cache 中分配空间且写入数据,数据不送到下一级 Cache 中。
- 写合并。如果写指令具有相同的双字地址,它们就能合并到单个写操作中。例如,两个字存储到同一双字地址就可以合并为一个写。

3.3 TMS320C621x/C671x 二级存储器

3.3.1 一级程序 Cache(L1P)

TMS320C621x/C671x 的 L1P 是一个含有 64 个集的 4KB 直接映射缓存,行大小为 64B。一个 32 位的 CPU 程序地址被分成 3 块以确定用来驻留数据的 L1P 物理地址。低 6 位指出程序获取包的第一个字的字节偏移量。由于 CPU 每次获取指令要求一个获取包,所以偏移量的低 5 位被忽略,仅用剩余的最高位确定行的哪一半送到 CPU。程序地址中紧接着的下 6 位确定数据驻留在哪个集中。L1P 是直接映射缓存,因而每个地址的数据仅能驻留在 64 个集的一个当中。程序地址的高 20 位是指明行中当前驻留着什么数据的标记。图 3-4 表示了被分为标记、集和偏移量 3 个字段的 32 位地址。

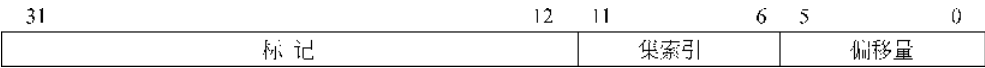


图 3-4 TMS320C621x/C671x L1P 地址分配

例如,地址 0x801ef183 划分为: 偏移量为 0x3,集索引为 0x6,标记为 0x801ef。

L1P 操作是由 CPU 控制状态寄存器(CSR)、L1P 冲洗基地址寄存器(L1PFBAR)、L1P 冲洗字计数寄存器(L1PFWC)以及缓存配置寄存器(CCFG)控制的。这些寄存器操作的详细描述参见 3.5 节。

L1P 命中在 1 个周期内完成,不阻塞 CPU。L1P 缺失而 L2 命中要使 CPU 阻塞 5 个周期。若 L1P 缺失,L2 也缺失,则 CPU 将一直阻塞,直到 L2 从外部存储器中得到数据并将其传输到 L1P 后,L1P 再把数据返回到 CPU。

3.3.2 一级数据 Cache(L1D)

TMS320C621x/C671x 的 L1D 是含有 64 个集的 4KB 的双路联合集缓存,行大小是 32B。来自 CPU 的物理地址被分成 4 个区以选择适当的 Cache 集来分配数据,并从集中选择正确的字。地址的低两位是地址的字偏移量;接下来的 3 位在包含请求数据的集中选择字;再下来 6 位用来识别适当的集来寻找请求数据;剩下的 21 位是地址的标记值。图 3-5 说明了一个 32 位地址如何被分成标记、集、字和偏移量 4 个字段。

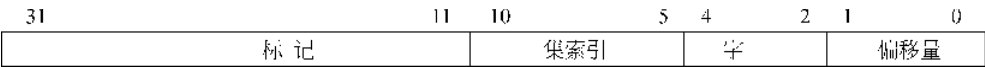


图 3-5 TMS320C621x/C671x L1D 地址分配

L1D 操作是由 CPU 控制状态寄存器(CSR)、L1D 冲洗基地址寄存器(L1DFBAR)、L1D 冲洗字计数寄存器(L1DFWC)和缓存配置寄存器(CCFG)控制的。这些寄存器操作的详细描述参见 3.6 节。

L1D 命中在 1 个周期内完成,不阻塞 CPU。L1D 缺失而 L2 命中将使 CPU 阻塞 4 个周期。若 L1D 缺失,L2 也缺失,则 CPU 将一直阻塞,直到 L2 从外部存储器中重新得到数据并将其传输到 L1D 后,L1D 再把数据返回到 CPU。

