# PCIe协议

## 2.2 事务层协议—包定义

##### 2.2.8.6.3 设备就绪状态(DRS)消息

设备就绪状态（DRS）协议使用PCI-SIG定义的VDM机制（参见6.23.1节）。DRS消息是PCI-SIG定义的VDM（Vendor-Defined Type 1 Message），该消息没有有效负载。

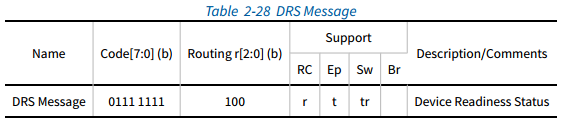
除了其他PCI-SIG定义的VDM规则之外，以下规则适用于DRS消息的形成:

表2-28和图2-31展示和定义了DRS消息。

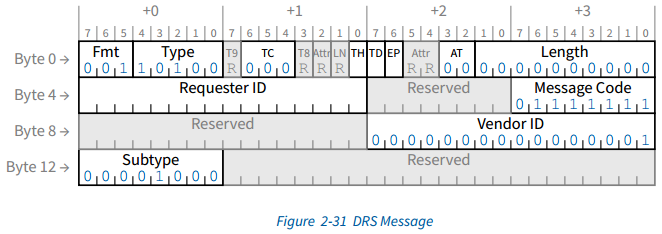
* TLP类型必须为Message。
* TC[2:0]字段必须为000b
* Attr[2:0]字段是保留的。
* Tag字段是保留的。
* Subtype字段必须为08h。
* “Message Routing”字段必须设置为100b——本地终止于接收方。

接收方可以选择检查是否违反这些规则(但不能检查保留位)。这些检查是独立可选的(参见6.2.3.4节)。如果执行这些检查的接收器确定某个TLP违反了这些规则，则该TLP是畸形TLP。

* 如果检查到该错误，那么这是一个与接收端口相关的报告错误(参见6.2节)。



DRS消息的格式如下图2-31所示：



## 6.23 准备就绪通知(RN)

准备就绪通知(RN)旨在减少软件在DRS事件或FRS事件之后向设备或功能发出配置请求之前需要等待的时间。RN包括设备就绪状态(DRS)和功能就绪状态(FRS)机制。这些机制提供了配置就绪的直接指示(参见“配置就绪”的术语和缩略词条目)。当使用DRS和FRS时，DRS和FRS允许改进CRS机制的行为，并且在重置后消除其相关的长达1秒的周期性轮询时间。

允许系统软件/固件提供取代FRS和/或DRS的机制，但是此类软件/固件机制超出本规范的范围。

* **实现时注意——优化配置准备**

强烈建议系统固件/软件的实施者尽可能避免不必要的延迟。强烈建议将硬件设计为消除或最大限度地减少所需的延迟，并充分利用本规范和相关规范中提供的机制来传达需要的延迟（如果有的话）。硬件实施者应适当记录实施行为，以使系统固件/软件能够实施最佳行为

即使有良好的文档，有些情况一开始可能会出现问题——例如，当需要从根端口配置空间读取时，系统固件如何从设备就绪状态(DRS)机制中受益？在这种情况下，需要平台特定知识，即Root Port支持立即就绪。

### 6.23.1 设备准备状态(DRS)

当实现时，DRS用以指示Device进入了Configuration-Ready状态。以下Device-Level的事件用以触发DRS，称为为DRS事件：

* 退出冷复位；
* 退出暖复位、热复位、Loopback、或者Disable；
* 退出L2/3 ready；
* 其他场景，例如端口由DL\_Down变为DL\_Up状态。

DRS消息协议要求如下：

* DRS没有启用或禁用机制。对于支持DRS的下行端口，必须设置Link Capabilities 2 Register中的DRS Supported位。对于支持DRS的上行端口，强烈建议设置Link Capabilities 2 Register中的DRS Supported位。对于上游端口，即使DRS Supported位为Clear，也允许发送DRS消息。
* 当与上游端口关联的Logic Bus上的所有非VF功能准备就绪时，在每次DL\_Down到DL\_Up转换之后，DRS消息必须由具有DRS功能的上游端口传输。
  + Type 0功能处于Configuration-Ready状态时，表示该功能已准备就绪。
  + Type 1功能作为Switch上行端口，当其处于Configuration-Ready状态时，其Secondary Bus上的所有功能都处于Configuration-Ready状态。
  + 非Switch上行端口的Type 1功能本身处于Configuration-Ready状态时即为ready。
* Device发出DRS消息后，除非后续发生了DRS事件，否则跟该DRS对应的Configuration-Ready的非VF Function不应返回CRS Completion消息。

与实现DRS的Switch相关的其他要求包括:

* Switch必须在所有端口中支持DRS功能。
* 在DRS信令控制字段的每个下行端口处的实现。
* 对于出现在Switch下行端口下的任何物理集成设备，Switch发送的DRS并不表示该设备的配置就绪情况。
  + 对于设备，DRS的实现和使用是独立于Switch的。

Root Port和Switch下行端口的附加要求包括:

* Switch下行端口及RP中需实现DRS Message Received位，以指示其接收到了DRS消息。
* **实现时注意——DRS消息和ACS源验证**

Function在被分配一个Bus Number之前被允许传输DRS消息。该消息Requester ID中的Bus Number为00h。如果下游端口启用了ACS源验证，这些消息(参见6.12.1.1节)可能会被检测为ACS违规错误。

### 6.7.4 系统固件中介(SFI)支持

系统固件中介（SFI）功能是下游端口的可选规范功能。一些SFI功能集中在可热插拔插槽上，如正在设置的插槽功能寄存器中的“可热插拔”位所示，而一些SFI函数在该上下文之外是有用的。如果下行端口支持SFI能力结构，则必须设置以下位：

* Link Capabilities register中的Data Link Layer Link Active Reporting Capable位
* Link Capabilities 2 register中的DRS Supported位
* Device Capabilities register中具有ERR\_COR Subclass Capable 位

#### 6.7.4.4 SFI与就绪通知的交互

SFI能力能够屏蔽接收到的设备就绪状态（DRS）消息的报告，并模拟正在接收的消息。当SFI的下游端口过滤（DPF）机制用于阻止下游端口下的设备或子层次结构的操作系统可见性时，此功能非常有用。规则如下：

* 当设置了SFI DRS Mask位时，Link Capabilities 2 register中的DRS消息接收位必须为0b。
* SFI DRS Received位必须始终指示DRS消息接收条件的实际状态。
* 当SFI DRS Mask位为Clear（清除）并且向SFI DRS Trigger（SFI DRS触发）位写入1b时，下行端口必须表现为收到DRS消息。
* **实现时注意——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。