

SinoMCU 8 位单片机

MC32P7331

用户手册

V1.0



上海晟矽微电子股份有限公司

Shanghai SinoMCU Microelectronics Co., Ltd.

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	7
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	ADC 特性参数	10
3	CPU 及存储器	11
3.1	指令集	11
3.2	程序存储器	13
3.3	数据存储器	14
3.4	堆栈	15
3.5	控制寄存器	15
3.6	用户配置字	18
4	系统时钟	20
4.1	内部高频 RC 振荡器	20
4.2	内部低频 RC 振荡器	21
4.3	外部晶体振荡器	22
4.4	系统工作模式	22
4.5	低功耗模式	24
5	复位	25
5.1	复位条件	25
5.2	上电复位	26
5.3	外部复位	26
5.4	低电压复位	26
5.5	看门狗复位	26
6	I/O 端口	27
6.1	通用 I/O 功能	27
6.2	内部上/下拉电阻	28
6.3	端口模式控制	29
7	定时器 TIMER	31
7.1	看门狗定时器 WDT	31
7.2	定时器 T0	31
7.3	定时器 T1	34
7.4	定时器 T2	37
7.5	定时器 T3	40
7.6	PWM 相关寄存器	42
8	模数转换器 ADC	45
8.1	ADC 概述	45

8.2	ADC 相关寄存器.....	46
8.3	ADC 操作步骤.....	48
9	低电压检测 LVD	49
10	中断.....	50
10.1	外部中断.....	50
10.2	定时器中断.....	50
10.3	ADC 中断.....	50
10.4	键盘中断.....	50
10.5	中断相关寄存器.....	51
11	特性曲线.....	53
11.1	I/O 特性.....	53
11.2	功耗特性.....	58
11.3	模拟电路特性.....	62
12	封装尺寸.....	68
12.1	SOP16.....	68
12.2	SOP14.....	68
12.3	DIP14	69
12.4	SOP8.....	69
12.5	DIP8	70
12.6	MSOP10.....	70
13	修订记录.....	71

1 产品概要

1.1 产品特性

- 8 位 CPU 内核
 - ◇ 精简指令集，8 级深度硬件堆栈
 - ◇ CPU 为双时钟，可在系统高/低频时钟之间切换
 - ◇ 系统高频时钟下 F_{CPU} 可配置为 4/8/16/32/64/128/256 分频
 - ◇ 系统低频时钟下 F_{CPU} 固定为 2 分频
- 存储器
 - ◇ 2K×16 位 OTP 型程序存储器，可通过间接寻址读取程序存储器内容
 - ◇ 256 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
- 2 组共 14 个 I/O
 - ◇ P0 (P00~P05)，P1 (P10~P17)
 - ◇ P13 为输入/开漏输出口，可复用为外部复位 RST 输入，编程时为高压 VPP 输入
 - ◇ P00/P01 可复用为外部时钟振荡器输入/输出
 - ◇ 所有端口均内置上拉和下拉电阻，均可单独使能/禁用
 - ◇ P00/P01 可选开漏或推挽输出
 - ◇ P16/P17 输出电流 4 级可选，其余端口输出电流 2 级可选
 - ◇ P1 所有端口均支持键盘中断唤醒功能
- 系统时钟源
 - ◇ 内置高频 RC 振荡器 (32MHz)，可用作系统高频时钟源，支持软件微调
 - ◇ 内置低频 RC 振荡器 (32KHz)，可用作系统低频时钟源
 - ◇ 支持外接低频晶体振荡器 (32768Hz)，可用作系统低频时钟源
- 系统工作模式
 - ◇ 高速模式：CPU 在高频时钟下运行，低频时钟源工作
 - ◇ 低速模式：CPU 在低频时钟下运行，高频时钟源可选停止或工作
 - ◇ HOLD1 模式：CPU 停止运行，高频时钟源工作
 - ◇ HOLD2 模式：CPU 停止运行，高频时钟源停止工作，低频时钟源工作
 - ◇ 休眠模式：CPU 停止运行，所有时钟源停止工作
- 内部自振式看门狗计数器 (WDT)
 - ◇ 溢出时间可配置：16ms/64ms/256ms/1024ms/2048ms/4096ms
 - ◇ 工作模式可配置：始终开启、始终关闭、低功耗模式下关闭
- 4 个定时器
 - ◇ 8 位定时器 T0，可实现外部计数、BUZ、8+3 模式 PWM (2 路输出通道可选)
 - ◇ 8 位定时器 T1，可实现外部计数、BUZ、8+3 模式 PWM (1 对死区可设的互补 PWM)
 - ◇ 8 位定时器 T2，可实现 BUZ、8+3 模式 PWM (4 路死区可设的互补加反向 PWM)
 - ◇ 8 位定时器 T3
- 1 个 12 位高精度逐次逼近型 ADC
 - ◇ 11 路外部通道：AN0~AN10；3 路内部通道：VDD/4、OUTV0/4、OUTV1/4
 - ◇ 参考电压可选：VDD、内部参考电压 V_{IR} (2V/3V/4V)、外部参考电压 V_{ER} (VERI 输入)
 - ◇ ADC 时钟：F_{HIRC} 的 32/64/128/256 分频
 - ◇ VERI 端口内部 3KΩ 上拉可选

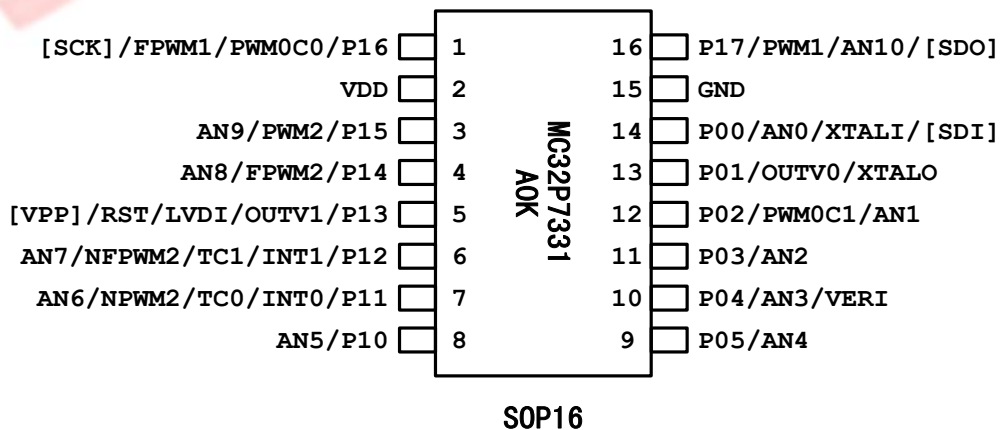
- 中断
 - ◇ 外部中断 (INT0~INT1), 键盘中断 (P10~P17)
 - ◇ 定时器中断 (T0~T3), ADC 中断
- 低电压复位 LVR: 2.2V/2.4V/3.6V
- 低电压检测 LVD
 - ◇ 1.8V/2.0V/2.1V/2.2V/2.4V/2.5V/2.6V/2.7V/2.8V/3.0V/3.2V/3.3V/3.6V/4.0V/4.2V
 - ◇ 可选择 LVDI 输入电压与内部 1V 比较
- 工作电压
 - ◇ $V_{LVR36} \sim 5.5V @ F_{cpu} = 0 \sim 8MHz$
 - ◇ $V_{LVR24} \sim 5.5V @ F_{cpu} = 0 \sim 4MHz$
 - ◇ $V_{LVR22} \sim 5.5V @ F_{cpu} = 0 \sim 2MHz$
- 封装形式: SOP16/SOP14/DIP14/SOP8/DIP8/MSOP10

1.2 订购信息

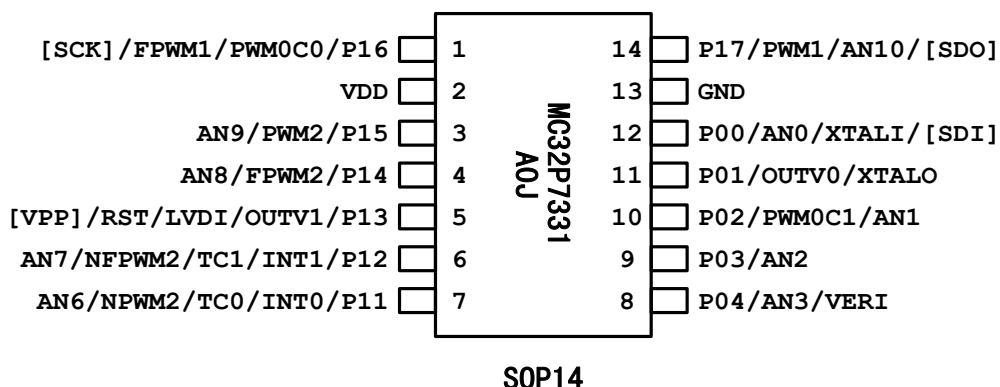
产品名称	封装形式	备注
MC32P7331A0K	SOP16	
MC32P7331A0J	SOP14	
MC32P7331A1J	SOP14	
MC32P7331A1B	DIP14	
MC32P7331A0I	MSOP10	
MC32P7331A0H	SOP8	
MC32P7331A0A	DIP8	
MC32P7331A1H	SOP8	

1.3 引脚排列

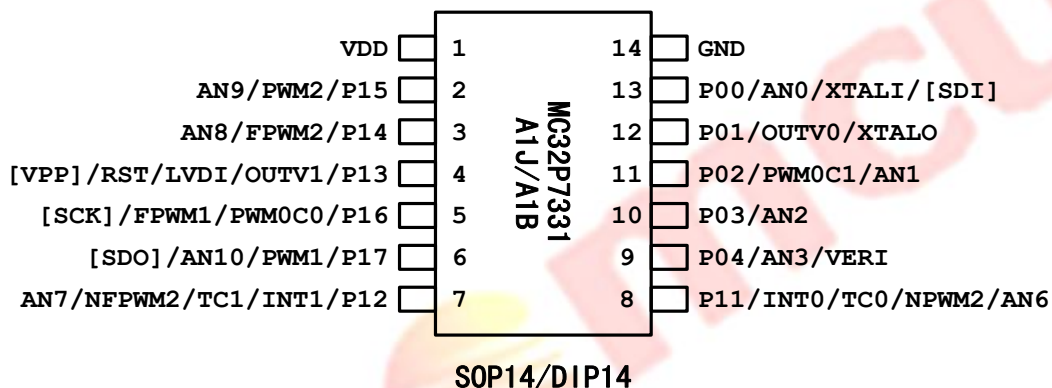
MC32P7331A0K



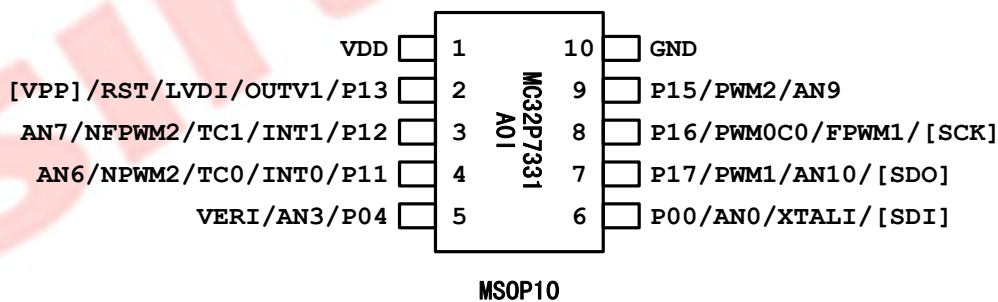
MC32P7331A0J



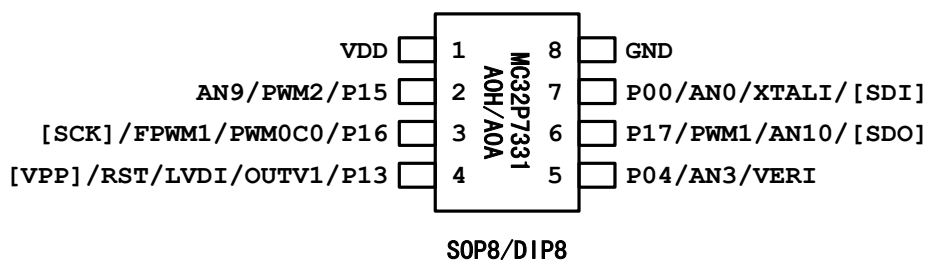
MC32P7331A1J/A1B



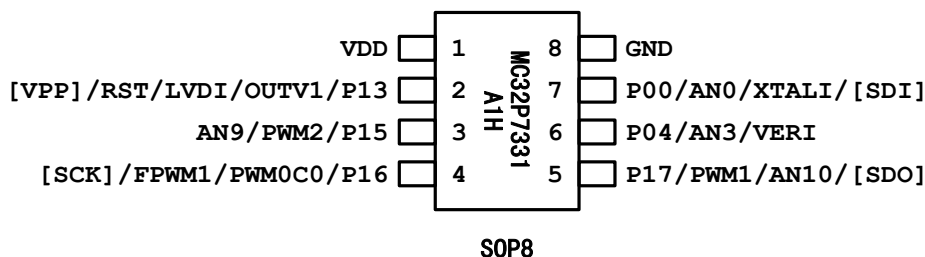
MC32P7331A0I



MC32P7331A0H/A0A



MC32P7331A1H



1.4 端口说明

端口名称	类型	功能说明
VDD	P	电源
GND	P	地
P0, P1 (除 P13)	D	GPIO, 内部上/下拉
P13	D	开漏 IO, 内部上/下拉
INT0~INT1	DI	外部中断输入
TC0~TC1	DI	定时器 T0~T1 的外部计数输入
PWM0C0~PWM0C1	DO	定时器 T0 的 PWM/BUZ 输出通道
PWM1	DO	定时器 T1 的 PWM/BUZ 输出
FPWM1	DO	定时器 T1 的 PWM 互补输出
PWM2	DO	定时器 T2 的 PWM/BUZ 输出
FPWM2	DO	定时器 T2 的 PWM 互补输出
NPWM2	DO	定时器 T2 的 PWM 反向输出
NFPWM2	DO	定时器 T2 的 PWM 互补反向输出
AN0~AN10	AI	ADC 模拟输入通道
OUTV0~OUTV1	AI	ADC 内部 1/4 分压输入通道
VERI	AI	ADC 外部参考电压输入
LVDI	AI	LVD 外部检测电压输入
XTALI, XTALO	A	外部 32768Hz 晶体振荡器输入/输出
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；D-数字输入输出，DI-数字输入，DO-数字输出；A-模拟输入输出，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
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 (除 P16/P17)	IVDDmax	30	mA
流出 GND 最大电流 (除 P00/P16/P17)	IGNDmax	30	mA
流过 P00 最大电流	IP00max	50	mA
流过 P16-P17 最大电流	IP16-P17max	120	mA

