5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

对于许多环境，I/O密集型工作负载可能会导致性能显著下降。每个I/O操作（入站或出站）都必须由VI拦截和处理，这增加了大量的平台资源开销。

SR-IOV提供了减少这些平台资源开销的工具SR-IOV的优点

·能够消除VI参与主要数据移动动作- DMA、存储器空间访问、中断处理等。消除VI拦截和处理每个I/O操作可以提供显著的

应用程序和平台性能改进。

·通过单根PCI管理器（SR-PCIM）控制SR-IOV资源配置和管理的标准化方法。

. 由于有多种实现选项-系统固件、VI、操作系统、I/O驱动程序等。 -SR-PCIM实现超出了本规范的范围。

·降低硬件要求和与在设备内潜在地供应大量I/O功能相关联的成本的能力。

·能够将SR-IOV与其他I/O虚拟化技术集成，例如地址转换服务（ATS）、地址转换和保护表（ATPT）技术以及中断重映射技术，以创建强大、完整的I/O虚拟化解决方案。

[图9-3](#bookmark1)显示了支持SR-IOV的平台示例。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0



SI

SI

SI

SI

SI

SI

虚拟化中间件

Sr-PCIM

|  |
| --- |
| 处理器 |
| 存储器 |

|  |  |  |
| --- | --- | --- |
| 翻译代理（TA） |  | 地址转换和保护表（ATPT） |
|  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | 根复合体（RC） |  |  |
|  | 根端口 | 根端口 |
|  | （RP） |  | （RP） |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | ATC |  |  | | --- | | PF0 |  |  |  |  | | --- | --- | --- | | VF 0，1 |  | VF 0，n |   PCIe设备 |

开关

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | ATC |  |  | | --- | | PF0 |  |  |  |  | | --- | --- | --- | | VF 0，1 |  | VF 0，n |   PCIe设备 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| |  | | --- | | ATC |  |  | | --- | | PF0 |  |  |  |  | | --- | --- | --- | | VF 0，1 |  | VF 0，n |   PCIe设备 |

A-0624A

图9-3使用SR-IOV和IOVEnabler的通用

SR-IOV通用平台配置由以下附加功能元素组成

·SR-PCIM-负责配置[SR-IOV扩展功能](#bookmark2)的软件，

物理功能和虚拟功能，以及相关错误事件和整体设备控制（如电源管理和热插拔服务）的处理。

·可选的转换代理（TA）-ATA是负责将PC Ie事务内的地址转换成相关联的平台物理地址的硬件或硬件和软件的组合。ATA可以

包含一个地址转换缓存（ATC），以加速传输表访问。ATA还可以支持

地址转换服务（ATS），其使得PCIe功能能够在对相关联的存储器的DMA访问之前获得地址转换。 有关ATS的优点和操作的更多详细信息，请参见第10章。

·可选的地址转换和保护表（ATPT）-ATPT包含由TA访问以处理PCIe请求（DMA读取、DMA写入或中断请求）的地址转换的集合。查看地址

翻译服务（[第10章）](#bookmark4)，用于其他详细信息。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

. 在PCIe中，中断被视为存储器写入操作。通过PCIe事务中包含的中断器标识符和地址的组合，中断可以被路由到任何中断器。

目标（例如，处理器核心）传送到相关联的I/O功能。

. DMA读取和写入请求通过PCIe传输中包含的路由ID和地址的组合进行转换。

·可选的地址转换高速缓存（ATC）-ATC可以存在于平台内的两个位置中-在TA内，其可以集成在RC内或位于RC上方-或者在PCIe设备内。在RC中，ATC使能

加速翻译查找。在器械内，ATC通过ATS技术填充。PCIe 表明它们包含转换地址的事务可以绕过平台的ATC，以便在不损害与ATPT技术相关联的益处的情况下提高性能。请参阅地址转换

服务（[第10章）](#bookmark5)，用于其他细节。

·可选的访问控制服务（ACS）-ACS在PCI Express拓扑中定义了一组控制点， 确定TLP应正常路由、阻塞还是重定向。在支持SR-IOV的系统中，ACS可以用于防止分配给VI或不同SI的设备功能彼此通信或与对等设备通信。重定向可以允许转换代理转换上游存储器TLP

在做出对等转发决定之前，可选的ACS P2P出口控制可以提供选择性阻止。 ACS与ATS相互作用。有关[9.3.7.6](#bookmark6)更多详细信息，请参见www.example.com部分。

·物理功能（PF）-PF是支持[SR-IOV扩展能力](#bookmark7)并且可由SR-PCIM、VI或SI访问的PCIe功能。

·虚拟功能（VF）-VF是可由SI直接访问的“轻量级”PCIe功能。

. 最低限度地，与功能的主要数据移动相关联的资源可用于SI。

配置资源应限制到可信的软件组件，如VI或SR-PCIM。

. VF可以由不同的SI串行共享（即，可以将VF分配给一个SI，然后重置并分配给另一个SI）。

. VF可以可选地从一个PF迁移到另一个PF。迁移过程本身不在本规范的范围之内，但是可以通过本规范中定义的配置控制来实现。

规范.

·与PF相关联的所有VF必须是与PF相同的设备类型（例如，相同的网络设备类型或相同的存储设备类型。）

要比较PCIe设备与支持PCIeSR-IOV的设备，请查看以下一组图。[图](#bookmark8)[9-4](#bookmark9)显示了一个符合PCIe标准的设备示例。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | ATC1 |   功能0   |  |  | | --- | --- | | |  | | --- | | 物理 资源1 | |   配置  资源  内部路由   |  | | --- | | ATC2 |   功能1   |  |  | | --- | --- | | |  | | --- | | 物理 资源2 | |   PCIe端口  PCIe设备   |  | | --- | | ATC3 |   功能2   |  |  | | --- | --- | | |  | | --- | | 物理 资源3 | | |

粤ICP备16062555号-1

图9-4多功能设备示例

该图示出了具有以下特性的示例多功能PCIe设备

·PCIe设备共享公共PCIe链路。所有功能共享的链路和PCIe功能通过功能0进行管理。

◦ 虽然该图仅示出了三个功能，但是通过使用替代路由标识符（ARI）能力，PCIe设备可以支持多达256个功能。

◦ 所有功能都使用通过PCI枚举过程捕获的单个总线号。

· 在该示例中，每个PCIe功能支持ATS能力，并且因此具有相关联的ATC以管理ATS获得的转换地址。

· 每个PCIe功能具有一组唯一的物理资源，包括单独的配置空间和BAR。

· 每个PCIe功能可以分配给一个SI。为了防止一个SI影响另一个SI，所有PCIe配置操作都应该被VI拦截和处理。

如该图所示，硬件资源随着所供应的功能的数量而缩放。取决于 由于设备的复杂性和大小，每个功能的增量成本将有所不同。为了减少硬件成本的增加，可以使用SR-IOV来构建一个设备，以支持单个PF和多个VF，如[图9-5所示。](#bookmark10)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PF 0   |  | | --- | | ATC |  |  |  |  | | --- | --- | --- | |  | 配置资源 |  | | |  | | --- | | 物理 资源 | | | |   VF 0.1   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |   内部路由  PCIe端口  VF 0.2   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |   支持PCIe SR-IOV的设备  VF 0，N   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | | |

A-0626A

图9-5示例SR-IOV单PF功能器件

[图9-5中的例子](#bookmark10)说明了一个带有NVF的PF。需要注意的主要观察结果：

·PF是符合PCIe的。

◦ 在初始和常规复位后，支持本规范中定义的SR-IOV功能的PCIe FuneTM应禁用SR-IOV功能。

◦ 要发现PF及其关联VF支持的页面大小，请使用

配置字段被读取。有关如何使用此字段对齐系统页边界上的PF或VF内存空间孔径的更多信息，请参见[第9.2.1.1.1节。](#bookmark11)

· PF命名**PFM**表示功能编号M的PF。

·VF命名**VFM，N**表示与PF M相关联的第N个VF。VF从1开始编号，因此与PF M相关联的第一VF是VF M，1。

· 每个VF与PF共享多个公共配置空间字段这些田地

适用于所有VF，并通过单个PF进行控制。共享减少了实现每个VF所需的硬件资源。

◦ AVF使用与PF相同的配置机制和报头类型。

◦ 与给定PF相关联的所有VF共享VF BAR集（参见[第4节9.3.3.1](#bookmark12)）并共享VF存储器

SR-IOV扩展功能（参见第9.3.3.3.4节）中的空间使能（MSE）位控制对VF存储器空间的访问。也就是说，如果VF MSE位为清除，则禁用为所有VF分配的内存映射空间。

◦ InitialVF和TotalVF字段（请参见www.example.com[节9.3.3.5](#bookmark14)和www.example.com[节9.3.3.6）](#bookmark15)用于发现可与PF关联的VF的最大数量。

◦ 如果器械不支持VF迁移，TotalVF和InitialVF应包含相同的值。如果设备支持VF迁移，当读取TotalVF时，PF必须返回可分配给PF的VF数量。对于此类设备，当读取InitialVF时，PF必须返回初始值 分配给PF的VF数量。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

· 每个功能、PF和VF被分配唯一的路由ID。每个PF的路由ID的构造如下：

部分2.2.4.2。每个VF的路由ID使用其相关联的PF的路由ID和该PF[的SR-IOV扩展能力中的字段来确定。](#bookmark16)

·假设所有PCIe和SR-IOV配置访问都是通过受信任的软件组件（诸如VI或SR-PCIM）进行的。

· 每个VF包含递送特定于功能的服务所需的非共享物理资源集（例如， 资源，如工作队列、数据缓冲区等）这些资源可以由SI直接访问，而不需要VI或SR-PCIM干预。

·一个或多个VF可以被分配给每个SI。分配策略不在本规范的范围内。

·虽然该示例示出了PF内的单个ATC，但是任何ATC的存在是可选的。此外，本规范不排除支持设备内每个VF的ATC的实现。

· 内部路由是特定于实现的。

·虽然关于PF操作存在许多潜在的使用模型，但是常见的使用模型是使用PF来严格地在VI的控制下引导设备或平台。一旦[SR-IOV扩展功能](#bookmark17)被

配置为使VF能够被分配给各个SI，PF承担更多的监督角色。例如，PF可以用于管理设备特定功能，诸如对每个VF的内部资源分配、对诸如PCIe链路或功能特定链路（例如，网络或存储链路）等。这些策略、管理和资源分配操作不在本文的范围之内。

规范.

另一个使用模型示例如[图9-6所示。](#bookmark18)在该示例中，设备支持多个PF，每个PF具有其自己的VF集合。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | VF 0.1 |  |  | | --- | | 物理 资源 |  |  |  |  | | --- | --- | --- | |  | 配置资源 |  | | |  | | --- | | 物理 资源 | | | |  |  | | --- | | ATC |   PF 0  内部路由  VF 0.2   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |   内部路由   |  |  |  | | --- | --- | --- | |  | 配置资源 |  | | PFM 物理  资源 | | |  |  | | --- | |  |   PCIe端口   |  | | --- | | ATC |   VF 0.3   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |   VF M，1   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |   VF M，2   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |   内部路由  支持PCIe SR-IOV的设备  VF M，3   |  |  | | --- | --- | | |  | | --- | | 物理 资源 | | |

粤ICP备16066271号-1

图9-6示例SR-IOV支持多PF的器件

需要注意的主要观察结果

· 每个PF可以被分配零个或多个VF。对于器械内的所有PF，每个PF的VF数量不需要相同。

·ARI扩展能力使得功能能够被分配给功能组，并且可以配置定义功能组仲裁。PF和VF可以被分配给功能G组并且利用相关联的仲裁能力。然而，在每个职能组

具体实施。

· PF和VF之间的内部路由是特定于实现的。

· 对于一些使用模型，所有PF可以是相同的设备类型;（例如，所有PF都提供相同的网络设备，或者都提供相同的存储设备功能。）对于其他使用模型，每个PF可以表示不同的 设备类型;（例如，在[图9-6中，](#bookmark18)一个PF可能代表一个网络设备，而另一个PF代表一个

加密设备）。

◦ 在re是设备类型之间的使用模型依赖关系的情况下，例如对于每个VF，

即网络设备类型，每个SI还需要加密设备类型的VF。 [SR-IOV](#bookmark19)[扩展功能](#bookmark20)提供了一种指示这些依赖关系的方法。过去的政策

构造这些依赖关系以及将VF的依赖集分配给给定SI都在本说明书的范围之外。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

如前面的示例所示，PF和VF的数量可以基于使用模型要求而变化。为了支持广泛的选项，SR-IOV设备可以支持以下数量和组合的PF和VF：

· 使用替代路由标识符（ARI）功能，设备可以支持多达256个PF。函数号分配是特定于实现的，并且在整个256函数号空间中可能是稀疏的。

·PF只能与设备捕获的总线号相关联[，如图9-7所示。](#bookmark21)

·SR-IOV设备可以消耗多于一个总线号。VF可以与设备的总线号范围内的任何总线号相关联-捕获的总线号加上y个

软件参见[第9.2.1.2f](#bookmark22)或详细信息。

. 使用多个总线号使设备能够支持非常大数量的VF-多达路由ID空间的大小，最小为用于标识中间总线的位。

. 如果软件没有配置足够的附加总线号，则为附加总线号实现的VF可能不可见。

执行说明

功能共址

ARI扩展功能使设备能够支持多达256个功能-任何类型的功能、PF或VF

组合-与捕获的总线号相关联。如果使用模型不需要超过256个

强烈建议函数、实现将所有函数、PF和VF放在捕获的总线号中，并且不需要额外的总线号来访问VF。



5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | |  | | --- | | 物理 资源 | | | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |  |  |  |  | | --- | --- | --- | |  | 配置资源 |  | | |  | | --- | | 物理 资源 | | | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | | |



|  |
| --- |
| 巴士N  配置资源 |

|  |
| --- |
| 总线N+1、N+2、. |



|  |  |
| --- | --- |
| |  | | --- | | 物理 资源 | |

|  |  |
| --- | --- |
| |  | | --- | | 物理 资源 | |

PCIe SR- IOV

能力的设备

内部路由

内部路由

内部路由

|  |
| --- |
| ATC |

|  |
| --- |
| ATC |

PCIe端口

VF255.2

VF255.3

VF255.1

VF0.1

VF0.2

VF0.3

PF 255

PF 0

粤ICP备16066888号-1

图9-7示例SR-IOV器件，具有多个总线编号

在最后一个示例中，如[图9-8所示](#bookmark23)，设备实现可以混合任意数量的函数、PF和VF。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | |  | | --- | | 物理 资源 | | | |  |  |  |  | | --- | --- | --- | |  | 配置资源 |  | | 物理 资源 | | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | |  |  |  |  | | --- | --- | --- | |  | 配置资源 |  | | |  | | --- | | 物理 资源 | | | |  |  |  | | --- | --- | | |  | | --- | | 物理 资源 | | |



|  |
| --- |
| 巴士N  配置资源 |

|  |
| --- |
| 总线N+1、N+2、. |



|  |  |
| --- | --- |
| |  | | --- | | 物理 资源 | |

|  |  |
| --- | --- |
| |  | | --- | | 物理 资源 | |



PCIe SR- IOV

能力的设备

内部路由

内部路由

内部路由

|  |
| --- |
| ATC |

|  |
| --- |
| ATC |

|  |
| --- |
| ATC |

PCIe端口

功能0

VF 15.3

VF 15.2

VF 15.1

VF 1.3

VF 1，2

VF 1.1

PF 15

PF 1

A-0629A

图9-8混合功能类型的SR-IOV设备

需要注意的主要观察结果

· 每个设备必须包含一个功能0。函数0可以是PF（即，它可以包括SR-IOV扩展能力）。

·任何功能组合都可以与捕获的总线号相关联

◦ 非VF只能与捕获的总线编号相关联。

· 如果支持ARI扩展功能，则函数可以签名为函数组。转让 策略超出了本规范的范围。如果不支持ARI扩展功能，功能仍然可以使用第6.3.3.4节中定义的功能仲裁功能。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

9.1.1 PCI技术互操作性

建立明确的互操作性需求对于任何技术的成功都是至关重要的。 为此，PCI-SIG I/O虚拟化规范被组织为最大化兼容实现的互操作性潜力。

从概念上讲，这可以看作是一组同心圆，定义了功能如何分层以构建具有IOV能力的组件，如[图9-9所示。](#bookmark24)

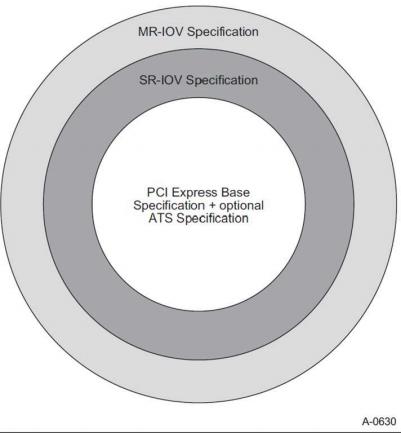


图9-9 I/O虚拟化互操作

需要注意的主要观察结果

I/O虚拟化扩展的核心是建立在此规范之上。所有IOV实现都必须

兼容[PCIe-1.1]或更高版本。 在适用的情况下，IOV规范注意到这些版本之间的相关增量。

. 没有一个IOV规范触及物理层。

. SR-IOV不涉及本规范中指定的数据链路或事务层。. [MR-IOV]不涉及本规范中指定的事务层。

. 所有I/O虚拟化功能都通过PCI Express扩展配置空间中实现的新PCI Express功能进行通信。

. I/O虚拟化规范对PCI或PCI-X规范没有影响。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

. 一个层次结构可以由PCI Express和PCI总线到PCI/PCI-X桥的混合组成。

PCI接口到PCI/PCI-X桥和PCI/PCI-X设备可以由多个SI串行共享。

·ATS定义了适用于任何功能的可选功能。ATS可以在SR-IOV组件中得到支持。

·为了实现SR-IOV设备，SR-IOV要求设备完全符合[PCIe]。 . 层次结构可以由SR-IOV和非SR-IOV组件的混合组成。例如是

层级可以包含SR-IOV和非SR-IOV端点设备的任何混合。

9.2 SR-IOV初始化和资源分配

9.2.1SR-IOV资源发现

以下部分描述软件如何确定设备具有SR-IOV能力并且随后通过虚拟功能配置空间来识别VF资源。

<9.2.1.1>配置SR-IOV功能

本节介绍在启用PF的IOV功能之前必须配置的字段。通过设置PF的VF使能位来使能[VF（参见第9.3.3节）。3.1）](#bookmark26)在SR-IOV扩展能力中。

[NumVF](#bookmark27)字段（参见[第www.example.com节9.3.3.7）](#bookmark28)定义了在相关PF中设置VF启用时启用的VF数量。

9.2.1.1.1配置[VF BAR](#bookmark31)机制

本节介绍如何配置[VF BAR](#bookmark32)以映射内存空间。 VF不支持I/O空间，因此[VF](#bookmark33)[BAR](#bookmark34)不应指示I/O空间。

系统[页面大小](#bookmark35)字段（请参见[第www.example.com节9.3.3.13](#bookmark36)）定义当启用PF的IOV功能时系统将用于映射VF的PCIe内存地址的页面大小。PF使用[系统页面大小](#bookmark37)字段将每个VF BAR定义的内存空间孔径与系统页面边界对齐。 为[系统页面大小](#bookmark39)选择的值必须是SR-IOV扩展功能中支持的页面大小之一（参见[www.example.com部分9.3.3.12）](#bookmark40)。

VF BAR的行为与普通PCI内存空间BAR相同（参见第7.5.1.2.1节），除了[VF BAR](#bookmark42)描述每个VF的孔径，而PCI BAR描述单个功能的孔径。的属性

如果实现了[VF可调整大小的BAR扩展能力](#bookmark44)（参见www.example.com[部分9.3.7.5）](#bookmark45)，则VF BAR中的某些位会受到该能力的影响。

·第7.5.1.2.1节中描述的用于确定函数的BAR的存储器孔径的行为适用

每一个[VF](#bookmark46)Bar 也就是说，每个VF BAR所需的存储器孔径[的大小](#bookmark47)可以通过写入所有[“1“然后读取VF BAR来确定。](#bookmark48)回读结果必须按照第

7.5.1.2.1.

·用于分配与第一VF相关联的每个BAR的起始存储器空间地址的行为也如第7.5.1.2.1节中所述。也就是说，写入每个[VF BAR](#bookmark49)的地址由设备用于 第一个VF的起始地址。

·[VF BAR](#bookmark50)与第7.5.1.2.1节中描述的BAR之间的差异在于，对于每个[VF BAR，](#bookmark51)与第二和更高VF相关联的存储器空间是从第一VF的起始地址和第二VF的起始地址导出的。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

记忆空间光圈对于任何给定的VF v，根据以下公式计算用于任何实现的BAR b）的其存储器空间端口的起始地址：

BARb VFv起始地址=[VF BAR](#bookmark52)b+（v- 1）x（[VF BAR](#bookmark53)数据大小）

其中[VF BAR](#bookmark54)b孔径尺寸是[VF BAR](#bookmark55)b的尺寸，如通过如在第www.example.com节中描述的通常BAR探测算法[9.3.3.14确定的。](#bookmark56)

在[VF启用](#bookmark57)和[VF MSE](#bookmark58)均已设置之前，不会启用VF存储空间（参见[第9.3.3.3.1](#bookmark59)和

[第9.3.3.3.4节](#bookmark60)）。请注意，更改[系统页面大小](#bookmark61)（请参阅www.example.com[部分9.3.3.13）](#bookmark62)可能会影响[VF BAR](#bookmark63)的孔径大小。

[图9-10](#bookmark64)显示了PF和VF内存空间孔径的示例。

PCIe存储空间

PF配置空间

|  |  |
| --- | --- |
|  |  |
| BAR 0（RW） | |
|  |  |
| VF BAR 0（RW） | |
|  |  |

(

|  |
| --- |
|  |
| PF BAR 0  存储器  空间光圈 |
|  |
| VF1BAR0存储空间孔径 |
| VF 2BAR0存储空间孔径 |
|  |
| VFNBAR0存储空间孔径 |
|  |

粤ICP备06036661号-1

图9-10单个BAR器件的BAR空间示例

<9.2.1.2>VFDiscovery

SR-IOV扩展能力中的[第一VF偏移](#bookmark65)和[VF步幅](#bookmark66)字段是16位路由ID偏移。这些偏移量用于计算具有以下限制的VF的路由ID

·PF中NumVF的值（[第www.example.com节9.3.3.7）](#bookmark68)可能会影响该PF的[第一VF偏移设置](#bookmark69)[（第www.example.com节9.3.3.9）](#bookmark70)和[VF](#bookmark71)[步幅](#bookmark72)[（第www.example.com节9.3.3.10）](#bookmark73)中的值。

·设备的最低编号PF（例如PF 0）中的ARI容量层级（第9.3.3.3.5节）中的值可能影响设备的所有PF中的[第一VF偏移t](#bookmark76)和[VF步幅](#bookmark77)中的值。

·PF的[NumVF](#bookmark78)仅在PF的[VF启用](#bookmark79)[（第9.3.3.3.1节）](#bookmark80)

·[ARI能力层级](#bookmark81)[（第9.3.3.3.5节）仅](#bookmark82)当[设备](#bookmark83)的所有PF中的VF启用为清除时才可更改。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

执行说明

NumVF和ARI能力层次

在配置[NumVF](#bookmark84)和[ARI能力层级（](#bookmark85)在适用的情况下）之后，软件可以读取[第一VF偏移](#bookmark86)和[VF](#bookmark87)[步幅](#bookmark88)以确定PF的VF将消耗多少总线编号。附加的总线编号（如果有的话）在[VF启用](#bookmark89)被设置之前不会被实际使用。



[表9-1](#bookmark90)描述了用于确定与每个VF关联的路由ID的算法。

表9-1 VF路由ID算法

|  |  |
| --- | --- |
| VF编号 | VF路由ID |
| VF 1 | (PF路由ID+[第一VF偏移）](#bookmark91)模216 |
| VF 2 | (PF路由ID +[第一VF偏移](#bookmark92)+[VF步幅）](#bookmark93)模216 |
| VF 3 | (PF路由ID +[第一VF偏移](#bookmark94)+2 \*[VF步幅）](#bookmark95) 模216 |
| … | … |
| VF N | (PF路由ID +[第一VF偏移](#bookmark96)+（N-1）\*[VF步幅）](#bookmark97)模216 |
| … | … |
| VF[NumVF](#bookmark98)（最后一个） | (PF路由ID +[第一VF偏移](#bookmark99)+（[NumVFs](#bookmark100)-1）\*[VF步幅）](#bookmark101)模216 |

在这个路由ID计算中使用的所有算法都是16位无符号丢弃所有进位。

所有VF和PF必须具有不同的路由ID。任何PF或VF的路由ID不得与任何其他PF或VF的路由ID重叠，因为[设备的](#bookmark102)所有PF上的NumVF设置有效。

[VF步幅](#bookmark103)和[第一VF偏移](#bookmark104)是常量。除本节前面所述外，它们的值可能不受设备的此功能或其他功能设置的影响。

VF可以驻留在与相关联的PF不同的总线号上。 如果例如[第一VF偏移](#bookmark105)具有值0100h，则这可以发生。 VF不应位于数值小于其相关PF的母线编号上。一个VF， 位于与其相关联PF相同的总线号上的设备编号不应位于数字上

比PF158小。

SR-IOVRCiEP设备的VF与和它们的PF相同的根复合体事件收集器（如果有的话）相关联。根复合体的根复合体事件收集器端点关联扩展功能中未报告此类VF 事件收集器。

根据第www.example.com节2.2.6.2，与上游端口相关联的具有SR-IOV功能的设备从任何类型0配置写入请求捕获总线号。支持SR-IOV的设备不从任何类型1捕获总线编号

