



# MC30P8160 用户手册

---

基于 RISC MC30 内核的 8 位触摸按键控制芯片

*Rev1.2.0*

本资料内容为晟矽微电子（以下简称“我公司”）版权所有。

我公司将力求本资料的内容做到准确无误，但同时保留在不通知用户的情况下，对本资料内容的修改权。如您需要获得最新的资料，请及时联系我公司。

我公司将尽最大努力为您提供高品质、高稳定性的产品。尽管如此，由于一般半导体器件的电气敏感性及其易受到外部物理损伤等固有特性存在，所以难免造成半导体器件出现故障或失效的可能。当您使用我公司产品时，有责任按照本资料以及相关资料中提到的“规则”来设计一个稳定及安全可靠的系统环境。另外，在遇到超规格（本资料中未描述到内容）的使用，请您提前咨询我公司，以免因我公司产品在一些特殊设备中或者特殊环境下的使用，导致财产损失、人员伤亡等严重后果的发生。因超规格的使用、未经咨询授权我公司产品的使用，我公司对此不承担任何法律责任。



目录

- 1 产品概要 ..... 4
  - 1.1 产品特性 ..... 4
  - 1.2 订购信息 ..... 5
  - 1.3 引脚排列 ..... 5
  - 1.4 端口说明 ..... 6
- 2 电气特性 ..... 8
  - 2.1 极限参数 ..... 8
  - 2.2 直流电气特性 ..... 8
  - 2.3 交流电气特性 ..... 9
  - 2.4 CMP 特性参数 ..... 9
- 3 CPU 与存储器 ..... 11
  - 3.1 指令集 ..... 11
  - 3.2 程序存储器 ..... 13
  - 3.3 数据存储单元 ..... 13
  - 3.4 堆栈 ..... 14
  - 3.5 控制寄存器 ..... 14
  - 3.6 用户配置字 ..... 17
- 4 时钟 ..... 18
  - 4.1 内部高频 RC 振荡器 ..... 18
  - 4.2 内部低频 RC 振荡器 ..... 18
  - 4.3 系统工作模式 ..... 19
  - 4.4 低功耗模式 ..... 19
- 5 复位 ..... 21
  - 5.1 复位条件 ..... 21
  - 5.2 上电复位 ..... 22
  - 5.3 外部复位 ..... 22
  - 5.4 低电压复位 ..... 22
  - 5.5 看门狗复位 ..... 23
- 6 I/O 端口 ..... 24
  - 6.1 通用 I/O 功能 ..... 24
  - 6.2 内部上/下拉电阻 ..... 24
  - 6.3 端口模式控制 ..... 25



---

7	定时器 TIMER .....	27
7.1	看门狗定时器 WDT .....	27
7.2	定时器 T0.....	28
7.3	定时器 T1.....	31
7.4	定时器 T2.....	37
8	比较器 CMP 和电压检测 EVD .....	41
8.1	CMP 概述.....	41
8.2	CMP 相关寄存器 .....	42
9	触摸按键模块 TKM .....	45
9.1	TKM 概述.....	45
9.2	TKM 相关寄存器 .....	45
9.3	TKM 操作步骤.....	49
10	中断 .....	50
10.1	外部中断.....	50
10.2	定时器中断 .....	50
10.3	键盘中断.....	51
10.4	TKM 计数中断.....	51
10.5	CMP 中断.....	51
10.6	中断相关寄存器 .....	51
11	特性曲线.....	54
11.1	I/O 特性 .....	54
11.2	功耗特性 .....	58
11.3	模拟电路特性.....	62
12	封装尺寸 .....	64
12.1	SOP8 .....	64
12.2	SOT23-6 .....	64
13	修订记录.....	65



## 1 产品概要

### 1.1 产品特性

- 8 位 CPU 内核
  - ◇ 精简指令集，5 级深度硬件堆栈
  - ◇ CPU 为单时钟，仅在系统主时钟下运行
  - ◇ 系统主时钟下  $F_{CPU}$  固定为  $F_{OSC}$  的 2 分频
- 程序存储器
  - ◇ 1K×14 位 OTP 型程序存储器
- 数据存储器
  - ◇ 64 字节 SRAM 型通用数据存储器，支持直接寻址、间接寻址等多种寻址方式
- 1 组共 6 个 I/O
  - ◇ P1 (P10~P15)
  - ◇ 所有端口均支持施密特输入，均可选推挽或开漏输出
  - ◇ P14 可复用为外部复位 RST 输入，编程时为高压 VPP 输入
  - ◇ 所有端口均内置上拉和下拉电阻，均可单独使能
  - ◇ P14 可复用为外部中断输入，支持外部中断唤醒功能
  - ◇ 所有端口均支持键盘中断唤醒功能，并可单独使能
- 时钟源
  - ◇ 内置高频 RC 振荡器 (16MHz)，其 1/2/4/8/16/32 分频时钟，可用作系统主时钟源
  - ◇ 内置低频 RC 振荡器 (32KHz)，可用作系统主时钟源、或外设低频时钟源
- 系统工作模式
  - ◇ 运行模式：CPU 在系统主时钟下运行
  - ◇ 休眠模式（低功耗模式）：CPU 暂停，系统主时钟源停止
- 内部自振式看门狗计数器 (WDT)
  - ◇ 与定时器 T0 共用预分频器
  - ◇ 溢出时间可配置：4.5ms/18ms/72ms/288ms（无预分频）
  - ◇ 工作模式可配置：始终开启、始终关闭，也可软件控制开启或关闭
- 3 个定时器
  - ◇ 8 位定时器 T0，支持外设低频时钟，可实现外部计数功能，与 WDT 共用预分频器
  - ◇ 8 位定时器 T1，可实现外部计数、4 路共周期独立占空比的 PWM（可组合成 2 对互反的带死区互补 PWM）
  - ◇ 16 位定时器 T2，支持单次计时模式可用于触摸按键检测



- 1 个模拟比较器 CMP
  - ◇ 输入共模 0 ~ (VDD-1.4V)，输出可选有/无回滞（单边 6%）
  - ◇ 正端输入可选择外部输入电压、或内部基准电压  $V_{IR}$  (0.5V)
  - ◇ 负端输入可选择外部输入电压、或外部输入电压/VDD 的内部分压电压
  - ◇ 输出端电平可选择上升沿或下降沿触发中断，可从端口输出且支持输出取反
  - ◇ 可实现 VDD 或外部输入的电压检测（EVD）功能
- 1 个触摸按键模块 TKM
  - ◇ 4 路触摸按键输入（TKN0~TKN3）
- 中断
  - ◇ 外部中断（INT），键盘中断（P10~P15）
  - ◇ 定时器中断（T0~T2），TKM 计数中断
  - ◇ CMP 中断
- 低电压复位 LVR
  - ◇ 关闭/1.8V/2.0V/2.7V/3.6V
- 工作电压
  - ◇  $V_{LVR36} \sim 5.5V @ F_{cpu} = 0\sim 8MHz$
  - ◇  $V_{LVR20} \sim 5.5V @ F_{cpu} = 0\sim 4MHz$
  - ◇  $V_{LVR18} \sim 5.5V @ F_{cpu} = 0\sim 1MHz$
  - ◇  $V_{POR} \sim 5.5V @ F_{cpu} = 0\sim 32KHz/2$
- 封装形式
  - ◇ SOP8/SOT23-6

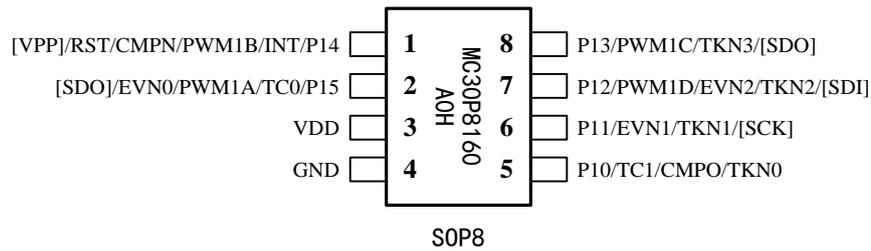
## 1.2 订购信息

产品名称	封装形式	备注
MC30P8160A0H	SOP8	
MC30P8160A0T	SOT23-6	

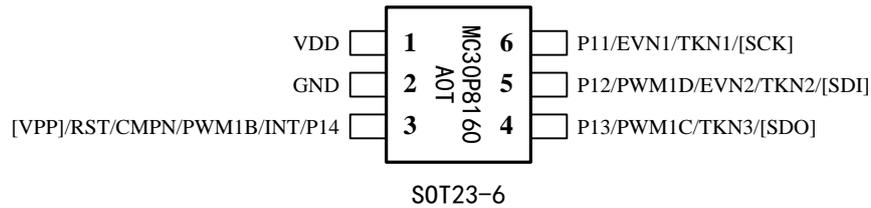
## 1.3 引脚排列



## MC30P8160A0H



## MC30P8160A0T



## 1.4 端口说明

端口名称	类型	功能说明
VDD	P	电源
GND	P	地
P10~P15	D	GPIO（推挽输出或开漏输出），内部上/下拉
INT	DI	外部中断输入
TC0~TC1	DI	定时器 T0~T1 的外部计数输入
PWM1A~PWM1D	DO	定时器 T1 的 4 路 PWM 输出
CMPN	AI	CMP 负端外部输入
EVN0	AI	CMP 正端外部输入；CMP 电压检测外部输入通道
EVN1~EVN2	AI	CMP 电压检测外部输入通道
CMPO	DO	CMP 输出
TKN0~TKN3	AI	触摸按键输入通道
RST	DI	外部复位输入
SCK, SDI, SDO	D	编程时钟/数据输入/数据输出接口
VPP	P	编程高压输入

注：P-电源端口；D-数字端口，DI-数字输入，DO-数字输出；A-模拟端口，AI-模拟输入，AO-模拟输出。





## 2 电气特性

### 2.1 极限参数

参数	符号	值	单位
电源电压	VDD	-0.3~6.0	V
I/O 输入电压	Vin	-0.3~VDD+0.3	V
工作温度	Ta	-40~85	°C
储存温度	Tstg	-65~150	°C
流入 VDD 最大电流	IVDDmax	50	mA
流出 GND 最大电流	IGNDmax	50	mA

注：若芯片工作条件超过极限值，则将造成永久性损坏；若芯片长时间工作在极限条件下，则将影响其可靠性。

### 2.2 直流电气特性

VDD=5V, T=25°C

特性	符号	端口	条件	最小	典型	最大	单位
工作电压	VDD	VDD	Fcpu=8MHz@FHIRC(16M)/2	VLVR36		5.5	V
			Fcpu=4MHz@FHIRC(8M)/2	VLVR20		5.5	
			Fcpu=2MHz@FHIRC(4M)/2	VLVR20		5.5	
			Fcpu=1MHz@FHIRC(2M)/2	VLVR18		5.5	
			Fcpu=500KHz@FHIRC(1M)/2	VLVR18		5.5	
			Fcpu=250KHz@FHIRC(500K)/2	VLVR18		5.5	
			Fcpu=16KHz@FLIRC(32K)/2	VPOR		5.5	
输入漏电流	Ileak	所有输入脚	VDD=5V	-1		1	μA
输入高电平	Vih	所有输入脚	SMT 开启, SMTVS 配置	0.8VDD			V
			SMT 开启, SMTVS 配置	2.0			
			SMT 关闭	2.0			
输入低电平	Vil	所有输入脚	SMT 开启, SMTVS 配置			0.2VDD	V
			SMT 开启, SMTVS 配置			0.8	
			SMT 关闭			1.0	
上拉电阻	Rpu	P1	VDD=5V, Vin=0		20		KΩ
下拉电阻	Rpd	P1	Vin=VDD=5V		20		KΩ



输出源电流	Ioh	推挽输出脚	Voh=VDD-0.6V, IOHS 配置		15		mA
			Voh=VDD-0.6V, IOHS 配置 此配置对 P14 脚无效, 保持大驱动		5		mA
输出灌电流	Iol	所有输出脚	Vol=0.6V		20		mA
运行模式功耗	Irun	VDD	Fcpu=8MHz@HIRC(16M)		2.9		mA
			Fcpu=4MHz@HIRC(8M)		1.6		mA
			Fcpu=2MHz@HIRC(4M)		0.9		mA
			Fcpu=1MHz@HIRC(2M)		580		μA
			Fcpu=500KHz@HIRC(1M)		420		μA
			Fcpu=250KHz@HIRC(500K)		330		μA
			Fcpu=16KHz@LIRC(32K), LVR 开		8		μA
休眠模式功耗	Istop	VDD	LIRC 关		0.3	2	μA
			LIRC 开		2.5	5	μA
低压复位电压	VLVR	VDD	LVRVS 配置	-15%		+15%	V
上电复位电压	VPOR	VDD	LVR 关闭	-30%	1.5	+30%	V
LVR 回滞电压		VDD			6%	12%	

注：条件项中，无关模块默认关闭，无关端口设为低电平无负载输出或内部上/下拉电阻无效且外接 GND 的输入。

## 2.3 交流电气特性

特性	符号	条件	最小	典型	最大	单位
HIRC 振荡频率	FHIRC	VDD=2.0V~5.5V, T=25°C	-3%	16	+3%	MHz
		VDD=2.0V~5.5V, T=-40°C~85°C	-5%		+5%	
LIRC 振荡频率	FLIRC	VDD=5V, T=25°C	-50%	32	+50%	KHz

## 2.4 CMP 特性参数

VDD=5V, T=25°C

特性	符号	条件	最小	典型	最大	单位
比较器有效工作电压	VCMP	T=-40°C~85°C	2.0		5.5	V
电压比较电路工作电流	ICMP			10		μA
VDD 检测电路工作电流	IEVD	VDD=5V, EVDIS=00		20	40	μA
输入失调电压	Voffset		-15		+15	mV
输入共模电压	Vcom		0		VDD-1.4	V
输出回滞电压	Vhys			6%		V



响应时间	T <sub>RESP</sub>	CMP 正端 > CMP 负端		250		μs
		CMP 正端 < CMP 负端		0.3		μs
通道切换等待时间	T <sub>CCW</sub>				20	μs
电压检测阈值电压	V <sub>EVD</sub>		-5%		+5%	V
内部基准电压	V <sub>IR</sub>		-10%	0.5	+10%	V
电阻分压比值			-1%		+1%	



## 3 CPU 与存储器

### 3.1 指令集

芯片的指令集为精简指令集。除程序跳转类指令外，其他指令均为单周期指令，即执行时间为 1 个指令周期（CPU 时钟周期）；所有指令均为单字指令，即指令码仅占用 1 个程序存储器地址空间。

指令汇总表

助记符	说明	操作	周期	长度	标志
ADDAR R	R 和 A 相加，结果存入 A	$R+A \rightarrow A$	1	1	C,DC,Z
ADDRA R	R 和 A 相加，结果存入 R	$R+A \rightarrow R$	1	1	C,DC,Z
ADCAR R	R 和 A 相加（带 C 标志），结果存入 A	$R+A+C \rightarrow A$	1	1	C,DC,Z
ADCRA R	R 和 A 相加（带 C 标志），结果存入 R	$R+A+C \rightarrow R$	1	1	C,DC,Z
RSUBAR R	R 和 A 相减，结果存入 A	$R-A \rightarrow A$	1	1	C,DC,Z
RSUBRA R	R 和 A 相减，结果存入 R	$R-A \rightarrow R$	1	1	C,DC,Z
RSBCAR R	R 和 A 相减（带 C 标志），结果存入 A	$R-A-/C \rightarrow A$	1	1	C,DC,Z
RSBCRA R	R 和 A 相减（带 C 标志），结果存入 R	$R-A-/C \rightarrow R$	1	1	C,DC,Z
ANDAR R	R 和 A 与操作，结果存入 A	$R \text{ and } A \rightarrow A$	1	1	Z
ANDRA R	R 和 A 与操作，结果存入 R	$R \text{ and } A \rightarrow R$	1	1	Z
ORAR R	R 和 A 或操作，结果存入 A	$R \text{ or } A \rightarrow A$	1	1	Z
ORRA R	R 和 A 或操作，结果存入 R	$R \text{ or } A \rightarrow R$	1	1	Z
XORAR R	R 和 A 异或操作，结果存入 A	$R \text{ xor } A \rightarrow A$	1	1	Z
XORRA R	R 和 A 异或操作，结果存入 R	$R \text{ xor } A \rightarrow R$	1	1	Z
COMAR R	对 R 取反，结果存入 A	$R \text{ 取反} \rightarrow A$	1	1	Z
COMR R	对 R 取反，结果存入 R	$R \text{ 取反} \rightarrow R$	1	1	Z
RLAR R	R 循环左移（带 C 标志），结果存入 A	$R[7] \rightarrow C; R[6:0] \rightarrow A[7:1]; C \rightarrow A[0]$	1	1	C
RLR R	R 循环左移（带 C 标志），结果存入 R	$R[7] \rightarrow C; R[6:0] \rightarrow R[7:1]; C \rightarrow R[0]$	1	1	C
RRAR R	R 循环右移（带 C 标志），结果存入 A	$R[0] \rightarrow C; R[7:1] \rightarrow A[6:0]; C \rightarrow A[7]$	1	1	C
RRR R	R 循环右移（带 C 标志），结果存入 R	$R[0] \rightarrow C; R[7:1] \rightarrow R[6:0]; C \rightarrow R[7]$	1	1	C
SWAPAR R	交换 R 的高低半字节，结果存入 A	$R[7:4] \rightarrow A[3:0]; R[3:0] \rightarrow A[7:4]$	1	1	-
SWAPR R	交换 R 的高低半字节，结果存入 R	$R[7:4] \rightarrow R[3:0]; R[3:0] \rightarrow R[7:4]$	1	1	-
MOVRA R	将 A 存入 R	$A \rightarrow R$	1	1	-
MOVAR R	将 R 存入 A	$R \rightarrow A$	1	1	Z
MOVR R	将 R 存入 R	$R \rightarrow R$	1	1	Z
CLRA	将 A 清零	$0 \rightarrow A$	1	1	Z



