目 录

1. 产品概述 2
2. 主要特性 2
3. 产品型号一览表 4
4. 系统框图 5
5. 管脚配置 6
6. 中央处理器 8
6.1 指令集 8
6.2 ROM 10
6.3 RAM 11
6.4 CPU SFR 11
6.5 SFR 14
6.6 OPTION 16
7. 功能模块 17
7.1 Clock 17
7.2 Mode 18
7.3 INT 19
7.4 GPIO 24
7.5 TIMER 28
7.5.1 TIMER0 28
7.5.2 TIMER1 30
7.5.3 TIMER2 34
7.6 LVD 40
7.7 TOUCH 41
7.8 Mini I2C 43
7.9 WDT 45
7.10 Reset 46
8. 电气特性 47
8.1 极限参数 47
8.2 直流特性 47
8.3 振荡器特性 48
9. 封装尺寸 49
9.1 SOP16封装 49
9.2 QFN3*3-16封装 50
10. 历史记录 51

触控 IO 型 8-Bit MCU

文件编号：PT-DS26005

1. 产品概述

PT8P2107 是一款 RISC 内核，TOUCH 型 8 位 OTP 单片机，内部集成电容式触摸感应模块、PWM、DAC、Mini IIC 等外设，其主要用作触摸按键开关，广泛适用于触控调光、电子玩具、消费电子、家用电器等领域，具有低功耗、高可靠性、宽工作电压范围的突出优势。

2. 主要特性

■ CPU

- RISC 内核，支持 73 条指令
- 支持 8 级硬件堆栈
- 指令周期可配置为 2T/4T/8T/16T
- 复位向量位于 000h
- 支持 9 种中断源，T0、T1、T2、INT0、INT1、TOUCH、KEY、LVD、IIC
- 2 级中断优先级可设，高优先级中断入口地址：018h，低优先级中断入口地址：008h
- 支持直接与间接数据寻址方式
- 程序存储器 OTP ROM：2K*16bit
- 数据存储器 SRAM：240*8bit

■ I/O 口

- 14 个双向 I/O 端口，带 SMIT 输入，可配置为漏极开路、内置上拉电阻及下拉电阻
- P10~P17、P01~P04 可配置为触摸通道
- P05 可复用为触摸采样电容端口 CMOD
- 所有端口均支持键盘中断功能
- I/O 输入电压门限 2 档可选：0.7*VDD/0.3*VDD、0.5*VDD/0.2*VDD

■ 电容式触摸感应模块

- 触摸模块内部集成 2.3V/2.0V 可选的 LDO，并采用电荷分享方式实现触摸检测，具有很高的可靠性和抗干扰能力
- 10 路外部触摸输入（其中 TCH0~TCH5 为高穿透力通道）
- 支持触摸引脚与 LED 驱动引脚复用，节省芯片引脚资源
- 支持键值 DAC 输出

■ 定时器

- TIMER0
  - 自动装载 8 位定时器，支持预分频功能
- TIMER1
  - 自动装载 10 位定时器，支持预分频功能
  - 2 路独立的 PWM(PWM0~PWM1)

- TIMER2
  - 自动装载 8 位定时器，支持预分频功能
  - 3 路独立的 PWM(PWM2~PWM4)
  - PWM3 支持单线级联 LED 驱动，可应用于 RGB 七彩灯带

■ Mini I2C

- 1 路精简 I2C 接口，仅支持从机模式
- 支持标准/快速模式，最高支持 400Kbit/S

■ CPU 保护系统及工作模式

- 4 种系统复位方式：
  - 上电复位(POR)
  - 低压复位(LVR)
  - 看门狗(WDT)溢出复位
  - 软件复位(WRST)
- 支持 3 种工作模式
  - Normal 模式：正常工作模式
  - STOP 模式：低功耗模式，CPU 停止工作，外设停止工作
  - 唤醒方式：外部中断、KEY 中断、TIMER0 中断、WDT 溢出、TOUCH 中断
  - IDLE 模式：仅 CPU 停止工作，其它外设可以工作
  - 唤醒方式：所有中断、WDT 溢出
- 内嵌 LVR 功能，复位阈值可选为： 关闭、1.8V、2.0V、2.2V、2.4V、2.7V、2.9V、3.6V
- （误差±0.1V）
- 内嵌 LVD 功能，检测阈值可选为： 2.2V、2.5V、2.7V、3.0V、3.3V、3.6V、3.9V、4.5V
- （误差±0.1V）
- 内嵌 WDT，支持预分频功能。4 档 WDT 溢出时间可选：8ms、16ms、128ms、256ms

■ 时钟系统

- 内部 RC 高频振荡器 16MHz 精度：±0.5%(typ)
- 内部 RC 低频振荡器 131KHz 精度：±5%(typ)

■ 工作温度范围

- -40℃ ~ +85℃

■ 电压工作范围

- 8MHz @3.0V~5.5V
- 4MHz @2.2V~5.5V
- 2MHz @1.8V~5.5V

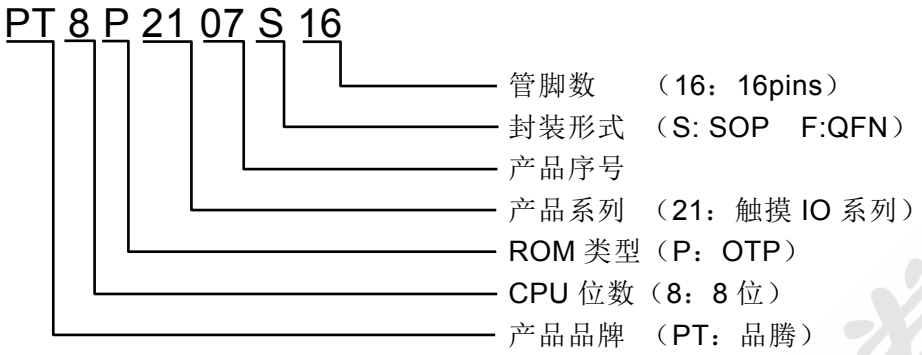
■ 抗干扰能力

- HBM ESD：优于 4000V

■ 封装形式

- SOP16、QFN16

3. 产品型号一览表



Part No	ROM	RAM	Touch	PWM	Package
PT8P2107S16	2K*16BIT	240*8BIT	10	5	SOP16
PT8P2107F16	2K*16BIT	240*8BIT	10	5	QFN16

4. 系统框图

基于 RISC 的架构绝大部分指令都只需一个指令执行周期，少部分需要两个指令执行周期。内置 2K*16bit OTP，240*8bit SRAM；同时内部集成了 TOUCH、TIMER、PWM、Mini IIC、WDT 等外设。

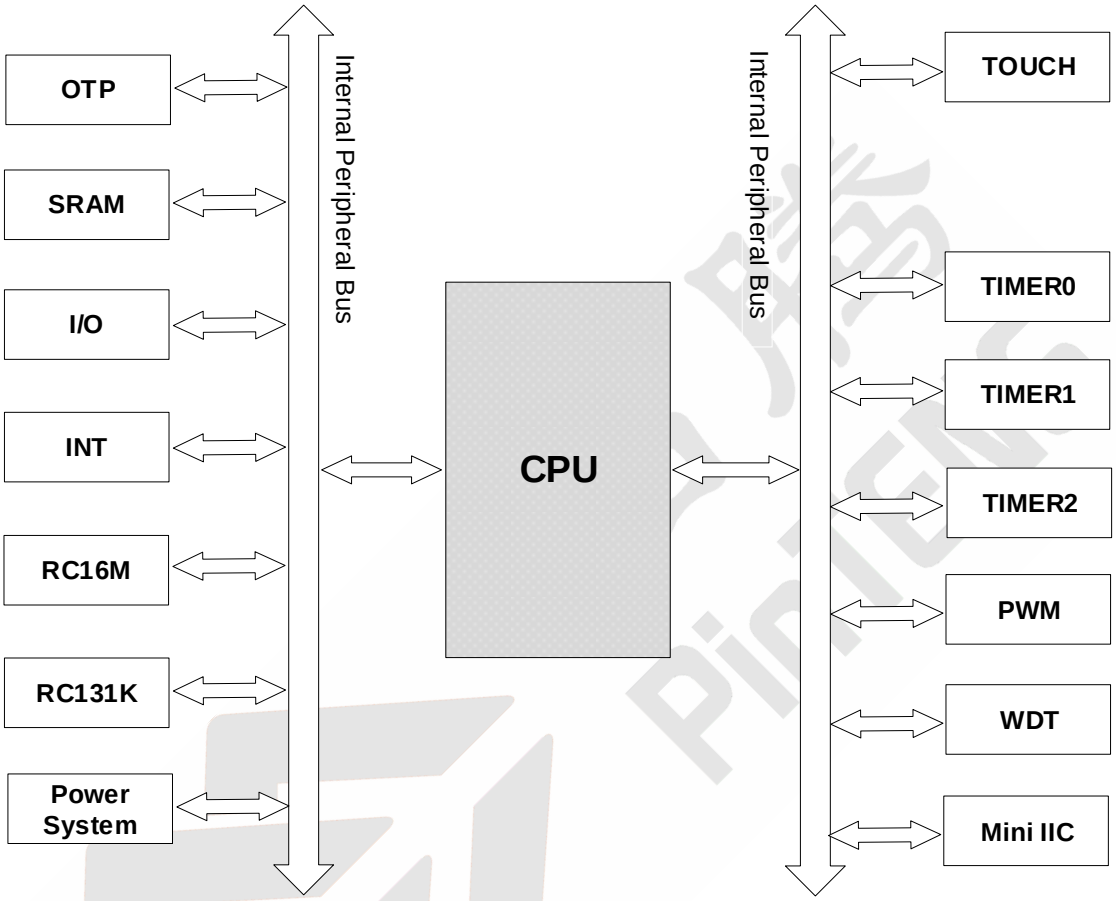


图 1 系统框图

5. 管脚配置

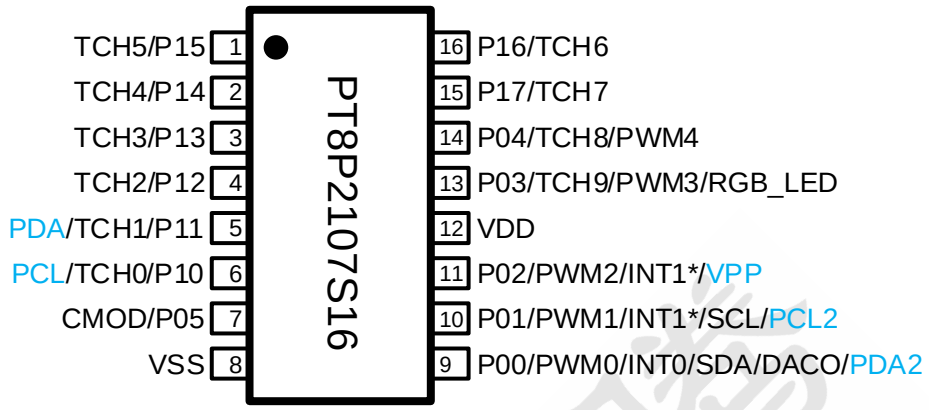


图 2 SOP16 封装示意图

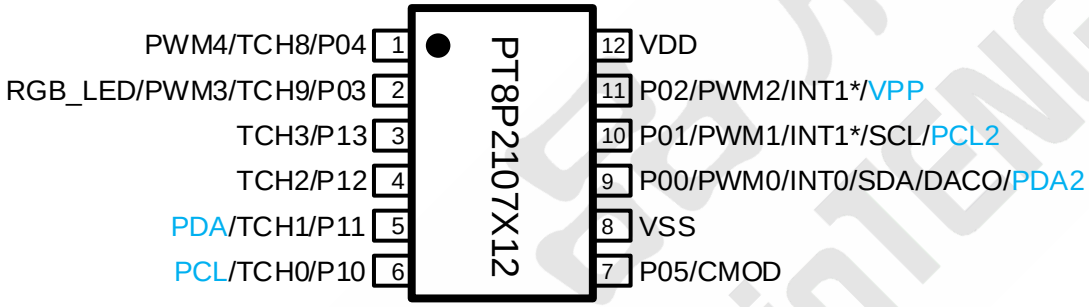


图 3 XSOP12L 封装示意图

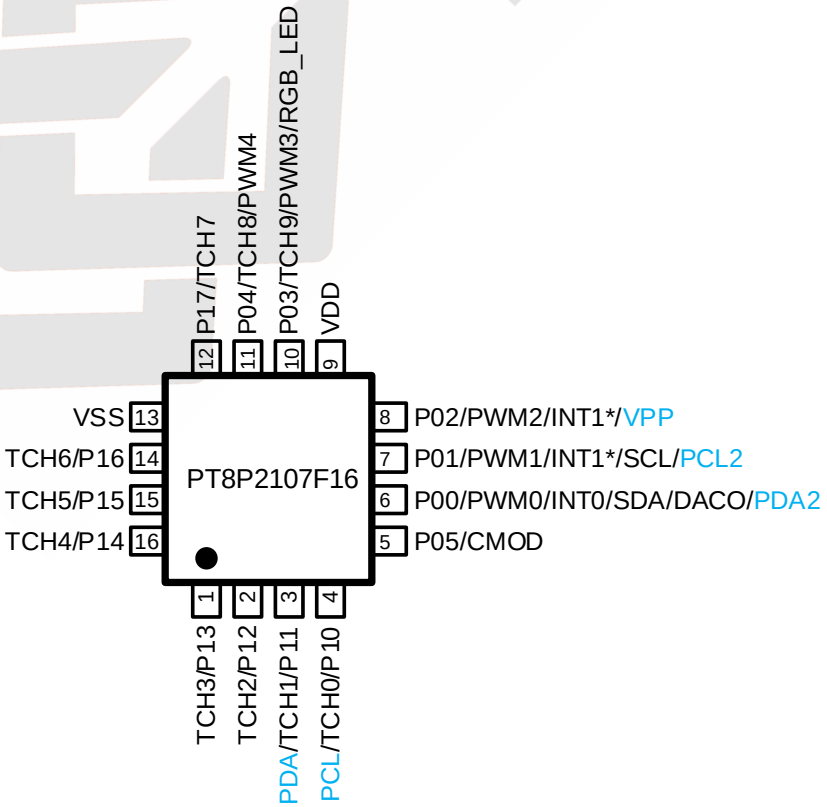


图 4 QFN16 封装示意图

表 1 管脚信号说明表

管脚名称	I/O类型	管脚说明
VSS	P	地
VDD	P	电源
CMOD	I/O	采样电容接入脚
PWM[i]	O	PWM[i]输出口，i=0~4
TCH[i]	I	触摸通道[i]，i=0~11
P0[i]	I/O	P0输入/输出IO[i]，可配置弱上拉/下拉、开漏输出功能，i=0~5
P1[i]	I/O	P1输入/输出IO[i]，可配置弱上拉/下拉、开漏输出功能，i=0~7
INT[i]	I	外部中断[i]输入口，i=0~1
PCL	I	烧录时钟线
PDA	I/O	烧录数据线
SCL	I/O	IIC时钟线
SDA	I/O	IIC数据线
DACO	AO	DAC电压输出口
VPP	P	烧录高压
RGB_LED	O	级联LED驱动

