# PCI Express原生热插拔

PCI Express架构被设计为本机支持电缆、外接卡和模块的热添加和热移除(“热插拔”)。PCI Express本地热插拔提供了一个机制“工具箱”，允许使用自一致的基础设施支持不同的用户/操作人员模型。这些机制可以用于实现有序的添加/删除，这依赖于与操作系统的协调(例如，传统的PCI热插拔)，以及异步删除，它不需要与操作系统进行锁步同步。本节定义热插拔机制集，并指定热插拔元素(如指示灯和按钮)在系统中实现时必须如何表现

## 6.7.1热插拔原理

支持热插拔型号的物理要素如表6-7所示。外形尺寸规格必须定义这些元素在该外形尺寸中的使用方式。对于给定的外形规格，可能只需要一些可用的热插拔元素，或者甚至不需要这些元素。在所有情况下，外形规格必须通过所包含的元素的选择来定义对系统或用户施加的所有假设和限制。仅用于选定外形尺寸的硅组件实现被允许仅支持相关外形尺寸所需的那些元素。

|  |  |
| --- | --- |
| 要素 | 功能 |
| 指示灯 | 显示槽位的电源和需要关注的状态 |
| 手动操作闩锁(MRL) | 固定适配器 |
| MRL传感器 | 允许端口和系统软件检测正在打开的MRL |
| 机电联锁 | 防止从插槽中移除适配器 |
| 注意按钮 | 允许用户请求热插拔操作 |
| 软件用户界面 | 允许用户请求热插拔操作 |
| 槽位编号 | 提供槽位的视觉识别 |
| 电源控制器 | 一种或多种软件控制的电子元件，用于控制插槽或适配器的电源，并在故障情况下监控电源 |
| 带外存在检测 | 确定插槽中适配器的物理存在的方法，不依赖于物理层 |

### 6.7.1.1指示灯

定义了两个指示灯:电源指示灯和注意指示灯。每个指示灯处于三种状态中的一种:亮、灭或闪烁。热插拔系统软件通过写入与指示器相关的命令寄存器来独占控制指示器状态(下面有一个例外)。指示灯要求必须包含在所有外形规格中。对于给定的形式因素，指示器可能是必需的，也可能是可选的，或者根本不适用

#### 6.7.1.1.1注意指示灯

注意指示灯为黄色或琥珀色，表示操作出现问题或热插拔槽正在被识别，便于操作人员定位

|  |  |
| --- | --- |
| 指示灯状态 | 含义 |
| 灭 | 正常-正常操作。 |
| 亮 | 注意-此槽有操作问题 |
| 闪烁 | 定位-槽正在根据用户的请求被识别 |

**注意指示灯灭**

注意指示灯处于“灭”状态，表示适配器(如果有)和热插拔槽都不需要注意。

**注意指示灯常亮**

注意指示灯常亮，表示适配器或槽位存在操作问题。

操作问题是指阻止适配器继续操作的情况。操作系统或其他系统软件确定是否有特定条件阻止适配器继续操作，以及点亮注意指示是否合适。操作问题的示例包括与外部布线、适配器、软件驱动程序和电源故障相关的问题。一般情况下，“注意指示灯”处于“开”状态，表示尝试操作失败或发生了意外事件。

注意指示器不用于报告在验证热插拔操作请求时检测到的问题。验证是一个术语，用于系统软件执行的任何检查，以确保所请求的操作是可行的、允许的，并且不会导致问题。验证失败的例子包括拒绝执行热插拔操作的权限、功率预算不足以及在接受热插拔请求之前可能检测到的其他条件。

**注意指示灯闪烁**

“注意指示灯”闪烁表示系统软件正在识别该槽位，供人工操作人员查找。

此行为由用户控制(例如，从软件用户界面或管理工具)。

#### 6.7.1.1.2电源指示灯

电源指示灯指示槽位的上电状态，必须为绿色。电源指示灯的状态如表6-9所示。

|  |  |
| --- | --- |
| 指示灯状态 | 含义 |
| 灭 | 断电-允许插入或拆卸适配器。 |
| 亮 | 上电-不允许插入或拆卸适配器。 |
| 闪烁 | 电源转换-热插拔操作正在进行中，不允许插入或拔出适配器。 |

**电源灯灭**

电源指示灯处于“灭”状态，表示允许插拔适配器。如果形状因素需要，则关闭槽的主电源。请注意，根据外形因素，即使主电源关闭且电源指示灯关闭，其他电源/信号也可能保持打开状态。在使用[CEM]外形的示例中，如果平台为热插拔插槽提供Vaux，并且MRL关闭，则即使电源指示灯关闭，MRL切换的任何信号也会连接到插槽。当MRL打开时，由MRL交换的信号断开。当插槽未上电或允许插拔适配器时，系统软件必须使插槽的电源指示灯熄灭。有关详细信息，请参考相应的外形规格。

