

39	INT_UART2	UART2 中断	VIC1
38	INT_UART1	UART1 中断	VIC1
37	INT_UART0	UART0 中断	VIC1
36	INT_AC97	AC 中断	VIC1
35	INT_PCM1	PCM1 中断	VIC1
34	INT_PCM0	PCM0 中断	VIC1
33	INT_EINT3	外部中断 20~27	VIC1
32	INT_EINT2	外部中断 12~19	VIC1
31	INT_LCD[2]	LCD 中断. 系统 I/F 完成	VIC0
30	INT_LCD[1]	LCD 中断. VSYNC 中断	VIC0
29	INT_LCD[0]	LCD 中断. FIFO 不足	VIC0
28	INT_TIMER4	定时器 4 中断.	VIC0
27	INT_TIMER3	定时器 3 中断.	VIC0
26	INT_WDT	看门狗定时器中断.	VIC0
25	INT_TIMER2	定时器 2 中断.	VIC0
24	INT_TIMER1	定时器 1 中断.	VIC0
23	INT_TIMER0	定时器 0 中断.	VIC0
22	INT_KEYPAD	键盘中断.	VIC0
21	INT_ARM_DMAS	ARM DMAS 中断.	VIC0
20	INT_ARM_DMA	ARM DMA 中断.	VIC0
19	INT_ARM_DMAERR	ARM DMA 错误中断.	VIC0
18	INT_SDMA1	安全 DMA1 中断.	VIC0
17	INT_SDMA0	安全 DMA0 中断.	VIC0
16	INT_MFC	MFC 中断.	VIC0
15	INT_JPEG	JPEG 中断.	VIC0
14	INT_BATF	电池故障中断.	VIC0
13	INT_SCALER	TV 转换器中断.	VIC0
12	INT_TVENC	TV 编码器中断.	VIC0

11	INT_ 2D	2D 中断.	VICO
10	INT_ ROTATOR	旋转器中断.	VICO
9	INT_POSTO	后处理器中断.	VICO
8	INT_ 3D	3D 图像控制器中断.	VICO
7	Reserved	保留	VICO
6	INT_ I2S0 \INT_ I2S1 \INT_ I2SV40	I2S0 中断或 I2S1 中断或 I2SV40 中断	VICO
5	INT_ I2C1	I2C1 中断	VICO
4	INT_ CAMIF_P	照相机接口中断	VICO
3	INT_ CAMIF_C	照相机接口中断	VICO
2	INT_ RTC_TIC	RTC TIC 中断	VICO
1	INT_ EINT1	外部中断 4~11	VICO
0	INT_ EINT0	外部中断 0~3	VICO