配置写入请求。具有SR-IOV能力的RCiEP使用实现特定的机制来分配它们的总线号。

交换机对总线号的处理与基本PCIe相同。在基本PCIe中，交换机发送所有配置

向设备请求二级总线号（含）到下级总线号（含）范围内的请求。1型 寻址次级总线号的配置请求被转换为0型配置请求，



158. SR-IOV紧接在下游端口之下的设备总是具有设备编号0，并且因此总是满足该条件。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

针对次级总线号（不含）和次级总线号（含）之间的总线号的类型1配置请求作为类型1配置请求转发到设备。

注意：公交车号码是受约束的资源。强烈建议设备避免在总线号使用中留下“漏洞”，以避免浪费总线号。

所有PF必须位于设备

执行说明

跨越多个总线编号的VF

作为示例，考虑支持单个PF的SR-IOV设备。最初，0个PF0可见。 软件设置[ARI](#bookmark106)[功能层次结构](#bookmark107)。根据[SR-IOV扩展能力，](#bookmark108)它确定[：InitialVF](#bookmark109)为600，[FirstVF Offset](#bookmark110)为1，[VF](#bookmark111)[Stride](#bookmark112)为1。

· 如果软件将NumVF设置在[0 ... 255]范围内，则设备使用单个总线编号。

· 如果软件将NumVF设置在[256 ... 511]范围内，则设备使用两个总线编号。

· 如果软件将NumVF设置在[5 12 ... 600]范围内，则设备使用三个总线编号。

PF 0和VF 0，1到VF 0，255始终位于第一个（捕获的）总线编号上。 VF0，256到VF0，511总是在第二总线号上（捕获的总线号加1）。VF0，512至VF 0，600始终位于第三个公交车号码上（捕获的公交车号码加2）。



软件应配置开关辅助和次级总线编号字段，以将足够的总线编号路由到设备。如果没有足够的总线号可用，软件应通过在启用SR-IOV之前不启用SR-IOV和/或减少设备的部分或所有PF的NumVF来减少设备的总线号要求。

在PFn中设置VF启用后，设备必须启用VFn，1至VFn，m（包括），其中m是InitialVF和NumVF中的较小值。接收到以位于

捕获的总线号必须正常处理请求。接收到以不在捕获的总线号上的已启用VF为目标的类型1配置请求的设备必须正常处理该请求。接收针对设备捕获的总线号的类型1配置请求的设备必须遵循处理不支持的

请求。此外，如果[VF MSE](#bookmark120)被设置，则每个启用的VF必须响应于寻址与该VF相关联的存储器空间的PCIe存储器事务。

未启用的功能（即，PCIExpress结构中不存在用于m）以上VF的功能。 根据第2.3.1节，寻址不存在的函数将导致返回未验证的请求（UR）。这包括附加总线编号上的功能。

执行说明

具有PF和开关功能的多功能设备

SR-IOV设备可以消耗多个总线号。除了第一辆巴士外，其他巴士的编号如下：

连续的并且紧跟在分配给设备的第一个BUS号码之后。如果SR-IOV设备还包含PCI-PCI桥接器（具有类型1配置空间报头），则在以下情况下必须考虑SR-IOV使用情况：

为这些网桥编程辅助总线号。 软件应该首先确定VF使用的最后一个总线号，然后配置任何共址网桥使用高于该值的总线号。



5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.2.1.3>函数依赖项列表

PCI设备在功能之间可能具有特定于供应商的依赖性。例如，函数0和1可以提供用于控制相同底层硬件的不同机制。在这种情况下，设备编程模型可能要求将这些依赖函数作为一个集合分配给SI。

函数依赖项列表用于描述这些依赖项（或指示不存在函数依赖项）。 软件应将PF和VF分配给SI，以满足依赖性。

参见[第9.3.3.8f节](#bookmark122)或详细信息。

<9.2.1.4>网站地图

如果分配了中断资源，PF和VF支持MSI、MSI-X中断或两者。 VF不应执行INTx。中断在[第9.5节中描述。](#bookmark123)

9.2.2 SR-IOV复位机制

本节介绍基于PCI定义的重置机制如何影响支持SR-IOV的设备。它还描述了用于重置单个VF和单个PF及其相关VF的机制。

<9.2.2.1>SR-IOV常规复位

按照第6.6.1节中的规则，对支持SR-IOV的器械进行常规复位应使所有功能（包括PF和VF）复位至其原始通电状态。[第9.3节描述](#bookmark124)了字段的行为

定义了

注：常规复位清除PF中的[VF E使能](#bookmark125)。因此，VF在常规复位之后不再存在。

<9.2.2.2>针对VF的FLR

VF必须支持功能级别复位（FLR）。

注：软件可使用FLR重置VF。VF的FLR影响 VF的状态，但不影响其在PCI中的

配置空间或PCI总线地址空间。VF BARn值（参见[第9.3.3.14](#bookmark126)）和[VF MSE](#bookmark127)（参见[第](#bookmark128) [9.3.3.3.4](#bookmark129)），并且[9.3.7.5](#bookmark130)VF可调整BAR能力值（见www.example.com节）不受向VF发布的FLR的影响。

以PF为目标<9.2.2.3>

PF必须支持FLR。

FLR到PF将重置PF状态以及SR-IOV扩展功能，包括[VF启用](#bookmark131)，这意味着VF不再存在。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

9.2.3 IOV重新初始化和重新分配

如果[VF Enable（VF启用](#bookmark132)）在设置后被清除，则与PF相关的所有VF都不再存在，并且必须不再

发出PCIe事务或响应配置空间或存储器空间访问。VF[使能](#bookmark133)被清除后，VF不得保留任何状态（包括粘性位）。

9.2.4 VF迁移

VF迁移是[ MR-IOV ]中的可选功能。不支持**MR-IOV扩展功能的设备**不支持VF迁移功能。

在多根系统中，VF可以在虚拟层次结构之间迁移。

要进行VF迁移，[必须同时设置“VF迁移能力”](#bookmark135)和“[VF迁移启用](#bookmark136)”。[支持VF迁移](#bookmark137)

向SR-PCIM指示VF迁移硬件存在并且多根PCI管理器（MR-PCIM）已启用其使用。VF迁移的硬件支持是可选的。 用于VF迁移的SR-PCIM支持也是可选的。

[VF迁移启用](#bookmark138)向设备硬件和MR-PCIM指示SR-PCIM也已启用VF迁移。支持VF迁移对SR-PCIM提出了以下要求

·需要确定VF是活动的、休眠的还是非活动的。活动的VF可供单根（SR）使用。默认的VF可以由SR配置，但不会发出事务。处于非活动状态的VF SR无法使用。

·需要参与**移民**行动。[Migrate In](#bookmark139)操作用于向SR提供 VFA[移民](#bookmark139)

操作由MR-PCIM发起。SR-PCIM可以接受[Migrate In](#bookmark139)操作并将VF移动到[Active.Available](#bookmark141)状态。

·需要参与**迁出**行动。Migrate[Out](#bookmark140)操作用于请求从SR的使用中优雅地移除活动VF。SR-PCIM可以接受[移出](#bookmark140)并将VF移动到

[不活动。不可用](#bookmark142)状态。

·需要处理**移民在保留ractions**。这是当MR-PCIM撤回对SR的VF提议并且VF移回[非活动](#bookmark143)不可用状态时。

<9.2.4.1>初始VF状态

本节介绍[VF迁移状态数组中的初始值](#bookmark145)（参见[第9.3.3.15.1节）](#bookmark146)。

如果[InitialVF](#bookmark147)[（第www.example.com节9.3.3.5）](#bookmark148)为非零，则VF 1至VFInitialVF处于活动[可用](#bookmark149)状态。如果[TotalVF](#bookmark150)[（参见](#bookmark151)[9.3.3.6](#bookmark152)）大于[InitialVF，则](#bookmark153)VFInitialVF+第1粗略VFTotalVF处于[非活动不可用](#bookmark154)状态。如果[VF迁移](#bookmark155)[启用](#bookmark156)[（第9.3.3.3.2节）](#bookmark157)被清除，则不使用高于InitialVF的VF。

如果[InitialVFs](#bookmark159)为0，则没有VF处于[Active.Available](#bookmark160)状态。如果[TotalVF](#bookmark161)等于[InitialVF，则](#bookmark162)所有VF都处于[活动状态。可用](#bookmark163)

状态如果[TotalVFs](#bookmark164)为0，则没有VF与此PF相关联，并且没有[VF迁移状态数组。](#bookmark165) [图9-11](#bookmark166)描述了这种初始VF状态。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

VF1

活动。可用VF

VF2



初始VF

VF n



VF n+1

不活动。不可用的VF

总VF

VF m

粤ICP备16036662号-1

图9-11初始[VF迁移状态数组](#bookmark167)

<9.2.4.2>[VF迁移状态](#bookmark168)转换

VF迁移遵循[图9-12所示的状态图。](#bookmark169)所示的状态值包含在与VF相关联的VF状态数组条目中。由实线指示的状态转换由MR-PCIM发起。状态转换

由点线和虚线指示的由SR-PCIM发起。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| VF非活动（SR不可用  SR：设置  非活动 .不可用  00b  VF使能和VF>InitialVF  MR：请求移入  SR：设置VF  移徙身份 | | | | | SR：完成迁移 | VF激活（SR完全可用  10b  VF使能且VF≤ InitialVF  11b  活性  .MigrateOut  MR：收回移出  SR：设置VF  移徙身份  MR：请求移出  SR：设置VF  移徙身份  SR：设置  活性  .可用 |
| SR：清除VF启用  SR：清除VF  SR：清除VF启用  VF禁用  使  SR：清除VF启用 |
|  |  |  |  |  |
| MR：收回移入  SR：设置VF  移徙身份 | | | | |
| Dormont.迁移  01b | | | | |
| SR：激活  SR：停用 |
| VF Dorsal（对SR有反应） | | | | |

粤ICP备06033663号-1

图9-12[VF迁移状态](#bookmark170)图

SR-PCIM通过向[VF迁移状态数组写入新值来启动状态转换。](#bookmark171)设备忽略写入

当SR-PCIM尝试启动[表9-2](#bookmark172)以外的任何状态转换时，将发生事务和无状态转换

.

表9-2 SR-IOV VF迁移状态表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 当前[VF启用](#bookmark173) | 当前VF状态 | 写入[VF使能](#bookmark174) | 写入VF状态 | 意义 |
| 1 | Dormant.MigrateIn | 1 | [Active.Available](#bookmark175) | SR激活VF。 |
| 1 | [Active.Available](#bookmark176) | 1 | Dormant.MigrateIn | SR停用VF。 |
| 1 | Active.MigrateOut | 1 | [Inactive.Unavailable](#bookmark177) | SR完成[移出。](#bookmark140) |
| 1 | 任何 | 0 | 任何 | SR禁用所有VF。 |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 当前[VF启用](#bookmark178) | 当前VF状态 | 写入[VF使能](#bookmark179) | 写入VF状态 | 意义 |
|  |
| 0 | 任何 | 1 | 任何 | SR启用所有VF。  VF转换为[Active.Available](#bookmark180)或  [不活动。](#bookmark181)根据VF编号和[InitialVF不可用。](#bookmark182) |

执行说明

软件状态迁移变更检测

SR-PCIM通常需要通过在写入VF迁移状态后重新读取VF迁移状态来验证状态更改是否生效。



处于[非活动.不可用状态的VF](#bookmark184)不能被软件以任何方式使用。以非活动VF为目标的请求接收

不支持的请求（UR）。 在转换到非活动或休眠状态的100 ms内，设备必须确保不会使用指定的路由ID发出新的交易。

MR-PCIM通过使用[ MR-IOV ]中指定的不同数据结构启动状态转换。这种影响

转换在[VF迁移状态数组](#bookmark185)和[VF迁移状态](#bookmark186)位中是可识别的。由MR-PCIM启动的所有状态转换都会使[VF迁移状态](#bookmark187)位置1。

此迁移状态机制存在于支持VF迁移的每个VF中。迁移状态机不受函数依赖项列表的影响（请参见[9.2.1.3](#bookmark121)和www.example.com[部分9.3.3.8）](#bookmark188)。

[VF迁移状态](#bookmark189)不影响功能状态。如果VF状态需要作为[移出](#bookmark140)和/或[移入](#bookmark139)操作的一部分重置，SR-PCIM必须发出FLR来完成此操作。在没有FLR的情况下发生VF迁移时的VF行为

未定义。

执行说明

FLR和VF迁移

从系统A到系统B的VF迁移通常会涉及系统A中的一个FLR，

在接受MigrateIn操作之后但在使用VF之前，在系统B中执行MigrateOut操作和第二个FLR。 系统A使用第一个FLR来确保其数据不会泄漏。系统B使用第二个FLR来确保它以干净的状态开始。



9.3配置

9.3.1 SR-IOV配置概述

本节提供了SR-IOV添加的实现PF和VF的要求。

与所有函数一样，PF在配置空间中是可配置的。PF包含第9.3.3节所述的SR-IOV扩展功能[。](#bookmark190)PF用于发现、配置和管理与PF相关的VF以及其他用途

在本说明书中描述。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

9.3.2配置空间

支持SR-IOV的PF应实现以下章节中定义的[SR-IOV扩展功能](#bookmark191)。 VF应实现以下章节中定义的配置空间字段和功能。

9.3.3 SR-IOV扩展功能

这里定义的[SR-IOV扩展能力](#bookmark191)是必须在支持SR-IOV的每个PF中实现的PCIe扩展能力。该能力用于描述和控制PF的SR-IOV能力。

对于多功能器械，支持SR-IOV的每个PF应提供本节中定义的能力结构。 该能力结构可以存在于具有类型0配置空间Header的任何功能中。这种能力必须

不存在于具有类型1配置空间报头的函数中。 [图9-13](#bookmark192)显示了SR-IOV扩展能力结构。



31 20

下一个能力偏移

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3130292827262524 | 2322212019181716 | | 15141312111098 | | 76543210 |
| PCI Express扩展功能头 | | | | | |
| SR-IOV能力寄存器（RO） | | | | | |
| SR-IOV状态寄存器 | | | SR-IOV控制寄存器（RW） | | |
| 总VF（RO） | | | 初始VF（RO） | | |
| RsvdP | Fcn Dep Link（RO） | | NumVF（RW） | | |
| VF跨步（RO） | | | FirstVF偏移（RO） | | |
| VF设备ID（RO） | | | RsvdP | | |
|  | | 支持的页面大小（RO） | |  | |
| 系统页面大小（RW） | | | | | |
| VF BAR 0（RW） | | | | | |
| VF BAR1（RW） | | | | | |
| VF BAR2（RW） | | | | | |
| VF BAR3（RW） | | | | | |
| VF BAR4（RW） | | | | | |
| VF BAR5（RW） | | | | | |
| VF迁移状态数组偏移（RO） | | | | | |

图9-13[SR-IOV扩展能力](#bookmark191)

字节偏移+000h

+004h+008h+00Ch+010h+014h+018h+01Ch+020h+024h+028h+02Ch+030h+034h+038h+03Ch

<9.3.3.1>SR-IOV扩展能力报头（偏移00 h）

[表9-3](#bookmark194)定义了[SR-IOV扩展能力报头](#bookmark193)。SR-IOV扩展功能的能力ID为0010 h。

|  |
| --- |
| 19 16 |
|  |

|  |
| --- |
| 15 0 |
|  |

PCI Express扩展能力ID能力版本

图9-14 SR-IOV扩展能力报头

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

表9-3 SR-IOV扩展能力报头

|  |  |  |
| --- | --- | --- |
| 位  位置 | 寄存器描述 | 属性 |
| 十五比零 | **PCI Express扩展能力ID**-该字段是指示扩展能力的性质和格式的PCI-SIG定义的ID号。  SR-IOV扩展能力的扩展能力ID为0010 h。 | RO |
| 十九点十六分 | **Capability Version**-此字段是PCI-SIG定义的版本号，指示当前Capability结构的版本。  此版本的规格必须为1h。 | RO |
| 三十一点二十分 | **下一个功能偏移量**-此字段包含到下一个PCI Express功能结构的偏移量，如果功能的链接列表中不存在其他项，则为000 h。  对于在配置空间中实现的扩展能力，该偏移相对于PCI兼容配置空间的开始，因此必须始终为000h（对于PCI兼容配置空间的终止列表）或000h（对于PCI兼容配置空间的终止列表  能力）或大于0FFh。 | RO |

<9.3.3.2>SR-IOV功能寄存器（04 h）

[表9-4](#bookmark196)定义了SR-IOV能力字段的布局。

0

2

3

20

1

|  |
| --- |
| 31 21 |
|  |

|  |
| --- |
| RsvdP |

支持VF迁移

保留ARI功能层次结构

VF 10-Bit TagQuantizer

VF迁移消息转发器

图9-15SR-IOV功能寄存器表9-4SR-IOV功能寄存器

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 0 | **VFMigration Capable**-如果设置，PF为Migration Capable，并在MigrationCapableMR-PCIM下运行。 | RO |
| 1 | 保留ARI功能层次结构  PCIExpress端点：  如果置位，则[ARI能力层次](#bookmark199)位在某些电源状态转换中保留。RCiEP：  不适用-强烈建议将此位硬连线至0 b。 | RO |
| 2 | **支持VF10位标签解码器**-如果设置，所有VF必须支持10位标签解码器功能。如果清除，则VF必须不支持10位标记转换器能力。  如果PF的设备功能2寄存器中的10位标签转发器支持位为清零，则该位必须为清零。 | HwInit |



第1123页

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 三十一点二十一分 | **VF迁移中断消息号**-指示用于迁移中断的MSI/MSI-X向量。 如果[VF迁移能力](#bookmark137)为清除，则此字段中的值未定义。 | RO |

9.3.3.2.1[支持VF迁移](#bookmark137)

设置VF[Migration Capable](#bookmark137)表示PF支持VF Migration。如果清除，PF不支持VF迁移。

VF迁移支持是[ MR-IOV ]的可选功能。如果PF未实现VF迁移的硬件，则该位应实现为RO值0b。如果PF实现VF迁移的硬件，则该位由

[ MR-IOV ]中定义的机制，并指示是否存在VF迁移支持。

仅实现SR-IOV的设备应将该字段实现为RO值零。 支持VF迁移的PF必须实现MSI或MSI-X中断（或两者）。

9.3.3.2.2[保留ARI能力层次结构](#bookmark198)

[ARI Capable Hierarchy Preserved（ARI能力层次保留）](#bookmark198)被设置为指示PF在某些电源状态转换期间保留[ARI Capable Hierarchy](#bookmark203)位（参见[第9.3.3.3.5节](#bookmark204)）。组件必须设置此位或设置No\_Soft\_Reset位（参见[第9.6.2节](#bookmark205)）。建议组件设置此位，即使它们也设置了No\_Soft\_Reset。

[保留的ARI能力层级P](#bookmark198)仅存在于设备的最低编号的PF（例如PF 0）中。[ARI CapableHierarchy Preserved](#bookmark198)在设备的其他PF中为只读零。

[ARI Capable Hierarchy Preserved](#bookmark198)不适用于RCiEP，其值未定义（参见[第9.3节）。3.3）](#bookmark206)。

9.3.3.2.3支持[VF 10位标记请求](#bookmark200)

如果PF支持10位标签解密器功能，则允许其关联的VF支持10位标签解密器

功能也是如此，但这是可选的。特别是对于大部分流量同时分布在多个VF上的使用模型，单个VF可能没有必要使用10-Bit标签，这样它们就可以支持>256个未完成流量 每个人都有未发布的请求。

对于给定的PF，要求其关联的VF中的全部或全部不支持10位标签识别器能力。 这避免了不必要的实施和管理复杂性。 参见www.example.com[部分9.3.5.9。](#bookmark208)

支持10位标签转发器功能的VF具有超出其他功能类型的额外要求，以便简化错误处理并减少与一个VF影响其他流量的10位标签相关错误的可能性。

· 如果SR-IOV控制寄存器中的[VF 10位标签验证器使能](#bookmark209)位被设置，则每个VF必须对其生成的所有非发布请求使用10位标签。

· 对于每个未完成的10位标签请求，如果VF接收到与未完成的请求匹配的完成th，而不是标签[9：8]为00b，则VF必须防止该请求（最终）生成完成

错误，而是通过避免数据损坏的特定于设备的机制来处理错误。

强烈建议软件不要将意外完成错误配置为不可纠正的错误。这避免了它们触发系统错误或硬件错误遏制机制，如下游端口 遏制（DPC）。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0



**执行说明**

NoVF 10位标记完成程序支持位

没有VF 10位标记完成程序支持位。如果PF支持10位标签完成器功能，则其所有关联的VF都需要支持10位标签完成器功能，如第www.example.com节所述[9.3.5.9。](#bookmark211)这有助于避免PCIe层次结构的复杂性，其中一些完成器支持10位标签功能，而一些不支持。

9.3.3.2.4[VF迁移验证消息编号](#bookmark201)

[VF迁移中断消息编号](#bookmark201)必须包含用于与某些VF迁移事件关联生成的中断消息的MSI或MSI-X向量编号。

对于MSI，[VF迁移中断消息编号](#bookmark201)必须指示MSI消息编号[0.. [31]用于参考

本节中描述的某些SR事件。当软件写入“多消息启用”字段中的“多消息启用”字段时，如果分配给该功能的MSI消息数量发生变化，则需要该功能更新该字段，以确保该字段

用于MSI的消息控制寄存器。

对于MSI-X，[VF迁移中断消息号](#bookmark201)必须指示MSI-X表条目[0.. 2047]用于生成中断消息。

如果MSI和MSI-X都被实现，它们被允许使用不同的向量，尽管软件一次只能启用一种机制。如果MSI和MSI-X中断都已启用，则此字段未定义。

如果[VF Migration Capable](#bookmark137)（VF迁移能力）为Clear（清除），则此字段未定义。

执行说明

迁移和MSI

如果PF实现MSI，并且软件将Multiple Message Enable设置为比Multiple Message Capable更小的值，则可能会发生MSI向量的某些共享。这可能会创建复杂的软件结构，

向量可能需要定向到SR-PCIM和PF设备驱动程序。



<9.3.3.3>SR-IOV控制寄存器（偏移08 h）

[表9-5](#bookmark213)定义了SR-IOV控制字段的布局

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 15 6 | 5 | 4 | 3 | 2 | 1 | | 0 |
| RsvdP |  |  |  |  |  |  | |

VF启用

VF迁移启用

VF迁移启用VF MSE

ARI能力层次

VF 10-BitTag Quantizer Enable

图9-16SR-IOV控制寄存器表9-5SR-IOV控制寄存器

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 0 | **VFEnable**-启用/禁用VF。Defau lt值为0b。 | RW |
| 1 | **VF迁移启用**-启用/禁用VF迁移支持。默认值为0b。见[章节](#bookmark215)  [9.3.3.3.2.](#bookmark216) | RW或RO  （见  描述） |
| 2 | **VF迁移中断启用**-启用/禁用VF迁移状态更改中断。默认值为0b。 | RW |
| 3 | **VFMSE**- 为虚拟函数启用内存空间。默认值为0b。 | RW |
| 4 | **ARI能力层次**-PCIExpress端点：  该位在器件的最低编号PF中必须为RW，在所有其他PF中必须硬连线为0b。 如果该位的值为1b，则允许设备在捕获的总线号的功能号8至255中定位VF。否则，设备必须像非ARI设备一样定位VF。  该位不受任何PF或VF的FLR影响。 默认值为0b。  RCiEP：  不适用-此位必须硬连线至0 b。  在根复合体内，始终允许将VF分配给[第一VF偏移](#bookmark218)和[VF步幅](#bookmark219)规则所允许的任何功能编号（参见[第www.example.com节9.3.3.9](#bookmark220)和[第www.example.com节9.3.3.10）](#bookmark221)。 | RW或RO  （见  描述） |
| 5 | **VF10位标签识别器使能**-如果设置，则所有VF必须为它们生成的所有非发布请求使用10位标签。如果清除，VF不能使用10位标签的非发布请求，他们产生.见[章节](#bookmark207) [9.3.3.2.3.](#bookmark207)  如果软件在任何VF有未完成的非发布请求时更改此位的值，则结果未定义。  如果[SR-IOV功能寄存器](#bookmark195)中的[VF 10位标签识别器支持](#bookmark200)位为清除，则此位为  允许硬连线到0b。 默认值为0b。 | RW或RO |



第1126页

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

9.3.3.3.1[VF启用](#bookmark25)

[VF启用](#bookmark25)管理VF到关联PF的分配。如果VF启用被设置，则与PF关联的

在PCI Express结构中可访问。设置后，VF会响应PCI Express交易，并可能会按照PCI Express端点功能的规则发出PCI Express交易。

如果[VF Enable（VF启用](#bookmark25)）为Clear（清除），则VF被禁用，并且在PCI Express结构中不可见;对这些VF的请求应接收UR，并且这些VF不应发出PCI Express事务。

为了允许组件执行内部初始化，在将[VF启用](#bookmark25)位从清零更改为置位后，系统不允许向由[VF启用](#bookmark25)位启用的VF发出请求，直到以下之一为真：

·至少100 ms已过去

·从PF收到FRS消息，原因代码为[VF启用](#bookmark25)d

·至少[VF启用](#bookmark25)时间已过。[VF启用](#bookmark25)时间是（1）与VF相关的准备时间报告能力中的复位时间值，或（2）由系统软件/固件159确定的值。

根复合体和/或系统软件必须在设置[VF使能](#bookmark25)位后至少允许1.0s，然后才能

确定未能返回有效配置请求的成功完成状态的VF失败。

在设置[VF使能](#bookmark25)位之后，允许由该[VF使能](#bookmark25)位使能的VF将CRS状态返回到

如果配置请求未准备好为有效的

配置请求。在PF发送原因代码为[VF启用](#bookmark25)d的FRS消息后，不允许与该PF关联的VF在没有干预VF禁用或其他有效重置条件的情况下返回CRS。在返回A

成功完成任何请求，则不允许VF在没有VF禁用或其他有效重置条件的情况下返回CRS。

由于VF没有MSE位（VF中的MSE由PF中SR-IOV扩展功能中的[VF MSE](#bookmark217)位控制），因此软件可能会在VF准备好处理之前发出内存请求。因此，内存R要求

在满足以下至少一个条件之前，不得向

·VF已成功响应（不返回CRS）配置请求。

·在向VF发出FLR之后，以下至少一项为真：自发布FLR以来，至少已过了1.0 s。

. VF支持FRS，并且在发出FLR后，从VF收到FRS消息，原因代码为FLR完成d。

. 自发布FLR以来，至少已经过了FLR时间。FLR时间是（1）中的FLR时间值 与VF相关联的就绪时间报告能力，或者（2）由系统软件/固件153确定的值。

·在PF中设置[VF启用](#bookmark25)后，至少有以下一项为真：自设置[VF Enable](#bookmark25)（VF启用）以来，至少已过去1.0 s。