**电源灯亮**

电源指示灯常亮，表示热插拔操作完成，槽位主电源打开，不允许插拔适配器。

**电源灯闪烁**

电源指示灯闪烁，表示槽位正在上电或下电，不允许插拔适配器。

当用户按下注意键或通过热插拔软件界面进行热插拔操作时，电源指示灯闪烁，并提供视觉反馈。

### 6.7.1.2手动保留闩锁(MRL)

MRL是一种手动操作的保留机制，它将适配器固定在插槽中，并防止用户移除设备。MRL刚性地将适配器固定在插槽中，以便电缆可以连接而不会产生间歇性接触的风险。在不提供MRL传感器的平台中，允许同时按住两个或多个适配器的MRL。

### 6.7.1.3 MRL传感器

MRL传感器是一种开关、光学设备或其他类型的传感器，它将槽位的MRL位置报告给下行端口。MRL传感器报告关闭时，MRL完全关闭，并在所有其他时间打开(即，如果MRL完全打开或处于中间位置)。

如果槽位安装了电源控制器，当MRL传感器检测到MRL断开时，必须自动将槽位主电源断开。

如果Vaux、SMBus等信号被MRL转换，当MRL传感器检测到MRL打开时，这些信号必须自动从槽位中移除，当MRL传感器检测到MRL再次关闭时，这些信号必须恢复到槽位中.

请注意，热插拔控制器不会根据MRL传感器的变化自动改变电源指示灯或注意力指示灯的状态。

**实现注意**

**MRL传感器处理**

在没有MRL传感器的情况下，对于某些外形因素，可以使用带外存在检测来处理开关信号。在这种情况下，当带外存在检测表明插槽中没有适配器时，被转换的信号将自动从插槽中移除。

如果MRL Sensor在热插拔控制器上没有对应的MRL Sensor输入，建议MRL Sensor路由到热插拔控制器的电源故障输入。这允许在MRL打开时关闭活动适配器

### 6.7.1.4机电联锁

机电联锁是一种物理锁定适配器或MRL的机制，直到系统软件释放它。机电联锁的状态由软件设置，除非响应随后的软件命令，否则不得改变。特别是，即使拔掉热插拔槽的电源，也必须保持机电联锁的状态。

机电联锁的当前状态必须始终反映在插槽状态寄存器中的机电联锁状态位中，该状态必须在任何命令更改后200毫秒内更新。软件必须在发出命令切换机电联锁状态后至少等待1秒，然后才能发出另一个切换状态的命令。系统可以选择性地扩展对联锁的控制，以提供适配器的物理安全性。

### 6.7.1.5注意按钮

注意按钮是一种瞬时触点按钮开关，位于每个热插拔插槽附近或在适配器上，用户按下该适配器以启动该插槽的热插拔操作。无论按钮的物理位置如何，信号都由与插槽对应的下行端口相关联的热插拔硬件处理并指示给软件。

注意按钮必须允许用户发起热添加和热删除操作，而不管按钮的物理位置如何

如果电源指示灯存在，则通过闪烁的方式向操作人员提供视觉反馈(如果系统软件接受了注意按钮发起的请求)。电源指示灯开始闪烁后，有5秒的中断间隔，在此期间，第二次按下注意按钮将取消操作。

如果注意按钮发起的操作由于任何原因失败，建议系统软件通过软件用户界面显示错误消息解释失败，或将错误消息添加到系统日志中。

### 6.7.1.6软件用户界面

系统软件提供了一个用户界面，允许发起热插拔，并允许监控占用的槽位。热插拔用户界面的详细讨论是特定于操作系统的，因此超出了本文档的范围。

在具有多个热插拔插槽的系统上，系统软件必须允许用户在每个插槽上独立于所有其他插槽的状态发起操作。因此，当另一个槽位正在进行热插拔操作时，用户可以通过软件用户界面或注意按钮对其中一个槽位发起热插拔操作，而不管第一次操作是通过哪个接口启动的。

### 6.7.1.7槽位编号

物理槽位标识符(定义见[PCI-Hot-Plug-1.1]，章节1.5)由可选的机箱号和槽位的物理槽位号组成。物理槽位号是机箱中槽位的唯一标识。系统软件根据端口中的寄存器确定物理槽位号。0号机箱预留给主机箱使用。其他机箱的机箱号必须是从PCI-to-PCI桥的机箱号寄存器中获得的非零值(参见[PCI-to-PCI-Bridge-1.2]，第13.4节)。

无论与每个槽位相关联的形状因素如何，每个物理槽位号在机箱中都必须是唯一的。

### 6.7.1.8电源控制器

电源控制器是由一个或多个分立组件组成的元件，在软件的控制下，根据特定的外形因素设置热插拔插槽的电源状态。电源控制器还必须监控槽位的电源故障情况(如相关的外形规格所定义)，这些故障发生在槽位的主电源轨和(如果支持的话)辅助电源轨上。

