

TMS320x2833x,2823xDSC 接口扩展

(XINTF)

参考指南

文献编号:SPRU949D

September2007–Revised January 2010

前言.....	5
1 功能描述.....	7
1.1 与 TMS320x281x XINTF 的区别.....	7
1.2 与 TMS320x2834x XINTF 的区别.....	8
1.3 访问 XINTF 空间.....	8
1.4 写操作紧跟读操作的流水线保护.....	9
2 XINTF 功能配置.....	10
2.1 外部接口(XINTF)配置寄存器及时序寄存器的设置.....	10
2.2XINTF 时钟.....	11
2.3 写缓存.....	12
2.4 每个 XINTF 区域的访问建立/激活/跟踪状态的定时.....	12
2.5 对每个区采样 XREADY.....	13
2.6 储体转接.....	13
2.7 区数据总线宽度.....	14
3 外部 DMA 支持(XHOLD,XHOLDA).....	16
4 建立, 激活及跟踪状态的配置.....	17
4.1 忽略 USERREADY 信号.....	17
4.2 同步模式(USERREADY=1,READYMODE=0).....	18
4.3 异步模式(USERREADY=1,READYMODE=1).....	18
5 配置 XBANK 周期.....	22
6 XINTF 寄存器.....	23
6.1 XINTF 定时寄存器.....	23
6.2 XINTF 配置寄存器.....	27
6.3 XBANK 寄存器.....	29
6.4 XREVISION 寄存器.....	29
6.5 XRESET 寄存器.....	29
7 信号说明.....	31
8 波形.....	32
附录 A 修订历史记录.....	37

图目录

1 外部接口框图.....	9
2 访问流程图.....	11

3	XTIMCLK 和 SYSCLKOUT 之间的关系.....	12
4	XINTF 典型 16 位数据总线连接.....	14
5	XINTF 典型 32 位数据总线连接.....	15
6	XTIMING0/6/7 寄存器分布.....	23
7	XINTF 配置寄存器(XINTCNF2).....	27
8	XBANK 寄存器.....	29
9	XREVISION 寄存器.....	29
10	XRESET 寄存器.....	30
11	XTIMCLK 和 XCLKOUT 模式波形.....	33
12	通用读周期(XTIMCLK=SYSCLKOUT 模式).....	34
13	通用读周期(XTIMCLK=½SYSCLKOUT 模式).....	35
14	通用写周期(XTIMCLK=SYSCLKOUT 模式).....	36

表目录

1	16 位模式性能.....	15
2	32 位模式性能.....	15
3	依据 XTIMCLK 周期的持续脉冲.....	17
4	Lead/Tail 数值与 XTIMCLK/X2TIMING 方式之间的关系.....	20
5	Active 数值与 XTIMCLK/X2TIMING 方式之间的关系.....	21
6	有效 XBANK 配置.....	22
7	XINTF 配置和控制寄存器映射.....	23
8	XTIMING0/6/7 寄存器字段说明.....	23
9	XINTF 配置寄存器字段说明.....	27
10	XBANK 寄存器字段说明.....	29
11	XREVISION 寄存器字段说明.....	29
12	XRESET 寄存器字段说明.....	30
13	XINTF 信号说明.....	31
14	修订.....	37

前言

本文档介绍了在 F2833x 或 F2823x 设备中使用的外部接口（XINTF）。该 XINTF 是一个非复用的异步总线。

符号规定

本文档使用下列约定。

- 十六进制数字显示后缀 h 或一个前缀 0x。例如，下面的数字是十六进制的 40（十进制 64）：40H 或 0x40。
- 在本文件中的寄存器的图形显示和表格说明。
 - 每个寄存器图形显示成若干区域的矩形，它代表寄存器中的每一个字段。每个字段都标有其位的名称，它的开始和结束位数字在上，其读/写属性在下面。图例含属性所使用的符号解释。
 - 寄存器图形中保留位的指定一个位，是为未来设备的扩展使用。

德州仪器相关文档

下面的文档描述的相关设备和相关的支持工具。这些文件的副本在互联网 www.ti.com 上公布。提示：在 www.ti.com 网站的搜索框中输入文献编号。

数据手册和勘误表—

SPRS439 - TMS320F28335, TMS320F28334, TMS320F28332, TMS320F28235, TMS320F28234, TMS320F28232 数字信号控制器 (DSC) 数据手册包含了引脚, 信号说明, 以及 F2833x/2823x 设备的电子和定时规范。

SPRZ272- TMS320F28335, F28334, F28332, TMS320F28235, F28234, F28232 数字信号控制器 (DSC) 硅勘误表, 描述了不同版本硅的警告和使用说明。

CPU 用户指南—

SPRU430 - TMS320C28x CPU 和指令集参考指南描述了中央处理单元 (CPU) 和 TMS320C28x 定点数字信号处理器 (DSP) 的汇编语言指令。它还描述了这些 DSP 的仿真功能。

SPRUE02 - TMS320C28x 浮点单元和指令集参考指南描述浮点运算单元并且包括 FPU 的指令。

外围指南—

SPRU566 - TMS320x28xx, 28xxx DSP 外设参考指南描述了 28X 数字信号处理器 (DSP) 的外设参考指南。

SPRUFB0 - TMS320x2833x, 2823x 系统控制和中断参考指南描述了 2833x 和 2823x 数字信号控制器 (DSCS) 的各种中断和系统控制功能。

SPRU812 - TMS320x2833x, 2823x 模拟到数字转换器 (ADC) 参考指南描述如何配置和使用片上 ADC 模块, 这是一个 12 位流水线的 ADC。

SPRU949 - TMS320x2833x, 2823x DSC 外部接口 (XINTF) 参考指南描述 XINTF, 当它在 2833x 和 2823x 设备使用时, 它是一个非复用的异步总线。

SPRU963 - TMS320x2833x, 2823x 引导 ROM 参考指南描述 bootloader (工厂编程的引导加载软件) 的目的和功能, 并提供了代码示例。它还描述设备的片上引导 ROM 的其他内容, 并确定位于该内存中的所有信息的位置。

SPRUFB7 - TMS320x2833x, 2823x 多通道缓冲串行端口 (McBSP) 参考指南描述 2833x 和 2823x 设备的 McBSP。McBSP 允许在一个系统中的 DSP 和其他设备之间的接口的直接访问。

SPRUFB8 - TMS320x2833x, 2823x 直接内存访问 (DMA) 模块参考指南描述 2833x 和 2823x 设备的 DMA。

SPRUG04 - TMS320x2833x, 2823x 增强型脉宽调制器 (ePWM) 模块参考指南描述了增强型脉宽调制器的主要领域, 包括数字电机控制, 开关电源控制, UPS (不间断电源), 及其他形式的电源转换。

SPRUG02 - TMS320x2833x, 2823x 高分辨率脉宽调制器 (HRPWM) 参考指南描述了操作的脉冲宽度调制器的高分辨率扩展 (HRPWM)

SPRUFG4 - TMS320x2833x, 2823x 增强捕捉 (eCAP) 模块参考指南描述增强型捕捉模块。它包括模块说明和寄存器。

SPRUG05 - TMS320x2833x, 2823x 增强的正交编码脉冲 (eQEP) 模块参考指南描述了 eQEP 模块, 在高性能运动和位置控制系统中, 它与线性或旋转增量编码器连接测量旋转机械的位置, 方向, 和速度信息。它包括模块说明和寄存器

SPRUEU1 - TMS320x2833x, 2823x 增强型控制器局域网 (ECAN) 参考指南描述在电气噪声环境中使用确定的协议与其它控制器通信的串行通信 ECAN。

SPRUZF5 - TMS320x2833x, 2823x 串行通信接口 (SCI) 参考指南介绍的 SCI, 它是一个两线异步串行口, 俗称为 UART。"SCI 模块支持 CPU 与其它异步外设之间的数字通信, 它使用标准的非归零 (NRZ) 格式。

SPRUEU3 - TMS320x2833x, 2823x DSC 串行外设接口 (SPI) 参考指南介绍了 SPI - 一个高速

同步串行输入/输出 (I/O) 端口 - 允许在编程的位传输速率下可编程数据流 (1 到 16 位) 被移入和移出器件。

SPRUG03 - TMS320x2833x, 2823x 内部集成电路 (I2C) 模块参考指南描述的内部集成电路 (I2C) 模块的功能和操作。

工具指南一

SPRU513 - TMS320C28x 汇编语言工具 V5.0.0 用户指南介绍了汇编语言工具 (汇编器和其他用于开发汇编语言代码工具), 汇编指令, 宏, 通用对象文件格式, 和 TMS320C28x 器件符号调试指令。

SPRU514 - TMS320C28x 优化 C/ C++ 的编译器 V5.0.0 用户指南介绍了 TMS320C28x™C / C++ 编译器。这个编译器接受 ANSI 标准的 C / C++ 源代码和为 TMS320C28x 器件的 TMS320 DSP 汇编语言源代码生成。

SPRU608 - TMS320C28x 指令集仿真器技术概述介绍了这样一个仿真器, 它应用于针对 TMS320C2000 IDE 的 Code Composer Studio 中, 仿真 C28x™核心指令集。

SPRU625 - TMS320C28x DSP / BIOS 的 5.32 应用编程接口 (API) 参考手册介绍了使用 DSP / BIOS 进行的开发工作。

外部接口 (XINTF) 采用非复用的异步总线, 类似的 TMS320x281x 外部接口。

本指南适用 TMS320x2833x 系列处理器的 XINTF。这包括所有 2833x 系列内基于 Flash 和基于 RAM 的器件。

1 功能描述

在图 1 中显示 XINTF 是映射到三个固定存储映像区域。

每个 28X XINTF 区都有一个片选信号, 用于访问特定区域。在一些器件上的两个区域的片选信号内部“与”在一起, 形成一个共享芯片选择。在这种方式中, 同一个存储器可连接到两个区, 或可以用外部解码逻辑来区分隔这两个区。

三个区域, 每个区都可以编程指定等待状态数, 选通信号建立和保持时间。在一个读访问和写访问中, 等待状态数, 选通信号建立时间和保持时间均可分别指定。此外, 每个区域都可以使用 XREADY 信号扩展外部等待状态或不扩展。可编程的等待状态, 芯片选择和可编程选通时间使接口与外部存储器和外设相脱离。

每个空间的访问等待、选择、建立及保持时间可以通过 XTIMINGx 寄存器进行配置。外部接口的访问时钟频率由内部的 XTIMCLK 提供, XTIMCLK 可以等于 SYSCLKOUT 或 SYSCLKOUT/2。XTIMCLK 对所有的 Zone 空间有效。而且, XINTF 的总线周期由 XCLKOUT 的上升沿开始。计时及事件由 XTIMCLK 上升沿触发。

1.1 与 TMS320x281x XINTF 的区别

XINTF 在本文档中描述的功能非常类似 TMS320x281x XINTF。主要区别是:

- 数据总线宽度:

每一个 XINTF 空间都可以独立配置为 16 位或 32 位数据总线, 由于 32 位模式可以单周期读写 32 位数据, 因此能提高系统的性能。数据总线的宽度不影响每个 XINTF 访问空间的大小。在 32 位模式下, 低位地址线 XA0 作为低半字的使能控制信号使用。而 281x 的 XINTF 只支持 16 位操作模式。

- 寻址范围:

28335 处理器的 XINTF 接口有 20 位地址线, Zone6 和 Zone7 支持 1Mx16 字的寻址空间, 而 281x 最大寻址范围为 512Kx16 字。

- 直接存储器访问(DMA):

所有 XINTF zone 存储扩展区都连接到 DMA 模块上。当 CPU 处理其他数据时, 可以使用

DMA 控制器实现数据的交换。281x 没有 DMA 控制器。

- XINTF 时钟使能信号:

为了降低处理器功耗, 2833x 的 XINTF 时钟信号(XTIMCLK)默认情况下被禁止, 可以通过寄存器 PCLKCR3 的第 12 位置 1 使能时钟。关闭 XTIMCLK 不影响 XCLKOUT 的输出, 两个信号独立控制。在 281x 系列 DSP 上 XTIMCLK 信号总是被使能

- XINTF 引脚复用:

很多 XINTF 引脚与通用目的 IO 复用, 在使用 XINTF 之前必须通过 GPIO 复用寄存器配置相应的功能。2812 处理器的 XINTF 引脚为专用引脚。

- 外部扩展区 Zone 和片选信号:

2833x 的 XINTF 扩展区减少到三个:Zone0,Zone6 和 Zone7。每一个扩展区都有专门的片选信号。Zone0 仍然保留了读紧跟写操作的保护功能。281x 有些外部扩展区公用一个片选信号, 其中 Zone0 和 Zone1 公用 XZCSOANDI,Zone6 和 Zone7 公用 XZCS6AND7。

- Zone7 存储器映射:

2833x 的 Zone7 总是映射到外部扩展空间, 而 281x 要根据 MP/MC 输入信号的状态确定映射的空间地址。Zone6 和 Zone7 不公用任何地址空间, 281x 的 Zone7 和 Zone6 映射地址相同。

- Zone 存储器地址:

Zone0 起始地址为 0x4000, 寻址范围为 4Kx16 位;281x 的 Zone0 起始地址为 0x2000, 寻址范围为 8Kx16 位。Zone6 和 Zone7 寻址范围都是 1Mx16 位, 起始地址分别为 0x100000 和 0x200000。在 281x 上这两个空间的寻址范围分别为 512Kx16 位和 16Kx16 位。

- EALLOW 保护:

2833x 的 XINTF 寄存器支持 EALLOW 保护, 281x 的 XINTF 寄存器并不支持 EALLOW 保护。对于时序信息, 请参阅您的特定器件的最新数据手册。

1.2 与 TMS320x2834x XINTF 差异

在本文档中描述的 XINTF, 在功能上是与 TMS320x2833x/2823x XINTF 非常相似的。

主要区别是:

- XA0 和 WE1

对于 F2833x/F2823x, XA0 和 WE1 共用一个引脚; 然而, C2834x 设, 他们是独立的引脚。

- XBANK 周期选择

您必须根据 XTIMCLK 和 XCLKOUT 的比例确定延迟周期数。请参考第 5 节。而 C2834x 设备没有这个要求。

对于时序信息, 请参相应特定器件的最新数据手册。

1.3 访问 XINTF 空间

在 28x 中, XINTF Zone 是一段直接访问外部接口的存储器映射区域。与某个 Zone 关联的外设或存储设备可以直接被 CPU 或者 CCS 访问。

每个空间的读写访问操作时序都可以独立进行配置。每个空间都有一个与之相关联的片选信号, 而且片选信号低电平有效。在 2833x 和 2823x 的片子上, 每个空间的片选信号是相互独立的。

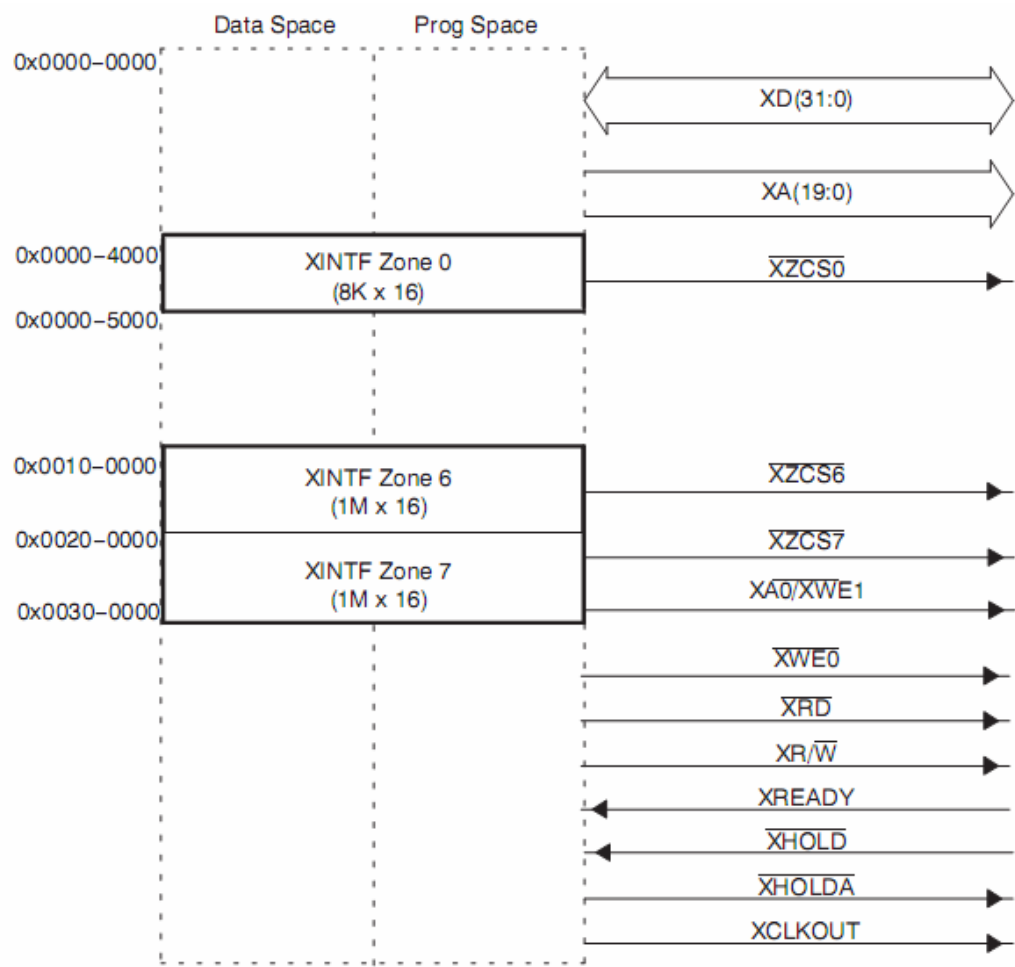
所有的空间共享 20 位的外部地址总线, 处理器根据被选通的 Zones 而产生相应的地址。具体如下:

- Zone0 分配的外部地址范围为 0x00000-0x00FFF。当 CPU 访问 Zone0 空间的第一个字时, 地址总线产生 0X00000; 当 CPU 访问 Zone0 空间的最后一个字时, 地址总线产生 0X00FFF 地址。需要注意的是, 访问 Zone0 空间时需要将 Zone0 的片选信号(XZCS0)拉低。

- Zone6 和 Zone7 共享外部地址总线, 地址范围为 0x00000-0xFFFFF。因为 Zone6 和 Zone7

空间使用两个不同的片选信号（XZCS6 or XZCS7），因此可以通过片选信号来区分对两个空间的访问，只需要将相应的片选信号拉低即可。

图 1、外部接口框图

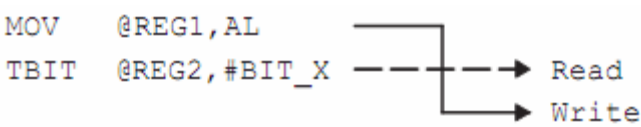


- A 每一个区域都可以设定不同的等待状态，建立和保持时间。并且支持通过片选信号来执行一个特定区域的访问。这些特点使芯片与许多外部的存储器和外设实现无缝连接。
- B 区 1 - 5 保留为将来的扩展。
- C 当 XINTF 时钟是在 PCLKCR3 启用时，所有区域启用。

1.4 写操作紧跟读操作的流水线保护

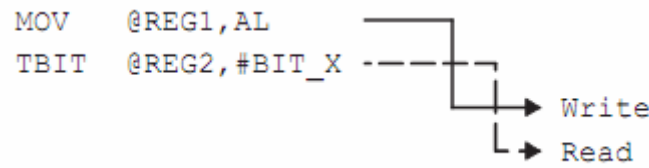
在 28x 的 CPU 流水线中，读操作的相位在写操作相位的前面。由于这个操作次序，写操作紧跟读操作，实际顺序是读操作执行后进行写操作。

例如，实现向一个地址写然后从另一个地址读的操作指令，因为写操作紧跟读操作的流水线保护，所以在写之前已经完成了读，如下所示：



在 28x 上，硬件会保护外设寄存器所占用的存储空间，避免操作次序被颠倒。这样的存储空间称为写操作紧跟读操作流水线保护空间。在 F28335 上，Zone0 空间为默认的写操作紧跟读操作流水线保护。写和读 Zone0 空间采用程序描述的次序。例如，完成一个写操作紧

跟读操作的指令执行次序如下：



28X CPU 自动保护写操作紧跟读操作到相同的内存区域。上面描述的保护机制是地址不一样的情况下，但在某一受保护内存的区域。在这种情况下，执行的顺序是通过 CPU 在读取发生之前自动插入足够的 NOP 指令周期完成写操作来保护的。

该执行的顺序只有当外设映射到 XINTF 时才有保护意义。一个寄存器的写操作可更新状态位到另一个寄存器。在这种情况下，第一个寄存器的写必须在第二个寄存器的读发生前完成。如果读写操作在正常的顺序发生，错误的状态可能会被读取当写在读后发生。这逆转只有当内存映射到 XINTF 才有意义。因此，0 区一般不被用来访问存储器，而只是将用于访问外设。

如果其他区域用于访问外设，需要写操作紧跟读操作的指令顺序，以下解决方案可用于：

- 写和读指令间添加 3 个 NOP 汇编指令。如果代码经分析发现是其他原因导致的流水线堵塞可用少于 3 个 NOP。
- 移动其他指令到读之前，以确保写读之间，至少有三个 CPU 周期相隔。
- 使用 MV 编译器写和读访问之间自动插入 NOP 汇编指令。这个选项应该谨慎使用，因为这个非正常次序的执行是只有当访问外设映射到 XINTF 和非正常的内存访问才能考虑。

