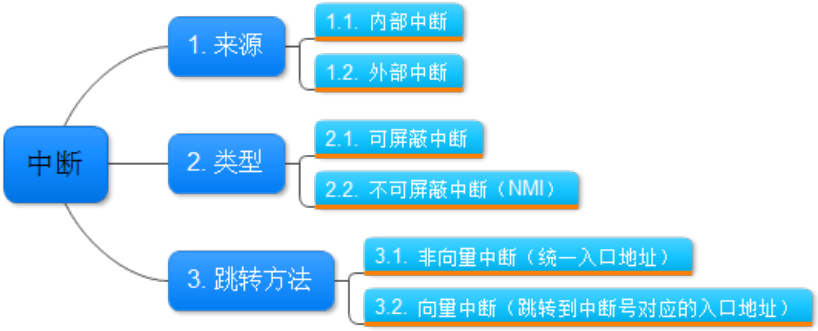
专题12-通过中断实现按键功能

一、中断处理流程深度剖析

1.1、中断概念

中断是指CPU执行程序的过程中，出现了突发事件急待处理，CPU必须暂停执行当前的程序，转去处理突发事件。



1.2、中断源

12.3 INTERRUPT SOURCES

The S3C6410X supports 64 interrupt sources as shown in the table below.

The S3C6410X can not support performance monitoring interrupt of the ARM1176JZF-S.

12.3 INTERRUPT SOURCES

The S3C6410X supports 64 interrupt sources as shown in the table below.

The S3C6410X can not support performance monitoring interrupt of the ARM1176JZF-S.

Int. No.	Sources	Description	Group
63	INT_ADC	ADC EOC interrupt	VIC1
62	INT_PENDNUP	ADC Pen down/up interrupt	VIC1
61	INT_SEC	Security interrupt	VIC1
60	INT_RTC_ALARM	RTC alarm interrupt	VIC1
59	INT_IrDA	IrDA interrupt	VIC1
58	INT_OTG	USB OTG interrupt	VIC1
49	INT_SPI1/INT_HSMMC2	SPI1 interrupt or HSMMC2 interrupt	VIC1
48	INT_SPI0	SPI0 interrupt	VIC1
47	INT_UHOST	USB Host interrupt	VIC1
46	INT_CFC	CFCON interrupt	VIC1
45	INT_NFC	NFCON interrupt	VIC1
44	INT_ONENAND1	OneNAND interrupt from bank 1	VIC1
43	INT_ONENAND0	OneNAND interrupt from bank 0	VIC1
42	INT_DMA1	DMA1 interrupt	VIC1

Int. No.	Sources	Description	Group
41	INT_DMA0	DMA0 interrupt	VIC1
40	INT_UART3	UART3 interrupt	VIC1
39	INT_UART2	UART2 interrupt	VIC1
38	INT_UART1	UART1 interrupt	VIC1
37	INT_UART0	UART0 interrupt	VIC1
36	INT_AC97	AC97 interrupt	VIC1

35	INT_PCM1	PCM1 interrupt	VIC1
34	INT_PCM0	PCM0 interrupt	VIC1
33	INT_EINT3	External interrupt 20 ~ 27	VIC1
32	INT_EINT2	External interrupt 12 ~ 19	VIC1
31	INT_LCD[2]	LCD interrupt. System I/F done	VIC0
30	INT_LCD[1]	LCD interrupt. VSYNC interrupt	VIC0
29	INT_LCD[0]	LCD interrupt. FIFO underrun	VIC0
28	INT_TIMER4	Timer 4 interrupt	VIC0
27	INT_TIMER3	Timer 3 interrupt	VIC0
26	INT_WDT	Watchdog timer interrupt	VIC0
25	INT_TIMER2	Timer 2 interrupt	VIC0
24	INT_TIMER1	Timer 1 interrupt	VIC0
23	INT_TIMER0	Timer 0 interrupt	VIC0
22	INT_KEYPAD	Keypad interrupt	VIC0
21	INT_ARM_DMAS	ARM DMAS interrupt	VIC0
20	INT_ARM_DMA	ARM DMA interrupt	VIC0
19	INT_ARM_DMAERR	ARM DMA Error interrupt	VIC0
18	INT_SDMA1	Secure DMA1 interrupt	VIC0
17	INT_SDMA0	Secure DMA0 interrupt	VIC0
16	INT_MFC	MFC interrupt	VIC0
15	INT_JPEG	JPEG interrupt	VIC0
14	INT_BATF	Battery fault interrupt	VIC0
13	INT_SCALER	TV Scaler interrupt	VIC0
12	INT_TVENC	TV Encoder interrupt	VIC0
11	INT_2D	2D interrupt	VIC0
10	INT_ROTATOR	Rotator interrupt	VIC0
9	INT_POST0	Post processor interrupt	VIC0
8	INT_3D	3D Graphic Controller interrupt	VIC0
7	Reserved	Reserved	VIC0

Int. No.	Sources	Description	Group
6	INT_I2S0 | INT_I2S1 | INT_I2SV40	I2S 0 interrupt or I2S 1 interrupt or I2S V40 interrupt	VIC0
5	INT_I2C1	I2C 1 interrupt	VIC0
4	INT_CAMIF_P	Camera interface interrupt	VIC0
3	INT_CAMIF_C	Camera interface interrupt	VIC0
2	INT_RTC_TIC	RTC TIC interrupt	VIC0
1	INT_EINT1	External interrupt 4 ~ 11	VIC0
0	INT_EINT0	External interrupt 0 ~ 3	VIC0

Note: 1. 6410X PoP A Type doesn't support "INT_NFC"

