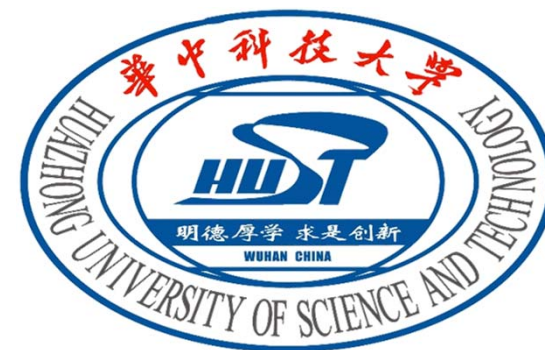


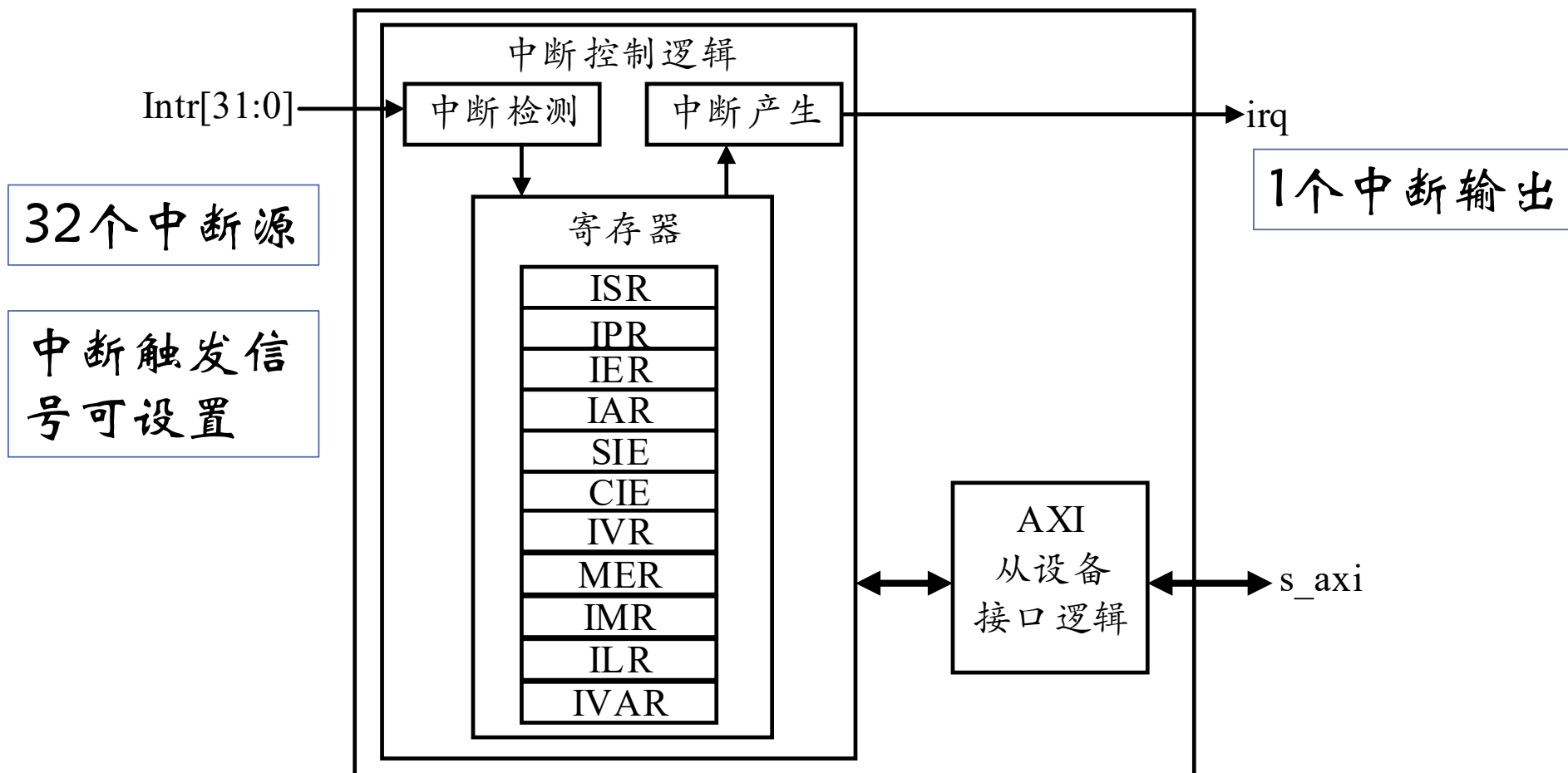
微机原理与接口技术

AXI INTC 中断控制器

华中科技大学 左冬红



INTC基本结构框图



INTC工作模式

普通中断模式：由总中断服务程序软件查询中断状态寄存器识别中断源、由总中断服务程序软件调用具体中断源的中断服务程序，总中断服务程序软件维护中断向量表

快速中断模式：由硬件识别中断源，并在CPU的中断响应周期直接输出中断源的中断向量，INTC硬件提供中断向量表IVAR

INTC寄存器存储空间映射

寄存器名称	偏移地址	含义
ISR	0x0	中断状态寄存器
IPR*	0x4	中断悬挂寄存器
IER	0x8	中断使能寄存器
IAR	0xC	中断响应寄存器
SIE*	0x10	中断使能设置寄存器
CIE*	0x14	中断使能清除寄存器
IVR*	0x18	中断类型码寄存器
MER	0x1C	主中断使能寄存器
IMR*	0x20	中断模式寄存器