如果没有电源控制器，则热插拔控制器必须根据插槽中适配器的变化自动设置热插拔槽的电源状态。

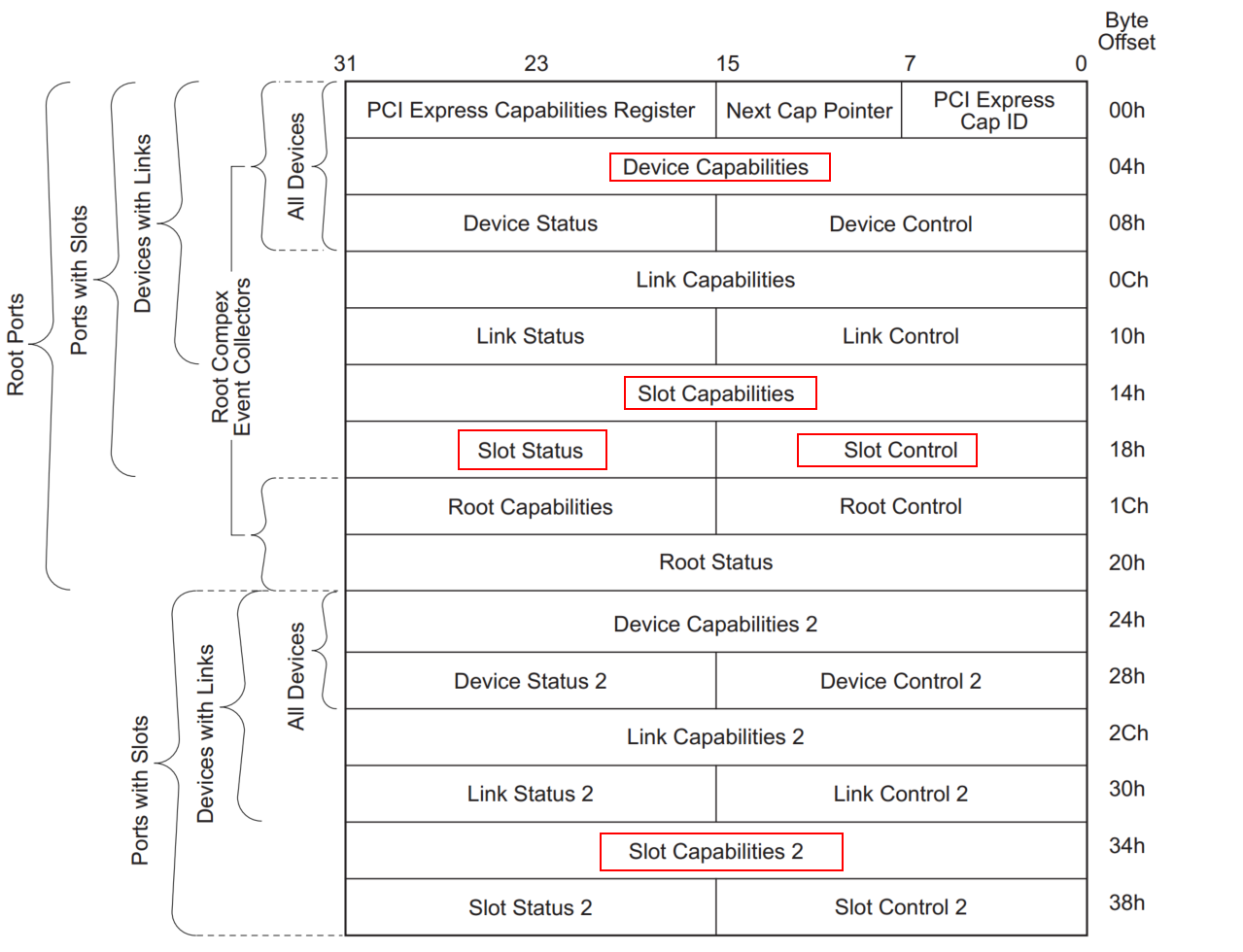
电源控制器独立监控主备电源故障。当电源控制器检测到热插拔槽位主电源故障时，应自动设置内部主电源故障锁扣，并将主电源从热插拔槽位中拔出(不影响辅助电源)。同样，当电源控制器检测到热插拔槽位辅助电源故障时，必须自动设置内部辅助电源故障锁扣，并在不影响主电源的情况下，将辅助电源从热插拔槽位取出。只要电源故障状态处于锁存状态，就必须保持插槽的电源关闭，无论软件是否进行写操作以打开热插拔插槽的电源。当软件关闭热插拔槽的电源时，主电源故障锁扣清除。辅助电源故障锁存器清除的机制是特定于外形因素的，但通常需要从热插拔插槽中移除辅助电源。例如，当打开插槽的MRL时，一种形式可能会移除辅助电源，而另一种形式可能需要将适配器从插槽中物理移除。有关具体要求，请参阅相关的外形规格。

由于插槽控制寄存器中的电源控制器控制位反映了最后写入的值，而不是电源控制器的实际状态，这意味着电源控制器控制位的值与电源故障条件下插槽的电源状态之间可能存在不一致。为了确定是否是由于电源故障导致槽位掉电，软件必须使用电源故障软件通知来检测电源故障。为了确定请求的上电操作是否失败，软件必须使用第6.7.3.3节中描述的热插拔槽上电超时机制。

软件不能假设写槽控制寄存器来改变热插拔槽的电源状态会导致立即的电源状态转换。上电后，软件必须等待数据链路层状态改变事件，如6.7.3.3节所述。关闭电源后，软件必须等待至少1秒才能执行任何操作这依赖于电源已经从热插拔插槽中移除。例如，软件不允许关闭电源指示灯(如果存在)或试图在完成1秒等待期之前打开电源控制器。