2 XINTF 功能配置

本节是 XINTF 各种参数综述，这些参数可被配置为适应特定系统要求的参数。取决于使用的精确的配置 28X 器件的工作频率，XINTF 的转换特点，和外部器件的定时要求。这些器件详细资料参照后面的章节。

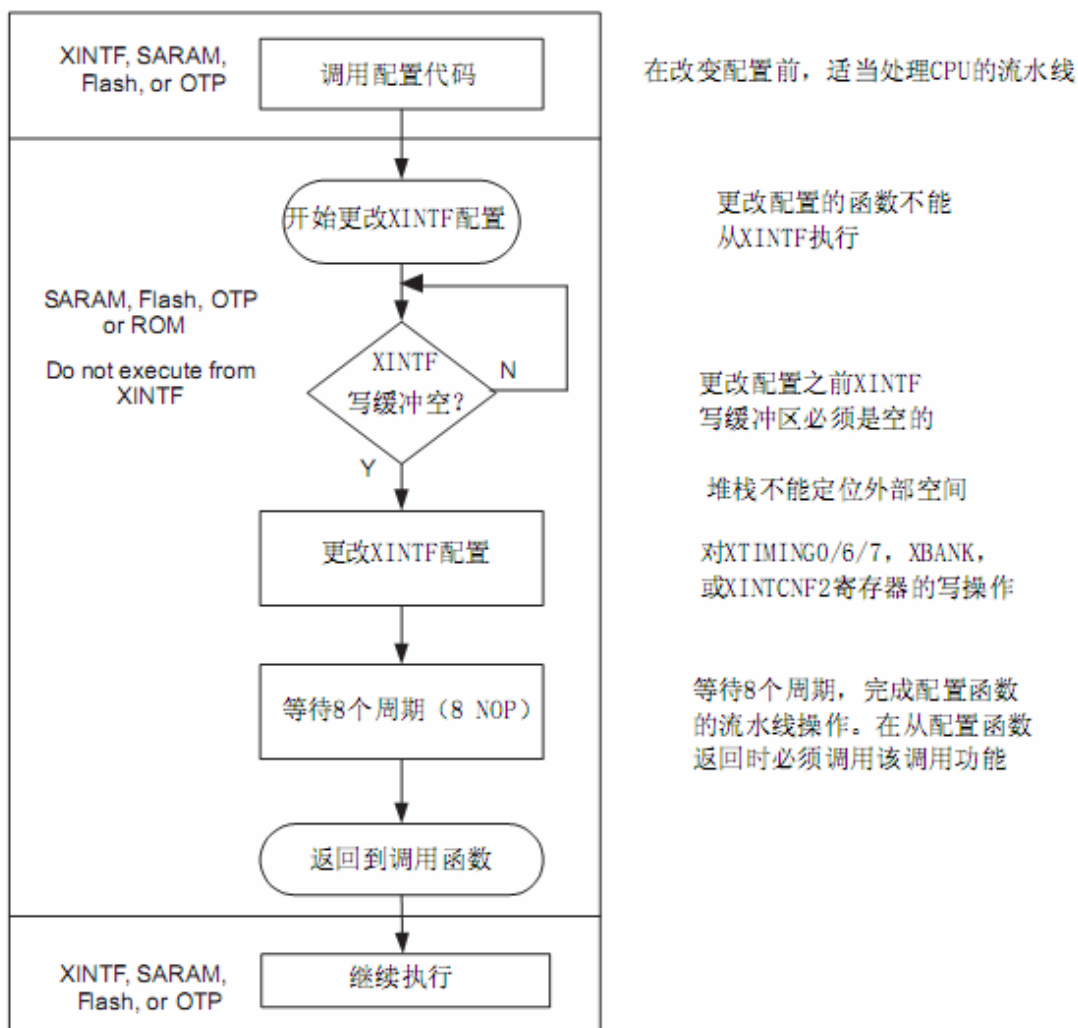
因为 XINTF 配置参数的变化会导致访问定时的变化，因此配置这些参数的代码不应该由 XINTF 本身执行。

2.1 外部接口(XINTF)配置寄存器及时序寄存器的设置

在改变 XINTF 配置寄存器和时序寄存器的过程中，不能对 XINTF 进行如下操作：仍在 CPU 流水线上的指令对 XINTF 的访问、XINTF 写缓冲器内的写访问、数据读写和预先取指操作。为保证在改变配置过程中不访问 XINTF，配置过程需要遵循以下步骤：

- 1、确保 DMA 没有访问 XINTF。
- 2、任何配置 XTIMING0/6/7, XBANK 或 XINTCNF2 寄存器的操作，都必须采用图 2 所示的流程：

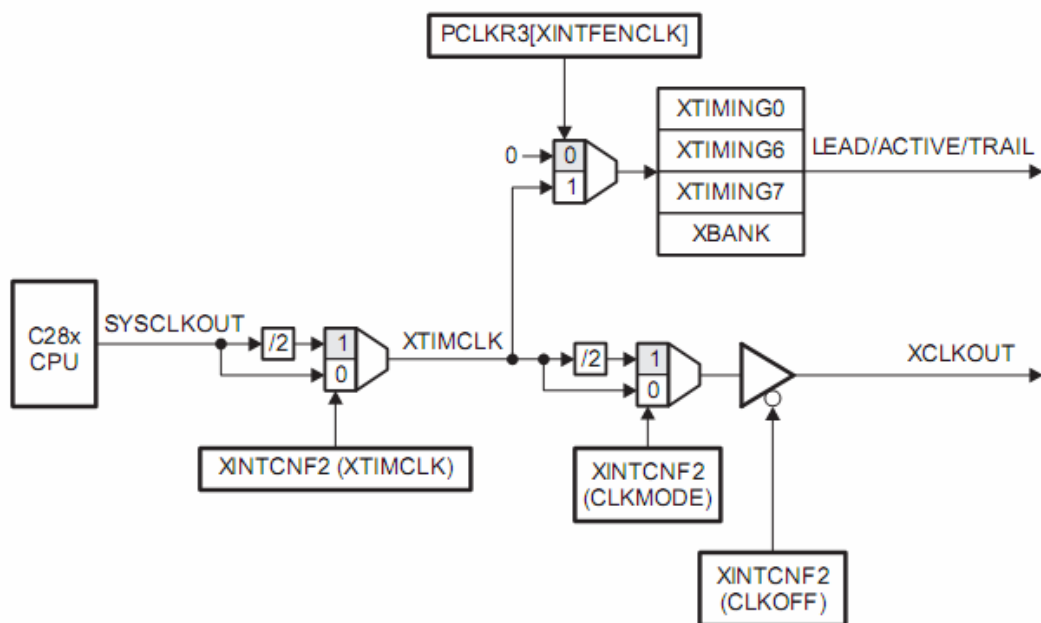
表 2、访问流程图



2.2 XINTF 时钟

XINTF 模块使用的时钟有两个：内部时钟 XTICK 和外部时钟输出 XCLKOUT。图 3 显示了这两个时钟和 CPU 时钟 SYSCLKOUT 之间的关系。

图 3 、XTIMCLK 和 SYSCLKOUT 之间的关系



所有的外部扩展访问都是以内部的时钟 XTIMCLK 为参考的，因此在配置 XINTF 时，首先要检测 XINTF 的 XTIMCLK 所需的与 SYSCCLKOUT 有关的比例，通过 XINTFCNF2 寄存器配置。XTIMCLK 可以配置为 SYSCCLKOUT，也可以配置为 SYSCCLKOUT/2，XTIMCLK 默认的值是 SYSCCLKOUT/2。

外部接口还提供一个时钟输出 XCLKOUT，所有外部接口的访问都是在 XCLKOUT 的上升沿开始。可以通过 XINTFCNF2 寄存器的 CLKMODE 位配置 XCLKOUT 的频率。XCLKOUT 可以配置为 XTIMCLK，也可以配置为 XTIMCLK/2，XCLKOUT 默认的值是 XTIMCLK/2，也就是 SYSCCLKOUT/2。

此外，该时钟输出可以由外部逻辑锁定关闭。为了减少系统噪声干扰，可以将 XINTCNF2[CLKOFF]置位来关闭 XCLKOUT 时钟输出。

2.3 写缓冲

默认情况下，写缓冲是被屏蔽的。大多数情况下，为提高 XINTF 的性能，需要使能写缓冲访问模式。写缓冲在不停止 CPU 的情况下，最多可允许 3 个数据通过缓冲方式向 XINTF 写数据。写缓冲器的深度可以在 XINTFCNF2 寄存器内配置。

2.4 XINTF Zone 访问的建立、激活和跟踪时序

XINTF 是直接访问外部接口的存储器映射区域。任何对 XINTF 空间的读或写操作的时序都可以分为三个阶段：建立、激活和跟踪。在寄存器 XTIMING 中可以设置每个 XINTF 空间访问各阶段时等待的 XTIMCLK 周期数。读写访问操作的时序可以独立进行配置。除此之外，为了能够与慢速外设接口，还可以使用 X2TIMING 位使访问特定空间的建立、激活和跟踪等待状态延长 1 倍。

在建立阶段，访问空间的片选信号变为低电平，产生的地址放在地址总线上(XA)。建立的周期可以通过 XTIMING 寄存器进行配置。默认情况下，建立周期设置为最大，读写访问都是 6 个 XTIMCLK 周期。

在激活阶段访问外部设备，如果是读访问，读选通信号(XRD)变为低电平，数据锁存到 DSP；如果进行的是写访问，写使能(XWE)选通信号变为低电平，数据放到数据总线(XD)上。如果访问的空间配置为判断 XREADY 信号操作方式，外设可以控制 XREADY 信号扩展激活状态周期，使激活状态周期超过寄存器设置的等待周期。

如果不使用 XREADY 信号，总的激活周期就等于一个 XTIMCLK 加上 XTIMING 寄存器中设

置的等待周期数。默认情况下，读写访问的激活等待周期都是 14 个 XTIMCLK 周期。

跟踪周期是指读写选择信号变为高电平后，保持片选信号低电平的一段时间。在 XTIMING 寄存器中可以设置跟踪周期的 XTIMCLK 的个数。默认情况下，跟踪周期设置为最大，读写访问都是 6 个 XTIMCLK 周期。

根据系统设计的要求，可以配置空间的建立、激活和跟踪周期长度，以适合具体外设接口的访问。在配置过程中，需要考虑以下几个问题：

- 最小等待状态的要求，在第 4 节介绍
- XINTF 的时序特性，参考相应的数据手册
- 外部器件的时序特性
- 28x 芯片和外设间的附加延时

2.5 对每个区采样 XREADY

SP 通过检测 XREADY 信号，从而可以延长 DSP 访问处设的激活阶段。器件上所有的 XINTF 空间公用一个 XREADY 信号。每个空间都可以进行独立的配置检测，或不检测 XREADY 信号。此外，每个空间还可以选择同步检测 XREADY 信号或异步检测 XREADY 信号。