. PF支持FRS，并且在[设置VF启用](#bookmark25)后，从PF收到FRS消息，原因代码为[VF启用](#bookmark25)d。

. 自设置VF启用后，至少[已过VF启用](#bookmark25)时间。[VF启用](#bookmark25)时间是（1）与VF关联的准备时间报告功能中的重置时间值，或（2）

由系统软件/固件160确定。



159.例如，ACPI表。

160.例如，ACPI表。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

在已向VF发出FLR或已在相关联的PF的[SR-IOV扩展能力](#bookmark191)中设置[VF启用](#bookmark214)之后，允许VF静默地丢弃内存请求，直到VF成功响应（不返回CRS）任何

请求过程中

清除[VF使能](#bookmark214)有效地破坏VF。 设置[VF Enable可](#bookmark214)有效创建VF。在先前清除后设置[VF启用](#bookmark214)应产生一组新的VF。如果PF处于D0功率状态，则新VF处于

D0未初始化状态。如果PF处于低功率状态，则行为未定义（参见第9.6.1节和第9.6.2节）。

清除[VF启用时，](#bookmark214)支持FRS的PF应发送FRS消息（FRS ReasonVF禁用），以指示此操作何时完成。如果有未完成的未发布邮件，PF不允许发送此 PF或与PF有关联的任何VF发出的请求。FRS消息只能在这些

请求已完成（或超时）。

清除[VF启用](#bookmark214)后，[SR-IOV扩展功能](#bookmark191)或[VF迁移状态数组](#bookmark222)中没有字段（请参见[第节](#bookmark223)

[9.3.3.15.1）](#bookmark224)，直到：

·VF Enable[（VF启用](#bookmark29)）清除后至少经过1.0 s。

·PF支持FRS，并且在[VF启用](#bookmark29)被清除后，从PF收到FRS消息，其中VF禁用原因代码。

[9.3.3.7 NumVF部分、](#bookmark225)[9.3.3.5 InitialVF部分、www.example.com TotalVF](#bookmark226)[部分9.3.3.6](#bookmark227)[9.3.3.9 First VF Offset部分、](#bookmark228)[9.3.3.13](#bookmark229)[System PageSize部分](#bookmark230)和[9.3.3.14](#bookmark231)VF BARx部分描述了与该字段相关联的附加语义。

9.3.3.3.2[启用VF迁移](#bookmark156)

必须设置“[VF迁移启用](#bookmark156)”以允许在此PF上进行VF迁移。详情参见[MR-IOV]

如果[VF Migration Capable（VF迁移能力）](#bookmark137)为Set（设置），则此位为RW，默认值为0b。否则，该位为RO零。 当[VF启用](#bookmark29)被设置时，该字段为RO。

9.3.3.3.3[VF迁移启用](#bookmark217)

每次以下条件的逻辑AND从“否”转换为“真”时，必须发送MSI

·VF

·关联的向量是未标记的。

·[VF迁移中断使能](#bookmark217)位被设置。

·[VF迁移状态](#bookmark232)位被设置。

使用的MSI或MSI-X向量由[VF迁移中断消息编号](#bookmark201)字段指示。 如果[VF Migration Capable](#bookmark137)（VF迁移能力）为Clear（清除），则此字段未定义。

注：VF迁移事件描述见[第9.2.4节。](#bookmark134)

9.3.3.3.4[VF MSE](#bookmark217)（内存空间启用）

[VF MSE](#bookmark217)控制与此PF相关的所有活动VF的内存空间启用，就像功能的PCI命令寄存器中的内存空间启用位一样。此位的默认值为0b。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

当[VF启用](#bookmark174)被设置时，VF内存空间将仅在[VF MSE](#bookmark217)被设置时响应。如果在设置VF启用时尝试访问虚拟功能的内存空间，则VF应遵循[ PCIe ]中定义的相同错误报告 而[VF MSE](#bookmark217)是清除的。

执行说明

VF MSE和VF使能

当VF Enab le为Clear时，VF内存空间将响应Unsupported Request。因此，[当VF Enable（VF使能）为Clear（清除）时，VF MSE](#bookmark217)为[“无关”;但是，在设置VF Enable（VF使能）之前，软件可以选择在对VF BARn寄存器进行清零之后设置VF MSE。](#bookmark174)



9.3.3.3.5[ARI能力层级](#bookmark203)

对于与上游端口相关联的设备[，RI能力层次结构](#bookmark203)是对设备的提示，即ARI已经在设备正上方的根端口或交换机下游端口中被启用。软件应设置此位以匹配ARI 设备正上方的根端口或交换机下游端口中的转发使能位。

[ARI能力层次](#bookmark203)结构仅存在于设备的编号最低的PF中（例如PF 0），并影响设备的所有PF。[ARI能力层次结构](#bookmark203)在设备的其他PF中为只读零。

设备可以使用[ARI能力层级的设置](#bookmark203)来确定[第一VF偏移的值](#bookmark233)（参见[第9.3.3节）。（九）](#bookmark234)

和[VFStride](#bookmark235)（参见www.example.com[部分9.3.3.10](#bookmark236)）。如果在任何PF中设置了VF Enable（VF启用），则未定义更改ARI Capable Hierarchy（ARI能力层次）的效果。常规复位时，该位必须设置为默认值。该位不受任何PF或VF的FLR影响。如果

如果[ARI Capable Hierarchy Preserved（保留ARI能力层次）](#bookmark198)被置位（见[第9.3.3.2.2节](#bookmark202)）或No\_Soft\_Reset（无软复位）被置位（见[第9.6.2节](#bookmark237)），则该PF从D3 Hot到D0的功率状态转换不会影响该位的值（见[第9.6.2节）](#bookmark238)。

[ARI能力层次](#bookmark203)不适用于RCiEP。

执行说明

ARI能力层次

对于与上游端口相关联的设备，该设备无法知道ARI是否已被启用。如果ARI被启用，设备可以通过将VF分配给大于7的功能号来节省总线号。 公交车的号码。ARI的定义见第6.13节。

由于RCiEP不与上游端口相关联，因此ARI不适用，并且VF可以被分配给任何 第[一VF偏移](#bookmark239)和[VF步幅](#bookmark240)允许的根复合体内的函数编号（参见[第www.example.com节9.3.3.8](#bookmark241)和[9.3.3.9）](#bookmark242)。



<9.3.3.4>SR-IOV状态寄存器（Offset0Ah）

[表9-6](#bookmark243)：定义了SR-IOV状态字段的布局。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |
| --- | --- |
| 15 1 | 0 |
| RsvdZ |  |

VF迁移状态

图9-17 SR-IOV状态

表9-6 SR-IOV状态

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 0 | **VFMigrationStatus**-表示MR-PCIM已发出VF移入或移出请求。为了确定事件的原因，软件可以扫描VF状态阵列。默认值为0b。 | RW1C |

9.3.3.4.1[VF迁移状态](#bookmark244)

当[VF启用、](#bookmark214)[VF迁移能力](#bookmark137)和[VF迁移启用](#bookmark156)被设置且[VF迁移状态数组](#bookmark245)条目中发生某些状态更改时，VF迁移状态被设置（参见[第9.2.4节](#bookmark134)和[第9.3.3.15.1节f](#bookmark246)或详细信息）。

VF[MigrationStatus的此设置](#bookmark232)不受[VF Migration Reset Enable的值](#bookmark217)或MSI或MSI-X的任何控件的影响。

<9.3.3.5>初始值VF（偏移0Ch）

[InitialVF](#bookmark226)向SR-PCIM指示最初与PF相关联的VF的数量。Ini tialVF的最小值为0。

对于在单根模式下运行的设备，此字段为HwInit，并且必须包含与TotalVF相同的值[。](#bookmark248)

对于在多根模式下运行的设备，当[VF Enable（VF启用）](#bookmark29)为Clear（清除）时，MR-PCIM可更改此字段的值。注：MR-PCIM用于影响该字段的机制见[MR-IOV]。

如果[VF Migration Enable（VF迁移启用）](#bookmark156)为Set（设置），[VF Enable（VF启用](#bookmark29)）为Clear（清除），然后Set（设置），则Initi alVF的值可能会更改。这是必要的，因为某些VF可能已迁移到其他PF，并且可能不再可用于此PF。

<9.3.3.6>TotalVF（偏移0Eh）

[TotalVF](#bookmark248)表示可与PF关联的VF的最大数量。TotalVF的最小值为0。

对于在单根模式下运行的设备，此字段为HwInit，并且必须包含与InitialVF相同的值[。](#bookmark226)对于在多根模式下操作的设备，该字段的值可以由MR-PCIM改变。

注：MR-PCIM用于影响该字段的机制见[MR-IOV]。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.3.3.7>NumVF（偏移10h）

[NumVF](#bookmark250)控制可见VF的数量。 SR-PCIM将NumVF设置为创建VF过程的一部分。将[NumVF](#bookmark250)设置为有效值并设置[VF启用后，该VF数量](#bookmark214)应在PCI Express结构中可见。 可见VF具有为其保留的函数编号，但可能不存在。如果VF存在，它应响应PCI Express

交易，并应遵循本规范规定的所有规则。如果存在以下任一情况，则存在VF

·[VF迁移能力](#bookmark197)是清除的，并且VF数量小于或等于[TotalVF。](#bookmark249)

·[VF Migration Capable](#bookmark197)被设置并且相关联的VF处于[Active.Available](#bookmark251)或[Dorset.MigrateIn](#bookmark252)状态（参见[第9.2.4节）](#bookmark134)

如果[NumVFs](#bookmark250)设置为大于[TotalVFs的值，则结果未定义。](#bookmark164)

[仅](#bookmark250)当VF Enable（VF[启用](#bookmark174)）为Clear（清除）时，才可写入NumVF。如果在[VF Enable（VF启用）](#bookmark174)为Set（设置）时写入NumVF，则结果未定义。

NumVFs的初始值未定义。

<9.3.3.8>功能扩展cy链接（偏移12h）

设备的编程模型可以在功能集合之间具有供应商特定的依赖性。 [FunctionDependency Link](#bookmark241)字段用于描述这些依赖关系。

此字段描述PF之间的依赖关系。 VF依赖关系与其关联PF的依赖关系相同。

如果PF独立于设备的其他PF，则该字段应包含其自己的功能编号。

如果PF依赖于设备的其他PF，则该字段应包含同一个PF中下一个PF的功能编号

函数依赖列表。函数依赖列表中的最后一个PF应包含函数依赖列表中的第一个PF的函数编号。

如果PF p和PF q在相同的函数依赖列表中，则被分配VF p，n的任何SI也应当被分配给VFq，n。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0



执行说明

函数依赖关系链接示例

考虑以下场景：

|  |  |  |  |
| --- | --- | --- | --- |
| SR-IOV字段 | PF 0 | PF 1 | PF 2 |
| [函数依赖关系链接](#bookmark188) | 1 | 0 | 2 |
| [NumVF](#bookmark119) | 4 | 4 | 6 |
| [第一个VF偏移](#bookmark253) | 4 | 4 | 4 |
| [VF跨步](#bookmark254) | 3 | 3 | 3 |

|  |  |  |
| --- | --- | --- |
| 功能号 | 描述 | 独立 |
| 0 | PF 0 | 没有 |
| 1 | PF 1 | 没有 |
| 2 | PF 2 | 是的 |
| 3 | 功能不存在 |  |
| 4 | VF 0，1（又名PF 0 VF 1） | 没有 |
| 5 | VF 1，1（又名PF 1 VF1） | 没有 |
| 6 | VF 2，1（又名PF 2 VF 1） | 是的 |
| 7 | VF 0.2 | 没有 |
| 8 | VF 1，2 | 没有 |
| 9 | VF 2.2 | 是的 |
| 10 | VF 0.3 | 没有 |
| 11 | VF 1.3 | 没有 |
| 12 | VF 2.3 | 是的 |
| 13 | VF 0.4 | 没有 |
| 14 | VF 1.4 | 没有 |
| 15 | VF 2.4 | 是的 |
| 16至17 | 功能不存在 |  |
| 18 | VF 2.5 | 是的 |
| 19至20 | 功能不存在 |  |
| 21 | VF 2.6 | 是的 |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |
| --- | --- | --- |
| 功能号 | 描述 | 独立 |
| 22至255 | 功能不存在 |  |

在本例中，功能4和5必须分配给同一SI。同样，函数7和8、10和11以及13和14必须一起分配。如果PF分配给SI，则功能0和1也必须一起分配。功能2、6、9、12、15、18和21是独立的，并且可以以任何方式分配给SI。

功能依赖列表中的所有PF应具有相同的[InitialVF、](#bookmark247)[TotalVF](#bookmark248)和[VF迁移能力](#bookmark137)字段值。

SR-PCIM应确保在将功能依赖列表中的任何VF分配给SI之前，该功能依赖列表中的所有PF具有相同的[NumVF、](#bookmark225)[VF启用](#bookmark29)和[VF迁移启用](#bookmark156)字段值。

VF迁移和VF映射操作对于每个VF独立发生。 SR-PCIM不得将VF分配给SI，直到其可以分配所有相关VF。例如，使用上述场景，如果VF0，2（功能7）和VF1，2（功能8）都处于[非活动不可用](#bookmark255)或[休眠迁移](#bookmark256)进入状态，则SR-PCIM不应将任一VF分配给SI，直到两个VF都达到[活动可访问](#bookmark257)状态。

类似地，SR-PCIM不应从SI中删除VF，直到其可以删除所有相关VF。例如使用

在上述情况下，只有当VF 0，2和VF 1，2都达到

[活动。迁移出](#bookmark258)状态。 SR-PCIM不应将功能转换为[非活动。](#bookmark259)不可用，除非SI已停止使用所有相关功能。



<9.3.3.9>第一VF偏移（偏移14h）

[第一VF偏移](#bookmark220)是一个常量，定义了与PF相关联的第一VF的路由ID偏移，PF包含

这种能力结构。第一个VF的16位路由ID是通过使用无符号的16位算术将该字段的内容添加到包含该字段的PF的路由ID（忽略任何进位）来计算的。

VF不应位于数字小于其关联PF的总线编号上。

当编号最低的PF的ARI Capable Hierarchy值更改或当此PF的NumVF值更改时，此字段可能会更改值。

注：如果[NumVFs](#bookmark225)为0，则不使用[第一VF偏移](#bookmark220)如果[NumVFs](#bookmark225)大于0，则[第一VF关闭设置](#bookmark220)不得为零。

<9.3.3.10>VF步幅（偏移设置16小时）

[VF步幅](#bookmark221)定义了与PF相关联的所有VF从一个VF到下一个VF的路由ID偏移，其中PF包含 能力结构。 新VF的16位路由ID是通过使用无符号16位算术将此字段的内容添加到当前VF的路由ID来计算的，忽略任何进位。

当编号最低的PF的ARI Capable Hierarchy值更改或当此PF的NumVF值更改时，此字段可能会更改值。

注意：如果[NumVF](#bookmark225)为0或1，[则VFStride](#bookmark221)如果[NumVFs](#bookmark225)大于1，则[VF步幅](#bookmark221)不能为零。

<9.3.3.11>VF设备ID（O偏移1Ah）

此字段包含应为每个VF向SI显示的设备ID

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

[VF设备ID](#bookmark260)可能与PF设备ID不同。VF[设备ID](#bookmark260)必须由供应商管理。供应商必须确保所选的[VF设备ID](#bookmark260)不会导致使用不兼容的设备驱动程序。

<9.3.3.12>支持的页面大小（偏移1通道）

此字段指示PF支持的页面大小。如果设置了位n，则此PF支持2n +12的页大小。例如，如果设置了位0，则PF支持4 KB的页面大小。PF需要支持4-KB、8-KB、64-KB、256-KB、1-MB和4-MB页面大小。所有其他页面大小都是可选的。

页面大小描述了VF BAR资源的最小对齐要求，如第www.example.com节中所述[9.3.3.13。](#bookmark261)

执行说明

非预取地址空间

非预取地址的空间限制为4 GB以下的地址。32位系统中的预取地址空间也是有限的。强烈建议供应商利用[系统页面大小](#bookmark262)功能来节省地址空间，同时支持具有较大页面的系统。



<9.3.3.13>系统页面大小（偏移量20h）

此字段定义系统将用于映射VF的存储器地址的页面大小。软件必须将“系统页面大小”的值[设置](#bookmark262)为“[支持的页面大小](#bookmark40)”字段中设置的页面大小之一（参见[第www.example.com节9.3.3.12](#bookmark40)）。 与[支持的页面大小](#bookmark40)一样，如果在[系统页面大小中设置了位n](#bookmark262)，则与此PF相关联的VF需要支持2n+12的页面大小。例如，如果设置了位1，则系统使用的是8 KB的页面大小。 如果[系统页面大小](#bookmark262)为零，则结果未定义。

如果在系统页面大小中设置了多个位，则结果未定义[。](#bookmark262) 如果在[系统页面大小中设置了一个位，](#bookmark262)而在支持[的页面大小中未设置，则结果未定义。](#bookmark40)

设置[系统页面大小](#bookmark262)时，需要与此PF关联的VF来对齐[系统页面大小](#bookmark262) 边界每个VF BARn或VF BARn对（参见[第www.example.com节9.3.3.14）](#bookmark263)应在[系统页面大小](#bookmark262)边界上对齐。每个 定义非零地址空间的VF BARn或VF BARn对的大小应设置为消耗[系统页面大小](#bookmark262)字节的整数倍。VF内需要页面大小对齐的所有数据结构应在[系统页面大小](#bookmark262)边界上对齐。

当[系统页面大小](#bookmark262)被写入时，VF启用必须为零。如果在VF Enable（VF启用）被设置时写入System Page Size（系统页面大小），则结果未定义。

默认值为0000 0001h（即，4KB）。

<9.3.3.14>VFBAR0（偏移24 h）、VFBAR1（偏移28 h）、VFBAR2（偏移2Ch）、VFBAR3（偏移

VFBAR4（偏移34 h）、VFBAR5（偏移38 h）

这些字段必须定义VF的基本地址寄存器（BAR）。 这些字段的行为与普通PCI BAR相同，如第7.5.1节所述。如第节所述，可以通过写入所有1并回读BAR的内容来调整它们的

7.5.1.2.1，符合定义BAR类型字段的低阶位。

这些字段的属性可能会受到[VF可调整大小的BAR扩展功能的影响](#bookmark264)（请参见www.example.com[部分9.3.7.5）](#bookmark265)。

每个BAR解码的地址空间量应为[系统页面大小的整数倍。](#bookmark262)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

每个VF BARn在通过写入1并读回内容来“调整大小”时

每个BAR由单个虚拟功能消耗和对齐所需。 当写入实际地址值时，[VF Enable](#bookmark214)和[VF MSE](#bookmark217)被置位，BAR映射[NumVF](#bookmark250)BAR。换句话说，基地址是第一个 与该PF相关联的VF BARn和所有后续VF BARn地址范围如下所述。

VF BAR应仅支持32位和64位内存空间。VF中不支持P CI I/O空间。位0的任何

实现的VF BARx必须是RO 0b，用于映射64位存储器VF BAR对的高32位的VF BARx除外。对齐要求和大小读取是针对单个VF的，但是当[VF启用](#bookmark174)被设置并且[VF MSE](#bookmark217)被设置时，BAR包含ll（[NumVF）](#bookmark250)VF BARn的基地址。

用于确定由VF BARn映射的地址空间量的算法与标准BAR算法的不同之处如下：

1. 通过[VF Resizable BAR Extended Capability（VF调整BAR扩展功能）调整BAR的大小](#bookmark266)（参见[第www.example.com节9.3.7.5）（](#bookmark267)如果已实施）。

2. 在读取低位以确定BAR是32位BAR还是64位BAR对之后，确定大小 以及通过将所有1写入到VF BARn（或对于64位BAR对，写入VF BARn和VF BARn+1）并读回BAR或BAR对的内容来满足对准要求。将读取返回的位掩码转换为大小， 第7.5.1.2.1节所述的对齐值。 该值是单个VF的大小和对齐方式。

3. 将步骤2中的值乘以NumVF中设置[的值](#bookmark28)，以确定在[VF Enable](#bookmark29)和[VF](#bookmark217)MSE设置后BAR或BAR对将映射的总空间量。

对于每个VF BARn字段，n对应于VF BAR空间之一。[表9-7](#bookmark268)显示了n和函数BAR之间的关系.

表9-7BAR偏移

|  |  |
| --- | --- |
| n | 类型0标头中的BAR偏移量 |
| 0 | 10h |
| 1 | 14h |
| 2 | 18h |
| 3 | 1Ch |
| 4 | 20h |
| 5 | 24h |

系统页面大小更改后，所有VF BARn寄存器的内容均不确定。

<9.3.3.15>VF迁移状态数组偏移（偏移3Ch）

如果[VF迁移能力](#bookmark137)[（第9.3.3.2.1节）被设置且](#bookmark201)[TotalVF（第www.example.com节9.3.3.6）不为零](#bookmark164)，则该寄存器应包含指向VF迁移状态数组的PFBAR相对指针。 如果[VF MigrationCapable（VF迁移](#bookmark137)能力）为Clear（清除），则该寄存器为RO Zero（RO零）。的[VF](#bookmark271)

[迁移状态数组](#bookmark272)定义见[第9.3.3.15.1节](#bookmark273)。VF[迁移状态Ar射线偏移的布局定义](#bookmark269)见[表](#bookmark274)[9-8。](#bookmark275)

2 0

31 3

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |
| --- | --- |
| VF迁移状态偏移 |  |

VF迁移状态BIR

图9-18VF迁移状态数组偏移表9-8VF迁移状态数组偏移

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 二比零 | **VF迁移状态BIR**-指示功能的基址寄存器中的哪一个  空间  BIR值BAR  **0** BAR0 10小时  **1** BAR 1 14小时  **2** BAR 218小时  **3** BAR3 1通道  **4** BAR4 20小时  **5** 酒吧5 24小时  **6** 保留  **7** 保留  对于64位BAR，[VF迁移状态BIR](#bookmark276)指示最低DW。 如果[TotalVF](#bookmark249)为0，则此字段未定义。 | RO |
| 三十一比三 | **VFMigrationState Offset**（VF迁移状态偏移）-用作函数基址寄存器之一所含地址的偏移量[，以指向VF迁移状态数组的基址。](#bookmark278)较低的三个[VF迁移状态BIR](#bookmark276)位由软件屏蔽（设置为零）以形成32位QW对齐偏移。  如果[TotalVF](#bookmark249)为0，则此字段未定义。 | RO |

如果映射用于[VF迁移状态阵列](#bookmark279)的地址空间的BAR还映射不与VF迁移相关联的其他可用地址空间，则在其他地址空间中使用的位置不得与VF迁移状态阵列所在的位置共享任何自然对齐的8 KB地址范围。

9.3.3.15.1 VF迁移状态数组

使用SR-IOV扩展能力块的[VF迁移状态阵列偏移](#bookmark269)寄存器来定位VF迁移状态阵列。

VF[迁移状态阵列](#bookmark273)具有用于每个VF的[VF迁移状态条目](#bookmark281)。[VF迁移状态数组](#bookmark273)的总大小为 [NumVFs](#bookmark28)字节。如果TotalVF为0，则[VF迁移状态数组](#bookmark273)不存在。[表9-9](#bookmark282)定义了[VF迁移状态数组](#bookmark273)条目的布局。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |
| --- | --- | --- |
| 7 2 | | 1 0 |
| RsvdP |  | |

VF迁移状态

图9-19VF迁移状态入口表9-9**VF迁移状态入口**

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 一比零 | **VF迁移状态**-关联VF的状态。 当[VF Enable（VF启用](#bookmark214)）为Clear（清除）时，此字段未定义。  VF[迁移状态数组的初始值](#bookmark280)在第www.example.com节中描述[9.2.4.1。](#bookmark144) | RW |

[VF迁移状态](#bookmark170)包含[表9-10所示的值。](#bookmark283)

表9-10 VF迁移状态描述

|  |  |  |
| --- | --- | --- |
| VF状态 | VF存在 | 描述 |
| 00b | 没有 | **不活动。不可用**-VF不存在于SR中，也不迁移入或迁移出。 |
| 01b | 没有 | **Dormit.**MigrateIn-VF可供SR使用。 VF存在，但无法初始化事务。 |
| 10b | 是的 | 已请求SR放弃使用VF。 |
| 11b | 是的 | **活动。可用**- 功能齐全。可能被分配到一个SI。 |

VF[迁移状态数组的初始值](#bookmark171)在[第www.example.com节9.2.4.1](#bookmark144)。VF[迁移状态阵列在VF启用被清除后1.0 s内](#bookmark171)返回到第www.example.com节中所述[9.2.4.1](#bookmark144)的配置。

软件通过向条目写入新的状态值来启动状态转换。

有效的状态转换如[表9-11](#bookmark286)和[表9-12所示](#bookmark287)，如[图9-12所示](#bookmark169)。SR只能请求[表9-11](#bookmark288)中的变更。[表9-12](#bookmark289)中的变更由MR-PCIM使用[ MR-IOV ]中描述的机制发起， 具有此处描述的SR可见效果。SR-PCIM发出的且未在[表9-11中列出的任何转换](#bookmark290)均被忽略，且不会更改VF状态。

表9-11 SR-PIM发起的VF迁移状态转换

|  |  |  |  |
| --- | --- | --- | --- |
| 当前状态 | 新状态 | 变更发起人 | SR变化的可见效果 |
| [Dormant.MigrateIn](#bookmark284) | [Active.Available](#bookmark176) | Sr-PCIM | VF激活  VF现在存在。 |
|  |
| [Active.Available](#bookmark176) | [Dormant.MigrateIn](#bookmark284) | Sr-PCIM | VF停用  VF已经不存在了。 |
| [Active.MigrateOut](#bookmark285) | [Inactive.Unavailable](#bookmark177) | Sr-PCIM | VF移出完成 |
|  |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 当前状态 | 新状态 | 变更发起人 | SR变化的可见效果 |
|  |  |  | VF已经不存在了。 |

表9-12 MR-PIM发起的VF迁移状态转换

|  |  |  |  |
| --- | --- | --- | --- |
| 当前状态 | 新状态 | 变更发起人 | SR变化的可见效果 |
| [Active.Available](#bookmark176) | [Active.MigrateOut](#bookmark285) | Mr-PCIM | VF移出请求  VF仍然存在。 设置[VF迁移状态。](#bookmark244) |
| [Inactive.Unavailable](#bookmark177) | [Dormant.MigrateIn](#bookmark284) | Mr-PCIM | VF移入请求  VF仍然不存在。 设置[VF迁移状态。](#bookmark244) |
|  |
| [Dormant.MigrateIn](#bookmark284) | [Inactive.Unavailable](#bookmark177) | Mr-PCIM | VF移入收回  VF仍然不存在。 设置[VF迁移状态。](#bookmark244) |
|  |
| [Active.MigrateOut](#bookmark285) | [Active.Available](#bookmark176) | Mr-PCIM | VF移出收回  VF仍然存在。 设置[VF迁移状态。](#bookmark244) |

9.3.4 PF/VF配置Space Header

本节定义PF和VF配置空间字段的要求。

本章通篇列出的寄存器定义建立了现有PCI-SIG规范与支持PCIe SR-IOV的设备的PF/VF定义之间的映射。

<9.3.4.1>PF/VF类型0配置空间标题

[图9-20](#bookmark291)详细说明了PCIExpress 0型配置空间报头的寄存器字段的分配。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

31

0

|  |  |  |  |
| --- | --- | --- | --- |
| 设备ID | | 供应商ID | |
| 地位 | | 命令 | |
| 类代码 | | | 修订ID |
| BIST | 报头类型 | 主延迟计时器 | 高速缓存行大小 |
| 基址寄存器 | | | |
| Cardbus CIS指针 | | | |
| 子系统ID | | 子系统供应商ID | |
| 扩展ROM基址 | | | |
| 保留 | | | 能力指针 |
| 保留 | | | |
| 最大纬度 | 最小值\_Gnt | 中断引脚 | 中断线 |

字节偏移

00h

04h

08h

0Ch

10h

14h

18h

1Ch

20h

24h

28h

2Ch

30h

34h

38h

3Ch

OM14316

图9-20 PF/VF类型0配置空间报头

本节定义了PCI ExpressType 0配置空间报头专用的寄存器的PCI专用解释。

错误报告字段在第9.4节中有更详细的描述[。](#bookmark292)

9.3.4.1.1供应商ID寄存器更改（偏移00h）

此只读寄存器标识设备的制造商。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

所有VF中的此字段在读取时返回FFFFh。 VI软件应从相关PF返回供应商ID值，作为VF的供应商ID值。

9.3.4.1.2设备ID寄存器更改（偏移02 h）

此只读寄存器标识特定设备。

所有VF中的此字段在读取时返回FFFFh。 VI软件应从相关PF返回VF器械ID（参见[第www.example.com节9.3.3.11）](#bookmark260)值，作为V F的器械ID。

执行说明

传统PCI探测软件

返回设备ID和供应商ID值的FFFFh允许某些旧版软件忽略VF。见章节

7.5.1.1.1.