## 6.7.2根据热插拔元素关联分组的寄存器

本节中描述的寄存器按热插拔元素分组，以传递与实现每个元素相关的所有寄存器。与实现热插拔插槽的每个下游端口相关的寄存器字段位于PCI Express Capability结构中的Device Capabilities、slot Capabilities、slot Control、slot Status和slot Capabilities 2寄存器中(参见章节7.5.3)。寄存器报告与适配器上的设备功能相关的热插拔元素的存在，这些寄存器位于设备功能寄存器中(也在PCI Express Capability结构中)。



### 6.7.2.1注意按钮寄存器

注意按钮存在(槽位能力寄存器和设备能力寄存器)-该位表示注意按钮是由机箱(槽位能力寄存器)还是适配器(设备能力寄存器)电气控制的。（当设置1时，表示该槽位的注意按钮由机箱电气控制。）

注意按钮按下(槽位状态寄存器)-当机箱电控的注意按钮按下时设置此位。

注意按钮按下使能(槽位控制寄存器)-当设置时，此位使能注意按钮按下事件的软件通知(参见6.7.3.4节)。

当设置为1b时，此位启用关注按钮按下事件的软件通知(参见章节6.7.3)。

如果插槽能力寄存器中的注意按钮当前位为0b，则允许该位以0b的值为只读。

缺省值为0b

### 6.7.2.2注意指示灯寄存器

注意指示器存在(槽位能力寄存器和设备能力寄存器)-该位表示注意指示器是由机箱(槽位能力寄存器)还是适配器(设备能力寄存器)电气控制的。

注意指示灯控制(槽位控制寄存器)—写入时，将机箱电控的注意指示灯设置为写入状态。

### 6.7.2.3电源指示灯寄存器

电源指示灯在位(槽位能力寄存器和设备能力寄存器)-该位表示电源指示灯是由机箱(槽位能力寄存器)还是适配器(设备能力寄存器)电气控制的。

电源指示灯控制(槽位控制寄存器)—写时，将机箱用电控制的电源指示灯置于写状态。

### 6.7.2.4电源控制器寄存器

Power Controller Present (Slot Capabilities Register) -该位表示是否实现了电源控制器。

电源控制器控制(槽位控制寄存器)-根据写入的值打开或关闭电源控制器。

如果实现了电源控制器，则写入该位时根据定义的编码设置插槽的电源状态。这个位的读必须反映最近一次写的值，即使相应的热插拔命令没有完成，除非软件发出写命令，如果需要的话，而不等待前面的命令完成，在这种情况下，读值是未定义的。

请注意，在某些情况下，电源控制器可能独立于电源控制器控制设置，根据检测到的故障情况自动移除槽位电源或不响应上电请求。

定义的编码是:0b:Power On, 1b: Power Off

如果插槽能力寄存器中的电源控制器当前位为Clear，则对该位的写入无效，并且该位的读值未定义。

电源故障检测(槽位状态寄存器)-当在槽位或适配器检测到电源故障时设置此位

如果配置了支持电源故障检测的电源控制器，当电源控制器检测到该槽位的电源故障时，设置该位。请注意，根据硬件能力，有可能在任何时候检测到电源故障，而不受电源控制器控制设置或插槽占用的影响。如果不支持电源故障检测，则不能设置此位.

电源故障检测使能(槽位控制寄存器)-当设置时，该位使能电源故障事件的软件通知(参见6.7.3.4节)。

### 6.7.2.5状态检测寄存器

带内PD Disable Supported (Slot Capabilities 2 Register) -该位表示槽位是否支持禁用带内存在检测，允许带外存在检测状态独立于带内存在检测状态报告。

带内PD禁用(插槽控制寄存器)-当设置时，该位禁用带内存在检测机制，不影响存在检测状态位，允许该位专用于报告带外存在检测。

存在检测状态(槽位状态寄存器)-该位表示槽位中存在适配器。

在位检测改变(槽位状态寄存器)-当检测到在位检测状态改变时设置此位。

状态检测更改使能(槽位控制寄存器)-当设置时，该位使能软件通知状态检测更改事件(参见6.7.3.4节)。

### 6.7.2.6 MRL传感器寄存器

MRL Sensor Present (Slot Capabilities Register) -该位表示是否实现了MRL传感器。

MRL Sensor Changed (Slot Status Register) -当MRL Sensor的状态值发生变化时，设置该位。

MRL Sensor Changed Enable (Slot Control Register) -当设置时，该位使能MRL Sensor Changed事件的软件通知(见6.7.3.4节)。

MRL传感器状态寄存器(Slot Status Register) -如果实现了MRL传感器，该寄存器将报告MRL传感器的状态。

### 6.7.2.7机电联锁寄存器

机电联锁存在(槽位能力寄存器)-该位表示是否实现了机电联锁。

机电联锁状态(槽位状态寄存器)-这位反映机电联锁的当前状态。

机电联锁控制(插槽控制寄存器)-当设置为1b时，此位切换机电联锁的状态

### 6.7.2.8命令完成寄存器

No Command Completed Support (Slot Capabilities Register)—当该位设置为1b时，表示当热插拔控制器完成发出的命令时，该插槽不生成软件通知。