6. 中央处理器

6.1 指令集

表 2 MCU 指令集

类别	指令格式	指令意义	周期	标志位
算术运算	ADDK K	A ← A + K	1	C DC Z
	ADD A, R	A ← A + R	1	
	ADDR A, R	R ← A + R	1	
	ADDC A, R	A ← A + R + C	1	
	ADDCR A, R	R ← A + R + C	1	
	SUBK K	A ← A - K	1	
	SUB A, R	A ← A - R	1	
	SUBR A, R	R ← A - R	1	
	SUBC A, R	A ← A - R - (~C)	1	
	SUBCR A, R	R ← A - R - (~C)	1	
	DAR R	A进行BCD调整，存入R中	1	C
	DAA	A进行BCD调整，存入A中	1	C
	DAAF	A进行BCD调整，存入A中	1	C DC
	MUL	(A)*(B)的低字节存入A中，(A)*(B)的高字节存入B中	4	~
	DIV	(A)/(B)的商存入A中，(A)/(B)的余数存入B中	4	OV
逻辑运算	ANDK K	A ← A & K	1	Z
	AND A, R	A ← A & R	1	Z
	ANDR A, R	R ← A & R	1	Z
	ORK K	A ← A | K	1	Z
	OR A, R	A ← A | R	1	Z
	ORR A, R	R ← A | R	1	Z
	CPL R	A ← NOT(R)	1	Z
	CPLR R	R ← NOT(R)	1	Z
	XORK K	A ← A ^ K	1	Z
	XOR A, R	A ← A ^ R	1	Z
	XORR A, R	R ← A ^ R	1	Z
	BCPL R, b	R的第b个位取反后写回R	1	~
递增和递减指令	INC R	A ←R + 1	1	Z
	INCR R	R ←R + 1	1	Z
	INCSZ R	A ←R + 1，如果A=0，则跳过下一条指令	1 or 2	~
	INCSZR R	R ←R + 1，如果R=0，则跳过下一条指令	1 or 2	~
	DEC R	A ←R - 1	1	Z
	DECR R	R ←R - 1	1	Z
	DECSZ R	A ←R - 1，如果A=0，则跳过下一条指令	1 or 2	~
	DECSZR R	R ←R - 1，如果R=0，则跳过下一条指令	1 or 2	~
移位指令	RLC R	A ← R带进位循环左移1位	1	C
	RLCR R	R ← R带进位循环左移1位	1	C
	RRC R	A ← R带进位循环右移1位	1	C
	RRCR R	R ← R带进位循环右移1位	1	C
	RL R	A ← R循环左移1位	1	~
	RLR R	R ← R循环左移1位	1	~
	RR R	A ← R循环右移1位	1	~
	RRR R	R ← R循环右移1位	1	~
数据传送	MOV A, R	A ← R	1	Z
	MOV R, A	R ← A	1	~
	MOVK K	A ← K	1	~
	MOVRR R	R ← R，两个R为同一地址，影响Z	1	Z
	XCH R	A/R内容对调	2	~

位操作	BCLR	R, b	R[b] ← 0	1	~
	BSET	R, b	R[b] ← 1	1	~
转移指令	JMP	AA	PC←AA，AA为13bit值，JMP可跳转8K ROM空间	2	~
	BTSZ	R, b	如果R[b]=0，则跳过下一条指令	1 or 2	~
	BTSNZ	R, b	如果R[b]=1，则跳过下一条指令	1 or 2	~
	CALL	AA	Push pc+1，then PC←AA，AA为13bit值，CALL可跳转8K ROM空间	2	~
	RET		PC值出栈	2	~
	RETK	K	PC值出栈同时K赋给累加器A	2	~
	RETI		PC值出栈同时全局中断使能置1	2	~
	SZR	R	R ← R，如果R=0，则跳过下一跳指令	1 or 2	~
	SZ	R	A ← R，如果R=0，则跳过下一跳指令	1 or 2	~
	SE	R	如果A=R，则跳过下一条指令	1 or 2	C Z
	SEK	K	如果A=K，则跳过下一条指令	1 or 2	C Z
其它指令	NOP		空指令不作任何操作	1	~
	CLR	R	把RAM（R）中的值赋0	1	Z
	SET	R	把RAM（R）中的值赋0xff	1	~
	CLRWDT		Clear WDT	1	~
	SWAP	R	R的高四位和低四位交换，结果放入A	1	~
	SWAPR	R	R的高四位和低四位交换，结果放入R	1	~
	STOP		芯片进入STOP状态	1	~
	IDLE		芯片进入IDLE状态	1	~
	MPSEL	AA	[MPH0，MPL0]←AA，AA为9bit值，设置间接寻址地址寄存器	1	~
	MSTEP	K	MP←MP + K(-128≤K≤127)	1	~
	ESTEP	K	EADR←EADR + K(-128≤K≤127)	1	~
查表	MOVC	R	ROM[EADRH，EADRL]→ [EDATH，R]把ROM地址(EADRH，EADRL)中的值高8位赋给EDATH，低8位赋值给R	2	~

参数说明：

R： 数据存储器地址　　A： 工作寄存器　　K： 立即数

b： 位选择(0~7)　　PC： 程序计数器　　C： 进位标志

DC： 半加进位标志　　Z： 结果为零标志　　OV： 除数为 0 标志

6.2 ROM

程序存储用于存放用户程序、数据等信息，单片机内部集成了 2K×16bit 的 CODE 区和 16×16bit 的 INFO(OPTION)区。CODE 区由 12 位 PC 指针访问，复位地址为 000h，中断向量地址 008h、018h，支持 8 级硬件堆栈，程序存储器和堆栈结构如下：

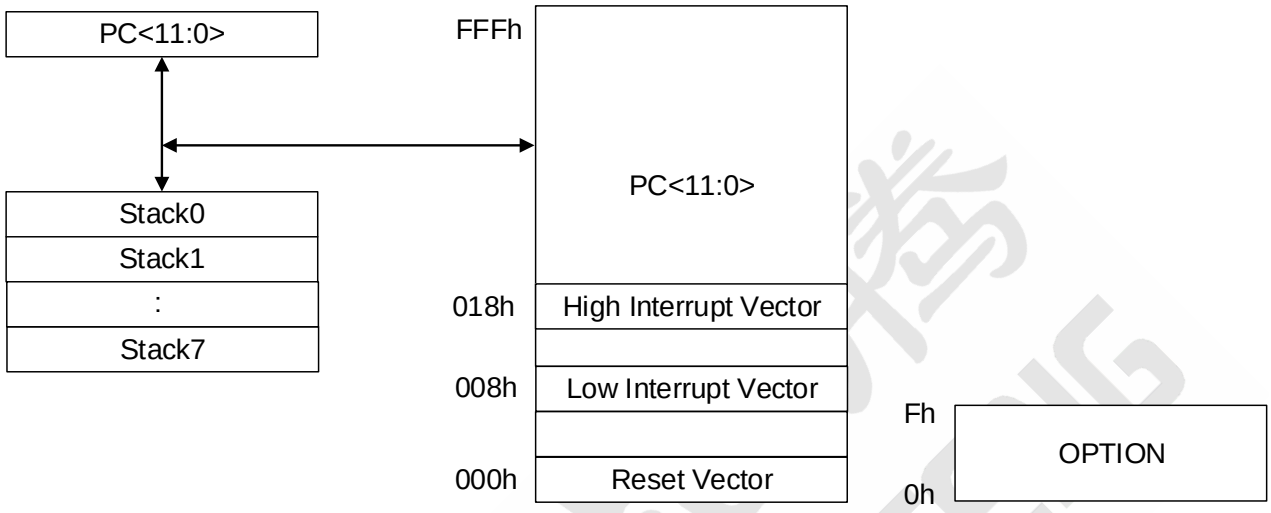


图 5 程序存储器 ROM

■ **说明：堆栈级数为 8 级，如果用户使用时超过此级数，则会导致功能出错**

6.3 RAM

有一个 240×8bit 的数据存储空间。逻辑地址和 SRAM 物理地址映射关系如下:

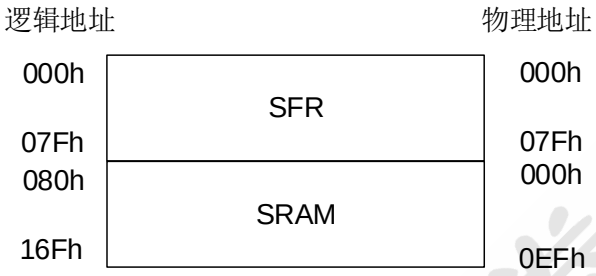


图 6 数据存储器 SRAM

1. 直接寻址

SFR/SRAM 地址空间都可以直接寻址，地址在指令编码中指定。

2. 间接寻址

间接寻址空间包括所有 SRAM 空间及 SFR 空间。间接寻址通过间接寻址地址寄存器{MPH0，MPL0}来访问其所指向的地址，其与直接寻址时的地址是完全一致的。

6.4 CPU SFR

CPU 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
000h	IAR0	通过{MPH00, MPL0}访问数据区（不是一个实际的物理地址）								xxxx xxxx
002h	MPL0	MPL0[7:0]								0000 0000
003h	MPH0	-	-	-	-	-	-	-	MPH00	---- ---0
004h	GPCR0	GPCR0[7:0]								xxxx xxxx
005h	GPCR1	GPCR1[7:0]								xxxx xxxx
006h	STATUS	-	-	-	OV	F0	Z	DC	C	---0 0xxx
007h	ACC	ACC[7:0]								xxxx xxxx
008h	PCL	PCL[7:0]								0000 0000
009h	EADRH	INFOS	-	-	-	EADRH[3:0]				0--- 0000
00Ah	EADRL	EADRL[7:0]								0000 0000
00Bh	EDATH	EDATH[7:0]								xxxx xxxx
00Dh	B	B[7:0]								0000 0000
00Eh	GPCR2	GPCR2[7:0]								xxxx xxxx
00Fh	GPCR3	GPCR3[7:0]								xxxx xxxx

- **间接寻址寄存器(IAR0, MPH0,MPL0)：**

IAR0 不是一个实际的物理地址。间接寻址时通过间接寻址地址寄存器{MPH0,MPL0}来访问其所指向的地址，其与直接寻址时的地址是完全一致的。

- **状态寄存器(STATUS，006h)：**

状态寄存器包含运算标志，结果标志。

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	OV	F0	Z	DC	C
Access	-	-	-	RO	RW	RW	RW	RW
Default	-	-	-	0	0	x	x	x

Bit[4] **OV**：溢出标志位（只在除法指令生效）

1：执行除法指令时，B 寄存器值为 0

0：执行除法指令时，B 寄存器值不为 0

Bit[3] **F0**：F0 标志位，用户自定义标志位

Bit[2] **Z**：零标志

1：算术或逻辑操作结果为 0

0：算术或逻辑操作结果不为 0

Bit[1] **DC**：辅助进位标志/借位标志，用于借位时，极性相反

Bit[0] **C**：进位标志/借位标志，用于借位时，极性相反

- **累加器(ACC，007h)：**

Accumulator 是一个内部数据转化、指令操作和存放操作结果的存储单元

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	ACC[7:0]							
Access	RW							
Default	0xxx							

- **乘除法寄存器(B，00Dh)：**

B 寄存器用来存放乘除法逻辑运算的结果

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	B[7:0]							
Access	RW							
Default	0x00							

- **PC 指针低 8 位(PCL，008h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	PCL[7:0]							
Access	RW							
Default	0x00							

PCL 只能通过 ADDR A, PCL 指令跳转（注：执行此指令后 PC=PC+PCL, PCL=PCL+ACC；除此指令外，对 PCL 操作的其它指令不能改变 PC 值）。

软件可以读取它得到 PC 的低 8 位的值，比如：“MOV A,PCL”等，对 PCL 的间接寻址也是无效的

- **MOVC 地址寄存器(EADRH，009h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

Name	INFOS	-	-	-	EADRH[3:0]
Access	RW	-	-	-	RW
Default	0	-	-	-	0000

Bit[7] **INFOS**：MOVC 指令对 INFO ROM 区访问选择位

1：MOVC 指令访问 INFO ROM 区

0：MOVC 指令访问 CODE ROM 区

Bit[3:0] **EADRH[3:0]**：读取程序存储器地址的高 4 位

- **MOVC 地址寄存器(EADRL，00Ah)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	EADRL[3:0]							
Access	RW							
Default	0x00							

Bit[7:0] **EADRL**：读取程序存储器地址的低 8 位

- **MOVC 数据寄存器(EDATH，00Bh)**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	EDATH[7:0]							
Access	RW							
Default	0x00							

Bit[7:0] **EDATH**：程序存储器数据在 SFR 中的存储地址

ROM[EADRH，EADRL]→[EDATH，R]，即把 ROM 地址[EADRH，EADRL]中的值高 8 位赋给 SFR 的 EDATH，低 8 位赋值给 SRAM 的 R 地址

举例：将程序存储器地址 0123h 中的数据传送到 SFR 的 EDATH 和 55h，然后再将高 8 位数据赋值给 ACC

```
MOVK   0x01
MOV    EADRH, A    //send ROM Hbyte address to EADRH
MOVK   0x23
MOV    EADRL, A    //send ROM Lbyte address to EADRL
MOVC   0x55
MOV    A, EDATH    //send Hbyte data to ACC
```