9.3.4.1.3命令寄存器更改（偏移04h）

除表9-13中注明的情况外，PF和VF功能在第7.5.1.1.3节中定义[。](#bookmark294)对于标记为RsvdP的VF字段，PF设置应用于VF。

表9-13命令寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 0 | **I/O空间启用**-不适用于VF。必须硬连线到VF的0b。 | 基地 | 0b |
| 1 | **内存空间启用**-不适用于VF。必须硬连线到VF的0b。VF存储空间由VF控制寄存器中的VF MSE位控制。 | 基地 | 0b |
| 2 | **总线主机使能**-具有总线主机使能设置的VF的事务不能被具有总线主机使能清除的VF的事务阻止。 | 基地 | 基地 |
| 6 | 奇偶校验错误响应 | 基地 | RsvdP |
| 8 | SERR#启用 | 基地 | RsvdP |
| 10 | **禁用**-不适用于VF。 | 基地 | 0b |

9.3.4.1.4状态寄存器更改（偏移06h）

PF和VF功能在第7.5.1.1.4节中定义，[表9-14中注明的除外。](#bookmark295)

表9-14状态寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 3 | **身份验证状态**-不适用于VF。 | 基地 | 0b |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

9.3.4.1.5修订ID寄存器变更（偏移08h）

该寄存器指定设备特定的修订标识符。

VF中报告的值可能与PF中报告的值不同。

9.3.4.1.6类别代码寄存器更改（偏移09h）

类代码寄存器为RO，用于标识器件的通用功能，在某些情况下，还用于标识特定的寄存器级编程接口。PF和相关VF中的字段在读取时必须返回相同的值。

9.3.4.1.7高速缓存行大小寄存器更改（偏移0Ch）

此字段由PCI Express设备实现为读写字段，用于传统兼容性目的，但对任何PCI Express设备行为没有影响。物理职能部门继续将此字段作为RW实施。对于虚拟函数，

该字段是RO零。

9.3.4.1.8延迟定时器寄存器更改（偏移0Dh）

此字段不适用于P CI Express。该寄存器必须为RO零。

9.3.4.1.9报头类型寄存器更改（偏移0Eh）

该字节标识预定义报头的第二部分的布局（从配置空间中的字节10 h开始）以及器件是否包含多个功能。此寄存器中的位7用于标识多功能 设备.对于SR-IOV器件，只有在有多个功能时，此寄存器中的位7才会置位。 VF不影响位7的值。位6到0标识预定义标题的第二部分的布局。对于VF，此寄存器必须是RO

零.

9.3.4.1.10 BIST寄存器更改（偏移0Fh）

VF不应支持BIST，必须将此字段定义为RO零。

如果在设备的任何PF中打开VF Enable（VF启用），则软件不得在与该设备相关的任何Function（功能）中调用BIST。

9.3.4.1.11基址寄存器寄存器更改（偏移10h、14h、

对于VF，这些寄存器中的值为RO零。

注：另见www.example.com上的[第9.2.1.1.1节](#bookmark11)和第[19.3.3.14节。](#bookmark126)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

9.3.4.1.12Cardbus CIS指针寄存器更改（偏移28 h）

对于VF，不使用此寄存器，且应为RO零。

9.3.4.1.13子系统供应商ID寄存器更改（偏移2Ch）

此只读字段标识子系统的制造商。PF和相关VF中的字段在读取时必须返回相同的值。

9.3.4.1.14子系统ID寄存器更改（偏移2Eh）

此只读字段标识特定子系统。 器械的PF和VF值可能不同。

9.3.4.1.15扩展ROM基址寄存器寄存器更改（偏移30h）

扩展ROM基址寄存器允许在PF中实现。扩展ROM基址 寄存器在VF中为RO零。VI可以选择通过仿真为VF提供对PF扩展ROM基址寄存器的访问。

PF不允许实现扩展ROM地址解码器共享（参见第7.5.1.2.4节）。

9.3.4.1.16能力指针寄存器更改（偏移34h）

与第7.5.1.1.11节中的描述无差异。

9.3.4.1.17双通道线路寄存器更改（偏移3通道）

此字段不适用于VF。此字段必须为RO零。

9.3.4.1.18双引脚寄存器更改（偏移3Dh）

此字段不适用于VF。此字段必须为RO零。

9.3.4.1.19Min\_Gnt寄存器/Max\_Lat寄存器更改（偏移3Eh/3Fh）

这些寄存器不适用于PCI Express。他们一定是RO零。

9.3.5 PCI Express功能和更改

PCIExpress功能（参见第7.5.3节）用于识别PCI Express设备，并表示支持PCI Express功能。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

图7-21详细说明了PCI Express功能中寄存器字段的分配。PF和VF需要根据下文所述的例外情况和附加要求来实现此功能。

<9.3.5.1>PCIExpress功能寄存器更改（偏移00h）

PCIExpress功能寄存器在www.example.com上的第一节中进行了描述7.5.3.2，其中描述的功能适用于PF和VF。

<9.3.5.2>PCIExpress功能寄存器更改（Offset02h）

PCI Express功能寄存器标识PCI Express设备类型和相关功能。PF和VF功能在第www.example.com节7.5.3.2中定义。

<9.3.5.3>设备功能寄存器更改（偏移04h）

设备功能寄存器标识PCI Express设备特定的功能。图7-24详细说明了设备功能寄存器中寄存器字段的分配;[表9-15](#bookmark297)提供了相应的位定义。

除表9-15中注明的地方外，PF和VF功能在第www.example.com部分中定义7.5.3.3[。](#bookmark298)

表9-15器件功能寄存器C更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 四比三 | 支持的体模功能-设置VF启用时，使用体模功能  不允许此PF和相关VF的数字，读取时此字段必须返回00b。 | 基地 | 00b |
| 二十五点十八分 | 捕获的插槽功率限制值 | 基地 | 未定义 |
| 二十七点二十六分 | 捕获的插槽功率限制比例 | 基地 | 未定义 |
| 28 | 功能级复位能力-PF和VF需要。 | 1b | 1b |

<9.3.5.4>设备控制寄存器更改（Offset08h）

设备控制寄存器控制PCI Express设备特定参数。图7-25详细说明了器件控制寄存器中寄存器字段的分配;[表9-16](#bookmark300)提供了相应的位定义。

除表9-16中注明的情况外，PF和VF功能在www.example.com上的Secti中定义7.5.3.4[。](#bookmark301)对于标记为RsvdP的VF字段，PF设置应用于VF。

表9-16设备控制寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 0 | 可更正的错误报告启用  参见[第9.4f](#bookmark302)或有关错误报告的详细信息 | 基地 | RsvdP |
| 1 | 非致命错误报告启用 | 基地 | RsvdP |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
|  | 参见[第9.4f](#bookmark303)或有关错误报告的详细信息 |  |  |
| 2 | 致命错误报告启用  参见[第9.4f](#bookmark304)或有关错误报告的详细信息 | 基地 | RsvdP |
| 3 | 不支持的请求报告启用  参见[第9.4f](#bookmark305)或有关错误报告的详细信息 | 基地 | RsvdP |
| 4 | 实现轻松订购 | 基地 | RsvdP |
| 七点五 | 最大有效负载大小 | 基地 | RsvdP |
| 8 | 扩展标记字段激活 | 基地 | RsvdP |
| 9 | 幻象功能启用 | 基地 | RsvdP |
| 10 | 辅助电源PM启用 | 基地 | RsvdP |
| 11 | 启用无监听 | 基地 | RsvdP |
| 十四点十二分 | 最大读取请求大小 | 基地 | RsvdP |
| 15 | 启动功能级别重置-PF和VF需要。  注：在PF中设置Initiate Function Level Reset（启动功能级别复位）将复位VF Enable（VF启用），这意味着在FLR完成后VF不再存在。 | 基地 | 基地 |

<9.3.5.5>设备状态寄存器更改（偏移0Ah）

设备状态寄存器提供有关PCI Express设备特定参数的信息。图7-26详细说明了设备状态寄存器中寄存器字段的分配;[表9-17](#bookmark307)提供了相应的位定义。

除表9-17中所述外，PF和VF功能在第www.example.com部分中定义7.5.3.5[。](#bookmark308)

表9-17设备状态寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 4 | 检测到辅助电源 | 基地 | 0b |
| 6 | 检测到紧急断电 | 基地 | 0b |

<9.3.5.6>链路功能寄存器更改（偏移0Ch）

链路功能寄存器标识PCI Express链路特定功能。图7-27详细说明了链路功能寄存器中寄存器字段的分配。

PF和VF功能定义见7.5.3.6节。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.3.5.7>链接控制寄存器更改（偏移10h）

链路控制寄存器控制PCI Express链路特定参数。图7-28详细说明了链路控制寄存器中寄存器字段的分配。

除表9-18中注明的情况外，PF和VF功能在www.example.com上的Secti中定义7.5.3.7[。](#bookmark309)对于标记为RsvdP的VF字段，PF设置应用于VF。

表9-18链路控制寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 1：0的比分 | 活动状态电源管理（ASPM）控制 | 基地 | RsvdP |
| 3 | 读取完成总线（RCB必须硬连接到0 bf或VF。 | 基地 | RsvdP |
| 6 | 通用时钟配置 | 基地 | RsvdP |
| 7 | 扩展同步 | 基地 | RsvdP |
| 8 | 启用时钟电源管理 | 基地 | RsvdP |
| 9 | 硬件自主宽度禁用 | 基地 | RsvdP |

<9.3.5.8>链接状态寄存器更改（偏移12h）

链路状态寄存器提供有关PCI Express链路特定参数的信息。图7-29详细说明了链路状态寄存器中寄存器字段的分配。

PF功能在第www.example.com节7.5.3.8。对于VF，此寄存器中的所有字段均为RsvdZ，PF设置适用于VF。

<9.3.5.9>设备功能2寄存器更改（偏移24小时）

除表9-19中所述外，PF和VF功能在第www.example.com节中定义7.5.3.15[。](#bookmark211)

表9-19设备功能2寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 三比零 | 支持的完成范围  VF值必须与PF值相同。 | 基地 | 基地 |
| 4 | 支持完成禁用  VF值必须与PF值相同。 | 基地 | 基地 |
| 6 | 支持AtomicOp路由  不适用于端点。 | RsvdP | RsvdP |
| 7 | 支持32位 | 基地 | 基地 |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
|  | VF值必须与PF值相同。 |  |  |
| 8 | 支持64位  VF值必须与PF值相同。 | 基地 | 基地 |
| 9 | 128-支持位CAS完成程序  VF值必须与PF值相同。 | 基地 | 基地 |
| 16 | 支持10位标记完成程序  VF值必须与PF值相同。 | 基地 | 基地 |
| 17 | 支持10位  VF值必须等于SR-IOV功能寄存器中VF 10位标签识别器支持位的值。参见[第9.3.3.2.3](#bookmark207) | 基地 | 基地 |
| 二十五点二十四分 | 支持紧急断电  VF值必须与PF值相同。 | 基地 | 基地 |
| 26 | **紧急功率降低要求**VF值必须与PF值相同**。** | 基地 | 基地 |

<9.3.5.10>设备控制2寄存器更改（偏移28h）

除表9-20中所述外，PF和VF功能在第www.example.com节中定义7.5.3.16[。](#bookmark310)

表9-20设备控制2寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 三比零 | 完成时间值  PF值适用于所有关联的VF。 | 基地 | RsvdP |
| 4 | 完成资格  PF值适用于所有关联的VF。 | 基地 | RsvdP |
| 6 | AtomicOptener Enable  PF值适用于所有关联的VF。 | 基地 | RsvdP |
| 8 | IDO请求启用  PF值适用于所有关联的VF。 | 基地 | RsvdP |
| 9 | IDO完成启用  PF值适用于所有关联的VF。 | 基地 | RsvdP |
| 11 | 紧急功率降低请求  此位仅存在于与上游端口相关的一个函数中。该函数永远不会是VF。 | 基地 | RsvdP |



第1146页

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 12 | 10位标签转换器使能  SR-IOV控制寄存器中的VF 10位标签转发器使能位的值适用于所有相关VF。 | 基地 | RsvdP |

<9.3.5.11>设备状态2寄存器更改（fset2Ah）

PF和VF功能定义见7.5.3.17节。

<9.3.5.12>链接功能2寄存器更改（偏移2Ch）

PF和VF功能定义见7.5.3.18节。

<9.3.5.13>链接控制2寄存器更改（偏移30h）

PF和VF功能定义见7.5.3.19节。

<9.3.5.14>链接状态2寄存器更改（偏移32h）

除表9-21中所述外，PF和VF功能在第www.example.com节中定义7.5.3.20[。](#bookmark311)标记为RsvdZ的VF字段使用关联PF的值。

表9-21链路状态2寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 0 | 当前去加重电平 | 基地 | RsvdZ |

9.3.6 PCI标准功能

PCI标准功能的SR-IOV使用如[表9-22所述。](#bookmark312)标记为n/a的项目不适用于PF或VF。

表9-22 PCI标准功能的SR-IOV使用

|  |  |  |  |
| --- | --- | --- | --- |
| 能力ID | 描述 | PF属性 | VF属性 |
| 00h | 产品展示 | 基地 | 基地 |
| 01h | PCI电源管理接口 | 基地 | 可选的. 参见[第9.6节。](#bookmark313) |
| 02h | AGP | n/a | n/a |
| 03h | VPD | 基地 | 可选的. 参见www.example.com[部分9.3.6.1。](#bookmark314) |
| 04h | 插槽识别 | n/a | n/a |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 能力ID | 描述 | PF属性 | VF属性 |
| 05h | MSI | 基地 | 参见www.example.com[部分9.5.1.1。](#bookmark315) |
| 06h | CompactPCI热插拔 | n/a | n/a |
| 07h | PCI-X | n/a | n/a |
| 08h | HyperTransport | n/a | n/a |
| 09h | 厂商特定 | 基地 | 基地 |
| 0啊 | 调试端口 | 基地 | 基地 |
| 0Bh | CompactPCI中央资源控制 | n/a | n/a |
| 0Ch | PCI热插拔 | 基地 | n/a |
| 0Dh | PCI桥子系统ID | n/a | n/a |
| 0Eh | AGP 8x | n/a | n/a |
| 0FH | 安全设备 | n/a | n/a |
| 10h | PCI Express | 基地 | 参见[第9.3.5节。](#bookmark296) |
| 11h | MSI-X | 参见[9.5.1.2](#bookmark316)[部分](#bookmark317)和  [9.5.1.3网站。](#bookmark318) | 参见[9.5.1.2](#bookmark319)[部分](#bookmark320)和  [9.5.1.3。](#bookmark321) |
| 12h | 串行ATA数据/索引配置 | 基地 | n/a |
| 13h | 高级功能 | n/a | n/a |
| 14h | 增强分配 | 基地 | 不能执行。 |
| 15h | 压平门式桥（FPB） | n/a | n/a |

<9.3.6.1>VPD功能

PCI中的它在SR-IOV中仍然是可选的。

实现VPD能力的VF和PF必须确保通过VPD能力在VF和/或PF之间不存在“数据泄漏”。

9.3.7 PCI Express扩展功能更改

PCI Express扩展功能的SR-IOV使用如[表9-23所示。](#bookmark322)标记为n/a的项目不适用于PF或VF（例如，对于仅存在于根复合体中或仅存在于功能0中的能力）。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

表9-23 PCI Express扩展功能的SR-IOV使用

|  |  |  |  |
| --- | --- | --- | --- |
| 扩展  能力  ID | 描述 | PF属性 | VF属性 |
| 0000H | 产品展示 | 基地 | 基地 |
| 0001H | 高级错误报告扩展功能（AER） | 基地 | 参见[第9.4.2节。](#bookmark323) |
| 0002小时 | 虚拟通道扩展能力（02h） | 基地 | 不能执行。 参见www.example.com[部分9.3.7.1。](#bookmark324) |
| 0003H | 设备序列号扩展功能 | 基地 | 参见[9.3.7.2部分](#bookmark325) |
| 0004小时 | 电源管理扩展功能 | 基地 | 不能执行。参见[9.3.7.3部分](#bookmark326) |
| 0005小时 | 根复合体链接声明扩展能力 | n/a | n/a |
| 0006小时 | 根复合体内部链路控制扩展能力 | n/a | n/a |
| 0007小时 | 根复合体事件收集器终结点关联扩展功能 | n/a | n/a |
| 0008小时 | 多功能虚拟C通道扩展功能 | 基地 | 不能执行。 参见www.example.com[部分9.3.7.1。](#bookmark327) |
| 0009小时 | 虚拟信道扩展能力（09h） | 基地 | 不能执行。 参见www.example.com[部分9.3.7.1。](#bookmark328) |
| 000Ah | 报头扩展能力 | n/a | n/a |
| 000Bh | 特定于供应商的扩展功能 | 基地 | 基地 |
| 000Ch | 配置访问相关扩展功能 | n/a | n/a |
| 000Dh | ACS扩展功能 | 参见www.example.com[部分9.3.7.6。](#bookmark329) | 参见www.example.com[部分9.3.7.6。](#bookmark330) |
| 000Eh | 扩展能力（ARI） | 参见www.example.com[部分9.3.7.7。](#bookmark331) | 参见www.example.com[部分9.3.7.7。](#bookmark332) |
| 000Fh | ATS扩展功能 | 参见www.example.com[部分9.3.7.8。](#bookmark333) | 参见www.example.com[部分9.3.7.8。](#bookmark334) |
| 0010小时 | [SR-IOV扩展功能](#bookmark190) | 参见[第9.3.3节。](#bookmark190) | 不能执行。见[第9.3.3](#bookmark190) |
| 0011 h | [MR-IOV扩展功能](#bookmark134)（MR-IOV） | 不能执行。 参见www.example.com[部分9.3.7.9。](#bookmark335) | 不能执行。 参见www.example.com[部分9.3.7.9。](#bookmark336) |
| 0012小时 | 组播扩展功能 | 参见www.example.com[部分9.3.7.10。](#bookmark337) | 参见www.example.com[部分9.3.7.10。](#bookmark338) |
| 0013小时 | 页面请求扩展能力（PRI） | 参见www.example.com[部分9.3.7.11。](#bookmark339) | 参见[9.3.7.11部分](#bookmark340) |
| 0014小时 | 为AMD保留 | 基地 | 基地 |
| 0015小时 | 可调整大小的BAR扩展功能 | 基地 | 不能执行。参见[9.3.7.4部分](#bookmark341) |
| 0016小时 | 动态功率分配扩展能力（DPA） | 参见www.example.com[部分9.3.7.12。](#bookmark342) | 不能执行。 参见[第9.3节。7点12分](#bookmark343) |



第1149页

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 扩展  能力  ID | 描述 | PF属性 | VF属性 |
| 0017小时 | TPH扩展能力（TPH） | 参见www.example.com[部分9.3.7.13。](#bookmark344) | 参见[9.3.7.13部分](#bookmark345) |
| 0018小时 | LTR扩展功能 | 基地 | 不能执行。LTR使用功能0进行控制，功能0从不是VF。 |
| 0019小时 | 辅助PCI Express扩展功能 | 基地 | 不能执行。8.0GT/s使用功能0控制，功能0从不是VF。 |
| 001Ah | PMUX扩展功能 | 基地 | 不能执行。PMUX使用函数0进行控制，函数0从不是VF。 |
| 001Bh | PASID扩展能力 | 基地 | 参见[9.3.7.14部分](#bookmark346) |
| 001Ch | LN中继器扩展能力（LNR） | 基地 | 基地 |
| 001Dh | DPC扩展功能 | n/a | n/a. |
| 001Eh | L1 PM子状态扩展能力 | 基地 | 不能执行。L1 PM子状态为  使用函数0进行控制，它永远不是VF。 |
| 001Fh | 精确时间管理扩展功能（PTM） | 基地 | 不能执行。PTM控制端口，不得在VF中实现。 |
| 00时20分 | PCI Express over M-PHY扩展功能（M-PCIe） | 基地 | 不能执行。M-PHY使用功能0来控制，功能0从不是VF。 |
| 0021小时 | FRS提供扩展功能 | n/a | n/a |
| 0022小时 | 准备时间报告扩展功能 | 基地 | 参见[9.3.7.15部分](#bookmark347) |
| 0023小时 | 指定的特定于供应商的扩展功能 | 基地 | 基地 |
| 0024小时 | [VF可调整BAR扩展功能](#bookmark348) | 参见[9.3.7.5部分](#bookmark349) | 未执行。参见[9.3.7.5部分](#bookmark350) |
| 0025小时 | 数据链路特性扩展能力 | 基地 | 不能执行。 |
| 0026小时 | 物理层16.0 GT/s扩展功能 | 基地 | 不能执行。 |
| 0027小时 | 接收机扩展能力处的车道余量 | 基地 | 不能执行。 |
| 0028小时 | Hierarchy ID扩展功能 | 基地 | 基地 |
| 0029小时 | 本机PCIe盘柜管理扩展功能（NPEM） | 基地 | 不得实施 |

<9.3.7.1>虚拟频道/MFVC

VF使用关联PF的虚拟通道。 因此，VF不包含任何虚拟信道能力。 VF不应包含MFVC能力。此功能仅存在于功能0中，功能0永远不会是VF。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.3.7.2>设备序列号

设备序列号扩展功能可能存在于PF中。如果PF包含该功能，则其值适用于所有关联的VF。允许VF实现此功能，但不建议VF实现此功能。实现此功能的VF

功能必须返回与其关联PF报告的相同的设备序列号值。

<9.3.7.3>电源管理

PF中可能存在功率限制扩展功能，但VF不得实现该功能。如果PF包含该功能，则它必须报告覆盖所有关联VF的值。

可调整<9.3.7.4>的BAR

可调整大小的BAR扩展能力可以存在于PF中。 由于VF不实现标准BAR，因此VF中必须不存在该功能。 PF[的Resizable BAR设置不会影响SR-IOV扩展功能中的任何设置。](#bookmark191)

<9.3.7.5>BAR扩展功能

VF[可调整BAR扩展功能仅](#bookmark266)允许在实现至少一个VFBAR的PF中实现，并且影响PF的VF BAR的大小和基础。 由于VF本身并不实现BAR，因此VF中必须不存在该能力。PF可以实现[VF可调整大小BAR扩展能力](#bookmark266)和可调整大小BAR扩展能力两者。 能力，因为每个能力都是独立运作的。

VF[可调整BAR扩展功能](#bookmark266)是允许PF能够调整其VF的BAR大小的可选功能。[VF Resizable BAR扩展功能](#bookmark266)允许硬件将

可通过[VF Resizable BAR扩展功能](#bookmark266)和控制寄存器以及系统软件进行操作， 通过[VF可调整大小的BAR控制寄存器的VF BAR大小字段将最佳大小传送回硬件。](#bookmark352)

硬件立即将大小推断反映在相应VF BAR的只读位中。推断的大小是从系统页面大小和[VF BAR大小](#bookmark353)字段解码的值中的较大者。硬件必须清除任何位，