CLRR	R	将 R 清零	$0 \rightarrow R$	1	1	Z
INCR	R	R 自加 1	$R+1 \rightarrow R$	1	1	Z
INCAR	R	R 加 1, 结果存入 A	$R+1 \rightarrow A$	1	1	Z
DECR	R	R 自减 1	$R-1 \rightarrow R$	1	1	Z
DECAR	R	R 减 1, 结果存入 A	$R-1 \rightarrow A$	1	1	Z
JZR	R	R 自加 1: 结果为 0 则跳过下一条指令	$R+1 \rightarrow R$ : 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
JZAR	R	R 加 1, 结果存入 A: 结果为 0 则跳过下一条指令	$R+1 \rightarrow A$ : 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
DJZR	R	R 自减 1: 结果为 0 则跳过下一条指令	$R-1 \rightarrow R$ : 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
DJZAR	R	R 减 1, 结果存入 A: 结果为 0 则跳过下一条指令	$R-1 \rightarrow A$ : 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
BCLR	R,b	将 R 的第 b 位清 0	$0 \rightarrow R[b]$	1	1	-
BSET	R,b	将 R 的第 b 位置 1	$1 \rightarrow R[b]$	1	1	-
JBCLR	R,b	若 R 的第 b 位为 0, 则跳过下一条指令	若 $R[b]=0$ , 则 $PC+2 \rightarrow PC$	1/2	1	-
JBSET	R,b	若 R 的第 b 位为 1, 则跳过下一条指令	若 $R[b]=1$ , 则 $PC+2 \rightarrow PC$	1/2	1	-
ADDAI	I	I 和 A 相加, 结果存入 A	$I+A \rightarrow A$	1	1	C,DC,Z
ISUBAI	I	I 和 A 相减, 结果存入 A	$I-A \rightarrow A$	1	1	C,DC,Z
ANDAI	I	I 和 A 与操作, 结果存入 A	$I \text{ and } A \rightarrow A$	1	1	Z
ORAI	I	I 和 A 或操作, 结果存入 A	$I \text{ or } A \rightarrow A$	1	1	Z
XORAI	I	I 和 A 异或操作, 结果存入 A	$I \text{ xor } A \rightarrow A$	1	1	Z
MOVAI	I	将 I 存入 A	$I \rightarrow A$	1	1	-
CALL	K	子程序调用	$PC+1 \rightarrow TOS$ ; $K \rightarrow PC[10:0]$	2	1	-
GOTO	K	无条件跳转	$K \rightarrow PC[10:0]$	2	1	-
RETURN		从子程序返回	$TOS \rightarrow PC$	2	1	-
RETAI	I	从子程序返回, 并将 I 存入 A	$TOS \rightarrow PC$ ; $I \rightarrow A$	2	1	-
RETIE		从中断返回	$TOS \rightarrow PC$ ; $1 \rightarrow GIE$	2	1	-
NOP		空操作	空操作	1	1	-
DAA		BCD 码加法操作后, 将 A 的值调整为 BCD 码	$A(\text{HEX 码}) \rightarrow A(\text{BCD 码})$	1	1	C
DSA		BCD 码减法操作后, 将 A 的值调整为 BCD 码	$A(\text{HEX 码}) \rightarrow A(\text{BCD 码})$	1	1	-
CLRWDT		将看门狗计数器清零	$0 \rightarrow WDCNT$	1	1	TO,PD
STOP		进入低功耗模式	$0 \rightarrow WDCNT$ ; CPU 暂停	1	1	TO,PD

注:

- 1、A-算术逻辑单元累加器 ALU, R-数据存储器, I-立即数, K-程序存储器地址, TOS-堆栈栈顶;
- 2、对于条件跳转类指令, 若跳转条件成立, 则执行时间需 2 个指令周期, 否则仅需 1 个指令周期;



### 3.2 程序存储器

芯片的程序存储器为 OTP 型存储器，1K×14 位的地址空间范围为 0000H~03FFH。程序存储器地址分配如下图所示：

复位起始地址 (0000H)
通用程序区 (0001H - 0007H)
中断入口地址 (0008H)
通用程序区 (0009H - 03FFH)

### 3.3 数据存储器

芯片的数据存储器包括通用数据存储器 GPR (64 字节) 和特殊功能寄存器 SFR，地址映射如下表所示。GPR 和 SFR 均可直接寻址或通过 INDF 间接寻址。

数据存储器区地址映射表

地址	类型	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
00H-07H	SFR	INDF	T0CNT	PCL	STATUS	FSR		P1	
08H-0FH		MCR	T0CR	PCLATH	PDCON	ODCON	PUCON	INTE	INTF
10H-4FH	GPR	通用数据存储器区							
50H-57H	SFR	PWM1CR0	PWM1CR1	PWM1CR2				DDR1	
58H-5FH		TMRCR	T1CR	T1CNT	T1LOAD	T1DATA	T1DATB	T1DATC	T1DATD
60H-67H		T2CR	T2CNTH	T2CNTL	T2LOADH	T2LOADL	P1KBCR		P1ADCR
68H-6FH		CMPCR0	CMPCR1	CMPCR2	TKCR0	TKCR1	TKCR2	TKCNTH	TKCNTL
70H-7FH	保留	保留							

注：上表中灰色部分的存储器地址为系统保留区，禁止对其中未定义的地址进行读写操作。

数据存储器寻址方式地址组成

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	寻址方式
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---	------



/	/	/	/	/	/	/	/	/	取自指令的 7 位地址	直接寻址方式
/	/	/	/	/	/	/	/	/	FSR	间接寻址方式

直接寻址方式，是以指令的低 7 位为数据存储器地址，通过指令访问，寻址范围 0~7FH。例如，采用直接寻址方式将数据 55H 写入数据存储器 10H 地址中：

```
MOVAI    55H
MOVRA    10H          ; 将 55H 写入数据存储器 10H 地址中
```

间接寻址方式，是以 FSR 为数据存储器地址指针，通过 INDF 访问，寻址范围 0~7FH。例如，采用间接寻址方式将数据 55H 写入数据存储器 10H 地址中：

```
MOVAI    10H
MOVRA    FSR
MOVAI    55H
MOVRA    INDF        ; 将 55H 写入 FSR 所指地址的数据存储器中
```

### 3.4 堆栈

芯片的堆栈为 5 级深度的硬件堆栈。当 CPU 响应中断或执行子程序调用指令时，会自动将下一条指令的 PC 值压栈保存；当 CPU 执行中断返回或子程序返回指令时，会自动将栈顶内容出栈载入 PC。

### 3.5 控制寄存器

数据指针寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
FSR	-	FSR6	FSR5	FSR4	FSR3	FSR2	FSR1	FSR0
R/W	R	R/W						
初始值	1	X	X	X	X	X	X	X

BIT[6:0] FSR[6:0] – 数据指针寄存器

FSR：间接寻址方式的指针。



## 间接寻址寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
INDF	INDF7	INDF6	INDF5	INDF4	INDF3	INDF2	INDF1	INDF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] INDF[7:0] – 间接寻址寄存器

INDF: INDF 不是物理寄存器, 对 INDF 操作实际是对 FSR 所指向地址的数据存储器进行操作, 从而实现间接寻址功能。

## 程序指针计数器低字节

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
PCL	PC7	PC6	PC5	PC4	PC3	PC2	PC1	PC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] PC[7:0] – 程序指针计数器低 8 位

## 程序指针计数器高位缓存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
PCLATH	-	-	-	-	-	-	PCH1	PCH0
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

BIT[1:0] PCH[1:0] – 程序指针计数器高 2 位缓存器 (仅 C 编译模式下有效)

程序指针计数器 (PC) 有以下几种操作模式:

- ◇ 顺序运行指令:  $PC = PC + 1$ ;
- ◇ 程序跳转指令 GOTO/CALL:  $PC =$  指令码低 10 位;
- ◇ 返回指令 RETIE/RETURN/RETAI:  $PC =$  堆栈栈顶 (TOS);

对 PCL 操作指令 (汇编模式):

- ◇ 对 PCL 操作的加法指令:  $PC = (PC[9:0] + ALU[7:0])$ ;
- ◇ 对 PCL 操作的其他指令:  $PC = (PC[9:8]:ALU[7:0](ALU \text{ 运算结果}))$ ;

对 PCL 操作指令 (C 编译模式):

- ◇ 对 PCL 操作的所有指令:  $PC = (PCLATH[1:0]:ALU[7:0](ALU \text{ 运算结果}))$ ;



## CPU 状态寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
STATUS	WKUP	-	-	TO	PD	Z	DC	C
R/W	R/W	-	-	R	R	R/W	R/W	R/W
初始值	0	-	-	1	1	X	X	X

BIT[7] WKUP – 唤醒中断源标志位

- 0: 芯片被其他中断唤醒;
- 1: 芯片被外部中断或键盘中断唤醒;

BIT[4] TO – 看门狗溢出标志位

- 0: 发生 WDT 溢出;
- 1: 上电复位, 或执行 CLRWDT/STOP 指令;

BIT[3] PD – 进入低功耗模式标志位

- 0: 执行 STOP 指令;
- 1: 上电复位, 或执行 CLRWDT 指令;

BIT[2] Z – 零标志位

- 0: 算术或逻辑运算的结果不为零;
- 1: 算术或逻辑运算的结果为零;

BIT[1] DC – 半字节进位/借位标志位

- 0: 加法运算中半字节无进位; 减法运算中半字节有借位;
- 1: 加法运算中半字节有进位; 减法运算中半字节无借位;

BIT[0] C – 进位/借位标志位

- 0: 加法运算中无进位; 减法运算中有借位; 移位操作中移出位为 0;
- 1: 加法运算中有进位; 减法运算中无借位; 移位操作中移出位为 1;



### 3.6 用户配置字

芯片为保证系统正常工作，会将关键模块的配置信息预先存储于单独的存储器区域内，在上电或其他复位发生后将配置信息载入寄存器中，通过寄存器控制关键模块的工作状态。该部分存储器中用户可选的内容即为用户配置字，可在烧录用户程序代码时进行配置与烧录。

芯片的用户配置字，定义如下：

符号	功能说明
OSCM	系统主时钟选择： F <sub>OSC</sub> =内部高频 RC 振荡器时钟 F <sub>HIRC</sub> 的后分频时钟； F <sub>OSC</sub> =内部低频 RC 振荡器时钟 F <sub>LIRC</sub> ；
HIRCDS	HIRC 后分频输出选择： F <sub>OSC</sub> =F <sub>HIRC</sub> /1； F <sub>HIRC</sub> /2； F <sub>HIRC</sub> /4； F <sub>HIRC</sub> /8； F <sub>HIRC</sub> /16； F <sub>HIRC</sub> /32；
RSTEN	RST 外部复位端口设置： P14 为外部复位脚； P14 为输入/输出脚；
LVRVS	LVR 复位电压选择：(LVR 电压应满足由 F <sub>CPU</sub> 决定的工作电压特性) LVR 关闭 (V <sub>POR</sub> =1.5V)； 1.8V； 2.0V； 2.7V； 3.6V；
WDTM	WDT 模式设置： WDT 始终关闭； WDT 始终开启；
WDTT	WDT 溢出时间（无预分频）选择：(时间为典型值) 4.5ms； 18ms； 72ms； 288ms；
SMTEN	端口输入施密特设置： 输入 SMT 功能无效； 输入 SMT 功能有效；
SMTVS	端口施密特阈值选择： 2.0V/0.8V； 0.8VDD/0.2VDD；
IOHS	端口源电流输出选择： 源电流正常输出； 源电流限流输出； 注：[VPP]/RST/P14 引脚不支持限流输出，此位选择[源电流限流输出]P14 脚仍保持正常电流输出；
MCUSEL	芯片模式选择：(C 编译模式支持 C 程序对常量数组的定义，汇编模式则不支持) 汇编模式； C 编译模式；
ENCR	程序代码加密设置： 程序代码加密； 程序代码不加密；



## 4 时钟

芯片系统仅在系统主时钟  $F_{osc}$  下工作，部分外设模块可在系统主时钟  $F_{osc}$  或外设低频时钟  $FLCLK$  下工作。

系统主时钟  $F_{osc}$  可通过配置字  $OSCM$  和  $HIRCDS$  选择以下时钟：

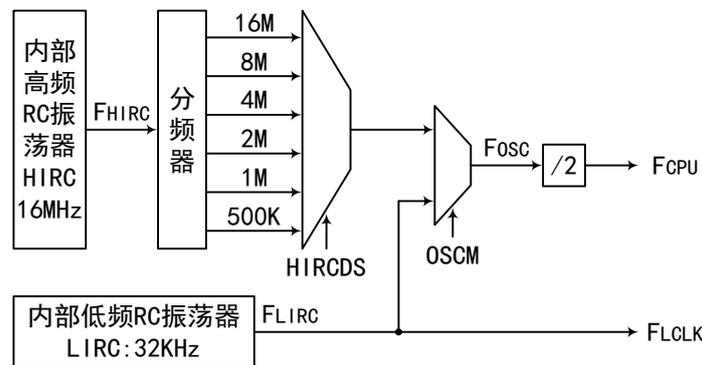
- ◇ 内部高频 RC 振荡器 HIRC (16MHz) 时钟  $F_{HIRC}$  的 1/2/4/8/16/32 分频时钟；
- ◇ 内部低频 RC 振荡器 LIRC (32KHz) 时钟  $F_{LIRC}$ ；

外设低频时钟  $FLCLK$  固定为内部低频 RC 振荡器 LIRC (32KHz) 时钟  $F_{LIRC}$ 。

CPU 的时钟源固定为系统主时钟  $F_{osc}$ ，CPU 的时钟频率  $F_{CPU}$  固定为  $F_{osc}$  的 2 分频。

WDT (看门狗) 电路的时钟源固定为内部低频 RC 振荡器 LIRC。

系统时钟示意图



### 4.1 内部高频 RC 振荡器

芯片内置 1 个振荡频率为 16MHz 的高精度 HIRC 振荡器，其分频时钟 (16MHz/8MHz/4MHz/2MHz/1MHz/500KHz) 可用作系统主时钟源。

### 4.2 内部低频 RC 振荡器



芯片内置 1 个振荡频率典型值为 32KHz 的 LIRC 振荡器，可用作系统主时钟源或外设低频时钟源，也用于系统上电延时控制、看门狗定时器（WDT）等电路。

注：若系统主时钟源为 HIRC，则仅在 WDT 开启、或 T0（时钟源为 FLIRC）使能时，LIRC 才工作。

### 4.3 系统工作模式

芯片支持运行模式和休眠模式 2 种系统工作模式。

工作模式	模式切换条件	系统工作状态
运行模式	系统复位	主时钟源工作，CPU 在主时钟或其分频时钟下运行
	休眠模式下，CPU 唤醒	
休眠模式	运行模式下，执行 STOP 指令	主时钟源停止，CPU 暂停

注：若系统主时钟为 LIRC，则在 WDT 开启、或 T0（时钟源为 FLIRC）使能后进入休眠模式，LIRC 也将一直工作。

### 4.4 低功耗模式

芯片的休眠模式即为低功耗模式。

执行 STOP 指令可使系统进入低功耗模式，同时对系统会产生以下影响：

- ◇ CPU 停止运行；
- ◇ 根据不同模式停止相应时钟源的振荡；
- ◇ RAM 内容保持不变；
- ◇ 所有的输入/输出端口保持原有状态；
- ◇ 定时器若其时钟源未停止，则可继续工作；

以下情况可使系统退出低功耗模式：

- ◇ 芯片复位；
- ◇ WDT 溢出（若低功耗模式下 WDT 及其时钟源保持继续工作）；
- ◇ 外部中断请求发生（若有外部中断功能并有效）；
- ◇ 定时器中断请求发生（若低功耗模式下定时器及其时钟源保持继续工作）；
- ◇ 键盘中断请求发生（若有键盘中断功能并有效）；



