作业三及答案

根据“Example\_ADC”中实验内容，打开程序LAB11\_main.c，DSP2833x\_PieCtrl.c以及相关头文件，阅读程序段落。

1. 摘录与中断设置相关的程序语句。

2. 函数InitPieCtrl()实现的功能。

3. 函数InitPieVectTable()实现的功能。

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

答案：

1.摘录与中断设置相关的程序语句。

DINT; //禁止 CPU中断，禁止全局中断

InitPieCtrl(); //初始化PIE控制寄存器

IER=0x0000; //禁用所有CPU中断并清除CPU中断标志位

IFR=0x0000;

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

EALLOW;

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

EDIS;

…………

EINT;

2. 函数InitPieCtrl()实现的功能。

DINT;

// Disable the PIE

PieCtrlRegs.PIECTRL.bit.ENPIE = 0;

// Clear all PIEIER registers:

PieCtrlRegs.PIEIER1.all = 0;

PieCtrlRegs.PIEIER2.all = 0;

……

// Clear all PIEIFR registers:

PieCtrlRegs.PIEIFR1.all = 0;

PieCtrlRegs.PIEIFR2.all = 0;

……

**禁止所有中断；**

**清除所有中断标志；**

3. 函数InitPieVectTable()实现的功能。

int16 i;

Uint32 \*Source = (void \*) &PieVectTableInit;

Uint32 \*Dest = (void \*) &PieVectTable;

EALLOW;

for(i=0; i < 128; i++)

\*Dest++ = \*Source++;

EDIS;

// Enable the PIE Vector Table

PieCtrlRegs.PIECTRL.bit.ENPIE = 1;

**中断向量表赋值；**

**所有的中断函数在DSP2833x\_DefaultIsr.c中有定义。**

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

**将用户自定义的中断函数首地址赋值给中断向量。**