6.5 SFR

特殊功能寄存器（SFR）包含系统专用寄存器和辅助专用寄存器，详细描述如下所示：

表 3 寄存器列表（SFR）

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
000h	IAR0	通过{MPH00, MPL0}访问数据区（不是一个实际的物理地址）								xxxx xxxx
002h	MPL0	MPL0[7:0]								0000 0000
003h	MPH0	-	-	-	-	-	-	-	MPH00	---- ---0
004h	GPCR0	GPCR0[7:0]								xxxx xxxx
005h	GPCR1	GPCR1[7:0]								xxxx xxxx
006h	STATUS	-	-	-	OV	F0	Z	DC	C	---0 0xxx
007h	ACC	ACC[7:0]								xxxx xxxx
008h	PCL	PCL[7:0]								0000 0000
009h	EADRH	INFOS	-	-	-	EADDRH[3:0]				0--- 0000
00Ah	EADRL	EADRL[7:0]								0000 0000
00Bh	EDATH	EDATH[7:0]								xxxx xxxx
00Ch	IE0	GIE/GIEL	GIEH	INT1IE	INT0IE	KEYIE	T2IE	T1IE	T0IE	0000 0000
00Dh	B	B[7:0]								0000 0000
00Eh	GPCR2	GPCR2[7:0]								xxxx xxxx
00Fh	GPCR3	GPCR3[7:0]								xxxx xxxx
010h	IF0	-	-	INT1IF	INT0IF	KEYIF	T2IF	T1IF	T0IF	--00 0000
011h	IE1	-	-	-	-	IICIE	LVDIE	-	THIE	---- 0000
012h	IF1	THIF3	THIF2	THIF1	THIF0	IICIF	LVDIF	-	-	0000 000-
013h	IP0	IPEN	-	INT1IP	INT0IP	KEYIP	T2IP	T1IP	T0IP	0-00 0000
014h	IP1	-	-	-	-	IICIP	LVDIP	-	THIP	---- 0000
015h	INTS	-	-	-	-	INT1S[1:0]		INT0S[1:0]		---- 0000
016h	INTEN	-	-	-	-	-	-	EINT1	EINT0	---- --00
017h	WDTCON	LRCEN	WDTSEL	WDTEN	TO	PD	PS[2:0]			1011 1111
019h	P0	-	-	P05	P04	P03	P02	P01	P00	--xx xxxx
01Ah	P0OD	-	-	P0OD5	P0OD4	P0OD3	P0OD2	P0OD1	P0OD0	--00 0000
01Bh	P0PH	-	-	P0PH5	P0PH4	P0PH3	P0PH2	P0PH1	P0PH0	--11 1111
01Ch	P0PD	-	-	P0PD5	P0PD4	P0PD3	P0PD2	P0PD1	P0PD0	--00 0000
01Dh	P0OE	-	-	P0OE5	P0OE4	P0OE3	P0OE2	P0OE1	P0OE0	--11 1111
01Eh	P0K	-	-	P0K5	P0K4	P0K3	P0K2	P0K1	P0K0	--00 0000
020h	P1	P17	P16	P15	P14	P13	P12	P11	P10	xxxx xxxx
021h	P1OD	P1OD7	P1OD6	P1OD5	P1OD4	P1OD3	P1OD2	P1OD1	P1OD0	0000 0000
022h	P1PH	P1PH7	P1PH6	P1PH5	P1PH4	P1PH3	P1PH2	P1PH1	P1PH0	1111 1111
023h	P1PD	P1PD7	P1PD6	P1PD5	P1PD4	P1PD3	P1PD2	P1PD1	P1PD0	0000 0000
024h	P1OE	P1OE7	P1OE6	P1OE5	P1OE4	P1OE3	P1OE2	P1OE1	P1OE0	1111 1111
026h	P1K	P1K7	P1K6	P1K5	P1K4	P1K3	P1K2	P1K1	P1K0	0000 0000
029h	OPTION	CMODFS	CDCVS	OSCO	-	LVREN	VTHS	-	WRST	000- 10-0
02Bh	T0CON0	T0CKS	-	-	-	T0FS[2:0]			T0EN	0--- 0000
02Ch	T0C	T0CT[7:0]								0000 0000

02Dh	T0OVR	T0OVR[7:0]								xxxx xxxx
02Eh	T1CON0	-	-	-	-	T1FS[2:0]			T1EN	---- 0000
02Fh	T1OVRL	T1OVRL[7:0]								xxxx xxxx
030h	T1OVRH	-	-	-	-	-	-	T1OVERH[1:0]		---- --xx
031h	T1CL	T1CL[7:0]								0000 0000
032h	T1CH	-	-	-	-	-	-	T1CTH[1:0]		---- --00
033h	T2CON0	-	-	-	-	T2FS[2:0]			T2EN	---- 0000
034h	T2OVR	T2OVR[7:0]								xxxx xxxx
035h	T2C	T2C[7:0]								0000 0000
036h	PWMCON0	RGBEN	RGBTR0[1:0]		PWM4EN	PWM3EN	PWM2EN	PWM1EN	PWM0EN	0000 0000
037h	PWMCON1	RGBTR1[2:0]			PWM4S	PWM3S	PWM2S	PWM1S	PWM0S	0000 0000
038h	PWM0DL	PWM0DL[7:0]								xxxx xxxx
039h	PWM0DH	-	-	-	-	-	-	PWM0DTH[1:0]		---- --xx
03Ah	PWM1DL	PWM1DL[7:0]								xxxx xxxx
03Bh	PWM1DH	-	-	-	-	-	-	PWM1DTH[1:0]		---- --xx
03Ch	PWM2D	PWM2D[7:0]								xxxx xxxx
03Dh	PWM3D	PWM3D[7:0]								xxxx xxxx
03Eh	PWM4D	PWM4D[7:0]								xxxx xxxx
03Fh	LEDAT0	LEDAT07	LEDAT06	LEDAT05	LEDAT04	LEDAT03	LEDAT02	LEDAT01	LEDAT00	xxxx xxxx
040h	LEDAT1	LEDAT15	LEDAT14	LEDAT13	LEDAT12	LEDAT11	LEDAT10	LEDAT09	LEDAT08	xxxx xxxx
041h	LEDAT2	LEDAT23	LEDAT22	LEDAT21	LEDAT20	LEDAT19	LEDAT18	LEDAT17	LEDAT16	xxxx xxxx
05Ah	TLSEG0	TLDAT7	TLDAT6	TLDAT5	TLDAT4	TLDAT3	TLDAT2	TLDAT1	TLDAT0	0000 0000
05Bh	TLSEG1	-	-	-	-	-	-	TLDAT9	TLDAT8	---- --00
064h	DACCON	DACOEN	-	-	DACSEL	DACOUT[3:0]				0--0 0000
070h	LVDCON	-	-	-	LVDOUT	LVDEN	LVDSEL[2:0]			---0 0000
075h	IICCON	IICEN	STASTA	STOSTA	-	-	-	-	IICESTA	000- ---0
076h	IICDAT	IICDAT[7:0]								0000 0000
077h	IICADR	IICADDR[6:0]							IICWR	1010 0000

说明：

- ： 无效位，回读为‘0’

x ： 不定态

6.6 OPTION

表 4 配置选项 0(0h)

名称	位	默认值	说明
SUT[2:0]	[14:12]	111	PWRT & WDT计数周期选择位 (其值必须是分频率的倍数) 111：PWRT = WDT prescaler rate = 16ms (默认) 110：PWRT = 128us WDT prescaler rate = 16ms 101：PWRT = 128us WDT prescaler rate = 128ms 100：PWRT = WDT prescaler rate = 128ms 011：PWRT = WDT prescaler rate = 256ms 010：PWRT = 128us WDT prescaler rate = 8ms 001：PWRT = 128us WDT prescaler rate = 256ms 000：PWRT = WDT prescaler rate = 8ms

表 5 配置选项 1(1h)

名称	位	默认值	说明
PROTECT	[12]	1	代码保护选择位 1：代码不加密 code protection off 0：代码加密 code protection on
INT1SEL	[11]	1	INT1 输入脚选择位 1：INT1 从 P01 脚输入 0：INT1 从 P02 脚输入
OSCD[1:0]	[10:9]	11	指令周期选择位(Instructions Period Select) 11：4T，4 个 RC16M 周期 (默认) 10：2T，2 个 RC16M 周期 01：16T，16 个 RC16M 周期 00：8T，8 个 RC16M 周期
LVRSEL[6:0]	[6:0]	1101111	低电压复位点选择 1111111：3.6V 1111110：2.9V 111110x：2.7V 11110xx：2.4V（默认） 1110xxx：2.2V 110xxxx：2.0V 10xxxxx：1.8V 0xxxxxx：关闭

7. 功能模块

7.1 Clock

两个时钟源：RC16M、RC131K，系统时钟结构如下所示：

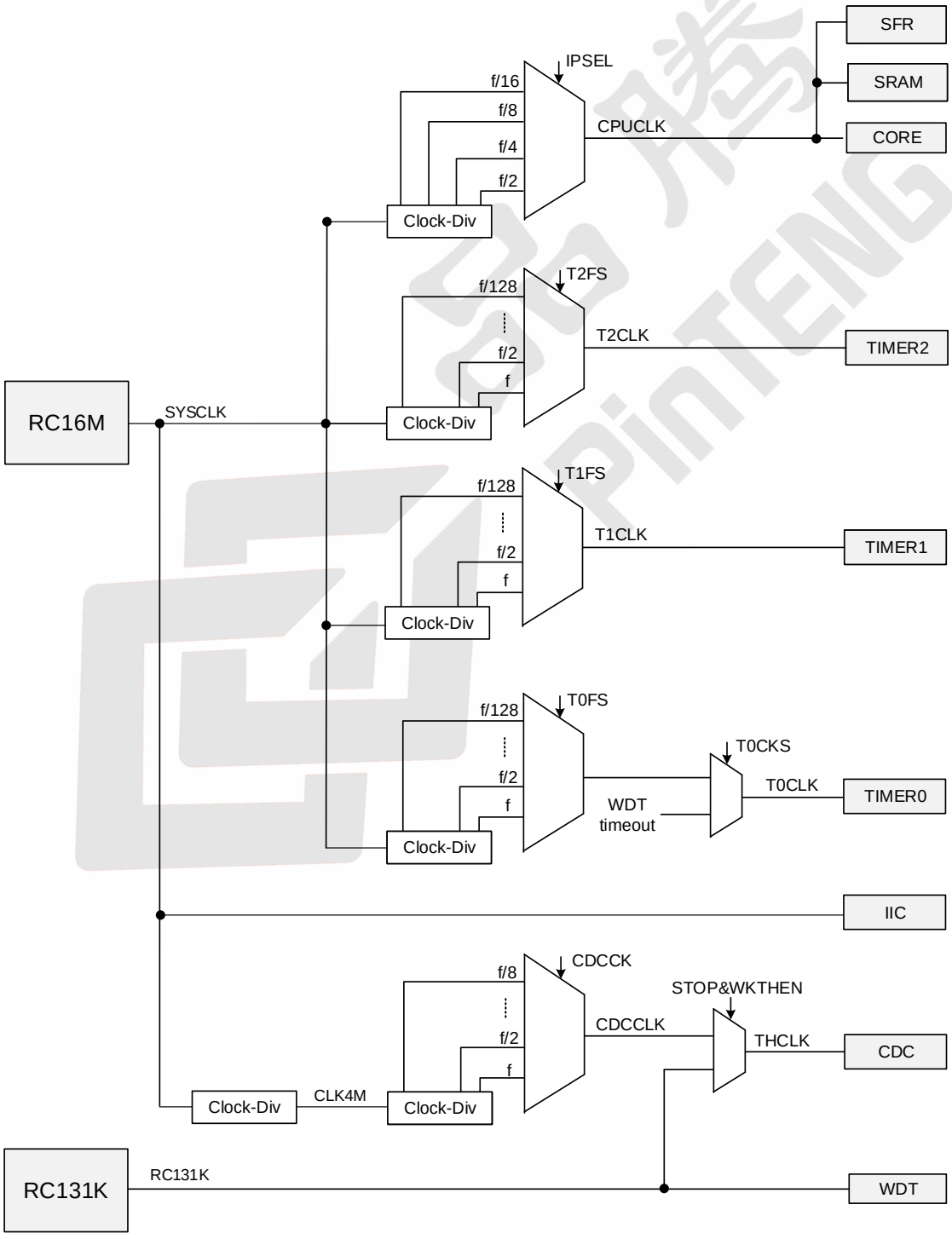


图 7 时钟结构图

7.2 Mode

支持 Normal 模式、STOP 模式、IDLE 模式，模式描述如下：

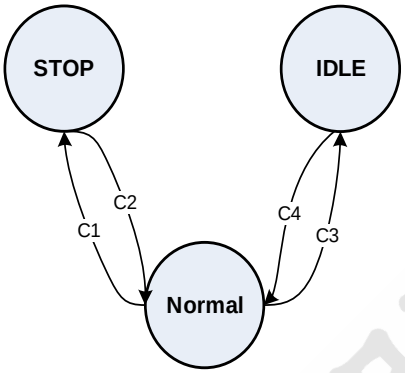


图 8 工作模式转换图

- STOP 模式下，CPU 及外设都停止工作，PD 位清零，TO 位置 1，看门狗清零同时保持运行状态，RC16M 停振，I/O 维持原状：
  - 外部中断 0 可唤醒 STOP 模式
  - 外部中断 1 可唤醒 STOP 模式
  - KEY 中断可唤醒 STOP 模式
  - TIMER0 中断可唤醒 STOP 模式（当 TIMER0 时钟源选择 WDT 溢出）
  - WDT 溢出可唤醒 STOP 模式
  - 触摸中断可唤醒 STOP 模式

- IDLE 模式下，除 CPU 外，其它外设都可工作：
  - 所有中断都可唤醒 IDLE 模式
  - WDT 溢出可唤醒 IDLE 模式

中断唤醒 STOP/IDEL 模式时，对应的中断屏蔽位需要置位。如果 GIE 为 0 则唤醒后继续执行后续指令，GIE 为 1 则唤醒后执行中断服务程序后再执行后续指令。