◇ 比较器 CMP 中断请求发生（若低功耗模式下比较器保持继续工作）；

注：

- 1、低功耗模式下触发中断请求时，若对应的中断使能位关闭，则不会退出低功耗模式；若对应的中断使能位开启而中断总使能位关闭，则仅唤醒 CPU 执行下一条指令；若对应的中断使能位和中断总使能位均开启，则唤醒 CPU 后将执行中断服务程序；
- 2、未使用或未封出的引脚，应将其对应的 I/O 端口设置为输出、输入上拉或输入下拉等稳定状态，以免因引脚浮空而产生漏电流或非预期的中断唤醒；



## 5 复位

### 5.1 复位条件

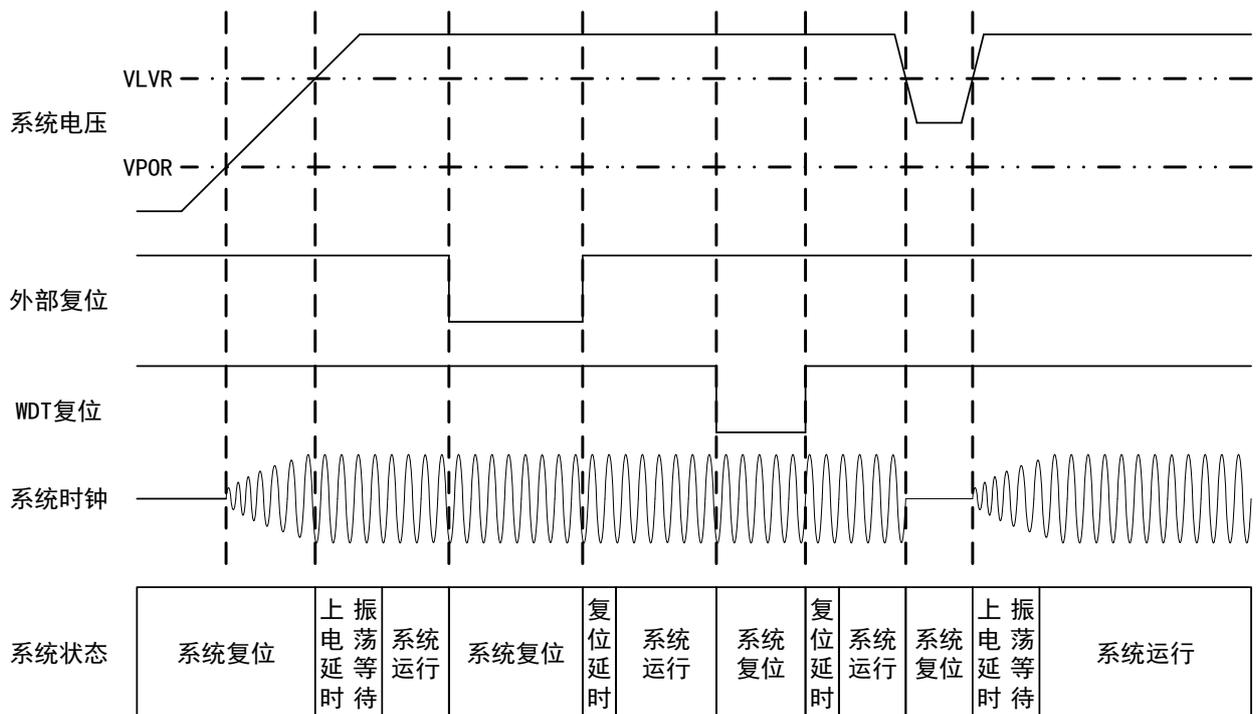
芯片共有如下几种复位方式：

- ◇ 上电复位 POR；
- ◇ 低电压复位 LVR；
- ◇ 外部复位；
- ◇ WDT 看门狗复位；

任何一种复位发生后，系统进入复位状态，执行初始化操作并重置 SFR 为复位初始值；复位条件解除后，系统退出复位状态，CPU 重新从程序存储器 0000H 地址处开始运行。

上电复位 POR 和低电压复位 LVR 会关闭系统主时钟振荡器，复位解除后才重新开启振荡器，因为振荡器起振和稳定需要一定的时间，所以系统将保持一定时间的上电延时（典型值为 9ms）以待振荡器稳定振荡后才开始工作；而外部复位、WDT 复位则不会关闭主时钟振荡器，复位解除后系统将在较短的复位延时后即开始工作。

下图是复位产生和系统工作状态之间的时序关系示意图：





注：若应用系统在上电或掉电回升时芯片的 VDD 电压上升较慢，则应在复位后 CPU 开始工作时先进行软件延时，以确保芯片开始工作时 VDD 电压已稳定在  $F_{CPU}$  对应的工作电压范围内。

## 5.2 上电复位

芯片的上电复位电路可以适应系统快速上电或慢速上电等情况，即使上电过程中发生电源电压抖动的情况也能保证系统可靠的复位。

上电复位过程主要包括以下几个步骤：

- (1) 检测系统电源电压，等待电压高于上电复位电压  $V_{POR}$  并保持稳定；
- (2) 若 LVR 功能开启，则需等待电压高于低电压复位电压  $V_{LVR}$  并保持稳定；
- (3) 若有外部复位功能并已开启，则需等待外部复位引脚电压高于  $V_{ih}$ ；
- (4) 初始化所有初始值确定的寄存器；
- (5) 开启主时钟振荡器，并等待一段时间以待振荡器稳定；
- (6) 上电复位结束，CPU 开始执行指令；

## 5.3 外部复位

芯片的外部复位功能可通过配置字 RSTEN 开启，引脚设为外部复位脚即为开启外部复位功能，端口内部上拉电阻将自动使能。外部复位输入端口 RST 为施密特结构，低电平有效，即当端口输入为高电平时系统正常运行，输入为低电平时系统复位。

## 5.4 低电压复位

芯片的低电压复位电压  $V_{LVR}$  可通过配置字 LVRVS 选择。LVR 检测电路具有一定的回滞特性，回滞电压约为 6%（典型值），当电源电压下降至  $V_{LVR}$  时发生 LVR 复位，反之电源电压需上升至  $V_{LVR}+6\%$  后 LVR 复位才解除。

若系统主时钟源为 HIRC，则 LVR 在系统进入休眠模式后将自动关闭，进入运行模式后自动开启；若系统主时钟源为 LIRC，则 LVR 将一直关闭。



## 5.5 看门狗复位

芯片的看门狗定时器（WDT）复位是一种对系统运行程序的保护机制。正常情况下，用户程序需定时对 WDT 执行清零操作，以避免 WDT 溢出。若发生异常情况，程序未及时清零 WDT，则芯片将因 WDT 溢出而产生看门狗复位，系统初始化后重新运行程序，从而返回受控状态。

*注：低功耗模式下 CPU 暂停工作，若此时发生 WDT 溢出，则仅唤醒 CPU 而不复位芯片。*



## 6 I/O 端口

### 6.1 通用 I/O 功能

芯片的输入/输出端口为一组 6 位端口 P1。所有端口均支持施密特输入，均可选推挽或开漏输出。除用作通用数字 I/O 端口外，部分端口还可复用为外部中断输入、PWM 输出、或 CMP 模拟输入等功能。

#### 端口数据寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
P1	-	-	P15D	P14D	P13D	P12D	P11D	P10D
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	X	X	X	X	X	X

BIT[5:0] P1nD – P1n 端口数据位 (n=5-0)

#### 端口方向寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
DDR1	-	-	DDR15	DDR14	DDR13	DDR12	DDR11	DDR10
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	1	1	1	1	1	1

BIT[5:0] DDR1n – P1n 端口方向控制位 (n=5-0)

- 0: 端口作为输出口，读端口操作将读取端口的数据位值；
- 1: 端口作为输入口，读端口操作将读取端口的电平状态；

### 6.2 内部上/下拉电阻

所有端口均具有内部上拉和下拉电阻，且均可单独控制其上/下拉电阻在端口处于输入状态（或端口数字 I/O 功能关闭）时是否有效。端口处于输出状态时，上/下拉电阻及其控制位无效。

#### 上拉电阻控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
PUCON	-	-	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU



R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	1	1	1	1	1	1

BIT[5:0] P1nPU – P1n 端口上拉电阻控制位 (n=5-0)

0: 端口内部上拉电阻有效;

1: 端口内部上拉电阻无效;

#### 下拉电阻控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
PDCON	-	-	P15PD	P14PD	P13PD	P12PD	P11PD	P10PD
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	1	1	1	1	1	1

BIT[5:0] P1nPD – P1n 端口下拉电阻控制位 (n=5-0)

0: 端口内部下拉电阻有效;

1: 端口内部下拉电阻无效;

## 6.3 端口模式控制

部分端口除可作为数字端口外，还可复用为模拟端口。端口输入或输出模拟信号时，若数字 I/O 功能同时开启，则会产生漏电流，可通过端口数模控制寄存器关闭端口的数字 I/O 功能（内部上/下拉电阻及其控制位不受影响）。

#### 端口数模控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
P1ADCR	-	-	P15DC	P14DC	P13DC	P12DC	P11DC	P10DC
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	0	0	0	0	0	0

BIT[5:0] P1nDC – P1n 端口数字功能控制位 (n=5-0)

0: 使能端口的数字 I/O 功能;

1: 关闭端口的数字 I/O 功能;

用作数字输出口时，P1 所有端口均可选择推挽输出或开漏输出。当端口为推挽输出、或开漏输出低



时，输入通路保持连接；而当端口开漏输出高时，输入通路将自动关断。

端口输出模式寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
ODCON	-	-	P15OD	P14OD	P13OD	P12OD	P11OD	P10OD
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	0	0	0	0	0	0

BIT[5:0] P1nOD – P1n 端口开漏输出控制位 (n=5-0)

0: 端口输出时为推挽输出；

1: 端口输出时为开漏输出；



## 7 定时器 TIMER

### 7.1 看门狗定时器 WDT

看门狗定时器 WDT 的时钟源为内部低频 RC 振荡器 LIRC, 可通过预分频器选择不同的 WDT 计数时钟频率。WDT 溢出将复位芯片或唤醒 CPU。

可通过配置字 WDTM 和寄存器位 WDTEN 共同决定是否开启 WDT: WDTM 选择始终关闭、或 WDTEN 为 0, 则 WDT 关闭; 而当 WDTM 选择始终开启且 WDTEN 为 1 时, WDT 才开启。若 WDT 开启, 则在休眠模式下 WDT 依然工作且溢出将唤醒 CPU, 而在 CPU 运行时 WDT 溢出将复位芯片。

WDT 和定时器 T0 共用一个预分频器, 并通过寄存器位决定预分频器的分配。当预分频器分配给 T0 时, WDT 时钟不分频; 而预分频器分配给 WDT 时, T0 时钟不分频。

执行 CLRWDT 指令或 STOP 指令将清零 WDT 计数器, 若预分频器分配给 WDT, 则还将清零预分频计数器 (预分频比不变)。

WDT 的基础溢出时间 (即无预分频的时间) 可配置为 4.5ms/18ms/72ms/288ms。

*注: WDT 溢出时间为典型值, 而实际值偏差较大, 必须保证清 WDT 的间隔时间小于 WDT 溢出时间的 1/4。*

#### 杂项控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
MCR	WDTEN	EIS	CMPOF	-	-	-	-	-
R/W	R/W	R/W	R	-	-	-	-	-
初始值	1	0	0	-	-	-	-	-

BIT[7] WDTEN – 看门狗使能位

0: 关闭 WDT;

1: 开启 WDT;

BIT[6] EIS – INT 端口外部中断功能使能位

0: 关闭端口的外部中断功能;

1: 使能端口的外部中断功能;

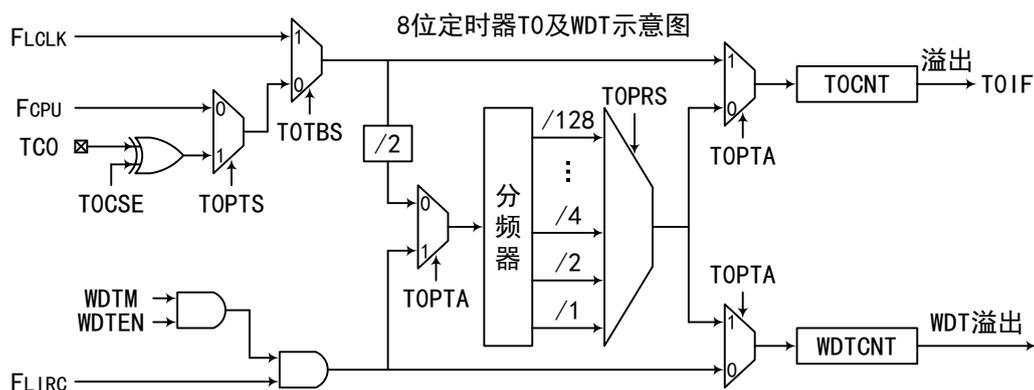


- BIT[5] CMPOF – CMP 比较结果状态标志位
- 0: CMP 比较结果为低, 即比较器正端输入电压低于负端输入电压; 或 CMP 关闭;
  - 1: CMP 比较结果为高, 即比较器正端输入电压高于负端输入电压;

## 7.2 定时器 T0

定时器 T0 为 8 位定时/计数器, 包含 1 个 8 位递增计数器、可编程预分频器、控制寄存器。

- ◇ 可通过预分频器设置时钟频率;
- ◇ 时钟源可选 F<sub>CPU</sub>、FLCLK (即 FLIRC)、或外部时钟 (TC0 输入);
- ◇ 支持溢出中断和溢出唤醒功能;



TOCNT 为 8 位可读写的递增计数器, 计数溢出到 0 时产生溢出信号并触发中断, 中断标志 TOIF 将被置 1。

预分频器为 T0 与 WDT 共用, 通过寄存器位 TOPTA 控制预分频器的分配。

TOPTA=0 时, 预分频器分配给 T0 使用, T0 周期 (溢出时间) = 预分频比 / T0 计数时钟频率, 写 TOCNT 将清零预分频计数器, 而执行 CLRWDT 或 STOP 指令则不影响预分频器的计数。

TOPTA=1 时, 预分频器分配给 WDT 使用, 执行 CLRWDT 或 STOP 指令将清零预分频计数器, 而写 TOCNT 则同样不影响预分频器的计数。

通过 TOPTA 改变预分频器的分配时, 也将清零预分频计数器。



清零预分频计数器的操作不会改变预分频比，而改变预分频比也不会清零预分频计数器。

当通过 T0TBS 选择 FLCLK 作为 T0 时钟时，在低功耗模式下 T0 将继续工作，溢出可唤醒。

#### 定时器 T0 控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T0CR	-	INTM	T0PTS	T0CSE	T0PTA	T0PRS2	T0PRS1	T0PRS0
R/W	-	R/W						
初始值	-	0	1	1	1	1	1	1

BIT[6] INTM – 外部中断 INT 触发方式选择位

- 0: 下降沿触发;
- 1: 上升沿触发;

BIT[5] T0PTS – T0 时钟源选择位 (仅 T0TBS=0 时有效)

- 0: T0 时钟源为 F<sub>CPU</sub>;
- 1: T0 时钟源为 TC0 输入的外部时钟 (端口需设为输入状态);

BIT[4] T0CSE – T0 外部时钟计数沿选择位

- 0: T0 在外部时钟上升沿计数;
- 1: T0 在外部时钟下降沿计数;

BIT[3] T0PTA – 预分频器分配控制位

- 0: 预分频器分配给 T0;
- 1: 预分频器分配给 WDT;

BIT[2:0] T0PRS[2:0] – T0 时钟预分频比选择位

T0PRS[2:0]	T0 时钟预分频比 (T0PTA=0)	WDT 时钟预分频比 (T0PTA=1)
000	1:2	1:1
001	1:4	1:2
010	1:8	1:4
011	1:16	1:8
100	1:32	1:16
101	1:64	1:32



110	1:128	1:64
111	1:256	1:128

## 定时器 T0 计数器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
TOCNT	TOCNT7	TOCNT6	TOCNT5	TOCNT4	TOCNT3	TOCNT2	TOCNT1	TOCNT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] TOCNT[7:0] – T0 计数器，为可读写的递增计数器

## 定时器控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
TMRCR	-	T0TBS	-	-	T2IE	T2IF	T1IE	T1IF
R/W	-	R/W	-	-	R/W	R/W	R/W	R/W
初始值	-	0	-	-	0	0	0	0

BIT[6] T0TBS – T0 时钟源选择位

0: T0 时钟源由 T0PTS 决定;

1: T0 时钟源为外设低频时钟 FLCLK;

BIT[3] T2IE – 定时器 T2 中断使能位

0: 屏蔽定时器 T2 中断;

1: 使能定时器 T2 中断;

BIT[2] T2IF – 定时器 T2 中断标志位

0: 未触发定时器 T2 中断;

1: 已触发定时器 T2 中断，需软件清 0;

BIT[1] T1IE – 定时器 T1 中断使能位

0: 屏蔽定时器 T1 中断;

1: 使能定时器 T1 中断;