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

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/4	VLVR36		5.5	V
			Fcpu=4MHz@FHIRC/8	VLVR24		5.5	
			Fcpu=2MHz@FHIRC/16	VLVR22		5.5	
			Fcpu=125KHz@FHIRC/256	VLVR22		5.5	
			Fcpu=16KHz@FLIRC/2	VLVR22		5.5	
			Fcpu=16384Hz@XTAL 32768	VLVR22		5.5	
定时器 工作电压	VTMR	VDD	FTMR=64MHz@HIRC 32MHz	VLVR36			V
			FTMR=32MHz@HIRC 32MHz	VLVR24			
			FTMR=16MHz@HIRC 32MHz	VLVR22			
输入漏电流	Ileak	所有输入脚	VDD=5V	-1		1	uA
输入高电平	Vih	所有输入脚		0.8VDD			V
输入低电平	Vil	所有输入脚				0.2VDD	V
输出拉电流	Ioh1	推挽输出脚 (除 P16,P17)	Voh=VDD-0.6V, CDRVS=0	12.5	25	37.5	mA
			Voh=VDD-0.6V, CDRVS=1	4	8	12	mA
	Ioh2	P16,P17	Voh=VDD-0.6V, PDRVS=00		25		mA
			Voh=VDD-0.6V, PDRVS=01		50		mA
			Voh=VDD-0.6V, PDRVS=10		75		mA
			Voh=VDD-0.6V, PDRVS=11		100		mA
输出灌电流	Iol1	所有输出脚(除 P00,P16,P17)	Vol=0.6V, CDRVS=0	12.5	25	37.5	mA
			Vol=0.6V, CDRVS=1	8	16	24	mA

	Iol2	P00	Vol=0.6V, LDRVS=0		16		mA
			Vol=0.6V, LDRVS=1		40		mA
	Iol3	P16,P17	Vol=0.6V, PDRVS=00		25		mA
			Vol=0.6V, PDRVS=01		50		mA
			Vol=0.6V, PDRVS=10		75		mA
			Vol=0.6V, PDRVS=11		100		mA
上拉电阻	Rpu1	所有输入脚	VDD=5V, Vin=0		50		KΩ
			VDD=3V, Vin=0		100		KΩ
	Rpu2	VERI	VDD=2V~5V, Vin=0, RESS0=1	1.5	3	6	KΩ
下拉电阻	Rpd	所有输入脚	Vin=VDD=5V		50		KΩ
			Vin=VDD=3V		100		KΩ
分压电阻 1	Rout	OUTV0	VDD=2V~5V, Vin=0	32	48	64	KΩ
		OUTV1	OUTV0/OUTV1 电压分压比例	-1%	1/4	+1%	Vout
分压电阻 2	Rbat	VDD	VDD=2V~5V, Vin=0	32	48	64	KΩ
			VDD 电压分压比例	-1%	1/4	+1%	VDD
动态功耗	Idd	VDD	VDD=5V, Fcpu=1MHz@HIRC		1.3		mA
			VDD=3V, Fcpu=1MHz@HIRC		0.8		mA
HOLD1 功耗	Ihold1	VDD	VDD=5V, CPU 停, HIRC/LIRC 开		750		uA
HOLD2 功耗	Ihold2	VDD	VDD=5V, CPU 停, HIRC 关, LIRC 开		3	6	uA
			VDD=3V, CPU 停, HIRC 关, LIRC 开		1	3	uA
休眠模式功耗	Istop	VDD	休眠模式, WDT 开, LVR/LVD 开		4	8	uA
			休眠模式, WDT 开, LVR/LVD 关		3	6	uA
			休眠模式, WDT 关, LVR/LVD 开		1	3	uA
			休眠模式, WDT 关, LVR/LVD 关		0.5	1	uA
低压复位电压	VLVR22	VDD		-10%	2.2	+10%	V
	VLVR24			-10%	2.4	+10%	V
	VLVR36			-10%	3.6	+10%	V
低压检测电压	VLVD	VDD	VLVDS=0001	-25%	1	+25%	V
			VLVDS=其他	-5%		+5%	V

注：功耗特性参数的条件说明中，诸如 HIRC/LIRC/WDT/LVR/LVD/ADC 等未注明模块，默认其为关闭状态。

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	F _{HIRC}	VDD=2.2V~5.5V, T=25°C	-2%	32	+2%	MHz
		VDD=2.2V~5.5V, T=-40°C~85°C	-10%	32	+10%	MHz
LIRC 振荡频率	F _{LIRC}	VDD=5V, T=25°C	-50%	32	+50%	KHz
32768 晶振起振电压		T=25°C	1.8			V
32768 晶振起振时间		VDD=5V, T=25°C		1		s

2.4 ADC 特性参数

VDD=5V, T=25°C

特性	符号	条件	最小	典型	最大	单位
ADC 有效工作电压	V _{ADC}	T=25°C	2.7		5.5	V
积分线性误差	INL	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =20us			±4	LSB
微分线性误差	DNL	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =20us			±2	LSB
零点偏移误差	EZ	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =20us			±4	LSB
增益误差	ET	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =20us			±4	LSB
转换时钟	F _{ADC}	VDD=5V			1	MHz
		VDD=3V			500	KHz
转换时间	T _{con}				27	1/F _{ADC}
ADC 输入电压	V _{AIN}		GND		V _{REF}	V
ADC 输入阻抗	R _{AIN}		2			MΩ
ADC 输入电流	I _{AIN}				2	uA
ADC 动态电流	I _{ADD}	VDD=5V, ADC 转换中		1	3	mA
ADC 静态电流	I _{ADS}	VDD=5V, ADC 关闭		0.1	1	uA
模拟信号源推荐阻抗	Z _{AIN}				10	KΩ
ADC 参考电压	V _{REF}	选择 VDD		VDD		V
		选择内部参考电压 V _{IR} , T=25°C	-1.5%	2/3/4	+1.5%	
		选择内部参考电压 V _{IR} , T=-40°C~85°C	-3%	2/3/4	+3%	
		选择外部参考电压 V _{ER}	2		VDD	
V _{IR} 有效工作电压	V _{VIR}	选择内部参考电压 V _{IR}	V _{IR} +0.5		VDD	V

3 CPU 及存储器

3.1 指令集

本芯片指令集为精简指令集。除程序跳转类指令，其余指令均为单周期指令，即执行时间为 1 个指令周期；所有指令均为单字指令，即指令码只占用 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 ACC 相加，结果存入 ACC	$R + ACC \rightarrow ACC$	1	1	C,DC,Z
ADDRA R	R 和 ACC 相加，结果存入 R	$R + ACC \rightarrow R$	1	1	C,DC,Z
ADCAR R	R 和 ACC 相加（带 C 标志），结果存入 ACC	$R + ACC + C \rightarrow ACC$	1	1	C,DC,Z
ADCRA R	R 和 ACC 相加（带 C 标志），结果存入 R	$R + ACC + C \rightarrow R$	1	1	C,DC,Z
RSUBAR R	R 和 ACC 相减，结果存入 ACC	$R - ACC \rightarrow ACC$	1	1	C,DC,Z
RSUBRA R	R 和 ACC 相减，结果存入 R	$R - ACC \rightarrow R$	1	1	C,DC,Z
RSBCAR R	R 和 ACC 相减（带 C 标志），结果存入 ACC	$R - ACC - C \rightarrow ACC$	1	1	C,DC,Z
RSBCRA R	R 和 ACC 相减（带 C 标志），结果存入 R	$R - ACC - C \rightarrow R$	1	1	C,DC,Z
ASUBAR R	ACC 和 R 相减，结果存入 ACC	$ACC - R \rightarrow ACC$	1	1	C,DC,Z
ASUBRA R	ACC 和 R 相减，结果存入 R	$ACC - R \rightarrow R$	1	1	C,DC,Z
ASBCAR R	ACC 和 R 相减（带 C 标志），结果存入 ACC	$ACC - R - C \rightarrow ACC$	1	1	C,DC,Z
ASBCRA R	ACC 和 R 相减（带 C 标志），结果存入 R	$ACC - R - C \rightarrow R$	1	1	C,DC,Z
ANDAR R	R 和 ACC 与操作，结果存入 ACC	$R \text{ and } ACC \rightarrow ACC$	1	1	Z
ANDRA R	R 和 ACC 与操作，结果存入 R	$R \text{ and } ACC \rightarrow R$	1	1	Z
ORAR R	R 和 ACC 或操作，结果存入 ACC	$R \text{ or } ACC \rightarrow ACC$	1	1	Z
ORRA R	R 和 ACC 或操作，结果存入 R	$R \text{ or } ACC \rightarrow R$	1	1	Z
XORAR R	R 和 ACC 异或操作，结果存入 ACC	$R \text{ xor } ACC \rightarrow ACC$	1	1	Z
XORRA R	R 和 ACC 异或操作，结果存入 R	$R \text{ xor } ACC \rightarrow R$	1	1	Z
COMAR R	对 R 取反，结果存入 ACC	$R \text{ 取反} \rightarrow ACC$	1	1	Z
COMR R	对 R 取反，结果存入 R	$R \text{ 取反} \rightarrow R$	1	1	Z
RLA	ACC 循环左移（带 C 标志）	$ACC[7] \rightarrow C$ $ACC[6:0] \rightarrow ACC[7:1]$ $C \rightarrow ACC[0]$	1	1	C
RLAR R	R 循环左移（带 C 标志），结果存入 ACC	$R[7] \rightarrow C$ $R[6:0] \rightarrow ACC[7:1]$ $C \rightarrow ACC[0]$	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
RRA	ACC 循环右移（带 C 标志）	$C \rightarrow ACC[7]$ $ACC[7:1] \rightarrow ACC[6:0]$ $ACC[0] \rightarrow C$	1	1	C

RRAR	R	R 循环右移 (带 C 标志), 结果存入 ACC	R[0]→C R[7:1]→ACC[6:0] C→ACC[7]	1	1	C
RRR	R	R 循环右移 (带 C 标志), 结果存入 R	R[0]→C R[7:1]→R[6:0] C→R[7]	1	1	C
SWAPAR	R	交换 R 的高低半字节, 结果存入 ACC	R[7:4]→ACC[3:0] R[3:0]→ACC[7:4]	1	1	-
SWAPR	R	交换 R 的高低半字节, 结果存入 R	R[7:4]→R[3:0] R[3:0]→R[7:4]	1	1	-
MOVRA	R	将 ACC 存入 R	ACC→R	1	1	-
MOVAR	R	将 R 存入 ACC	R→ACC	1	1	Z
MOVR	R	将 R 存入 R	R→R	1	1	Z
CLRA		将 ACC 清 0	0→ACC	1	1	Z
CLRR	R	将 R 清 0	0→R	1	1	Z
INCA		ACC 自加 1	ACC+1→ACC	1	1	-
INCR	R	R 自加 1	R+1→R	1	1	Z
INCAR	R	R 加 1, 结果存入 ACC	R+1→ACC	1	1	Z
DECA		ACC 自减 1	ACC-1→ACC	1	1	-
DECR	R	R 自减 1	R-1→R	1	1	Z
DECAR	R	R 减 1, 结果存入 ACC	R-1→ACC	1	1	Z
JZA		ACC 自加 1; 结果为 0 则跳过下一条指令	ACC+1→ACC; 结果为 0 则 PC+2→PC	1/2	1	-
JZR	R	R 自加 1; 结果为 0 则跳过下一条指令	R+1→R; 结果为 0 则 PC+2→PC	1/2	1	-
JZAR	R	R 加 1, 结果存入 ACC; 结果为 0 则跳过下一条指令	R+1→ACC; 结果为 0 则 PC+2→PC	1/2	1	-
DJZA		ACC 自减 1; 结果为 0 则跳过下一条指令	ACC-1→ACC; 结果为 0 则 PC+2→PC	1/2	1	-
DJZR	R	R 自减 1; 结果为 0 则跳过下一条指令	R-1→R; 结果为 0 则 PC+2→PC	1/2	1	-
DJZAR	R	R 减 1, 结果存入 ACC; 结果为 0 则跳过下一条指令	R-1→ACC; 结果为 0 则 PC+2→PC	1/2	1	-
BCLR	R, b	将 R 的第 b 位清 0	0→R[b]	1	1	-
BSET	R, b	将 R 的第 b 位置 1	1→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→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→PC	1/2	1	-
ADDAI	K	K 和 ACC 相加, 结果存入 ACC	K+ACC→ACC	1	1	C,DC,Z
ADCAI	K	K 和 ACC 相加 (带 C 标志), 结果存入 ACC	K+ACC+C→ACC	1	1	C,DC,Z
ISUBAI	K	K 和 ACC 相减, 结果存入 ACC	K-ACC→ACC	1	1	C,DC,Z
ISBCAI	K	K 和 ACC 相减 (带 C 标志), 结果存入 ACC	K-ACC-/C→ACC	1	1	C,DC,Z
ASUBAI	K	ACC 和 K 相减, 结果存入 ACC	ACC-K→ACC	1	1	C,DC,Z
ASBCAI	K	ACC 和 K 相减 (带 C 标志), 结果存入 ACC	ACC-K-/C→ACC	1	1	C,DC,Z
ANDAI	K	K 和 ACC 与操作, 结果存入 ACC	K and ACC→ACC	1	1	Z
ORAI	K	K 和 ACC 或操作, 结果存入 ACC	K or ACC→ACC	1	1	Z
XORAI	K	K 和 ACC 异或操作, 结果存入 ACC	K xor ACC→ACC	1	1	Z
MOVAI	K	将 K 存入 ACC	K→ACC	1	1	-
CALL	K	子程序调用	PC+1→TOS K→PC[12:0]	2	1	-