2. 6410X PoP D Type doesn't support "INT_ONENAND0" & "INT_ONENAND1"

1.3、中断生命周期

1.4、3SC6410中断控制流程

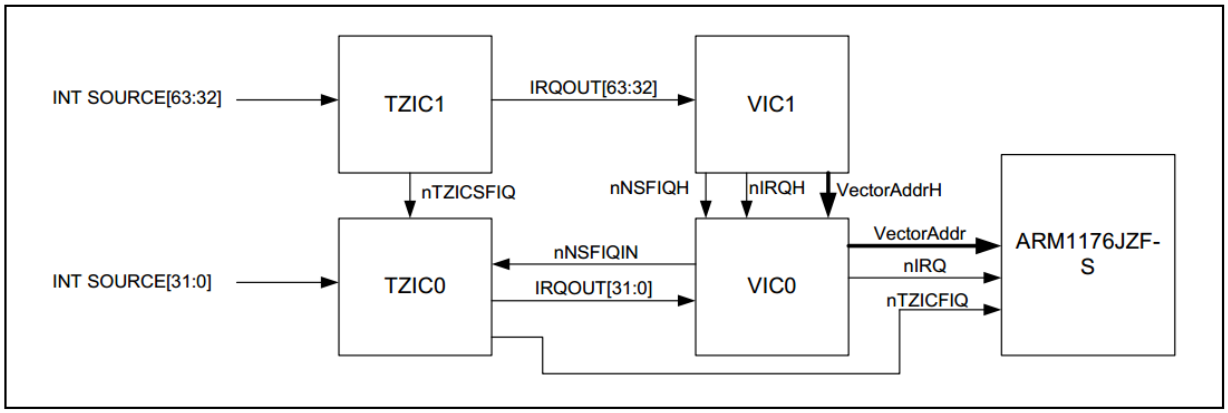


Figure 12-1. Interrupt Controller in S3C6410X

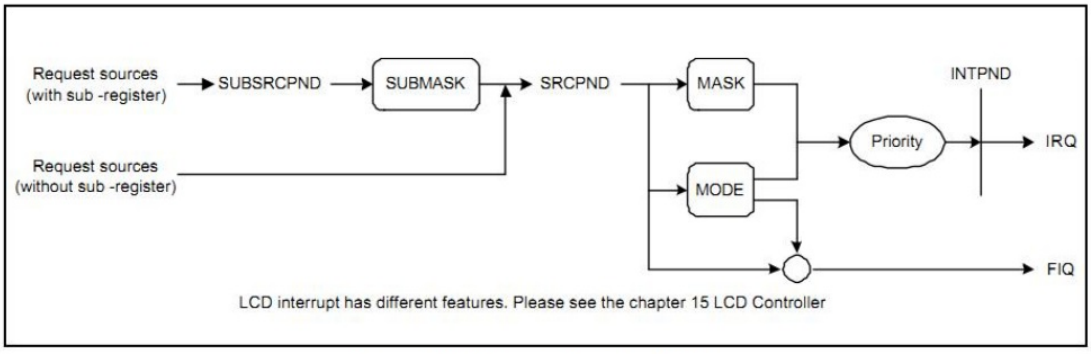
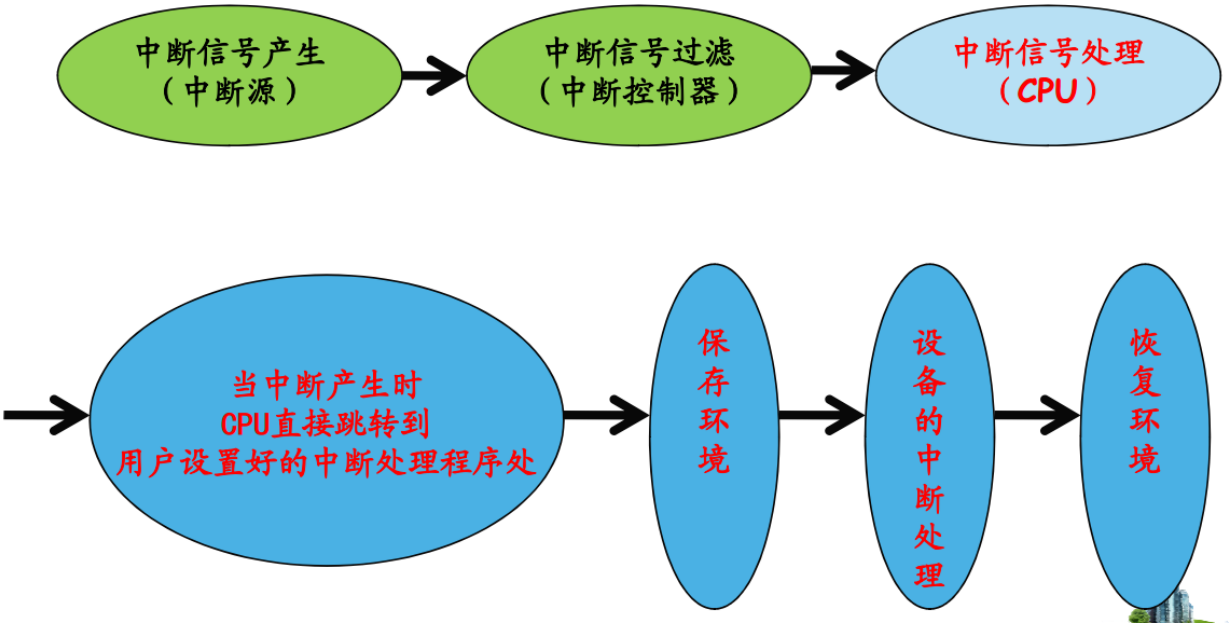