•同步采样

如果同步检测 XREADY 信号，则在激活状态结束之前，XREADY 信号的建立和保持时序必须与一个 XTIMCLK 的边沿相关联。也就是说，在访问确定的总的建立+激活周期之前，对 XREADY 信号采样一个 XTIMCLK 周期。

•异步采样

如果是异步检测 XREADY 信号，则在激活状态结束之前，XREADY 信号的建立和保持时序必须与 3 个 XTIMCLK 的边沿相关联。也就是说，在访问确定的总的建立+激活周期之前，对 XREADY 信号采样 3 个 XTIMCLK 周期。

对于同步和异步采样，如果检测到 XREADY 信号为低电平，周期的激活阶段将扩展一个 XTIMCLK 周期，在下一个 XTIMCLK 周期继续检测 XREADY 信号。一直重复这个过程，直到检测到 XREADY 信号为高电平，完成正常的访问。

如果设置一个空间检测 XREADY 信号，则对该空间的读写操作都检测 XREADY 信号。默认情况下，每个空间设置为异步检测 XREADY 信号。当使用 XREADY 信号时，需要考虑最小等待状态的需要，同步和异步检测 XREADY 信号对于最小的等待状态要求不同，与下面的因素有关：

- XINTF 的时序特性，参考相应的数据手册述。
- 外部器件的时序特性。
- 28x 芯片和外设间的附加延时。

2.6 储体转接

当从 XINTF 的一个空间切换到另一个空间时，为了能够及时地释放总线给其他设备使用，一些慢速外设可能需要额外的周期。空间切换允许用户指定一个特殊的空间，可以在该空间与其他空间来回切换的过程中增加额外的周期，增加的周期数在 XBANK 寄存器中配置。至于所增加的周期数目是有严格要求的，这个在以后的具体配置中我会提到。当然你也去找手册。

2.7 区数据总线宽度

每个 XINTF zone 都可以独立配置为 16 位或 32 位总线宽度。根据具体的配置，XAO/XWE1 信号的功能会发生相应的变化。当 XINTF zone 配置为 16 位操作模式时(XTIMINGx[XSIZE]=3)，XAO/XWE1 作为最低地址位 XA0 功能，此种配置模式的总线扩展连接图下图所示。表 1 和表 2 给出了 16 位和 32 位操作模式时 XWE0 和 XAO/XWE1 的功能配置。

总线的宽度可以由 XTIMINGx[XSIZE]来定义，三个区域的宽度（XTIMINGx[XSIZE]配置）彼

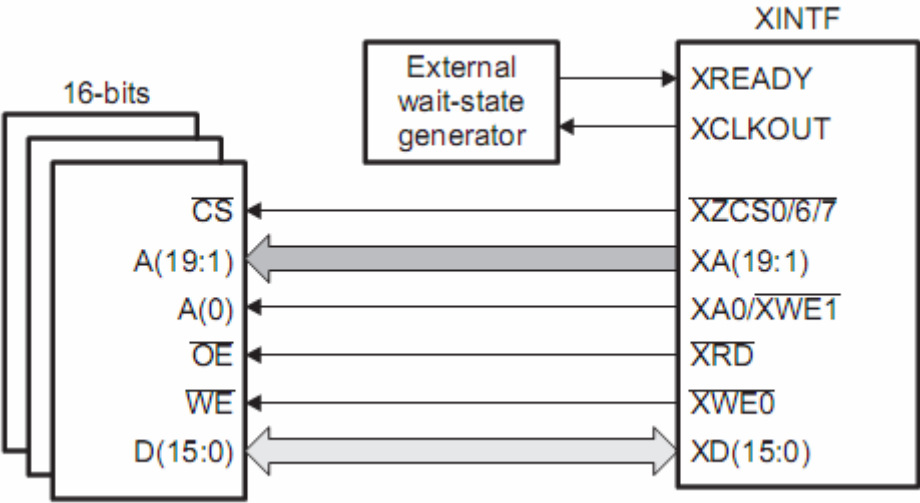
此不同，当连续一次访问两个总线宽度不同的 Zone 空间的时候，两个 Zone 的访问之间要加入延时时钟。这个可以通过第 5 节配置 XBANK 来实现。

例如，给定的区域配置如下：

- 0 区配置为 16 位模式 (XTIMING0[XSIZE] = 3)
- 6 区配置为 32 位模式 (XTIMING6[XSIZE] = 1)
- 7 区配置为 32 位模式 (XTIMING7[XSIZE] = 1)

如果需要连续访问 Zone0 和 Zone6 或者 Zone0 和 Zone7，那么要在 Zone0 访问过后加入至少一个空间切换延时来释放总线。(XBANK= 0, XBANK[BCYC]= 1)。请参阅第 5 节配置 XBANK 周期。

图 4 XINTF 典型 16 位数据总线连接



当 XINTF zone 配置为 32 操作模式时(XTIMINGx[XSIZE]=1)，XA0/XWE1 信号低电平有效，作为 XWE1 功能。使用 XWE1 和 XWE0 实现 32 位总线扩展操作

图 5、 XINTF 典型 32 位数据总线连接

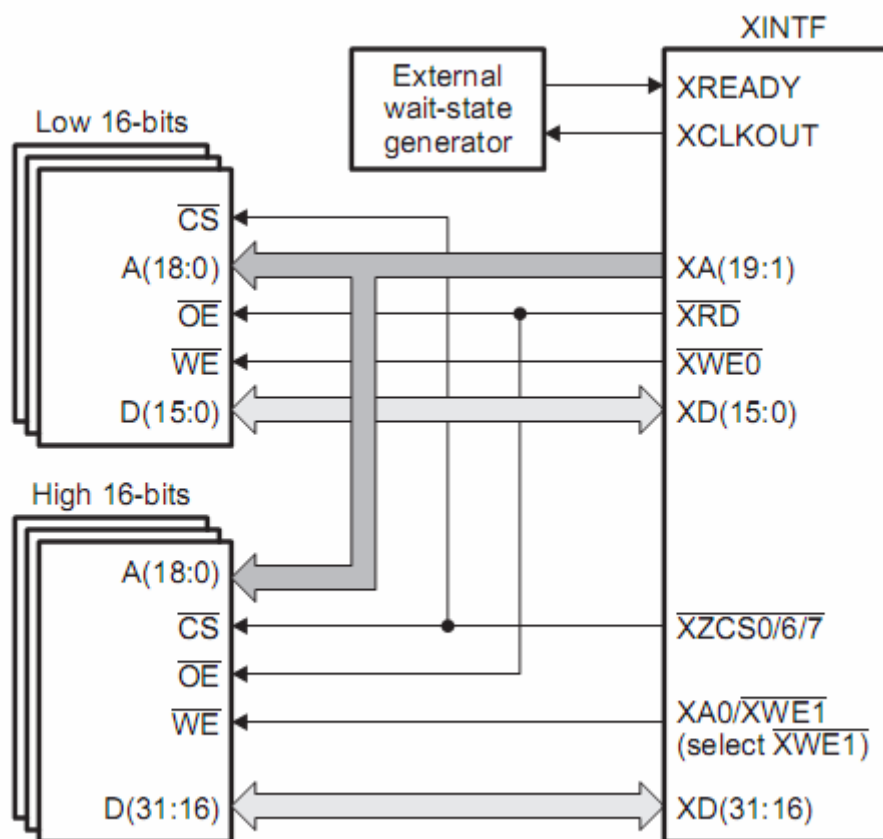


Table 1. 16位模式性能

16-bit Mode 写地址	XA0/XWE1	XWE0
空地址	1	1
16-bit 偶地址	0	0
16-bit 奇地址	1	0

Table 2. 32位模式性能

32-bit Mode 写地址	XA0/XWE1	XWE0
空地址	1	1
16-bit 偶地址	1	0
16-bit 奇地址	0	1
32-bit 值	0	0

3 外部 DMA 支持(XHOLD,XHOLDA)

对片外程序和数据区，XINTF 支持直接存储器访问（DMA）。通过 XHOLD 信号输入和 XHOLDA 输出来完成。当 XHOLD 低电平有效时，所有外部接口保持高阻态。在所有外部接口访问完成时，XHOLDA 低电平有效。当外部接口使外部器件的输出为高阻态以及其他器件控制对外部存储器或外设的访问时，XHOLDA 向外部器件发出信号。

当检测到一个有效的 XHOLD 信号时，XINTCNF2 寄存器中的 HOLD 方式位能自动产生 XHOLDA 信号并允许访问外部总线。当处于 HOLD 模式下，CPU 可以继续执行与存储器总线相连的片内存储器的代码。而 XHOLDA 低电平时，如果尝试访问外部接口，将产生一个未准

备好的条件，制止处理器运行。XINTCNF2 寄存器的状态位将指示 XHOLD 和 XHOLDA 信号的状态。

如果 XHOLD 是有效的，并且 CPU 尝试写 XINTF，则写不被缓存，并且 CPU 会停止。写缓冲区无效。

XINTCNF2 寄存器的 HOLD 模式位较 XHOLD 输入信号优先。因此，可使用户代码来确定是否有 XHOLD 请求。

采取任何行动之前，XHOLD 输入信号在 XINTF 的输入端被同步。同步与 XTIMCLK 有关。

XINTCNF2 寄存器的 HOLDS 位反映了 XHOLD 输入的当前同步状态。

复位时，HOLD 模式位有效，允许使用 XHOLD 请求装入外部存储器的引导。复位期间，如果 XHOLD 信号是低有效，像一般操作，XHOLDA 信号变为低电平。

上电时，在 XHOLD 同步锁存器中的任何未定义的值将被忽略，并在时钟稳定时，最终将被冲掉。因此，同步锁存器不需要进行复位。

如果检测到 XHOLD 信号为低电平，XHOLDA 信号只有在所有未决的 XINTF 周期完成后，变为低电平。如果把他们用于 XINTF，任何待决的 CPU 周期被封锁且 CPU 保持未准备好状态。

定义：

待定 XINTF 周期——当前在 XINTF 的 FIFO 队列中的任何周期。

待决 CPU 周期——不在 FIFO 队列中，但在 CPU 内核存储器总线上有效的任何周期。

XHOLD 信号直到 XHOLDA 信号有效才该被删除。生如果违反本规则，不可预知的结果会发生。

在 HOLD 模式下，XINTF 外部信号的状态如下：

Signal	HOLD 准许模式
XA(19:1)	高阻态
XD(31:0)	高阻态
XA0/ XWE1	高阻态
$\overline{\text{XRD}}$, $\overline{\text{XWE0}}$, $\overline{\text{XR}\overline{\text{W}}}$	高阻态
$\overline{\text{XZCS0}}$	高阻态
$\overline{\text{XZCS6}}$	高阻态
$\overline{\text{XZCS7}}$	高阻态