BIT[0] T1IF – 定时器 T1 中断标志位



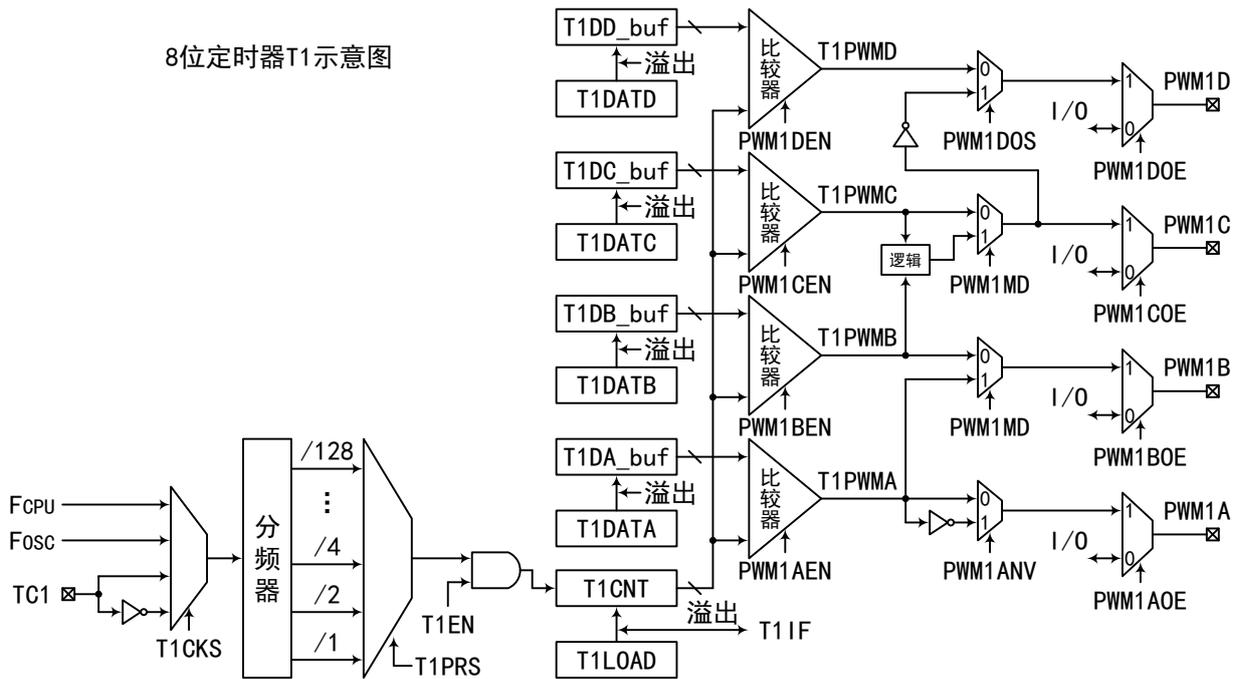
- 0: 未触发定时器 T1 中断;
- 1: 已触发定时器 T1 中断, 需软件清 0;

### 7.3 定时器 T1

定时器 T1 为 8 位定时/计数器, 包含 1 个 8 位递减计数器、可编程预分频器、控制寄存器、8 位重载寄存器和 4 个 8 位比较寄存器。

- ◇ 可通过预分频器设置时钟频率, 可通过重载寄存器控制计数周期;
- ◇ 支持 4 路 8 位共周期 PWM 输出, 可通过对应的比较寄存器分别设置每路 PWM 占空比;
- ◇ 可组合成 2 对互反的带死区互补 PWM;
- ◇ 支持溢出中断和溢出唤醒功能;

8位定时器T1示意图



定时器 T1, 可通过寄存器位 T1CKS 选择时钟源, 通过 T1PRS 选择时钟预分频比, 所选时钟源通过预分频器后产生 T1 计数器 T1CNT 的计数时钟 (上升沿计数)。写 T1CNT 将清零预分频计数器, 而预分频比保持不变。

T1EN=0 时, T1CNT 保持不变, 写重载寄存器 T1LOAD 将立即载入 T1CNT; T1EN=1 时, T1CNT 递减计数, 计数到 0 的时钟结束后产生溢出信号并触发中断, 中断标志 T1IF 将被置 1, 同时 T1 自动将当前 T1LOAD 值载入 T1CNT 并重新开始计数。



如图所示，定时器 T1 可实现 4 路共周期的 PWM 功能 (PWM1x, x=A,B,C,D, 下同)，可分别设置每路 PWM 占空比，可通过寄存器位使能或关闭 PWM 功能，并控制端口是否输出 PWM 波形。PWM1x 关闭时 T1PWMx 信号为低电平。PWM1x 使能后 T1CNT 从重载值开始递减计数直到计数溢出为一个 PWM 周期：当计数到与比较寄存器 T1DATx 相等时，T1PWMx 变为高电平；当计数溢出时，T1PWMx 变为低电平。

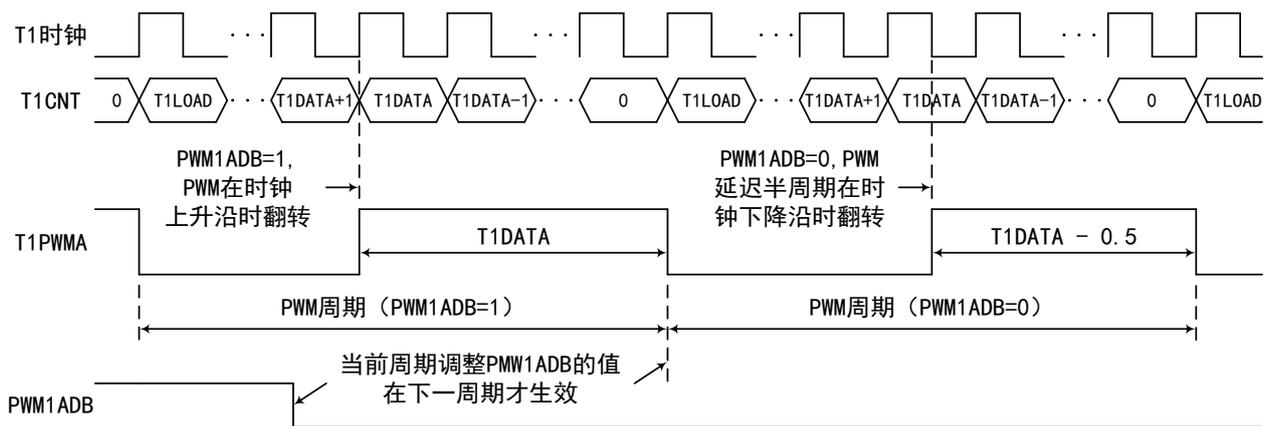
T1DATx 均配有 1 个 8 位比较缓冲器 (T1Dx\_buf) 用于与 T1CNT 比较，PWM1x 关闭时写 T1DATx 将立即载入缓冲器中，而 PWM1x 使能后写 T1DATx 则将在 T1 溢出时才载入缓冲器中。若要首个 PWM 周期和占空比准确，需先写重载寄存器和比较寄存器，再使能 PWM，最后开启定时器。

T1PWMx 信号 (x=A,B,C,D, 下同) 的占空比计算如下：

- ◇ 高电平时间 = (T1DATx) × T1CNT 计数时钟周期
- ◇ 周期 (T1 溢出时间) = (T1LOAD + 1) × T1CNT 计数时钟周期
- ◇ 占空比 (高电平时间/周期) = (T1DATx) / (T1LOAD + 1)

PWM1x 的占空比可通过寄存器位 PWM1xDB 微调半个计数时钟周期。PWM1xDB=1 时，PWM 周期内 T1PWMx 信号在计数时钟上升沿变为高电平；PWM1xDB=0 时，PWM 周期内 T1PWMx 信号滞后半个时钟周期在时钟下降沿变为高电平。

PWM1 半个时钟微调波形示意



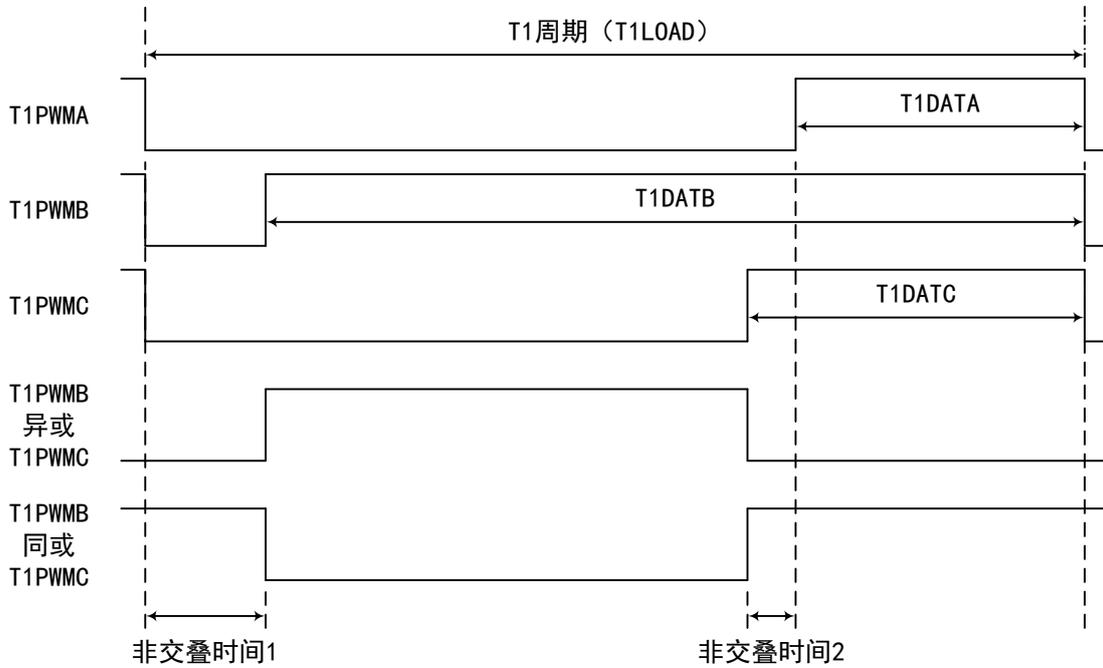
注：

- 1、PWM1x 的高电平时间需超过 1 个计数时钟周期 (即 T1DATx 需 >1)，半周期微调功能才会有效；
- 2、因 F<sub>CPU</sub> 的占空比不为 50%，所以当 PWM 计数时钟为 F<sub>CPU</sub> 的 1 分频时，微调功能的翻转延迟并不正好为半个时钟周期，而是延迟为 F<sub>CPU</sub> 的高电平时间；



如图所示，当 PWM1MD=1 时，端口 PWM1B 输出 T1PWMA 信号，端口 PWM1C 输出 T1PWMB 和 T1PWMC 的组合逻辑(同或/异或)信号,从而组合成 1 对带死区的互补 PWM;而端口 PWM1A 和 PWM1D 则可输出互补 PWM 的反向波形，从而组合成 2 对互反的带死区互补 PWM。

PWM1 互补波形示意



定时器 T1 控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T1CR	T1EN	PWM1AOE	PWM1AEN	T1CKS1	T1CKS0	T1PRS2	T1PRS1	T1PRS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7] T1EN – 定时器 T1 使能位

- 0: 关闭定时器 T1;
- 1: 开启定时器 T1;

BIT[6] PWM1AOE – PWM1A 端口输出使能位

- 0: 禁止端口输出脉宽调制波形;
- 1: 允许端口输出脉宽调制波形;

BIT[5] PWM1AEN – PWM1A 使能位

- 0: 关闭 PWM1A 功能;



1: 使能 PWM1A 功能;

BIT[4:3] T1CKS[1:0] – T1 时钟源选择位

T1CKS[1:0]	T1 时钟源
00	F <sub>CPU</sub>
01	F <sub>osc</sub>
10	TC1 上升沿
11	TC1 下降沿

BIT[2:0] T1PRS[2:0] – T1 时钟预分频比选择位

T1PRS[2:0]	T1 时钟预分频比
000	1 : 1
001	1 : 2
010	1 : 4
011	1 : 8
100	1 : 16
101	1 : 32
110	1 : 64
111	1 : 128

定时器 T1 计数器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T1CNT	T1CNT7	T1CNT6	T1CNT5	T1CNT4	T1CNT3	T1CNT2	T1CNT1	T1CNT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

BIT[7:0] T1CNT[7:0] – T1 计数器, 为可读写的递减计数器

定时器 T1 重载寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T1LOAD	T1LOAD7	T1LOAD6	T1LOAD5	T1LOAD4	T1LOAD3	T1LOAD2	T1LOAD1	T1LOAD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

BIT[7:0] T1LOAD[7:0] – T1 重载寄存器, 用于设置 T1 的计数周期

注: 定时器重载寄存器的值禁止为 0, 否则定时器将无法正常工作。



## 定时器 T1 比较寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T1DATA	T1DATA7	T1DATA6	T1DATA5	T1DATA4	T1DATA3	T1DATA2	T1DATA1	T1DATA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] T1DATA[7:0] – T1 比较寄存器 A，用于设置 PWM1A 的占空比

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T1DATB	T1DATB7	T1DATB6	T1DATB5	T1DATB4	T1DATB3	T1DATB2	T1DATB1	T1DATB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] T1DATB[7:0] – T1 比较寄存器 B，用于设置 PWM1B 的占空比

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T1DATC	T1DATC7	T1DATC6	T1DATC5	T1DATC4	T1DATC3	T1DATC2	T1DATC1	T1DATC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] T1DATC[7:0] – T1 比较寄存器 C，用于设置 PWM1C 的占空比

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T1DATD	T1DATD7	T1DATD6	T1DATD5	T1DATD4	T1DATD3	T1DATD2	T1DATD1	T1DATD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] T1DATD[7:0] – T1 比较寄存器 D，用于设置 PWM1D 的占空比

## PWM1 控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
PWM1CR0	-	-	PWM1DOE	PWM1DEN	PWM1COE	PWM1CEN	PWM1BOE	PWM1BEN
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	0	0	0	0	0	0

BIT[5] PWM1DOE – PWM1D 端口输出使能位

0: 禁止端口输出脉宽调制波形;

1: 允许端口输出脉宽调制波形;



- BIT[4] PWM1DEN – PWM1D 使能位  
0: 关闭 PWM1D 功能;  
1: 使能 PWM1D 功能;
- BIT[3] PWM1COE – PWM1C 端口输出使能位  
0: 禁止端口输出脉宽调制波形;  
1: 允许端口输出脉宽调制波形;
- BIT[2] PWM1CEN – PWM1C 使能位  
0: 关闭 PWM1C 功能;  
1: 使能 PWM1C 功能;
- BIT[1] PWM1BOE – PWM1B 端口输出使能位  
0: 禁止端口输出脉宽调制波形;  
1: 允许端口输出脉宽调制波形;
- BIT[0] PWM1BEN – PWM1B 使能位  
0: 关闭 PWM1B 功能;  
1: 使能 PWM1B 功能;

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
PWM1CR1	-	-	-	-	PWM1DOS	PWM1ANV	PWM1MD	PWM1LG
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	0	0	0	0

- BIT[3] PWM1DOS – PWM1D 端口输出信号选择位  
0: 输出 T1PWMD 信号;  
1: 输出 PWM1C 端口信号的反向波形;
- BIT[2] PWM1ANV – PWM1A 端口输出取反控制位  
0: 端口输出正向波形;  
1: 端口对电平取反后输出;
- BIT[1] PWM1MD – PWM1 输出模式选择位



- 0: P14 (PWM1B) 输出 T1PWMB 信号, P13 (PWM1C) 输出 T1PWMC 信号;  
 1: P14 输出 T1PWMA 信号, P13 输出 T1PWMB 和 T1PWMC 的组合逻辑信号;

BIT[0] PWM1LG – 组合逻辑控制位

- 0: 组合逻辑为 T1PWMB 与 T1PWMC 同或;  
 1: 组合逻辑为 T1PWMB 与 T1PWMC 异或;

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
PWM1CR2	-	-	-	-	PWM1DDB	PWM1CDB	PWM1BDB	PWM1ADB
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	1	1	1	1

BIT[3] PWM1DDB – T1PWMD 匹配翻转时钟沿选择位

- 0: 占空比匹配时, 脉宽调制信号电平滞后半个时钟周期翻转;  
 1: 占空比匹配时, 脉宽调制信号电平与计数时钟的有效沿同步翻转;

BIT[2] PWM1CDB – T1PWMC 匹配翻转时钟沿选择位

- 0: 占空比匹配时, 脉宽调制信号电平滞后半个时钟周期翻转;  
 1: 占空比匹配时, 脉宽调制信号电平与计数时钟的有效沿同步翻转;

BIT[1] PWM1BDB – T1PWMB 匹配翻转时钟沿选择位

- 0: 占空比匹配时, 脉宽调制信号电平滞后半个时钟周期翻转;  
 1: 占空比匹配时, 脉宽调制信号电平与计数时钟的有效沿同步翻转;