从读写变为只读，以便后续读取返回零。软件必须清除

SR-IOV控制寄存器，然后写入[VF BAR大小](#bookmark354)字段。写入[VF BARSize](#bookmark355)字段后，

对应的VF BAR未定义。为确保在对VF BAR进行重新编程后它包含有效地址，系统软件必须对VF BAR进行重新编程，并设置[VF MSE](#bookmark217)位（除非未分配资源）。

仅当相关的VF BAR为64位BAR时，才允许[VF Resizable BAR扩展功能](#bookmark45)和控制寄存器指示以4GB或更大容量工作的能力。

强烈建议功能不要在此功能中推荐任何支持的[VF BAR大小](#bookmark356)值，如果该值大于分配的有效利用的空间。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

执行说明

在资源分配过程

系统软件以类似于Resizable BAR功能的方式使用此功能。 系统软件必须首先配置系统页面大小寄存器（参见[第9.2.1.1www.example.com.1节）](#bookmark30)。潜在的可用存储器孔径大小为

由PF报告，并从[VF可调整BAR扩展功能](#bookmark45)和控制寄存器读取。由于分配的地址空间少于报告的最大大小可能会导致性能降低，因此软件应尽可能分配报告的最大大小。然后，软件将大小写入[VF可调整BAR](#bookmark357)

[控制寄存器](#bookmark358)，用于函数的应用程序VF BAR。在此之后，基地址被写入VFBAR。

出于互操作性的原因，硬件可能会将VF BAR的默认大小设置为较低的大小;该大小低于[VF可调整BAR能力寄存器中报告的最大大小。](#bookmark359) 软件不使用此

当资源小于期望值或由于没有空间而不可分配时，调整资源大小的能力将可能导致次优的资源分配。建议系统软件 负责在资源受限的环境中分配资源，将有限的地址空间适当地分配给所有内存映射的硬件，包括系统RAM。



VF[Resizable BAR扩展功能](#bookmark45)结构定义了PCI Express扩展功能，该功能位于PCIExpress扩展配置空间中，即前256个字节上方，如下[图9-21所示。](#bookmark360) 这

结构允许具有这种能力的PF被识别和控制。能力寄存器和控制寄存器被 为每个可调整大小的VF BAR实现。由于任何PF可以实现最多6个VF BAR，因此VF BAR可以被配置为： 可调整大小的BAR能力结构的范围可以从12字节长（对于单个VFBAR）到52字节长（对于所有6个VF BAR）。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 3130292827262524 | 2322212019181716 | | 15141312111098 | | 76543210 |
| PCI Express扩展功能头 | | | | | |
| VF可调整BAR容量寄存器（0） | | | | | |
|  | | VF可调整BAR控制寄存器（0） | |  | |
| VF可调整BAR容量寄存器（1） | | | | | |
|  | | VF可调整大小的BAR控制寄存器（1） | |  | |
| ... | | | | | |

图9-21[VF可调整BAR扩展功能](#bookmark45)

字节偏移+000h

+004h+008h+00Ch+010h+014h



31 20

下一个能力偏移

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

**9.3.7.5.1 VFResizablee BAR扩展能力报头（偏移00h）**

|  |
| --- |
| 19 16 |
|  |

|  |
| --- |
| 15 0 |
|  |

PCI Express扩展能力ID能力版本

图9-22VF可调整BAR扩展功能标题表9-24VF可调整BARE扩展功能标题

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 十五比零 | **PCI Express扩展功能ID**-此字段是PCI-SIG定义的ID号，指示扩展功能的性质和格式。  PCI Express扩展能力ID为0024h。 | RO |
| 十九点十六分 | **能力版本**-该字段是PCI-SIG定义的版本号，其指示存在的能力结构的版本。  此版本的规格必须为1h。 | RO |
| 三十一点二十分 | **下一个能力偏移量**-此字段包含到下一个PCI Express扩展能力结构的偏移量，如果在能力的链接列表中不存在 | RO |

9.3.7.5.2 VF可调整BAR能力寄存器（偏移04h）

[VF可调整BAR功能寄存器](#bookmark362)字段描述与可调整BAR中的定义

表7-116中的能力寄存器。如果这些描述是

位、它们的位置和它们的属性是相同的。因此，图7-145同样地分配这个寄存器中的寄存器字段.

9.3.7.5.3 VF可调整BAR控制寄存器（偏移08h）

VF[可调整BAR C控制寄存器](#bookmark357)位31：16遵循与表7-117中可调整BAR控制寄存器相同的定义。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 31 16 | 15 14 | 13 8 | 7 5 | 4 3 | 2 0 |
|  |  | VF BAR尺寸 |  |  |  |

VF BAR索引

RsvdP

VF可调整大小的BAR数量

RsvdP

与可调整大小的BAR控制寄存器相同

图9-23VFResizableBAR控制寄存器表9-25VFResizableBAR控制寄存器

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | 寄存器描述 | | 属性 |
| 二比零 | **VFBAR索引**-该编码值指向位于SR-IOV扩展能力中的该特定VFBAR[的开始。](#bookmark191)  **0** VF BAR位于偏移24 h处  **1** VF BAR位于偏移28 h  **2** VF BAR位于偏移2Ch  **3** VF BAR位于偏移30 h处  **4** VF BAR位于偏移34 h处  **5** VF BAR位于偏移38 h处  **别人** 所有其他编码都保留。  对于64位基址寄存器，[VF BAR Index](#bookmark363)表示较低的DWORD。  此值指示哪个VF B AR支持可协商大小。 | | RO |
| 七点五 | **NumberofVF Resizable BAR**-指示功能块的功能结构中可调整大小的VF BAR的总数。 见[图9-21。](#bookmark361)  此字段的值必须在01h到06h的范围内。 该字段在VF[可调整BAR控制寄存器](#bookmark358)（0）中有效（在offet08h），并且对于所有其他寄存器为RsvdP。 | | RO/RsvdP |
| 十三点八分 | **VFBAR大小**- 这是一个编码值。 | | RW |
| 0  1  2  3    43 | 1 MB（220字节）  2 MB（221字节）  4 MB（222字节）  8 MB（223字节）  …  8 EB（263字节） |
| 此字段的默认值等于VF BAR资源通过VF BAR的只读位请求的地址空间的默认大小。  软件必须只写入与VF Resizable中指示支持的值相对应的值  BAR能力和控制寄存器。写入不支持的值将产生未定义的结果。  当此寄存器字段被编程时，该值立即反映在资源的大小中，并以VF BAR中的只读位数进行编码。 | |



第1154页

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |
| --- | --- | --- |
| 比特位置 | 寄存器描述 | 属性 |
| 三十一点十六分 | 这些位**与图7-146中定义的可调整大小的BAR控制寄存器**位[31：16]相同。 如果这些描述为 | 看到 图7-146 |

<9.3.7.6>访问控制服务（ACS）扩展功能更改

ACS是可选的扩展功能。如果根联合体中的其他具有SR-IOV能力的设备实现内部对等事务，则需要ACS，并具有以下描述的附加要求。

除表9-26中注明的地方外，PF和VF功能在部分7.7.8.2[中定义。](#bookmark364)

具有SR-IOV能力的设备（实现至少一个PF的设备）中的除了支持与设备的对等事务的RCiEP之外的所有功能应实现ACS出口控制。

允许但不要求在RCiEP中实施ACS。显然，在单个根联合体内，一些RCiEP实现ACS，而一些RCiEP不实现。强烈建议根复合体

实现确保所有来自没有ACS功能的RCiEP（PF和VF）的访问都是第一次

在进一步解码和处理之前，由根复合体中的翻译代理（TA）进行处理。此类根复合体处理的细节不在本规范的范围内。

表9-26 ACS能力寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 5 | **ACSP 2 P出口控制**-如果支持设备内的对等事务，则对于非RCiEP的功能，要求为1b。 | 基地 | 基地 |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0



**执行说明**

支持功能直接分配的系统中的访问控制

通用VI通常为每个SI和VI本身具有单独的地址空间。如果这样的VI也

支持将功能直接分配给SI，由直接分配的功能发出的未转换存储器请求事务在相关SI内运行的软件的完全控制下，通常

引用与该SI相关联的地址空间。相反，由主机（MMIO请求）和由未直接分配的功能发出的存储器请求事务在VI的控制下，并且通常

引用一个或多个系统地址空间（例如， PCIe物理地址空间、与VI相关联的地址空间或与某个指定SI相关联的地址空间）。一般用途的VI不应

在这些不同的地址空间之间建立依赖性。因此，这些地址空间可以自由地重叠，这可能导致交换机对TLP的非预期路由。例如，上游内存请求

如果请求中的地址落在与直接分配的功能相关联的MMIO地址区域内，则由直接分配的功能发起并且旨在用于主存储器的TLPs可以改为被路由到下游端口。

下游端口。这种非预期的路由对SI和/或VI稳定性和完整性构成威胁。

为了防止这种情况，强烈建议供应商在支持具有直接功能分配的通用VI的平台中实现ACS。这种支持应包括：

· 在位于TA以下的交换机或根联合体中，ACS支持级别应遵循

本文档中为实现ACS扩展功能结构的下游交换机端口制定了指导原则。

注：位于TA上方的组件只能看到转换的内存请求，但此问题不适用于这些组件。

· 在能够进行对等事务的SR-IOV设备中，需要ACS支持。

·在能够进行点对点交易的多功能设备中，强烈建议供应商使用ACS P2P出口控制来实现ACS。

此外，平台供应商应测试ACS的存在，并在直接分配功能之前，在从TA到功能的路径上的根复合体和交换机中启用ACS。如果函数是对等函数，

功能，ACS也应该在功能中启用

<9.3.7.7>替代RUNNGID解释扩展功能（ARI）更改

ARI不适用于RCiEP;所有其他具有SR-IOV功能的设备（包括至少一个PF的设备）应实现 每个功能的ARI扩展能力。PF和VF功能定义见第7.8.7节。2，[表9-27中注明的除外。](#bookmark365)

表9-27 ARI能力寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 0 | **MFVC功能组能力（M）**- 以下描述的附加要求。 | 基地 | 基地 |
| 1 | **ACS功能组能力（A）**- 下文所述的附加要求。 | 基地 | 基地 |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 十五点八分 | **下一个功能编号**-使用第一VF偏移（参见www.example.com[节9.3.3.10）](#bookmark254)和VF步幅（参见www.example.com[节9.3.3.11）定位VF](#bookmark260)。 | 基地 | 未定义 |

任何使用可选功能仲裁表实现MFVC功能并使用多个总线号的设备应将MVFC功能组启用（M）功能位实现为功能0中的1b。

任何使用可选出口控制矢量实现ACS功能并消耗多个总线号的设备应将ACS功能组启用（A）功能位实现为功能0中的1b。

<9.3.7.8>地址翻译服务扩展功能更改（ATS）

ATS支持在SR-IOV设备中可选。如果VF实现ATS功能，则其关联的PF必须实现ATS功能。VF中的ATS能力及其相关联的PF可以独立地被启用。

PF和VF功能定义见第10.5节。1，[表9-28](#bookmark366)和[表9-29中注明的除外。](#bookmark367) 显示为ATS的属性与地址转换服务（[第10章）中指定的属性相同](#bookmark368)。

表9-28 ATS能力寄存器

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | 来自ATS的PF和VF寄存器差异 | PF  属性 | VF  属性 |
| 四比零 | **队列深度**无效-VF的硬连线为0。共享PF输入队列的深度 | ATS | RO |

表9-29 ATS控制寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | 来自ATS的PF和VF寄存器差异 | PF  属性 | VF  属性 |
| 四比零 | **最小翻译单元（STU）**- VF硬连接至0PF值适用于所有VF。 | ATS | RO |

ATS行为在地址传输服务（[第10章](#bookmark369)）中指定。ATS请求的目标函数为

无效ATS响应将具有与目标功能匹配的路由ID字段。每个职能部门都必须

实现足够的排队，以确保它可以容纳来自TA的最大数量的未完成无效请求（使用输入或输出排队）。

然而，与PF相关联的所有VF共享PF中的单个输入队列。 为了实施无效流控制，TA必须确保对共享PF队列（针对PF及其相关联的VF）的未完成的无效请求的总数不超过PF无效请求深度字段中的值。

<9.3.7.9>MR-IOV变更

PF和VF不应包含MR-IOV能力。

如果存在，则MR-IOV功能包含在组件的其他功能中。详情参见[MR-IOV]。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.3.7.10>多播变更

多播支持在SR-IOV设备中是可选的。如果VF实现多播功能，则其关联PF必须

实现多播功能。PF和VF功能定义见第7.9.11节，[表9-30、](#bookmark370)[表9-31](#bookmark371)和[表9-32中注明的除外。](#bookmark372)

表9-30组播能力寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 五比零 | **MC\_Max\_Group**- PF值适用于所有VF。 | 基地 | RsvdP |
| 十三点八分 | **MC\_Window\_Size\_Requested**-PF值适用于所有VF。 | 基地 | RsvdP |
| 15 | **支持MC\_ECRC\_Regeneration\_S**-不适用于端点。 | 基地 | RsvdP |

表9-31多播控制寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位 | PF和VF寄存器与基极的差异 | PF | VF |
| 位置 |  | 属性 | 属性 |
| 五比零 | **MC\_FLOW\_Group**- PF值适用于所有VF。 | 基地 | RsvdP |

表9-32组播基地址寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF属性 | VF属性 |
| 五比零 | **MC\_Index\_Position**- PF值适用于所有VF。 | 基地 | RsvdP |
| 63：12 | **MC\_Base\_Address**- PF值适用于所有VF。 | 基地 | RsvdP |

<9.3.7.11>页面请求接口更改（PRI）

页面请求接口功能在地址转换服务（[第10章）中定义](#bookmark373)。

PF允许支持页面请求接口。PF的页面请求接口

相关的VF PF允许实现页面请求接口功能，VF不得实现该功能。

即使在PF和VF之间共享寻呼请求接口，它也在寻呼请求消息的寻呼机ID字段中发送请求功能的ID（PF或VF），并且在所得到的PRG响应消息的目的地设备ID字段中期望请求功能的ID。

<9.3.7.12>动态功率分配更改（DPA）

VF可以不实现动态功率分配能力。

VF的功率分配使用PF的DPA能力（如果实现）进行管理。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.3.7.13>TPH转换器（TPH）

TPH路由器扩展能力可能存在于VF、PF中，也可能两者都存在，也可能两者都不存在。如果与给定PF关联的任何VF包含该功能，则与该PF关联的所有VF也必须支持TPH。

PF和VF中的所有TPH调压器扩展能力字段均按照第7.9.13节中的定义运行。

对于TPH解码器能力寄存器（偏移04h）中的字段，PF及其相关联的VF可以具有不同的值，但是与同一PF相关联的所有VF必须在所有字段中具有相同的值。

<9.3.7.14>PASID更改

允许端点设备支持PASID。单个功能的PASID配置

表示该设备的所有VF也使用该设备。 PF允许实现PASID功能，但VF不能实现它。

即使PASID配置在函数、PF和VF之间共享，设备也会在包含PASID的TLP的转发器ID字段中发送请求函数的ID（函数、PF或VF）。

<9.3.7.15>准备时间报告扩展功能和更改

就绪时间报告扩展能力可以存在于VF、PF中，或者两者都存在，或者两者都不存在。如果与给定PF相关联的任何VF包含该功能，则与该相同PF相关联的所有VF也必须支持就绪时间报告。

重置时间字段包含VF启用设置后所需的时间（参见[第9.6.1节）](#bookmark374)。DL\_Up时间字段为RsvdP。

与同一PF相关的所有VF应报告相同的时间值。

9.4 SR-IOV错误处理

SR-IOV设备使用第6.2节中定义的错误报告机制。定义为非功能特定的错误仅记录在PF中。

第6.2节定义了两种错误报告范例：基线功能和高级错误报告功能。所有PCI Express设备都需要基线错误报告功能，并定义了最低错误报告要求。高级错误报告功能是可选的，它是为更强大的错误报告而定义的，并通过特定的PCIExpress功能结构实现。

9.4.1基线误差报告

所有SR-IOV设备都必须支持基线错误报告能力，并进行一些修改以实现降低成本和实现复杂性的目标。

这些控制位仅在PF中有意义。当决定生成错误消息时，VF应使用相关PF中的err或报告控制位。

以下字段是VF中的RsvdP

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

·命令寄存器（参见[第9.3.4.1.3节）](#bookmark293)。SERR#启用

. 奇偶校验错误响应

· 设备控制寄存器（参见www.example.com[部分9.3.5.4）](#bookmark299)。可更正报告启用

. 非致命报告启用。致命报告启用

. 不支持的请求（UR）报告启用

每个VF应实现一种机制，以提供独立于任何其他功能的错误状态。 这对于为特定于函数的错误提供SI隔离是必要的。

必须在每个VF中实现以下基线错误报告

·状态寄存器（见[第9.3.4.1.4节）](#bookmark295)。主数据奇偶校验错误

. 已发出目标中止信号。收到目标中止。收到MasterAbort。 信号系统错误。检测到奇偶校验错误

· 设备状态寄存器（参见www.example.com[部分9.3.5.5）](#bookmark306)。检测到可纠正的错误

. 检测到非致命错误。检测到致命错误

. 检测到不支持的请求

每个VF在发出错误信号时应使用其自己的路由ID。

9.4.2高级错误报告

高级错误报告功能是可选的。如果AER未在PF中实施，则不得在相关VF中实施。如果AER在PF中实现，则在VF中是可选的。

第6.2.4节将错误分类为功能特定和非功能特定。每个实现高级错误报告功能的VF都必须为特定于函数的错误维护自己的错误报告状态。

<9.4.2.1>VF标头日志

在VF中实现AER的设备可以在与单个PF相关联的VF之间共享报头日志寄存器。有关详细信息，请参见www.example.com[9.4.2.10部分](#bookmark376)。

PF的报头日志寄存器独立于其相关联的VF，并且必须使用专用存储空间来实现。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

当实现一组精简的报头日志寄存器时，函数可能没有空间来记录与错误相关联的报头。在这种情况下，该功能应更新不可纠正的错误状态寄存器和高级错误

第6.2.4节要求的功能和控制寄存器;但是，当读取标头日志寄存器时，它应返回全1，以指示溢出条件，并且没有记录标头。

<9.4.2.2>高级错误报告功能更改

图7-122描述了AER扩展能力结构。

<9.4.2.3>高级错误报告扩展功能报头更改（偏移量00h）

此寄存器包含PCI Express扩展功能ID、功能版本和下一个功能偏移。 这些字段没有变化，在www.example.com部分中进行了描述7.8.4.1。

<9.4.2.4>无法纠正的错误状态寄存器更改（偏移04h）

不可纠正错误状态寄存器指示各个错误的错误检测状态。定义为非功能特定的错误记录在PF中。只有功能特定的错误记录在VF中。

除表9-33中注明的地方外，PF和VF功能在部分7.8.4.2[中定义。](#bookmark377)

表9-33不可纠正的错误状态寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 4 | 数据链路协议错误状态 | 基地 | 0b |
| 5 | 意外停机错误状态 | 基地 | 0b |
| 13 | 流控制协议错误状态 | 基地 | 0b |
| 17 | 接收器溢出状态 | 基地 | 0b |
| 18 | 格式错误的TLP状态 | 基地 | 0b |
| 19 | 错误状态 | 基地 | 0b |

<9.4.2.5>无法纠正的错误掩码寄存器更改（偏移08h）

除表9-34中所述外，PF和VF功能在www.example.com上的第节中定义7.8.4.3[。](#bookmark378)对于标记为RsvdP的VF字段，PF设置应用于VF。对于标记为0b的VF字段，错误不适用于VF。

表9-34无法纠正的错误掩码寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 4 | 数据链路协议错误掩码 | 基地 | 0b |
| 5 | 意外停机错误掩码 | 基地 | 0b |
| 12 | 中毒TLP收到的邮件 | 基地 | RsvdP |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 13 | 流量控制协议错误掩码 | 基地 | 0b |
| 14 | 完成咨询 | 基地 | RsvdP |
| 15 | 完成程序中止掩码 | 基地 | RsvdP |
| 16 | 意外完成掩码 | 基地 | RsvdP |
| 17 | 接收器溢出掩码 | 基地 | 0b |
| 18 | 变形TLP掩码 | 基地 | 0b |
| 19 | ECRC错误掩码 | 基地 | 0b |
| 20 | 不支持的请求错误掩码 | 基地 | RsvdP |
| 21 | ACS违规掩码 | 基地 | RsvdP |

<9.4.2.6>无法纠正的错误严重性寄存器更改（偏移0Ch）

除表9-35中注明的地方外，PF和VF功能在www.example.com上的Secti中定义7.8.4.4[。](#bookmark379)对于标记为RsvdP的VF字段，PF设置应用于VF。对于标记为0b的VF字段，错误不适用于VF。

表9-35不可纠正的错误严重性寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 4 | 数据链路协议错误严重性 | 基地 | 0b |
| 5 | 意外停机错误严重性 | 基地 | 0b |
| 12 | 中毒TLP接收严重度 | 基地 | RsvdP |
| 13 | 流控制协议错误严重性 | 基地 | 0b |
| 14 | 完成错误或严重性 | 基地 | RsvdP |
| 15 | 完成程序中止错误严重性 | 基地 | RsvdP |
| 16 | 意外完成错误严重度 | 基地 | RsvdP |
| 17 | 接收器溢出严重性 | 基地 | 0b |
| 18 | 变形TLP严重度 | 基地 | 0b |
| 19 | ECRC错误严重性 | 基地 | 0b |
| 20 | 不支持的请求错误严重性 | 基地 | RsvdP |
| 21 | ACS违规严重性 | 基地 | RsvdP |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.4.2.7>可纠正的错误状态寄存器更改（偏移10h）

可纠正错误状态寄存器指示各个可纠正错误的错误检测状态。定义为非功能特定的错误记录在PF中。只有功能特定的错误记录在VF中。

除表9-36中注明的地方外，PF和VF功能在部分7.8.4.5[中定义。](#bookmark380)

表9-36可纠正的错误状态寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 0 | 接收器错误状态 | 基地 | 0b |
| 6 | 不良TLP状态 | 基地 | 0b |
| 7 | 错误的DLLP状态 | 基地 | 0b |
| 8 | RENOW\_NUM滚动状态 | 基地 | 0b |
| 12 | 重播计时器状态 | 基地 | 0b |
| 15 | 标题日志溢出状态  如果VF实现报头日志共享（参见[第9.4.2.1](#bookmark375)），则此位硬连线至0b。 | 基地 | 基地/0 b |

<9.4.2.8>可更正错误掩码寄存器更改（偏移14小时）

除表9-37中注明的情况外，PF和VF功能在www.example.com上的Secti中定义7.8.4.6[。](#bookmark381)对于标记为RsvdP的VF字段，PF设置应用于VF。

表9-37可纠正错误掩码寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 0 | 接收器错误掩码 | 基地 | RsvdP |
| 6 | TLP掩码错误 | 基地 | RsvdP |
| 7 | DLLP掩码错误 | 基地 | RsvdP |
| 8 | RESIDO\_NUM翻转掩码 | 基地 | RsvdP |
| 12 | 重放定时器屏蔽 | 基地 | RsvdP |
| 13 | 建议性非致命错误掩码 | 基地 | RsvdP |
| 15 | 标头日志溢出掩码-如果VF实现标头日志共享（参见www.example.com[部分9.4.2.1](#bookmark375)），则此位为RsvdP。 | 基地 | Base/RsvdP |

<9.4.2.9>高级错误功能和控制寄存器更改（偏移18 h）

除表9-38中所述外，PF和VF功能在www.example.com上的第节中定义7.8.4.7[。](#bookmark382)对于标记为RsvdP的VF字段，PF设置应用于VF。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

表9-38高级错误能力和控制寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 6 | ECRC生成启用 | 基地 | RsvdP |
| 8 | ECRC检查启用 | 基地 | RsvdP |
| 9 | 支持多标题记录-如果VF实现标题日志共享（[请参见](#bookmark375)  [9.4.2.1](#bookmark375)），该位被硬连线到0b。 | 基地 | 基地/0 b |
| 10 | 多标题记录启用-如果VF实现标题日志共享（[请参见](#bookmark375)  [9.4.2.1](#bookmark375)），该位是RsvdP。 | 基地 | Base/RsvdP |
| 11 | TLP前缀日志存在-如果VF实现报头日志共享（参见[第www.example.com节9.4.2.1](#bookmark375)），则当报头日志由于溢出条件而包含全1时，此位为0 b。 | 基地 | 基础（参见 描述） |

<9.4.2.10>标头日志注册器更改（偏移量1Ch）

报头日志寄存器捕获对应于检测到的错误的TLP的报头另请参见www.example.com[部分9.4.2.1。](#bookmark375)

在VF中实现AER的设备可以在与单个PF相关联的VF之间共享报头日志寄存器。共享头日志必须至少有一个头的存储。

PF的报头记录寄存器独立于其相关联的VF，并且必须使用专用存储空间来实现。

当在VF中检测到错误时，应按照第6.2节的规定记录错误。如果一组共享的标头日志

寄存器，则VF可能没有空间来记录标头。在这种情况下，VF应更新其不可纠正的

错误状态寄存器和高级错误功能和控制寄存器，如第6.2节所要求;但是，当读取VF的报头日志寄存器时，它应返回所有1以指示溢出条件。

当VF的第一个错误指针有效时，VF所界定

第6.2节，当不可纠正错误状态寄存器的相应位置位时，第一个错误指针寄存器有效。 当标题日志条目被锁定时，其他错误不应覆盖此条目或任何其他VF的锁定条目。当标题条目解锁时，应可用于记录共享标题日志的任何VF的新错误。

PF和VF功能在第7节中定义。8.4.8，[表9-39中注明的除外。](#bookmark383)

表9-39报头日志寄存器更改

|  |  |  |  |
| --- | --- | --- | --- |
| 位 | PF和VF寄存器与基极的差异 | PF | VF |
| 位置 |  | 属性 | 属性 |
| 一百二十七比零 | 与错误相关的TLP报头（附加要求如上所 | 基地 | 基地 |

<9.4.2.11>根错误命令寄存器更改（偏移2Ch）

此寄存器不适用于设备。

<9.4.2.12>根错误状态注册表更改（偏移30小时）

此寄存器不适用于设备。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<9.4.2.13>错误源标识寄存器更改（偏移34h）

此寄存器不适用于设备。

<9.4.2.14>TLP前缀日志寄存器更改（fset38 h）

对于PF和VF，如果支持端-端TLP前缀，则实现此寄存器。

对于VF，如果实现了一组共享的报头日志寄存器（[第9.4.2节），1](#bookmark375)），由于溢出条件，当标题日志包含所有1时，此寄存器的内容未定义。

9.5SR-IOV中断

具有SR-IOV能力的设备利用第6.1节中定义的相同的GPRS信令机制。

9.5.1应急机制

有三种通知中断的方法：

·INTx·MSI

·MSI-X

PF可以实现INTx。VF不能实现INTx。如果请求中断资源，PF和VF应实现MSI或MSI-X或两者。每个PF和VF都必须实现自己独特的中断功能。

<9.5.1.1>MSI中断

MSI能力和PF及VF功能在第7.7节中定义，[表9-40中注明的除外。](#bookmark384)

表9-40 MSI功能：消息控制

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF属性 | VF属性 |
| 8 | 支持每矢量掩蔽 | 1b | 1b |

<9.5.1.2>MSI-X中断

MSI-X能力在第7.7节中定义，如[图9-24所示。](#bookmark385)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

31 16 15 8 7 3 2 0

|  |  |  |  |
| --- | --- | --- | --- |
| 消息控制 | 下一指针 | 能力ID | |
| 表偏移 | | | 表BIR |
| PBA偏移 | | | PBABIR |

图9-24 MSI-X功能

CP+00hCP+04hCP+08h

粤ICP备16038883号-1

PF和VF功能与第7.7.2节中定义的功能相同。

注意，对于VF，表偏移和PBA偏移值与VF的存储器地址空间有关。

<9.5.1.3>地址范围

如果映射用于MSI-X表或MSI-X PBA的地址空间的BAR还映射未被

与MSI-X结构、位置（例如，对于CSR），在其他地址空间中使用的地址空间不能自然共享任何 与MSI-X结构所在的系统页面大小地址范围对齐。MSI-X表和MSI-X PBA 允许共同驻留在自然对齐的系统页面大小地址范围内，尽管它们不得彼此重叠。

9.6SR-IOV电源管理

本节定义PCI Express SR-IOV电源管理功能和协议。

电源管理能力是第5章所述的PF所必需的。 对于VF，电源管理功能是可选的。

9.6.1 VF设备电源管理状态

如果VF未实现电源管理功能，则VF的行为就像其已编程为其关联PF的等效电源状态。

如果VF实现了电源管理电容能力，则除第9.6.4节中所述[外，第7.5节中定义了该功能。](#bookmark386)

如果VF实现电源管理功能，则如果PF被置于较低的

功率状态比VF。 软件应通过在降低所有VF的相关PF的功率状态之前将其置于较低功率状态来避免这种情况。

当VF已完成其内部初始化且VF的总线主机使能位被设置（参见[第9.3.4.1.3节](#bookmark293)）或SR-IOV控制器中的VF MSE位（参见[第www.example.com节9.3.3.3）](#bookmark212)扩展功能被设置时，处于D 0状态的VF处于D 0活动状态。如果出现以下任何情况，则VF

发生：

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

·VF已成功响应（不返回CRS）配置请求。

·向VF发出FLR后，以下情况之一为真：自发布FLR以来，至少已过了1.0 s。

. The VF supports Function Readiness Status and, after the FLR was issued, an FRS Message from the VF with Reason Code FLR Completed has been received.