12.3 矢量中断控制器功能模块图

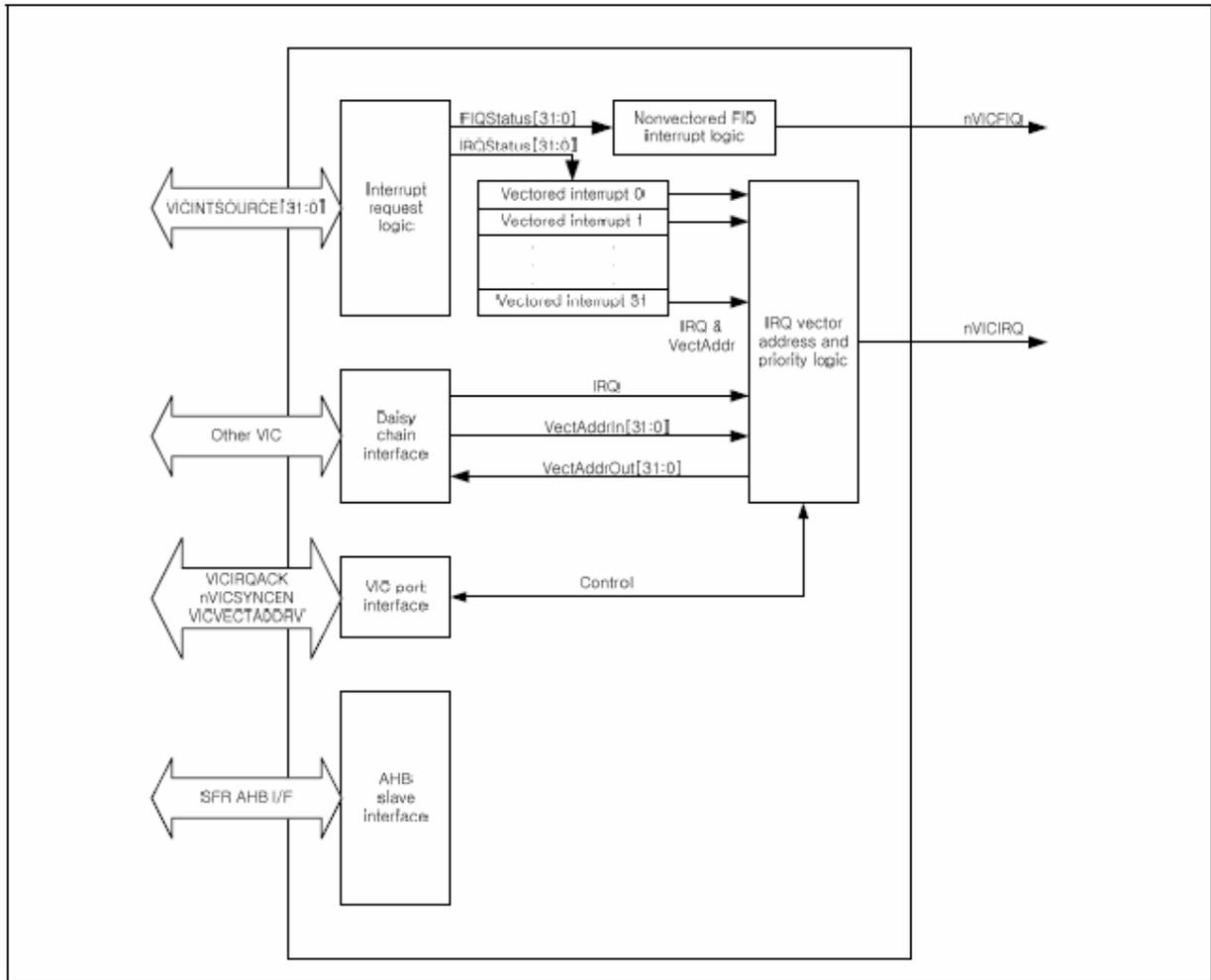


图 12-2 矢量中断控制器模块图

12.4 矢量中断控制器摘要

VIC0 的基础地址是 0x7120_0000

VIC1 的基础地址是 0x7130_0000

控制寄存器地址=基础地址+补偿区

寄存器	补偿区	类型	描述	复位值
VICxIRQSTATUS	0x000	读	IRQ 状态寄存器	0x00000000
VICxFIQSTATUS	0x004	读	FIQ 状态寄存器	0x00000000
VICxIRAWINTR	0x008	读	原始中断状态寄存器	0x00000000
VICxINTSELECT	0x00C	读写	中断选择寄存器	0x00000000
VICxINTENABLE	0x010	读写	中断使能寄存器	0x00000000
VICxINTENCLEAR	0x014	写	中断使能清除寄存器	-
VICxSOFTINT	0x018	读写	软件中断寄存器	0x00000000
VICxISOFTINTCLEAR	0x01C	写	软件中断清除寄存器	-
VICxPROTECTION	0x020	读写	保护使能寄存器	0x0
VICxSWPRIORITYMASK	0x024	读写	软件优先屏蔽寄存器	0x0FFFF
VICxPRIORITYDAISY	0x028	读写	菊花链的矢量优先寄存器	0xF
VICxVECTADDR0	0x100	读写	矢量地址 0 寄存器	0x00000000
VICxVECTADDR1	0x104	读写	矢量地址 1 寄存器	0x00000000
VICxVECTADDR2	0x108	读写	矢量地址 2 寄存器	0x00000000
VICxVECTADDR3	0x10C	读写	矢量地址 3 寄存器	0x00000000
VICxVECTADDR4	0x110	读写	矢量地址 4 寄存器	0x00000000
VICxVECTADDR5	0x114	读写	矢量地址 5 寄存器	0x00000000
VICxVECTADDR6	0x118	读写	矢量地址 6 寄存器	0x00000000
VICxVECTADDR7	0x11C	读写	矢量地址 7 寄存器	0x00000000
VICxVECTADDR8	0x120	读写	矢量地址 8 寄存器	0x00000000
VICxVECTADDR9	0x124	读写	矢量地址 9 寄存器	0x00000000
VICxVECTADDR10	0x128	读写	矢量地址 10 寄存器	0x00000000
VICxVECTADDR11	0x12C	读写	矢量地址 11 寄存器	0x00000000
VICxVECTADDR12	0x130	读写	矢量地址 12 寄存器	0x00000000
VICxVECTADDR13	0x134	读写	矢量地址 13 寄存器	0x00000000
VICxVECTADDR14	0x138	读写	矢量地址 14 寄存器	0x00000000