7.3 INT

系统有 9 种中断源:

1) 定时器 T0 溢出中断
2) 定时器 T1 溢出中断
3) 定时器 T2 溢出中断
4) INT0 中断
5) INT1 中断
6) 触摸中断
7) KEY 中断
8) LVD 中断
9) IIC 中断

系统支持的中断功能如下：

1. IF0/ IF1 为中断标志寄存器，决定该寄存器所发生的中断状态。IE0/IE1 为中断使能寄存器。IPEN 为中断优先级控制寄存器
2. 中断允许总控位 GIE/GIEL 有两个功能：当 IPEN 为低电平时，其为全局中断使能位；当 IPEN 为高电平且 GIEH 有效时，其为低优先级中断全局使能位，可开启低优先级中断。
3. 中断允许总控位 GIEH：当 IPEN 为高电平时，其为高优先级中断全局使能位
4. 支持两种中断优先级（高和低），所有中断源都可单独配置为高优先级或低优先级。允许低优先级中断服务程序嵌套高优先级中断服务程序，不允许同优先级之间的中断嵌套
5. 两个中断入口地址：008h 和 018h。当 IPEN 为低电平时，入口地址为 008h; 当 IPEN 为高电平时，默认入口地址为 008h(所有中断源默认为低优先级)，当某个中断源切换到高优先级时，中断入口地址切换到 018h
6. 进入中断服务程序前，PC、ACC 及 STATUS 会被压栈保护
7. 当中断优先级 IPEN 为低电平时，所有中断源的都为低优先级（软件错误地配置为高优先级时，硬件将强行将其变为低优先级）

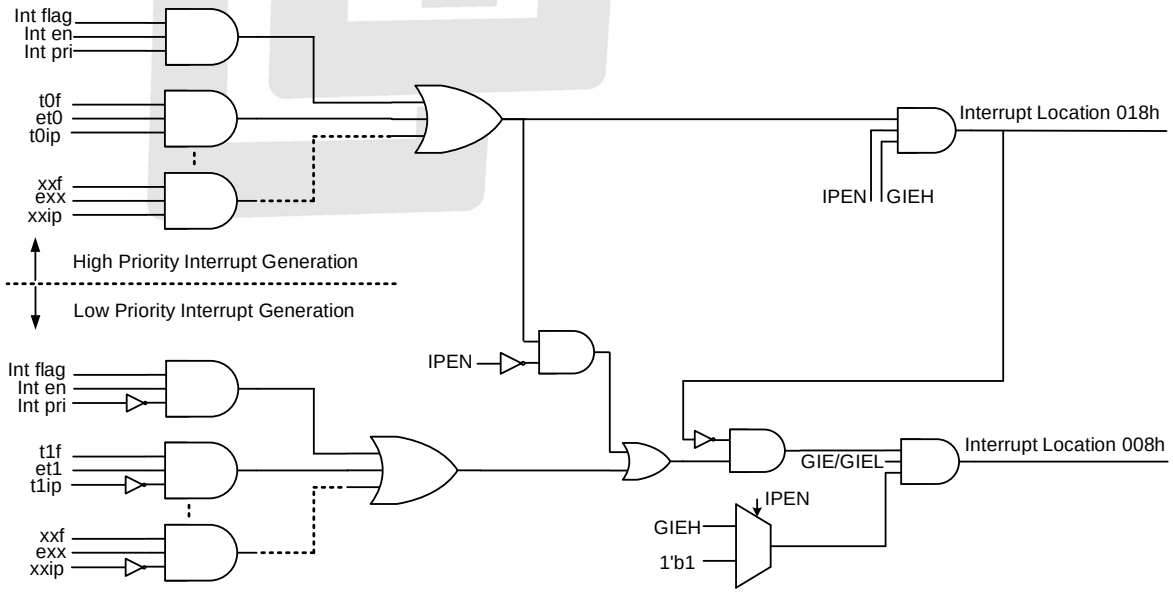


图 9 中断优先级功能图

中断模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
00Ch	IE0	GIE/GIEL	GIEH	INT1IE	INT0IE	KEYIE	T2IE	T1IE	T0IE	0000 0000
010h	IF0	-	-	INT1IF	INT0IF	KEYIF	T2IF	T1IF	T0IF	--00 0000
011h	IE1	-	-	-	-	IICIE	LVDIE	-	THIE	---- 0000
012h	IF1	THIF3	THIF2	THIF1	THIF0	IICIF	LVDIF	-	-	0000 000-
013h	IP0	IPEN	-	INT1IP	INT0IP	KEYIP	T2IP	T1IP	T0IP	0-00 0000
014h	IP1	-	-	-	-	IICIP	LVDIP	-	THIP	---- 0000
015h	INTS	-	-	-	-	INT1S1	INT1S0	INT0S1	INT0S0	---- 0000
016h	INTEN	-	-	-	-	-	-	EINT1	EINT0	---- --00

- **中断屏蔽寄存器 0(IE0，00Ch)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	GIE/GIEL	GIEH	INT1IE	INT0IE	KEYIE	T2IE	T1IE	T0IE
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	0	0	0	0	0	0	0	0

Bit[7] **GIE/GIEL**：中断允许控制位/低优先级中断允许控制位

打开中断优先级使能：

1：使能所有没有屏蔽的低优先级中断

0：禁止所有低优先级中断

关闭中断优先级使能：

1：使能所有没有屏蔽的中断

0：禁止所有中断

Bit[6] **GIEH**：高优先级中断允许控制位

1：使能所有没有屏蔽的高优先级中断

0：禁止所有高优先级中断

Bit[5] **INT1IE**：INT1 中断屏蔽位

1：使能外部中断 1

0：禁止外部中断 1

Bit[4] **INT0IE**：INT0 中断屏蔽位

1：使能外部中断 0

0：禁止外部中断 0

Bit[3] **KEYIE**：KEY 中断屏蔽位

1：使能 KEY 中断

0：禁止 KEY 中断

Bit[2] **T2IE**：TIMER2 溢出中断屏蔽位

1：使能 TIMER2 溢出中断

0：禁止 TIMER2 溢出中断

Bit[1] **T1IE**：TIMER1 溢出中断屏蔽位

1：使能 TIMER1 溢出中断

0：禁止 TIMER1 溢出中断

Bit[0] **T0IE**：TIMER0 溢出中断屏蔽位

1：使能 TIMER0 溢出中断

0：禁止 TIMER0 溢出中断

- **中断屏蔽寄存器 1(IE1，011h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	IICIE	LVDIE	-	THIE
Access	-	-	-	-	RW	RW	-	RW
Default	-	-	-	-	0	0	0	0

Bit[3] **IICIE**：IIC 中断屏蔽位

1：使能 IIC 中断

0：禁止 IIC 中断

Bit[2] **LVDIE**：LVD 中断屏蔽位

1：使能 LVD 中断

0：禁止 LVD 中断

Bit[0] **THIE**：触摸中断屏蔽位

1：使能触摸中断

0：禁止触摸中断

- **中断标志寄存器 0(IF0，010h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	INT1IF	INT0IF	KEYIF	T2IF	T1IF	T0IF
Access	-	-	RW0C	RW0C	RW0C	RW0C	RW0C	RW0C
Default	-	-	0	0	0	0	0	0

Bit[5] **INT1IF**：外部中断 1 标志，软件写 0 清，写 1 无效

Bit[4] **INT0IF**：外部中断 0 标志，软件写 0 清，写 1 无效

Bit[3] **KEYIF**：KEY 中断标志，软件写 0 清，写 1 无效

Bit[2] **T2IF**：TIMER2 溢出中断标志，软件写 0 清，写 1 无效

Bit[1] **T1IF**：TIMER1 溢出中断标志，软件写 0 清，写 1 无效

Bit[0] **T0IF**：TIMER0 溢出中断标志，软件写 0 清，写 1 无效

- **中断标志寄存器 1(IF1，012h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	THIF3	THIF2	THIF1	THIF0	IICIF	LVDIF	-	-
Access	RW0C	RW0C	RW0C	RW0C	RW0C	RW0C	-	-
Default	0	0	0	0	0	0	0	-

Bit[7] **THIF3**：CDCADR3 对应触摸通道的触摸中断标志，软件写 0 清，写 1 无效。CDCPAR 有效或 CDCAR3 设置为无效地址时，该位无效

Bit[6] **THIF2**：CDCADR2 对应触摸通道的触摸中断标志，软件写 0 清，写 1 无效。CDCPAR 有效或 CDCAR2 设置为无效地址时，该位无效

Bit[5] **THIF1**：CDCADR1 对应触摸通道的触摸中断标志，软件写 0 清，写 1 无效。CDCPAR 有效或 CDCAR1 设置为无效地址时，该位无效

Bit[4] **THIF0**：CDCADR0 对应触摸通道的触摸中断标志，软件写 0 清，写 1 无效。CDCAR0 设置为无效地址时，该位无效。当 CDCPAR 有效时，合并扫描完该位有效

Bit[3] **IICIF**：IIC 中断标志，软件写 0 清，写 1 无效

Bit[2] **LVDIF**：LVD 中断标志，软件写 0 清，写 1 无效

- **外部中断触发方式选择寄存器(INTS，015h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	INT1S1	INT1S0	INT0S1	INT0S0
Access	-	-	-	-	RW	RW	RW	RW
Default	-	-	-	-	0	0	0	0

Bit[3:2] **INT1S**：外部中断 1 触发方式选择

00：下降沿触发

01：下降沿触发

10：上升沿触发

11：下降沿或上升沿触发

Bit[1:0] **INT0S**：外部中断 0 触发方式选择

00：下降沿触发

01：下降沿触发

10：上升沿触发

11：下降沿或上升沿触发

- **外部中断触发方式选择寄存器(INTEN，016h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	EINT1	EINT0
Access	-	-	-	-	-	-	RW	RW
Default	-	-	-	-	-	-	0	0

Bit[1] **EINT1**：外部中断 1 使能选择

1：外部中断 1 使能

0：外部中断 1 禁止

Bit[0] **EINT0**：外部中断 0 使能选择

1：外部中断 0 使能

0：外部中断 0 禁止

- **中断优先级配置寄存器 0(IP0，013h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	IPEN	-	INT1IP	INT0IP	KEYIP	T2IP	T1IP	T0IP
Access	RW	-	RW	RW	RW	RW	RW	RW
Default	0	-	0	0	0	0	0	0

Bit[7] **IPEN**：中断优先级使能位

1：使能中断优先级

0：禁止中断优先级

Bit[5] **INT1IP**：外部中断 1 优先级选择位
1: 外部中断 1 为高优先级
0: 外部中断 1 为低优先级

Bit[4] **INT0IP**：外部中断 0 优先级选择位
1: 外部中断 0 为高优先级
0: 外部中断 0 为低优先级

Bit[3] **KEYIP**：按键中断优先级选择位
1: 按键中断为高优先级
0: 按键中断为低优先级

Bit[2] **T2IP**：TIMER2 中断优先级选择位
1: TIMER2 中断为高优先级
0: TIMER2 中断为低优先级

Bit[1] **T1IP**：TIMER1 中断优先级选择位
1: TIMER1 中断为高优先级
0: TIMER1 中断为低优先级

Bit[0] **T0IP**：TIMER0 中断优先级选择位
1: TIMER0 中断为高优先级
0: TIMER0 中断为低优先级

- **中断优先级配置寄存器 1(IP1，014h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	IICIP	LVDIP	-	THIP
Access	-	-	-	-	RW	RW	-	RW
Default	-	-	-	-	0	0	0	0

Bit[3] **IICIP**：IIC 中断优先级选择位
1: IIC 中断为高优先级
0: IIC 中断为低优先级

Bit[2] **LVDIP**：LVD 中断优先级选择位
1: LVD 中断为高优先级
0: LVD 中断为低优先级

Bit[0] **THIP**：触摸中断优先级选择位
1: 触摸中断为高优先级
0: 触摸中断为低优先级

■ **说明：当系统从中断子程序返回时需要将对应的中断标志位清零，只能采用如下几种方式：**

1. MOV 指令清相应的中断标志位。以清 T0IF 为例，清除方法如下所示：
   MOVK 0xFE
   MOV IF0,A
2. BCLR 指令清相应的中断标志位。以清 T0IF 为例，清除方法如下所示：
   BCLR IF0,0
3. ANDR，XORR 指令清相应中断标志位。以清 T0IF 为例，清除方法如下所示：
   MOVK 0xFE

ANDR A,IF0

7.4 GPIO

共有两组 GPIO 口，共 14 个 I/O 口，其中 P0 口为 6 脚 I/O 口，P1 口为 8 脚 I/O 口。

GPIO 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
019h	P0	-	-	P05	P04	P03	P02	P01	P00	--xx xxxx
01Ah	P0OD	-	-	P0OD5	P0OD4	P0OD3	P0OD2	P0OD1	P0OD0	--00 0000
01Bh	P0PH	-	-	P0PH5	P0PH4	P0PH3	P0PH2	P0PH1	P0PH0	--11 1111
01Ch	P0PD	-	-	P0PD5	P0PD4	P0PD3	P0PD2	P0PD1	P0PD0	--00 0000
01Dh	P0OE	-	-	P0OE5	P0OE4	P0OE3	P0OE2	P0OE1	P0OE0	--11 1111
01Eh	P0K	-	-	P0K5	P0K4	P0K3	P0K2	P0K1	P0K0	--00 0000
020h	P1	P17	P16	P15	P14	P13	P12	P11	P10	xxxx xxxx
021h	P1OD	P1OD7	P1OD6	P1OD5	P1OD4	P1OD3	P1OD2	P1OD1	P1OD0	0000 0000
022h	P1PH	P1PH7	P1PH6	P1PH5	P1PH4	P1PH3	P1PH2	P1PH1	P1PH0	1111 1111
023h	P1PD	P1PD7	P1PD6	P1PD5	P1PD4	P1PD3	P1PD2	P1PD1	P1PD0	0000 0000
024h	P1OE	P1OE7	P1OE6	P1OE5	P1OE4	P1OE3	P1OE2	P1OE1	P1OE0	1111 1111
026h	P1K	P1K7	P1K6	P1K5	P1K4	P1K3	P1K2	P1K1	P1K0	0000 0000
029h	OPTION	CMODFS	CDCVS	OSCO	-	LVREN	VTHS	-	WRST	000- 10-0