ILR*	0x24	中断级别寄存器
IVAR*	0x100~0x170	中断向量表寄存器

INTC寄存器含义

ISR\IER\IAR\IPR\SIE\CIE\IMR

寄存器的 D_i 对应中断源引脚 $Intr_i$

都是1有效

IVR

寄存器保存最高优先级中断源的编码

所有中断源使能，且中断请求输入引脚 $intr[0]$ 、 $intr[1]$ 、 $intr[2]$ 都没有产生中断请求， $intr[3]$ 产生了中断请求，此时IVR的值为0x3；如果之后 $intr[0]$ 产生中断请求，那么IVR的值变为0x0。

INTC寄存器含义

ILR

寄存器保存阻止的最高优先级中断源的编码

ILR为0x0阻止所有中断源

ILR为3表示阻止中断源intr[31:3]，仅允许中断源intr[2:0]

IVAR

共32个寄存器，每个寄存器都是4B

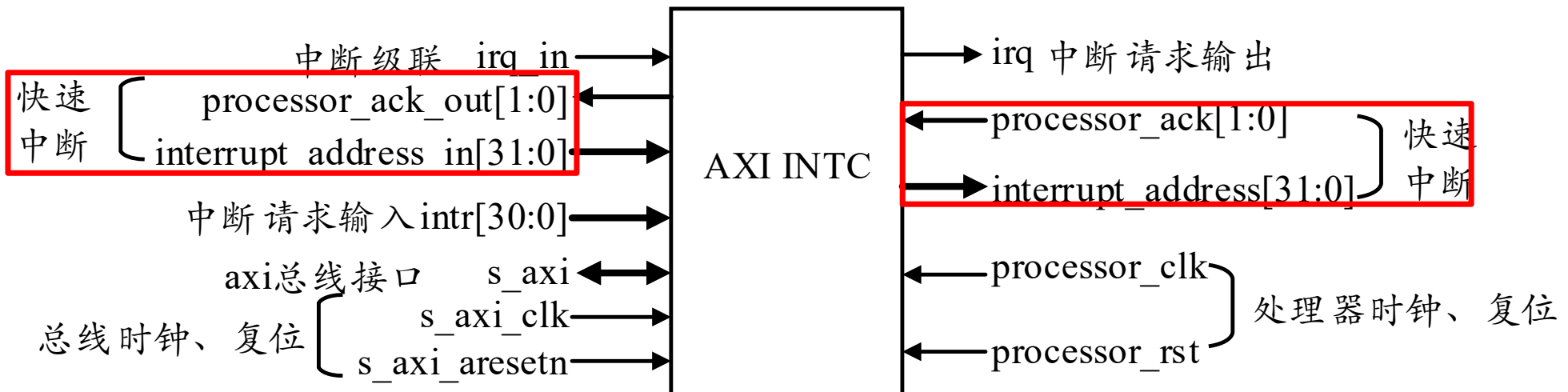
中断请求引脚intr[i]与IVAR偏移地址EA之间关系为：
 $EA = 0x100 + 4 \times i$