VICxVECTADDR15	0x13C	读写	矢量地址 15 寄存器	0x00000000

VICxVECTADDR16	0x140	读写	矢量地址 16 寄存器	0x00000000
VICxVECTADDR17	0x144	读写	矢量地址 17 寄存器	0x00000000
VICxVECTADDR18	0x1408	读写	矢量地址 18 寄存器	0x00000000
VICxVECTADDR19	0x14C	读写	矢量地址 19 寄存器	0x00000000
VICxVECTADDR20	0x150	读写	矢量地址 20 寄存器	0x00000000
VICxVECTADDR21	0x154	读写	矢量地址 21 寄存器	0x00000000
VICxVECTADDR22	0x158	读写	矢量地址 22 寄存器	0x00000000
VICxVECTADDR23	0x15C	读写	矢量地址 23 寄存器	0x00000000
VICxVECTADDR24	0x160	读写	矢量地址 24 寄存器	0x00000000
VICxVECTADDR25	0x164	读写	矢量地址 25 寄存器	0x00000000
VICxVECTADDR26	0x168	读写	矢量地址 26 寄存器	0x00000000
VICxVECTADDR27	0x16C	读写	矢量地址 27 寄存器	0x00000000
VICxVECTADDR28	0x170	读写	矢量地址 28 寄存器	0x00000000
VICxVECTADDR29	0x174	读写	矢量地址 29 寄存器	0x00000000
VICxVECTADDR30	0x178	读写	矢量地址 30 寄存器	0x00000000
VICxVECTADDR31	0x17C	读写	矢量地址 31 寄存器	0x00000000
VICxVECTPRIORITY0	0x200	读写	矢量优先 0 寄存器	0xF
VICxVECTPRIORITY1	0x204	读写	矢量优先 1 寄存器	0xF
VICxVECTPRIORITY2	0x208	读写	矢量优先 2 寄存器	0xF
VICxVECTPRIORITY3	0x20C	读写	矢量优先 3 寄存器	0xF
VICxVECTPRIORITY4	0x210	读写	矢量优先 4 寄存器	0xF
VICxVECTPRIORITY5	0x214	读写	矢量优先 5 寄存器	0xF
VICxVECTPRIORITY6	0x218	读写	矢量优先 6 寄存器	0xF
VICxVECTPRIORITY7	0x21C	读写	矢量优先 7 寄存器	0xF
VICxVECTPRIORITY8	0x220	读写	矢量优先 8 寄存器	0xF
VICxVECTPRIORITY9	0x224	读写	矢量优先 9 寄存器	0xF
VICxVECTPRIORITY10	0x228	读写	矢量优先 10 寄存器	0xF
VICxVECTPRIORITY11	0x22C	读写	矢量优先 11 寄存器	0xF