. 自发布FLR以来，至少已经过了FLR时间。FLR时间是（1）与VF相关联的准备时间报告功能中的FLR时间值，或者（2）由系统确定的值

软件/固件161.

·在PF中设置VF启用后，至少有以下一项为真：自设置VF Enable（VF启用）以来，至少已过去1.0 s。

. PF支持功能就绪状态，并且在设置VF启用后，收到来自PF的FRS消息（原因代码VF启用）。

·在将VF从D3热转换到D0之后，以下中的至少一个为真：

. 发出进入D0的请求后，至少已经过了10 ms。

. VF支持功能就绪状态，并且在发出进入D0的请求后，收到来自VF的FRS消息，原因代码为D3热到D0转换已完成。

. 自发出进入D0的请求以来，至少已经过了D3热到D0时间。D3热至D0时间为 （1）与VF相关的Readiness时间报告功能中的D3Hot to D0Time，或

(2)该值由系统软件/固件162确定。

9.6.2 PF设备电源管理状态

PF的功率管理状态（D状态）对其相关联的VF具有全局影响。如果VF没有实现功率管理能力，则它就好像处于其相关联的PF的等效功率状态。

如果VF实现电源管理功能，则如果PF被置于较低的

功率状态比VF。 软件应通过在降低所有VF的相关PF的功率状态之前将其置于较低功率状态来避免这种情况。

当PF被置于D3热状态时

·如果No\_Soft\_Reset位为Clear，则PF在D3 Hot到D 0转换时执行内部重置，并且其所有配置状态都恢复为默认值。

注：重置PF将重置VF使能，这意味着VF不再存在，并且在D3热到D0转换完成后，任何VF特定上下文都将丢失。

· 如果No\_Soft\_Reset位置位，则不会发生内部复位。 SR-IOV扩展能力保持状态，并且相关联的VF保持启用。

当PF被置于D3冷状态时，VF不再存在，任何VF特定上下文丢失，并且PME事件只能由PF发起。



161.例如，ACPI表。

162.例如，ACPI表。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

执行说明

强烈建议使用No\_Soft\_Reset

强烈建议在多功能器件的所有功能中设置No\_Soft\_Reset位。 本建议适用于PF。



9.6.3链路电源管理状态

VF电源状态不影响链路电源状态。

链路电源状态仅由PF中的设置控制，而与VF

9.6.4 VF电源管理能力

下表列出了PF和VF电源管理功能的要求

除[表9-41](#bookmark386)和[表9-42中注明的情况外，PF和VF功能在第7.5节中定义。](#bookmark387)

表9-41 SR-IOV电源管理控制/状态（PMCSR）

|  |  |  |  |
| --- | --- | --- | --- |
| 位  位置 | PF和VF寄存器与基极的差异 | PF  属性 | VF  属性 |
| 十四点十三分 | Data\_Scale | 基地 | 00b |
| 十二点九分 | 数据\_选择 | 基地 | 0000B |
| 3 | **No\_Soft\_Reset**-如果VF实现电源管理功能，则此字段的VF值必须与相关PF值相同。 | 基地 | 基地 |

表9-42 SR-IOV电源管理数据寄存器

|  |  |  |  |
| --- | --- | --- | --- |
| 比特位置 | PF和VF寄存器与基极的差异 | PF属性 | VF属性 |
| 七比零 | 数据 | 基地 | 00000000b |

9.6.5 VF紧急功率降低状态

如果VF中的Emergency Power Reduction Supported字段为非零，则该VF进入并退出Emergency Power

与关联PF同时处于还原状态。软件可以使用PF中的紧急功率降低检测位来模拟VF中的相应位。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

**ATS规范**

**10.1ATS体系结构概述**

10.

大多数现代系统架构都提供了从DMA（总线主控）I/O功能转换地址的功能。在许多实现中，

一种假设CPU和I/O看到的物理地址空间

功能等同。而在其他情况下，情况并非如此。编程到

I/O功能是由根联合体（RC）处理的“句柄”。这种处理的结果通常是

转换为中央复合体中的物理存储器地址。通常，处理包括访问权限检查，以确保DMA功能被允许访问所引用的存储器位置。

进行DMA地址翻译的目的各不相同，包括：

· 限制“损坏”或错误编程的DMA I/O功能的破坏性

· 提供分散/聚集

·重定向消息信令中断的能力（例如，MSI或MSI-X）到不同的地址范围，而无需与底层I/O功能协调

地址空间转换（32位I/O功能到更大的系统地址空间）

·虚拟化支持

不管动机如何，主机系统中DMA地址转换的存在对DMA访问具有一定的性能影响。

根据具体实现，DMA访问时间可能会因解析实际物理地址所需的时间而显著延长。如果实现需要访问驻留在主存中的转换表，

访问时间可以显著长于未翻译访问的时间。此外，如果每笔交易

需要多次存储器访问（例如， 对于表遍历），则存储器事务速率（即，与DMA相关联的开销）可能很高。

为了减轻这些影响，设计通常在执行地址转换的实体中包括地址转换缓存。

翻译.在CPU中，地址转换缓存通常被称为转换后备缓冲器

（TLB）。对于I/OTA，术语地址转换缓存或ATC用于将其与CPU使用的转换缓存区分开来。

虽然TLB和ATC之间有一些相似之处，但也有重要的区别。 ATLB服务于名义上一次运行一个线程的CPU的需要。然而，ATC通常处理来自多个I/O的

函数，每个函数都可以被认为是一个单独的线程。这种差异使得ATC的规模很难根据成本模型和预期的技术重用范围广泛的系统配置。

本说明书中描述的机制允许I/O设备参与转换过程并为其自己的存储器访问提供ATC。在设备中安装ATC的好处包括：

·通过分配地址转换缓存责任来减轻TA资源压力的能力（降低TA内“thrash hing”的概率

· 使ATC设备对系统

· 通过将预翻译请求发送到中央综合大楼，确保最佳访问延迟的可能性

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

该规范将提供允许PCIe设备与TA结合使用的互操作性，但是TA及其地址转换和保护表（ATPT）被视为特定于实现并且在规范之外。 本规范的范围。虽然可以在其他PCIe组件内实现ATS，但是本规范限于PCIe设备和PCIe根复合体集成端点（RCiEP）。

[图10-1](#bookmark388)显示了一个带有TA和ATPT的示例平台，以及一组PCIe设备和带有集成ATC的RC集成端点。ATA和ATPT是特定于实现的，可以是不同的，也可以是集成的

在给定的系统设计中。

|  |
| --- |
| 翻译代理（TA） |

|  |
| --- |
| 根端口（RP） |

|  |
| --- |
| 根端口（RP） |

开关

|  |
| --- |
| 存储器 |

|  |
| --- |
| 地址转换和保护表（ATPT） |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 根复合体（RC）   |  |  | | --- | --- | | RC集成端点 |  | | ATC | |  | |

|  |  |
| --- | --- |
| PCIe设备 |  |
| ATC |
|  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | | PCIe设备   |  | | --- | | ATC | |
| PCIe设备 | |  |
| ATC |
|  |

粤ICP备05058888号-1

图10-1带有TA、ATPT和ATC元素的平台示例

10.1.1地址转换服务（ATS）概述

ATS章节提供了一组新的TLP和相关的语义。ATS使用设备163和根联合体（RC）之间的请求完成协议来提供翻译服务。此外，在

存储器读和存储器写TLP。新的AT字段使得RC能够确定给定的请求是否已经经由ATS协议被翻译。



163. 本章中对设备的所有引用同样适用于PCIe设备或根复合体集成端点。 ATS在需求、语义、配置、错误处理等方面并不区分这两种类型。从软件的角度来看，具有ATS能力的根联合体集成端点必须与具有ATS能力的非集成设备表现相同。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

[图10-2](#bookmark389)说明了ATS翻译请求操作的基本流程。

|  |
| --- |
| 翻译代理（TA） |

|  |
| --- |
| 存储器 |

|  |
| --- |
| 地址转换和保护表（ATPT） |

|  |
| --- |
| 根复合体（RC） |

|  |
| --- |
| 根端口（RP） |

ATS

ATS

请求

完成

|  |  |
| --- | --- |
| PCIe设备 |  |
| ATC |
|  |

粤ICP备05058999号-1

图10-2 ATS转换请求/完成交换示例

在该示例中，功能特定的工作请求由单功能PCIe设备接收。该函数通过特定于实现的方法确定在其ATC内缓存翻译将是有益的。有 功能或软件在做出此类决定时可以考虑的因素数量;例如：

· 一种内存地址范围，它将在一段较长的时间内被频繁地访问，或者其相关的缓冲区内容需要很高的更新率

· 内存地址范围，如工作和完成队列结构，低延迟的数据缓冲区

通信、图形帧缓冲器、用于高速缓存功能特定内容的主机存储器等

鉴于设计和访问模式的多样性，没有单一的标准可以应用。

该功能生成ATS转换请求，该ATS转换请求通过PCIe层次结构向上游发送到RC，RC然后将其转发到TA。ATS转换请求使用与本文档中定义的相同的

规范.此外，在任何给定时间，多个ATS转换请求可以是未完成的;即，一个可以流水线 一个或多个TC上的多个请求。每个TC表示唯一的排序域并且定义必须由相关联的ATS传输完成使用的域。

在收到ATS翻译请求后，电信局长执行以下基本步骤：

1. 验证功能是否已配置为发出ATS转换请求。

2. 确定函数是否可以访问ATS转换请求所指示的内存，并具有相关的访问权限。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

3. 确定是否可以向函数提供转换。如果是，TA将发布该函数的翻译。

a. ATS需要支持各种页面大小，以适应各种ATPT和处理器实现。

I. 页面大小必须是2的幂，并且自然对齐。

二. 支持的最小页面大小为4096字节。需要ATS功能组件来支持此最小页面大小。

B. 必须告知职能部门需要支持的最小转换或无效大小，以便为职能部门提供优化其资源利用的机会。 最小转换大小必须为4096字节。

4. TA将请求的成功或失败传达给RC，RC生成ATS翻译完成并通过RP经由响应TLP发送到功能。

a. RC需要对每个ATS翻译请求至少一个ATS翻译完成进行基因评级;即，存在与请求的成功或失败无关的最低限度的1：1对应。

I. 成功的翻译可以导致每个请求一个或两个ATS翻译完成TLP。翻译完成度表示所涵盖的翻译范围。

二. RC可以流水线化多个ATS转换完成;即，RC可以返回多个ATS转换完成，并且这些ATS转换完成可以是相对于ATS转换请求的任何顺序。

三. RC需要使用与对应ATS转换请求相同的TC（业务类）来发送ATS转换完成。

B. 请求的地址可能无效。RC必须发出翻译完成通知，表明所请求的地址无法访问。

当功能接收到ATS翻译完成并更新其ATC以反映翻译或注意到翻译不存在时。该函数继续处理其工作请求，并生成后续的

基于完成的结果，使用转换的地址或未转换的地址来请求。

a. 与读取完成类似，需要一个功能来为每个完成分配资源空间，而不会导致PCIe链路上的背压。

B. 需要一个函数来丢弃可能“过时“的翻译完成。由于各种原因，可能会发生过时的翻译完成。

可以看出，ATS翻译请求和翻译完成处理在概念上是相似的，并且在 在许多方面，与PCIe读请求和读完成处理相同。这是为了降低设计复杂性，并简化ATS到现有和新的基于PCIe的解决方案中的集成。记住这一点，ATS 要求：

·ATS功能组件必须与[PCIe-1.1]兼容组件互操作。

·ATS通过新的能力和相关的配置结构启用。 要启用ATS，软件必须检测此功能并启用功能以发布ATSTLP。如果未启用某个功能，则该功能

要求不发出ATS转换请求，并要求发出TLPAT字段设置为“未转换“的所有DMA读写请求

·使用基于地址或路由器ID（RID）路由来路由ATSTLP。

·ATSTLP需要使用与本规范中指定的相同的

·要求ATSTLP未经修改地流过符合[PCIe-1.1]的交换机。

·允许函数混合翻译和未翻译的请求。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

·ATS事务被要求不依赖于存储器请求的地址字段来传送超出其当前使用的附加信息，如PCI-SIG所定义的。

执行说明

地址范围重叠

未翻译的地址范围和已翻译的地址范围很可能重叠，可能是全部重叠。这不是一个 这是ATS的要求，但可能是TA上的实现约束，使得存储器请求将被正确路由。



与上一个示例相反，[图10-3](#bookmark390)说明了一个示例多功能设备。在该示例设备中，

有三个功能。[图10-3中需要注意的要点是：](#bookmark391)

· 每个ATC都与一个功能相关联。每个ATS功能必须能够源和接收至少一个ATS转换请求或转换完成类型。

· 每个ATC都是按功能配置和访问的。多功能设备不需要在每个功能上实现ATS。

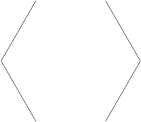
· 如果ATC实现在一组功能之间共享资源，则逻辑行为需要与完全独立的ATC实现一致。

|  |
| --- |
| PCIe端口 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 功能1 | |  | | --- | | ATC1 | | |  | | --- | | 物理 资源1 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 功能2 | |  | | --- | | ATC2 | | |  | | --- | | 物理 资源2 | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 功能3 | |  | | --- | | ATC3 | | |  | | --- | | 物理 资源3 | |



|  |
| --- |
| 配置资源 |

内部路由

PCIe设备

粤ICP备05059222号

图10-3带ATC per功能的多功能设备示例

与设备内的功能数量无关，需要以下

·要求功能不发出具有AT字段设置的任何TLP，除非TLP内的地址通过ATS转换请求和转换完成协议获得。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

· 要求每个ATC仅使用ATS协议进行填充;即，ATC内的每个条目必须响应于发出针对给定地址的ATS翻译请求的功能而经由ATS翻译完成来填充。

· 除非通过ATS协议，否则不能修改每个ATC。即：

◦ 主机系统软件不能修改ATC，除非通过本协议中定义的协议。

除非在ATC中使一个或多个翻译无效，否则不应使用规范。设备或功能复位

将是由软件执行以改变ATC的内容的操作的示例，但是复位仅允许使条目无效而不允许修改它们的内容。

◦ 主机系统软件不得使用器械上执行的软件修改ATC。

当TA确定功能不应再在其ATC内保持警报时，TA启动ATS无效协议。无效协议由单个无效请求和一个或多个无效完成组成。

|  |
| --- |
| 存储器 |

|  |
| --- |
| 翻译代理（TA） |

|  |
| --- |
| 根端口（RP） |

|  |
| --- |
| 步骤3. ATS无效完成[TC= 0，ITAGV=1000b，CC=1] |

|  |
| --- |
| 根复合体（RC） |

|  |
| --- |
| 地址转换和保护表（ATPT） |

|  |
| --- |
| 步骤1.ATS无效请求  [未转换地址，TC = 0，ITAG= 3] |

PCIe设备 ATC

|  |
| --- |
| 步骤2. 刷新匹配的ATC条目，排出或丢弃冲突的读取。 |

粤ICP备05059055号

图10-4具有单个失效请求和完成的失效

如[图10-4](#bookmark392)所示，ATS失效协议中基本上有三个步骤

1. 系统软件更新TA使用的表格中的条目。表更改后，TA

确定转换应该在ATC中被无效，并且发起无效请求TLP，该无效请求TLP从RP被发送到示例单功能设备。Invalidate Request发送一个 未转换的地址范围、TC和用于关联无效完成的RP唯一标记 无效请求

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

2. 该函数接收无效请求并使所有匹配的ATC条目无效。函数不是

要求在收到无效请求后立即刷新所有挂起的请求。如果事务在等待发送的队列中，则函数不必从队列中删除请求，即使

这些事务使用被无效的

a. 要求函数在所有未完成的读操作完成之前，

引用关联翻译地址的请求或翻译请求已失效或无效。

B. 需要一个函数来确保对RC的无效完成指示将在使用“陈旧”地址的任何先前发布的写入之后到达RC。

3. 当一个功能确定所有使用的翻译地址是完整的，它发出一个或多个ATS无效完成。

a. 对于可能引用了无效范围的每个TC，都会发出一个无效完成。这些完成充当刷新机制，以确保清除层次结构中的任何正在进行的

可能包含对转换后地址的引用的事务。

I. 所需完成的数量在每个无效完成中进行了传达。ATA或RC实现可以维护一个计数器，以确保所有Invalidate

在收到完成后，才认为翻译不再使用。

二. 如果发送了多个无效完成，则每个TC中发送的无效完成在第10.3.2节中详述的字段中必须相同[。](#bookmark393)

B. 无效完成包含来自无效请求的ITAG，以使RC能够关联无效请求和完成。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |
| --- |
| 存储器 |

|  |
| --- |
| 翻译代理（TA） |

|  |
| --- |
| 根端口（RP） |

|  |
| --- |
| 步骤3. ATS无效完成[TC= 0，ITAGV=1000b，CC= 2] |

|  |
| --- |
| 步骤4. ATS无效完成[TC=1，ITAGV=1000b，CC= 2] |

|  |
| --- |
| 根复合体（RC） |

|  |
| --- |
| 地址转换和保护表（ATPT） |

|  |
| --- |
| 步骤1.ATS无效请求  [未转换地址，TC = 0，ITAG= 3] |

|  |
| --- |
| 步骤2. 刷新匹配的ATC条目，排出或丢弃冲突的读取。 |

PCIe设备 ATC

粤ICP备05059991号-1

图10-5单个无效请求和多个无效完成

10.1.2页面请求接口扩展

ATS改进了基于DMA的数据移动的行为。相关的页面请求接口（PRI）提供

通过允许DMA操作被启动而不需要所有要移入或移出系统存储器的数据被固定，这提供了额外的优点。164与固定存储器相关联的开销可能是适度的，但负面影响可能是

从可分页池中移除大部分存储器对系统性能的影响可能是显著的。

PRI在功能上独立于ATS的其他方面。也就是说，支持ATS的设备不需要支持PRI，但是PRI依赖于ATS的能力。

可以构造智能I/O设备，以便充分利用更动态的内存接口。固定将始终

从设备的角度来看，具有最佳的性能特征-它希望访问的所有内存都保证存在。然而，保证设备可能触及的所有存储器的驻留可能是有问题的，并且迫使主机上的设备感知的次优水平。允许设备更独立地操作（当它需要不存在的存储器资源时，页面错误）提供了设备和主机之间的更高级别的耦合。165

用于利用页面请求接口的机制是非常特定于设备的。 作为这样一个接口可以提高整体系统性能的模型的一个例子，让我们检查一个高速LAN设备。这样的



164.锁定在适当的位置，这样它就不能被系统的动态分页机制换出。

165. 另一种方法是在设备和它的驱动程序之间提供一个专用接口，用于传递设备状态，以便驱动程序可以确保固定内存资源的可用性。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

设备知道它的突发速率，并且只需要具有与它在某个量内可以接收的一样多的可用于入站数据的物理缓冲器空间。可以使未固定的虚拟存储器页的向量对设备可用，

设备然后根据需要请求保持其突发窗口。这可以最大限度地减少设备所需的内存占用并简化与主机的接口，而且不会对性能产生负面影响。

分页的能力，回避了页表状态标志管理的问题。 典型TA将标志（例如， 脏和访问指示）。如果没有关于如何管理映射到函数的页面的任何额外提示，则此类TA将需要保守地假设，当它们授予函数读取或 写一个页面，该函数将使用该权限。 这样的可写页面需要在其转换后的地址可用于函数之前被标记为脏。

对于不支持分页的函数来说，这种保守的写权限获取时损坏行为通常不是一个重要问题，因为分页是固定的，将干净页面保存到内存的成本很少。但是，如果所有可写页面都被视为

脏的，因为这样的函数在不固定它们的可访问存储器占用空间的情况下操作，并且可能发出用于性能的推测性页面请求。在这样的系统中保存干净页面（而不是丢弃它们）的成本可能会降低其他有吸引力的分页技术的价值。这可能会导致重大的性能问题和风险功能 在后备存储器无法写入的情况下出现的问题，如CD-ROM。

翻译请求中的NoWrite（NW）标志表示函数愿意将其使用限制为仅读取页面，而不受原本授予的访问权限的影响。

如果设备选择通过发出具有NW标志设置的转换请求来请求仅读访问，并且稍后确定它需要写入页面，则设备必须发出新的转换请求。

在接收到具有NW标志Clear的转换请求时，TA被允许将相关联的页面标记为脏。是 强烈建议函数不要发出这样的请求，除非它们被给予明确的写权限。写许可的示例是主机向函数发出命令以从存储设备加载数据，

将数据写入内存。

10.1.3进程地址空间ID（PASID）

某些TLP可以可选地与进程地址空间ID（PASID）相关联。此值使用

PASIDTLP前缀。 PASIDTLP前缀在第6.20节中定义。PASIDTLP前缀允许用于：

· 内存请求（包括未翻译的AtomicOp请求）与未翻译的缓存

·地址翻译要求

· 页面请求消息

·ATS失效请求

· PRG响应消息

第6.20节定义了未转换内存请求的PASID TLP前缀的用法。 本节介绍其余TLP的PASIDTLP前缀。

当请求没有PASIDTLP前缀时，未转换地址表示与转换器ID相关联的地址空间。

当请求具有PASIDTLP前缀时，未转换地址表示与转换器ID和PASID值两者相关联的地址空间。

当响应具有PASIDTLP前缀时，PASID值反映与相应请求相关联的地址空间。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

每个函数都有一组独立的PASID值。 PASID字段是20位宽，然而有效宽度受根复合体（TA）支持的宽度和函数（ATC）支持的宽度中的较小者约束。PASID值的未使用高位必须为0b。

对于存在虚拟中介（VI）的系统中的端点，

带有关联PASID的客户端通常用于表示客户虚拟客户端（GVA）和未翻译客户端

不与PASID关联的密码表示访客物理密码（GPA）。TA可以被设计成 VI管理用于执行从GPA到已翻译的转换的表，而各个Guest操作系统管理用于执行从GVA到GPA的转换的表。当使用 相关联的PASID，TA执行两种转换并返回结果转换的地址（即，GVA至GPA

然后是GPA到指定地址）。ATC看不到中间GPA值。

当ATC使高速缓存的GPA映射无效时，它使GPA映射无效并且还使ATC中的所有GVA映射无效。当GPA失效完成时，VI可以安全地从Guest中

操作系统VI不需要知道哪些GVA映射涉及GPA映射。

10.2ATS翻译服务

ATA负责翻译。ATC可以缓存这些翻译。如果ATC通过PCIe与TA分离，则来自ATC的内存请求将需要能够指示是否转换了transaction中的地址。的

在本节中描述了对存储器事务的修改，以及用于在远程ATC和中央TA之间传送翻译的事务。

10.2.1具有地址类型的

具有ATC的函数可以发送包含转换或未转换地址的存储器读/写请求。如[图10-6](#bookmark394)和[图10-7](#bookmark395)所示，地址类型（AT）字段用于指示