GOTO K	无条件跳转	K→PC[12:0]	2	1	-
RETURN	从子程序返回	TOS→PC	2	1	-
RETAI K	从子程序返回，并将 K 存入 ACC	TOS→PC K→ACC	2	1	-
RETIE	从中断返回	TOS→PC 1→GIE	2	1	-
NOP	空操作	空操作	1	1	-
DAA	BCD 码加法后，将 ACC 的值调整为 BCD 码	ACC(HEX 码)→ACC(BCD 码)	1	1	C
DSA	BCD 码减法后，将 ACC 的值调整为 BCD 码	ACC(HEX 码)→ACC(BCD 码)	1	1	-
CLRWDT	将看门狗定时器清 0	0→WDT	1	1	TO,PD
STOP	进入低功耗模式	0→WDT；CPU 暂停	1	1	TO,PD

注：

- 1、ACC-算数逻辑单元累加器，R-数据存储器，K-立即数；
- 2、对于条件跳转类指令，若跳转条件成立，则指令需 2 个周期，否则只需 1 个周期；

3.2 程序存储器

芯片的程序存储器为 OTP 型存储器，2K×16 位的地址空间范围为 0000H~07FFH。程序存储器地址分配如下图所示：

复位向量 (0000H)
通用程序区 (0000H - 0007H)
中断向量 (0008H)
通用程序区 (0000H - 07FFH)

程序存储器支持间接寻址，可通过寄存器 INDF3 访问 FSR1×256+FSR0 指向的程序存储器地址中的内容。例：通过间接寻址读取程序存储器 0155H 地址中的内容，高 8 位存入数据存储器 11H 地址，低 8 位存入数据存储器 10H 地址

```

MOVAI    01H
MOVRA    FSR1          ; 将 01H 写入 FSR1
MOVAI    55H
MOVRA    FSR0          ; 将 55H 写入 FSR0
MOVAR    INDF3         ; 读取 FSR1×256+FSR0 指向的程序存储器地址 (0155H)
                        ; 中的内容，高 8 位存入 HIBYTE，低 8 位存入 A 寄存器
MOVRA    10H           ; 低 8 位存入数据存储器 10H 地址
MOVAR    HIBYTE        ; 从 HIBYTE 读取高 8 位
MOVRA    11H           ; 高 8 位存入数据存储器 11H 地址

```

3.3 数据存储器

数据存储器包括通用数据存储器 GPR 和特殊功能寄存器 SFR，具体地址分配参照下表。GPR 可直接寻址或通过 INDF0/INDF2 间接寻址，SFR 可直接寻址或通过 INDF1/INDF2 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
000H~0FFH	GPR	通用数据存储器区							
100H~17FH	保留	保留							
180H~187H	SFR	INDF0	INDF1	INDF2	HIBYTE	FSR0	FSR1	PCL	PFLAG
188H~18FH		MCR	INDF3	INTE	INTF	DRVCR		P1KBCR	
190H~197H		IOP0	OEP0	PUP0	PDP0	IOP1	OEP1	PUP1	PDP1
198H~19FH		T0CR	T0CNT	T0LOAD	T0DATA	T1CR	T1CNT	T1LOAD	T1DATA
1A0H~1A7H		T2CR	T2CNT	T2LOAD	T2DATA	T3CR	T3CNT	T3LOAD	
1A8H~1AFH		PWM0DR	PWM1DR	PWM2DR		PWMCR0	PWMCR1	OSCMR	LVDCR
1B0H~1B7H		ADCR0	ADCR1	ADCR2		ADRH	ADRL	ADIOS0	ADIOS1
1B8H~1BFH						PMOD	PWMCR3	PWMCR2	HIRCCR
1C0H~1F7H									
1F8H~1FFH					OSCCALH	OSCCALL			

注：上表中灰色部分地址为系统保留区，读出数据不确定，写入操作可能会影响芯片正常工作。

数据存储器寻址模式地址组成

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	寻址模式
0	0	0	0	0	0	0	来自指令的 9 位地址									直接寻址模式
0	0	0	0	0	0	0	0	FSR0								间接寻址模式 0
0	0	0	0	0	0	0	1	FSR1								间接寻址模式 1
FSR1								FSR0								间接寻址模式 2

直接寻址模式，是以指令的低 9 位为数据存储器地址，通过指令访问，寻址范围为 000H~1FFH。

例：通过直接寻址模式将数据 55H 写入数据存储器 010H 地址

MOVAI 55H

MOVRA 10H

：将数据 55H 写入数据存储器 10H 地址

间接寻址模式 0，是以 FSR0 为数据存储器地址指针，通过 INDF0 访问，寻址范围为 000H~0FFH。

例：通过间接寻址模式 0 将数据 55H 写入数据存储器 010H 地址

MOVAI 10H

MOVRA FSR0

MOVAI 55H

MOVRA INDF0

：将数据 55H 写入 FSR0 指向的数据存储器中

间接寻址模式 1，是以 FSR1 为数据存储器地址指针，通过 INDF1 访问，寻址范围为 100H~1FFH。

例：通过间接寻址模式 1 将数据 55H 写入数据存储器 110H 地址

MOVAI	10H	
MOVRA	FSR1	
MOVAI	55H	
MOVRA	INDF1	: 将数据 55H 写入 FSR1 指向的数据存储器中

间接寻址模式 2, 是以 [FSR1:FSR0] 为数据存储器地址指针, 通过 INDF2 访问, 寻址范围为 0000H ~ FFFFH。例: 通过间接寻址模式 2 将数据 55H 写入数据存储器 0110H 地址

MOVAI	01H	
MOVRA	FSR1	
MOVAI	10H	
MOVRA	FSR0	
MOVAI	55H	
MOVRA	INDF2	: 将数据 55H 写入 FSR1×256+FSR0 指向的数据存储器中

3.4 堆栈

8 级堆栈深度, 当程序响应中断或执行子程序调用指令时 CPU 会将 PC 自动压栈保存; 当执行中断返回指令或子程序返回指令时, 栈顶数据自动出栈载入 PC。

3.5 控制寄存器

间接寻址寄存器 0

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF0	INDF07	INDF06	INDF05	INDF04	INDF03	INDF02	INDF01	INDF00
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] INDF0[7:0] – 间接寻址寄存器 0

INDF0: INDF0 不是物理寄存器, 对 INDF0 寻址实际是对 FSR0 指向的数据存储器地址进行访问, 从而实现间接寻址功能。

间接寻址寄存器 1

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF1	INDF17	INDF16	INDF15	INDF14	INDF13	INDF12	INDF11	INDF10
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] INDF1[7:0] – 间接寻址寄存器 1

INDF1: INDF1 不是物理寄存器, 对 INDF1 寻址实际是对 FSR1+256 指向的数据存储器地址进行访问, 从而实现间接寻址功能。

间接寻址寄存器 2

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF2	INDF27	INDF26	INDF25	INDF24	INDF23	INDF22	INDF21	INDF20
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] INDF2[7:0] – 间接寻址寄存器 2

INDF2: INDF2 不是物理寄存器，对 INDF2 寻址实际是对 FSR1×256+FSR0 指向的数据存储器地址进行访问，从而实现间接寻址功能。

间接寻址寄存器 3

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF3	INDF37	INDF36	INDF35	INDF34	INDF33	INDF32	INDF31	INDF30
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] INDF3[7:0] – 间接寻址寄存器 3

INDF3: INDF3 不是物理寄存器，对 INDF3 寻址实际是对 FSR1×256+FSR0 指向的程序存储器地址进行访问，从而实现间接寻址功能。

注：寄存器 INDF3 仅可使用读取指令 (MOVAR INDF3) 进行读取操作，读取内容高 8 位存入 HIBYTE 寄存器，低 8 位存入 A 寄存器。

字操作高字节缓存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
HIBYTE	HIBYTE7	HIBYTE6	HIBYTE5	HIBYTE4	HIBYTE3	HIBYTE2	HIBYTE1	HIBYTE0
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] HIBYTE[7:0] – 字操作高字节缓存器

HIBYTE: 用于存放通过 INDF3 访问程序存储器时所读取内容的高 8 位数据。

数据指针寄存器 0

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR0	FSR07	FSR06	FSR05	FSR04	FSR03	FSR02	FSR01	FSR00
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] FSR0[7:0] – 数据指针寄存器 0

FSR0: 间接寻址模式 0 的指针，或间接寻址模式 2、3 的指针低 8 位。

数据指针寄存器 1

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR1	FSR17	FSR16	FSR15	FSR14	FSR13	FSR12	FSR11	FSR10
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] FSR1[7:0] – 数据指针寄存器 1

FSR1: 间接寻址模式 1 的指针, 或间接寻址模式 2、3 的指针高 8 位。

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 (PC) 有以下几种操作模式:

- ✧ 顺序运行指令: $PC = PC + 1$;
- ✧ 分支指令 GOTO/CALL: $PC =$ 指令码低 13 位;
- ✧ 返回指令 RETIE/RETURN/RETAI: $PC =$ 堆栈栈顶 (TOS);

对 PCL 操作指令:

- ✧ 对 PCL 操作的加法指令: $PC = (PC[12:0] + ALU[7:0])$;
- ✧ 对 PCL 操作的其它指令: $PC = \{PC[12:8]: ALU[7:0](ALU \text{ 运算结果})\}$;

CPU 状态寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PFLAG	-	-	-	-	-	Z	DC	C
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	X	X	X

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/借位标志位

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

BIT[0] C – 进/借位标志位

- 0: 加法运算时无进位; 减法运算时有借位; 移位后移出逻辑 0;
- 1: 加法运算时有进位; 减法运算时无借位; 移位后移出逻辑 1;

杂项控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
MCR	GIE	-	TO	PD	INT1M1	INT1M0	INT0M1	INT0M0
R/W	R/W	-	R	R	R/W	R/W	R/W	R/W
初始值	0	-	0	0	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[5] **TO** – 看门狗溢出标志位
0: 上电复位, 或执行 CLRWDWT/STOP 指令;
1: 发生 WDT 溢出;

BIT[4] **PD** – 进入低功耗模式标志位
0: 上电复位, 或执行 CLRWDWT 指令;
1: 执行 STOP 指令;

BIT[3:2] **INT1M[1:0]** – 外部中断 INT1 触发方式选择位

INT1M[1:0]	INT1 触发方式
00	上升沿触发
01	下降沿触发
10	电平变化触发
11	

BIT[1:0] **INT0M[1:0]** – 外部中断 INT0 触发方式选择位

INT0M[1:0]	INT0 触发方式
00	上升沿触发
01	下降沿触发
10	电平变化触发
11	

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中, 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。

本芯片的用户配置字, 定义如下:

符号	功能说明
FCPUS	高频时钟下 FCPU 分频选择: FCPU = FHOSC/4 ; FCPU = FHOSC/8 ; FCPU = FHOSC/16 ; FCPU = FHOSC/32 ; FCPU = FHOSC/64 ; FCPU = FHOSC/128 ; FCPU = FHOSC/256 ;
LOSCS	系统低频时钟源选择: 内部 32KHz 低频 RC 振荡器 ; 外部 32768Hz 晶振 ;
VLVRS	LVR 复位电压选择: 2.2V ; 2.4V ; 3.6V ;

WDTM	WDT 模式设置： WDT 始终关闭； WDT 低功耗模式下关闭； WDT 始终开启；
WDTT	WDT 溢出时间选择： PWRT=TWDT=16ms； PWRT=TWDT=64ms； PWRT=TWDT=256ms； PWRT=16ms，TWDT=1024ms； PWRT=64ms，TWDT=2048ms； PWRT=256ms，TWDT=4096ms；
RSTEN	RST 外部复位设置： P13 为外部复位脚； P13 为输入/输出脚；
ENCR	代码加密设置： 程序代码加密； 程序代码不加密；

HIRC 微调校准寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSCCALH	-	-	-	-	-	OSCCAL10	OSCCAL9	OSCCAL8
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	U	U	U

BIT[2:0] OSCCAL[10:8] – HIRC 频率微调值高 3 位（复位初始值为出厂设定值）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSCCALL	OSCCAL7	OSCCAL6	OSCCAL5	OSCCAL4	OSCCAL3	OSCCAL2	OSCCAL1	OSCCAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	U	U	U	U	U	U	U	U

BIT[7:0] OSCCAL[7:0] – HIRC 频率微调值低 8 位（复位初始值为出厂设定值）

注：

- 1、HIRC 最大调节范围为 28.8MHz(-10%)~33.28MHz(+4%) (以实际芯片为准)，因芯片及模块的工作电压受其时钟频率限制，所以微调 HIRC 时建议从初值（32MHz）往低频调节，以免影响正常的工作电压范围；
- 2、步长增减的频率是非线性的，最大约为 0.5%；
- 3、OSCCAL 中已存有 HIRC 出厂校准值，在修改 OSCCAL 前最好备份原值以便恢复；

HIRC 的微调可选择非同步模式和同步模式。非同步模式下，对 OSCCALH 的写操作将直接写入寄存器中；而同步模式下，对 OSCCALH 的写操作仅将数据写入 OSCCALH 的 3 位缓存器中，只有在对 OSCCALL 进行写操作的同时才将缓存器中的值加载至 OSCCALH 中。

HIRC 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
HIRCCR	-	-	-	-	-	-	-	HIRCAM
R/W	-	-	-	-	-	-	-	R/W
初始值	-	-	-	-	-	-	-	0

BIT[0] HIRCAM – HIRC 微调模式选择位

- 0：非同步模式；
1：同步模式；

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，可用作系统低频时钟源，也用于系统上电延时控制、WDT 定时器等电路。