VICxVECTPRIORITY12	0x230	读写	矢量优先 12 寄存器	0xF
VICxVECTPRIORITY13	0x234	读写	矢量优先 13 寄存器	0xF
VICxVECTPRIORITY14	0x238	读写	矢量优先 14 寄存器	0xF
VICxVECTPRIORITY15	0x23C	读写	矢量优先 15 寄存器	0xF
VICxVECTPRIORITY16	0x240	读写	矢量优先 16 寄存器	0xF
VICxVECTPRIORITY17	0x244	读写	矢量优先 17 寄存器	0xF
VICxVECTPRIORITY18	0x2408	读写	矢量优先 18 寄存器	0xF
VICxVECTPRIORITY19	0x24C	读写	矢量优先 19 寄存器	0xF
VICxVECTPRIORITY20	0x250	读写	矢量优先 20 寄存器	0xF
VICxVECTPRIORITY21	0x254	读写	矢量优先 21 寄存器	0xF
VICxVECTPRIORITY22	0x258	读写	矢量优先 22 寄存器	0xF
VICxVECTPRIORITY23	0x25C	读写	矢量优先 23 寄存器	0xF
VICxVECTPRIORITY24	0x260	读写	矢量优先 24 寄存器	0xF
VICxVECTPRIORITY25	0x264	读写	矢量优先 25 寄存器	0xF
VICxVECTPRIORITY26	0x268	读写	矢量优先 26 寄存器	0xF
VICxVECTPRIORITY27	0x26C	读写	矢量优先 27 寄存器	0xF
VICxVECTPRIORITY28	0x270	读写	矢量优先 28 寄存器	0xF
VICxVECTPRIORITY29	0x274	读写	矢量优先 29 寄存器	0xF
VICxVECTPRIORITY30	0x278	读写	矢量优先 30 寄存器	0xF
VICxVECTPRIORITY31	0x27C	读写	矢量优先 31 寄存器	0xF
VICxADDRESS	0xf00	读写	矢量地址寄存器	0x00000000

12.5 寄存器描述

12.5.1. IRQ 状态寄存器，VICIRQSTATUS

寄存器	地址	读/写	描述	复位值
-----	----	-----	----	-----

VIC0IRQSTATUS	0x7120_0000	读	IRQ 状态寄存器(VIC0)	0x0000_0000
VIC1IRQSTATUS	0x7130_0000	读	IRQ 状态寄存器(VIC1)	0x0000_0000

名称	位	描述	复位值
IRQStatus	[31:0]	在屏蔽之后,通过 VICxINTENABLE 和 VICxINTSELECT 寄存器显示中断的状态 0=中断不被激活(复位) 1=中断被激活 每个中断源都有一个寄存器位	0x0

12.5.2.FIQ 状态寄存器, VICFIQSTATUS

寄存器	地址	读/写	描述	复位值
VIC0FIQSTATUS	0x7120_0004	读	FIQ 状态寄存器(VIC0)	0x0000_0000
VIC1FIQSTATUS	0x7130_0004	读	FIQ 状态寄存器(VIC1)	0x0000_0000

名称	位	描述	复位值
FIQStatus	[31:0]	在屏蔽之后,通过 VICINTENABLE 和 VICINTSELECT 寄存器显示 FIQ 中断的状态 0=中断不被激活(复位) 1=中断被激活 每个中断源都有一个寄存器位	0x0

12.5.3.原始中断状态寄存器, VICRAWINTR

寄存器	地址	读/写	描述	复位值
VIC0RAWINTR	0x7120_0008	读	原始中断状态寄存器(VIC0)	0x0000_0000

VIC1RAWINTR	0x7130_0008	读	原始中断状态寄存器(VIC1)	0x0000_0000
-------------	-------------	---	-----------------	-------------