出现在请求头中。

字节0→字节4→字节8→字节12 →

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0  76543210 | | +1  76543210 | | | | | | +2  76543210 | | | | | | +3  76543210 | | |
| Fmt  0x1 | 类型  00000 | T9 | TC | T8 | ATTR | LN | 日 | TD | EP | Attr | 在 | 长度 | | | | |
| 请求者ID | | | | | | | | 标签[7：0] | | | | | 最后DW BE | | 第一个DW BE | |
| 地址[63] | | | | | | | | | | | | | | | | |
| 地址[31：2] | | | | | | | | | | | | | | | | R |

图10-6带有64位地址的内存请求报头

字节0→字节4→字节8→

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0  76543210 | | +1  76543210 | | | | | | +2  76543210 | | | | | +3  76543210 | | |
| Fmt  0x0 | 类型  00000 | T9 | TC | T8 | ATTR | LN | 日 | TD | EP | Attr | 在 | 长度 | | | |
| 请求者ID | | | | | | | | 标签[7：0] | | | | | 最后DW BE | 第一个DW BE | |
| 地址[31：2] | | | | | | | | | | | | | | | R |

图10-7带有32位地址的内存请求报头

请求中的AT字段是本规范的早期版本中的保留字段的重新定义。不实现ATC的功能将继续将AT字段设置为其定义的保留值（00b）。实现ATC的功能将设置表10中列出的AT字段[。](#bookmark396)

表10-1地址类型（AT）字段编码

|  |  |  |
| --- | --- | --- |
| AT[1：0]编码 | 助记 | 意义 |
| 00b | 非翻译 | ATA可以将地址视为虚拟地址或物理地址。 |
| 01b | 翻译请求 | TA将返回请求的地址字段中包含的地址的翻译 建成 此值仅对显式翻译请求有意义（参见[第10.2.2节）](#bookmark397)。如果TA在存储器中接收到AT字段设置为01b的TLP，则TA将发信号通知不支持的请求（UR  除了内存读取之外的请求 |
| 10b | 翻译 | 交易中的地址已由ATC翻译。如果允许与SourceID相关联的函数向系统存储器提供物理地址，则TA可能不翻译该物理地址。  地址.如果不允许该功能提供物理地址，则TA可以将其视为UR。 |
| 11b | 保留 | 如果TA接收到AT字段设置为11b的存储器请求TLP，则TA将发信号通知不支持的请求（UR）。 |

AT字段仅为内存请求定义。 该字段保留用于其他TLP。

10.2.2翻译请求

翻译请求的格式类似于内存读取的格式。 AT字段用于区分翻译请求和正常的内存读取。

翻译请求的请求头具有[图10-8](#bookmark398)和[图10-9中所示的格式。](#bookmark399)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

字节0→字节4→字节8→字节12→

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0  76543210 | | +1  7654321 0 | | | | | | +2  76543210 | | | | | +3  76543210 | | |
| Fmt  000 | 类型  00000 | T9 | TC | T8 | ATTR  R | LN  R | 日  R | TD | EP | Attr  XR | 在  01 | 长度  0000xx xx xx0 | | | |
| 请求者ID | | | | | | | | 标签[7：0] | | | | | 最后DW BE  1111 | 第一个DW BE  1111 | |
| 未翻译地址[63：32] | | | | | | | | | | | | | | | |
| 未翻译地址[31：12] | | | | | | | | | | | 保留 | | | | NW |

图10-8 64位转换请求报头

字节0→字节4→字节8→

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0  76543210 | | +1  76543210 | | | | | | +2  76543210 | | | | | +3  76543210 | | |
| Fmt  001 | 类型  00000 | T9 | TC | T8 | ATTR  R | LN  R | 日  R | TD | EP | Attr  XR | 在  01 | 长度  0000xx xx xx0 | | | |
| 请求者ID | | | | | | | | 标签[7：0] | | | | | 最后DW BE  1111 | 第一个DW BE  1111 | |
| 未翻译地址[31：12] | | | | | | | | | | | 保留 | | | | NW |

图10-9 32位翻译请求报头

转换请求与读取请求具有相同的完成超时间隔。

<10.2.2.1>属性字段

对于翻译请求，宽松排序（RO）位是适用的，并且允许被设置，其中它影响翻译请求。

其相关翻译完成的顺序Attr字段的其余部分A的缩写

翻译请求不得依赖于TA来保证翻译之间的任何特定 完成和任何其他请求或完成。翻译请求没有订购要求。TA可以相对于任何其他请求对转换请求重新排序。

执行说明

翻译请求订购

因为在转换请求和其他类型的请求之间不能假定排序，所以转换请求不进行有效的刷新/排序原语。



5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<10.2.2.2>长度字段

长度字段被设置为指示响应于该请求可以返回多少翻译。每个翻译的长度为8个字节，代表一个或多个STU（最小翻译单元）。长度字段的最大设置是RCB。翻译请求中的长度字段必须始终指示双数的DWORD。如果长度被设置为指示大于允许的值，或者如果长度字段的最低有效位为非零，则TA将 将请求视为格式错误的数据包。

如果长度字段的值大于2，则函数请求的转换范围大于单个STU。如果提供了额外的翻译，则假设是用于顺序增加的， 从请求的地址开始的相同大小的STU对齐区域。

<10.2.2.3>标签字段

标签字段与内存读取请求中的含义相同。

<10.2.2.4>未翻译地址字段

翻译请求包括32位或64位未翻译地址字段。 此字段指示要翻译的地址。 TA将根据翻译请求中的地址决定请求的有效性。TheTA被允许返回比要求更少的翻译，但不会返回更多。

当请求多个翻译时，如果翻译的范围不与翻译请求的隐含范围重叠，TA将不会返回翻译（这仅适用于初始值之后的翻译）。翻译请求的隐含范围是[2 STU+12 \*（Length/2）]字节。

翻译请求中的未翻译地址字段是第一个STU范围内的任何地址。地址位11：0 不存在于传输请求中，并且被暗示为零。如果一个打印机具有页面对齐请求集（请参见 第www.example.com节7.8.8.2），它必须确保位11：2为零。如果一个打印机已清除页面对齐请求，则允许它 为位11：2提供任何值。166Theta必须忽略位11：2以及任何不需要确定转换的低阶位。

例如，如果使用64位寻址用于具有被编程为STU为1的页对齐请求位集的函数（即，8192-字节页），位63：13是有效的，位12是由TA进行“或”运算，位11：0被暗示为零。

<10.2.2.5>无写（NW）标志

NoWrite标志在设置时，表示该函数正在为该转换请求只读访问。167

TA可以忽略No Write Flag，但是，如果TA响应的翻译标记为只读，

函数不得使用该转换发出内存写入事务。在这种情况下，该功能可以发出另一个转换请求，该转换请求具有无写标志清除，这可以导致新的转换完成，具有或不具有W（写）位设置。

在接收到具有NW标志Clear的转换请求时，TA被允许将相关联的页面标记为脏。强烈建议函数不要发出这样的请求，除非它们被给予明确的写权限。



166.注：页面对齐请求位是在ATS规范的修订版1.1中添加的。

167.注：在修订版1中添加了无写入标志1、ATS规范。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

<10.2.2.6>翻译请求上的PASID TLP前缀

如果转换请求具有PASIDTLP前缀，则未转换地址字段是由PASID字段指示的进程地址空间内的地址。

如果转换请求具有带已请求模式或已请求扩展位集的PASIDTLP前缀，则这些可以用于构造转换完成数据条目。

PASID扩展能力指示功能是否支持并被启用以发送和接收具有PASIDTLP前缀的TLP。

10.2.3翻译完成

翻译完成（Cpl或CpID）由TA为每个翻译请求发送。本说明书

介绍了“翻译完成”中字段的含义本规范中未定义的字段具有相同的

在本说明书中，阅读完成所禁止的含义。对于翻译完成，宽松的排序

如果对应的翻译请求RO位被置位，则RO位是适用的并且被发送为置位。Attr字段的其余部分为保留。

如果TA不能执行请求的翻译，则使用[图10-10所示的格式完成](#bookmark400)。

字节0→字节4→字节8→

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0  76543210 | | +1  76543210 | | | | | | +2  76543210 | | | | | | | | +3  76543210 | |
| Fmt  000 | 类型  01010 | T9 | TC | T8 | ATTR  R | LN  R | 日  R | TD | EP | Attr  XR | | | 在  RR | | 长度 | | |
| 完成者ID | | | | | | | | Cp湖状态s | | | | BCM  0 | 字节计数 | | | | |
| 请求者ID | | | | | | | |  | | | 标签 | | |  | | R | 低位地址 |

图10-10无数据的翻译完成

执行说明

没有数据的不成功的事务处理完成的字节计数字段

此规范的以前版本指示字节计数和较低地址字段应为00000000

0000b表示没有数据的不成功翻译完成。强烈建议实现不依赖于在Unsuccessful中将字节计数和较低地址字段设置为任何特定值

没有数据的翻译完成



完成状态字段的值和含义如[表10-2所示。](#bookmark401)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

表10-2无数据状态码的翻译完成

|  |  |  |
| --- | --- | --- |
| 值 | 地位 | 意义 |
| 000b | 成功 | 此完成状态具有“成功“的名义含义，Theta不会在CPL中返回此值。 |
| 001b | 不支持的请求（UR） | TA不支持该职能部门的翻译请求。如果一个函数接收到这个  完成代码，它必须禁用其ATC，并且在ATC重新启用之前不使用转换后的地址发送请求。对于职能部门内部可能正在进行的交易，职能部门可以终止或完成这些交易。接收此代码的函数用来报告此条件的机制是外部的 本规范的范围。检测到该错误的TA是“完成者发送具有UR/CA状态的完成”，并且应按照本规范中的定义进行操作。 |
| 010B | CRS | 此值不允许在任何由PCI Express功能发起的请求中使用。如果由职能部门接收，则应将其视为格式错误的TLP。 |
| 100b | 完成者中止（CA） | 由于TA中的一个错误，TA无法转换地址。这通常会导致向与ATC相关的设备驱动程序报告错误。 参见本规范中的AER。 |
| 所有  别人 | 保留 | 具有保留完成状态值的转换完成被视为完成状态为不支持的请求（001b）。 |

注意：如果返回值不是Success，则表示错误。

字节0→字节4→字节8→

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0  76543210 | | +1  7654321 0 | | | | | | +2  76543210 | | | | | | | | +3  76543210 | |
| Fmt  000 | 类型  01010 | T9 | TC | T8 | ATTR  R | LN  R | 日  R | TD | EP | Attr  X R | | | 在  RR | | 长度 | | |
| 完成者ID | | | | | | | | 下士地位  000 | | | | BCM  0 | 字节计数 | | | | |
| 请求者ID | | | | | | | |  | | | 标签 | | |  | | R | 低位地址 |

图10-11成功完成翻译

根据第2.2.9节和第2.3节设置字段。

翻译完成必须使用与翻译请求相同的TC发送。不要求该职能部门验证是否使用了相同的TC。

较低地址字段将包含将使分组与RCB语义一致的值。如果结果是

如果在单个数据包中返回，则低位地址设置为RCB减去字节计数。如果结果以多个数据包返回，则第一个数据包将具有RCB减去（总完成长度 \* 4）的较低地址字段，并且随后的数据包将 有一个较低的地址字段0000000b。 参见[第10.2.4f](#bookmark402)节或多个数据包完成的附加要求。

如果完成状态字段为000b，则转换成功，并且数据有效负载将跟随报头。数据有效载荷的内容如[图10-12所示。](#bookmark403)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0 | | | | | | | | +1 | | | | | | | | +2 | | | | | | | | +3 | | | | | | | |
| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|  | 翻译地址[63：32] | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |  |
| 翻译地址[31：12] | | | | | | | | | | | | | | | | | | | | S | N | 保留 | | | | 全球 | Priv | Exe | U | W | R |

A-0583A

图10-12翻译完成数据录入表10-3翻译完成数据字段

|  |  |
| --- | --- |
| 领域 | 意义 |
| S | **翻译大小**-如果翻译应用于4096字节的内存范围，则此字段为0 b。如果此字段为1b，则转换适用于大于4096字节的内存范围（请参见www.example.com[部分10.2.3.1）](#bookmark404)。 |
| N | **非监听访问**-如果此字段为1b，则使用此转换的读写请求必须清除属性字段中的无监听位。如果它是0b，则函数可以使用其他手段来确定是否应该设置无监听。 |
| 保留 | ATC应忽略这些位 |
| 全球 | **全局映射**-如果该位被置位，则允许ATC在所有P ASID中缓存该映射条目。如果清除，ATC为  允许仅在与请求PASID相关联的PASID中高速缓存该映射条目。 只有当相关的翻译请求具有PASIDTLP前缀时，此位才可置1。 |
| Exe | **允许执行**-如果此位被设置，则允许请求函数执行相关内存范围中包含的代码。  只有当相关的翻译请求具有一个PASIDTLP前缀，且执行请求位已置位时，此位才可置位。如果该位为置位，则R也必须为置位。  Priv位指示与Exe位关联的缓冲级别。如果设置了Priv，则Exe位表示权限 与函数中的已标记模式项相关联。如果Priv为Clear，则Exe位指示与函数中的非授权模式实体相关联的许可。  如果R是Set，则可以缓存该值。 |
| Priv | **加密模式访问**-如果此位为置位，则R、W和Exe指的是与加密模式实体关联的权限。如果此位为清除，则R、W和Exe表示与非特权模式实体相关的权限。  只有当相关的转换请求包含一个PASIDTLP前缀，且已设置模式请求位时，此位才可被设置。  此值必须缓存任何R、W或Exe值。 |
| U | **Untranslated access only（仅**未翻译访问）-当此字段为Set（设置）时，只能使用未翻译访问  地址，并且该转换完成数据条目的转换地址字段可能不会在AT设置为已转换的后续读/写请求中使用。如果R或W被设置，则可以缓存该值。 |



第1184页

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |
| --- | --- | --- |
| 领域 | 意义 | |
| R、W | **读、写**-这两个字段表示使用转换的请求所允许的事务类型。编码是： | |
| 00b  01b  10b  11b | 不允许读或写事务。 这个翻译被认为是无效的。的  转换地址、N、U和Exe字段的内容未定义。 具有此值的翻译不得缓存在ATC中（参见[第1节0.2.3.5）](#bookmark405)。  允许写入请求达到这个范围，但是不允许读取请求，除非它们是零长度读取。  允许针对此范围的读取请求（包括零长度读取），但不允许写入请求。  允许针对此范围的读取和写入请求 |
| Priv位表示与R和W相关的缓冲级别。如果设置了Priv，则R和W表示与功能中的已注册模式实体相关联的权限。如果Priv为Clear，则R和W表示与非  在函数中标记模式 | |
| <10.2.3.1>翻译地址字段  如果R和W字段均为清除，或者如果U为设置，则转换后的地址字段不能被函数用于任何目的。  如果R或W字段被设置，并且U字段被清除，则转换地址字段包含可以由功能在存储器请求中使用的地址，其中AT字段被设置为转换，并且功能可以高速缓存转换地址。缓存时，R和W字段必须与翻译完成条目存储相同的值。缓存的地址必须是转换完成中指示的地址范围的子集（该子集可以 包括整个范围）。  当转换后的地址被缓存在功能ATC中时，除了删除之外，功能不可能修改条目。当收到无效请求时，  与缓存地址的任何部分重叠的指定范围  不允许功能在其ATC中创建条目，除非该条目处于翻译完成中并且ATS能力中的E（启用）字段已设置。ATC高速缓存中在E字段置位之前写入的缓存不能用于内存请求。 当E字段为Set时，它们必须无效，或者忽略和不使用它们。  <10.2.3.2>翻译范围大小（S）字段  如果S为Set，则转换适用于大于4096字节的如果S = 1b，则T的位12被传送  地址用于指示范围是否大于8192字节。如果第12位为0b，则范围大小为8192 字节，但如果设置，则大于8192字节。如果S= 1，bit 12= 1b，则bit 13用于确定范围是否大于16384字节。如果第13位为0b，则范围大小为16384字节，但如果置位，则范围大小大于16384字节。  低阶地址位按顺序被消耗，以指示与转换相关联的范围的大小。注意：这个编码方法也被用来表示被无效的内存范围的大小。  [表10-4给出了不同翻译大小的例子。](#bookmark407) | | |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

表10-4使用S字段的转换大小示例

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 地址位 | | | | | | | | | | | | | | | | | | | | | S | 翻译  范围大小  以字节 |
| 63：32 \* | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 |
| X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | 0 | 4 K |
| X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | 0 | 1 | 8 K |
| X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | 0 | 1 | 1 | 16 K |
| X | X | X | X | X | X | X | X | X | X | X | X | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 M |
| X | X | X | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1G |
| X | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 4 G |

注意：

\* 高位地址位用于指示大于4 GB的范围的大小



大小字段被设置为以4096字节的倍数指示范围大小，而不管STU的设置如何。例如，如果STU被设置为指示最小转换是8192字节，则S应该在 翻译完成和所有无效请求。如果STU被设置为指示16384字节的最小值，则在所有转换和无效范围中S和位12都将被设置。

如果S被置位并且位63：12都是1b，则行为未定义。如果S被置位，并且位63为0b，并且位62：12都为1b，则请求使所有已验证的无效。

如果功能收到翻译完成，且翻译大小字段小于功能的编程STU值，则应将翻译完成视为具有完成状态UR。

<10.2.3.3>Non-snooped（N）字段

此字段被设置为指示在此转换范围内的目标存储器上的读取和写入请求必须清除请求报头中的无监听属性位。当该字段为0b时，允许该功能设置无监听操作

属性位。

注：当此字段被清除时，如果设备控制寄存器中的启用无监听字段被清除，则不允许功能在内存请求中设置无监听。

如果R或W被置位，则ATC可以缓存N位

当U为Set时，此字段的含义未定义，TA可以将此字段设置为任何值。转换具有不受特权级别影响的N字段的单个值。允许ATC缓存N字段，而不考虑 Priv位的值。

<10.2.3.4>UntranslatedAccess Only（U）字段

当不允许函数使用转换后的地址访问隐含的内存范围时，设置此字段（该范围由转换请求中的未转换地址和转换请求中的转换偏移量

翻译完成）。函数可以使用未转换的地址来访问范围，只要R和W字段允许访问。如果R或W被设置，则函数可以缓存该转换值。如果U字段被置位，则转换中的转换地址字段不一定是有效的存储器地址，并且功能可以不使用AT被置位为转换的读取或写入请求中的值。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

注：此字段的一个可能用途是避免不必要的无效。如果函数使用转换后的请求， 对于存储器的某些部分而不是其它部分，则U字段可以用在不使用转换请求的部分上。如果U字段为Set，则当转换发生变化时， 请求发送到函数。这种用法的一个例子是带有用于命令的环形缓冲区的函数。环形缓冲区可以被分配很长一段时间，并且具有非常高的重用（局部性）。出于这个原因，函数在其目标为命令缓冲区的存储器请求中使用转换后的地址是有用的。相同的

可能访问局部性差和重用率低的数据缓冲区。 对数据缓冲区的访问最好使用未转换的请求来处理。为数据缓冲区转换设置U字段可确保函数不会

尝试使用转换后的值来访问数据缓冲区，因此，当数据缓冲区映射发生更改时，

需要无效请求。当U被置位且R或W被置位时，ATC被允许缓存U、R、W Exe和Priv以及转换范围大小（请参见[Searchwww.example.com10.2.3.2）](#bookmark406)。如果这些值 变化

<10.2.3.5>读（R）和写（W）字段

这些字段指示返回的转换值是否可用于读或写存储器请求。ATC可能不会 如果R字段被清除，则使用转换值发出非零读取请求。ATC不得发出写请求 如果W字段被清除，则使用平移值。如果R字段和W字段都未被设置，则ATC可以不使用转换值发出任何类型的请求。如果R和W字段都被清除，则仍指示平移的范围，但 翻译中的其它值的含义是未定义的。

注：即使R = W = 0b，也会指示翻译条目的范围，以便在翻译中允许

建成例如，如果翻译请求具有六个DW的长度，则可以最多执行三个翻译

包含在翻译完成中。 第一和第三平移可以具有集合R或W，但是第二平移可以具有R=W= 0b。为了避免所指示的间隙的大小不明确，在翻译中指示间隙的范围。

即使R = W = 0b也是完整的

R= 0b，W= 0b状态用于指示转换中的地址字段不可用于形成用于后续请求的转换后的地址值。

当主机改变TA中的翻译时，为了使翻译存在，主机不需要发送

- 向ATC发出无效指示，使得其将知道翻译状态的改变。 由于ATC可能不会被通知翻译的变化，因此R=W=0b的翻译值必须不被缓存。

如果没有找到所请求地址的表项，TA将返回一个带有单个翻译值R=W=0b的CplD。

注：实现不应假设接收到具有R或W位设置（与Ubit的值无关）的转换响应意味着具有相同未转换地址的后续读取或写入请求将成功。尽管设备及其控制软件可以确保该属性，但是这样做的方法是： 在本说明书的范围之外。

Priv位用于限定R和W。如果设置了Priv，则R和W表示授予在 这个函数如果Priv为Clear，则R和W表示授予功能中的非授权模式实体的权限。两个特权级别的R和W值是独立的。ATC不得假设以下各项之间存在任何相关性：

与翻译关联的“隐藏模式”和“非隐藏模式”权限。

<10.2.3.6>允许执行（Exe）

如果Exe为Set，则允许请求函数在隐含的内存范围内执行代码。如果Exe为Clear，则不允许请求函数在隐含的内存范围内执行代码。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

函数执行代码的含义的定义超出了本规范的范围。 各种系统组件可以具有不同的指令集。当请求函数试图执行此位不允许的代码时，其内部的行为超出了本规范的范围。

如果TA支持执行权限，则关联的转换请求具有PASIDTLP，则只能执行位 执行请求位168的有效值为1b的前缀，并且在翻译完成数据条目中设置R。

否则，Exe位必须为Clear。 如果R为Set，则可以缓存此值。

Priv位用于限定Exe位。如果设置了Priv，则Exe位表示授予了对已配置模式的

功能中的实体。如果Priv为Clear，则Exe位指示授予该功能中的非授权模式实体的权限。两个特权级别的Exe位值是独立的。ATC不得假设任何相关性 在与转换关联的特权模式和非特权模式权限之间进行切换。

函数可以选择检查：

· 如果翻译请求中的执行请求位为清除，则相关翻译完成数据中的执行位也为清除。

· 如果Exe为Set，则R也为Set。

如果任一可选检查失败，则功能应发出意外完成（UC）信号。 这些检查是独立可选的。

<10.2.3.7>注册模式访问（Priv）

如果设置了Priv，则R、W和Exe指的是在所需的

功能如果Priv为Clear，则R、W和Exe是指授予请求功能中以非授权模式操作的实体的权限。

“非授权模式”和“非授权模式”的含义，以及实体在

保护模式或非保护模式取决于系统的保护模式，不在本规范的范围内。

当请求函数中的实体试图访问不允许访问的内存时，行为超出了本规范的范围。

只有当TA支持优先模式，并且相关的转换请求具有有效值为1b的PASIDTLP前缀（用于优先模式请求169位）时，才可以设置优先位。否则，Priv位必须为Clear。

R、W和Exe的扩展和非扩展模式版本是独立的。ATC可以缓存其中一个或两个

R、W和EXE的版本。 对于一个特权等级接收具有R=W=0b的转换的ATC可以不假设关于它对于另一特权等级可能接收的内容的任何事情。

如果R或W是Set，则可以缓存该值。 当缓存相应的R、W或Exe值时，必须缓存此值。

注意：由于只有当请求函数设置了缓存模式请求位时，才设置了Priv位，因此从不设置该位的函数应该始终接收清除的Priv位，因此不需要缓存它。

功能可以可选地检查当转换请求中的优先模式请求位为清除时，相关联的转换完成数据条目中的优先位也为清除。如果此可选检查失败，则功能应发出信号 意外完成（UC）。



168. 除非PASIDTLP前缀中的位为1，否则该位的有效值为0b。

169. 除非前缀中的位为1，否则位的有效值为0b。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

执行说明

执行权限和授权模式实施

请求函数决定一个特定的内存请求是否需要执行权限， 与已注册模式或非已注册模式实体相关联。 ATC执行Exe和Priv位指示的保护检查。



<10.2.3.8>全球地图（全球）

如果全局被设置，则允许请求功能在ATC中为该转换创建全局映射条目。如果 Global为Clear，则不允许请求函数在ATC中为该转换创建全局映射条目。全局映射条目适用于函数的所有PASID。他们允许ATC减少翻译的数量

并减少缓存结果所需的存储器。

允许函数忽略此位，并始终在ATC中创建非全局映射条目。 这可能导致在不同PASID下为同一未翻译地址请求多个翻译。

使用此位的函数还必须设置全局无效支持位（请参见10.5.1.2部分）。

10.2.4多个翻译的

允许ATC要求TA提供几乎连续范围的地址的翻译。它通过将翻译请求中的长度字段设置为请求翻译数量的两倍来实现这一点， 只要请求大小（Total Completion Length \*4）不大于Max\_Read\_Request\_Size或RCB。