4.3 外部晶体振荡器

芯片支持外接低频 32768Hz 的晶体振荡器作为系统低频时钟源。

外接晶振的实际使用中，对地的负载电容 C_x 是必须的。用户应使晶体离 XTALI/XTALO 引脚的距离尽可能短，这样有助于振荡器的起振和振荡的稳定性。

下表是典型频率晶振选用电容 C_x 的推荐值和相应最低起振电压参考值。

晶振频率 (Hz)	电容 C_x (pF)	最低起振电压 (V)
32768	10~30	1.8

注：因晶体品牌繁多且工艺差异较大，故上表中的参数仅供参考，具体应用请以晶体的实测结果为准。

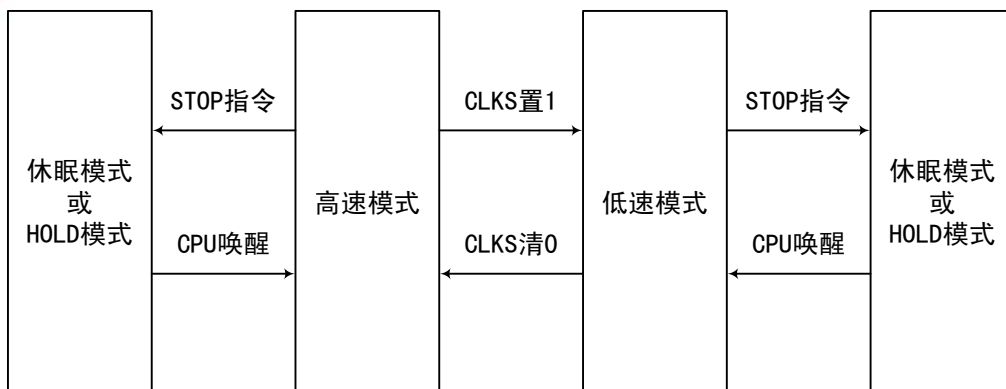
4.4 系统工作模式

芯片支持高速模式、低速模式、HOLD1 模式、HOLD2 模式和休眠模式等多种系统工作模式。

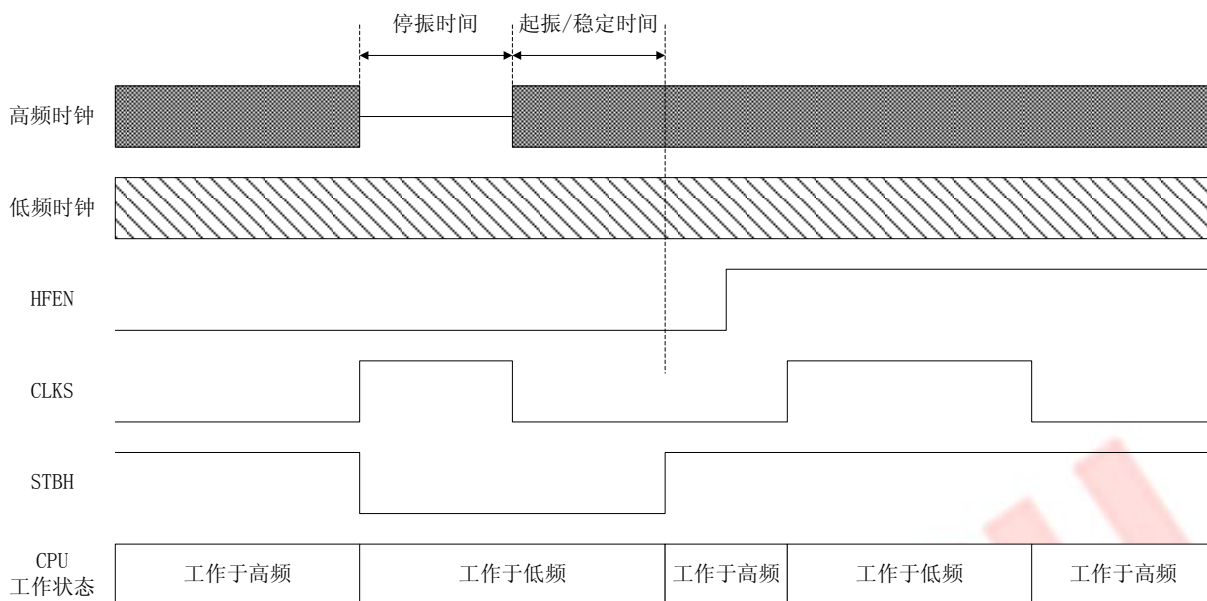
工作模式	切入条件	系统状态
高速	任意模式下，系统复位	CPU 高速运行，高/低频时钟源均工作
	低速模式下，CLKS 清 0	
	HOLD1/HOLD2/休眠模式下，CPU 唤醒	
低速	高速模式下，CLKS 置 1	CPU 低速运行，高频时钟源由 HFEN 决定
	HOLD1/HOLD2/休眠模式下，CPU 唤醒	
HOLD1	高/低速模式下，HFEN 置 1，执行 STOP 指令	CPU 暂停，高频时钟源工作，低频时钟源由 LFEN 决定
HOLD2	高/低速模式下，HFEN 清 0，LFEN 置 1，执行 STOP 指令	CPU 暂停，高频时钟源停止，低频时钟源工作
休眠	高/低速模式下，HFEN 清 0，LFEN 清 0，执行 STOP 指令	CPU 暂停，高/低频时钟源均停止

注：WDT 时钟源为 LIRC，WDT 开启时 LIRC 将一直工作而不受系统工作模式影响。

工作模式切换示意图



高低频时钟切换时序图



工作模式寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSCMR	-	-	STBL	STBH	-	CLKS	LFEN	HFEN
R/W	-	-	R	R	-	R/W	R/W	R/W
初始值	-	-	X	1	-	0	0	0

BIT[5] **STBL** – 低频时钟源振荡状态标志位

- 0: 低频时钟源停振或未稳定;
- 1: 低频时钟源已稳定振荡;

BIT[4] **STBH** – 高频时钟源振荡状态标志位

- 0: 高频时钟源停振或未稳定;
- 1: 高频时钟源已稳定振荡;

BIT[2] **CLKS** – CPU 时钟源选择位

- 0: 系统高频时钟作为 CPU 时钟源;
- 1: 系统低频时钟作为 CPU 时钟源;

BIT[1] **LFEN** – 低频时钟源使能位

- 0: 在休眠/HOLD 模式下, 低频时钟源停止工作;
- 1: 低频时钟源始终工作;

BIT[0] **HFEN** – 高频时钟源使能位

- 0: 在低速/休眠/HOLD 模式下, 高频时钟源停止工作;
- 1: 高频时钟源始终工作;

4.5 低功耗模式

芯片的低功耗模式包括休眠模式、HOLD1 模式、HOLD2 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态不变；
- ✧ 定时器若其时钟源未停止，则可以保持继续工作；

以下情况可使系统退出低功耗模式：

- ✧ 上电复位；
- ✧ 外部复位（若有外部复位功能）；
- ✧ 有 WDT 溢出（若低功耗模式下 WDT 保持继续工作）；
- ✧ 有外部中断请求发生（若有外部中断功能）；
- ✧ 定时器溢出中断发生（若低功耗模式下定时器保持继续工作）；
- ✧ 有键盘中断请求发生（若有键盘中断功能）；

注：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 后执行中断服务程序。

5 复位

5.1 复位条件

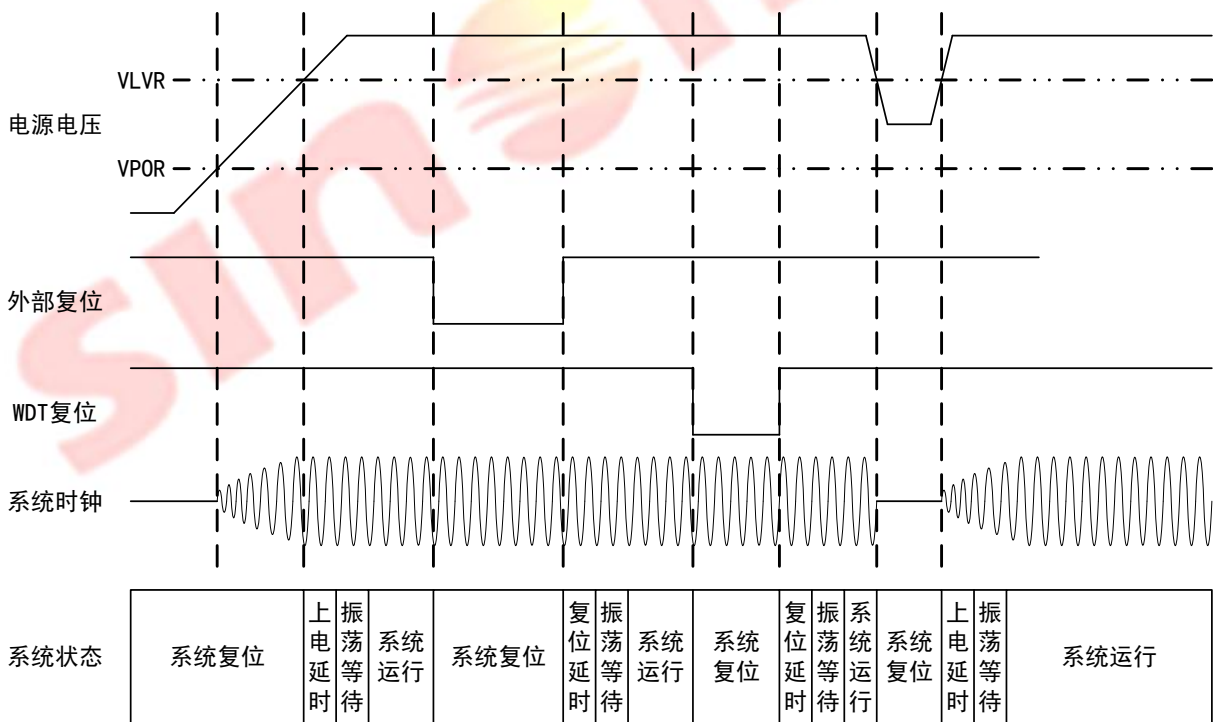
芯片共有如下几种复位方式：

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

任何一种复位发生时，系统均会重新从 0000H 地址处开始执行指令；并且系统还会将特殊功能寄存器 SFR 重置为复位初始值。

上电复位 POR 和低电压复位 LVR 会关闭系统主时钟振荡器，复位解除后才重新打开振荡器，因为振荡器起振和稳定需要一定的时间，所以系统会保持一定时间的上电延时和振荡等待后才开始工作；而外部复位和 WDT 复位不会关闭主时钟振荡器，复位解除时系统会在较短的复位延时和振荡等待后即开始工作。

下图是复位产生和系统工作状态之间的关系示意图：



5.2 上电复位

芯片的上电复位电路可以适应快速、慢速上电的情况，并且当芯片上电过程中出现电源电压抖动时均能保证系统可靠的复位。

上电复位过程可以概括为以下几个步骤：

- (1) 检测系统工作电压，等待电压高于上电复位电压 V_{POR} 并保持稳定；
- (2) 若有 LVR 功能，则需等待电压高于 V_{LVR} 并保持稳定；
- (3) 若有外部复位功能，则需等待复位引脚电压高于 V_{ih} ；
- (4) 初始化所有寄存器；
- (5) 开启主时钟振荡器，并等待一段时间以待振荡器稳定；
- (6) 上电结束，系统开始执行指令。

5.3 外部复位

本芯片的外部复位功能可通过用户配置字 **RSTEN** 开启，引脚配置成复位脚即开启外部复位功能，端口的内部上拉电阻自动有效。外部复位输入端口 **RST** 为施密特结构，低电平有效。当外部复位输入为高电平时，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点可通过用户配置字 **VLVRS** 选择。电压检测电路有一定的回滞特性，通常回滞电压为 0.1V 左右，当电源电压下降至 LVR 电压时 LVR 复位有效，反之则电源电压需上升至 LVR 电压+0.1V 后 LVR 复位才会解除。

5.5 看门狗复位

看门狗（WDT）复位是一种对程序正常运行的保护机制。正常情况下，用户程序需定时对 WDT 定时器进行清零操作，以保证 WDT 不溢出。若出现异常状况，程序未按时对 WDT 定时器清零，则芯片会因 WDT 溢出而产生看门狗复位，系统重新初始化，返回受控状态。

注：低功耗模式下 CPU 暂停工作，若此时有 WDT 溢出，则仅唤醒 CPU 而不产生复位。

6 I/O 端口

6.1 通用 I/O 功能

芯片的输入/输出端口包括一组 6 位端口 P0 和一组 8 位端口 P1。所有 I/O 端口均支持施密特输入；除 P13 外的 I/O 端口均支持推挽输出。

除用作通用数字 I/O 端口外，部分端口还具有外部中断输入、PWM 输出、或 ADC 模拟输入等复用功能。

端口数据寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOP0	-	-	P05D	P04D	P03D	P02D	P01D	P00D
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] P0nD – P0n 端口数据位 (n=5-0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOP1	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D
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] P1nD – P1n 端口数据位 (n=7-0)

端口方向寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OEP0	-	-	P05OE	P04OE	P03OE	P02OE	P01OE	P00OE
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] P0nOE – P0n 端口输出使能位 (n=5-0)

0: 端口作为输入口，读端口操作将读取端口的输入电平状态；

1: 端口作为输出口，读端口操作将读取端口的数据寄存器值；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OEP1	P17OE	P16OE	P15OE	P14OE	P13OE	P12OE	P11OE	P10OE
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] P1nOE – P1n 端口输出使能位 (n=7-0)

0: 端口作为输入口，读端口操作将读取端口的输入电平状态；

1: 端口作为输出口，读端口操作将读取端口的数据寄存器值；

6.2 内部上/下拉电阻

所有端口都有内部上拉和下拉电阻，均有独立的上/下拉电阻寄存器控制位，控制其上/下拉电阻在端口处于输入状态时是否有效。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
PUP0	-	-	P05PU	P04PU	P03PU	P02PU	P01PU	P00PU
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] P0nPU – P0n 端口上拉电阻控制位 (n=5-0)