MER

HIE(D1)为1表示使能硬件中断

ME(D0)为1表示允许Irq产生中断请求信号

INTC外部引脚



普通中断模式：由软件查询ISR识别中断源、由软件调用具体中断源的中断服务程序

快速中断模式：由硬件识别中断源，并通过Interrupt_address输出相应中断源的中断向量

INTC中断处理流程

引脚intr[31:0]接收到中断请求

中断状态保存在ISR中，并与IER相“与”

产生Irq信号

快速中断模式

中断响应周期，INTC送出中断向量，并清除中断状态寄存器相应位

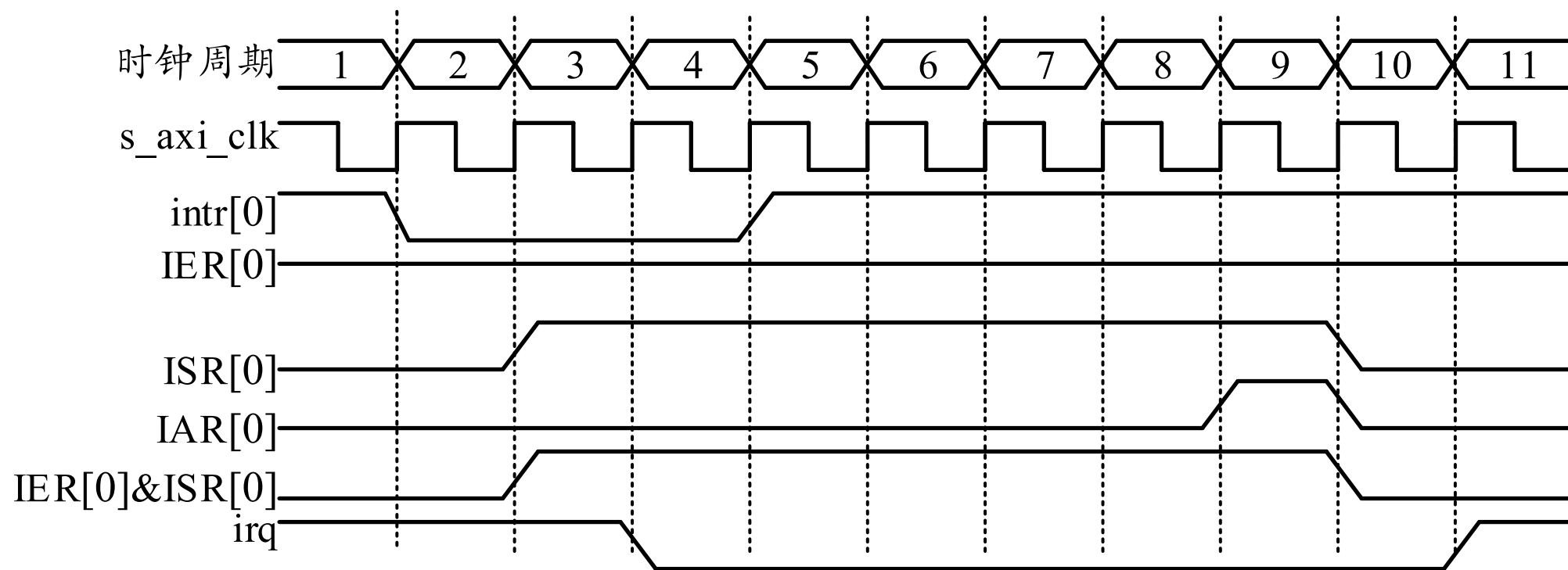