4 建立，激活及跟踪状态的配置

可以调整 XINTF 信号以匹配特定外围设备的的需求，如读写访问的建立及保持时间。时序参数，定时参数可以在每个 XINTF 区域的 XTIMING 寄存器中单独配置。每个区还可以配置为忽略或者采样 XREADY 信号。这可以让你最大限度地提高基于基于存储器或正在的访问的外围设备的 XINTF 效率。

表 3 显示了 XTIMING 寄存器中进行配置的参数与依据 XTIMCLK 周期的脉冲持续之间 TC (XTIM) 的关系，。

表 3 依据 XTIMCLK 周期的持续脉冲

	Description	Duration (ns) ⁽¹⁾ ⁽²⁾	
		X2TIMING = 0	X2TIMING = 1
LR	建立阶段，读访问	$\text{XRDLEAD} \times t_{\text{Q}(d,m)}$	$(\text{XRDLEAD} \times 2) \times t_{\text{Q}(d,m)}$
AR	激活阶段，读访问	$(\text{XRDACTIVE} + \text{WS} + 1) \times t_{\text{Q}(d,m)}$	$(\text{XRDACTIVE} \times 2 + \text{WS} + 1) \times t_{\text{Q}(d,m)}$
TR	跟踪阶段，读访问	$\text{XRDTRAIL} \times t_{\text{Q}(d,m)}$	$(\text{XRDTRAIL} \times 2) \times t_{\text{Q}(d,m)}$
LW	建立阶段，写访问	$\text{XWRLEAD} \times t_{\text{Q}(d,m)}$	$(\text{XWRLEAD} \times 2) \times t_{\text{Q}(d,m)}$
AW	激活阶段，写访问	$(\text{XWRACTIVE} + \text{WS} + 1) \times t_{\text{Q}(d,m)}$	$(\text{XWRACTIVE} \times 2 + \text{WS} + 1) \times t_{\text{Q}(d,m)}$
TW	跟踪阶段，写访问	$\text{XWRTRAIL} \times t_{\text{Q}(d,m)}$	$(\text{XWRTRAIL} \times 2) \times t_{\text{Q}(d,m)}$

(1) tc(xtim) - 周期时间, XTIMCLK

(2) WS 是指在使用 XREADY 时，通过硬件插入等待状态的数目。如果该区域配置为忽略 XREADY (USEREADY=0)=0 则 WS=0。

注：最小等待状态的配置必须通过每个区的 XTIMING 寄存器。这些等待状态是由与之接口的定时需要指定的。对于一个特定的设备要求的信息，请参阅该设备的数据手册。
不包括内部器件的硬件可以去检测非法的设置。

4.1 忽略 USEREADY 信号

如果 XREADY 信号被忽略 (USEREADY=0)，然后下面的要求必须得到满足：

建立： $LR \geq tc(XTIM)$

$LW \geq tc(XTIM)$

这些要求导致了下列 XTIMING 寄存器配置限制

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
Valid	≥ 1	≥ 0	≥ 0	≥ 1	≥ 0	≥ 0	0, 1

不采样 XREADY 时，有效和无效定时的例子：

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
Invalid ⁽¹⁾	0	0	0	0	0	0	0, 1
Valid	1	0	0	1	0	0	0, 1

(1)没有硬件来检测非法 XTIMING 配置

4.2 同步模式(USEREADY=1,READYMODE=0)

如果在 XREADY 信号进行采样在同步模式 (USEREADY= 1, READYMODE= 0)，那么必须满足以下要求：

- 1 建立： $LR \geq TC(XTIM)$
 $LW \geq TC(XTIM)$
- 2 激活： $AR \geq 2 \times TC(XTIM)$
 $AW \geq 2 \times TC(XTIM)$

注：限制不包括外部硬件等待状态。

这些要求导致了下列 XTIMING 寄存器的配置限制：

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 1	≥ 1	≥ 0	≥ 1	≥ 1	≥ 0	0, 1 ⁽¹⁾

(1) 没有硬件来检测非法 XTIMING 配置

有效和无效的时序，使用同步 XREADY 时的例子：

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
Invalid ⁽¹⁾	0	0	0	0	0	0	0, 1
Invalid ⁽¹⁾	1	0	0	1	0	0	0, 1
Valid	1	1	0	1	1	0	0, 1

(1) 没有硬件来检测非法 XTIMING 配置

4.3 异步模式(USEREADY=1,READYMODE=1)

如果在 XREADY 信号进行采样，在异步模式 (USEREADY= 1, READYMODE= 1)，那么必须满足以下要求：

- 1 建立： $LR \geq tc(XTIM)$
 $LW \geq tc(XTIM)$
- 2 激活： $AR \geq 2 \times tc(XTIM)$
 $AW \geq 2 \times tc(XTIM)$
- 3 建立 + 激活： $LR + AR \geq 4 \times tc(XTIM)$
 $LW + AW \geq 4 \times tc(XTIM)$

这些要求导致了下列 3 种可能的 XTIMING 寄存器的配置：

注：限制不包括外部硬件等待状态。
这些要求导致了下列 XTIMING 寄存器的配置限制：

XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 1	≥ 2	0	≥ 1	≥ 2	0	0, 1
XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
≥ 2	≥ 1	0	≥ 2	≥ 1	0	0, 1

有效和无效的时序，使用同步 XREADY 时的例子：

	XRDLEAD	XRDACTIVE	XRDTRAIL	XWRLEAD	XWRACTIVE	XWRTRAIL	X2TIMING
Invalid ⁽¹⁾	0	0	0	0	0	0	0, 1
Invalid ⁽¹⁾	1	0	0	1	0	0	0, 1
Invalid ⁽¹⁾	1	1	0	1	1	0	0
Valid	1	1	0	1	1	0	1
Valid	1	2	0	1	2	0	0, 1
Valid	2	1	0	2	1	0	0, 1

(1) 没有硬件来检测非法 XTIMING 配置
表 4 和表 5 显示建立/激活/跟踪的 XTIMCLK/X2TIMING 之间的关系。

表 4、lead/Trail 值与 XTIMCLK/X2TIMING 之间的关系

建立 跟踪值	[XTIMCLK] ⁽¹⁾	X2TIMING ⁽²⁾	建立 SYSCLKOUT 周期	跟踪 SYSCLKOUT 周期
公式	0	0	建立值 * 1	跟踪值 * 1
	0	1	建立值 * 2	跟踪值 * 2
	1	0	建立值 * 2	跟踪值 * 2
	1	1	建立值 * 4	跟踪值 * 4
0	X	X	无有效值	0
1	0	0	1	1
	0	1	2	2
	1	0	2	2
	1	1	4	4
2	0	0	2	2
	0	1	4	4
	1	0	4	4
	1	1	8	8
3	0	0	3	3
	0	1	6	6
	1	0	6	6
	1	1	12	12

- (1) XINTCNF2 [XTIMCLK]配置 SYSCLKOUT 和 XTIMCLK 之间的比例。
(2) X2TIMING 每个区域中的特定 XTIMING 寄存器特定配置区域。

表 5 Active 数值与 XTIMCLK/X2TIMING 方式之间的关系

Active Value	XTIMCLK ⁽¹⁾	X2TIMING ⁽²⁾	Total Active SYSCLKOUT Cycles (includes 1 implied active cycle)
公式	0	0	Active Value * 1 + 1
	0	1	Active Value * 2 + 1
	1	0	Active Value * 2 + 2
	1	1	Active Value * 4 + 2
0	0	X	1 or Invalid if XREADY used (USEREADY = 1)
	1	X	2 or Invalid if XREADY used (USEREADY = 1)
1	0	0	2
	0	1	3
	1	0	4
	1	1	6
2	0	0	3
	0	1	5
	1	0	6
	1	1	10
3	0	0	4
	0	1	7
	1	0	8
	1	1	14
4	0	0	5
	0	1	9
	1	0	10
	1	1	18
5	0	0	6
	0	1	11
	1	0	12
	1	1	22
6	0	0	7
	0	1	13
	1	0	14
	1	1	26
7	0	0	8
	0	1	15
	1	0	16
	1	1	30

5 配置 XBANK 周期

当从 XINTF 的一个空间切换到另一个空间时，为了能够及时地释放总线给其他设备使用，一些慢速外设可能需要额外的周期。空间切换允许用户指定一个特殊的空间，可以在该空间与其他空间来回切换的过程中增加额外的周期，增加的周期数在 XBANK 寄存器中配置。

你需要基于 XTIMCLK 和 XCLKOUT 速率来决定增加的周期数目。共有一下三种情况：

•情况 1: XTIMCLK = SYSCLKOUT

当 XTIMCLK 等于 SYSCLKOUT 的时候，XBANK[BCYC]的选择是没有限制。

•情况 2: XTIMCLK = 1 / 2 SYSCLKOUT 和 XCLKOUT = 1 / 2 XTIMCLK

在这种情况下，只要 XBANK[BCYC]的值不是 4 或者 6，其他的任何值都是允许的。

•情况 3: $\text{XTIMCLK} = 1 / 2 \text{ SYSCLKOUT}$ 和 $\text{XCLKOUT} = \text{XTIMCLK}$

•情况 4: $\text{XTIMCLK} = 1 / 4 \text{ SYSCLKOUT}$

延迟周期需要加在两个 Zone 访问之间，即延迟周期之前访问一个 Zone，之后再访问另一个 Zone。为了能够准确的将延迟周期加在两个访问之间，前一个空间总的访问时间要大于要加入延迟时间，这就要求配置的 $\text{XBANK}[\text{BCYC}]$ 值要小于空间总的访问时间。

用下面的例子来说明一下：假定 Zone7 使能空间切换($\text{XBANK}[\text{BANK}] = 7$)，那么任何针对 Zone7 的访问，无论是读或者写，都会被加入延迟周期。而且该延迟周期要小于前一个空间总的访问周期。也就是说：

- 一 如果访问 Zone0 之后紧接着访问 Zone7，那么 Zone0 的访问周期必须大于延迟周期；
 - 一 如果访问 Zone7 之后紧接着访问 Zone0，那么 Zone7 的访问周期必须大于延迟周期；
- 我们也可以用建立，激活，跟踪周期来确保访问时间长于延迟周期。由于 XREADY 只是延长了 Zone 的访问时间，所以此时并不用考虑它。