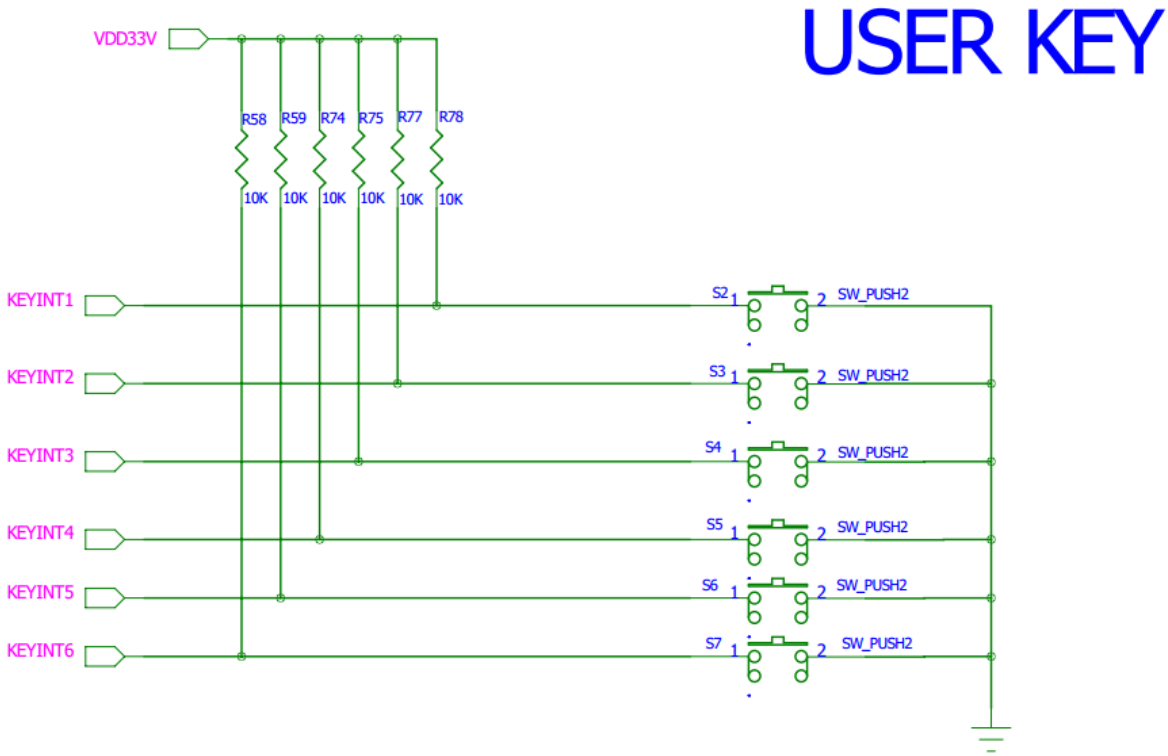
Figure 14-1. Interrupt Process Diagram

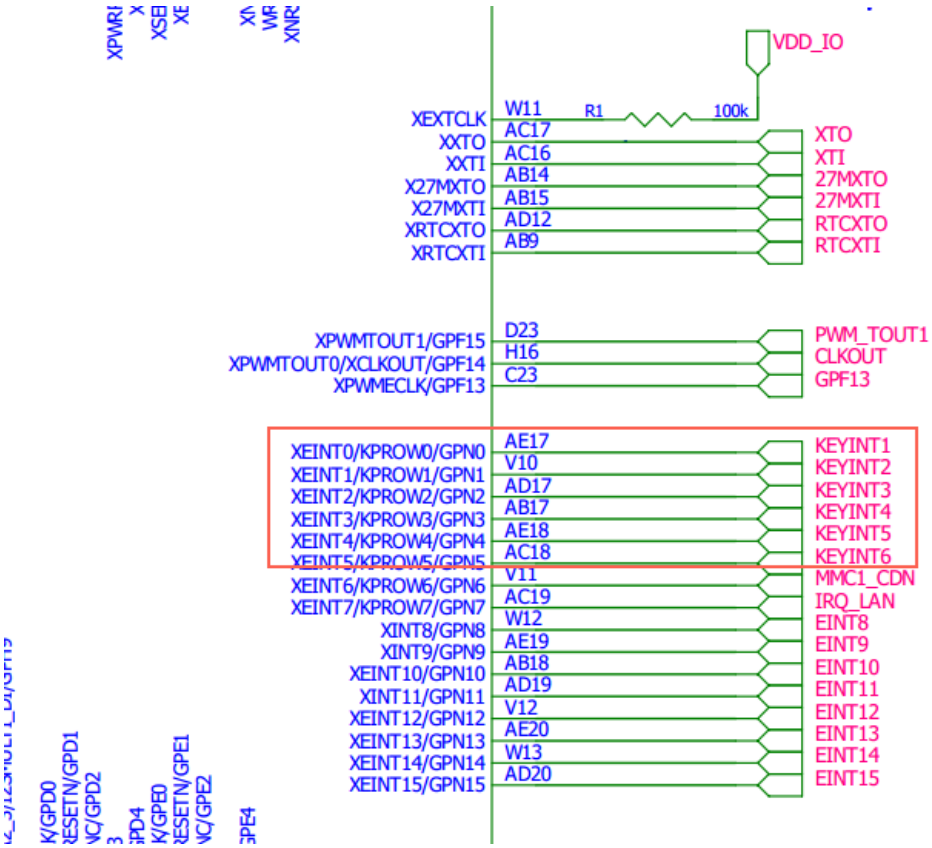


二、实现6410按键中断

2.1、初始化按键

2.1.1、按键原理图





2.1.2、配置GPIO寄存器

10.5.14 PORT N CONTROL REGISTERS

There are three control registers including GPNCON, GPNDAT and GPNPUD in the Port N Control Registers. GPNCON, GPNDAT and GPNPUD are alive part.