0: 端口内部上拉电阻无效;

1: 端口内部上拉电阻有效;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PUP1	P17PU	P16PU	P15PU	P14PU	P13PU	P12PU	P11PU	P10PU
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] P1nPU – P1n 端口上拉电阻控制位 (n=7-0)

0: 端口内部上拉电阻无效;

1: 端口内部上拉电阻有效;

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
PDP0	-	-	P05PD	P04PD	P03PD	P02PD	P01PD	P00PD
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] P0nPD – P0n 端口下拉电阻控制位 (n=5-0)

0: 端口内部下拉电阻无效;

1: 端口内部下拉电阻有效;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDP1	P17PD	P16PD	P15PD	P14PD	P13PD	P12PD	P11PD	P10PD
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] P1nPD – P1n 端口下拉电阻控制位 (n=7-0)

0: 端口内部下拉电阻无效;

1: 端口内部下拉电阻有效;

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依然有效）。

模拟端口控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADIOS0	P12ADS	P11ADS	P10ADS	P05ADS	P04ADS	P03ADS	P02ADS	P00ADS
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[7:5] **P1nADS** – P1 口工作模式选择位（n=2-0）

- 0: 端口用作数字 I/O;
- 1: 端口用作模拟 I/O;

BIT[4:0] **P0nADS** – P0 口工作模式选择位（n=5-2,0）

- 0: 端口用作数字 I/O;
- 1: 端口用作模拟 I/O;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADIOS1	-	-	-	-	-	-	P15ADS	P14ADS
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

BIT[1:0] **P1nADS** – P1 口工作模式选择位（n=5-4）

- 0: 端口用作数字 I/O;
- 1: 端口用作模拟 I/O;

所有端口均可通过寄存器位选择输出驱动电流。P00/P01 用作输出口时，可通过寄存器位选择推挽输出或开漏输出。

端口模式寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMOD	P17ADS	P13ADS	P01ADS	保留	-	-	P01OM	P00OM
R/W	R/W	R/W	R/W	R/W	-	-	R/W	R/W
初始值	0	0	0	0	-	-	0	0

BIT[7] **P17ADS** – P17 口工作模式选择位

- 0: 端口用作数字 I/O;
- 1: 端口用作模拟 I/O;

BIT[6] **P13ADS** – P13 口工作模式选择位

- 0: 端口用作数字 I/O;
- 1: 端口用作模拟 I/O;

BIT[5] **P01ADS** – P01 口工作模式选择位

0: 端口用作数字 I/O;

1: 端口用作模拟 I/O;

BIT[4] 保留位, 需固定写 “0”

BIT[1] **P01OM** – P01 口输出模式选择位

0: 端口输出为开漏输出;

1: 端口输出为推挽输出;

BIT[0] **P00OM** – P00 口输出模式选择位

0: 端口输出为开漏输出;

1: 端口输出为推挽输出;

端口驱动控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DRVCR	CDRVS	LDRVS	PDRVS1	PDRVS0	-	-	-	-
R/W	R/W	R/W	R/W	R/W	-	-	-	-
初始值	0	0	0	0	-	-	-	-

BIT[7] **CDRVS** – 除 P00/P16/P17 外其它普通端口驱动能力选择位

0: 增强驱动 (Ioh=25mA, Iol=25mA);

1: 普通驱动 (Ioh=8mA, Iol=16mA);

BIT[6] **LDRVS** – P00 端口驱动能力选择位

0: 16mA;

1: 40mA;

BIT[5:4] **PDRVS[1:0]** – P16/P17 端口驱动控制位

PDRVS[1:0]	P16/P17 驱动能力
00	25mA
01	50mA
10	75mA
11	100mA

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，WDT 计数器溢出将复位芯片或唤醒 CPU。

可通过用户配置字 WDTM 设置 WDT 工作模式。若选择始终开启，则 WDT 一直工作，高速/低速模式下 WDT 溢出将复位芯片，休眠/HOLD 模式下 WDT 溢出将唤醒 CPU；若选择低功耗模式下关闭，则 WDT 在休眠/HOLD 模式下自动停止、在其他方式唤醒 CPU 后恢复工作。

执行 CLRWDT 指令或 STOP 指令将清零 WDT 计数器。

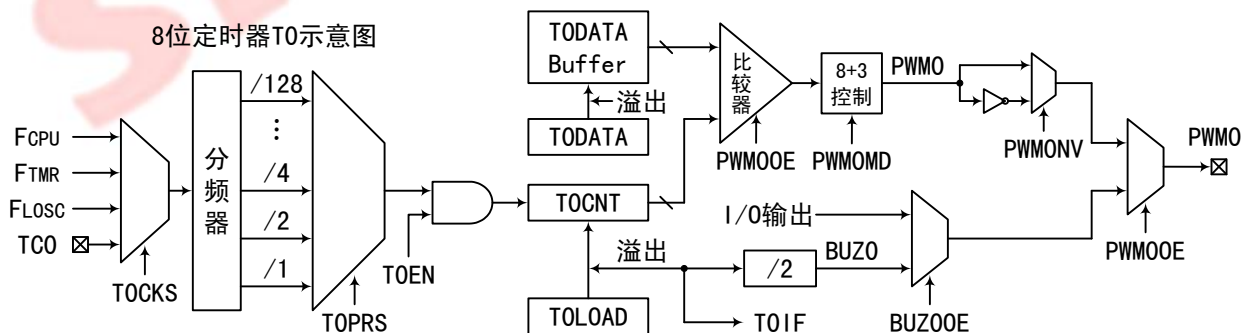
WDT 溢出时间可配置为 16ms/64ms/256ms/1024ms/2048ms/4096ms。

注：WDT 溢出时间为典型值，实际值偏差大，必须保证清 WDT 时间小于典型值的 1/4。

7.2 定时器 T0

定时器 T0 为 8 位定时/计数器，包含 1 个 8 位递减计数器、可编程预分频器、控制寄存器、8 位重载寄存器及比较寄存器。

- ◇ 可通过预分频比设置计数频率，可通过重载寄存器控制计数周期；
- ◇ 支持 8+3 模式 PWM 输出，可通过比较寄存器设置 PWM 占空比；
- ◇ PWM 有 2 路输出通道，可选择其中 1 路输出；
- ◇ 支持 BUZ 输出；
- ◇ 支持溢出中断和溢出唤醒功能；



注：

- 1、FHOSC 指系统高频时钟，FLOSC 指系统低频时钟，FTMR 为系统高频时钟经 TMRCKS[1:0]选频后的高频时钟；
- 2、使能或关闭 PWM、BUZ 输出时，应注意当前的端口状态及操作时机，以免输出尖峰脉冲；
- 3、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；
- 4、定时器时钟选择 FTMR 时，需注意 FTMR 的工作电压范围，详见直流电气特性；

定时器 T0 控制寄存器 T0CR 中，T0CKS 位可选择 T0 的时钟源，T0PRS 可选择 T0 的预分频比，所选中的时钟源通过预分频器后产生 T0 计数器 T0CNT 的计数时钟。时钟分频比可选择 1~128 分频，对 T0CNT 的写操作将使预分频器清零，分频比保持不变。

当 T0EN=0 时，T0CNT 保持不变，写重载寄存器 T0LOAD 将自动加载到 T0CNT 中；当 T0EN=1 时，T0CNT 递减计数，此时写 T0LOAD 不会立即加载，T0CNT 计数到 0 时产生 T0 溢出中断，标志位 T0IF 置 1，下一个时钟 T0LOAD 值自动载入 T0CNT 重新开始计数。

定时器 T0 可实现 BUZ 功能，当 BUZ0OE=1 且 PWM0OE=0 时，BUZ0 输出蜂鸣器驱动信号，频率为 T0 溢出频率的 2 分频。

定时器 T0 可实现 PWM 功能，PWM0OE 置 1 将使能 PWM 功能且端口 PWM0 将输出 PWM 波形。每个 PWM 周期内，计数器 T0CNT 从重载值开始递减计数：当计数到与比较寄存器 T0DATA 相等时，PWM0 信号变为高电平；当计数溢出时，PWM0 信号变为低电平。T0DATA 配有 1 个 8 位缓冲器用于与 T0CNT 比较，PWM0OE=0 时写 T0DATA 将立即加载到缓冲器中，而 PWM0OE=1 时写 T0DATA 则将在 T0 溢出时才载入缓冲器中。若要首个 PWM 周期准确，需先写重载寄存器和比较寄存器（设置周期和占空比），再开启定时器和 PWM 功能。

PWM0 信号的占空比计算如下：

- ✧ PWM0 高电平时间 = (T0DATA) × T0CNT 计数时钟周期
- ✧ PWM0 周期 (T0 的溢出周期) = (T0LOAD+1) × T0CNT 计数时钟周期
- ✧ PWM0 占空比 = (T0DATA) / (T0LOAD+1)

PWM0 支持 8+3 模式，可在普通模式 PWM 的时钟下增加占空比的调节精度，以每 8 个普通 PWM 周期作为一个大周期，通过 PWM0 占空比延展寄存器的占空比延展设置位 PWM0D[2:0] 决定在这 8 个 PWM 周期中的部分周期进行延展，这些周期内的高电平将延展一个计数时钟周期（即高电平时间为 T0DATA+1）。PWM0D[2:0] 仅用于延展控制，普通模式下无效。

8+3 模式相当于在整体上提高了 PWM 占空比的调节精度，而不是对单个 PWM 周期提高占空比精度。PWM 单个周期中正常的高电平时间由 T0DATA 决定，而 PWM0D[2:0] 决定每 8 个周期中哪几个周期的高电平需要延展一个时钟的宽度。

PWM0 的 8+3 模式一个大周期的占空比计算如下：

- ✧ PWM0 高电平时间 = [(T0DATA) × 8 + (PWM0D[2:0])] × T0CNT 计数时钟周期
- ✧ PWM0 周期 = (T0LOAD + 1) × 8 × T0CNT 的计数周期
- ✧ PWM0 占空比 = PWM0 高电平时间 / PWM0 周期

举例说明：

若 PWM0D[2:0]=000，则对 8 个普通的 PWM 周期都不延展，占空比与普通的 PWM 一样，等于 [T0DATA / (T0LOAD+1)]。

若 PWM0D[2:0]=001，则对 8 个普通的 PWM 周期中的第 1 个周期延展，在大周期中的占空比为 [(T0DATA + 1/8) / (T0LOAD+1)]。

若 PWM0D[2:0]=101，则对 8 个普通的 PWM 周期的第 (1、2、3、5、7) 个周期延展，在大周期中的占空比为 [(T0DATA + 5/8) / (T0LOAD+1)]。

PWM0 信号可通过输出电平控制位 PWM0NV 选择输出正向波形或负向波形，可通过输出通道选择位 PWM0OS 选择从端口 PWM0C0 (P16) 或端口 PWM0C1 (P02) 输出。

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	T0EN	PWM0OE	BUZ0OE	T0CKS1	T0CKS0	T0PRS2	T0PRS1	T0PRS0
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] **T0EN** – T0 使能控制位

0: 关闭 T0;

1: 开启 T0;

BIT[6] **PWM0OE** – PWM0 功能控制位

0: 关闭 PWM0 功能, 并禁止端口输出 PWM 波形;

1: 使能 PWM0 功能, 并允许端口输出 PWM 波形;

BIT[5] **BUZ0OE** – BUZ0 输出使能位

0: 关闭 BUZ0 输出;

1: 使能 BUZ0 输出 (仅 PWM0OE=0 时有效);

BIT[4:3] **T0CKS[1:0]** – T0 时钟源选择位

T0CKS[1:0]	T0 时钟源
00	FCPU
01	FTMR
10	FLOSC
11	TC0 上升沿

BIT[2:0] **T0PRS[2:0]** – T0 预分频比选择位

T0PRS[2:0]	T0 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 T0 计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T0CNT	T0CNT7	T0CNT6	T0CNT5	T0CNT4	T0CNT3	T0CNT2	T0CNT1	T0CNT0
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] **T0CNT[7:0]** – T0 计数器, 为可读写的递减计数器

定时器 T0 重载寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T0LOAD	T0LOAD7	T0LOAD6	T0LOAD5	T0LOAD4	T0LOAD3	T0LOAD2	T0LOAD1	T0LOAD0
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] **T0LOAD[7:0]** – T0 重载寄存器，用于设置 T0 的计数周期

定时器 T0 比较寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T0DATA	T0DATA7	T0DATA6	T0DATA5	T0DATA4	T0DATA3	T0DATA2	T0DATA1	T0DATA0
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] **T0DATA[7:0]** – T0 比较寄存器，用于设置 PWM0 的占空比

PWM0 占空比延展寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM0DR	-	-	-	-	-	PWM0D2	PWM0D1	PWM0D0
R/W	R	R	R	R	R	R/W	R/W	R/W
初始值	1	1	1	1	0	0	0	0

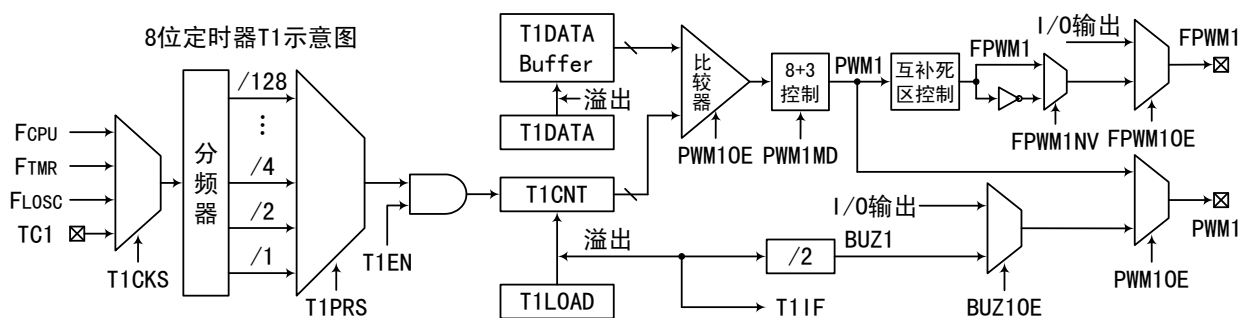
BIT[2:0] **PWM0D[2:0]** – PWM0 占空比延展设置位（仅 PWM0MD=1 时有效）