- 一 如果 X2TIMING 是 0，然后选择：
 - 一 $\text{XBANK}[\text{BCYC}] < \text{XWRLEAD} + \text{XWRACTIVE} + 1 + \text{XWRTRAIL}$ 和
 - 一 $\text{XBANK}[\text{BCYC}] < \text{XRDLEAD} + \text{XRDACTIVE} + 1 + \text{XRDTRAIL}$
- 一 如果 $\text{X2TIMING} = 1$ ，然后选择：
 - 一 $\text{XBANK}[\text{BCYC}] < \text{XWRLEAD} \times 2 + \text{XWRACTIVE} \times 2 + 1 + \text{XWRTRAIL} \times 2$ 和
 - 一 $\text{XBANK}[\text{BCYC}] < \text{XRDLEAD} \times 2 + \text{XRDACTIVE} \times 2 + 1 + \text{XRDTRAIL} \times 2$

表 6 列出了有效的 $\text{XBANK}[\text{BCYC}]$ 不同的时序配置的值。带头作用，积极和线索 XTIMING 寄存器区中指定值。在确定适当的 $\text{XBANK}[\text{BCYC}]$ 值，产量的访问时间最长的使用时间。这可能是读取或写入时机。

表 6.有效 XBANK 配置

Valid $\text{XBANK}[\text{BCYC}]$	Total Access Time	XRDLEAD or XWRLEAD	XRDACTIVE or XWRACTIVE	XRDTRAIL or XWRTRAIL	X2TIMING
< 5	$1 + (2 \times 1) + 1 = 5$	1	2	1	0
< 6	$1 + (3 \times 1) + 1 = 6$	1	3	1	0
< 7	$2 + (3 \times 1) + 1 = 7$	2	3	1	0
< 5	$1 \times 2 + 0 \times 2 + 1 + 1 \times 2$	1	0	1	1
< 5	$1 \times 2 + 1 \times 2 + 1 + 0 \times 2$	1	1	0	1

6 XINTF 寄存器

表 7 显示了 XINTF 的配置寄存器。这些寄存器的修改将影响 XINTF 的访问定时，应执行并仅运行 XINTF 以外的代码，才能修改这些寄存器。

表 7 XINTF 配置和控制寄存器映射

Name	Address	Size (x16)	Description ⁽¹⁾
XTIMING0	0x0000-0B20	2	XINTF Timing Register, Zone 0
XTIMING6 ⁽²⁾	0x0000-0B2C	2	XINTF Timing Register, Zone 6
XTIMING7	0x0000-0B2E	2	XINTF Timing Register, Zone 7
XINTCNF2 ⁽³⁾	0x0000-0B34	2	XINTF Configuration Register
XBANK	0x0000-0B38	1	XINTF Bank Control Register
XREVISION	0x0000-0B3A	1	XINTF Revision Register
XRESET	0x0000 0B3D	1	XINTF Reset Register

- (1) 所有 XINTF 寄存器受 EALLOW 保护。
- (2) XTIMING1 - XTIMING5 保留为将来的扩展，目前没有使用。
- (3) XINTCNF1 是保留的，目前没有使用。

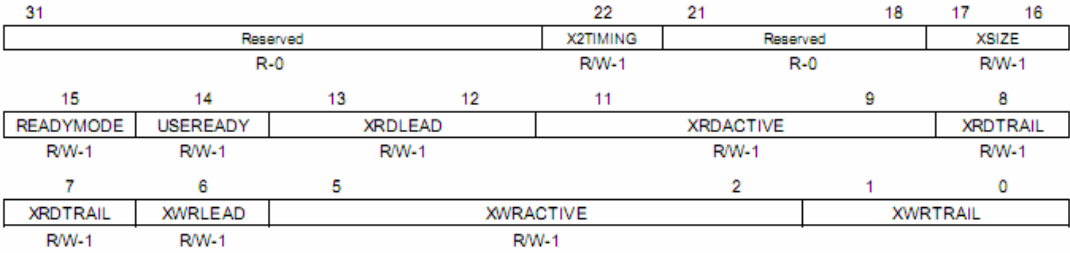
6.1 XINTF 定时寄存器

每个 XINTF 区都有一个定时寄存器。改变该寄存器的值会影响特定的定时。只有通过运行该区域以外的代码，才能修改一个区域的定时寄。

注意：

- 不同模式的最小等待状态要求是如第 2 节所示。
- 28X 的外部器件接口是可以有额外的时序约束。
详情请参阅供应商文档。
- 没有逻辑检测非法设置。

图 6、XTIMING0/6/7 寄存器分布



图例：R / W=读/写；R=只读；- N=复位后的值

表 8 XTIMING0/6/7 寄存器字段说明

位	域	值	说明 (1)
31:23	保留	0	
22	X2TIMING	0 1	该位指定每个定时寄存器XRDLEAD, XRDACTIVE, XRDTRAIL, XWRLEAD, XWRACTIVE, 的缩放因子。 数值的缩放比例为1:1 数值的缩放比例为2:1（一倍）。这是上电和复位的默认操作模式
21:18	保留	0	

位	域	值	说明 (1)
17:16	XSIZE		这两个位必须始终写为0, 1 (32位数据总线) 或1, 1 (16位数据总线)。
		00	任何其他组合被保留, 并会导致不正确的XINTF行为。
		01	保留 - 导致不正确XINTF行为
		10	32位接口。在这种模式下, 该区域将使用所有32个数据线。
		11	XAO/WE1信号的行为将如2.7节所述和WE1相同。
15	READYMODE		设置XREADY对区域的输入采样为同步或异步方式。
		0	如果XREADY不被采样 (USEREADY = 0) 则可忽略该位。
14	USEREADY		XREADY输入为同步方式。
		1	XREADY输入为异步方式。
13	XRDLEAD		确定对区的访问是进行采样还是忽略XREADY输入信号。
		0	区访问时, XREADY信号被忽略。
13:12	XRDLEAD	1	XREADY信号可用于进一步扩展XRDACTIVE和XWRACTIVE定义的已完成周期最小值的比例。
			这两位定义XTIMCLK周期的读周期导致等待状态周期。
			如果X2TIMING位被置位, 那么等待状态的周期增加一倍。
			在不同模式下的最低要求, 请参阅第4节。
			X2TIMING Read Lead Period
		00	X Invalid
		01	0 1 XTIMCLK 周期
			1 2 XTIMCLK 周期
		10	0 2 XTIMCLK 周期
			1 4 XTIMCLK 周期
		11	0 3 XTIMCLK 周期
			1 6 XTIMCLK 周期 (默认)
11:9	XRDACTIVE		三位组合, 用XTIMCLK周期定义读周期的激活等待状态时间, 激活的是默认周期为1 XTIMCLK周期。因此, 总的激活时间为 (1 + XWRACTIVE) 个XTIMCLK的周期。如果X2TIMING位被置位, 那么等待状态周期加倍。
			在不同模式下的最低要求, 请参阅第4节。
			X2TIMING X2TIMING读激活时间等待状态
		000	0 0
		001	0 1 XTIMCLK 周期
			1 2 XTIMCLK 周期
		010	0 2 XTIMCLK 周期
			1 4 XTIMCLK 周期
		011	0 3 XTIMCLK 周期
			1 6 XTIMCLK 周期
		100	0 4 XTIMCLK 周期
			1 8 XTIMCLK 周期
		101	0 5 XTIMCLK 周期
			1 10 XTIMCLK 周期
		110	0 6 XTIMCLK 周期
			1 12 XTIMCLK 周期
		111	0 7 XTIMCLK 周期
			1 14 XTIMCLK 周期 (默认)

位	域	值	说明 (1)
8:7	XRDTRAIL		二位组合, 用XTIMCLK周期定义读周期的跟踪等待状态周期。如果X2TIMING位被置位, 那么等待状态周期加倍。在不同模式下的最低要求, 请参阅第4节。
			X2TIMING 阅读径期
		00	0 0
		01	0 1 XTIMCLK 周期 1 2 XTIMCLK 周期
		10	0 2 XTIMCLK 周期 1 4 XTIMCLK 周期
		11	0 3 XTIMCLK 周期 1 6 XTIMCLK 周期 (默认)
6:5	XWRLEAD		Two-bit field that defines the write cycle lead wait state period, in XTIMCLK cycles. If the X2TIMING bit is set, then the number of wait states are doubled. See Section 4 for minimum requirements in different modes.
			X2TIMING Write Lead Period
		00	0 0
		01	0 1 XTIMCLK 周期 1 2 XTIMCLK 周期
		10	0 2 XTIMCLK 周期 1 4 XTIMCLK 周期
		11	0 3 XTIMCLK 周期 1 6 XTIMCLK 周期 (默认)
4:2	XWRACTIVE		三位组合, 用XTIMCLK周期定义写周期激活等待时间, 激活的时间默认为1 XTIMCLK周期。因此, 总的激活时间为(1+XWRACTIVE)个XTIMCLK的周期。如果X2TIMING位被置位, 那么等待状态周期加倍。在不同模式下的最低要求, 请参阅第4节。
			X2TIMING 写激活时间的等待状态
		000	0 0
		001	0 1 XTIMCLK 周期 1 2 XTIMCLK 周期
		010	0 2 XTIMCLK 周期 1 4 XTIMCLK 周期
		011	0 3 XTIMCLK 周期 1 6 XTIMCLK 周期
		100	0 4 XTIMCLK 周期 1 8 XTIMCLK 周期
		101	0 5 XTIMCLK 周期 1 10 XTIMCLK 周期
		110	0 6 XTIMCLK 周期 1 12 XTIMCLK 周期
		111	0 7 XTIMCLK 周期 1 14 XTIMCLK 周期 (默认)
1:0	XWRTRAIL		用XTIMCLK周期定义写周期跟踪等待期间, XTIMCLK周期。如果X2TIMING位被置位, 那么等待状态周期加倍。在不同模式下的最低要求, 请参阅第4节。
			X2TIMING 写跟踪周期
		00	x
		01	0 1 XTIMCLK 周期 1 2 XTIMCLK 周期
		10	0 2 XTIMCLK 周期 1 4 XTIMCLK 周期
		11	0 3 XTIMCLK 周期 1 6 XTIMCLK 周期 (默认)

6.2 XINTF 配置寄存器

表 7 XINTF 配置寄存器(XINTCNF2)

31											19	18	16
Reserved											XTIMCLK		
R-0											R/W-1		
15						12	11	10	9	8			
Reserved						HOLDAS	HOLDS	HOLD	Reserved				
R-0						R-x	R-y	R-0	R-1				
7	6	5	4	3	2	1	0						
WLEVEL		Reserved		Reserved	CLKOFF	CLKMODE	WRBUFF						
R-0		R-0		R-1	R/W-0	R/W-1	R/W-0						