普通中断模式

优先级判定电路，设置IVR

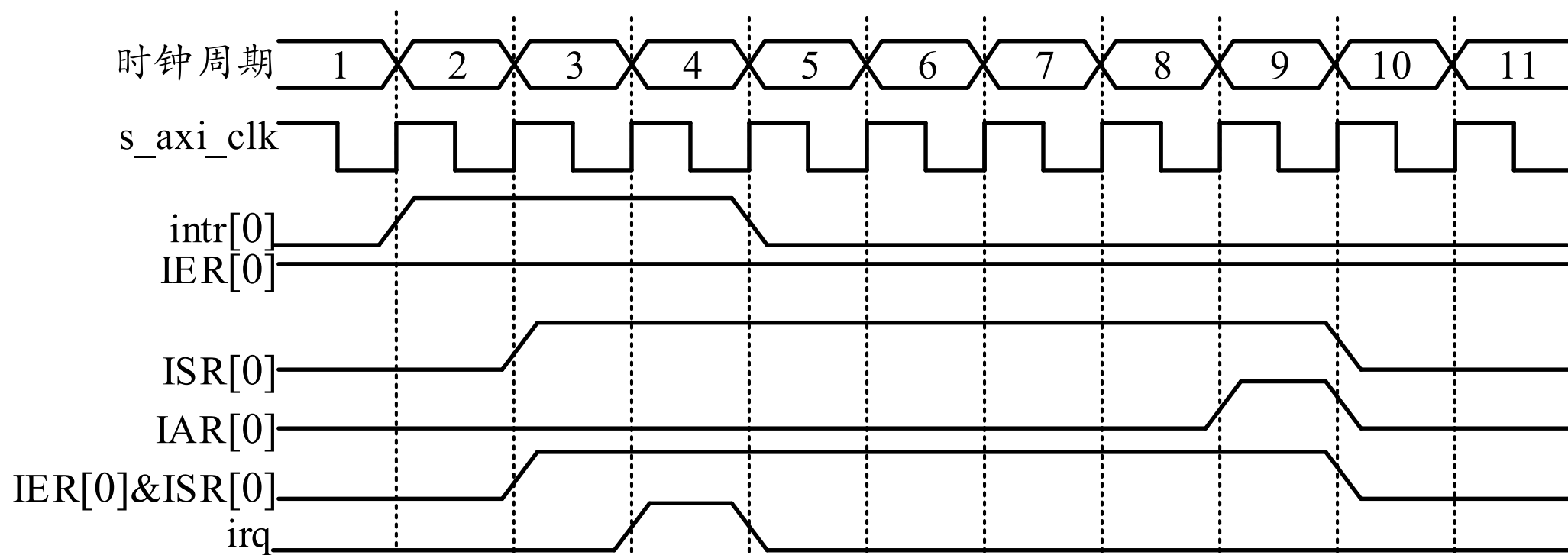
软件读取IVR或ISR识别中断源

软件写IAR清除中断状态寄存器相应位

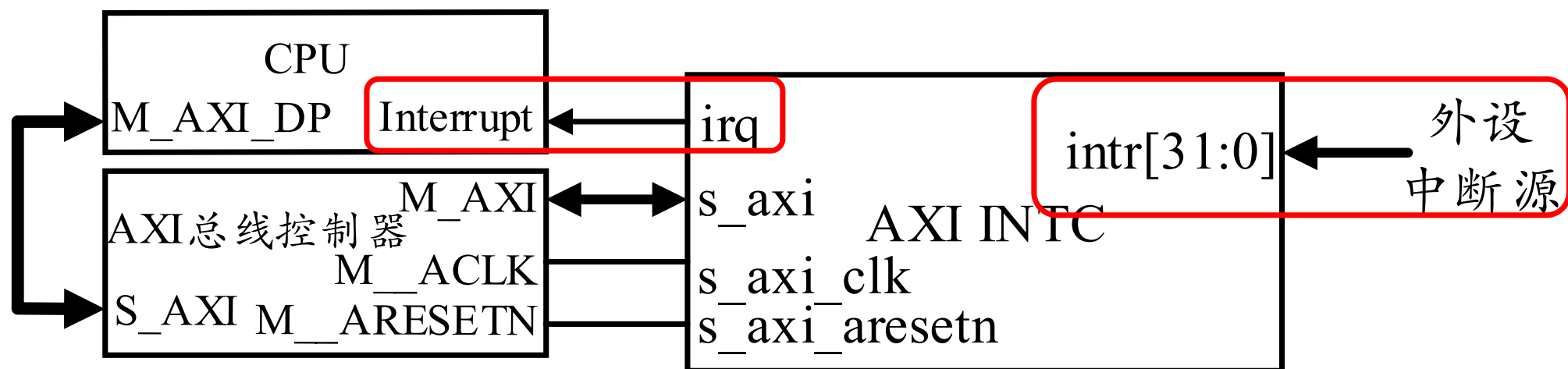
INTC中断信号产生时序——低电平有效



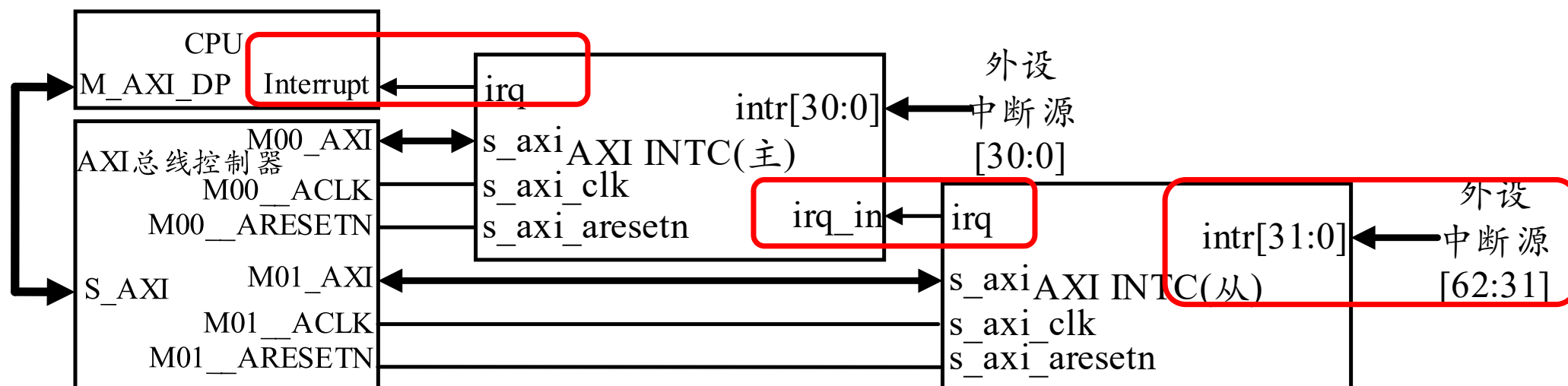
INTC中断信号产生时序——上升沿有效



INTC应用电路-单模块普通中断模式

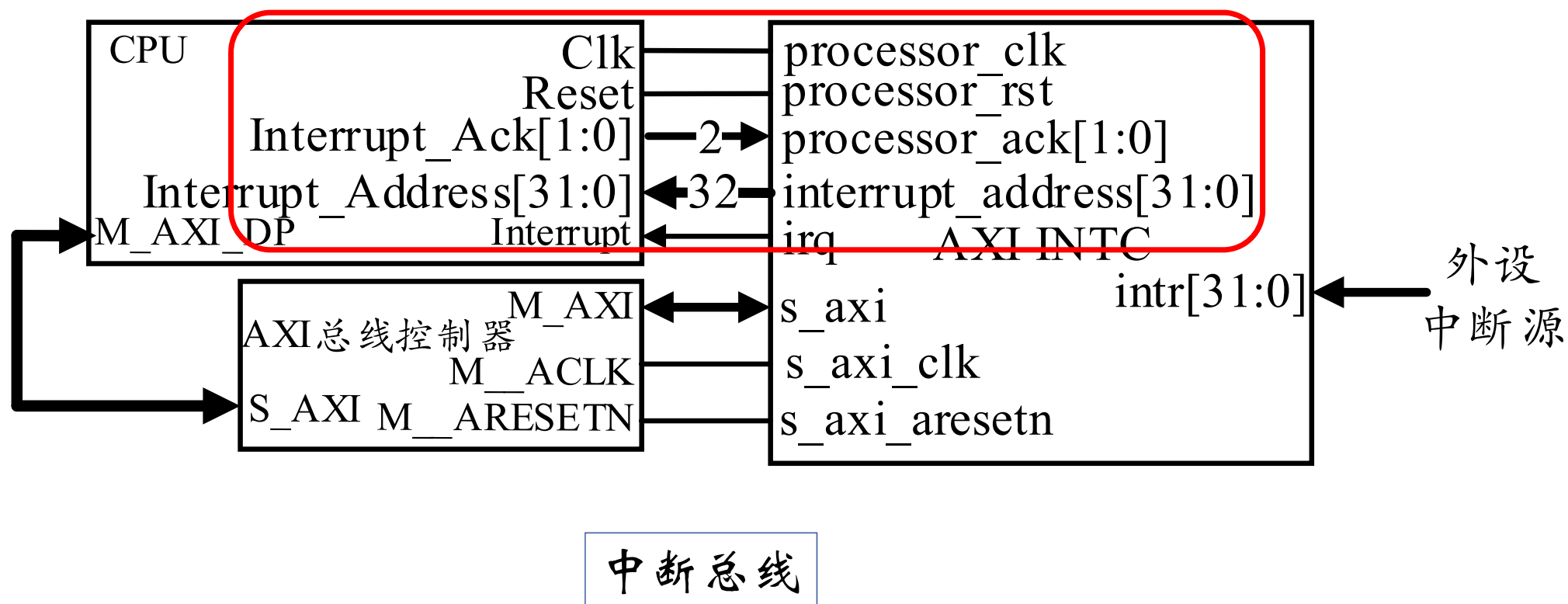