PWM0D[2:0]	PWM0 每 8 个周期的占空比延展周期
000	PWM 每 8 个周期内有 0 个周期的占空比延展
001	PWM 每 8 个周期内有 1 个周期的占空比延展（第 1 个周期）
010	PWM 每 8 个周期内有 2 个周期的占空比延展（第 1,5 个周期）
011	PWM 每 8 个周期内有 3 个周期的占空比延展（第 1,3,5 个周期）
100	PWM 每 8 个周期内有 4 个周期的占空比延展（第 1,3,5,7 个周期）
101	PWM 每 8 个周期内有 5 个周期的占空比延展（第 1,2,3,5,7 个周期）
110	PWM 每 8 个周期内有 6 个周期的占空比延展（第 1,2,3,5,6,7 个周期）
111	PWM 每 8 个周期内有 7 个周期的占空比延展（第 1,2,3,4,5,6,7 个周期）

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 位重载寄存器及比较寄存器。

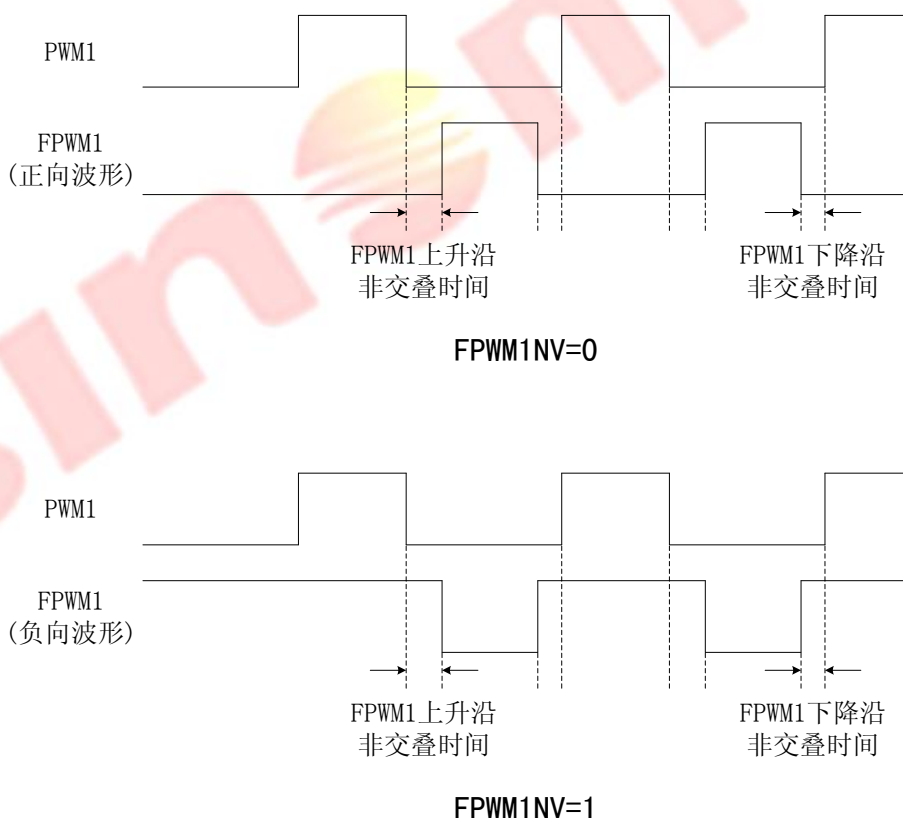
- ✧ 可通过预分频比设置计数频率，可通过重载寄存器控制计数周期；
- ✧ 支持 8+3 模式 PWM 输出，可通过比较寄存器设置 PWM 占空比；
- ✧ 可输出 2 路带死区的互补 PWM 波形；
- ✧ 支持 BUZ 输出；
- ✧ 支持溢出中断和溢出唤醒功能；



定时器 T1 的定时/外部计数功能、BUZ 功能与定时器 T0 完全相同。

定时器 T1 可实现 PWM 功能，并支持 8+3 模式，其功能完全与定时器 T0 相同。并且定时器 T1 除了可生成正常 PWM1 信号外，还可生成带死区控制的反向互补信号 FPWM1。端口 FPWM1 可通过输出使能位 FPWM1OE 选择是否输出 PWM 波形，还可通过输出电平控制位 FPWM1NV 选择输出 FPWM1 信号的正向波形或负向波形。

PWM1 输出波形示意



注：

- 1、首个 PWM 周期内，互补信号的 PWM 功能及互补死区功能无效，输出为固定电平；
- 2、应用互补 PWM 时，非交叠时间的设置应匹配 PWM 周期及占空比的设置，否则 PWM 无法正常工作；

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	PWM1OE	BUZ1OE	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** – T1 使能控制位

0: 关闭 T1;

1: 开启 T1;

BIT[6] **PWM1OE** – PWM1 功能控制位

0: 关闭 PWM1 功能, 并禁止端口输出 PWM 波形;

1: 使能 PWM1 功能, 并允许端口输出 PWM 波形;

BIT[5] **BUZ1OE** – BUZ1 输出使能位

0: 关闭 BUZ1 输出;

1: 使能 BUZ1 输出 (仅 PWM1OE=0 时有效);

BIT[4:3] **T1CKS[1:0]** – T1 时钟源选择位

T1CKS[1:0]	T1 时钟源
00	FCPU
01	FTMR
10	FLOSC
11	TC1 上升沿

BIT[2:0] **T1PRS[2:0]** – T1 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

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，用于设置 PWM1 的占空比

PWM1 占空比延展寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1DR	-	-	-	-	-	PWM1D2	PWM1D1	PWM1D0
R/W	R	R	R	R	R	R/W	R/W	R/W
初始值	1	1	1	1	0	0	0	0

BIT[2:0] **PWM1D[2:0]** – PWM1 占空比延展设置位（仅 PWM1MD=1 时有效）

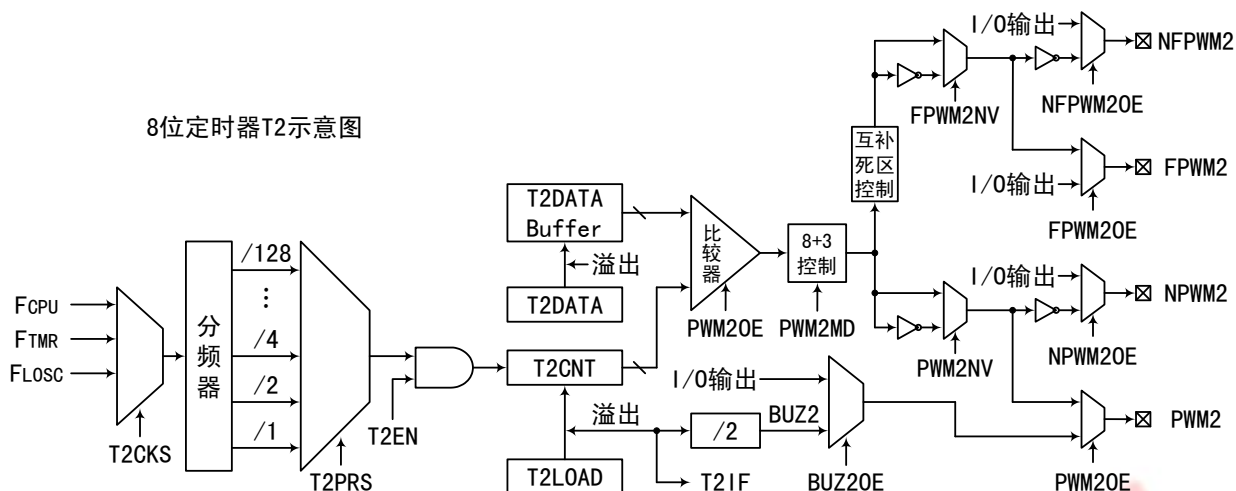
PWM1D[2:0]	PWM1 每 8 个周期的占空比延展周期
000	PWM 每 8 个周期内有 0 个周期的占空比延展
001	PWM 每 8 个周期内有 1 个周期的占空比延展（第 1 个周期）
010	PWM 每 8 个周期内有 2 个周期的占空比延展（第 1,5 个周期）
011	PWM 每 8 个周期内有 3 个周期的占空比延展（第 1,3,5 个周期）
100	PWM 每 8 个周期内有 4 个周期的占空比延展（第 1,3,5,7 个周期）
101	PWM 每 8 个周期内有 5 个周期的占空比延展（第 1,2,3,5,7 个周期）
110	PWM 每 8 个周期内有 6 个周期的占空比延展（第 1,2,3,5,6,7 个周期）
111	PWM 每 8 个周期内有 7 个周期的占空比延展（第 1,2,3,4,5,6,7 个周期）

7.4 定时器 T2

定时器 T2 为 8 位定时器，包含 1 个 8 位递减计数器、可编程预分频器、控制寄存器、8 位重载寄存器及比较寄存器。

- ◇ 可通过预分频比设置计数频率，可通过重载寄存器控制计数周期；
- ◇ 支持 8+3 模式 PWM 输出，可通过比较寄存器设置 PWM 占空比；
- ◇ 可输出 2 路带死区的互补 PWM 波形，以及该互补波形的 2 路反向波形；
- ◇ 支持 BUZ 输出；
- ◇ 支持溢出中断和溢出唤醒功能；

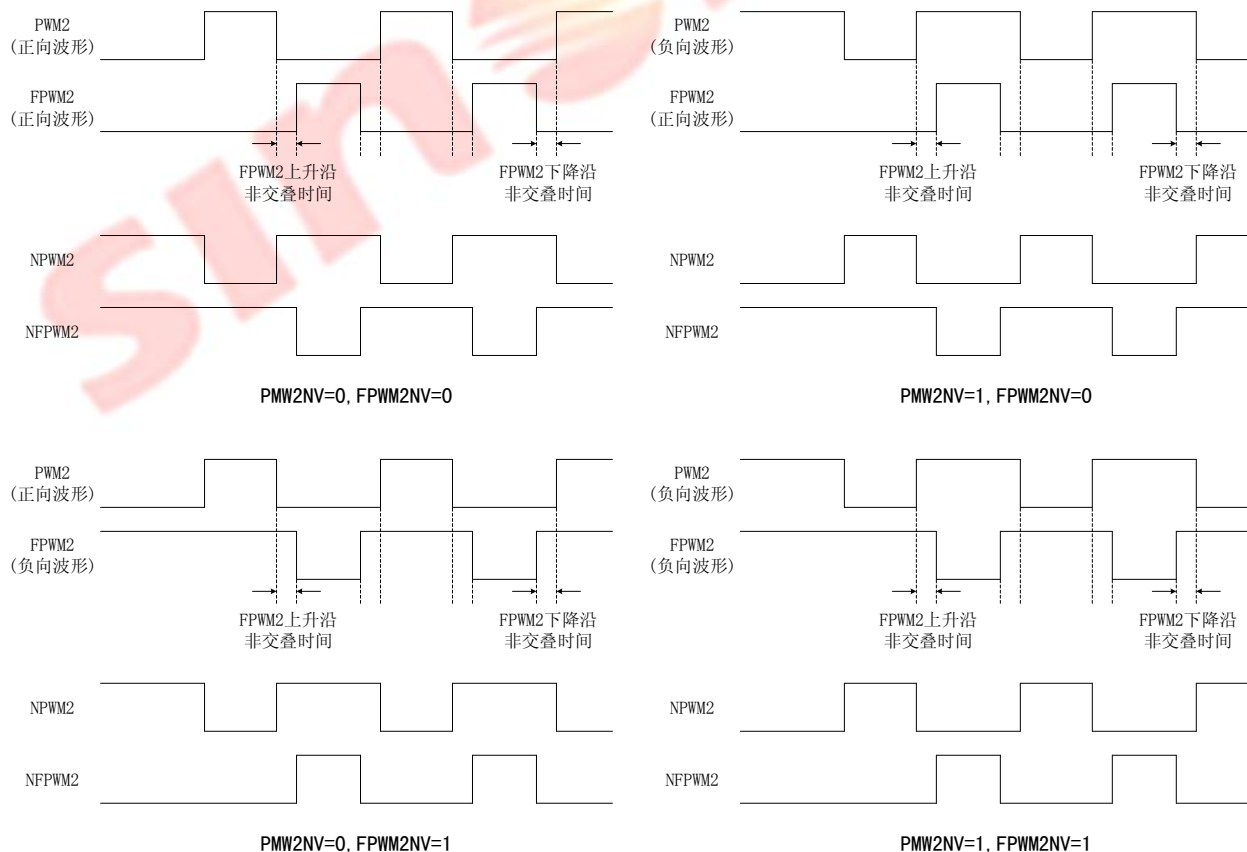
8位定时器T2示意图



定时器 T2 的定时功能、BUZ 功能与定时器 T0 完全相同。

定时器 T2 可实现 PWM 功能，并支持 8+3 模式，其功能完全与定时器 T0 相同。并且定时器 T2 除了可生成正常 PWM2 信号及带死区控制的反向互补信号 FPWM2 外，还可生成 PWM2 和 FPWM2 信号的反向信号 NPWM2 和 NFPWM2。2 路互补信号 PWM2/FPWM2 可通过输出电平控制位选择输出信号的正向波形或负向波形。4 路端口可通过输出使能位选择是否输出 PWM 波形。

PWM2 输出波形示意



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	PWM2OE	BUZ2OE	T2CKS1	T2CKS0	T2PRS2	T2PRS1	T2PRS0
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] **T2EN** – T2 使能控制位

0: 关闭 T2;

1: 开启 T2;

BIT[6] **PWM2OE** – PWM2 功能控制位

0: 关闭 PWM2 功能, 并禁止端口输出 PWM 波形;

1: 使能 PWM2 功能, 并允许端口输出 PWM 波形;

