

## 什么是 PCIe INTx 中断？

PCIe 总线有两种中断方式，INTx 和 MSI。

PCIe 中的 INTx 中断是为了兼容早期 PCI 特性及一些应用软件，INTx 边带信号输出连接中断控制器，再输出给 CPU。

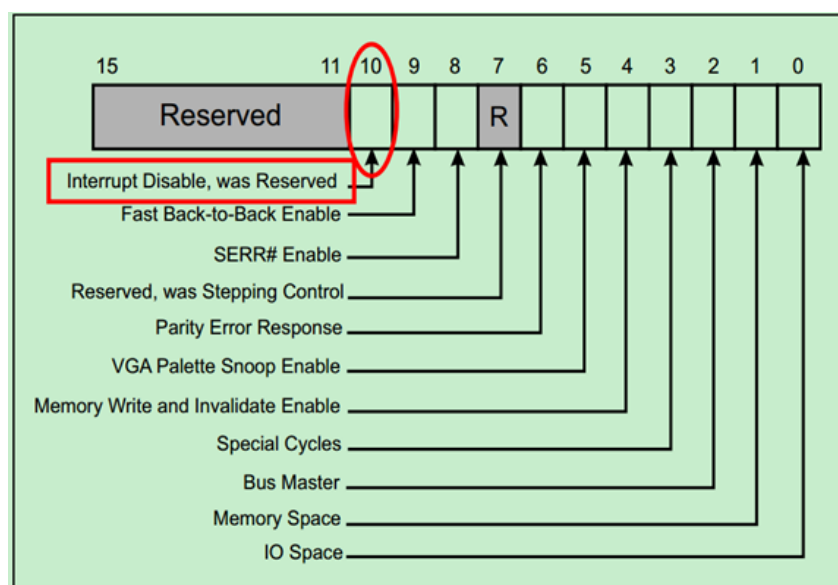
INTx 中断机制是基于边带信号，为了降低系统硬件设计的复杂性，使用基于消息（发 Posted Write 到特定地址）的 MSI 中断机制。PCIe 总线虽然继承 PCI 总线的 INTx 中断机制，但是 PCIe 设备之间的中断信息传输并非使用边带信号，而是基于消息 Message。

INTx 一共有四个中断，INTA#、INTB#、INTC#、INTD#。PCIe 协议通过 Message 来模拟物理线来产生 INTx，并且定义了两种类型，一种用于触发叫——Assert\_INTx，一种用于解除叫——Deassert\_INTx。其中 Assert\_INTx 消息表示 INTx 信号的下降沿。

Deassert\_INTx 消息表示 INTx 信号的上升沿。当发送这两种消息时，PCIe 设备还会将配置空间的相关中断状态 bit 的值更新。

## 如何配置 INTx

想要触发 INTx 中断，**第一步**需要配置 PCIe 总线配置空间中的命令寄存器（**Configuration Command Register**）来禁止 INTx 中断机制。如下图 10bit，就是 INTx 专属的使能控制信号，不会控制 MSI 相关功能。**第二步** disable MSI 中断功能。



**第三步**，PCIe 进入能触发中断相关状态位的场景。

## 如何检查 INTx

INTx Message 有 8 种，分别为 Assert/Deassert INTA、INTB、INTC、INTD。所有 Ports 的 Message INTx 只送给 Upstream Port，检查 Message Code [7:0] 是否是场景中正常触发并解除的 INTx 中断。

在 PCIe Switch 设备中，由于有多 port 口，PCIe 协议规定了经过 switch 桥的 INTx 映射关系。根据 device number，四种中断进入 switch 前的 INTx 与出来的 INTx 需要符合如下图例。

Device Number of Delivering INTx	INTx Message Type at Input	INTx Message Type at Output
0, 4, 8, 12 etc.	INTA	INTA
	INTB	INTB
	INTC	INTC
	INTD	INTD
1, 5, 9, 13 etc.	INTA	INTB
	INTB	INTC
	INTC	INTD
	INTD	INTA
2, 6, 10, 14 etc.	INTA	INTC
	INTB	INTD
	INTC	INTA
	INTD	INTB
3, 7, 11, 15 etc.	INTA	INTD
	INTB	INTA
	INTC	INTB
	INTD	INTC

在这里稍微解释下为什么要做一下映射，用一句话说就是为了尽量避免 function 共用一个 INTx。如果一个 RC 下游所有设备的 function 都使用 INTA，没有桥做映射，就会路由到同一个 IRQ。那么当 INTA 产生了，就要检查所有的 function，这样系统就造成了延迟。所以将 INTx 按照 device number 粗略划分成 4 组进行映射，尽量减少延迟。