INTC应用电路-级联普通中断模式

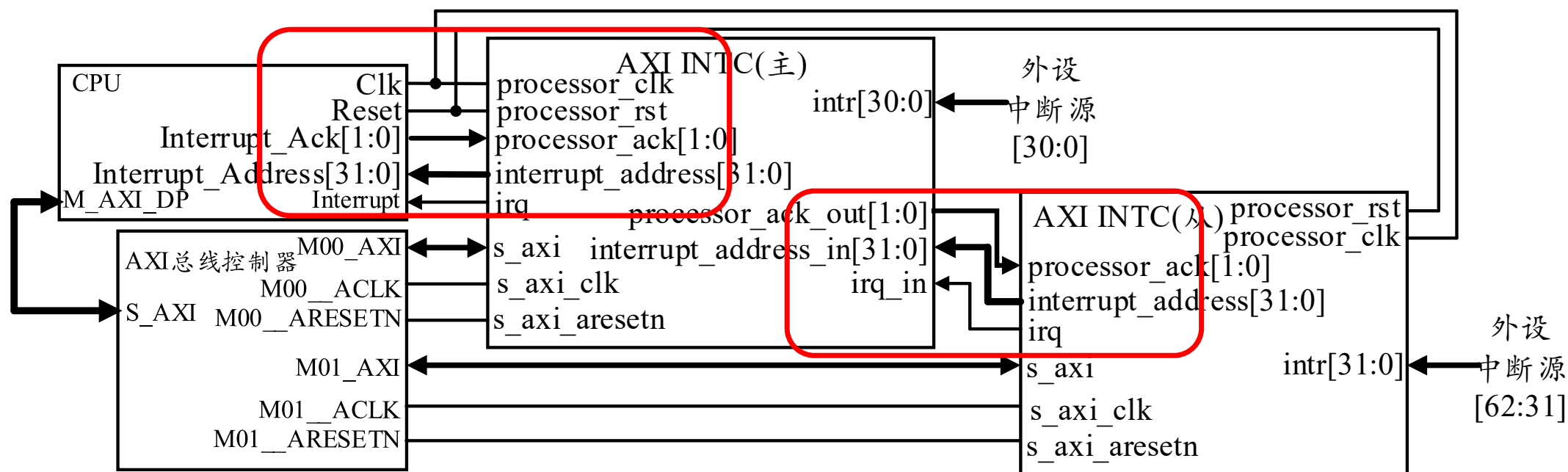


63个中断源

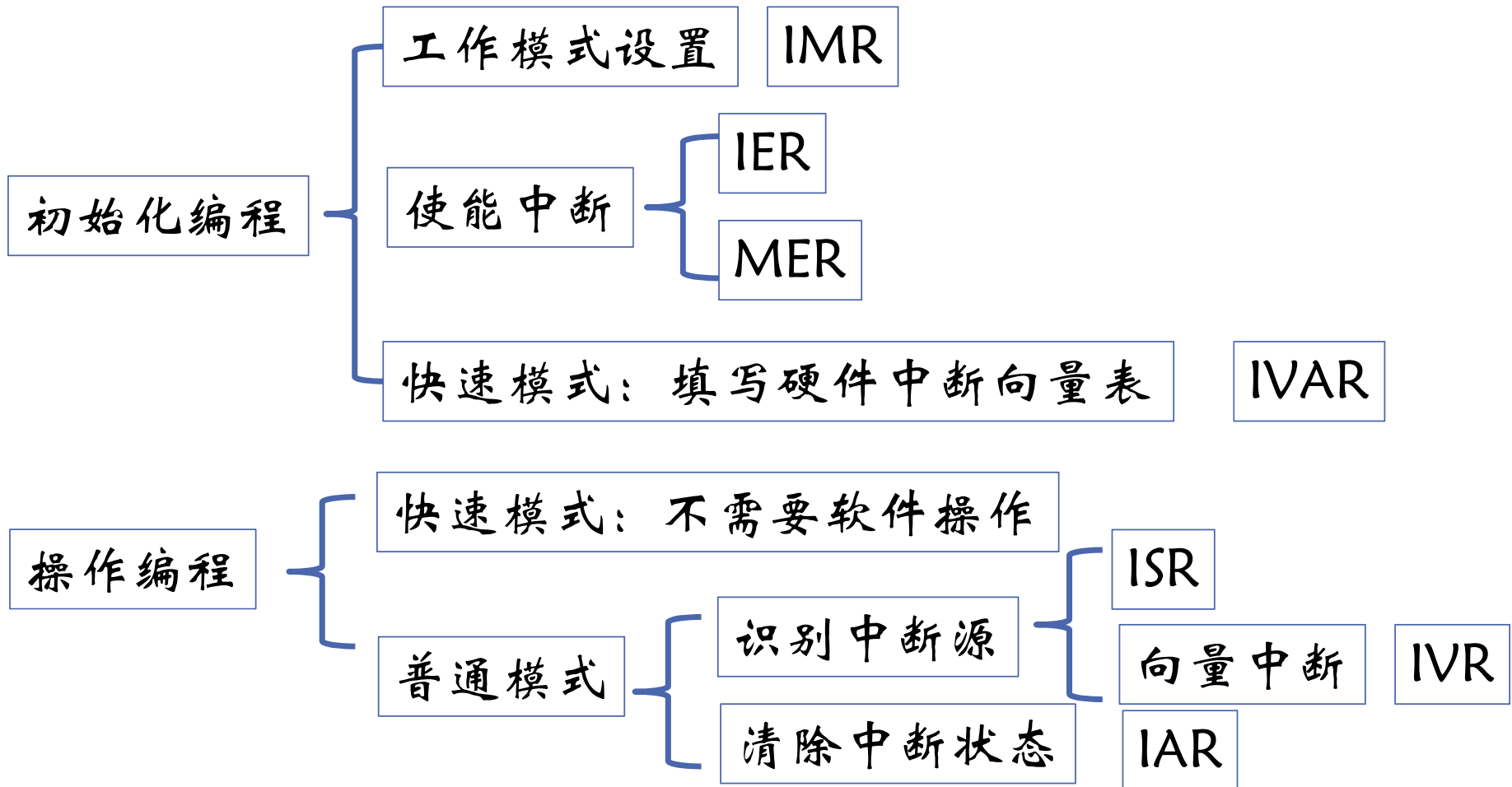
INTC应用电路-单模块快速中断模式



INTC应用电路-级联快速中断模式



INTC编程控制



单模块INTC编程示例

INTC基地址为0x41200000，若INTC中断请求输入引脚intr[1:0]连接了中断源，编写INTC普通中断模式初始化控制程序段

寄存器名称	偏移地址	含义