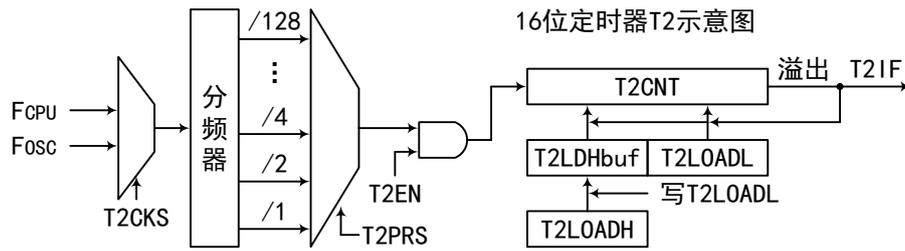
BIT[0] PWM1ADB – T1PWMA 匹配翻转时钟沿选择位

- 0: 占空比匹配时, 脉宽调制信号电平滞后半个时钟周期翻转;  
 1: 占空比匹配时, 脉宽调制信号电平与计数时钟的有效沿同步翻转;

## 7.4 定时器 T2

定时器 T2 为 16 位定时器, 包含 1 个 16 位递减计数器、可编程预分频器、控制寄存器、16 位重载寄存器。

- ◇ 可通过预分频器设置时钟频率, 可通过重载寄存器控制计数周期;
- ◇ 支持单次计时模式 (触摸模式), 可控制触摸模块是否进行电容充放电计数;
- ◇ 支持溢出中断和溢出唤醒功能;



定时器 T2，可通过寄存器位 T2CKS 选择时钟源，通过 T2PRS 选择时钟预分频比，所选时钟源通过预分频器后产生 T2 计数器 T2CNT 的计数时钟（上升沿计数）。写 T2CNT 将清零预分频计数器，而预分频比保持不变。

16 位 T2CNT 的高字节 T2CNTH 配有缓冲器，读写 T2CNTH 实际为读写缓冲器中内容，T2CNTH 实际内容仅在读写 T2CNTL 时才自动与缓冲器交互，因此写 T2CNT 需先写 T2CNTH 再写 T2CNTL（硬件同时自动将缓存器内容载入 T2CNTH），而读 T2CNT 则需先读 T2CNTL（硬件同时自动将 T2CNTH 内容载入缓冲器中）再读 T2CNTH。

T2EN=0 时，T2CNT 保持不变，写重载寄存器 T2LOAD 将立即载入 T2CNT；T2EN=1 时，T2CNT 递减计数，计数到 0 的时钟结束后产生溢出信号并触发中断，中断标志 T2IF 将被置 1，同时 T2 自动将当前 T2LOAD 值载入 T2CNT 并重新开始计数。

16 位 T2LOAD 的高字节 T2LOADH 配有缓冲器（T2LDHbuf），写 T2LOADL 时会同时将 T2LOADH 内容载入该缓冲器中，因此调整 T2LOAD 值时需先写 T2LOADH 再写 T2LOADL。此时若 T2EN=0，则会同时再将[缓冲器:T2LOADL]载入 T2CNT；若 T2EN=1，则会在 T2 溢出后才将[缓冲器:T2LOADL]载入 T2CNT。

定时器 T2 在单次计时模式下，T2EN 置 1 将允许 16 位 TKM 电容充放计数器 TKCNT 计数（需 TKEN 和 TKCON 均为 1），当 T2 溢出时，中断标志 T2IF 将被置 1，同时 T2EN 自动清 0 且 TKCNT 停止计数。

#### 定时器 T2 控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T2CR	T2EN	T2MOD	-	T2CKS1	T2CKS0	T2PRS2	T2PRS1	T2PRS0
R/W	R/W	R/W	-	R/W	R/W	R/W	R/W	R/W
初始值	0	0	-	0	0	0	0	0

BIT[7] T2EN - 定时器 T2 使能位

0: 关闭定时器 T2;

1: 开启定时器 T2;



- BIT[6] T2MOD – T2 工作模式选择位  
 0: 连续定时/计数模式;  
 1: 单次计时模式 (触摸模式);

- BIT[4:3] T2CKS[1:0] – T2 时钟源选择位

T2CKS[1:0]	T2 时钟源
00	F <sub>CPU</sub>
01	F <sub>osc</sub>
10	-
11	-

- BIT[2:0] T2PRS[2:0] – T2 时钟预分频比选择位

T2PRS[2:0]	T2 时钟预分频比
000	1 : 1
001	1 : 2
010	1 : 4
011	1 : 8
100	1 : 16
101	1 : 32
110	1 : 64
111	1 : 128

#### 定时器 T2 计数器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T2CNTH	T2CNT15	T2CNT14	T2CNT13	T2CNT12	T2CNT11	T2CNT10	T2CNT9	T2CNT8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

- BIT[7:0] T2CNT[15:8] – T2 计数器高 8 位, 为可读写的递减计数器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T2CNTL	T2CNT7	T2CNT6	T2CNT5	T2CNT4	T2CNT3	T2CNT2	T2CNT1	T2CNT0
R/W								
初始值	1	1	1	1	1	1	1	1



BIT[7:0] T2CNT[7:0] – T2 计数器低 8 位，为可读写的递减计数器

定时器 T2 重载寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T2LOADH	T2LOAD15	T2LOAD14	T2LOAD13	T2LOAD12	T2LOAD11	T2LOAD10	T2LOAD9	T2LOAD8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

BIT[7:0] T2LOAD[15:8] – T2 重载寄存器高 8 位，用于设置 T2 的计数周期

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
T2LOADL	T2LOAD7	T2LOAD6	T2LOAD5	T2LOAD4	T2LOAD3	T2LOAD2	T2LOAD1	T2LOAD0
R/W								
初始值	1	1	1	1	1	1	1	1

BIT[7:0] T2LOAD[7:0] – T2 重载寄存器低 8 位，用于设置 T2 的计数周期

注：定时器重载寄存器的值禁止为 0，否则定时器将无法正常工作。

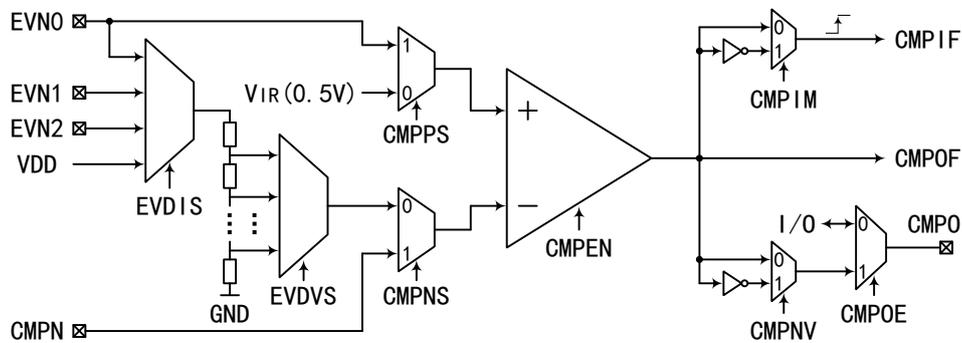


## 8 比较器 CMP 和电压检测 EVD

### 8.1 CMP 概述

芯片内置 1 个模拟比较器 CMP，包含内部基准电压  $V_{IR}$  (0.5V) 生成电路和内部电阻分压电路。

- ◇ 输入共模  $0 \sim (V_{DD}-1.4V)$ ，输出可选有/无回滞（单边 6%）；
- ◇ 正端输入可选择外部输入电压、或内部基准电压  $V_{IR}$  (0.5V)；
- ◇ 负端输入可选择外部输入电压、或外部输入电压/ $V_{DD}$  的内部分压电压；
- ◇ 输出端电平可选择上升沿或下降沿触发中断，可从端口输出且支持输出取反；
- ◇ 可实现  $V_{DD}$  或外部输入的电压检测（EVD）功能；



CMP 可实现 2 路外部输入电压（EVN0 与 CMPN）之间的比较，也可实现外部输入电压 CMPN 与内部基准电压  $V_{IR}$  (0.5V) 的比较，比较结果可通过输出状态标志位 CMPOF 读取。CMP 输出端电平可通过 CMPIM 选择上升或下降沿触发中断，并可通过 CMPOE 选择是否从端口输出。

CMP 还可实现  $V_{DD}$  或外部输入的电压检测（EVD）功能。通过寄存器位 EVDVS 选择电压检测阈值，实际是选择内部分压电阻的分压比例，分压后的电压与内部基准电压  $V_{IR}$  的比较结果，即为  $V_{DD}$  或外部输入电压与电压检测阈值的比较结果，从而实现电压检测功能。

注：

- 1、开启 CMP、切换输入通道或电压检测阈值等操作，需待电路稳定（时间  $>200\mu s$ ）后 CMP 输出才有效；
- 2、比较器支持无回滞输出，此时 CMP 输出受外部电路影响较大，应针对实际应用进行软件去抖处理；
- 3、内部分压电路采用串联电阻分压方式，即按照电压检测阈值换算出的电阻分压比，从 600 个  $3K\Omega$  电阻的串联电路中抽取对应的分压信号，因此对外部输入进行电压检测时，需确保外部电路不会影响内部电阻分压比；
- 4、CMP 应用中，需通过端口数模控制寄存器关闭相应端口的数字 I/O 功能，以防止产生漏电流；



## 8.2 CMP 相关寄存器

### 比较器控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
CMPCR0	CMPEN	CMPPS	CMPNS	EVDIS1	EVDIS0	CMPHY	CMPNV	CMPOE
R/W								
初始值	0	0	0	0	0	0	0	0

BIT[7] CMPEN – 比较器 CMP 使能位

- 0: 关闭 CMP;
- 1: 开启 CMP;

BIT[6] CMPPS – CMP 正端输入选择位

- 0: CMP 正端输入为内部基准电压  $V_{IR}$ ;
- 1: CMP 正端输入为端口 EVN0 输入电压;

BIT[5] CMPNS – CMP 负端输入选择位

- 0: CMP 负端输入为外部输入电压（或 VDD 电压）的内部分压电压;
- 1: CMP 负端输入为端口 CMPN 输入电压;

BIT[4:3] EVDIS[1:0] – EVD 被检测电压选择位

EVDIS[1:0]	EVD 被检测电压
00	VDD
01	端口 EVN0 输入电压
10	端口 EVN1 输入电压
11	端口 EVN2 输入电压

BIT[2] CMPHY – CMP 输出回滞控制位

- 0: CMP 输出无回滞;
- 1: CMP 输出有回滞（单边 6%）;



- BIT[1] CMPNV – CMPO 端口输出取反控制位  
 0: 端口输出 CMP 比较结果的正向电平;  
 1: 端口对 CMP 比较结果取反后输出;

- BIT[0] CMPOE – CMPO 端口输出使能位  
 0: 禁止端口输出 CMP 比较结果;  
 1: 允许端口输出 CMP 比较结果;

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
CMPCR1	-	-	EVDVS5	EVDVS4	EVDVS3	EVDVS2	EVDVS1	EVDVS0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	0	0	0	0	0	0

- BIT[5:0] EVDVS[5:0] – EVD 电压检测阈值选择位

EVDVS [5:0]	检测值 (V)	电阻 分压比									
00H	0.56	536/600	10H	1.68	179/600	20H	2.80	107/600	30H	3.92	77/600
01H	0.63	476/600	11H	1.75	171/600	21H	2.87	105/600	31H	3.99	75/600
02H	0.70	429/600	12H	1.82	165/600	22H	2.94	102/600	32H	4.06	74/600
03H	0.77	390/600	13H	1.89	159/600	23H	3.01	100/600	33H	4.13	73/600
04H	0.84	357/600	14H	1.96	153/600	24H	3.08	97/600	34H	4.20	71/600
05H	0.91	330/600	15H	2.03	148/600	25H	3.15	95/600	35H	4.27	70/600
06H	0.98	306/600	16H	2.10	143/600	26H	3.22	93/600	36H	4.34	69/600
07H	1.05	285/600	17H	2.17	138/600	27H	3.29	91/600	37H	4.41	68/600
08H	1.12	268/600	18H	2.24	134/600	28H	3.36	89/600	38H	4.48	67/600
09H	1.19	252/600	19H	2.31	130/600	29H	3.43	87/600	39H	4.55	66/600
0AH	1.26	238/600	1AH	2.38	126/600	2AH	3.50	86/600	3AH	4.62	65/600
0BH	1.33	226/600	1BH	2.45	122/600	2BH	3.57	84/600	3BH	4.69	64/600
0CH	1.40	214/600	1CH	2.52	119/600	2CH	3.64	82/600	3CH	4.76	63/600
0DH	1.47	204/600	1DH	2.59	116/600	2DH	3.71	81/600	3DH	4.83	62/600
0EH	1.54	195/600	1EH	2.65	113/600	2EH	3.78	79/600	3EH	4.90	61/600
0FH	1.61	186/600	1FH	2.73	110/600	2FH	3.85	78/600	3FH	4.97	60/600

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
--	--------	--------	--------	--------	--------	--------	--------	--------



CMPCR2	-	-	-	-	-	CMPIM	CMPIE	CMPIF
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2]      CMPIM – CMP 中断触发方式选择位

0: CMP 输出的上升沿触发中断;

1: CMP 输出的下降沿触发中断;

BIT[1]      CMPIE – 比较器 CMP 中断使能位

0: 屏蔽 CMP 中断;

1: 使能 CMP 中断;

BIT[0]      CMPIF – 比较器 CMP 中断标志位

0: 未触发 CMP 中断;

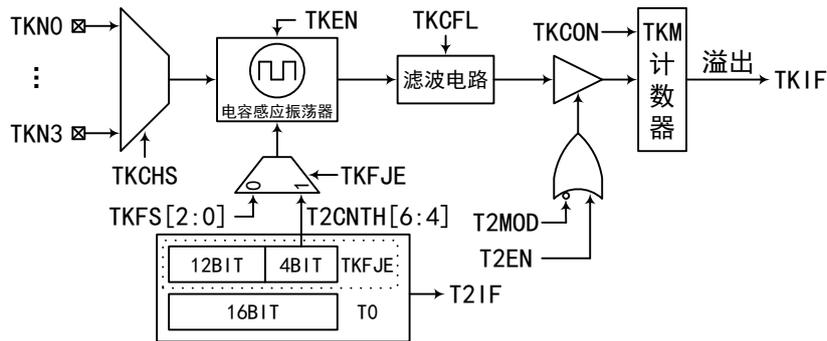
1: 已触发 CMP 中断, 需软件清 0;



## 9 触摸按键模块 TKM

### 9.1 TKM 概述

芯片内置 1 个带有电容充放电电路的触摸按键模块 TKM，支持 4 路触摸按键输入 (TKN0~TKN3)。通过 16 位定时器 T2 和 16 位电容充放电递增计数器 TKCNT，分时监控外部触摸按键电路的电容循环充放电的频率，从而根据频率变化识别外部按键状态。



注:

- 1、T2 设置为触摸模式时，T2EN 写 1 启动 T2 计时，T2 溢出后将自动清零 T2EN 以关闭 T2；
- 2、T2 设置为触摸模式时，TKCNT 开始计数的条件为 TKEN=1、TKCON=1 后 T2EN 写入 1，当 T2 溢出后将自动停止计数；T2 设置为定时模式时，TKCNT 开始计数的条件为 TKEN=1 后 TKCON 写入 1，而 TKCON 写 0 则停止计数并同时清零 TKCNT，所以此时需要先读取 TKCNT 的值再清零 TKCON；
- 3、TKCFL 选择计数器 TKCNT 输入时钟的滤波能力，可以滤除触摸按键引脚的杂波信号，提高按键抗干扰性。应用中可根据实测结果选择合适的滤波能力；
- 4、T2 设置为触摸模式且 TKM 自动跳频功能开启 (TKFJE=1) 后，T2 将配置为 4bit×12bit 工作模式，即每次 T2CNT[11:0] 递减至 0 溢出时均重新加载为 T2LOAD[11:0]，而同时 T2CNT[15:12] 仍为自动减 1 而非重新加载，直至整个 16 位 T2CNT 递减计数至 0，产生溢出中断；
- 5、自动跳频时，跳频所选频率不再由 TKFS 设置，而是通过 T2CNT[14:12] 与自动跳频控制位 TKFJ7-TKFJ0 决定；

### 9.2 TKM 相关寄存器

TKM 控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
TKCR0	TKEN	TKCON	TKCFL	-	TKFJE	TKFS2	TKFS1	TKFS0
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	R/W
初始值	0	0	0	-	0	0	0	0



- BIT[7] TKEN – TKM 使能位  
 0: 关闭 TKM;  
 1: 开启 TKM;
- BIT[6] TKCON – TKM 电容充放计数控制位  
 0: 禁止 TKM 电容充放计数并清零计数器;  
 1: 允许 TKM 电容充放计数;
- BIT[5] TKCFL – TKM 计数时钟滤波时间选择位  
 0: TKM 计数时钟滤波时间 (典型值) 为 10ns;  
 1: TKM 计数时钟滤波时间 (典型值) 为 50ns;
- BIT[3] TKFJE – TKM 自动跳频使能位  
 0: 关闭 TKM 自动跳频功能;  
 1: 开启 TKM 自动跳频功能;
- BIT[2:0] TKFS[2:0] – TKM 振荡频率选择位

TKFS[2:0]	TKM 振荡频率
000	8.0MHz
001	7.2MHz
010	6.0MHz
011	5.4MHz
100	4.8MHz
101	2.6MHz
110	1.0MHz
111	0.5MHz