L1D 和 L2 Cache 之间有一个写缓冲器。这个缓冲器能最多保持 4 个写缺失的字。只要这个缓冲器没有满,那么就能在不阻塞 CPU 的情况下接受新的 L1D 写缺失或牺牲操作。一个 L1D 写缺失,或一个命中写缓冲器入口的侦听,将阻塞 CPU 直到缓冲器重新被冲洗。

C621x/C671x 的 L1D 和 L2 之间的写路径是 32 位宽的。这个路径能在单周期内传输两个 16 位宽的写缺失到 L2 的不同组。如果两个写缺失要传到同一组,则在两个周期内串行传输。一个 L1D 牺牲写操作需要两周期,因为在一个周期内只有一半的 L1D 线能沿牺牲路径向下传输。

3.3.3 二级 Cache(L2)

L2 有 4 种操作模式,取决于 CCFG 寄存器的状态。图 3-6 给出了 TMS320C621x/C671x 在各种 L2 模式下,L2 SRAM 的映射存储器空间和 Cache 划分。它也示出了 L2 的存储器配置是如何影响 Cache 和 SRAM 的比例的。

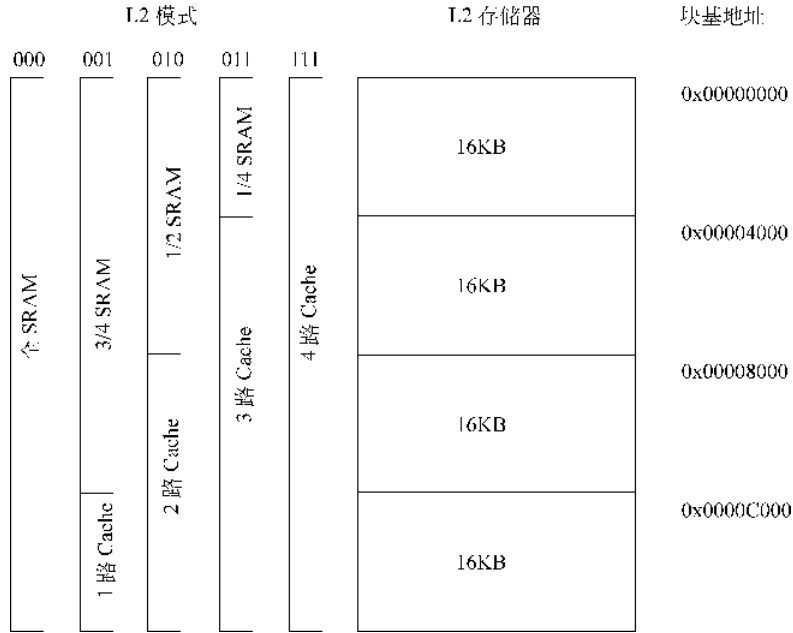


图 3-6 TMS320C621x/C671x L2 存储器配置

TMS320C621x/C671x 的 L2 是一个能够以几种模式操作的 64KB 存储器。L2 操作由缓存配置寄存器 (CCFG)、L2 冲洗基地址寄存器 (L2FBAR)、L2 冲洗字计数寄存器 (L2FWC)、L2 清除基地址寄存器 (L2CBAR)、L2 清除字计数寄存器 (L2CWC)、L2 冲洗寄存器 (L2FLUSH)、L2 清除寄存器 (L2CLEAN) 控制。这些寄存器和 L2 操作的详细描述参见 3.7 节。图 3-7、图 3-8 和图 3-9 分别示出了 32 位地址如何被划分为标记、集索引和偏移量 3 个字段。



图 3-7 TMS320C621x/C671x 的 L2 地址分配：64KB 或 48KB 缓存 (L2MODE=011b 或 111b)



图 3-8 TMS320C621x/C671x 的 L2 地址分配：32KB 缓存 (L2MODE=010b)

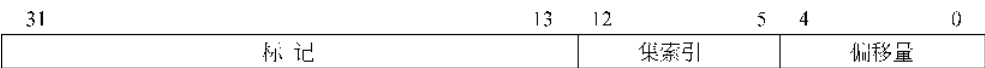


图 3-9 TMS320C621x/C671x 的 L2 地址分配：16KB 缓存 (L2MODE=001b)

