

FM3316 中断应用说明

简介

FM3316 内核有 16 个中断源，含 1 个不可屏蔽的软件中断（TRAP）、1 个不可屏蔽的硬件中断（NMI）和 14 个可屏蔽的硬件中断。

FM3316 的每个外设有自己的中断控制使能与中断标志，外设的中断挂靠在内核的中断源上，所以外设使用时，不仅要打开外设自己的中断控制使能，同时要打开其挂靠内核的中断源使能

中断源与外设挂靠

内核中断源	内核中断描述	内核中断入口地址	内核使能位	优先级控制位	内核中断标志位	中断号	挂靠外设
TRAP	TRAP 指令中断	FF:007B	--	--	--	15	仿真使用
NMI	NMI 中断	FF:003B	--	--	--	7	停振检测 NWKUPx 引脚中断 LPRUN 时钟错误中断
Int_0	外部中断 0	FF:0003	IE.EX0	IPH0.IPHX0 IPL0.IPLX0	TCON.IE0	0	RTC、LTBC、LPTIM、LVD
T/C0	定时器 0 中断	FF:000B	IE.ET0	IPH0.IPHT0 IPL0.IPLT0	TCON.TF0	1	
Int_1	外部中断 1	FF:0013	IE.EX1	IPH0.IPHX1 IPL0.IPLX1	TCON.IE1	2	GPIO 引脚中断
T/C1	定时器 1 中断	FF:001B	IE.ET1	IPH0.IPHT1 IPL0.IPLT1	TCON.TF1	3	
Serial_Port	串口中断	FF:0023	IE.ES	IPH0.IPHS IPL0.IPLS	SCON.TI SCON.RI	4	
T/C2	定时器 2 中断	FF:002B	IE.ET2	IPH0.IPHT2 IPL0.IPLT2	T2CON.TF2 T2CON.EXF2	5	
PCA	PCA 中断	FF:0033	IE.EC	IPH0.IPHC IPL0.IPLC	CCON.CF CCON.CCFx	6	PCA
Intextra_n0	附加中断 0	FF:0043	AIE.AIE0	AIPH.AIPH0 AIPL.AIPL0	AIF.AIF0	8	ET1/2 ET3/4
Intextra_n1	附加中断 1	FF:004B	AIE.AIE1	AIPH.AIPH1 AIPL.AIPL1	AIF.AIF1	9	ADC
Intextra	附加中断 2	FF:0053	AIE.AIE2	AIPH.AIPH2	AIF.AIF2	10	UART0/1/2/3

内核中断源	内核中断描述	内核中断入口地址	内核使能位	优先级控制位	内核中断标志位	中断号	挂靠外设
_n2				AIPL.AIPL2			
Intextra_n3	附加中断 3	FF:005B	AIE.AIE3	AIPH.AIPH3 AIPL.AIPL3	AIF.AIF3	11	SPI /I2C/7816
Intextra_n4	附加中断 4	FF:0063	AIE.AIE4	AIPH.AIPH4 AIPL.AIPL4	AIF.AIF4	12	DISP
Intextra_n5	附加中断 5	FF:006B	AIE.AIE5	AIPH.AIPH5 AIPL.AIPL5	AIF.AIF5	13	Flash 控制器
Intextra_n6	附加中断 6	FF:0073	AIE.AIE6	AIPH.AIPH6 AIPL.AIPL6	AIF.AIF6	14	DMA

中断优先级

优先级高低位组合可产生四级优先级，以 PCA 中断为例，优先级定义见表示。其它中断源类似。

IPHC	IPLC	中断优先级	中断优先级说明
0	0	0	最低优先级
0	1	1	次低优先级
1	0	2	次高优先级
1	1	3	最高优先级

中断应用编程注意事项（及其重要）

1、中断例程

例如：NMI 中断

```
void nmi_int (void) interrupt 7 using 2
```

```
{ // 用户代码...
```

```
    NMIFLAG = 0;
```

```
}
```

(注：nmi_in 是中断函数名称，7 是中断号，2 是寄存器组)

2、如果在中断服务函数 中断中使用寄存器，那么必须处理好 using 的使用问题：

1、中断服务函数使用 using 指定与主函数不同的寄存器组（主函数一般使用 Register bank 0）。

2、中断优先级相同的中断 可用 **using** 指定相同的寄存器组,但优先级不同的 中断 必须使用不同的寄存器组

3、函数同时被中断函数与 **main** 函数也调用, 则此函数必须申明为可重入函数

VOID func(para..) reentrant;

其实还有一种办法: 如果被调用函数的代码不是很长, 还是将该函数复制一份, 用不同的函数名代替, 这种情况适合 **ROM** 有足够多余的空间。