- **P0 端口读写数据寄存器(P0，019h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	P05	P04	P03	P02	P01	P00
Access	-	-	RW	RW	RW	RW	RW	RW
Default	-	-	x	x	x	x	x	x

Bit[5:0] **P0[i]**：P0[i]端口读写数据

读该寄存器：如果是做输入时，读的数据是外部输入；如果是做输出时，读的数据是 P0 寄存器值

写该端口，为输出模式时写数据从 PAD 输出

- **P1 端口读写数据寄存器(P1，020h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	P17	P16	P15	P14	P13	P12	P11	P10
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	x	x	x	x	x	x	x	x

Bit[7:0] **P1[i]**：P1[i]端口读写数据

读该寄存器：如果是做输入时，读的数据是外部输入；如果是做输出时，读的数据是 P1 寄存器值

写该端口，为输出模式时写数据从 PAD 输出

- **P0 开漏控制寄存器(P0OD，01Ah)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	P0OD5	P0OD4	P0OD3	P0OD2	P0OD1	P0OD0
Access	-	-	RW	RW	RW	RW	RW	RW

Default	-	-	0	0	0	0	0	0

Bit[5:0] **P0OD[i]**：P0[i]开漏使能

1：使能

0：禁止

- **P1 开漏控制寄存器(P1OD，021h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	P1OD7	P1OD6	P1OD5	P1OD4	P1OD3	P1OD2	P1OD1	P1OD0
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	0	0	0	0	0	0	0	0

Bit[7:0] **P1OD[i]**：P1[i]开漏使能

1：使能

0：禁止

- **P0 上拉电阻控制寄存器(P0PH，01Bh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	P0PH5	P0PH4	P0PH3	P0PH2	P0PH1	P0PH0
Access	-	-	RW	RW	RW	RW	RW	RW
Default	-	-	1	1	1	1	1	1

Bit[5:0] **P0PH[i]**：P0[i]内部上拉使能

1：禁止

0：使能

- **P1 上拉电阻控制寄存器(P1PH，022h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	P1PH7	P1PH6	P1PH5	P1PH4	P1PH3	P1PH2	P1PH1	P1PH0
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	1	1	1	1	1	1	1	1

Bit[7:0] **P1PH[i]**：P1[i]内部上拉使能

1：禁止

0：使能

- **P0 下拉电阻使能寄存器(P0PD，01Ch)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	P0PD5	P0PD4	P0PD3	P0PD2	P0PD1	P0PD0
Access	-	-	RW	RW	RW	RW	RW	RW
Default	-	-	0	0	0	0	0	0

Bit[5:0] **P0PD[i]**：P0[i]的下拉电阻使能

1：使能下拉电阻

0：禁止下拉电阻

- **P1 下拉电阻使能寄存器(P1PD，023h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	P1PD7	P1PD6	P1PD5	P1PD4	P1PD3	P1PD2	P1PD1	P1PD0
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	0	0	0	0	0	0	0	0

Bit[7:0] **P1PD[i]**：P1[i]的下拉电阻使能

1：使能下拉电阻

0：禁止下拉电阻

- **P0 I/O 方向控制寄存器(P0OE，01Dh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	P0OE5	P0OE4	P0OE3	P0OE2	P0OE1	P0OE0
Access	-	-	RW	RW	RW	RW	RW	RW
Default	-	-	1	1	1	1	1	1

P0OE 设为“1”表示该脚为输入（高阻抗），设为“0”表示该脚为输出，P0OE 寄存器可读写，系统复位以后设置为输入（高阻抗）。

- **P1 I/O 方向控制寄存器(P1OE，024h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	P1OE7	P1OE6	P1OE5	P1OE4	P1OE3	P1OE2	P1OE1	P1OE0
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	1	1	1	1	1	1	1	1

P1OE 设为“1”表示该脚为输入（高阻抗），设为“0”表示该脚为输出，P1OE 寄存器可读写，系统复位以后设置为输入（高阻抗）。

- **P0 唤醒模式使能(P0K，01Eh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	P0K5	P0K4	P0K3	P0K2	P0K1	P0K0
Access	-	-	RW	RW	RW	RW	RW	RW
Default	-	-	0	0	0	0	0	0

Bit[5:0] **P0K[i]**：P0[i]KEY 中断使能

1：使能

0：禁止

- **P1 唤醒模式使能(P1K，026h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	P1K7	P1K6	P1K5	P1K4	P1K3	P1K2	P1K1	P1K0
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	0	0	0	0	0	0	0	0

Bit[7:0] **P1K[i]**：P1[i]KEY 中断使能

1：使能

0：禁止

- **OPTION 寄存器(OPTION，029h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	CMODFS	CDCVS	OSCO	-	LVREN	VTHS	-	WRST
Access	RW	RW	RW	-	RW	RW	-	WO
Default	0	0	0	-	1	0	-	0

Bit[7] **CMODFS**：选择 CMOD 口作为模拟口还是数字口

1：CMOD 口作为数字口

0：CMOD 口作为触摸/AD 模拟口

Bit[6] **CDCVS**：CDCLDO 电压选择

1：2.0V

0：2.3V

Bit[5] **OSCO**：时钟输出使能

1：P01 口输出 1M 时钟，P00 口输出 131K 时钟

0：无时钟输出

Bit[3] **LVREN**：LVR 使能位

1：当 OPTION 中 LVRSEL 不为关闭项时开启低压复位

0：关闭 LVR

Bit[2] **VTHS**：IO 输入电压门限选择

1：0.45*VDD/0.2*VDD

0：0.7*VDD/0.3*VDD

Bit[0] **WRST**：软件复位

1：写 1 使能软件复位，复位后硬件自动清零

0：写 0 无效

说明：

1） P0/P1 有相应的上拉/下控制位(**P0PH/P0PD/P1PH/P1PD** 寄存器)来设置使能内部上拉。如果设置为输出模式，内部上拉/下拉功能会自动关闭

2） P0/P1 有相应的开漏控制位(**P0OD/P1OD** 寄存器)来设置使能开漏输出。当开漏配置有效且数据寄存器值为 1 时，即使配置为输出模式，上拉功能也可以开启。上拉下拉同时开启时只有上拉有效

3） 使能 PWM 功能时，需要把对应的 IO 口设置为输出

4） IO 口做触摸功能时，建议把对应的 IO 口设置为输出且输出低电平

5） IIC 模块使能时，把对应的 IO 口设置为输入，可外接上拉电阻也可以开启对应 IO 口的内部上拉电阻，其阻值为 5K。如果需要主机发 STA 信号唤醒 STOP，则把 P0K0 使能即可，不需使能 KEYIE 也可唤醒 STOP

6） DACO 使能输出时，建议把 P00 口设置为输出且输出低电平

7.5 TIMER

7.5.1TIMER0

TIMER0 为 8 位向上定时器，其从 T0OVR 开始计数，当其计数值达到 0xFF 后，产生 TIMER0 溢出信号。其特性如下：

1. 支持时钟预分频功能
2. 定时功能

TIMER0 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
02Bh	T0CON0	T0CKS	-	-	-	T0FS[2:0]			T0EN	0--- 0000
02Ch	T0C	T0C[7:0]								0000 0000
02Dh	T0OVR	T0OVR[7:0]								xxxx xxxx

- **TIMER0 控制寄存器 0(T0CON0，02Bh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	T0CKS	-	-	-	T0FS[2:0]			T0EN
Access	RW	-	-	-	RW			RW
Default	0	-	-	-	000			0

Bit[7] **T0CKS**：TIMER0 时钟选择

1：WDT 溢出一次 TIMER0 计数加 1（需 WDTEN 置 0）

0：系统分频时钟

Bit[3:1] **T0FS[2:0]**：TIMER0 工作于定时模式时时钟分频选择

000：不分频

001：2 分频

010：4 分频

011：8 分频

100：16 分频

101：32 分频

110：64 分频

111：128 分频

Bit[0] **T0EN**：TIMER0 使能位

1：使能 TIMER0

0：禁止 TIMER0

- **定时器计数值(T0C，02Ch)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	T0C[7:0]							
Access	RO							
Default	0x00							

Bit[7:0] **T0C**：TIMER0 计数值

- **定时器计数周期值(T0OVR，02Dh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	T0OVR[7:0]							
Access	RW							
Default	0xxx							

Bit[7:0] **T0OVR：**预设置的定时周期寄存器

实际定时周期为：0xFF- T0OVR+1（不能设置为 0xFF）

TIMER0 能被配置为普通的定时模式，当 TIMER0 被启动后，定时周期寄存器 T0OVR 的值将会被装载到定时器中，当定时器的计数器计满后将产生 TIMER0 溢出信号，同时 T0OVR 将会由硬件重新自动装载到计数器中。

TIMER0 工作于定时模式时的配置流程如下：

1. 配置时钟源选择 T0CKS 及分频选择 T0FS
2. 配置 T0OVR
3. 使能 TIMER0
4. 计满溢出后，产生溢出信号

7.5.2TIMER1

TIMER1 为 10 位向上定时器，其从 T1OVR 开始计数，当其计数值达到 0x3FF 后，产生 TIMER1 溢出信号。其特性如下：

1. 支持时钟预分频功能
2. 定时功能
3. 2 路独立的 PWM 输出

TIMER1 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
02Eh	T1CON0	-	-	-	-	T1FS[2:0]			T1EN	---- 0000
036h	PWMCON0	-	-	-	-	-	-	PWM1EN	PWM0EN	---- --00
037h	PWMCON1	-	-	-	-	-	-	PWM1S	PWM0S	---- --00
02Fh	T1OVRL	T1OVRL[7:0]								xxxx xxxx
030h	T1OVRH	-	-	-	-	-	-	T1OVERH[1:0]		---- --xx
031h	T1CL	T1CL[7:0]								0000 0000
032h	T1CH	-	-	-	-	-	-	T1CTH[1:0]		---- --00
038h	PWM0DL	PWM0DL[7:0]								xxxx xxxx
039h	PWM0DH	-	-	-	-	-	-	PWM0DTH[1:0]		---- --xx
03Ah	PWM1DL	PWM1DL[7:0]								xxxx xxxx
03Bh	PWM1DH	-	-	-	-	-	-	PWM1DTH[1:0]		---- --xx

- **TIMER1 控制寄存器 0(T1CON0，02Eh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	T1FS[2:0]			T1EN
Access	-	-	-	-	RW			RW
Default	-	-	-	-	0			0

Bit[3:1] **T1FS[2:0]**：定时器 1 时钟分频选择

000：不分频

001：2 分频

010：4 分频

011：8 分频

100：16 分频

101：32 分频

110：64 分频

111：128 分频

Bit[0] **T1EN**：TIMER1 使能位

1：使能 TIMER1

0：禁止 TIMER1

- **PWM 控制寄存器 0(PWMCON0，036h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	PWM1EN	PWM0EN

Access	-	-	-	-	-	-	RW	RW
Default	-	-	-	-	-	-	0	0

Bit[1] **PWM1EN：TIMER1** PWM1 使能位

1：使能 TIMER1 PWM1 输出

0：关闭 TIMER1 PWM1 输出

Bit[0] **PWM0EN：TIMER1** PWM0 使能位

1：使能 TIMER1 PWM0 输出

0：关闭 TIMER1 PWM0 输出

- **PWM 控制寄存器 1(PWMCON1，037h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	PWM1S	PWM0S
Access	-	-	-	-	-	-	RW	RW
Default	-	-	-	-	-	-	0	0

Bit[1] **PWM1S**：TIMER1 PWM1 输出有效电平选择位

1：先输出低电平，占空比为低电平宽度

0：先输出高电平，占空比为高电平宽度

Bit[0] **PWM0S**：TIMER1 PWM0 输出有效电平选择位

1：先输出低电平，占空比为低电平宽度

0：先输出高电平，占空比为高电平宽度

- **TIMER1 预设置的周期寄存器(T1OVRL，02Fh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	T1OVRL[7:0]							
Access	RW							
Default	0xxx							

Bit[7:0] **T1OVRL**：TIMER1 预设置的周期寄存器低 8 位

- **TIMER1 预设置的周期寄存器(T1OVRH，030h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	T1OVRH[1:0]	
Access	-	-	-	-	-	-	RW	
Default	-	-	-	-	-	-	xx	

Bit[1:0] **T1OVRH**：TIMER1 预设置的周期寄存器高 2 位

实际周期为：0x3FF – {T1OVRH, T1OVRL}+1（不能设置为 0x3FF）

- **TIMER1 计数寄存器(T1CL，031h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	T1CL[7:0]							
Access	RO							
Default	0x00							

Bit[7:0] **T1CL**：TIMER1 计数值低 8 位

- **TIMER1 计数寄存器(T1CH，032h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	T1CH[1:0]	
Access	-	-	-	-	-	-	RO	
Default	-	-	-	-	-	-	00	

Bit[1:0] **T1CH**：TIMER1 计数值高 2 位

实际计数值为: {T1CH, T1CL}

- **PWM0 占空比设置寄存器(PWM0DL，038h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	PWM0DL[7:0]							
Access	RW							
Default	0xxx							

Bit[7:0] **PWM0DL**：PWM0 占空比低 8 位

PWM0 实际占空比为：

({PWM0DH, PWM0DL}-{T1OVRH, T1OVRL}+1) / (0x3FF –{T1OVRH,T1OVRL}+1)

- **PWM1 占空比设置寄存器(PWM0DH，039h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	PWM0DH[1:0]	
Access	-	-	-	-	-	-	RW	
Default	-	-	-	-	-	-	xx	

Bit[1:0] **PWM0DH**：PWM0 占空比高 2 位。写入 PWM0DH 不会立即生效，需有写入 PWM0DL 的操作才会生效。PWM0 实际占空比为：

({PWM0DH, PWM0DL}-{T1OVRH, T1OVRL}+1) / (0x3FF –{T1OVRH, T1OVRL}+1)

- **PWM1 占空比设置寄存器(PWM1DL，03Ah)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	PWM1DL[7:0]							
Access	RW							
Default	0xxx							

Bit[7:0] **PWM1DL**：PWM1 占空比低 8 位。PWM1 实际占空比为：

({PWM1DH, PWM1DL}-{T1OVRH, T1OVRL}+1) / (0x3FF –{T1OVRH, T1OVRL}+1)