Register	Address	R/W	Description	Reset Value
GPNCON	0x7F008830	R/W	Port N Configuration Register	0x00
GPNDAT	0x7F008834	R/W	Port N Data Register	Undefined
GPNPUD	0x7F008838	R/W	Port N Pull-up/down Register	0x55555555

GPNCON	Bit	Description	Initial State
GPN0	[1:0]	00 = Input 01 = Output 10 = Ext. Interrupt[0] 11 = Key pad ROW[0]	00
GPN1	[3:2]	00 = Input 01 = Output 10 = Ext. Interrupt[1] 11 = Key pad ROW[1]	00
GPN2	[5:4]	00 = Input 01 = Output 10 = Ext. Interrupt[2] 11 = Key pad ROW[2]	00
GPN3	[7:6]	00 = Input 01 = Output 10 = Ext. Interrupt[3] 11 = Key pad ROW[3]	00
GPN4	[9:8]	00 = Input 01 = Output 10 = Ext. Interrupt[4] 11 = Key pad ROW[4]	00
GPN5	[11:10]	00 = Input 01 = Output 10 = Ext. Interrupt[5] 11 = Key pad ROW[5]	00
GPN6	[13:12]	00 = Input 01 = Output 10 = Ext. Interrupt[6] 11 = Key pad ROW[6]	00
GPN7	[15:14]	00 = Input 01 = Output 10 = Ext. Interrupt[7] 11 = Key pad ROW[7]	00
GPN8	[17:16]	00 = Input 01 = Output 10 = Ext. Interrupt[8] 11 = Reserved	00

GPNDAT	Bit	Description
GPN[15:0]	[15:0]	When the port is configured as input port, the corresponding bit is the pin state. When the port is configured as output port, the pin state is the same as the corresponding bit. When the port is configured as functional pin, the undefined value will be read.

GPNPUD	Bit	Description
GPN[n]	[2n+1:2n] n = 0~15	00 = pull-up/down disabled 01 = pull-down enabled 10 = pull-up enabled 11 = Reserved.

设置按键2和按键3为外部中断引脚。

```
1 #define GPNCON  ((volatile unsigned long *)0x7f008830)
2 #define GPNPUD  ((volatile unsigned long *)0x7f008838)
3
4 void button_init_ext_int(void)
5 {
6       *(GPNCON) = ((0b10) | (0b10 << 2));  //set key1 and key6
7 }
```

2.2、初始化中断控制器

10.5.22 EXTERNAL INTERRUPT CONTROL REGISTERS

External Interrupt is consists of 10 groups numbered from 0 to 9. Only external interrupt group 0 is used for wake-up source in Stop and Sleep mode. And, In idle mode, all interrupts can be wake-up source, the other groups of external interrupts also can be the sources.

The following table is the list of external interrupt control registers. Group 0 has dedicated pins and each interrupt in this group can be controlled more detail than the other groups. For that, S3C6410 presents several registers for group 0 and they have bits for a pair of interrupt signals. In case of the other groups, there are several registers for these groups, but one register can control 2 or more groups. Digital filter count source of group 0 is FIN(Ext.Clock), and other group digital filter count is PCLK.

Register	Address	R/W	Description	Reset Value
EINT0CON0	0x7F008900	R/W	External Interrupt 0(Group0) Configuration Register 0	0x0
EINT0CON1	0x7F008904	R/W	External Interrupt 0(Group0) Configuration Register 1	0x0
EINT0FLTCON0	0x7F008910	R/W	External Interrupt 0(Group0) Filter Control Register 0	0x0
EINT0FLTCON1	0x7F008914	R/W	External Interrupt 0(Group0) Filter Control Register 1	0x0
EINT0FLTCON2	0x7F008918	R/W	External Interrupt 0(Group0) Filter Control Register 2	0x0
EINT0FLTCON3	0x7F00891C	R/W	External Interrupt 0(Group0) Filter Control Register 3	0x0
EINT0MASK	0x7F008920	R/W	External Interrupt 0(Group0) Mask Register	0x0FFFFFFF
EINT0PEND	0x7F008924	R/W	External Interrupt 0(Group0) Pending Register	0x0
EINT12CON	0x7F008200	R/W	External Interrupt 1,2(Group1,2) Configuration Register	0x0
EINT34CON	0x7F008204	R/W	External Interrupt 3,4(Group3,4) Configuration Register	0x0
EINT56CON	0x7F008208	R/W	External Interrupt 5,6(Group5,6) Configuration Register	0x0
EINT78CON	0x7F00820C	R/W	External Interrupt 7,8(Group7,8) Configuration Register	0x0
EINT9CON	0x7F008210	R/W	External Interrupt 9(Group9) Configuration Register	0x0
EINT12FLTCON	0x7F008220	R/W	External Interrupt 1,2(Group1,2) Filter Control Register	0x0
EINT34FLTCON	0x7F008224	R/W	External Interrupt 3,4(Group3,4) Filter Control Register	0x0
EINT56FLTCON	0x7F008228	R/W	External Interrupt 5,6(Group5,6) Filter Control Register	0x0
EINT78FLTCON	0x7F00822C	R/W	External Interrupt 7,8(Group7,8) Filter Control Register	0x0
EINT9FLTCON	0x7F008230	R/W	External Interrupt 9(Group9) Filter Control Register	0x0
EINT12MASK	0x7F008240	R/W	External Interrupt 1,2(Group1,2) Mask Register	0x00FF7FFF
EINT34MASK	0x7F008244	R/W	External Interrupt 3,4(Group3,4) Mask Register	0x3FFF03FF
EINT56MASK	0x7F008248	R/W	External Interrupt 5,6(Group5,6) Mask Register	0x03FF007F
EINT78MASK	0x7F00824C	R/W	External Interrupt 7,8(Group7,8) Mask Register	0x7FFFFFFF
EINT9MASK	0x7F008250	R/W	External Interrupt 9(Group9) Mask Register	0x000001FF