BIT[5] **BUZ2OE** – BUZ2 输出使能位

0: 关闭 BUZ2 输出;

1: 使能 BUZ2 输出 (仅 PWM2OE=0 时有效);

BIT[4:3] **T2CKS[1:0]** – T2 时钟源选择位

T2CKS[1:0]	T2 时钟源
00	FCPU
01	FTMR
10	FLOSC
11	保留

BIT[2:0] **T2PRS[2:0]** – T2 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
T2CNT	T2CNT7	T2CNT6	T2CNT5	T2CNT4	T2CNT3	T2CNT2	T2CNT1	T2CNT0
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[7:0]** – T2 计数器, 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
T2LOAD	T2LOAD7	T2LOAD6	T2LOAD5	T2LOAD4	T2LOAD3	T2LOAD2	T2LOAD1	T2LOAD0
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[7:0] – T2 重载寄存器，用于设置 T2 的计数周期

定时器 T2 比较寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2DATA	T2DATA7	T2DATA6	T2DATA5	T2DATA4	T2DATA3	T2DATA2	T2DATA1	T2DATA0
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] T2DATA[7:0] – T2 比较寄存器，用于设置 PWM2 的占空比

PWM2 占空比延展寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2DR	-	-	-	-	-	PWM2D2	PWM2D1	PWM2D0
R/W	R	R	R	R	R	R/W	R/W	R/W
初始值	1	1	1	1	0	0	0	0

BIT[2:0] PWM2D[2:0] – PWM2 占空比延展设置位（仅 PWM2MD=1 时有效）

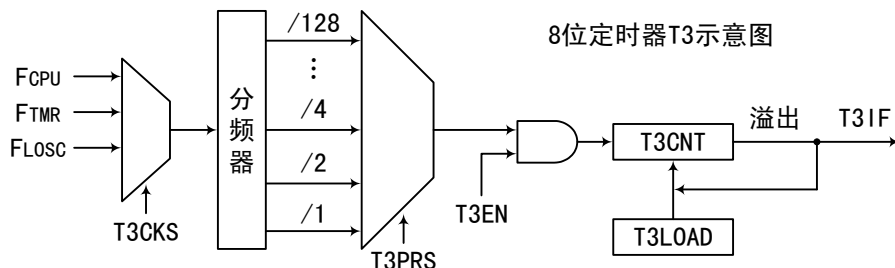
PWM2D[2:0]	PWM2 每 8 个周期的占空比延展周期
000	PWM 每 8 个周期内有 0 个周期的占空比延展
001	PWM 每 8 个周期内有 1 个周期的占空比延展（第 1 个周期）
010	PWM 每 8 个周期内有 2 个周期的占空比延展（第 1,5 个周期）
011	PWM 每 8 个周期内有 3 个周期的占空比延展（第 1,3,5 个周期）
100	PWM 每 8 个周期内有 4 个周期的占空比延展（第 1,3,5,7 个周期）
101	PWM 每 8 个周期内有 5 个周期的占空比延展（第 1,2,3,5,7 个周期）
110	PWM 每 8 个周期内有 6 个周期的占空比延展（第 1,2,3,5,6,7 个周期）
111	PWM 每 8 个周期内有 7 个周期的占空比延展（第 1,2,3,4,5,6,7 个周期）

7.5 定时器 T3

定时器 T3 为 8 位定时器，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。

- ✧ 可通过预分频比设置计数频率，可通过重载寄存器控制计数周期；
- ✧ 支持溢出中断和溢出唤醒功能；

定时器 T0 的定时功能与定时器 T0 完全相同。



定时器 T3 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3CR	T3EN	-	-	T3CKS1	T3CKS0	T3PRS2	T3PRS1	T3PRS0
R/W	R/W	-	-	R/W	R/W	R/W	R/W	R/W
初始值	0	-	-	0	0	0	0	0

BIT[7] **T3EN** – T3 使能控制位

0: 关闭 T3;

1: 开启 T3;

BIT[4:3] **T3CKS[1:0]** – T3 时钟源选择位

T3CKS[1:0]	T3 时钟源
00	FCPU
01	FTMR
10	FLOSC
11	保留

BIT[2:0] **T3PRS[2:0]** – T3 预分频比选择位

T3PRS[2:0]	T3 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 T3 计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3CNT	T3CNT7	T3CNT6	T3CNT5	T3CNT4	T3CNT3	T3CNT2	T3CNT1	T3CNT0
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] **T3CNT[7:0]** – T3 计数器，为可读写的递减计数器

定时器 T3 重载寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3LOAD	T3LOAD7	T3LOAD6	T3LOAD5	T3LOAD4	T3LOAD3	T3LOAD2	T3LOAD1	T3LOAD0
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] **T3LOAD[7:0]** – T3 重载寄存器，用于设置 T3 的计数周期

7.6 PWM 相关寄存器

PWM 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCR0	FPWM1OE	PWM2NV	FPWM1NV	PWM0NV	FPWM1R1	FPWM1R0	FPWM1F1	FPWM1F0
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] **FPWM1OE** – FPWM1 端口输出使能位

0: 禁止端口输出 PWM 波形;

1: 允许端口输出 PWM 波形 (仅 PWM1OE=1 时有效);

BIT[6] **PWM2NV** – PWM2 信号输出电平控制位

0: 端口输出 PWM2 信号的正向波形;

1: 端口输出 PWM2 信号的负向波形;

BIT[5] **FPWM1NV** – FPWM1 信号输出电平控制位 (PWM1 输出电平不受影响)

0: 端口输出 FPWM1 信号的正向波形;

1: 端口输出 FPWM1 信号的负向波形;

BIT[4] **PWM0NV** – PWM0 信号输出电平控制位

0: 端口输出 PWM0 信号的正向波形;

1: 端口输出 PWM0 信号的负向波形;

BIT[3:2] **FPWM1R[1:0]** – FPWM1 上升沿非交叠时间选择位

FPWM1R[1:0]	上升沿非交叠时间宽度
00	1 个 PWM 时钟周期
01	2 个 PWM 时钟周期
10	3 个 PWM 时钟周期
11	4 个 PWM 时钟周期

BIT[1:0] **FPWM1F[1:0]** – FPWM1 下降沿非交叠时间选择位

FPWM1F[1:0]	下降沿非交叠时间宽度
00	2 个 PWM 时钟周期

01	4 个 PWM 时钟周期
10	6 个 PWM 时钟周期
11	8 个 PWM 时钟周期

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCR1	-	-	-	TMRCKS1	TMRCKS0	PWM2MD	PWM1MD	PWM0MD
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	1	0	0	0	0

BIT[4:3] **TMRCKS[1:0]** – 定时器高频时钟 F_{TMR} 选频控制位

TMRCKS[1:0]	F_{TMR} 频率
00	$F_{HOSC}/2$
01	
10	F_{HOSC}
11	$F_{HOSC} \times 2$

BIT[2] **PWM2MD** – PWM2 工作模式控制位

0: PWM2 工作于普通模式;

1: PWM2 工作于 8+3 模式;

BIT[1] **PWM1MD** – PWM1 工作模式控制位

0: PWM1 工作于普通模式;

1: PWM1 工作于 8+3 模式;

BIT[0] **PWM0MD** – PWM0 工作模式控制位

0: PWM0 工作于普通模式;

1: PWM0 工作于 8+3 模式;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCR2	-	-	FPWM2D5	FPWM2D4	FPWM2D3	FPWM2D2	FPWM2D1	FPWM2D0
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] **FPWM2D[5:0]** – FPWM2 上升/下降沿非交叠时间选择位

FPWM2D[5:0]	上升沿非交叠时间宽度	下降沿非交叠时间宽度
00 0000	1 个 PWM 时钟周期	1 个 PWM 时钟周期
00 0001	2 个 PWM 时钟周期	2 个 PWM 时钟周期
.....		
11 1110	63 个 PWM 时钟周期	63 个 PWM 时钟周期
11 1111	64 个 PWM 时钟周期	64 个 PWM 时钟周期

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCR3	FPWM2NV	FPWM2OE	NPWM2OE	NFPWM2OE	-	PWM0OS	-	-
R/W	R/W	R/W	R/W	R/W	-	R/W	-	-
初始值	0	0	0	0	-	0	-	-

- hr/>
- BIT[7] **FPWM2NV** – FPWM2 信号输出电平控制位
0: 端口输出 FPWM2 信号的正向波形;
1: 端口输出 FPWM2 信号的负向波形;
- BIT[6] **FPWM2OE** – FPWM2 端口输出使能位
0: 禁止端口输出 PWM 波形;
1: 允许端口输出 PWM 波形 (仅 PWM2OE=1 时有效);
- BIT[5] **NPWM2OE** – NPWM2 端口输出使能位
0: 禁止端口输出 PWM 波形;
1: 允许端口输出 PWM 波形 (仅 PWM2OE=1 时有效);
- BIT[4] **NFPWM2OE** – NFPWM2 端口输出使能位
0: 禁止端口输出 PWM 波形;
1: 允许端口输出 PWM 波形 (仅 PWM2OE=1 时有效);
- BIT[2] **PWM0OS** – PWM0 输出通道选择位
0: PWM0 波形从 P16 口输出;
1: PWM0 波形从 P02 口输出;

8 模数转换器 ADC

8.1 ADC 概述

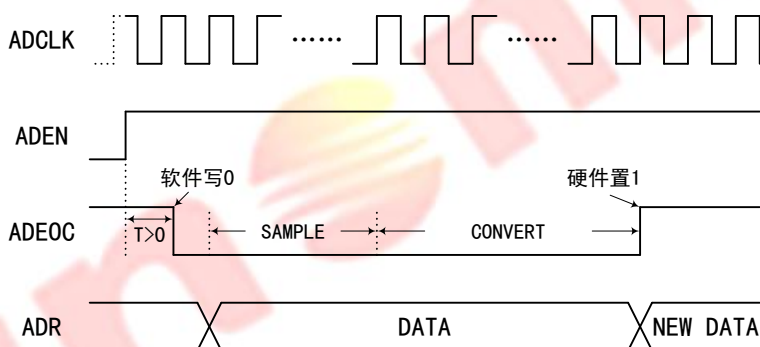
芯片内置 1 个 12 位高精度逐次逼近型的模数转换器 ADC。

- ◇ 11 路外部通道：AN0~AN10；3 路内部通道：VDD/4、OUTV0/4、OUTV1/4；
- ◇ 参考电压可选：VDD、内部参考电压 V_{IR} (2V/3V/4V)、外部参考电压 V_{ER} (V_{ER1} 输入)；
- ◇ ADC 时钟：F_{HIRC} 的 32/64/128/256 分频；

ADC 模块可通过寄存器位 ADEN 开启，通过 ADCKS 选择转换时钟，通过 ADCHS 选择转换的模拟通道，通过 ADEOC 启动并标识 ADC 的转换状态。当 ADEOC 为 1 时对其写 0 将启动模数转换；转换完成后结果存入 ADRH/ADRL 中，ADEOC 自动置 1，同时中断标志 ADIF 置 1 触发 ADC 中断。

采样 (SAMPLE) 时间固定为 15 个 ADCLK (即 ADC 时钟周期)，转换 (CONVERT) 时间固定为 12 个 ADCLK，一次 ADC 转换为 27 个 ADCLK。

ADC 转换时序如下图所示：



注：

- 1、AD 转换过程中或者 ADEN 未使能时，ADRH/ADRL 中的数据未知，应在 AD 转换结束且 ADEN 使能的情况下读取 AD 转换数据；
- 2、若选择内部参考电压 V_{IR} ，则需保证 $VDD > (V_{IR} + 0.5V)$ ，否则 V_{IR} 将随之下降；
- 3、使能 ADC 模块、切换参考电压等操作后，需待电路稳定 (时间 $> 200\mu s$) 后才能启动 AD 转换；切换输入通道后，受外部输入影响，前两次转换的结果会有误差，建议舍弃；
- 4、AD 转换精度受参考电压精度的影响，且内部参考电压下的转换精度，比外部参考电压略低 2 个 LSB 左右；
- 5、转换时钟越慢、采样时间越长，则越能过滤外部输入的波动，越能保证 AD 转换的精度；

8.2 ADC 相关寄存器

ADC 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCR0	ADCHS3	ADCHS2	ADCHS1	ADCHS0	ADRSEL		ADEOC	ADEN
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	0

BIT[7:4] **ADCHS[3:0]** – ADC 模拟通道选择位

ADCHS[3:0]	ADC 模拟通道选择
0000	AN0
0001	AN1
0010	AN2
0011	AN3
0100	AN4
0101	AN5
0110	AN6
0111	AN7
1000	AN8
1001	AN9
1010	VDD/4
1011	OUTV0/4
1100	OUTV1/4
1101	保留
1110	保留
1111	AN10

BIT[3] **ADRSEL** – ADC 转换结果位数选择位

- 0: ADC 转换结果为 8 位;
- 1: ADC 转换结果为 12 位;

BIT[1] **ADEOC** – ADC 启动位及转换结束标志位

- 0: AD 转换过程中, 转换结束后自动置 1;
- 1: AD 转换结束, 对 ADEOC 写入 0 启动 AD 转换;

BIT[0] **ADEN** – ADC 功能使能位

- 0: 关闭 ADC 功能;
- 1: 使能 ADC 功能;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCR1	ADCKS2	ADCKS1	ADCKS0	保留	RESS0	ADVRS2	ADVRS1	ADVRS0
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[7:5] **ADCKS[2:0]** – ADC 转换时钟选择位

ADCKS[2:0]	ADC 转换时钟 F _{ADC}
100	1MHz (F _{HIRC} /32)
101	500KHz (F _{HIRC} /64)
110	250KHz (F _{HIRC} /128)
111	125KHz (F _{HIRC} /256)

BIT[4] 保留位，需固定写“0”

BIT[3] **RESS0** – VREF 端口内部 3KΩ 上拉电阻使能位

0: 关闭 3KΩ 上拉;

1: 使能 3KΩ 上拉;

BIT[2:0] **ADVRS[2:0]** – ADC 参考电压选择位

ADVRS[2:0]	ADC 参考电压选择
000	内部 2V
001	内部 3V
010	内部 4V
011	VDD
100-110	保留
111	外部 VREF