图例：R / W=读/写; R =只读;- N=复位后的值/ X = XHOLDA 输出; Y = XHOLD 输入

表 9、XINTF 配置寄存器字段说明

Bit	Field	Value	Description ^(*)
31:19	保留		
18:16	XTIMCLK	000 XTIMCLK = SYSCLKOUT/1 001 XTIMCLK = SYSCLKOUT/2 (默认) 010 保留 100 保留 101 保留 110 保留 111 保留	这些位按照XTIMING和XBANK寄存器的定义，选择建立、激活和跟踪的定时转换操作的基本时钟，此设置会影响XINTF所有区。只有通过运行XINTF之外的代码，才能修改XTIMCLK的比例。
15:12	保留		
11	HOLDAS	0 1	该位反映XHOLDA输出信号的当前状态。用户可以读取该位，以确定如果外部接口目前是否同意访问外部设备。 0 XHOLDA 输出信号为低。 1 XHOLDA 输出信号为高。
10	HOLDS	0 1	该位反映XHOLD输入信号的当前状态。用户可以读取该位，以确定外部接口是否正在请求访问外部总线。 0 XHOLD 输入信号是低的。 1 XHOLD 输入信号是高的。
9	HOLD	0 1	该位接纳外部器件的一个请求，这个外部器件驱动XHOLD的输入信号和XHOLDA输出信号。 如果此位被置位，当XHOLD和XHOLDA均为低（外部总线访问允许），那么XHOLDA信号被强制为高（在当前周期结束）且外部接口退出高阻抗模式。 在复位XRS时，此位被设置为零。如果复位时，XHOLD信号是低电平有效，则总线和所有的选通信号必须处于高阻态，且XHOLDA信号也被驱动为低电平有效。 当HOLD模式使能且XHOLDA为低电平有效（外部总线同意激活），那么内核仍然可以从内存中执行代码。如果访问外部接口，将产生一个没有准备好的信号且内核停止直到XHOLD信号被删除。 自动接纳外部设备的一个请求，这个外部器件驱动XHOLD输入信号和XHOLDA输出信号为低（默认） 不能自动接纳外部设备的一个请求，这个外部器件驱动XHOLD输入信号为低电平，而XHOLDA的输出信号保持高电平。

Bit	Field	Value	Description ⁽¹⁾
8	保留		
7:6	WLEVEL	00 当前缓冲区可检测的写入数如下： 缓冲区空 01 当前写缓冲区有一个值 10 当前写缓冲区有两个值 11 当前写缓冲区有三个值	
5:4	保留		
3	CLKOFF	0 关闭XCLKOUT模式。这样可省电和降噪。复位时该位清0。 1 XCLKOUT使能。（默认） XCLKOUT被禁用。	
2	CLKMODE	0 XCLKOUT除以2模式。所有总线定时和使能方式无关，将从XCLKOUT的上升沿开始。 1 上电和复位默认方式为/ 2模式。CLKMODE位的变更，应当由XINTF外的代执行码。 0 XCLKOUT等于XTIMCLK 1 XCLKOUT是XTIMCLK 的一半（默认）	
1:0	WRBUFF	00 不必等待XINTF写访问完成，写缓冲区允许处理器继续执行。写缓冲器深度可选如下： 无写缓冲。CPU停止，直到XINTF上的写完成。 注：复位（XRS）的默认方式。 01 写一次被缓冲，CPU将停止，直到XINTF的写周期开始（XINTF上，有一个读周期激活）。 10 写二次被缓冲，当第三次写时CPU将停止。 CPU会停止，直到第一次写开始了它在XINTF上的周期。 11 写三次被缓冲。如果紧接着写第四次，CPU会停止，直到第一次写开始了它在XINTF上的周期。保留执行的顺序，例如，按照可接受的顺序来完成写操作。在XINTF读操作时，处理器停止运行直到所有待决的写操作读访问完成为止。如果缓冲区已满，任何对缓冲区待决的读或写都将使处理器停止运行。 写缓冲深度是可以改变的，但是，一般建议仅在缓冲区空时改变缓冲区的深度（这可以通过读“写缓冲区深度级别”来检查）。当缓冲区深度级别不为零时，写这些位可能有不可预知的结果。	

6.3 XBANK 寄存器

图 8、XBANK 寄存器

15	6	5	3	2	0
Reserved					BANK
R-0					R/W-1

图例：R / W=读/写; R=只读;- N=复位后的值

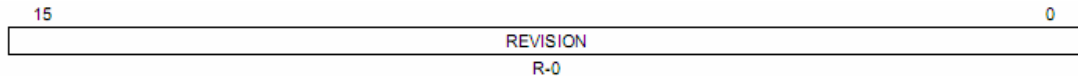
表 10、XBANK 寄存器说明

Bit	Field	Value	Description ⁽¹⁾
15:6	Reserved		
5:3	BCYC	000 0 周期 001 1 XTIMCLK 周期 010 2 XTIMCLK 周期 011 3 XTIMCLK 周期 100 4 XTIMCLK 周期 101 5 XTIMCLK 周期 110 6 XTIMCLK 周期 111 7 XTIMCLK 周期（默认）	这些位对增加指定区域内或外的连续访问，读或写，程序或数据空间时的XTIMCLK周期数进行了详细说明。XTIMCLK周期可以是0到7的数值。 在复位（XRS）时，默认值为7 个XTIMCLK周期（14 SYSCLKOUT周期）。
2:0	BANK	000 区0 001 保留 010 保留 011 保留 100 保留 101 保留 110 区6 111 区7（默认情况下，选中复位时）	这些位说明XINTF区域0区到7区中的哪些存储体有效。复位时，选择XINTF的7区域。

(1) 该寄存器是 EALLOW 保护。...

6.4 XREVISION 寄存器

图 9、XREVISION 寄存器



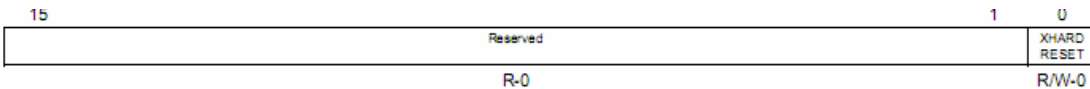
图例：R / W=读/写; R =只读;- N=复位后的值

表 11、XREVISION 寄存器说明

Bit	Field	Value	Description
15:0	REVISION		当前XINTF修正。用于内部使用/参考。仅用于测试目的。服从改变。

6.5 XRESET 寄存器

图 10、XRESET 寄存器



图例：R / W=读/写; R =只读;- N=复位后的值

表 12、XRESET 寄存器说明

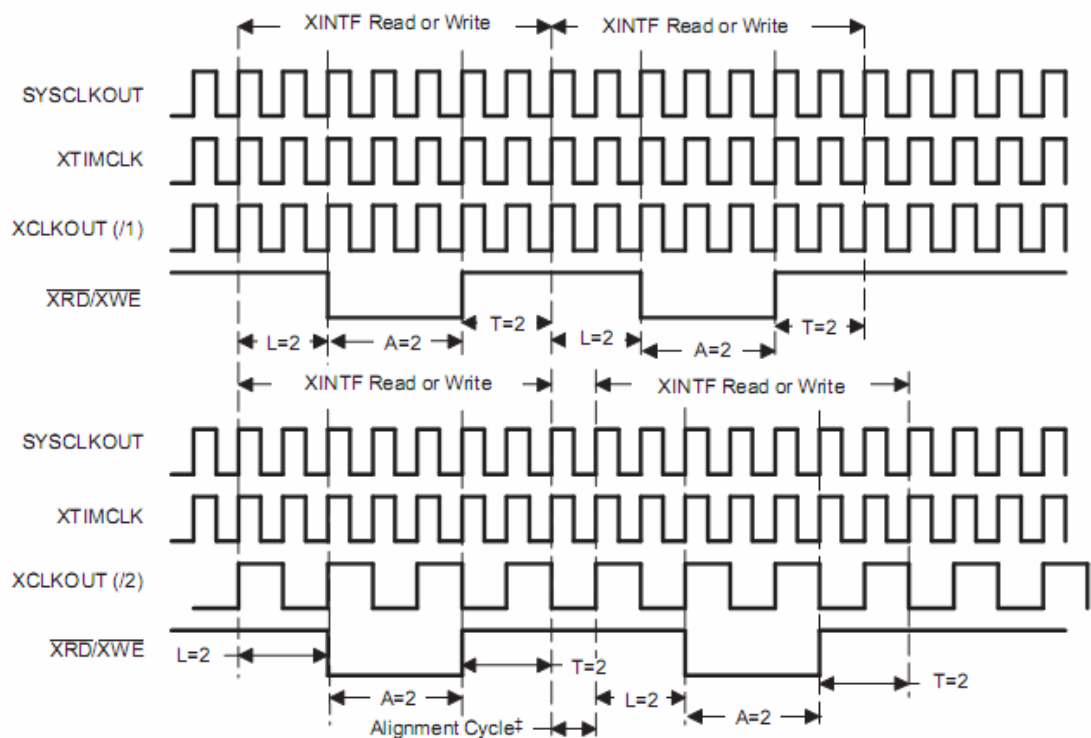
Bit	Field	Value	Description ⁽¹⁾
31	保留		
30	XHARDRESET	0 1	CPU检测到在DMA传输时XREADY信号被维持低电平情况下，执行强制复位。写0无影响。该位始终读回0。 XINTF强制复位。XTIMING, XBANK 和XINTCNF2寄存器将返回自己的默认状态，所有XINTF信号变为无效状态。任何挂起的访问将会丢失包括写缓冲区间中的数据。DMA的任何摊档条件将被释放。