命令完成(槽位状态寄存器)-当热插拔控制器完成发出的命令并准备接受下一个命令时设置此位。

命令完成中断使能(插槽控制寄存器)-当设置时，当一个命令由热插拔控制逻辑完成时，该位使能软件通知(见6.7.3.4节)。

### 6.7.2.9端口能力和槽位信息寄存器

Slot Implemented (PCI Express Capabilities Register) -当设置时，该位表示与该下行端口关联的链路连接到一个槽位。

物理槽位号(槽位能力寄存器)-这个硬件初始化字段表示附加到端口的物理槽位号。

Hot-Plug Capable (Slot Capabilities Register) -当设置时，该位表示该槽位能够支持热插拔。

Hot-Plug Surprise (Slot Capabilities Register) -当设置时，该位表示该插槽启用了处理异步删除的Hot-Plug Surprise机制。参见6.7.6节。

### .6.7.2.10热插拔中断控制寄存器

热插拔中断使能(槽位控制寄存器)-当设置时，该位使能在使能的热插拔事件上产生热插拔中断。

## 6.7.3 PCI Express热插拔事件

具有热插拔功能的下行端口支持以下热插拔事件:

•槽位事件:

◦按下注意键

◦检测到电源故障

◦MRL传感器更改

◦状态检测改变

•命令完成事件

•数据链路层状态改变事件

这些事件中的每一个都有一个status字段，表示事件已经发生，但还没有被软件处理，还有一个enable字段，表示事件是否为软件通知启用。有些事件还有capability字段，用于指示端口是否支持该事件类型。按事件类型对这些字段进行分组将在第6.7.2节中列出，每个单独的字段将在第7.5.3节中进行描述。

### 6.7.3.1槽位事件

具有热插拔功能的下游端口监视它所控制的插槽，以查看上面列出的插槽事件。当检测到其中一个槽位事件时，Port通过设置与该事件相关的状态字段来指示该事件已经发生。此时，事件处于挂起状态，直到软件清除状态字段。

一旦某个槽位事件挂起，该槽位上所有该类型的后续事件都会被忽略，直到该事件被清除。端口必须继续监视槽的所有其他槽事件类型，并在它们发生时报告它们。

如果通过关联的enable字段使能，槽位事件必须产生软件通知。如果端口上不支持该事件(如关联的capability字段所示)，则软件不得为该事件启用软件通知。第6.7.3.4节描述了将此通知报告给软件的机制。

### 6.7.3.2命令已完成事件

由于改变一些热插拔元素的状态可能不会立即发生，因此PCI Express支持热插拔命令和命令完成事件。所有支持热插拔的端口都需要支持热插拔命令，如果报告了该能力，则需要支持命令完成事件。

软件通过发出针对端口槽控制寄存器的任何部分的写事务，向具有热插拔能力的下游端口发出命令。对槽控制寄存器的单个写操作被认为是单个命令，即使该写操作影响槽控制寄存器中的多个字段。为了响应此事务，端口必须执行请求的操作，然后为命令完成事件设置相关的状态字段。

即使命令发出时已经设置了状态字段，端口也必须正常处理命令。如果单个命令导致启动多个操作，则未指定操作执行的顺序。与单个命令执行相关的所有操作不能超过1秒。

如果命令完成事件不被支持，如Slot Capabilities寄存器的No command completed Support字段中的值1b所示，则具有热插拔功能的端口必须处理一个写事务，该事务针对端口的Slot Control寄存器的任何部分，而不依赖于先前的Slot Control写。软件被允许按顺序发出多个插槽控制写入，而在写入之间没有任何延迟。

如果支持命令完成事件，则软件必须等待命令完成后再发出下一个命令。但是，如果在命令执行的1秒限制之后没有设置状态字段，则允许软件重复该命令或发出下一个命令。如果软件在端口完成前一个命令的处理之前发出写操作，并且在1秒的时间限制之前，端口可以接受或放弃写操作。这样的写入被认为是一个编程错误，并且可能导致槽控制寄存器和热插拔元件状态之间的差异。为了从这种编程错误中恢复并使控制器返回到一致状态，软件必须向插槽控制寄存器发出符合命令完成规则的写操作。

如果通过关联的enable字段启用，则命令的完成必须生成软件通知。

此规则的例外是由于写入Slot Control寄存器而导致命令完成事件的软件通知失效的命令。这样的命令必须如上所述进行处理，但不能生成软件通知。

### 6.7.3.3数据链路层状态改变事件

数据链路层状态改变事件表明链路状态寄存器中数据链路层链路活动位的状态发生了变化。支持数据链路层状态改变事件和这些事件的软件通知是需要热插拔的下行端口。如果支持此事件，则当数据链路层链路活跃位的值发生变化时，端口设置与该事件相关的状态字段。

此事件允许软件间接确定电源何时已应用于新热插拔适配器。

软件必须在Data Link Layer Link Active位读到1b后等待100ms，才能对热添加设备发起配置访问(参见6.6节)。软件必须允许在数据链路层链路激活位读到1b后1秒，然后才允许确定热插拔设备未能返回一个有效配置请求的成功完成是一个损坏的设备(参见章节6.6)。