ISR	0x0	中断状态寄存器
IPR*	0x4	中断悬挂寄存器
IER	0x8	中断使能寄存器
IAR	0xC	中断响应寄存器
SIE*	0x10	中断使能设置寄存器
CIE*	0x14	中断使能清除寄存器
IVR*	0x18	中断类型码寄存器
MER	0x1C	主中断使能寄存器
IMR*	0x20	中断模式寄存器
ILR*	0x24	中断级别寄存器
IVAR*	0x100~0x170	中断向量表寄存器

```
Xil_Out32(0x41200020, 0x0);  
//普通中断模式  
Xil_Out32(0x41200008, 0x3);  
//使能intr[1:0]  
Xil_Out32(0x4120001c, 0x3);  
//使能硬件中断输出
```


单模块INTC编程示例

INTC基地址为0x41200000，若INTC中断请求输入引脚intr[1:0]连接了中断源，编写INTC快速中断模式初始化控制程序段

寄存器名称	偏移地址	含义
ISR	0x0	中断状态寄存器
IPR*	0x4	中断悬挂寄存器
IER	0x8	中断使能寄存器
IAR	0xC	中断响应寄存器
SIE*	0x10	中断使能设置寄存器
CIE*	0x14	中断使能清除寄存器
IVR*	0x18	中断类型码寄存器
MER	0x1C	主中断使能寄存器
IMR*	0x20	中断模式寄存器
ILR*	0x24	中断级别寄存器
IVAR*	0x100~0x170	中断向量表寄存器

```
Xil_Out32(0x41200020, 0x3);//
Xil_Out32(0x41200008, 0x3);//
Xil_Out32(0x4120001c, 0x3);//
Xil_Out32(0x41200100, ISR0); //
Xil_Out32(0x41200104, ISR1); //
```

小结

- INTC 结构
- 工作模式
- 寄存器功能
- 应用电路
- 初始化编程

下一讲：微处理器中断控制