如申请多个翻译，只要翻译数量不超过申请翻译数量，电讯局长可退回一个或多个翻译。对于TA来说，返回的翻译少于

并且不发送错误指示，除非在访问数据时存在错误。

如果翻译完成包含多个翻译，则所有翻译必须具有相同的指定大小。此外，连续的转换必须应用于在相同的时间内邻接先前转换的虚拟地址范围。

建成

如果一个转换同时具有R= 0和W= 0b，则TA仍必须将用于编码完成大小的大小字段和转换地址字段的低位设置为适当的值。

翻译完成中的每个翻译都将与翻译请求的已实现内存范围有一些重叠（参见[第10.2.2节）](#bookmark397)。

一个成功的翻译组件必须由一个或两个CPLD组成。

如果转换完成CplD具有大于长度字段的四倍的字节计数，则需要额外的CplD来完成事务。

如果翻译完成CplD的字节计数等于长度字段的四倍，则数据包完成请求。对于这样的CplD，如果字节计数和低位地址的和不是RCB的倍数，则CplD是RCB的整数倍。

最后一个序列。如果没有收到此请求的先前CplD，则发生错误，并且应丢弃所有转换值。

注意：TA可能会截断完成结果的原因有多种。例如，请求可能会要求一个地址范围，而不是所有的地址都被定义。如果第一个转换有效但位于

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

一页翻译的结尾。TA在查找下一页翻译时，可能会发现该页无效，因此地址无效。将返回有效的地址范围，并且没有错误指示。当

截断翻译完成不允许TA用无效条目填充响应（R= 0b，W = 0b）。

注意：有多种原因导致TA可能将一个翻译完成分成多个TP. 例如，如果

转换完成的虚拟地址解析为跨越

在存储器系统中，对TA的完成可以由存储器分解成多个完成。当从系统存储器接收到传输完成的每个部分时，允许将其发送到函数，而不是要求TA累积结果。

10.3 ATS失效

ATS使用本节所示的消息来保持TA和ATC之间的一致性本规范

假设每个ATC都有一个TA。TA（与其相关软件一起）必须通过发出无效请求来确保ATC中缓存的地址转换不会过期。

10.3.1使请求无效

当TA中的翻译发生变化，并且该翻译可能包含在功能中的ATC中时，TA（在 ATPT（与其相关联的软件结合）必须向ATC发送无效请求，以保持ATPT和ATC之间的适当同步。 无效请求用于从ATC清除地址范围的特定子集。无效请求被限制为覆盖4096字节页的2次方。

无效请求的格式如[图10-13所示。](#bookmark408)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 6 | | 5 | +043 | |  |  | | --- | --- | | 2 | 1 | | 0 | 7 6 | | 5 | +143 | | 2 1 | | 0 | 7 6 5 | | | +243 | | 2 | 1 | 0 | 7 | |  |  | | --- | --- | | 6 | 5 | | +343 | |  |  | | --- | --- | | 2 | 1 | | 0 |
| 0 | Fmt1 1 | | 类型 10010 | | | R | TC | | | R | 阿特河 | R | | TD | EP | AttrRR | | R | | 长度  00 0000 0010 | | | | | | |
|  | 请求者ID | | | | | | | | | | | | | 0 | 0 | 0 | ITAG | | | | | 信息代码00000001 | | | | |
| 设备ID | | | | | | | | | | | | | | 保留 | | | | | | | | | | | | |
| 保留 | | | | | | | | | | | | | | | | | | | | | | | | | | |

A-0584A

图10-13无效请求消息

Invalidate Request是一个MsgD事务，包含64位数据。无效请求消息可以在任何TC中发送。 ITag字段被构造为0到31的值，TA使用它来唯一地标识它发出的请求。ATA必须确保一旦使用了一个ITag，在相应的无效完成或供应商特定的超时机制（见下文）释放之前，它不会被重新使用。

TA可以具有用于其发出的所有无效的ITag值的单个池，或者其可以具有用于每个设备ID或任何其他组合的池。在不同功能上具有多个ATC的设备必须为每个ATC ID单独管理ITags。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

无效请求中指定的地址范围可以跨越一个或多个STU4096字节页面。无效范围需要自然对齐，并且不得小于STU4096字节页。收到无效通知后 对于范围小于STU的请求，ATC可以（1）用信号通知不支持的请求，或者（2）将请求的范围舍入到大于或等于STU的值。

执行说明

使完成无效

设备应在1分钟（+50%-0%）内响应无效请求。有限制的时间允许ATPT实现无效完成超时并重用关联的ITag值。ATPT设计是

具体实施。因此，使完成超时无效及其相关的错误处理不在本规范的范围内。



有效负载的内容是要无效的未转换地址范围。有效负载格式如RemoveWord{RD}标签所示，用于帮助生成ToC/ToF/ToT。

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 6 | | 5 | +04I3 | |  |  | | --- | --- | | 2 | 1 | | 0 | 7 | |  |  | | --- | --- | | 6 | 5 | | +14I3 | |  |  | | --- | --- | | 2 | 1 | | 0 | 7 | |  |  | | --- | --- | | 6 | 5 | | +243 | | | 0 | 7 | |  |  | | --- | --- | | 6 | 5 | | +343 | 2 | 1 0 | |
|  | 未翻译地址[63：32] | | | | | | | | | | | | | | | | | | | | |  |
| 未翻译地址[31：12] | | | | | | | | | | | | | | S | 保留 | | | | | | |  |

|  |  |
| --- | --- |
| 2 | 1 |

▲



全局无效

图10-14使请求消息体

S字段用于指示被无效的范围是否大于4096字节。其含义与翻译完成相同（参见[10.2.3.1](#bookmark404)和[10.2.3.2）](#bookmark406)。

全局无效位指示无效请求消息影响所有PASID值（参见[第10.3.8节）](#bookmark409)。 除非无效请求具有PASIDTLP前缀，否则保留该位。如果全局，ATC将忽略该位

无效支持位为清除（参见[第10.3.8节）](#bookmark410)。

10.3.2完成无效

当一个函数完成一个无效操作时，它将向TA发送一个或多个无效完成消息。这些消息必须使用从无效请求中提取的信息进行标记，以使TA能够关联 无效完成与无效请求。

失效日期完成消息的格式如[图10-13所示。](#bookmark408)

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 7 6 | | 5 | +043 | |  |  | | --- | --- | | 2 | 1 | | 0 | 7 6 | | 5 | +1  4 3 2 1 0 | | | | | 7 6 5 | | | +243 | | 2 1 0 | | | +3  7 6 5 4 3 2 1 0 | | | | | | | |
| 0 | Fmt0 1 | | 类型  1 0 0 1 0 | | | 0 | TC | | | 0 | 阿特河 | 0 | 0 | TD | EP | AttrRR | | R | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 请求者ID | | | | | | | | | | | | | | 保留 | | | | | | | | 信息代码00000010 | | | | | | | |
| 设备ID | | | | | | | | | | | | | | 保留 | | | | | | | | | | | | | CC | | |
| ITag向量 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

A-0586A

图10-15作废完成信息格式

Invalidate Completion消息是由ID路由的Msg事务。Invalidate的验证者ID

完成消息设置为包含ATC的功能的验证器ID。无效完成的设备ID字段被设置为TA的请求器ID。ATC可以从授权器ID导出TA的授权器ID 对应的Invalidate请求的字段。或者，由于ATC仅与单个TA相关联，因此ATC 可以采样并存储来自基本复位或FLR之后的第一无效请求的验证者ID。

随后的无效完成消息可以使用该值来设置无效完成消息的设备ID字段。

完成计数（CC）字段指示必须为关联的无效请求发送的单个无效完成消息的数量。 将CC字段设置为0表示必须发送八个响应。θ是

负责收集与给定标签相关联的所有响应，然后再考虑相应的无效请求是否完成。

无效完成消息可以在任何TC上发送，与接收到发起无效请求的TC无关。 这使得实现方式能够利用无效完成来将未完成的事务推送到TA，以保证满足所需的无效语义。利用单个上游TC的实现包括

需要在所用TC中发送单个无效完成。

ITag向量字段用于指示已完成的无效请求。来自无效请求的32个可能的ITag字段值中的每一个都由ITag向量字段中的单个位表示。最低有效位（位0;即，在图10-13所示的无效完成消息的示意性表示中的最右边的位）的ITag向量字段对应于ITag字段值0。ITag向量字段的最高有效位（位

对应于ITag字段值31。如果满足以下条件，则允许实现通过在单个消息中设置多个ITag向量位来合并多个无效完成：

·无效完成流在同一TC中。

·无效完成具有相同的CC值。

·输入验证完成的所有片段必须具有相同的请求ID、CC和ITag向量字段。

收到没有未完成失效请求的ITag的失效完成的ATA应报告此情况

使用特定于实现的机制时出错。一种可能的这种机制是将无效完成报告为意外完成（UC）。

不支持ATS的函数将把Invalidate Request视为UR。

支持ATS的功能需要发送一个Invalidate Completion来响应Invalidate Request

与总线主机使能位是否置位无关。并不是说，即使总线主机使能被清除，也必须满足上述条件。用于设备实现这一点的方法是依赖于实现的。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

执行说明

总线主机启用更改

当总线主机使能从置位变为清除时，不应再排队等待更多的内存请求。可以

当BME被清除时，存在排队的写请求。这些请求可以阻止无效完成。这些请求必须被发送或丢弃。 这将确保潜在地依赖于未完成的无效的所有未完成的写入事务完成。



10.3.3使竞争语义无效

在ATC可以为给定的无效请求返回无效完成之前，它必须确保满足以下条件：

·由功能发起的所有新请求将不利用陈旧地址转换。

·利用与无效范围匹配的转换地址的所有未完成的读取请求已经完成或被标记为被丢弃（丢弃的方法是特定于实现的）。

·利用与无效范围匹配的转换地址的所有未完成的提交写入已被推送到TA。要求ATC在每个TC中发送一份无效完成消息的副本，

已发布写操作，但不知道是否已推送到TA。 CC字段必须在Invalidate Completion消息的每个副本中设置为相同的值，以指示发送的副本数量。 TA负责收集所有已发送的响应，然后再将失效视为完成。

执行说明

暗示TC冲洗

当做出关于哪个TC发送无效完成的决定时，ATC可以以实现特定的方式推断所发布的公告写入已经被推送到TA。例如，一个函数发送了一个 读事务到TA以上的目的地并接收到其相应的响应可以推断任何

在同一TC中发布的先前已发布的写入已推送到TA。



10.3.4请求接受规则

根据本节中列举的请求接受规则，不允许函数创建

一种依赖关系，在这种依赖关系中，对一个投递的传输的接受取决于对一个投递的传输的接受。

交易日鉴于无效请求和无效完成都是重新发布的交易，职能部门不得根据无效完成的传输来接受无效请求。的 实现这一点的方法是特定于实现的。

在其配置空间中具有ATS功能的功能必须能够接受无效请求并发送无效完成，即使ATS未启用。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

执行说明

队列深度无效

ATC仅与单个TA相关联。每个TA被限制为总共32个未完成的无效到任何给定的

空管这将对单个ATC有效的未完成失效请求的数量限制为32。为避免职位互相依赖，ATC最多可接受32个调查验证请求。

ATC可以选择实现最大大小的输入队列来保存无效请求。或者，ATC可以选择实现最大大小的输出队列来保存无效完成。注意

排队无效完成需要的每个条目的状态明显更少，从而导致比输入队列缓冲潜在更有效的实现。

请注意，选择是实现输入队列还是输出队列（或两者的混合）对确保无死锁行为没有影响。但是，关于排队的实现选择可能会对性能产生重大影响（参见[第10.3.5节）](#bookmark412)。



10.3.5使流量控制无效

由于缓存体系结构和排队策略的多样性，

无效延迟和吞吐量。TA可能以超过平均ATC服务速率的速率生成无效请求。 当这种情况发生时，基于信用的流控制机制将节流TA发行速率。这样做的副作用是通过基于信用的流控制机制将拥塞扩散到其他信道和链路。 根据这种拥塞的频率和持续时间，性能可能会受到影响。强烈建议TA及其相关软件实现更高级别的流量控制机制。

为了帮助实施无效流控制，ATC必须在对链路进行反压之前公布它可以缓冲的无效请求的数量。 此字段适用于该函数服务的所有无效，与无效的大小无关。此值在ATS功能的Invalidate Queue Depth

结构（见第7.8.8节）。 值为0 0000 b表示此函数不需要无效流控制。

执行说明

使流量控制无效

函数可以指示当以下一个或多个为真时不需要无效流

1. 该功能可以以无效请求的最大到达率处理无效。

2. 该函数不会或很少会导致链路回压（性能损失可以忽略不计）。

3. 该功能可以完全缓冲最大数量的传入无效，而不会对链路产生反压。



10.3.6无效排序语义

无效请求和翻译完成可以使用不同的TC来发送，并且因此相对于彼此是无序的（从链路的角度来看）。ATC必须确保正确的失效行为

当无效请求绕过转换完成到达重叠区域时，将保持无效。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

ATC必须“窥探”其未完成的转换请求队列，以防止所有到达的无效请求。当监听N\*STU大小的转换请求时（N是2的幂），ATC必须监听从STU开始的地址范围 包含指定地址的对齐区域，并在后面结束（N-1）个STU大小的页面。

如果无效请求与未完成的翻译请求中的地址范围重叠，则翻译请求必须被标记为无效，并且其对应的翻译响应的结果必须在传输 无效的完成。如果在发送无效完成之前收到转换响应，

如果满足无效完成语义（参见[第10.3.3节）](#bookmark411)，则实现可以自由地使用转换结果发出请求。

执行说明

无效中的请求范围重叠

在上面的描述中，N是在翻译请求中请求的STU大小的翻译的数量。这等于（翻译请求中的长度字段）/2。

例如：

STU为00 0010b，表示16384字节的页面。

未完成的翻译请求具有长度字段0000000100b，指示覆盖32768字节范围的两个翻译。

翻译请求的高阶48位是00000FFF FFFFh。

请求中地址的低位16位是11xxxxxxxxxxxxb，表示转换请求覆盖的范围与327个68字节的边界重叠（实际上，请求跨越了16 TB的边界）。

如果返回两个转换，则它们将覆盖00000 FFFFFFF C000和0000100000000000h处的两个STU大小的区域。

接收到的无效请求具有高位48位0000 1000 0000和低位16位00011xxxxxxxxxxx b。

ATC必须检测到与翻译请求的一部分相关的翻译现已无效 并且必须丢弃与无效区域相关联的翻译完成（为了简化，允许ATC丢弃所有翻译完成）。

应当注意，如果转换请求不跨越请求的对齐边界，则无效请求的处理被简化。上述示例中的翻译请求未与

32768字节边界。如果它被分成两个请求，那么将

使用翻译请求中的地址使请求无效。不要求将翻译请求拆分为对齐的请求。



10.3.7隐式失效事件

以下事件将导致所有ATC条目无效

·常规复位（所有形式）

· 功能级重置

· ATS能力中的E字段从清除更改为设置

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

以下事件将导致使用特定PASID请求的所有非全局映射ATC条目无效

·停止使用本规范中定义的PASID

当这些暗示的无效事件发生时，不会发送显式的Invalidate Completion消息。

执行说明

隐式失效和PASID

当ATS功能中的E字段被设置时，软件可能不会更改任何PASID启用。的

当软件设置E字段时发生的无效也会使具有相关PASID值的ATC条目无效。



10.3.8 PASIDTLP前缀和全局无效

本节中的要求适用于支持PASIDTLP前缀的函数。对于带有PASIDTLP前缀的无效请求，ATC应：

·如果关联的d PASID值大于或等于2 Max PASIDWidth，则可选地用信号通知不支持请求（UR）。只要出现超出范围的PASID值，即使忽略PASID值（见下文），也可能发出此错误信号。

· 如果PASID启用为清除，则返回无效完成。

· 如果函数支持全局无效（参见7.8.8.2部分）：

. 如果请求中的全局无效位为Set，则在与任何PASID值相关联的指定内存范围内使ATC中的全局和非全局映射条目无效，并返回

失效完成。PASIDTLP前缀中的PASID值将被忽略。

. 如果请求中的全局无效位为清除，则仅使ATC中使用相关PASID值请求的指定内存范围内的非全局映射条目无效，并返回无效完成。

可以保留ATC中用于一些或所有所指示的存储器范围的全局映射条目。

· 如果函数不支持全局无效（参见7.8.8.2），则在指定的内存范围内，使用相关PASID值请求ATC中的条目无效，并返回无效

建成

· 如果ATC中没有匹配的条目，则不使ATC条目无效，并返回无效完成。对于没有PASIDTLP前缀的无效请求，ATC应：

· 使指示内存范围内的ATC条目无效，这些条目在没有PA SID值的情况下被请求。

· 在所有地址上使用任何PASID值请求的ATC条目无效。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

10.4页面请求服务

页面请求的一般模型如下：

1. 功能确定它需要访问ATS转换不可用的页面。

2. 该功能使相关的寻呼请求接口向其RC发送寻呼请求消息。 页面请求消息包含页面地址和页面请求组（PRG）索引。PRG指数用于 标识事务并用于将请求与响应进行匹配。

3. 当RC确定其对请求的响应时（这通常将使所请求的页面驻留），它将PRG响应消息发送回请求功能。

4. 然后，该功能可以采用ATS来请求对所请求的页面的翻译

寻呼请求消息是以消息代码4路由到根联合体

（00000100 b）。RC用于缓冲请求的机制是特定于实现的。唯一的要求是RC不能默默地丢弃请求。

所有的寻呼请求消息和PRG响应消息都以PCIe业务类0传输。具有非0业务类的寻呼请求消息或PRG响应消息将被RC或端点视为格式错误的TLP，

收到同样的中间路由元件（例如，开关）不应检测到此错误。

可以使用寻呼请求消息和PRG响应消息的Attr字段中的宽松排序和基于ID的排序比特。保留Attr字段中的NoSnoop位。

寻呼请求服务允许将寻呼请求分组为寻呼请求组（PRG）。一个PRG可以包含一个或多个页面请求。PRG中的所有页面都由主机集体响应。PRG中的各个页面是

通过共享相同的PRG索引，被识别为属于公共PRG。PRG的最后一个请求在其寻呼请求消息中被标记为这样。一个请求信贷是

每个页面请求所消耗的资源（而不是每个PRG）。

PRG响应消息是通过ID路由回到请求功能的PCIe消息请求。它被使用

系统软件，以警告功能与相应PRG相关联的页面请求已经被发送。

满意页面请求机制不保证任何请求完成顺序，并且所有请求本质上独立于所有其他并发未完成的请求。如果一个函数需要一个特定的请求，

如果在另一个请求之前完成，则需要在发出后续请求之前完成初始请求。函数在不确定其驻留状态的情况下特定地请求页面和/或发出多个

对同一页的同时未处理的请求

页面请求接口被分配特定数量的页面请求消息信用。RC（系统软件）可以

以任何认为适当的方式分配可用信用额。主机选择采用的任何措施，以确保页面请求接口正确计量信用（页面请求接口不会使用超过其分配的信用） 是一种实现选择。页面请求接口不允许超额订阅可用的请求 （如果在根处超过缓冲区限制，那么这样做可导致禁用页请求机制）。的网页

请求接口的页面请求分配是静态的。它决定了页面请求接口何时被启用，并且只能通过禁用然后重新启用接口来更改。

10.4.1页面请求消息

函数使用页面请求消息向其关联的主机发送页面请求。页面请求指示函数所需的页面。与函数相关联的页面请求接口被赋予特定的页面请求

配置中页面请求接口不应发出超出其页面请求分配的页面请求。

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

页面请求包含所需页面的未转换地址、所需的

页面，以及PRG索引。PRG索引是一个9位标量，由函数分配，用于标识相关页面

请求过程中可以使用单个PRG索引来请求多个页面。当要关联多个页面时

对于给定的PRG，除了与给定PRG相关联的最后一个请求（在最后一个请求中设置该标志）之外，在所有请求中清除页面请求记录中的最后一个标志。页面请求被大量响应。不可能有回应

（响应失败错误除外），直到根收到PRG的最后一个请求。功能在任何给定时间可以具有未完成的PRG的数量小于或等于相关联的页面请求接口的未完成页面请求分配。一个请求组可以包含对同一页的多个请求，并且多个未完成的PRG可以请求同一页。

寻呼请求消息的前两个DW包含标准PCIe消息报头。消息的后两个DW包含页面请求特定的数据字段。

00h

04h

08h

0通道

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| +0 | | | | | | +1 | | | | | | | +2 | | | | | | | +3 | | | | | |
| 7 6 5 4 | | | | |  |  |  | | --- | --- | --- | | 3 | 2 | 1 | | 0 | 7 6 | | |  |  | | --- | --- | | 5 | 4 3 2 1 | | | | | 0 | 7 6 5 | | | |  |  | | --- | --- | | 4 3 | 2 1 | | | | 0 | 7 | |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | 6 | 5 | 4 | 3 | 2 | 1 | | | | | 0 |
| 0 | 0 | 1 | 类型  （10000） | | | 的t9 | TC | | T8 | Attr | R | | TD | EP | Attr | | R | 长度（0） | | | | | | | |
| 请求者ID | | | | | | | | | | | | | 标签 | | | | | | | 消息代码（0000 0100b） | | | | | |
| [63：32]第63话：你是谁？ | | | | | | | | | | | | | | | | | | | | | | | | | |
| 页面地址[31：12] | | | | | | | | | | | | | | | | | 页面请求组索引 | | | | | L | W | R | |

A-0737A

图10-16页面请求消息

表10-5寻呼请求消息数据字段

|  |  |
| --- | --- |
| 领域 | 意义 |
| R | **请求读取访问**-此字段在设置时表示请求功能请求对相关页面的读取访问。当清除时，此字段指示请求功能将不读取相关联的页面。  对于具有PASIDTLP前缀且已设置执行请求位的页面请求，必须设置R字段。如果R和W均为Clear（清除）且L为Set（设置），则这是一个停止标记（参见[第10.4.1.2.1节）](#bookmark413)。 |
| W | **请求写入**访问- 此字段在设置时指示请求功能寻求写入访问和/或  零长度读访问的assistiated页。清除时，此字段表示请求函数将不会写入相关页。  在接收到具有W字段Set的寻呼请求消息时，允许主机将相关联的寻呼标记为脏。因此，除非函数被给予明确的写权限，否则函数不能发出这样的请求。  如果R和W均为Clear（清除）且L为Set（设置），则这是一个停止标记（参见[第10.4.1.2.1节）](#bookmark414)。 |
| L | **PRG中的最后请求**-当设置时，该字段指示相关联的页面请求是PRG中的最后请求。  相关PRG 一个PRG可以有一个单一的条目，在这种情况下，PRG由一个单一的请求组成，其中该字段被设置。清除时，此字段表示将使用此记录的PRG索引提交其他页面请求。 |

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

|  |  |
| --- | --- |
| 领域 | 意义 |
|  | 如果R和W均为Clear（清除）且L为Set（设置），则这是一个停止标记（参见[第10.4.1.2.1节）](#bookmark415)。 |
| 页面  请求  组  指数 | **页面请求组索引**- 此字段包含函数为相关页面请求提供的标识符一  函数不需要使用PRG索引值的整个可用范围。 主机将永远不会用先前未由功能发出并且当前不是未完成的请求PRG索引的PRG索引来响应（除了发出响应失败时，在这种情况下，主机不需要在错误响应中保留相关联的请求的PRG索引值）。 |
| 页面 地址 | **页面地址**-此字段包含要加载的页面的未翻译地址。对于大于4096字节的页，忽略此字段的最低有效位。例如，当使用 正在请求8096字节的页面。 |

执行说明

最后一位和松弛排序

如果多个页面请求与单个PRG索引相关联，则PRG的最后一个页面请求除了具有Last\_flagSet之外还应当具有Relaxed\_Ordering属性位Clear。所有其他页面请求消息可以将Relaxed Ordering属性位设置为任何值。



<10.4.1.1>PASID TLP前缀用法

PASID扩展功能指示函数是否支持PASIDTLP前缀以及是否启用发送和接收它们。

允许支持PASIDTLP前缀的函数在页面请求消息中发送PASID TLP前缀。 PASID字段包含所请求页面的进程地址空间，请求执行和请求重命名模式位指示所请求的访问。

如果PRG中的一个寻呼请求消息具有PASIDTLP前缀，则该PRG中的所有寻呼请求消息必须包含相同的PASIDTLP前缀。当PASIDTLP前缀不一致时，行为未定义。

支持PASIDTLP前缀并设置了PRG响应PASID所需位的函数（参见10.5.2.3节）， 如果相关联的寻呼请求消息具有PASID TLP前缀，则预期PRG响应消息将包含PASIDTLP前缀。对于这样的PRG响应消息，保留请求执行和请求刷新模式比特 并且PASID字段包含来自相关联的寻呼请求消息的PASID。

<10.4.1.2>管理PRG请求上的PASID TLP前缀使用

停止ping和开始使用PASID是有规则的。

本节介绍了适用于已在正在停止的PASID中发出页面请求消息的函数的附加规则。不需要额外的规则来开始使用PASID的页面请求接口。

当停止使用特定PASID时，可以选择使用停止标记消息，以避免在功能指示特定PASID的停止请求已经完成之前等待PRG响应消息。

若要在不使用停止标记消息的情况下停止，该功能应：

5.0-1.0-PUB - PCI Express®基本规范修订版5.0版本1.0

1. 停止为此PASID发送新的页面请求消息。

2. 完成传输此PASID的任何多页页面请求消息（即 发送具有L位设置的寻呼请求消息）。

3. 等待与PASID的任何未完成页面请求消息相关的PRG响应消息。

4. 确认PASID已停止使用设备特定机制。此机制必须指示不会生成停止标记消息。

若要使用停止标记信息停止，功能应：

1. 停止为此PASID发送新的页面请求消息。

2. 完成传输此PASID的任何多页页面请求消息（即 发送具有L位设置的寻呼请求消息）。

3. 在内部将此PASID的所有未完成的页面请求消息标记为陈旧。与这些请求相关联的PRG响应消息将返回页面请求分配信用和PRG索引值，但否则将被忽略。170

4. 确认PASID已停止使用设备特定机制。此机制必须指示将生成停止标记消息。

5. 发送停止标记消息以向主机指示该PASID的所有后续页面请求消息都是针对PASID值的新用途。

注：步骤4和5可以按顺序或并行执行。

10.4.1.2.1停止标记消息

停止标记消息指示函数已停止使用页面请求接口，并已传输所有

特定PASID的挂起页面请求消息停止标记消息相对于页面是强排序的

请求消息，并用于向主机推送页面请求消息。当主机接收到停止标记消息时，这表明与被停止的PASID相关联的所有页面请求消息已经被递送

并且具有相同PASID值的任何后续寻呼请求消息与该PASID值的新的具体化相关联。

停止标记消息没有响应。它们没有PRG索引，也不消耗页面请求分配（参见www.example.com部分10.5.2.5）。

停止标记消息位布局如图10-17所示。



170.页面请求分配在函数的所有PASID之间共享（参见www.example.com部分10.5.2.5）。如果PRG Response PASID Required（需要PRG响应PASID）为Clear（清除），则PRG Index（PRG索引）值将在Functio n（函数）的所有PASID之间共享（参见第www.example.com节10.5.2.3）。

页面1200