EINT9MASK	0x7F008250	R/W	External Interrupt 9(Group9) Mask Register	0x000001FF
EINT12PEND	0x7F008260	R/W	External Interrupt 1,2(Group1,2) Pending Register	0x0
EINT34PEND	0x7F008264	R/W	External Interrupt 3,4(Group3,4) Pending Register	0x0
EINT56PEND	0x7F008268	R/W	External Interrupt 5,6(Group5,6) Pending Register	0x0

EINT78PEND	0x7F00826C	R/W	External Interrupt 7,8(Group7,8) Pending Register	0x0
EINT9PEND	0x7F008270	R/W	External Interrupt 9(Group9) Pending Register	0x0
PRIORITY	0x7F008280	R/W	Priority Control Register	0x3FF
SERVICE	0x7F008284	R	Current Service Register	0x00
SERVICEPEND	0x7F008288	R/W	Current Service Pending Register	0x00

EINT0FLTCON0	Bit	Description	Initial State
FLTEN	[31]	Filter Enable for EINT6, 7 0 = disables 1 = enabled	0
FLTSEL	[30]	Filter Selection for EINT6, 7 0 = delay filter 1 = digital filter(clock count)	0
EINT6, 7	[29:24]	Filtering width of EINT6, 7 This value is valid when FLTSEL is 1.	000
FLTEN	[23]	Filter Enable for EINT4, 5 0 = disables 1 = enabled	0
FLTSEL	[22]	Filter Selection for EINT4, 5 0 = delay filter 1 = digital filter(clock count)	0
EINT4, 5	[21:16]	Filtering width of EINT4, 5 This value is valid when FLTSEL is 1.	000
FLTEN	[15]	Filter Enable for EINT2, 3 0 = disables 1 = enabled	0
FLTSEL	[14]	Filter Selection for EINT2, 3 0 = delay filter 1 = digital filter(clock count)	0
EINT2, 3	[13:8]	Filtering width of EINT2, 3 This value is valid when FLTSEL is 1.	000
FLTEN	[7]	Filter Enable for EINT 0, 1 0 = disables 1 = enabled	0
FLTSEL	[6]	Filter Selection for EINT0, 1 0 = delay filter 1 = digital filter(clock count)	0
EINT0, 1	[5:0]	Filtering width of EINT0, 1 This value is valid when FLTSEL is 1.	000

```
1 #define GPNDAT  ((volatile unsigned long *)0x7f008834)
2 #define GPNPUD  ((volatile unsigned long *)0x7f008838)
3
4 #define EINT0CON0       ((volatile unsigned long *)0x7f008900)
5 #define EINT0MASK       ((volatile unsigned long *)0x7f008920)
6 #define EINT0PEND       ((volatile unsigned long *)0x7f008924)
7
8 #define VIC0INTENABLE   ((volatile unsigned long *)0x71200010)
9 #define VIC0VECTADDR0   ((volatile unsigned long *)0x71200100)
```

```
10 #define VIC0ADDRESS     ((volatile unsigned long *)0x71200f00)
11 #define VIC1ADDRESS     ((volatile unsigned long *)0x71300f00)
12
13 void key1_handle(void)
14 {
15         __asm__ volatile (
16                 "sub lr, lr, #4\n"
17                 "stmfd sp!, {r0-r12, lr}\n"
18                 :
19                 :
20         );
21
22         switch ( (*GPNDAT) & 0x3 ) {
23                 case 0x2:               //key0
24                         led_off(); break;
25                 case 0x1:               //key1
26                         led_on(); break;
27                 default:
28                         break;
29         }
30
31         *(EINT0PEND) = ~0x0;
32         *(VIC0ADDRESS) = 0;
33         *(VIC1ADDRESS) = 0;
34
35         __asm__ volatile (
36                 "ldmfd sp!, {r0-r12, pc}^ \n"
37                 :
38                 :
39         );
40
41 }
42
43 void irq_init(void)
44 {
45         //set GPN0,1 Falling edge triggered
46         *(EINT0CON0) = 0x2;
47
48         //chear EINT0MASK
49         *(EINT0MASK) = 0x0;
50
51         //enable EINT
52         *(VIC0INTENABLE) = 0x1;
53
54         *(VIC0VECTADDR0) = (int)key1_handle;
55
56         __asm__ volatile (
57                 "mrc p15,0,r0,c1,c0,0\n"
58                 "orr r0,r0,#(1<<24)\n"
59                 "mcr p15,0,r0,c1,c0,0\n"
60
61                 "mrs r0,cpsr\n"
62                 "bic r0, r0, #0x80\n"
63                 "msr cpsr_c, r0\n"
64                 :
65                 :
66         );
67
68
69 }
```