注: 上述频率值为引脚外接 15pF 电容时设计值, 实际值与引脚寄生电容有关。

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
TKCR1	TKIE	TKIF	TKCF	-	-	-	TKCHS1	TKCHS0
R/W	R/W	R/W	R/W	-	-	-	R/W	R/W
初始值	0	0	0	-	-	-	0	0

- BIT[7] TKIE – TKM 计数中断使能位



- 0: 屏蔽 TKM 计数中断;
- 1: 使能 TKM 计数中断;

BIT[6] TKIF – TKM 计数中断标志位

- 0: 未触发 TKM 计数中断;
- 1: 已触发 TKM 计数中断, 需软件清 0;

BIT[5] TKCF – TKM 电容充放计数状态标志位

- 0: TKM 停止电容充放计数;
- 1: TKM 电容充放计数中;

BIT[1:0] TKCHS[1:0] – 触摸按键输入通道选择位

TKCHS[1:0]	触摸按键输入通道
00	TKN0
01	TKN1
10	TKN2
11	TKN3

注: 当端口用于触摸按键输入时, 需先关闭端口的数字 I/O 功能及其内部上/下拉电阻。

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
TKCR2	TKFJ7	TKFJ6	TKFJ5	TKFJ4	TKFJ3	TKFJ2	TKFJ1	TKFJ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

BIT[7] TKFJ7 – 0.5MHz 振荡频率参与自动跳频控制位

- 0: 0.5MHz 振荡频率不参与自动跳频;
- 1: 0.5MHz 振荡频率参与自动跳频;

BIT[6] TKFJ6 – 1.0MHz 振荡频率参与自动跳频控制位

- 0: 1.0MHz 振荡频率不参与自动跳频;
- 1: 1.0MHz 振荡频率参与自动跳频;

BIT[5] TKFJ5 – 2.6MHz 振荡频率参与自动跳频控制位

- 0: 2.6MHz 振荡频率不参与自动跳频;



1: 2.6MHz 振荡频率参与自动跳频;

BIT[4] TKFJ4 – 4.8MHz 振荡频率参与自动跳频控制位

0: 4.8MHz 振荡频率不参与自动跳频;

1: 4.8MHz 振荡频率参与自动跳频;

BIT[3] TKFJ3 – 5.4MHz 振荡频率参与自动跳频控制位

0: 5.4MHz 振荡频率不参与自动跳频;

1: 5.4MHz 振荡频率参与自动跳频;

BIT[2] TKFJ2 – 6.0MHz 振荡频率参与自动跳频控制位

0: 6.0MHz 振荡频率不参与自动跳频;

1: 6.0MHz 振荡频率参与自动跳频;

BIT[1] TKFJ1 – 7.2MHz 振荡频率参与自动跳频控制位

0: 7.2MHz 振荡频率不参与自动跳频;

1: 7.2MHz 振荡频率参与自动跳频;

BIT[0] TKFJ0 – 8.0MHz 振荡频率参与自动跳频控制位

0: 8.0MHz 振荡频率不参与自动跳频;

1: 8.0MHz 振荡频率参与自动跳频;

#### TKM 电容充放计数器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
TKCNTH	TKCNT15	TKCNT14	TKCNT13	TKCNT12	TKCNT11	TKCNT10	TKCNT9	TKCNT8
R/W	R	R	R	R	R	R	R	R
初始值	0	0	0	0	0	0	0	0

BIT[7:0] TKCNT[15:8] – TKM 电容充放计数器高 8 位，为只读的递增计数器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
TKCNTL	TKCNT7	TKCNT6	TKCNT5	TKCNT4	TKCNT3	TKCNT2	TKCNT1	TKCNT0
R/W	R	R	R	R	R	R	R	R
初始值	0	0	0	0	0	0	0	0



BIT[7:0] TKCNT[7:0] – TKM 电容充放计数器低 8 位，为只读的递增计数器

### 9.3 TKM 操作步骤

触摸按键模块的操作步骤：

- (1) 设置相应端口为输入端口，关闭端口的内部上/下拉电阻；
- (2) 通过端口数模控制寄存器，关闭相应端口的数字 I/O 功能；
- (3) 设置 T2MOD 选择 T2 为单次计时模式；
- (4) 设置 TKEN 开启 TKM 及其电容感应振荡器；
- (5) 设置 TKFL 选择 TKCNT 计数时钟的滤波时间；
- (6) 设置 TKCHS 选择相应按键输入通道；
- (7) 设置 TKFS 选择相应振荡频率；
- (8) 设置 TKCON 为 1 允许 TKCNT 计数；
- (9) 设置 T2EN 为 1 启动 T2 计时并使能 TKM 计数时钟；
- (10) 等待 T2IF 置 1 或利用 T2 中断或等待 T2EN 清零；
- (11) 读取 TKM 计数结果 (TKCNTH/TKCNTL)；
- (12) 设置 TKCON 为 0，并执行 TKCNTH 和 TKCNTL 的写零操作，清零计数结果；
- (13) 重复 (6) ~ (12) 对不同的通道进行转换或对同一通道多次转换；



## 10 中断

芯片的中断源包括外部中断 (INT)、定时器中断 (T0~T2)、CMP 中断和键盘中断等。可通过中断总使能位 GIE 屏蔽所有中断。

CPU 响应中断的过程如下：

- ◇ CPU 响应中断源触发的中断请求时，自动将当前指令之后将要执行的下一条指令的地址压栈保存，自动清 0 中断总使能位 GIE 以暂停响应后续中断。与复位不同，硬件中断不停止当前指令的执行，而是暂时挂起中断继续执行当前指令，完成后再处理中断。
- ◇ CPU 响应中断后，程序跳至中断入口地址 (0008H) 开始执行中断服务程序，中断服务程序应先保存累加器 A 和状态寄存器 STATUS，然后处理被触发的中断。
- ◇ 中断服务程序处理完中断后，应先恢复累加器 A 和状态寄存器 STATUS，再执行 RETIE 指令以返回主程序。系统将自动恢复 GIE 为 1，然后从堆栈取出此前保存的 PC 值，CPU 从响应中断时正在执行指令的下一条指令的地址处开始继续运行。

*注：应用外部中断功能或键盘中断功能，需将相应端口设为输入状态。*

### 10.1 外部中断

芯片具有 1 路外部中断源 INT，可通过寄存器位 INTM 选择上升沿或下降沿等触发方式。外部中断触发时，中断标志 INTIF 将被置 1，若 GIE 为 1 且相应的外部中断使能位 INTIE 为 1，则产生外部中断。

*注：若通过寄存器位 EIS 置 1 使能端口的外部中断功能，则端口的键盘中断唤醒功能无效。*

### 10.2 定时器中断

定时器 T<sub>n</sub> (n=0-2) 在计数溢出时将触发定时器中断，中断标志 T<sub>n</sub>IF (n=0-2) 将被置 1，若 GIE 为 1 且相应的定时器中断使能位 T<sub>n</sub>IE (n=0-2) 为 1，则产生定时器中断。



### 10.3 键盘中断

芯片具有 6 路键盘中断源，均可单独使能或关闭端口的键盘中断功能。任意一路使能键盘中断功能的端口，其输入电平发生变化时均将触发键盘中断，中断标志 KBIF 将被置 1，若 GIE 为 1 且键盘中断使能位 KBIE 为 1，则产生键盘中断。

#### 键盘中断控制寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
P1KBCR	-	-	P15KE	P14KE	P13KE	P12KE	P11KE	P10KE
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	0	0	0	0	0	0

BIT[5:0] P1nKE – P1n 端口键盘中断功能使能位 (n=5-0)

0: 关闭端口的键盘中断功能;

1: 使能端口的键盘中断功能;

### 10.4 TKM 计数中断

触摸按键模块 TKM 在计数器 TKCNT 溢出时将触发 TKM 计数中断，中断标志 TKIF 将被置 1，若 GIE 为 1 且 TKM 计数中断使能位 TKIE 为 1，则产生 TKM 计数中断。

### 10.5 CMP 中断

比较器 CMP 的输出可选择上升沿或下降沿等方式触发比较器中断，中断标志 CMPIF 将被置 1，若 GIE 为 1 且比较器中断使能位 CMPIE 为 1，则产生比较器中断。

### 10.6 中断相关寄存器



## 中断使能寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
INTE	GIE	-	-	-	-	INTIE	KBIE	TOIE
R/W	R/W	-	-	-	-	R/W	R/W	R/W
初始值	0	-	-	-	-	0	0	0

BIT[7] GIE – 中断总使能位

0: 屏蔽所有中断;

1: 由相应的中断使能位决定 CPU 是否响应中断源所触发的中断;

BIT[2] INTIE – INT 中断使能位

0: 屏蔽 INT 中断;

1: 使能 INT 中断;

BIT[1] KBIE – 键盘中断使能位

0: 屏蔽键盘中断;

1: 使能键盘中断;

BIT[0] TOIE – 定时器 T0 中断使能位

0: 屏蔽定时器 T0 中断;

1: 使能定时器 T0 中断;

## 中断标志寄存器

	Bit[7]	Bit[6]	Bit[5]	Bit[4]	Bit[3]	Bit[2]	Bit[1]	Bit[0]
INTF	-	-	-	-	-	INTIF	KBIF	TOIF
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2] INTIF – INT 中断标志位

0: 未触发 INT 中断;

1: 已触发 INT 中断, 需软件清 0;

BIT[1] KBIF – 键盘中断标志位

0: 未触发键盘中断;

1: 已触发键盘中断, 需软件清 0;



BIT[0]      T0IF - 定时器 T0 中断标志位  
0: 未触发定时器 T0 中断;  
1: 已触发定时器 T0 中断, 需软件清 0;



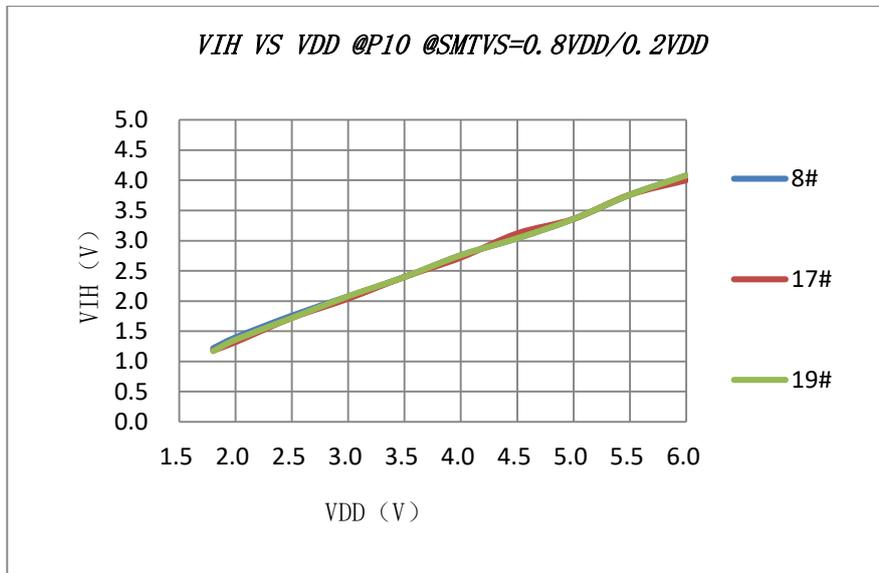
## 11 特性曲线

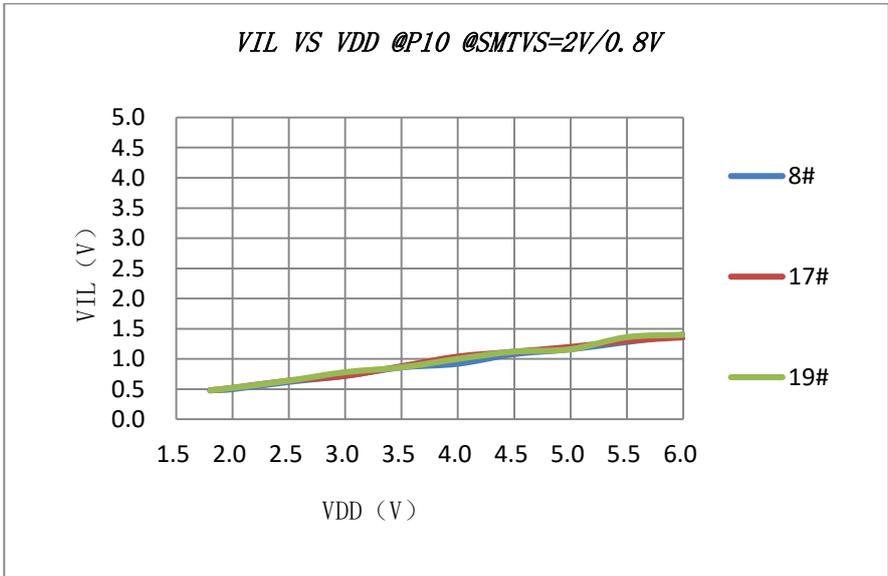
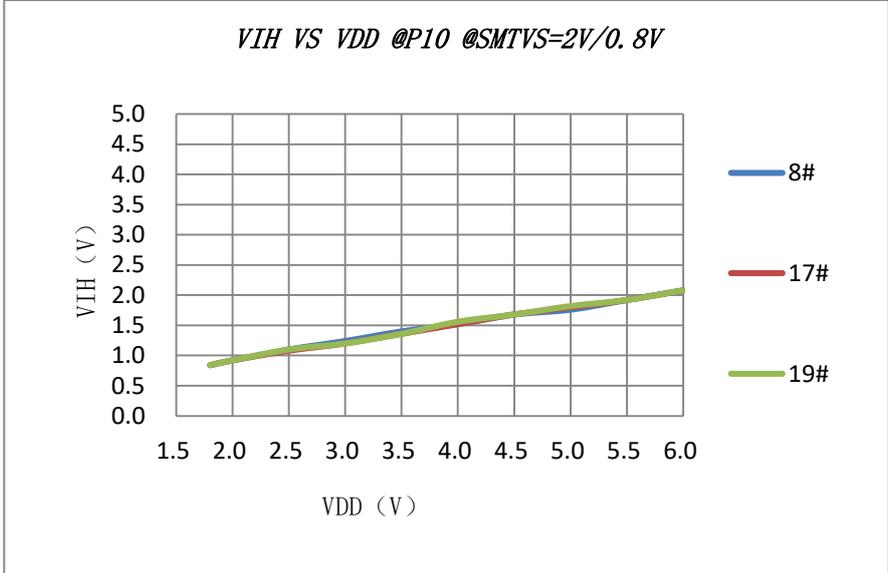
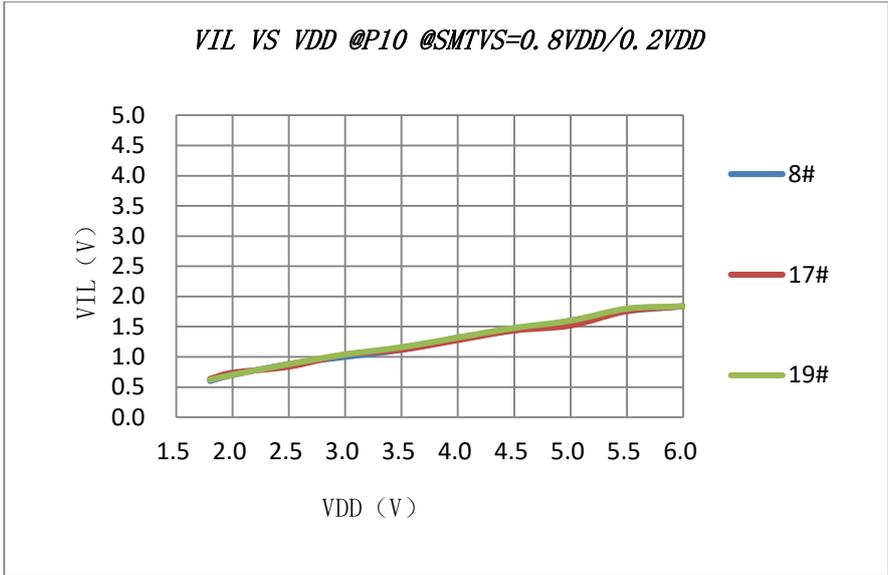
注:

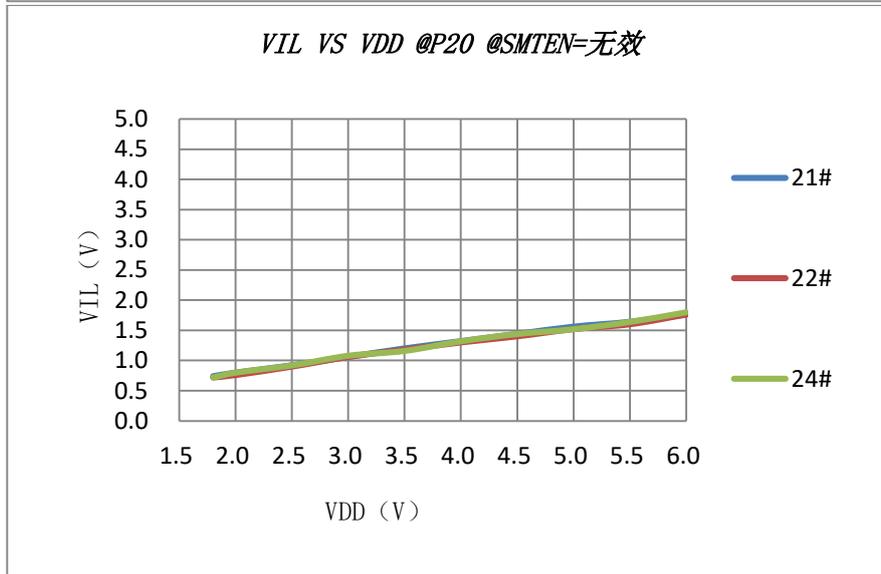
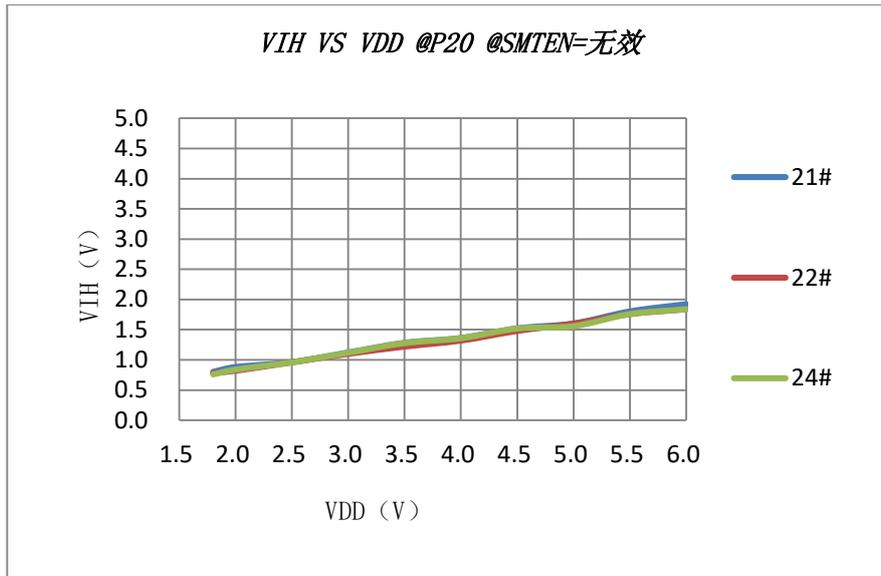
- 1、特性曲线图中数据均源自抽样实测，仅作为应用参考，部分数据因生产工艺偏差，可能与实际芯片不符；为保证芯片能正常工作，请确保其工作条件符合电气特性参数说明；
- 2、图文中若无特别说明，则电压特性曲线的温度条件为  $T=25^{\circ}\text{C}$ ，温度特性曲线的电压条件为  $VDD=5\text{V}$ ；

### 11.1 I/O 特性

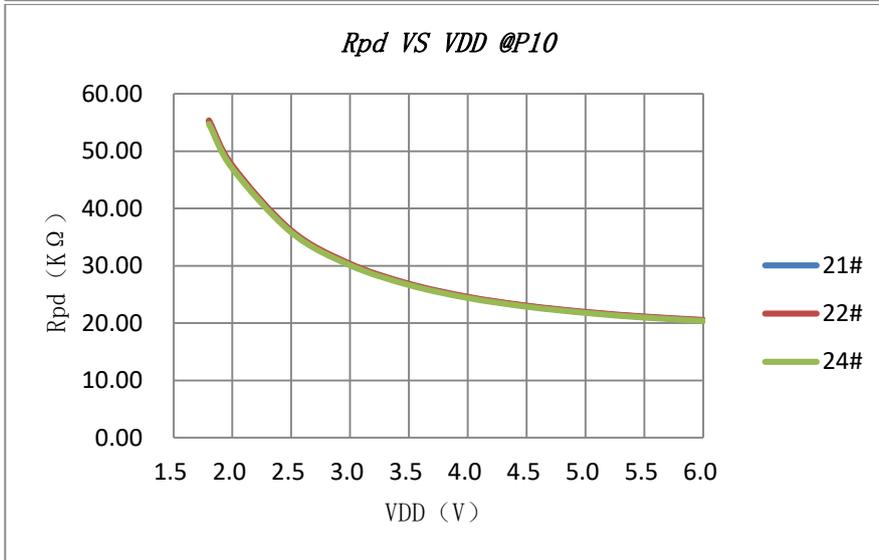
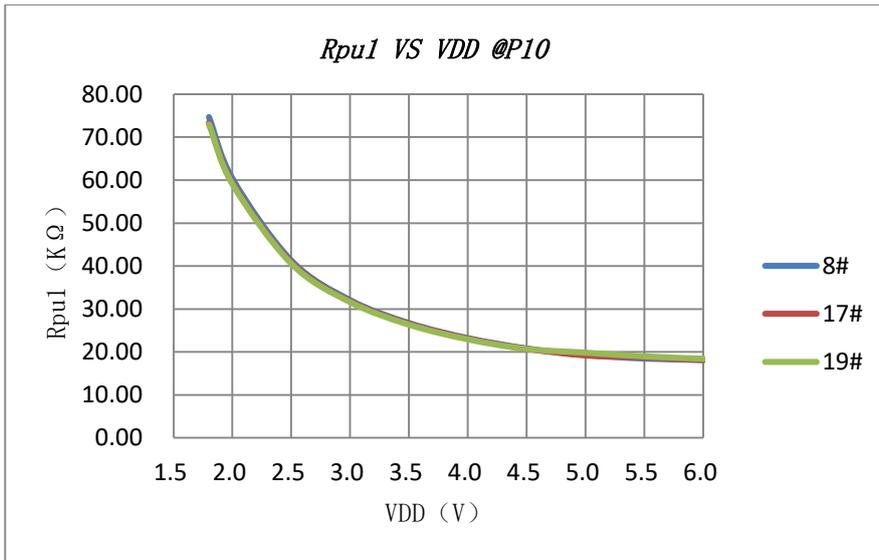
输入 SMT 阈值电压 VS 电源电压



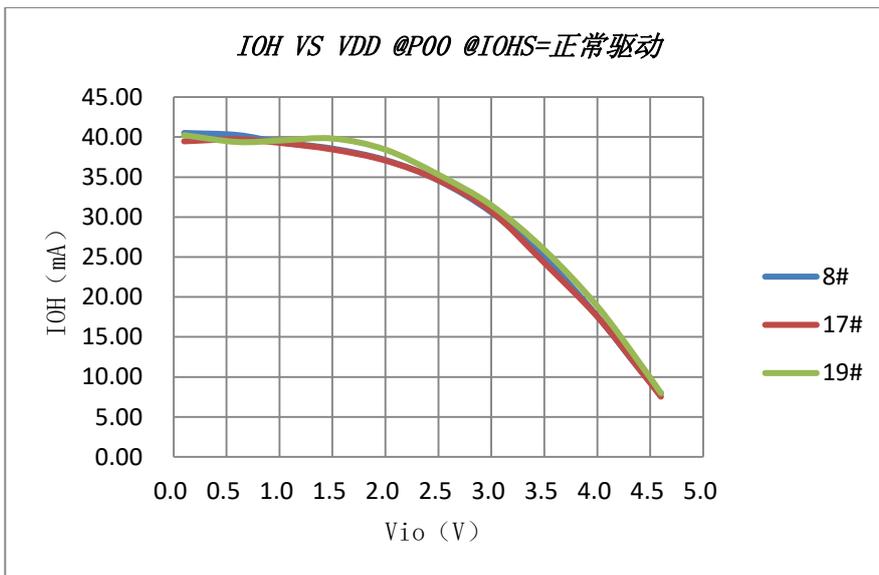


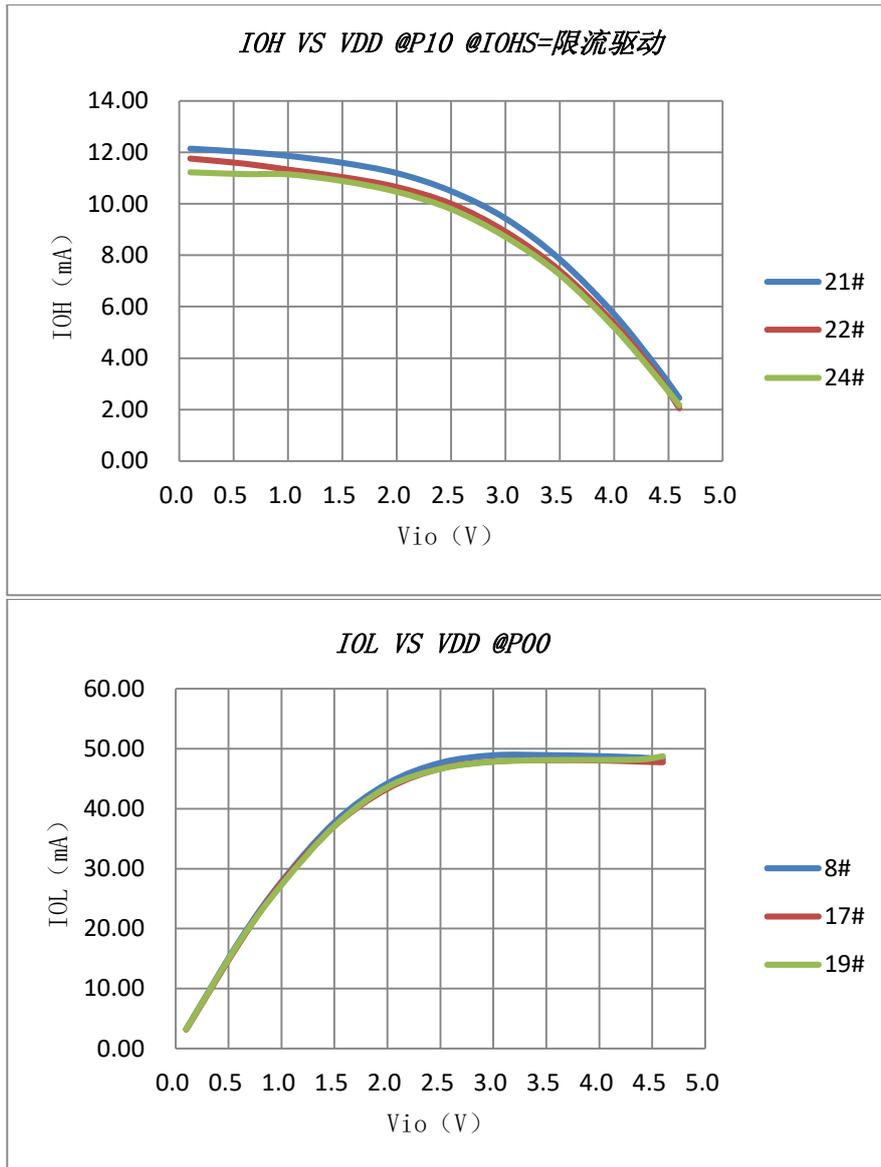


上/下拉电阻值 VS 电源电压



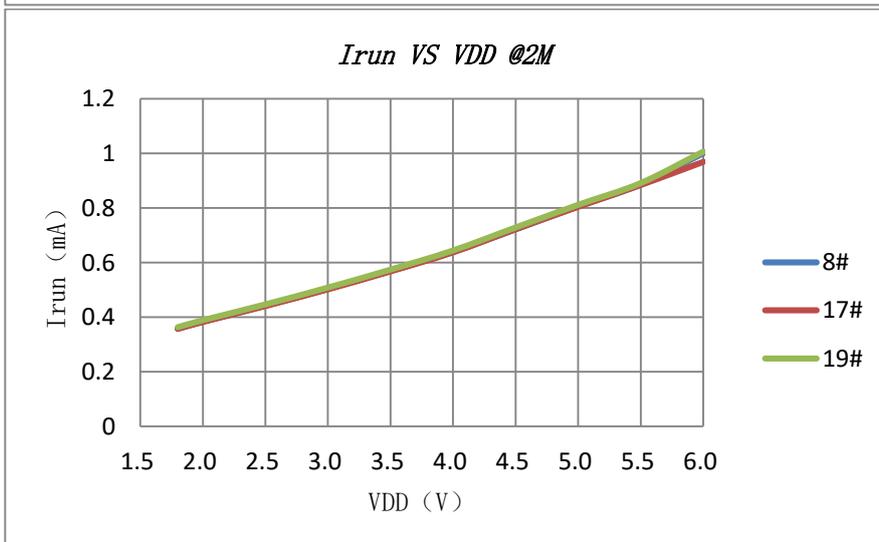
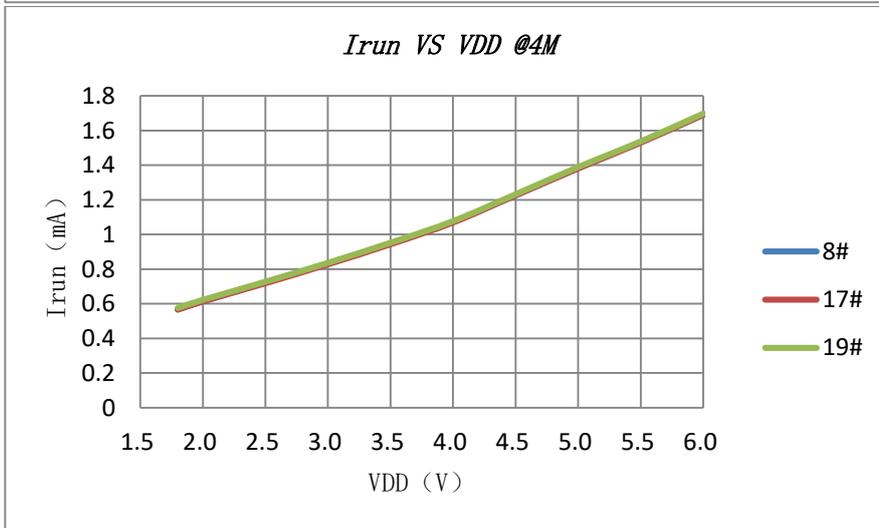
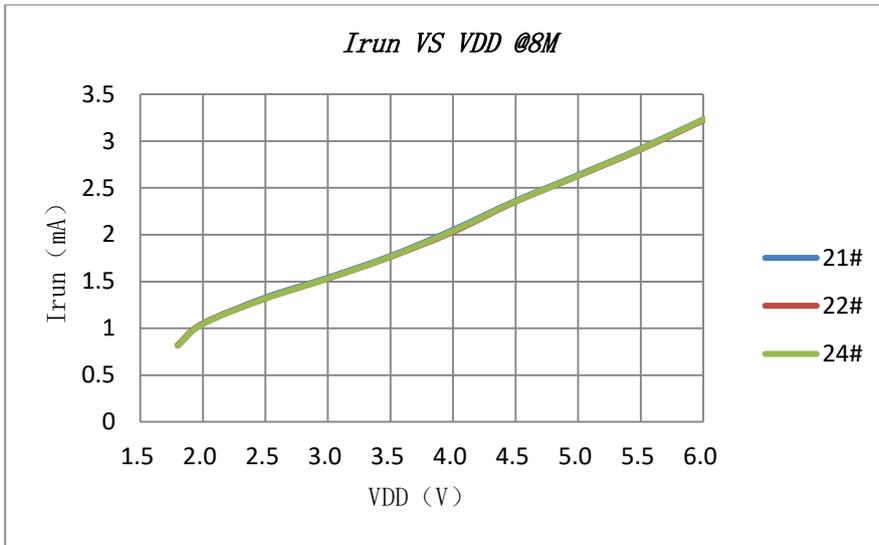
I/O 输出电流 VS 端口电压 (VDD=5V)