- **PWM1 占空比设置寄存器(PWM1DH，03Bh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	PWM1DH[1:0]	
Access	-	-	-	-	-	-	RW	
Default	-	-	-	-	-	-	xx	

Bit[1:0] **PWM1DH**：PWM1 占空比高 2 位。写入 PWM1DH 不会立即生效，需有写入 PWM1DL 的操作

才会生效。PWM1 实际占空比为：

({PWM1DH, PWM1DL}-{T1OVRH, T1OVRL}+1) / (0x3FF –{T1OVRH, T1OVRL}+1)

1. 定时模式

定时器 TIMER1 能被配置为普通的定时模式，当 TIMER1 被启动后，定时周期寄存器的值将会被装载到定时器中，当定时器的计数器计满后将产生 TIMER1 溢出信号，同时周期寄存器将会由硬件重新自动装载到计数器中，然后启动向上计数。

TIMER1 工作于定时模式时的配置流程如下：

1. 配置时钟源选择 T1FS
2. 配置 T1OVR
3. 使能 TIMER1
4. 计满溢出后，硬件产生 T1 溢出信号

2. PWM 模式

定时器 TIMER1 能被配置为 PWM 模式，用于 2 路独立的 PWM 输出。其有自动装载功能，即当 TIMER1 被启动后，PWM 载波周期寄存器的值及 2 路 PWM 占空比的值将会被装载到定时器 TIMER1 中，当定时器计满后，PWM 载波周期寄存器及 2 路 PWM 占空比的值将会由硬件重新自动装载到计数器中，然后启动向上计数。

TIMER1 工作于 PWM 模式时的配置流程如下：

1. 配置时钟源选择 T1FS
2. 配置 T1OVR
3. 配置 PWMxDH
4. 配置 PWMxDL
5. 配置 PWMxS
6. 使能 PWMxEN
7. 使能 TIMER1

注意：上述第 3 步和第 4 步，顺序不能调换。

周期为(0x3FF - T1OVR+1)，PWMx 高电平脉宽为 PWMxD。PWMx 波形输出如下：

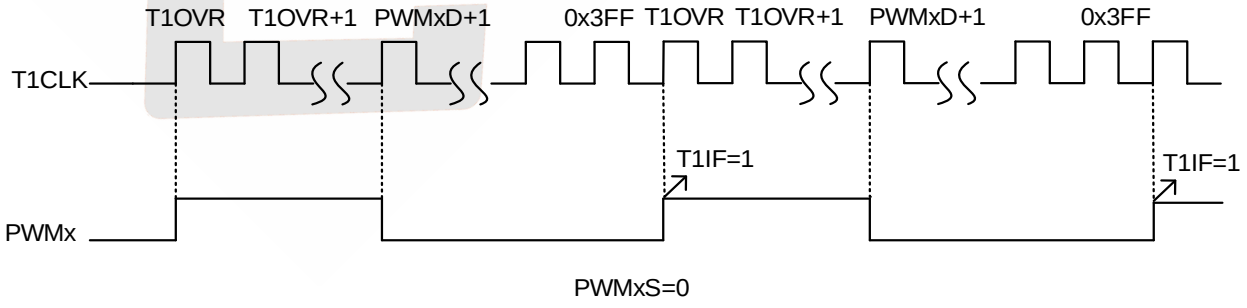


图 10 PWM 输出时序图

7.5.3TIMER2

TIMER2 为 8 位向上定时器，其从 T2OVR 开始计数，当其计数值达到 0xFF 后，产生 TIMER2 溢出信号。支持三种模式：

1. 定时模式：定时功能
2. PWM 模式：3 路独立的 PWM 输出
3. RGB 级联模式：PWM3 通道支持单线级联 LED 驱动

TIMER2 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
033h	T2CON0	-	-	-	-	T2FS[2:0]			T2EN	---- 0000
036h	PWMCON0	RGBEN	RGBTR0[1:0]		PWM4EN	PWM3EN	PWM2EN	-	-	0000 00--
037h	PWMCON1	RGBTR1[2:0]			PWM4S	PWM3S	PWM2S	-	-	0000 00--
034h	T2OVR	T2OVR[7:0]								xxxx xxxx
035h	T2C	T2C[7:0]								0000 0000
03Ch	PWM2D	PWM2D[7:0]								xxxx xxxx
03Dh	PWM3D	PWM3D[7:0]								xxxx xxxx
03Eh	PWM4D	PWM4D[7:0]								xxxx xxxx
03Fh	LEDAT0	LEDAT07	LEDAT06	LEDAT05	LEDAT04	LEDAT03	LEDAT02	LEDAT01	LEDAT0	LEDAT07
040h	LEDAT1	LEDAT15	LEDAT14	LEDAT13	LEDAT12	LEDAT11	LEDAT10	LEDAT09	LEDAT1	LEDAT15
041h	LEDAT2	LEDAT23	LEDAT22	LEDAT21	LEDAT20	LEDAT19	LEDAT18	LEDAT17	LEDAT2	LEDAT23

- **TIMER2 控制寄存器 0(T2CON0，033h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	T2FS[2:0]			T2EN
Access	-	-	-	-	RW			RW
Default	-	-	-	-	000			0

Bit[3:1] **T2FS[2:0]**：定时器 2 时钟分频选择

000：不分频

001：2 分频

010：4 分频

011：8 分频

100：16 分频

101：32 分频

110：64 分频

111：128 分频

Bit[0] **T2EN**：TIMER2 使能位

1：使能 TIMER2

0：禁止 TIMER2

- **PWM 控制寄存器 0(PWMCON0，036h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0

Name	RGBEN	RGBTR0[1:0]	PWM4EN	PWM3EN	PWM2EN	-	-
Access	RW	RW	RW	RW	RW	-	-
Default	0	00	0	0	0	-	-

Bit[7] **RGBEN**：PWM3 作为 RGB_LED 使能

1：使能 RGB_LED 级联驱动（PWM3EN 需关闭）

0：关闭 RGB_LED 级联驱动

Bit[6:5] **RGBTR0[1:0]**：PWM3 作为 RGB_LED 控制时，输出 0 的高电平时长选择

00：250ns（4xT_{RC16M}）

01：312ns（5xT_{RC16M}）

10：375ns（6xT_{RC16M}）

11：437ns（7xT_{RC16M}）

Bit[4] **PWM4EN**：PWM4 使能位

1：使能 PWM4 输出

0：关闭 PWM4 输出

Bit[3] **PWM3EN**：PWM3 使能位

1：使能 PWM3 输出

0：关闭 PWM3 输出

Bit[2] **PWM2EN**：PWM2 使能位

1：使能 PWM2 输出

0：关闭 PWM2 输出

- **PWM 控制寄存器 2(PWMS，037h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	RGBTR1[2:0]			PWM4S	PWM3S	PWM2S	-	-
Access	RW			RW	RW	RW	-	-
Default	000			0	0	0	-	-

Bit[7:5] **RGBTR1[2:0]**：PWM3 作为 RGB_LED 控制时，输出 1 的高电平时长选择

000：562ns（9xT_{RC16M}）

001：625ns（10xT_{RC16M}）

010：687ns（11xT_{RC16M}）

011：750ns（12xT_{RC16M}）

100：812ns（13xT_{RC16M}）

101：875ns（14xT_{RC16M}）

110：937ns（15xT_{RC16M}）

111：1000ns（16xT_{RC16M}）

Bit[4] **PWM4S**：PWM4 输出有效电平选择位

1：先输出低电平，占空比为低电平宽度

0：先输出高电平，占空比为高电平宽度

Bit[3] **PWM3S**：PWM3 输出有效电平选择位

1：先输出低电平，占空比为低电平宽度

0：先输出高电平，占空比为高电平宽度

Bit[2] **PWM2S**：PWM2 输出有效电平选择位

1：先输出低电平，占空比为低电平宽度

0：先输出高电平，占空比为高电平宽度

- **TIMER2 预设置的周期寄存器(T2OVR，034h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	T2OVR[7:0]							
Access	RW							
Default	0x00							

Bit[7:0] **T2OVR**：TIMER2 预设置的周期寄存器

实际定时周期为：0xFF- T2OVR+1（不能设置为 0xFF）

- **TIMER2 计数寄存器(T2C，035h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	T2C[7:0]							
Access	RO							
Default	0x00							

Bit[7:0] **T2C**：TIMER2 计数值

- **PWM2 占空比设置寄存器(PWM2D，03Ch)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	PWM2D[7:0]							
Access	RW							
Default	0xxx							

Bit[7:0] **PWM2D**：PWM2 占空比

PWM2 实际占空比为：(PWM2D-T2OVR+1) / (0xFF–T2OVR+1)

- **PWM3 占空比设置寄存器(PWM3D，03Dh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	PWM3D[7:0]							
Access	RW							
Default	0xxx							

Bit[7:0] **PWM3D**：PWM3 占空比

PWM3 实际占空比为：(PWM3D-T2OVR+1) / (0xFF–T2OVR +1)

- **PWM4 占空比设置寄存器(PWM4D，03Eh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	PWM4D[7:0]							
Access	RW							
Default	0xxx							

Bit[7:0] **PWM4D**：PWM4 占空比

PWM4 实际占空比为：(PWM4D-T2OVR+1) / (0xFF–T2OVR +1)

- **LED 驱动数据寄存器 0(LEDAT0，03Fh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	LEDAT07	LEDAT06	LEDAT05	LEDAT04	LEDAT03	LEDAT02	LEDAT01	LEDAT00
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	x	x	x	x	x	x	x	x

- **LED 驱动数据寄存器 1(LEDAT1，040h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	LEDAT15	LEDAT14	LEDAT13	LEDAT12	LEDAT11	LEDAT10	LEDAT09	LEDAT08
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	x	x	x	x	x	x	x	x

- **LED 驱动数据寄存器 2(LEDAT2，041h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	LEDAT23	LEDAT22	LEDAT21	LEDAT20	LEDAT19	LEDAT18	LEDAT17	LEDAT16
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	x	x	x	x	x	x	x	x

1. 定时模式

定时器 TIMER2 能被配置为普通的定时模式，当 TIMER2 被启动后，定时周期寄存器的值将会被装载到定时器中，当定时器的计数器计满后将产生 TIMER2 溢出信号，同时周期寄存器将会由硬件重新自动装载到计数器中，然后启动向上计数。

TIMER2 工作于定时模式时的配置流程如下：

1. 配置时钟源选择 T2FS
2. 配置 T2OVR
3. 使能 TIMER2
4. 计满溢出后，硬件产生 T2 溢出信号

2. PWM 模式

定时器 TIMER2 能被配置为 PWM 模式，用于 3 路独立的 PWM 输出。其有自动装载功能，即当 TIMER2 被启动后，PWM 载波周期寄存器的值及 3 路 PWM 占空比的值将会被装载到定时器 TIMER2 中，当定时器计满后，PWM 载波周期寄存器及 3 路 PWM 占空比的值将会由硬件重新自动装载到计数器中，然后启动向上计数。

TIMER2 工作于 PWM 模式时的配置流程如下：

1. 配置时钟源选择 T2FS
2. 配置 T2OVR ，PWMxD
3. 配置 PWMxS
4. 使能 PWMxEN
5. 使能 TIMER2

周期为(0xFF–T2OVR+1)，PWMx 高电平脉宽为 PWMxD。PWMx 波形输出如下：

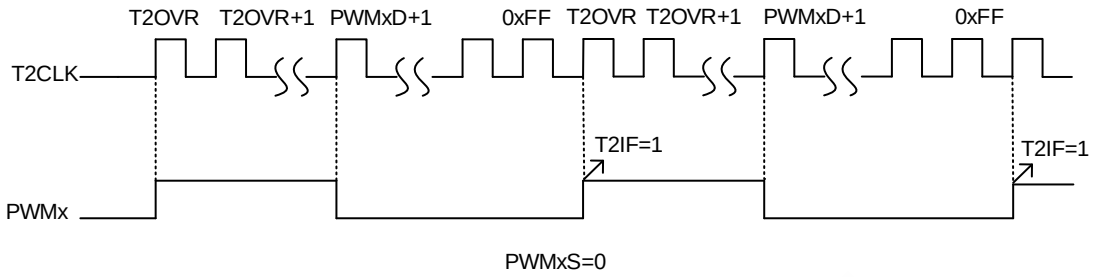


图 11 PWM 输出时序图

3. RGB 级联模式

PWM3 通道支持单线级联 LED 驱动，从 PWM3 口串行输出 RGB 驱动信号。级联 LED 的典型驱动时序图及位码如下图所示。

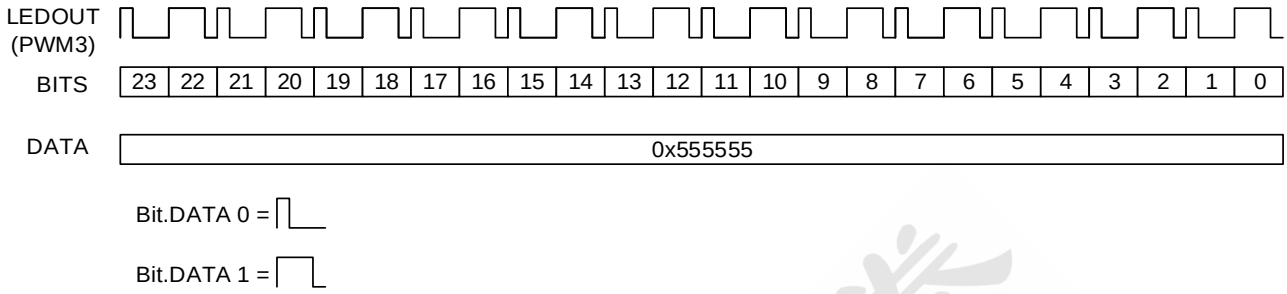


图 12 级联 LED 时序图

WS2812 等类似芯片在上电复位或者输入端接受低电平达到复位时间后，其输入端接受到第一组 24 位数据会送到内部锁存器保存。随后输入端接受的数据经过内部整形电路整形放大，从输出端输出给下一个级联点，每经过一个级联点传输，数据信号减少 24 位。级联点保存的 24 位数据分为高中低三组各 8 位，分别控制 RGB 三色的发光强度。