2.2.1、配置为下降沿中断

EINT0CON0	Bit	Description	Initial State
Reserved	[31]	Reserved	0
EINT15, 14	[30:28]	Setting the signaling method of the EINT15 and EINT14. 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000
Reserved	[27]	Reserved	0
EINT13, 12	[26:24]	Setting the signaling method of the EINT13 and EINT12. 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000
Reserved	[23]	Reserved	0
EINT11, 10	[22:20]	Setting the signaling method of the EINT11 and EINT10. 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000
Reserved	[19]	Reserved	0
EINT9, 8	[18:16]	Setting the signaling method of the EINT9 and EINT8. 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000
Reserved	[15]	Reserved	0
EINT7, 6	[14:12]	Setting the signaling method of the EINT7 and EINT6. 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000
Reserved	[11]	Reserved	0
EINT5, 4	[10:8]	Setting the signaling method of the EINT5 and EINT4. 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000
Reserved	[7]	Reserved	0
EINT3, 2	[6:4]	Setting the signaling method of the EINT3 and EINT2 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000
Reserved	[3]	Reserved	0
EINT1, 0	[2:0]	Setting the signaling method of the EINT0 and EINT1 000 = Low level 001 = High level 01x = Falling edge triggered 10x = Rising edge triggered 11x = Both edge triggered	000

```
45      //set GPN0,1 Falling edge triggered
46      *(EINT0CON0) = 0x2;
```

2.2.2、取消屏蔽外部中断

EINT0MASK	Bit	Description	Initial State
EINT27	[27]	0 = Enable Interrupt 1= Masked	1
EINT26	[26]	0 = Enable Interrupt 1= Masked	1
EINT25	[25]	0 = Enable Interrupt 1= Masked	1
EINT24	[24]	0 = Enable Interrupt 1= Masked	1
EINT23	[23]	0 = Enable Interrupt 1= Masked	1
EINT22	[22]	0 = Enable Interrupt 1= Masked	1
EINT21	[21]	0 = Enable Interrupt 1= Masked	1
EINT20	[20]	0 = Enable Interrupt 1= Masked	1
EINT19	[19]	0 = Enable Interrupt 1= Masked	1
EINT18	[18]	0 = Enable Interrupt 1= Masked	1
EINT17	[17]	0 = Enable Interrupt 1= Masked	1
EINT16	[16]	0 = Enable Interrupt 1= Masked	1
EINT15	[15]	0 = Enable Interrupt 1= Masked	1
EINT14	[14]	0 = Enable Interrupt 1= Masked	1
EINT13	[13]	0 = Enable Interrupt 1= Masked	1
EINT12	[12]	0 = Enable Interrupt 1= Masked	1
EINT11	[11]	0 = Enable Interrupt 1= Masked	1
EINT10	[10]	0 = Enable Interrupt 1= Masked	1
EINT9	[9]	0 = Enable Interrupt 1= Masked	1
EINT8	[8]	0 = Enable Interrupt 1= Masked	1
EINT7	[7]	0 = Enable Interrupt 1= Masked	1
EINT6	[6]	0 = Enable Interrupt 1= Masked	1
EINT5	[5]	0 = Enable Interrupt 1= Masked	1
EINT4	[4]	0 = Enable Interrupt 1= Masked	1
EINT3	[3]	0 = Enable Interrupt 1= Masked	1
EINT2	[2]	0 = Enable Interrupt 1= Masked	1
EINT1	[1]	0 = Enable Interrupt 1= Masked	1
EINT0	[0]	0 = Enable Interrupt 1= Masked	1

```
48      //chear EINT0MASK
49      *(EINT0MASK) = 0x0;
```

2.2.3、设置外部中断跳转地址

12.6.12 VECTOR ADDRESS REGISGERS, VICVECTADDR[0-31]

REGISTER	ADDRESS	R/W	DESCRIPTION	RESET VALUE
VIC0VECTADDR[31:0]	0x7120_0100 ~ 0x7120_017C	R/W	Vector Address [31:0] Register (VIC0)	0x0000_0000
VIC0VECTADDR[31:0]	0x7130_0100 ~ 0x7130_017C	R/W	Vector Address [31:0] Register. (VIC1)	0x0000_0000

Name	BIT	DESCRIPTION	RESET VALUE
VectorAddr	[31:0]	Contains ISR vector addresses.	0x0000_0000

```
54      *(VIC0VECTADDR0) = (int)key1_handle;
```

2.2.4、使能向量中断方式

```
56      __asm__ volatile (
57              "mrc p15,0,r0,c1,c0,0\n"
58              "orr r0,r0,#(1<<24)\n"
59              "mcr p15,0,r0,c1,c0,0\n"
```

```
60
61              "mrs r0,cpsr\n"
62              "bic r0, r0, #0x80\n"
63              "msr cpsr_c, r0\n"
64              :
65              :
66      );
```

2.2.5、使能外部中断

12.5 SUMMARY OF VIC REGISTERS

- Base address of VIC0 is 0x7120_0000
- Base address of VIC1 is 0x7130_0000
- Address of control register = base address + offset