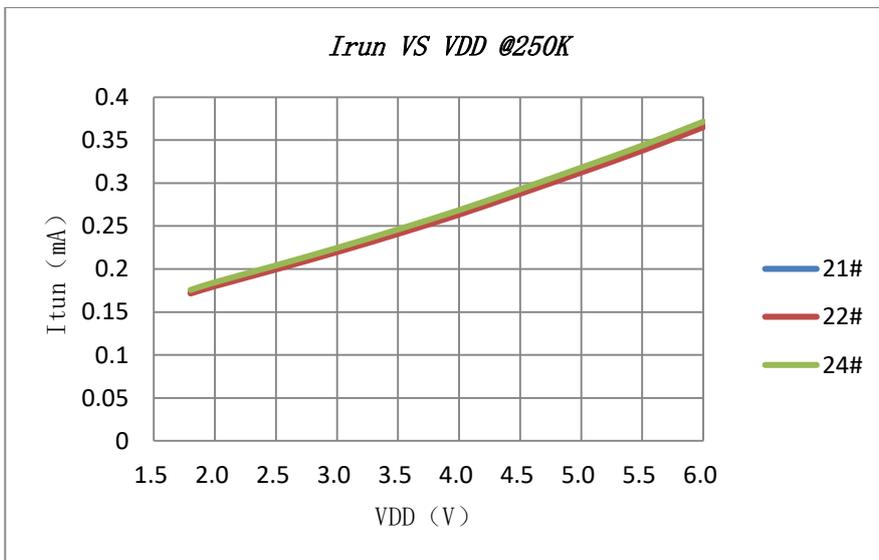
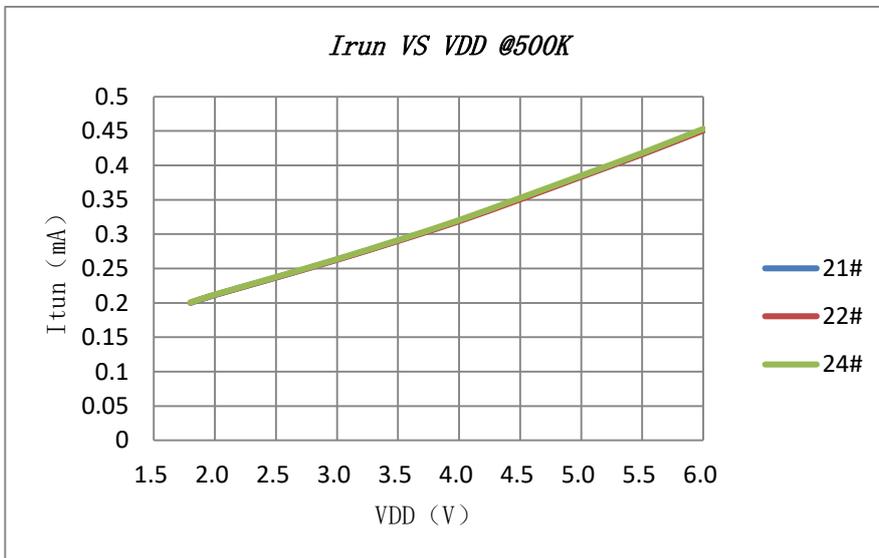


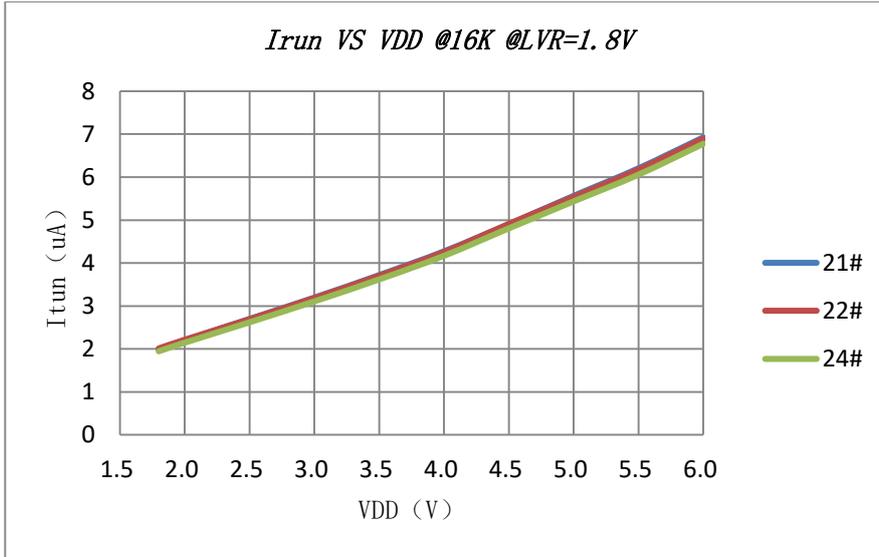


## 11.2 功耗特性

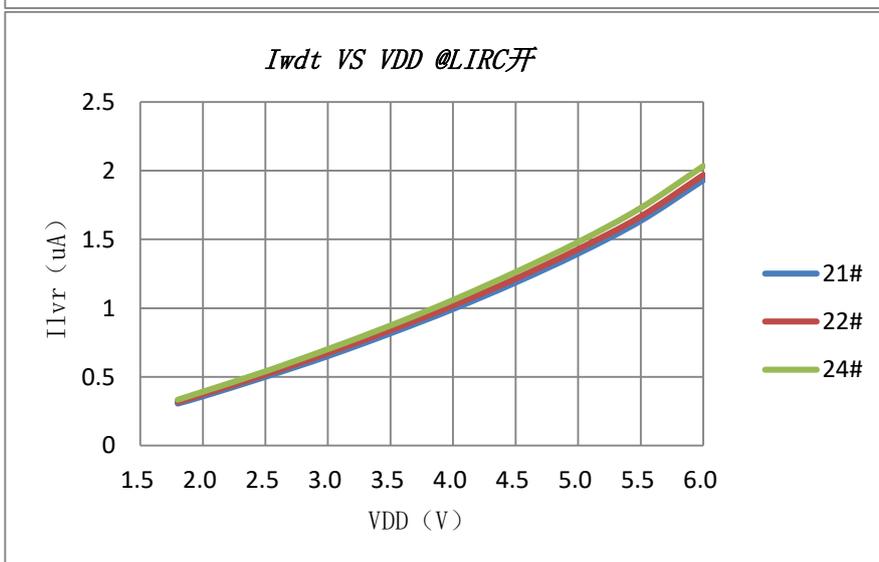
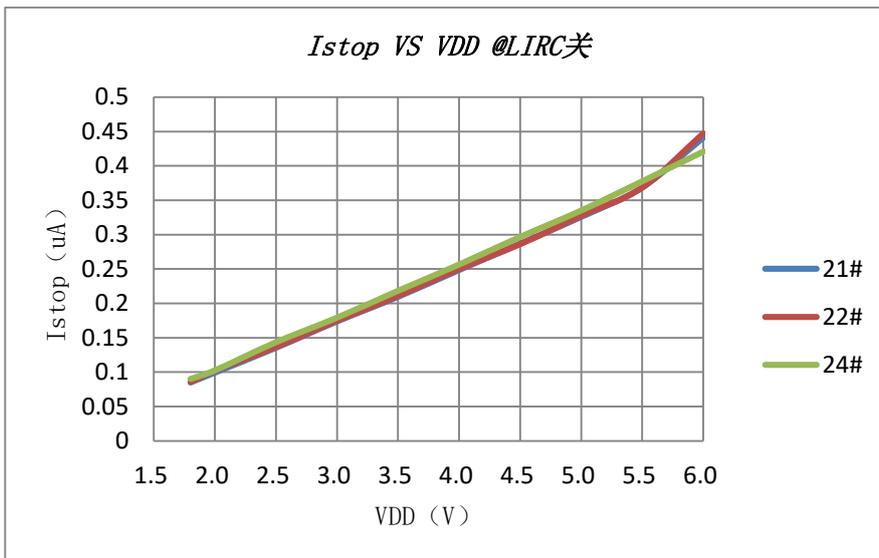
运行模式 功耗 VS 电源电压







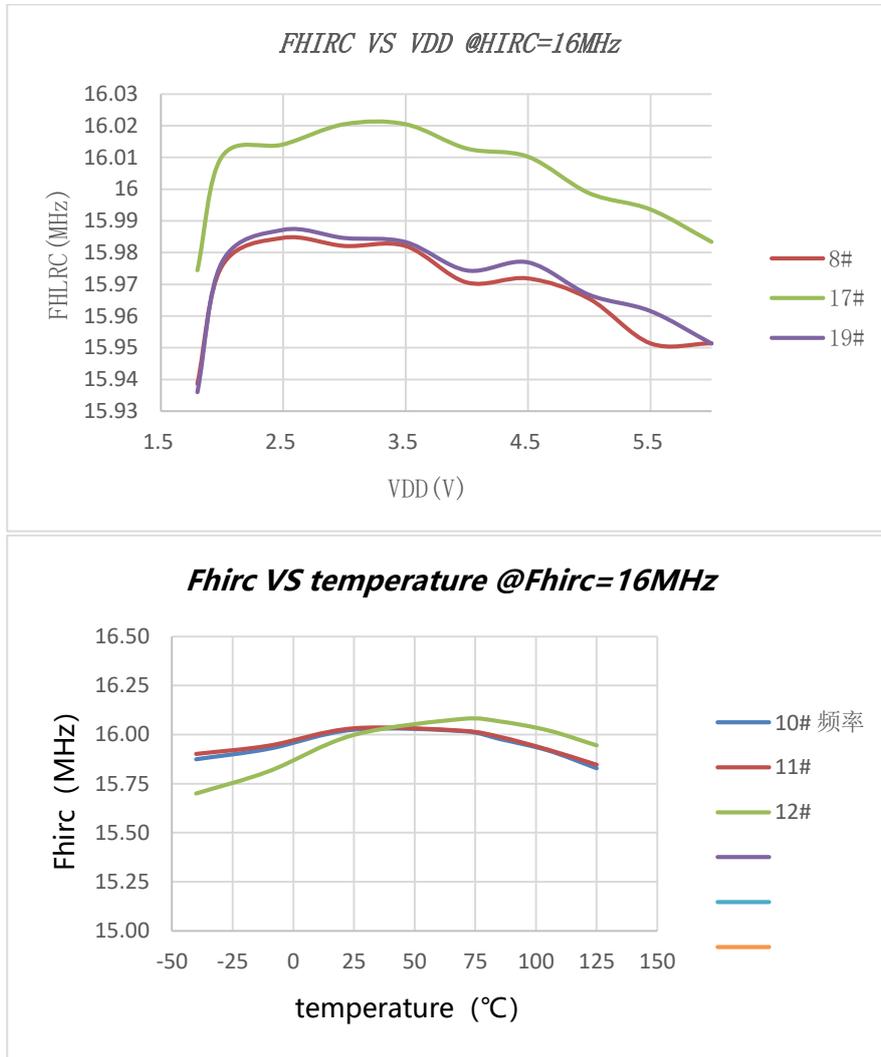
休眠模式 功耗 VS 电源电压



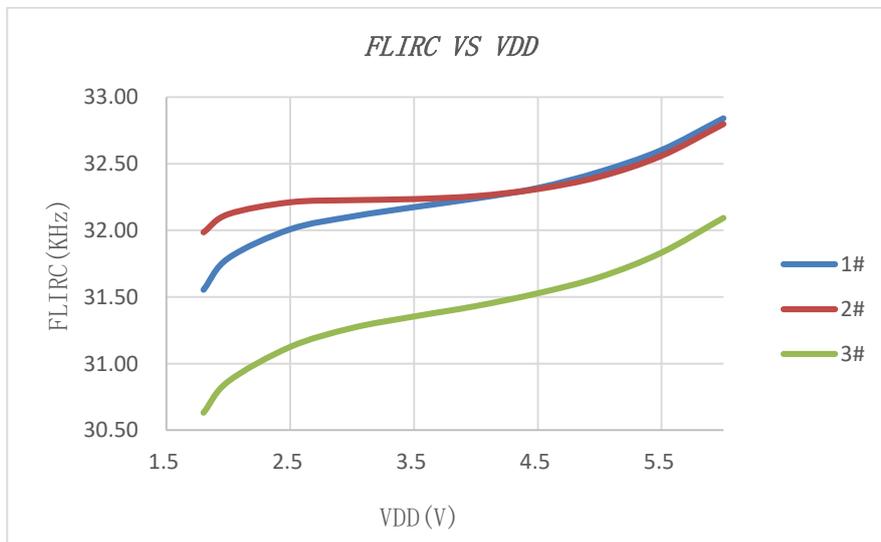


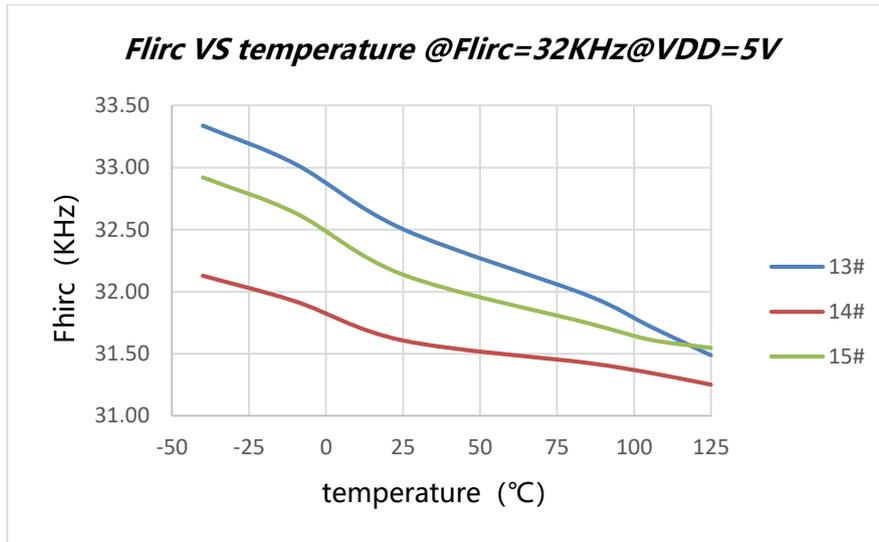
### 11.3 模拟电路特性

HIRC 频率 VS 电源电压/温度



LIRC 频率 VS 电源电压/温度

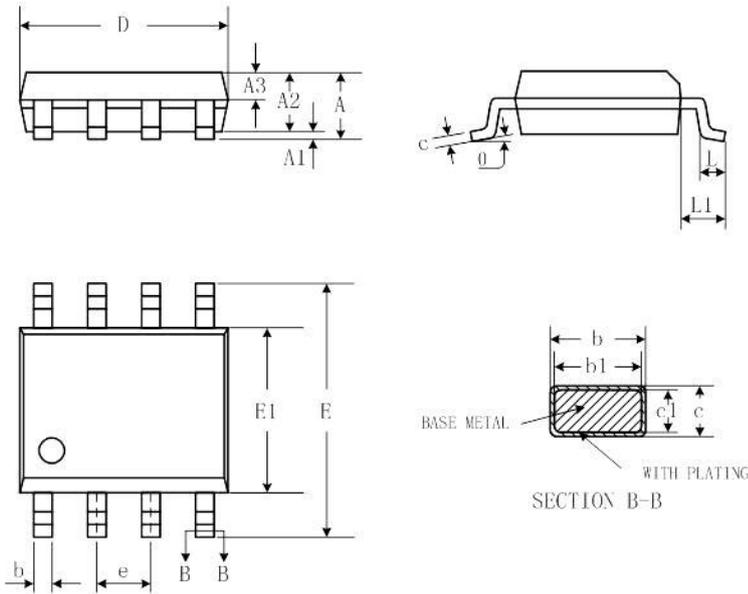






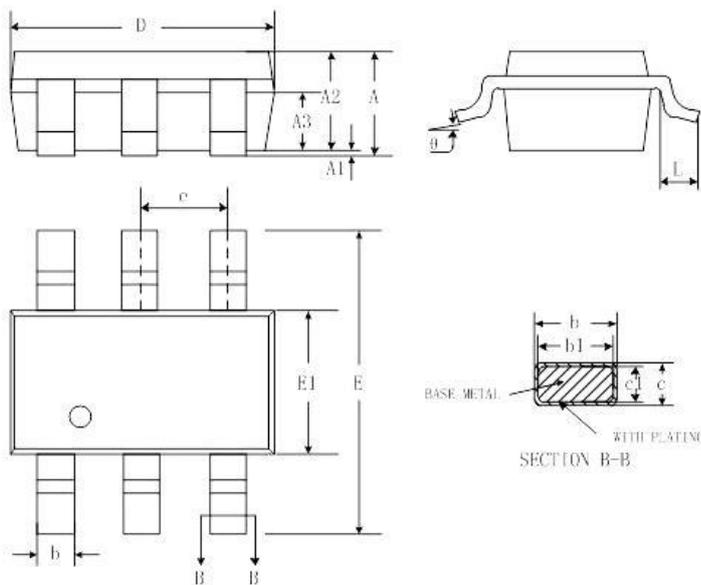
## 12 封装尺寸

### 12.1 SOP8



SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	-	-	1.77
A1	0.08	0.18	0.28
A2	1.20	1.40	1.60
A3	0.55	0.65	0.75
b	0.39	-	0.48
b1	0.38	0.41	0.43
c	0.21	-	0.26
c1	0.19	0.20	0.21
D	4.70	4.90	5.10
E	5.80	6.00	6.20
E1	3.70	3.90	4.10
e	1.27BSC		
L	0.50	0.65	0.80
L1	1.05BSC		
θ	0	-	8°

### 12.2 SOT23-6



SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	-	-	1.35
A1	0.04	-	0.15
A2	1.00	1.10	1.20
A3	0.55	0.65	0.75
b	0.30	-	0.50
b1	0.30	0.40	0.45
c	0.08	-	0.22
c1	0.08	0.13	0.20
D	2.72	2.92	3.12
E	2.60	2.80	3.00
E1	1.40	1.60	1.80
e	0.95BSC		
L	0.30	-	0.60
θ	0	-	8°



## 13 修订记录

版本	日期	修订内容
V1.0	2023-03-09	发布初版;
V1.1	2023-07-25	修正系统时钟示意图中笔误; 调整 LVR 电压精度范围为±15%;
V1.2	2024-07-05	补充 VPP/P14 脚不支持[源电流限流输出]功能相关描述;