数据链路层状态改变事件必须在引发热插入事件的1秒内发生。如果支持电源控制器，则从软件启动写入槽控制寄存器到打开电源时开始测量超时间隔。如果不支持电源控制器，超时时间从存在检测槽位事件开始计算。如果在1秒内没有发生数据链路层状态改变事件，则允许软件进行热添加操作时超时。软件在此超时后所采取的动作是具体实现的。

### 6.7.3.4热插拔事件软件通知

支持热插拔的下行端口必须支持在热插拔事件上产生中断。如第6.7.3.1节和6.7.3.2节所述，每个热插拔事件都有一个中断生成的使能位和一个状态位，该状态位表示事件发生但尚未被软件处理。在槽控制寄存器中也有一个热插拔中断使能位，作为所有热插拔事件的主启用/禁用位。

如果端口启用了使用INTx消息的电平触发中断信令，只要满足以下条件，就必须断言虚拟INTx线:

•命令寄存器中的中断禁用位设置为0b。

•插槽控制寄存器中的热插拔中断使能位设置为1b。

•Slot status寄存器中至少有一个热插拔事件状态位和Slot Control寄存器中相关的使能位都设置为1b。

请注意，在请求服务时，同一函数中的所有其他中断源都将断言相同的虚拟INTx线。

如果端口使用MSI或MSI- x启用边缘触发中断信令，则每次以下条件的逻辑与从FALSE转换为TRUE时，必须发送中断消息:

* •关联向量被揭开(如果MSI不支持PVM，则不适用)。

•插槽控制寄存器中的热插拔中断使能位设置为1b。

•Slot status寄存器中至少有一个热插拔事件状态位和Slot Control寄存器中相关的使能位都设置为1b。

请注意，PME和热插拔事件中断(当两者都实现时)总是共享相同的MSI或MSI- x向量，如PCI Express Capabilities寄存器中的中断消息编号字段所示。

当中断生成被禁用时发生热插拔事件，并且随后启用中断生成时，端口可以选择发送MSI。

如果相关的外形规格要求产生唤醒事件，那么支持热插拔的下游端口必须支持在系统处于睡眠状态或端口处于设备状态D1、D2或D3H时发生的热插拔事件产生唤醒事件(使用PME机制)。

软件使热插拔事件能够产生唤醒事件，方法是启用软件通知，如6.7.3.1节所述。请注意，为了使软件在保持唤醒生成使能的同时禁用中断生成，必须清除热插拔中断使能位。对于支持唤醒生成的形状因子，如果满足以下三个条件，则必须生成唤醒事件:

•使能事件的状态寄存器从Clearto Set转换

•端口处于设备状态D1, D2或D3等

•端口电源管理控制/状态寄存器中的PME\_En位为Set。

请注意，热插拔控制器代表热插拔设备产生唤醒，该设备不需要有辅助(或主)电源。

## 6.7.4系统固件中介(SFI)支持

系统固件中介(SFI)能力是下游端口的可选规范特性。一些SFI功能专注于热插拔插槽，如Slot Capabilities寄存器中正在设置的Hot-Plug Capable位所示，而一些SFI功能在该上下文中是有用的。如果下游端口支持SFI能力结构，则必须设置以下位:

•链路能力寄存器中的数据链路层链路主动报告能力位

•链路能力2寄存器中的DRS支持位

•设备能力寄存器中的ERR\_COR子类能力位

### 6.7.4.1 SFI ERR\_COR事件信令

SFI功能不支持生成INTx或MSI/MSI- x中断，因为该功能旨在由系统固件使用。

具有SFI的下行端口必须支持ERR\_COR信令，而不管它是否支持高级错误报告(AER)。SFI ERR\_COR事件信令由SFI控制寄存器中的SFI OOB PD Changed Enable、SFI DLL State Changed Enable和SFI DRS signaling Enable位独立启用。这些事件由SFI状态寄存器中的SFI OOB PD Changed、SFI DLL State Changed和SFI DRS Received位表示。

如果在设备控制寄存器中设置了可纠错报告使能位，则每次满足使能条件之一时，端口必须发送ERR\_COR消息。SFI ERR\_COR事件信令不能在设备状态寄存器中设置可纠正错误检测位，因为此事件不作为错误处理。

**实现注意**

**DPC DL\_Active与SFI DLL状态改变的ERR\_COR信令**

DPC为DL\_Active实现ERR\_COR信令，而SFI为SFI DLL State Changed实现ERR\_COR信令，它们是相关但不相同的条件。DL\_Active条件发生在链路状态寄存器中的数据链路层链路活动位从0b变为1b时，并且该位可以被SFI控制寄存器中的SFI DLL状态掩码位屏蔽。当SFI状态寄存器中的SFI DLL状态位通过变为Set或变为Clear改变其值时，就会发生SFI DLL State Changed情况，这种情况总是基于实际的数据链路层状态。

### 6.7.4.2 SFI下行端口过滤(DPF)

下游端口过滤(DPF)是一种机制，在这种机制中，下游端口可以处理针对其下组件的指定请求TLPs，就好像链接处于DL\_Down中一样。参见2.9.1节。

DPF有两种过滤请求tlp的模式，目标是下游端口以下的组件。第一种模式过滤所有这样的请求tlp;第二种模式只过滤配置请求TLPs。其他tlp不能被DPF过滤或阻断。