Register	Offset	Type	Description	Reset Value
VICxIRQSTATUS	0x000	R	IRQ Status Register	0x00000000
VICxFIQSTATUS	0x004	R	FIQ Status Register	0x00000000
VICxRAWINTR	0x008	R	Raw Interrupt Status Register	0x00000000
VICxINTSELECT	0x00C	RW	Interrupt Select Register	0x00000000
VICxINTENABLE	0x010	RW	Interrupt Enable Register	0x00000000
VICxINTENCLEAR	0x014	W	Interrupt Enable Clear Register	-
VICxSOFTINT	0x018	RW	Software Interrupt Register	0x00000000
VICxSOFTINTCLEAR	0x01C	W	Software Interrupt Clear Register	-
VICxPROTECTION	0x020	RW	Protection Enable Register	0x0
VICxSWPRIORITYMASK	0x024	RW	Software Priority Mask Register	0xFFFF
VICxPRIORITYDAISY	0x028	RW	Vector Priority Register for Daisy Chain	0xF
VICxVECTADDR0	0x100	RW	Vector Address 0 Register	0x00000000
VICxVECTADDR1	0x104	RW	Vector Address 1 Register	0x00000000
VICxVECTADDR2	0x108	RW	Vector Address 2 Register	0x00000000
VICxVECTADDR3	0x10C	RW	Vector Address 3 Register	0x00000000
VICxVECTADDR4	0x110	RW	Vector Address 4 Register	0x00000000
VICxVECTADDR5	0x114	RW	Vector Address 5 Register	0x00000000
VICxVECTADDR6	0x118	RW	Vector Address 6 Register	0x00000000
VICxVECTADDR7	0x11C	RW	Vector Address 7 Register	0x00000000
VICxVECTADDR8	0x120	RW	Vector Address 8 Register	0x00000000
VICxVECTADDR9	0x124	RW	Vector Address 9 Register	0x00000000
VICxVECTADDR10	0x128	RW	Vector Address 10 Register	0x00000000
VICxVECTADDR11	0x12C	RW	Vector Address 11 Register	0x00000000
VICxVECTADDR12	0x130	RW	Vector Address 12 Register	0x00000000
VICxVECTADDR13	0x134	RW	Vector Address 13 Register	0x00000000
VICxVECTADDR14	0x138	RW	Vector Address 14 Register	0x00000000
VICxVECTADDR15	0x13C	RW	Vector Address 15 Register	0x00000000
VICxVECTADDR16	0x140	RW	Vector Address 16 Register	0x00000000
VICxVECTADDR17	0x144	RW	Vector Address 17 Register	0x00000000
VICxVECTADDR18	0x148	RW	Vector Address 18 Register	0x00000000

Register	Offset	Type	Description	Reset Value
VICxVECTADDR19	0x14C	RW	Vector Address 19 Register	0x00000000
VICxVECTADDR20	0x150	RW	Vector Address 20 Register	0x00000000
VICxVECTADDR21	0x154	RW	Vector Address 21 Register	0x00000000
VICxVECTADDR22	0x158	RW	Vector Address 22 Register	0x00000000
VICxVECTADDR23	0x15C	RW	Vector Address 23 Register	0x00000000
VICxVECTADDR24	0x160	RW	Vector Address 24 Register	0x00000000
VICxVECTADDR25	0x164	RW	Vector Address 25 Register	0x00000000
VICxVECTADDR26	0x168	RW	Vector Address 26 Register	0x00000000
VICxVECTADDR27	0x16C	RW	Vector Address 27 Register	0x00000000
VICxVECTADDR28	0x170	RW	Vector Address 28 Register	0x00000000
VICxVECTADDR29	0x174	RW	Vector Address 29 Register	0x00000000
VICxVECTADDR30	0x178	RW	Vector Address 30 Register	0x00000000
VICxVECTADDR31	0x17C	RW	Vector Address 31 Register	0x00000000
VICxVECPRIORITY0	0x200	RW	Vector Priority 0 Register	0xF
VICxVECTPRIORITY1	0x204	RW	Vector Priority 1 Register	0xF
VICxVECTPRIORITY2	0x208	RW	Vector Priority 2 Register	0xF
VICxVECTPRIORITY3	0x20C	RW	Vector Priority 3 Register	0xF
VICxVECTPRIORITY4	0x210	RW	Vector Priority 4 Register	0xF
VICxVECTPRIORITY5	0x214	RW	Vector Priority 5 Register	0xF
VICxVECTPRIORITY6	0x218	RW	Vector Priority 6 Register	0xF
VICxVECTPRIORITY7	0x21C	RW	Vector Priority 7 Register	0xF
VICxVECTPRIORITY8	0x220	RW	Vector Priority 8 Register	0xF
VICxVECTPRIORITY9	0x224	RW	Vector Priority 9 Register	0xF
VICxVECTPRIORITY10	0x228	RW	Vector Priority 10 Register	0xF
VICxVECTPRIORITY11	0x22C	RW	Vector Priority 11 Register	0xF
VICxVECTPRIORITY12	0x230	RW	Vector Priority 12 Register	0xF
VICxVECTPRIORITY13	0x234	RW	Vector Priority 13 Register	0xF
VICxVECTPRIORITY14	0x238	RW	Vector Priority 14 Register	0xF
VICxVECTPRIORITY15	0x23C	RW	Vector Priority 15 Register	0xF
VICxVECTPRIORITY16	0x240	RW	Vector Priority 16 Register	0xF
VICxVECTPRIORITY17	0x244	RW	Vector Priority 17 Register	0xF
VICxVECTPRIORITY18	0x248	RW	Vector Priority 18 Register	0xF
VICxVECTPRIORITY19	0x24C	RW	Vector Priority 19 Register	0xF
VICxVECTPRIORITY20	0x250	RW	Vector Priority 20 Register	0xF

Register	Offset	Type	Description	Reset Value
VICxVECTPRIORITY21	0x254	RW	Vector Priority 21 Register	0xF
VICxVECTPRIORITY22	0x258	RW	Vector Priority 22 Register	0xF
VICxVECTPRIORITY23	0x25C	RW	Vector Priority 23 Register	0xF
VICxVECTPRIORITY24	0x260	RW	Vector Priority 24 Register	0xF
VICxVECTPRIORITY25	0x264	RW	Vector Priority 25 Register	0xF
VICxVECTPRIORITY26	0x268	RW	Vector Priority 26 Register	0xF
VICxVECTPRIORITY27	0x26C	RW	Vector Priority 27 Register	0xF
VICxVECTPRIORITY28	0x270	RW	Vector Priority 28 Register	0xF
VICxVECTPRIORITY29	0x274	RW	Vector Priority 29 Register	0xF
VICxVECTPRIORITY30	0x278	RW	Vector Priority 30 Register	0xF
VICxVECTPRIORITY31	0x27C	RW	Vector Priority 31 Register	0xF
VICxADDRESS	0xF00	RW	Vector Address Register	0x00000000