(1) 该寄存器 EALLOW 保护。

7 信号说明

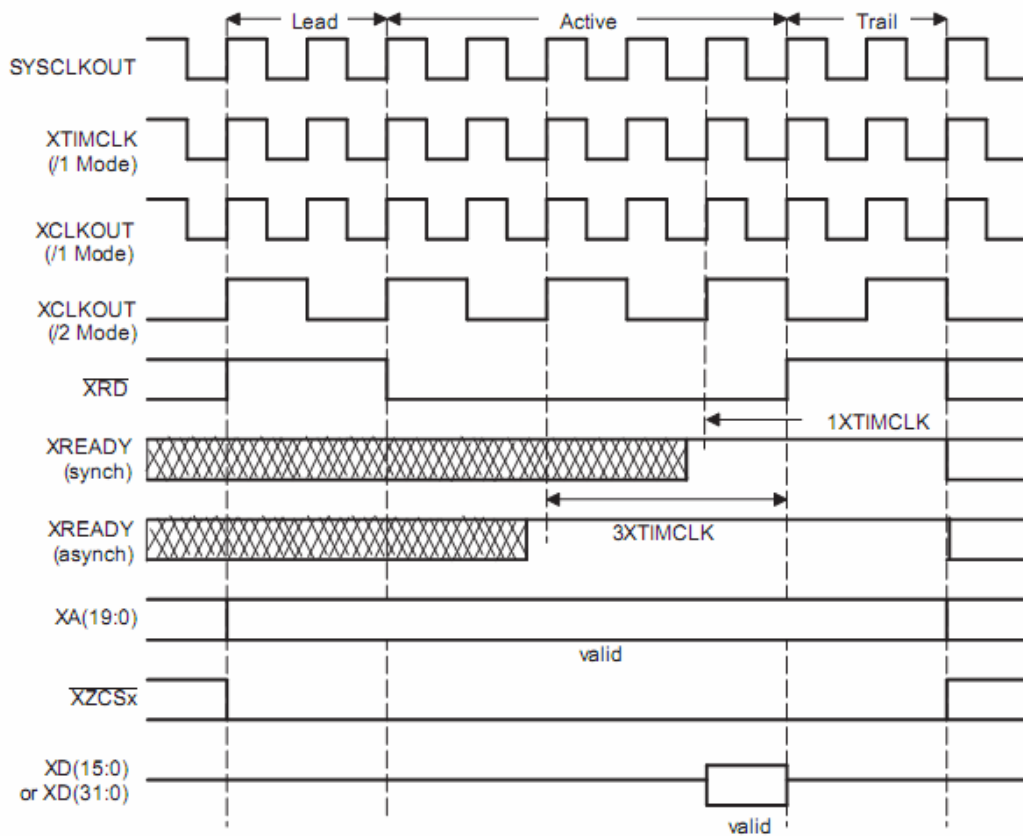
表 13 、 XINTF 信号说明

Name	Type	Description
XD(31:0)	I/O/Z	双向32位数据总线。16位的只有在XD（15:0）模式下使用。
XA(31:1)	O/Z	地址总线。在XCLKOUT的上升沿把地址放到总线上，直到下一次访问的到来。专用器件可能没有所有32个地址线。专用器件可查看特定数据手册。
XA0/ XWE1	O/Z	在16位数据模式，这个信号是最显著的地址线（XA0）。在32位数据模式，这个信号是低电平有效的写选通XWE1。 XWE1是与XWE0一起使用，正如在2.7节所示32位总线操作。
XCLKOUT	O/Z	来自XTIMCLK的单脉冲，用于片上和片外等待状态的产生及作为一个通用的时钟源。根据复位时的XINTCNF2寄存器中的CLKMODE位的值，XCLKOUT的频率和XTIMCLK频率相同或为其一半。 $XCLKOUT = XTIMCLK / 2$ $XTIMCLK = SYSCLKOUT / 2$
XWE0	O/Z	低电平有效的写选通。在16位模式下，在所有的总线模式和数据大小类型中，该信号拉为低电平。在32位模式下，如图5所示。通过XTIMINGx寄存器中的建立、激活和跟踪周期，对每区的写选通波形进行说明。
XRD	O/Z	低电平有效的读选通。所有总线模式和数据大小类型中，该信号拉为低电平。通过XTIMINGx寄存器中的建立、激活和跟踪周期，对每区的读选通波形进行说明。注：XRD和XWE0信号是相互排斥的。
XR/W	O/Z	读非写控制。高电平时，这个信号表明读周期有效，当低电平时，则表明写周期是有效的。这个信号通常是高电平。XR / W信号与XRD和XWE0信号有类似的功能。一般来说，用户选择使用XWE0和XWE1，因为它们更清楚和更容易使用。



- A X2TIMING=0, XRDLEAD/ XWRLEAD= 2, XRDACTIVE/ XWRACTIVE= 2,XRDTRAIL/ XWRTRAIL= 2
 B 队列周期。必须确保所有总线周期在 XCLKOUT 的上升沿。

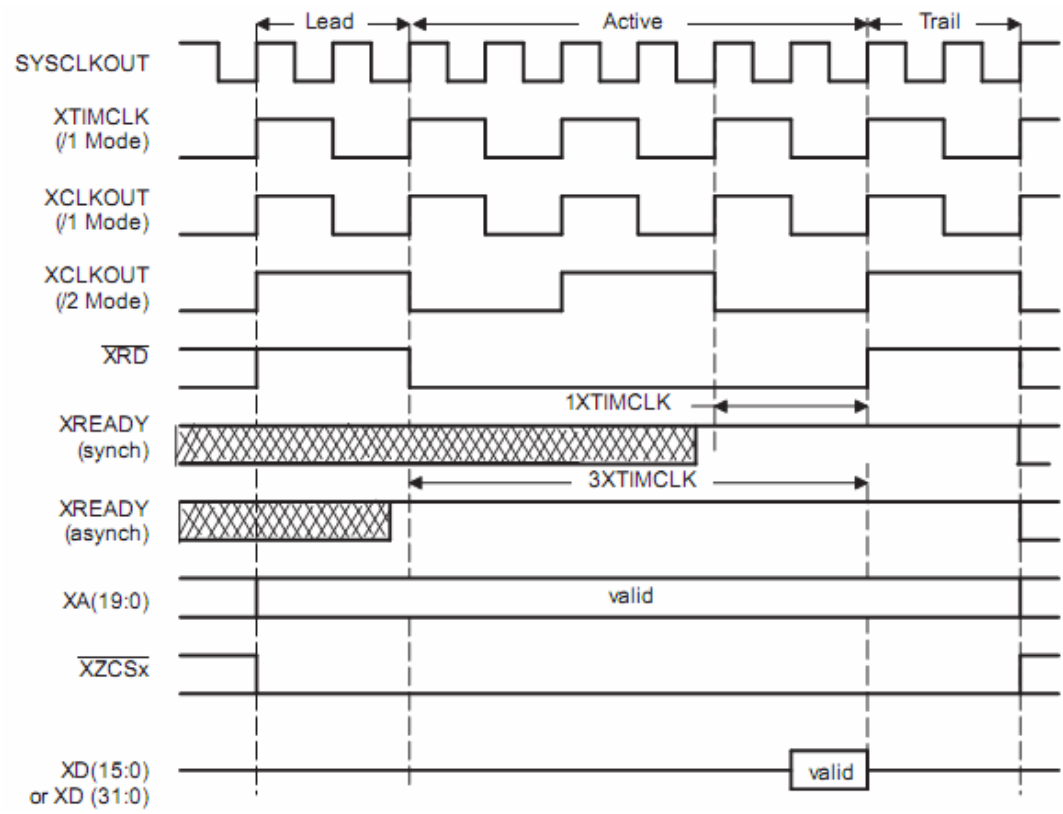
图 12、通用读周期（XTIMCLK= SYSCLKOUT 模式）



- A XRDLEAD=2,XRDACTIVE=4,XRDTRAIL=2

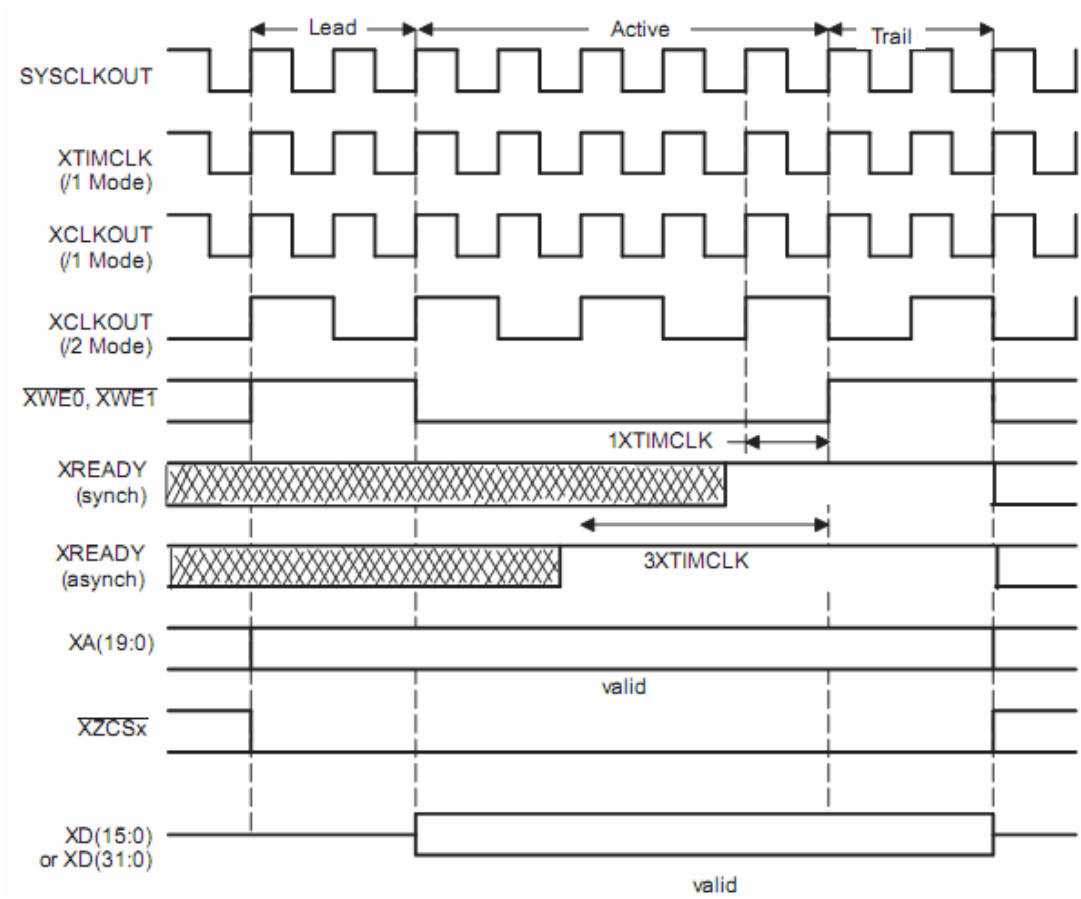
XREADY 信号可被同步采样，异步采样或忽略。如果是同步采样，在有效期结束前，然后 XREADY 信号必须满足一个 XTIMCLK 边沿的建立和保持时间。如果是异步采样，在有效期结束前，然后 XREADY 信号必须满足三个 XTIMCLK 边沿的建立和保持时间。如果 XREADY 在采样间隔为低电平，有效阶段增加一个 XTIMCLK 期间，在下一个 XTIMCLK 上升沿，XREADY 输入信号被采样。在采样间隔，XCLKOUT 不起作用。

图 13、通用读周期 (XTIMCLK=1/2 SYSCLKOUT 模式)



A XRDLEAD = 1, XRDACTIVE=3, XRDTRAIL= 1

图 14、通用的写周期 (XTIMCLK= SYSCLKOUT 模式)



- A XWRACTIVE = 2, XWRACTIVE=4, XWRTRAIL= 2
- B 如果建立和激活的时序参数设置足够低，它可能不会产生一个有效的 XREADY 信号。没有添加硬件检测。

附录 A 修订历史记录

此文件修改包括以下的技术变化。

表 14、修订

位置	编辑/新增/删除
Global	脚注中使用的术语“X2TIMCLK”，是无效的。它被“X2TIMING。”替换