注意：输入端接受低电平达到复位时间，只会让级联点重新接受保存第一组 24 位数据，内部锁存器数据并不会复位为 0，而是保持不变直到接受到第一组 24 位数据。

在级联 LED 时序图中，位码 0(DATA0)的高电平时间宽度由 RGBTR0 配置，位码 1(DATA1)的高电平时间宽度由 RGBTR1 配置，位周期固定为 20 个 RC16M 时钟周期。当 RGBEN 为 1 时，级联 LED 驱动使能，LEDAT0/1/2 为 LED 的数据寄存器(LEDAT)，共 24 位，其按照 MSB 到 LSB 顺序发送(发放顺序为：LEDAT2[7], LEDAT2[6],..., LEDAT2[0], LEDAT1[7], LEDAT1[6],...,LEDAT1[0], LEDAT0[7], LEDAT0[6],..., LEDAT0[0])。当 RGBEN 使能后，LEDAT2 写入数据后开始发送(软件先写 LEDAT0，再写 LEDAT1，最后写 LEDAT2)。当 RGBEN 关闭时，写 LEDAT0/1/2 寄存器无操作。

RGB_LED 级联驱动模式下：

1) T2FS 硬件固定为 3'b000，软件配置无效（TIMER2 采用 RC16M 时钟）
2) T2OVR 硬件固定为 8'd236，软件配置无效（PWM 周期固定为 20 个 RC16M 时钟周期）
3) PWM3S 硬件固定为 0，软件配置无效(PWM3 输出先高后低)
4) PWM3D 软件配置无效，由 RGBTR0（位码 0）和 RGBTR1（位码 1）配置
5) 发完 LEDAT 的 24 位数据后，才会产生 TIMER2 中断

TIMER2 工作于 RGB 模式时的配置流程如下：

1. 配置位 0 高电平宽度 RGBTR0、位 1 高电平宽度 RGBTR1
2. 使能 RGBEN
3. 使能 TIMER2
4. 写入 LEDAT0
5. 写入 LEDAT1
6. 写入 LEDAT2
7. 发生 TIMER2 中断，表示发送完成
8. 循环上述 4~7

7.6 LVD

提供 8 档低电压检测（LVD）功能，当 LVD 检测有效，LVDOUT 标识将会自动置位为 1。

LVD 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
070h	LVDCON	-	-	-	LVDOUT	LVDEN	LVDSEL[2:0]			---0 0000

- **LVD 控制寄存器(LVDCON，070h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	LVDOUT	LVDEN	LVDSEL[2:0]		
Access	-	-	-	RO	RW	RW		
Default	-	-	-	0	0	000		

Bit[4] **LVDOUT**：低压检测标志位

1：低压检测有效

0：低压检测无效

Bit[3] **LVDEN**：低电压检测使能位

1：使能 LVD

0：关闭 LVD

Bit[2:0] **LVDSEL[2:0]**：低电压检测点选择

000：2.2V

001：2.5V

010：2.7V

011：3.0V

100：3.3V

101：3.6V

110：3.9V

111：4.5V

7.7 TOUCH

触摸模块(TOUCH)采用电容数字转换器（Cap Digital Convertor，CDC）结构，把触摸通道的电容值转换为计数值，程序读取其数值后进行算法处理，实现稳定可靠的触摸按键检测，最多支持 10 个通道。

- **LED 显示使能寄存器(TLSEG0，05Ah)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	TLDAT7	TLDAT6	TLDAT5	TLDAT4	TLDAT3	TLDAT2	TLDAT1	TLDAT0
Access	RW	RW	RW	RW	RW	RW	RW	RW
Default	0	0	0	0	0	0	0	0

Bit[7] **TLDAT7**：TLDAT7=1，TCH7 对应的 IO 口作为 LED 显示使能

Bit[6] **TLDAT6**：TLDAT6=1，TCH6 对应的 IO 口作为 LED 显示使能

Bit[5] **TLDAT5**：TLDAT5=1，TCH5 对应的 IO 口作为 LED 显示使能

Bit[4] **TLDAT4**：TLDAT4=1，TCH4 对应的 IO 口作为 LED 显示使能

Bit[3] **TLDAT3**：TLDAT3=1，TCH3 对应的 IO 口作为 LED 显示使能

Bit[2] **TLDAT2**：TLDAT2=1，TCH2 对应的 IO 口作为 LED 显示使能

Bit[1] **TLDAT1**：TLDAT1=1，TCH1 对应的 IO 口作为 LED 显示使能

Bit[0] **TLDAT0**：TLDAT0=1，TCH0 对应的 IO 口作为 LED 显示使能

- **LED 显示使能寄存器(TLSEG1，05Bh)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	-	-	-	-	-	-	TLDAT9	TLDAT8
Access	-	-	-	-	-	-	RW	RW
Default	-	-	-	-	-	-	0	0

Bit[1] **TLDAT9**：TLDAT9=1，TCH9 对应的 IO 口作为 LED 显示使能

Bit[0] **TLDAT8**：TLDAT8=1，TCH8 对应的 IO 口作为 LED 显示使能

- **DAC 控制寄存器(DACCON，064h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	DACOEN	-	-	DACSEL	DACOUT[3:0]			
Access	RW	-	-	RW	RW			
Default	0	-	-	0	0000			

Bit[7] **DACOEN：**

0：关闭 DAC 并且 P00 作为普通 IO 口

1：使能 DAC 并且 P00 作为 DAC 输出口

Bit[4] **DACSEL：**VDAC 电压选择

0：V_{DAC}=VDD

1：V_{DAC}=V_{LDO}

说明：V_{LDO}为2.3V或2.0V

Bit[3:0] **DACOUT[3:0]：**DAC 输出电压选择

0000：0V

0001：1/10V_{DAC}

0010：2/10V_{DAC}

0011：3/10V_{DAC}

0100：4/10V_{DAC}

0101：5/10V_{DAC}

0110：6/10V_{DAC}

0111：7/10V_{DAC}

1000：8/10V_{DAC}

1001：9/10V_{DAC}

1010：V_{DAC}

其他：V_{DAC}

7.8 Mini I2C

Mini I2C 模块支持芯片与外围 IIC 器件以标准 IIC 协议进行串行数据传输，仅支持从机模式，最高支持 400Kbits/s。

Mini I2C 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
075h	IICCON	IICEN	STASTA	STOSTA	-	-	-	-	IICESTA	000- ---0
076h	IICDAT	IICDAT[7:0]								0000 0000
077h	IICADR	IICADR[6:0]							IICWR	1010 0000

- **IIC 控制寄存器(IICCON，075h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	IICEN	STASTA	STOSTA	-	-	-	-	IICESTA
Access	RW	RO	RO	-	-	-	-	RO
Default	0	0	0	-	-	-	-	0

Bit[7] **IICEN:** IIC 使能位

1：使能 IIC

0：禁止 IIC

Bit[6] **STASTA:**

1：已经接收到 START 位，进入数据传输阶段，硬件自动清零

0：没有接收到 START 位

Bit[5] **STOSTA:**

1：已经接收到 STOP 位，接收到 START 位后硬件自动清零

0：没有接收到 STOP 位

Bit[0] **IICESTA:**

1：接收到主机回 NACK

0：无错误

- **IIC 数据寄存器(IICDAT，076h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	IICDAT[7:0]							
Access	RW							
Default	0x00							

Bit[7:0] **IICDAT:** 该寄存器寻址两个寄存器，一个发送移位寄存器和一个接收锁存寄存器 IICDAT 的写入将发送字节装载到移位寄存器中，IICDAT 的读取将返回接收锁存器中的内容

- **IIC 地址寄存器(IICADR，077h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	IICADR[6:0]							IICWR
Access	RW							RO
Default	1010000							0

Bit[7:1] **IICADR[6:0]:** 设定 IIC 从机地址

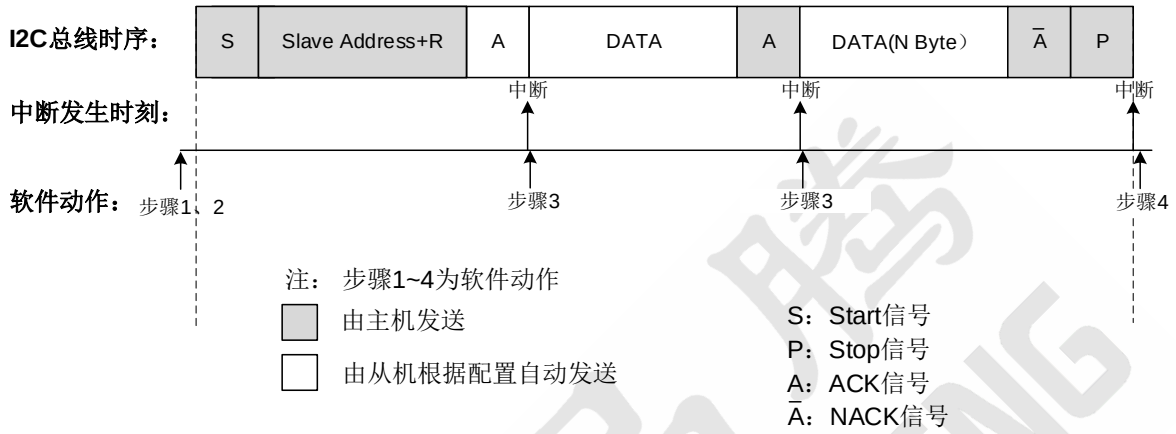
Bit[0] **IICWR:**

1：发送数据

0：接收数据

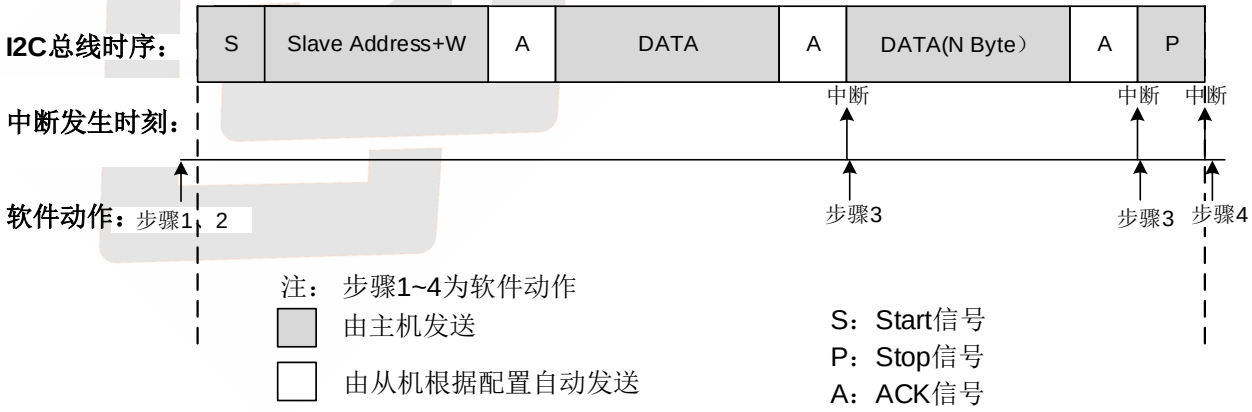
IIC 通讯流程：

1. 发送数据流程



1) IIC 初始化，配置 IE1 中的 IICIE 位
2) 配置 IICADR 从机地址，配置 IICEN 位使能 IIC
3) 数据发送
   a) 若开启中断使能，则进中断，写入 IICDAT（第一次进中断需判断 IICWR）
   b) 若未开启中断使能，查询到中断标志位后，清除标志位，写入 IICDAT（第一次查询到中断标志位需判断 IICWR）
4) 待 IICCON 中的 STOSTA 为"1"时，代表已经接收到主机发出的 STOP 信号，数据传输结束

2. 接收数据流程



1) IIC 初始化，配置 IE1 中的 IICIE 位
2) 配置 IICADR 从机地址，配置 IICEN 位使能 IIC
3) 数据接收
   a) 若开启中断使能，则进中断，读取 IICDAT（第一次进中断需判断 IICWR）
   b) 若未开启中断使能，查询到中断标志位后，清除标志位，读取 IICDAT（第一次查询到中断标志位需判断 IICWR）
4) 待 IICCON 中的 STOSTA 为"1"时，代表已经接收到主机发出的 STOP 信号，数据传输结束

7.9 WDT

看门狗定时器（WDT）无需任何外部电路即可工作，睡眠模式同样工作，启用看门狗超时发生后 MCU 重启复位。

在一般操作或睡眠模式情况下，看门狗定时器溢出会导致 MCU 复位同时 TO 位被清零。

看门狗溢出时间可以通过配置选项 0 的 SUT[1:0]设置为 16ms、8ms、256ms、128ms，溢出周期变长可以通过设置 PS 使看门狗定时器分频最大达到 1:128，溢出周期最大为 36.8 秒。

STOP 下的 WDT 溢出复位会导致芯片复位，在进睡眠之前还没完成的操作不会再继续。

CLRWDT 指令和 STOP 指令能使 WDT 溢出时间和预置器清零。

WDT 模块相关寄存器：

地址	名称	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	复位值
017h	WDTCON	LRCEN	WDTSEL	WDTEN	TO	PD	PS[2:0]			1011 1111

- **WDT 控制寄存器(WDTCON，017h)：**

BIT	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Name	LRCEN	WDTSEL	WDTEN	TO	PD	PS[2:0]		
Access	RW	RW	RW	RO	RO	RW		
Default	1	0	1	1	1	111		

Bit[7] **LRCEN:** 内部 RC131K 时钟使能位

1：使能 RC131K 时钟

0：禁止 RC131K 时钟

Bit[6] **WDTSEL**：WDT 功能选择位

1：WDT 溢出唤醒，可以唤醒 IDLE 或 STOP 模式

0：WDT 溢出复位

Bit[5] **WDTEN:** WDT 使能位

1：使能 WDT

0：禁止 WDT

Bit[4] **TO**：时间溢出标志

1：当系统上电时或执行“CLRWDT”或“STOP”指令后

0：看门狗定时器溢出

Bit[3] **PD**：Powerdown flag bit

1：当系统上电时或执行“CLRWDT”指令后

0：当执行“STOP”指令后

Bit[2:0] **PS[2:0]**：WDT 时钟分频选择控制位

000：WDT Rate=1:1

001：WDT Rate=1:2

010：WDT Rate=1:4

011：WDT Rate=1:8

100：WDT Rate=1:16

101：WDT Rate=1:32

110：WDT Rate=1:64

111：WDT Rate=1:128

7.10 Reset

有 4 种复位方式：

1. 上电复位(POR)：上电过程中电压低于某一电压值使芯片保持复位，直到高于 POR 门限
2. 低压复位(LVR)：当检测到电压低于某一电压值时会对芯片复位，保证芯片只在正常电压范围内工作
3. 看门狗 WDT 溢出复位：看门狗超时后 MCU 重启复位
4. 软件复位（WRST）：对 OPTION 寄存器中 WRST 写 1 产生软件复位