名称	位	描述	复位值
RawInterrupt	[31:0]	<p>在屏蔽之前，通过 VICINTENABLE 和 VICINTSELECT 寄存器显示 FIQ 中断的状态</p> <p>0=屏蔽之前中断不被激活（复位）</p> <p>1=屏蔽之前中断被激活</p> <p>因为原始寄存器可以直接看到原始的输入中断，因此不知道复位值。</p> <p>每个中断源都有一个寄存器位。</p>	0x0

12.5.4.中断选择寄存器, VICINTSELET

寄存器	地址	读/写	描述	复位值
VIC0INTSELECT	0x7120_000C	读/写	中断选择寄存器(VIC0)	0x0000_0000
VIC1INTSELECT	0x7130_000C	读/写	中断选择寄存器(VIC1)	0x0000_0000

名称	位	描述	复位值
IntSelect	[31:0]	<p>为中断请求选择中断的状态</p> <p>0=IRQ 中断（复位）</p> <p>1=FIQ 中断</p> <p>每个中断源都有一个寄存器位。</p>	0x0

12.5.5.中断使能寄存器, VICINTENABLE

寄存器	地址	读/写	描述	复位值
VIC0INTENABLE	0x7120_0010	读/写	中断使能寄存器(VIC0)	0x0000_0000
VIC1INTENABLE	0x7130_0010	读/写	中断使能寄存器(VIC1)	0x0000_0000

名称	位	描述	复位值
IntEnable	[31:0]	<p>使能中断请求，允许中断到达处理器</p> <p>读：</p> <p>0=中断禁止（复位）</p> <p>1=中断使能</p> <p>中断使能只能用寄存器设置。VICINTENCLEAR 寄存器用来清除中断使能。</p> <p>写：</p> <p>0=没有影响</p> <p>1=中断使能</p> <p>每个中断源都有一个寄存器位。</p>	0x0

12.5.6.中断使能清除，VICINTENCLEAR

寄存器	地址	读/写	描述	复位值
VIC0INTENCLEAR	0x7120_0014	写	中断使能清除寄存器(VIC0)	-
VIC1INTENCLEAR	0x7130_0014	写	中断使能清除寄存器(VIC1)	-

名称	位	描述	复位值
IntEnable Clear	[31:0]	<p>在 VICINTENABLE 寄存器内清除相应的位</p> <p>0=没有影响（复位）</p> <p>1=在 VICINTENABLE 寄存器内中断 disabled</p> <p>每个中断源都有一个寄存器位。</p>	-

12.5.7.软件中断寄存器，VICSOFTINT

寄存器	地址	读/写	描述	复位值
VIC0SOFTINT	0x7120_0018	读/写	软件中断寄存器(VIC0)	0x0000_0000
VIC1SOFTINT	0x7130_0018	读/写	软件中断寄存器(VIC1)	0x0000_0000

名称	位	描述	复位值
IntEnable	[31:0]	<p>在中断屏蔽之前设置 HIGH 位对选择的源产生软件中断</p> <p>读:</p> <p>0=软件中断不被激活(复位)</p> <p>1=软件中断被激活</p> <p>写:</p> <p>0=没有影响</p> <p>1=软件中断使能</p> <p>每个中断源都有一个寄存器位。</p>	0x0

12.5.8.软件中断清除寄存器，VICSOFTINTCLEAR

寄存器	地址	读/写	描述	复位值
VIC0SOFTINTENCLEAR	0x7120_001C	写	软件中断清除寄存器(VIC0)	-
VIC1SOFTINTENCLEAR	0x7130_001C	写	软件中断清除寄存器(VIC1)	-

名称	位	描述	复位值
SoftInt Clear	[31:0]	<p>在 VICSOFTINT 寄存器内清除相应的位</p> <p>0=没有影响(复位)</p> <p>1=在 VICSOFTINT 寄存器内中断 disabled</p>	-