3.3.4 TMS320C621x/C671x 数据对齐

下列数据应用时有严格的对齐限制：

- 双字：对于 C67x，双字按照偶数个 8 字节 (双字) 边界对齐，并且总是从最低 3 位都是 0 的字节地址开始，双字只用于 LDDW 指令触发的读取。C67x 的存储操作和 C62x 的读取、存储操作不使用双字。
- 字：字按照偶数个 4 字节 (字) 边界对齐，并且总是从最低两位都是 0 的字节地址开始。字访问需要两个相邻的 16 位宽的组。
- 半字：半字按照偶数个两字节 (半字) 边界对齐，并且总是从最低 1 位是 0 的字节地址开始。半字访问需要完整的 16 位宽的组。
- 字节：字节访问没有对齐限制。

3.3.5 控制寄存器

除了上述缓存控制寄存器，存储器属性寄存器 (MAR) 也控制 TMS320C621x/C671x 的 Cache 操作。图 3-10 示出了 TMS320C621x 和 TMS320C671x 存储器属性寄存器的格式。MARs 字段及其对 Cache 影响的详细描述参见 3.7.4 小节。

表 3-2 列出了影响 TMS320C621x/C671x 控制寄存器和它们在存储器映射中的地址。

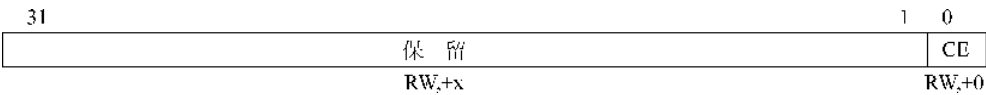


图 3-10 TMS320C621x/C671x L2 内存属性寄存器(MAR0~MAR15)

表 3-2 TMS320C621x/C671x 内部存储器控制寄存器地址

寄存器地址	寄存器符号	寄存器描述
01840000h	CCFG	缓存配置寄存器
01844000h	L2FBAR	L2 冲洗基地址寄存器
01844004h	L2FWC	L2 冲洗字计数寄存器
01844010h	L2CBAR	L2 清除基地址寄存器
01844014h	L2CWC	L2 清除字计数寄存器
01844020h	L1PFBAR	L1P 冲洗基地址寄存器
01844024h	L1PFWC	L1P 冲洗字计数寄存器
01844030h	L1DFBAR	L1D 冲洗基地址寄存器
01844034h	L1DFWC	L1D 冲洗字计数寄存器
01845000h	L2FLUSH	L2 冲洗寄存器
01845004h	L2CLEAN	L2 清除寄存器
01848200h	MAR0	控制 CE0 范围为 80000000h~80FFFFFFh
01848204h	MAR1	控制 CE0 范围为 81000000h~81FFFFFFh
01848208h	MAR2	控制 CE0 范围为 82000000h~82FFFFFFh
0184820Ch	MAR3	控制 CE0 范围为 83000000h~83FFFFFFh
01848240h	MAR4	控制 CE1 范围为 90000000h~90FFFFFFh
01848244h	MAR5	控制 CE1 范围为 91000000h~91FFFFFFh
01848248h	MAR6	控制 CE1 范围为 92000000h~92FFFFFFh
0184824Ch	MAR7	控制 CE1 范围为 93000000h~93FFFFFFh
01848280h	MAR8	控制 CE2 范围为 A0000000h~A0FFFFFFh
01848284h	MAR9	控制 CE2 范围为 A1000000h~A1FFFFFFh
01848288h	MAR10	控制 CE2 范围为 A2000000h~A2FFFFFFh
0184828Ch	MAR11	控制 CE2 范围为 A3000000h~A3FFFFFFh
018482C0h	MAR12	控制 CE3 范围为 B0000000h~B0FFFFFFh
018482C4h	MAR13	控制 CE3 范围为 B1000000h~B1FFFFFFh
018482C8h	MAR14	控制 CE3 范围为 B2000000h~B2FFFFFFh
018482CCh	MAR15	控制 CE3 范围为 B3000000h~B3FFFFFFh

