1. 以下是LAB11\_main.c中与中断设置相关的程序语句：

1. DINT; //禁止 CPU中断，禁止全局中断

2. InitPieCtrl(); //初始化PIE控制寄存器

3. IER=0x0000; //禁用所有CPU中断并清除CPU中断标志位

4. IFR=0x0000;

5. InitPieVectTable(); //初始化PIE向量表 里面包含了 PieCtrlRegs.PIECTRL.bit.ENPIE=1

6. PieVectTable.EPWM1\_INT =&epwm1\_timer\_adc\_isr; //第三组第一中断

7. PieCtrlRegs.PIEIER3.bit.INTx1 = 1; // Enable ADCINT interrupt in PIE

8. PieCtrlRegs.PIECTRL.bit.ENPIE=1; //打开PIE中断,使能PIE

9. IER |= M\_INT3; //打开CPU第3组中断

10. EINT; //使能全局中断，允许中断响应

1. 函数InitPieCtrl()实现的功能

函数InitPieCtrl()的功能是将所有的 PIE 控制寄存器初始化为已知状态。它首先禁止了 CPU 级别的中断，并将所有的 PIEIER 和 PIEIFR 寄存器清零，以确保在启用 PIE 之前没有未处理的中断。处理完这些之后，它将 PieCtrlRegs.PIECTRL.bit.ENPIE 设置为0，禁用整个 PIE。

1. 函数InitPieVectTable()实现的功能

函数InitPieVectTable()的功能是将PIE向量表初始化为一个已知状态。

它首先使用两个指针达成了用PieVectTableInit数组的值初始化PieVectTable的目的，然后使能了PIE向量表（PieCtrlRegs.PIECTRL.bit.ENPIE = 1）。

1. 语句PieVectTable.EPWM1\_INT =&epwm1\_timer\_adc\_isr; 实现的功能。

这个语句将epwm1\_timer\_adc\_isr函数的地址赋值给EPWM1\_INT中断向量，即设置EPWM1\_INT中断服务程序的入口地址为epwm1\_timer\_adc\_isr，当EPWM1产生中断时，CPU会跳转到epwm1\_timer\_adc\_isr函数的地址处执行。这样就实现了将用户编写的中断服务函数与相应中断向量关联起来的功能。