DPF的一个关键用例是保证异步系统软件活动(如总线扫描)不会在常规重置后无意中向尚未准备好的设备发送配置请求，因为这样的访问会导致未定义的硬件行为。参见6.6.1节。

DPF的另一个关键用例是支持固件优先功能，使系统固件在收到异步热添加通知后，能够在使设备对操作系统可见之前配置新添加的设备。对于这个用例，SFI CAM机制使下游端口本身能够生成针对下游组件的配置请求TLPs，并且这些TLPs不会被DPF机制过滤或阻止。参见6.7.4.3节、7.9.21.5节和7.9.21.6节。

### 6.7.4.3 SFI CAM

当启用DPF时，SFI配置访问方法(CAM)为SFI感知系统固件提供了一种方法，使下游端口代理(通过)配置请求瞄准下游端口以下的组件。SFI CAM始终处于启用状态。

要使用SFI CAM，软件首先写入SFI CAM地址寄存器，指定目标配置地址。

然后，软件读取或写入SFI CAM数据寄存器，以生成代理配置请求并将其传输到下游组件。

以下规则适用:

•用于代理配置请求的所有TLP字段与针对SFI CAM数据寄存器的配置请求中的TLP字段相同，有以下例外:

◦目标总线号，设备号和功能号来自SFI CAM地址寄存器。

◦扩展寄存器号和寄存器号来自SFI CAM地址寄存器。

◦重新生成LCRC。

◦如果存在，则再生ECRC。

•SFI CAM不能对请求应用完成超时机制。

•系统固件必须确保在写入SFI CAM地址寄存器和随后读取或写入SFI CAM数据寄存器完成之间，没有其他线程修改SFI CAM地址寄存器;否则，结果为未定义。

•如果检测到与代理配置请求相关的错误，则报告与下游端口实现SFI CAM相关的错误(参见6.2节)。

•上游完井必须未经修改通过下游端口。

**实现注意**

**SFI CAM地址和数据寄存器的序列化使用**

**如上所述，系统固件必须确保在写入SFI CAM地址寄存器和随后读取或写入SFI CAM数据寄存器完成之间，没有其他线程修改SFI CAM地址寄存器。例如，可以使用信号量或其他同步机制来确保这种序列化。**

**对于处理器存储到配置空间的指令被有效发布的平台，软件仍然必须确保在另一个软件线程修改SFI CAM数据寄存器之前完成生成的配置写。在这样的平台上，确定Configuration Write何时完成的机制是特定于平台的。**

**给定适当的序列化，即使根复合体自动将配置请求作为新请求重新发出，SFI CAM也可以正确地处理导致CRS补全的配置请求。**

**重新发出的配置请求将再次发送到SFI CAM数据寄存器，关联的下游端口将再次生成针对下游组件的配置请求。只要SFI CAM地址寄存器在配置请求完成之前没有被其他软件修改，该序列就可以无限重复，直到返回非crs完成或发生完成超时。**

**当启用CRS软件可见性时，SFI CAM仍然可以正确地处理导致CRS完井的配置请求。具有CRS完成状态的任何完成都将返回到原始请求程序，该请求程序根据CRS软件可见性语义的要求处理它们。参见第2.3.2节。**

**实现注意**

**与SFI CAM一起使用分配的Bus number**

**当下游端口启用DPF时，SFI CAM可以被SFI感知的系统固件用来配置和访问端口下面的子层次结构，而不需要其他软件能够这样做。虽然端口下面的总线号配置通常对其他软件不可见，但配置用于端口下面的总线号应限于已分配给端口的总线号，因为通过端口上行的tlp可能包含配置了总线号的id。如果这些tlp检测到并记录了任何错误，那么其他软件就可以看到这些总线号，如果它们与系统中其他地方使用的总线号重叠，就会造成混乱。**

### 6.7.4.4 SFI与就绪通知的交互

SFI功能能够屏蔽接收到的设备就绪状态(DRS)消息的报告，并模拟它们被接收。当使用SFI的下游端口过滤(DPF)机制来阻止操作系统对下游端口以下的设备或子层次结构的可见性时，此功能非常有用。

规则:

•当设置了SFI DRS掩码位时，链路状态2寄存器中的DRS报文接收位必须为0b。

•SFI DRS Received位必须总是指示DRS Message Received的实际状态。

•当SFI DRS掩码位为Clear并且向SFI DRS触发位写入1b时，下行端口必须表现得像收到了DRS消息一样。

**实现注意**

**设备准备的SFI透明优化**

**在热添加操作之后，某些设备可能需要比允许的更长时间才能达到配置就绪状态。参见6.6.1节。**

**如果系统固件知道这样的设备，它可以使用SFI DPF机制阻止操作系统对新添加的设备的可见性，等待必要的时间使设备成为配置就绪，然后将设备公开给操作系统。**

**为了避免操作系统不必要地等待额外的时间以使新暴露的设备成为配置就绪，系统固件可以使用SFI DRS触发器位让下游端口模拟DRS消息的接收。支持DRS的操作系统可以立即发现并配置新公开的设备。**

**新暴露的设备本身并不一定需要具有DRS功能。由于上游端口被明确允许发送DRS消息，即使它的DRS支持位是Clear，它上面的下游端口可以合法地模拟从它接收DRS消息，即使它不能发送DRS消息。**

**还应该注意的是，在系统固件意识到设备提前成为配置就绪的情况下，系统固件可以使用SFI DRS触发机制将其暴露给操作系统。**

**虽然SFI不打算由操作系统软件使用，但建议支持SFI的平台上使用的操作系统实现对DRS的支持，这样整个系统就可以从这种优化的Device Readiness时序中获益。**

**实现注意**

**DPF和功能就绪状态(FRS)消息**

**DPF (Downstream Port Filtering)不影响FRS消息的生成和传播。当设备作为异步热添加操作的一部分准备就绪时，不会生成FRS消息。但是，如果系统固件在设备上执行导致FRS事件的操作，则产生的FRS消息可能对操作系统是可见的。参见2.2.8.6.4和6.23.2节。**

### 6.7.4.5 SFI抑制热插拔意外功能

如果插槽支持HPS (Hot-Plug Surprise)功能，如slot Capabilities Register中的Hot-Plug Surprise位所示，则SFI控制寄存器中的SFI HPS Suppress位可用于强制Hot-Plug Surprise位为Clear，并禁用相关的Hot-Plug Surprise功能。

当下游端口/槽位组合同时支持HPS和下游端口遏制(DPC)时，HPS抑制是有用的。不建议DPC与HPS同时使用，因此如果插槽启用了HPS功能，则不应该启用DPC。如果软件希望使用DPC，软件应该首先设置SFI HPS抑制位，以禁用HPS功能，使DPC正常工作。

**实现注意**

**热插拔意外功能的软件协商**

**假设系统固件拥有SFI Capability结构，建议为了与旧操作系统向后兼容，在支持异步移除的插槽上默认启用Hot-Plug Surprise功能。然后，如果插槽也支持DPC，并且操作系统希望使用它，操作系统将请求系统固件抑制HPS，系统固件将决定是否设置或清除SFI HPS抑制位。**

## 6.7.5固件支持热插拔

一些系统包括支持热插拔的根端口和交换机，这些系统是在ACPI兼容的操作系统支持本地热插拔之前发布的，可以使用ACPI固件来传播热插拔事件。如果使用具有本机支持的操作系统，则必须禁用热插拔寄存器的固件控制。提供ACPI固件来传播热插拔事件的平台还必须提供将控制传递给操作系统的机制。该方法的细节在PCI固件规范中描述。

## 6.7.6异步删除

异步删除是指在没有事先警告操作系统的情况下，由于包含错误而删除适配器或禁用下游端口链接。这与有序移除形成对比，有序移除操作是通过定义良好的用户操作序列和系统管理设施，以锁定步骤的方式与操作系统一起执行的。例如，用户按下Attention Button请求操作系统删除适配器的权限，但是在操作系统停止适配器的活动并授予删除权限之前，用户实际上并没有从插槽中删除适配器。

由于异步删除在PCI Express层次结构的其余部分或操作系统必须意识到该事件之前进行，因此除了标准PCI热插拔之外，还需要特别考虑。本节概述了可能作为异步删除的副作用而发生的PCI Express事件以及处理异步删除的机制。

由于与插槽相关的下行端口的物理层和数据链路层都可能意外地删除异步，因此可纠正错误可能作为事件的副作用(即接收器错误，坏的TLP和坏的DLLP)报告。如果报告了这些错误，软件应该将它们作为此事件的预期部分来处理。

请求程序可能会遇到与已接受的请求相关的完成超时，但永远不会被删除的完成程序完成。在此上下文中产生的任何完成超时错误都应作为此事件的预期部分进行处理。

异步删除可能导致下游端口从DL\_Active到DL\_Down的转换。这种转换可能导致意外关闭错误。此外，PCI Express层次结构域中的请求者可能不会立即意识到这种转换，并继续向必须由与插槽关联的下游端口处理的已移除的完成者发出请求。

作为整体异步热插拔架构的一部分，下游端口遏制(DPC)或热插拔意外(HPS)机制都可用于支持异步移除。相关参考模型见附录1。

**实现注意**

**热插拔意外机制不支持异步热插拔**

**正如Slot Capabilities Register中的Hot-Plug Surprise位所示，热插拔意外(HPS)机制不适合用于异步热插拔。DPC是支持异步热插拔的推荐机制。关于支持这两种机制的槽位，请参见6.7.4.4节。**

**对于异步删除，使用HPS有严重的缺点。除了那些本质上导致链接中断的错误之外，不可纠正的错误需要被配置为使系统崩溃，由软件异步处理，或者被忽略。这些错误包括与已发布内存写相关的不可纠正错误、带有有毒数据的tlp和完成超时。被软件忽略或异步处理的不可纠正的错误可能使驱动程序无法确定哪些高级操作成功完成，哪些没有。**

**DPC提供了支持异步删除的健壮机制。TLP流在触发DPC的不可纠正错误时干净地停止。支持遏制错误恢复(CER)的操作系统/驱动程序堆栈可以完全透明地从许多瞬时PCIe不可纠正错误中恢复。DPC可以同时支持异步删除和CER**