12.6.5 INTERRUPT ENABLE REGISTER, VICINTENABLE

REGISTER	ADDRESS	R/W	DESCRIPTION	RESET VALUE
VIC0INTENABLE	0x7120_0010	R/W	Interrupt Enable Register (VIC0)	0x0000_0000
VIC1INTENABLE	0x7130_0010	R/W	Interrupt Enable Register (VIC1)	0x0000_0000

Name	BIT	DESCRIPTION	RESET VALUE
IntEnable	[31:0]	Enables the interrupt request lines, which allow the interrupts to reach the processor. Read: 0 = interrupt disabled (reset) 1 = Interrupt enabled The interrupt enable can only be set using this register. The VICINTENCLEAR Register must be used to disable the interrupt enable. Write: 0 = no effect 1 = interrupt enabled. On reset, all interrupts are disabled. There is one bit of the register for each interrupt source	0x0

```
51      //enable EINT
52      *(VIC0INTENABLE) = 0x1;
```

2.3、中断处理

2.3.1、保存环境变量

```
13 void key1_handle(void)
14 {
15      __asm__ volatile (
16              "sub lr, lr, #4\n"
17              "stmfd sp!, {r0-r12, lr}\n"
18              :
19              :
20      );
21
```

2.3.2、处理中断

按键0关灯，按键1开灯。

```
22      switch ( (*GPNDAT) & 0x3 ) {
23              case 0x2:               //key0
24                      led_off(); break;
25              case 0x1:               //key1
26                      led_on(); break;
27              default:
28                      break;
29      }
30
```

2.3.3、清除中断

EINT0PEND	Bit	Description	Initial State
EINT27	[27]	0 = Not occur 1= Occur interrupt	0
EINT26	[26]	0 = Not occur 1= Occur interrupt	0
EINT25	[25]	0 = Not occur 1= Occur interrupt	0
EINT24	[24]	0 = Not occur 1= Occur interrupt	0
EINT23	[23]	0 = Not occur 1= Occur interrupt	0
EINT22	[22]	0 = Not occur 1= Occur interrupt	0
EINT21	[21]	0 = Not occur 1= Occur interrupt	0
EINT20	[20]	0 = Not occur 1= Occur interrupt	0
EINT19	[19]	0 = Not occur 1= Occur interrupt	0
EINT18	[18]	0 = Not occur 1= Occur interrupt	0
EINT17	[17]	0 = Not occur 1= Occur interrupt	0
EINT16	[16]	0 = Not occur 1= Occur interrupt	0
EINT15	[15]	0 = Not occur 1= Occur interrupt	0
EINT14	[14]	0 = Not occur 1= Occur interrupt	0
EINT13	[13]	0 = Not occur 1= Occur interrupt	0
EINT12	[12]	0 = Not occur 1= Occur interrupt	0
EINT11	[11]	0 = Not occur 1= Occur interrupt	0
EINT10	[10]	0 = Not occur 1= Occur interrupt	0
EINT9	[9]	0 = Not occur 1= Occur interrupt	0
EINT8	[8]	0 = Not occur 1= Occur interrupt	0
EINT7	[7]	0 = Not occur 1= Occur interrupt	0
EINT6	[6]	0 = Not occur 1= Occur interrupt	0
EINT5	[5]	0 = Not occur 1= Occur interrupt	0
EINT4	[4]	0 = Not occur 1= Occur interrupt	0
EINT3	[3]	0 = Not occur 1= Occur interrupt	0
EINT2	[2]	0 = Not occur 1= Occur interrupt	0
EINT1	[1]	0 = Not occur 1= Occur interrupt	0
EINT0	[0]	0 = Not occur 1= Occur interrupt	0

NOTES: 1. Each bit is cleared by writing "1"
2. EINT0~27 of Group0 are wake-up source in stop and sleep mode.
3. When EINT0~27 are used as wake-up sources, EINT_mask(System controller register) must be set un-mask

12.6.14 VECTOR ADDRESS REGISTER, VICADDRESS

REGISTER	ADDRESS	R/W	DESCRIPTION	RESET VALUE
VIC0ADDRESS	0x7120_0F00	R/W	Vector Address Register (VIC0)	0x0000_0000
VIC1ADDRESS	0x7130_0F00	R/W	Vector Address Register (VIC1)	0x0000_0000

Name	BIT	DESCRIPTION	RESET VALUE
VectAddr	[31:0]	Contains the address of the currently active ISR, with reset value 0x00000000. A read of this register returns the address of the ISR and sets the current interrupt as being serviced. A read must only be performed while there is an active interrupt. A write of any value to this register clears the current interrupt. A write must only be performed at the end of an interrupt service routine.	0x0

```
31      *(EINT0PEND) = ~0x0;
32      *(VIC0ADDRESS) = 0;
33      *(VIC1ADDRESS) = 0;
34
```

2.3.4、恢复环境变量

```
35      __asm__ volatile (
36              "ldmfd sp!, {r0-r12, pc}^ \n"
37              :
38              :
39      );
40
41 }
```

2.4、要正常使用中断，还需要初始化IRQ模式下的R13

```
195 stack_init:
196     msr cpsr_c, #0xd2
197     ldr sp, =0x53000000     //init R13_IRQ
198
199     msr cpsr_c, #0xd3
200     ldr sp, =0x54000000     //init R13_SVC
201     mov pc, lr
```