根据不同的复位方式硬件对 TO 和 PD 位置 1 或清零。

TO、PD 状态位影响事件

No.	事件	TO	PD
1	Power-on	**1**	**1**
2	WDT Time-Out	**0**	**u**
3	STOP instruction	**1**	**0**
4	CLRWDT instruction	**1**	**1**

复位时序:

1. 复位锁存器置 1，PWRT&OST 清零
2. 当内部的 POR、LVR、WDT 溢出复位脉冲加载完成后，PWRT 开始计数
3. PWRT 溢出以后，OST 开始计数延迟
4. OST 延迟完成以后，复位锁存器清零芯片最后得到的复位信号

PWRT(Power-up Reset TIMER)：上电复位计数器

上电复位计数器延迟时间由 SUT[2:0]设置，只要 PWRT 在运行，IC 就一直保持在复位状态。VDD、温度和其它变化而会影响 PWRT 控制 IC 的延迟时间。

上电复位时间：

Oscillator Mode	POR/LVR	WDT time-out Reset/WRST
RC	16ms /8ms /256 ms /128ms/128us	42us

OST(Oscillator Start-up TIMER)：振荡启动计数器

在上电过程中，PWRT 延迟时间之后振荡启动计数器会再提供一个 64 个周期的延迟以使内部 RC16M 时钟稳定，当 OST 计数时，IC 保持为复位状态，直到 OST 计满后复位撤除。另外 RC16M 信号振幅需达到振荡器输入最大振幅后，OST 计数器才开始计数。

8. 电气特性

8.1 极限参数

表 6 极限参数

参数	标号	条件	范围	单位
供电电压	VDD	-	-0 to +6.0	V
输入电压	V_I	所有 I/O口	-0.3 to VDD + 0.3	V
工作温度	T_A	-	-40 to + 85	℃
储藏温度	T_{STG}	-	-50 to + 125	℃

8.2 直流特性

表 7 如无特殊说明 VDD = 2.5V~5.5V，Temp = 25°C

参数	标号	条件	最小值	典型值	最大值	单位
工作电压	VDD	F_{CPU}=2MHz	1.8	5	5.5	V
		F_{CPU}=4MHz	2.2	5	5.5	V
		F_{CPU}=8MHz	3.0	5	5.5	V
触摸工作电压范围	VDD	F_{CPU}=4MHz	2.3	5	5.5	V
ADC 工作电压范围	VDD	F_{CPU}=4MHz	2.7	5	5.5	V
输入高电压	V_{IH}	VTHS=0	0.7*VDD			V
输入低电压	V_{IL}				0.3*VDD	V
输入高电压	V_{IH}	VTHS=1	0.5*VDD			V
输入低电压	V_{IL}				0.2*VDD	V
输出拉电流	I_{OH1}	VDD=5V, V_{OH}=0.9VDD		15		mA
输出灌电流	I_{OL}	VDD=5V, V_{OL}=0.1VDD		30		mA
输入上拉电阻	R_{PH}			30		KΩ
输入下拉电阻	R_{PL}			30		KΩ
WDT 电流		VDD=3V		0.5		uA
		VDD=5V		1.5		
WDT 周期	T_{WDT}	VDD =2.5V~5.5V	14		16.4	ms
LVR 电压	V_{LVR}		-0.1		+0.1	V
LVR 电流	I_{LVR}	VDD=5V		125		uA
LVD 电压	V_{LVD}		-0.1		+0.1	V
LVD 电流	I_{LVD}	VDD=5V		35		uA
LVR 迟滞				50	60	mV
POR 电压	V_{POR}		-0.1	1.5	+0.3	V
POR 电流	I_{POR}	VDD =3V		0.3		uA
		VDD =5V		0.35		uA
正常工作模式电流	I_{NM}	Normal mode，ADC enable VDD=5V，F_{CPU} =4M		3.0		mA
		Normal mode，ADC disable VDD=5V，F_{CPU} =4M		2.2		

		Normal mode，ADC enable VDD=3V，F_{CPU} =4M		1.65		
		Normal mode，ADC disable VDD=3V，F_{CPU} =4M		1		
		Normal mode，ADC enable VDD=5V，F_{CPU} =8M		3.4		
		Normal mode，ADC disable VDD=5V，F_{CPU} =8M		2.75		
		Normal mode，ADC enable VDD=3V，F_{CPU} =8M		1.9		
		Normal mode，ADC disable VDD=3V，F_{CPU} =8M		1.3		
低功耗模式电流	I_{SM}	STOP mode，WDT enable VDD=5V，F_{CPU} =4M，LVROFF (128ms 溢出唤醒)		3		uA
		STOP mode，WDT disable VDD=5V，F_{CPU} =4M，LVROFF		0.6		
		STOP mode，WDT enable VDD=3V，F_{CPU} =4M，LVROFF (128ms 溢出唤醒)		2.3		
		STOP mode，WDT disable VDD=3V，F_{CPU} =4M，LVROFF		0.4		

8.3 振荡器特性

表 8 如无特殊说明 VDD =5.0V，TA = 25ºC，VDD 滤波电容 0.1 uF

参数	标号	条件	最小值	典型值	最大值	单位
高速时钟	f_{HIRC}			16		MHz
高速时钟精度		VDD =5.0V/@25 ºC	-0.5	–	0.5	%
		VDD =2.5V~5.5V/@25 ºC	-3	–	+3	%
		VDD =2.5V~5.5V/-40~85 ºC	-5	–	+5	%
低速时钟	f_{LIRC}			131		KHz
低速时钟精度		VDD =5.0V/@25 ºC	-5	–	+5	%
		VDD =1.8V~5.5V/@25 ºC	-5	–	+5	%
		VDD =1.8V~5.5V/-40~85 ºC	-10	–	+10	%

9. 封装尺寸

9.1 SOP16封装

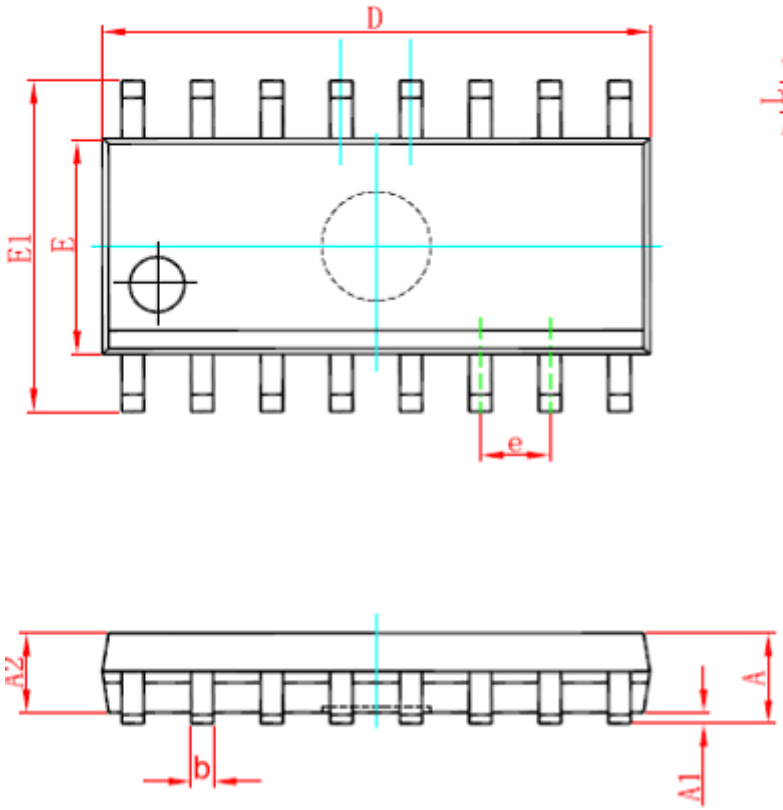


图 13 SOP16 封装图

表 9 SOP16 封装尺寸

符号	尺寸（mm 单位）			符号	尺寸（Inches 单位）		
	最小值	典型值	最大值		最小值	典型值	最大值
A	1.35	-	1.75	**A**	0.053	-	0.069
A1	0.10	-	0.25	**A1**	0.004	-	0.010
A2	1.35	-	1.55	**A2**	0.053	-	0.061
b	0.33	-	0.51	**b**	0.013	-	0.020
c	0.17	-	0.25	**c**	0.007	-	0.010
D	9.80	-	10.2	**D**	0.386	-	0.402
E	3.80	-	4.00	**E**	0.150	-	0.157
E1	5.80	-	6.20	**E1**	0.228	-	0.244
e	-	1.270	-	**e**	-	0.050	-
L	0.40	-	1.27	**L**	0.016	-	-
θ	0°	-	8°	**θ**	0°	-	8°

9.2 QFN3*3-16封装

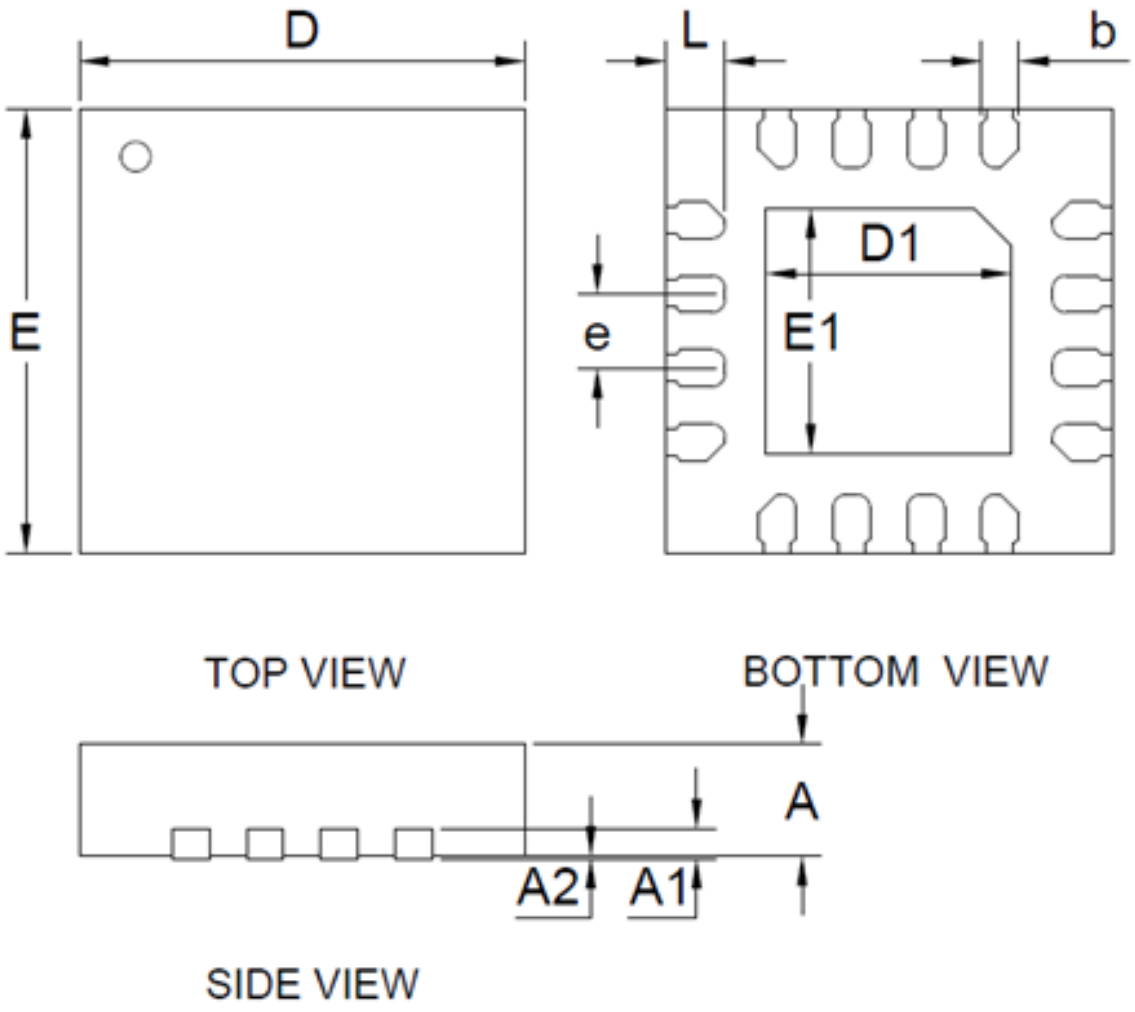


图 14 QFN3*3-16 封装图

表 10 QFN3*3-16 封装尺寸

符号	尺寸（mm 单位）		
	最小值	典型值	最大值
A	0.70	0.75	0.80
A1	0.203REF		
A2	0.00	0.02	0.05
D	2.95	3.00	3.05
E	2.95	3.00	3.05
D1	1.60	1.65	1.70
E1	1.60	1.65	1.70
b	0.20	0.25	0.30
e	0.50BSC		
L	0.35	0.40	0.45

10. 历史记录

版本号	修改记录	发布日期
V1.0	初版	2025-12-29
V1.1	更新MINI I2C模块描述笔误	2026-03-10
V1.2	添加DACCON寄存器关于V_{LDO}电压值的描述	2026-04-03
V1.3	添加TLSEG0/1寄存器描述	2026-04-27
V1.4	4MHZ最低工作电压由2.0V改为2.2V	2026-07-20
V1.5	删除XSOP12L封装	2026-09-21
V1.6	1、更新QFN3*3-16封装尺寸 2、修改指令集笔误	2026-10-08

最终版本以官网为准，请及时下载查阅！