3.4 TMS320C64x 二级存储器

3.4.1 一级程序 Cache(L1P)

C64x 的 L1P 是具有 32B 行大小和 512 个集的 16KB 直接映射缓存。32 位的 CPU 程序地址被分成 3 块来确定保存数据的 L1P 的物理位置。低 5 位指出程序获取包的字节偏移

量。紧接着后面的 9 位用来指出在哪个集中保存数据。L1P 是直接映射缓存,每一个地址的数据只能唯一地被保存在 512 集的一个当中。剩下的高 18 位则是一个唯一标记,它标明了当前行上是什么数据。图 3-11 说明被分成标记、集和偏移量的 32 位地址。



图 3-11 TMS320C64x L1P 地址分配

L1P 操作受 CPU 控制状态寄存器(CSR)、L1P 冲洗基地址寄存器(L1PFBAR)、L1P 冲洗字计数器寄存器(L1PFWC)和缓冲配置寄存器(CCFG)控制。这些寄存器操作的详细描述参见 3.5 节。

L1P 命中在 1 个周期中完成,不阻塞 CPU。1 个 L1P 缺失、L2 命中将使 CPU 阻塞 0~7 个周期,具体周期数取决于缺失发生时执行包的并行性和流水线的执行节拍。

流水线的性能是恒定的,与执行包的对齐无关。对于 C64x 结构,执行包能超过获取包的边界。例如,含有 5 条、5 条和 6 条指令的 3 个执行包只占两个获取包。

若 L1P 缺失、L2 也缺失,CPU 将阻塞,直到 L2 从外存重新得到数据并且将数据传输到 L1P 后,L1P 再将数据返回到 CPU。具体延迟将取决于保存外部程序的外存类型。

3.4.2 一级数据 Cache(L1D)

TMS320C64x 的 L1D 是 16KB 的 Cache,具有从 L1D 到 L2 存储器的 64 位宽的写总线。它是含有 128 个集的双路联合集 Cache,行大小为 64B。来自 CPU 的物理地址被分成 4 个区域来选择适当的 Cache 集来分配数据,并从该集中选择正确的字。较低的两位是地址字偏移量。以下 4 位在集中选择包含请求数据的字。后 7 位标明查找请求数据的适当的集。剩下的 20 位是地址的标记值。图 3-12 说明了一个 32 位地址被分成标记、集、字和偏移量 4 个字段。

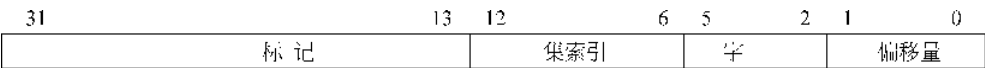


图 3-12 TMS320C64x L1D 地址分配

L1D 操作由 CPU 控制状态寄存器(CSR)、L1D 冲洗基地址寄存器(L1DFBAR)、L1D 冲洗字计数器寄存器(L1DFWC)和缓冲配置寄存器(CCFG)控制。这些寄存器操作的详细描述参见 3.6 节。

L1D 命中在 1 个周期内完成,不阻塞 CPU。L1D 缺失,L2 命中将使 CPU 阻塞 2~8 个周期。两个周期的损失是大量连续缺失时一种稳定状态的平均损失(见 3.4.3 小节)。单个 L1D 缺失的最糟糕情况是 L2 被配置为 Cache 并有一个 L2 Cache 命中,这可导致阻塞 8 个周期。L1D 缺失,L2 也缺失将使 CPU 一直阻塞,直到 L2 从外存中重新得到数据并将其传送到 L1D 后,L1D 再将数据送回 CPU。外部缺失的损失代价取决于用来保存数据的外存的类型和宽度。