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCR2	-	-	-	-	ADSPS3	ADSPS2	ADSPS1	ADSPS0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	1	0	0	0

BIT[3:0] **ADSPS[3:0]** – ADC 采样时间选择位

0000-1110: 禁用;

11111: ADC 采样时间为 15 个 ADC 时钟;

ADC 转换结果寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRH	ADR11	ADR10	ADR9	ADR8	ADR7	ADR6	ADR5	ADR4
R/W	R	R	R	R	R	R	R	R
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **ADR[11:4]** – ADC 转换结果高 8 位

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRL	-	-	-	-	ADR3	ADR2	ADR1	ADR0
R/W	-	-	-	-	R	R	R	R
初始值	-	-	-	-	X	X	X	X

BIT[3:0] **ADR[3:0]** – ADC 转换结果低 4 位

8.3 ADC 操作步骤

模数转换操作步骤:

- (1) 设置相应端口为输入端口，关闭上/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) 若转换时钟可选，则设置 ADCKS，选择适当的 ADC 转换时钟；
- (4) 若采样时间可选，则设置 ADSPS，选择适当的 ADC 采样时间；
- (5) 若参考电压可选，则设置 ADVRS，选择适当的参考电压；
- (6) 若数据格式可选，则设置 ADRSEL，选择 ADC 转换结果的数据格式；
- (7) ADEN 置 1，使能 ADC 模块；
- (8) 设置 ADCHS，选择 ADC 转换通道；
- (9) 延时等待电路稳定后，ADEOC 写 0，启动 AD 转换；
- (10) 等待 ADEOC 硬件置 1（或利用 ADC 中断）；
- (11) 读取 ADC 转换结果（ADRH、ADRL）；
- (12) 重复执行（8）~（11），对不同的通道进行转换或对同一通道进行多次转换；

9 低电压检测 LVD

芯片内嵌低电压检测模块，通过设置 **LV DEN** 可以开启或关闭该功能，**VL VDS** 选择电压比较值，可设置多个电压比较点，当 **VDD** 电压低于比较电压时 **LV DF** 将置 1，否则 **LV DF** 将清 0。

LVD 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDCR	LV DEN	VL VDS3	VL VDS2	VL VDS1	VL VDS0	-	-	LV DF
R/W	R/W	R/W	R/W	R/W	R/W	-	-	R
初始值	0	0	0	0	0	-	-	0

BIT[7] **LV DEN** – LVD 功能使能位

0: 关闭 LVD 功能;

1: 开启 LVD 功能;

BIT[6:3] **VL VDS[3:0]** – LVD 比较电压选择位

VL VDS[3:0]	LVD 电压比较值
0000	1.8V
0001	LVDI 输入电压与内部 1V 比较
0010	2.0V
0011	2.1V
0100	2.2V
0101	2.4V
0110	2.5V
0111	2.6V
1000	2.7V
1001	2.8V
1010	3.0V
1011	3.2V
1100	3.3V
1101	3.6V
1110	4.0V
1111	4.2V

BIT[0] **LV DF** – LVD 检测标志位

0: VDD 电压高于比较电压，或 LVD 关闭;

1: VDD 电压低于比较电压;

10 中断

芯片的中断源包括外部中断（INT0~INT1）、定时器中断（T0~T3）、ADC 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，自动将当前指令的下一条要执行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，而是暂时挂起中断直到当前指令执行完成。
- ✧ CPU 响应中断后，程序跳到中断向量地址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 PFLAG，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 PFLAG，然后执行 RETIE 返回主程序。此时芯片将自动恢复 GIE 为 1，然后从堆栈取出 PC 值，从中断产生时当前指令的下一条指令继续执行。

芯片的中断向量地址是 0008H。

10.1 外部中断

芯片有 2 路外部中断源 INT0/INT1，可选择上升沿、下降沿或电平变化等触发方式。外部中断触发时，中断标志 INTnIF (n=0-1) 将被置 1，若中断总使能位 GIE 为 1 且外部中断使能位 INTnIE (n=0-1) 为 1，则产生外部中断。

10.2 定时器中断

定时器 Tn (n=0-3) 在计数溢出时触发定时器中断，中断标志 TnIF (n=0-3) 将被置 1，若中断总使能位 GIE 为 1 且定时器中断使能位 TnIE (n=0-3) 为 1，则产生定时器中断。

10.3 ADC 中断

ADC 转换完成后触发 ADC 中断，中断标志 ADIF 将被置 1，若中断总使能位 GIE 为 1 且 ADC 中断使能位 ADIE 为 1，则产生 ADC 中断。

10.4 键盘中断

芯片有 8 路键盘中断源，均可通过寄存器位单独使能或屏蔽，任意一路使能的中断源的输入电平发生变化时，均会触发键盘中断，中断标志 KBIF 将被置 1，若中断总使能位 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	P17KE	P16KE	P15KE	P14KE	P13KE	P12KE	P11KE	P10KE
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] **P1nKE** – P1n 端口键盘中断使能位 (n=7-0)

0: 屏蔽端口键盘中断功能;

1: 使能端口键盘中断功能;

10.5 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	T3IE	ADIE	T2IE	KBIE	INT1IE	INT0IE	T1IE	TOIE
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] **T3IE** – 定时器 T3 中断使能位

0: 屏蔽定时器 T3 中断;

1: 使能定时器 T3 中断;

BIT[6] **ADIE** – ADC 中断使能位

0: 屏蔽 ADC 中断;

1: 使能 ADC 中断;

BIT[5] **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[4] **KBIE** – 键盘中断使能位

0: 屏蔽键盘中断;

1: 使能键盘中断;

BIT[3] **INT1IE** – INT1 中断使能位

0: 屏蔽 INT1 中断;

1: 使能 INT1 中断;

BIT[2] **INT0IE** – INT0 中断使能位

0: 屏蔽 INT0 中断;

1: 使能 INT0 中断;

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] **T0IE** – 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	T3IF	ADIF	T2IF	KBIF	INT1IF	INT0IF	T1IF	T0IF
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] **T3IF** – 定时器 T3 中断标志位

- 0: 未触发定时器 T3 中断;
- 1: 已触发定时器 T3 中断, 需软件清零;

BIT[6] **ADIF** – ADC 中断标志位

- 0: 未触发 ADC 中断;
- 1: 已触发 ADC 中断, 需软件清零;

BIT[5] **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, 需软件清零;

BIT[4] **KBIF** – 键盘中断标志位

- 0: 未触发键盘中断;
- 1: 已触发键盘中断, 需软件清零;

BIT[3] **INT1IF** – INT1 中断标志位

- 0: 未触发 INT1 中断;
- 1: 已触发 INT1 中断, 需软件清零;

BIT[2] **INT0IF** – INT0 中断标志位

- 0: 未触发 INT0 中断;
- 1: 已触发 INT0 中断, 需软件清零;

BIT[1] **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, 需软件清零;

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, 需软件清零;

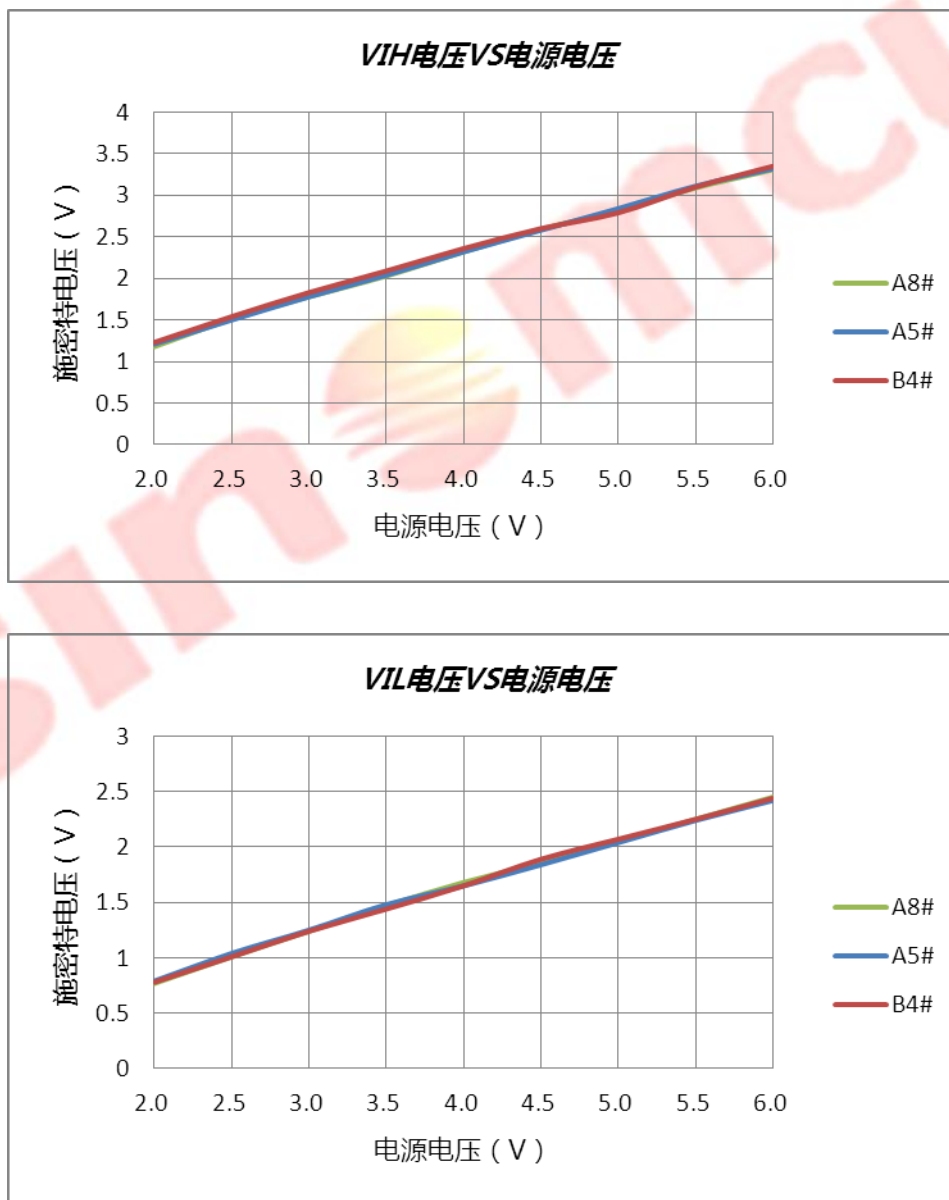
11 特性曲线

注：

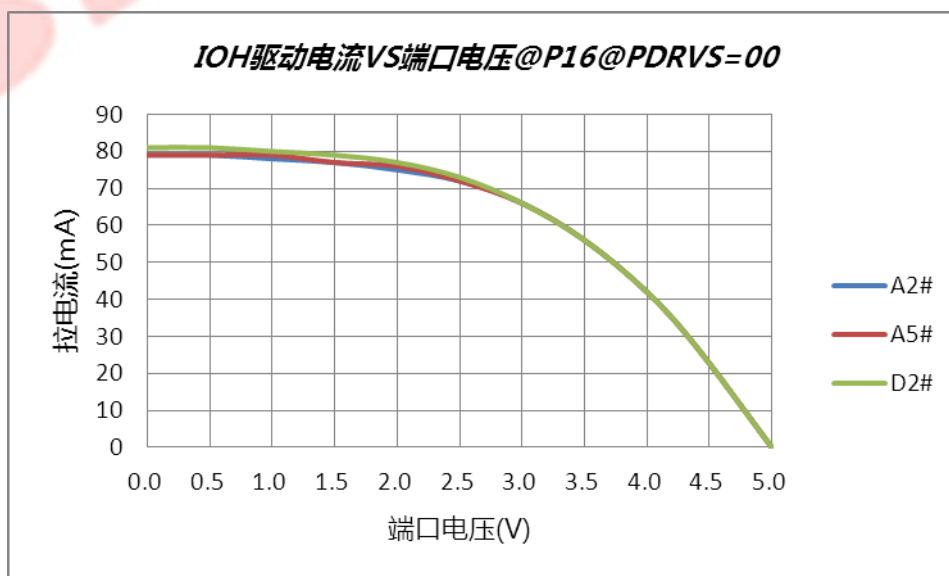
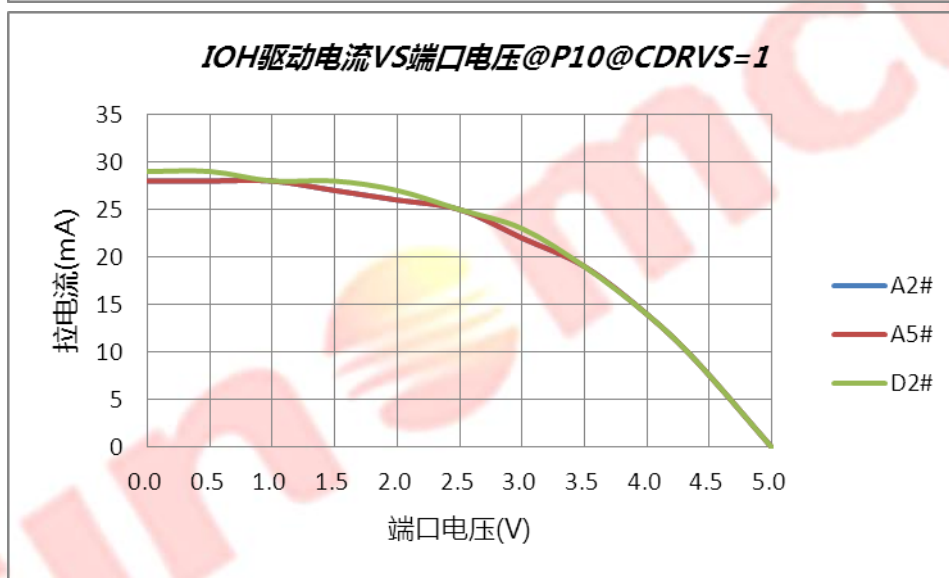
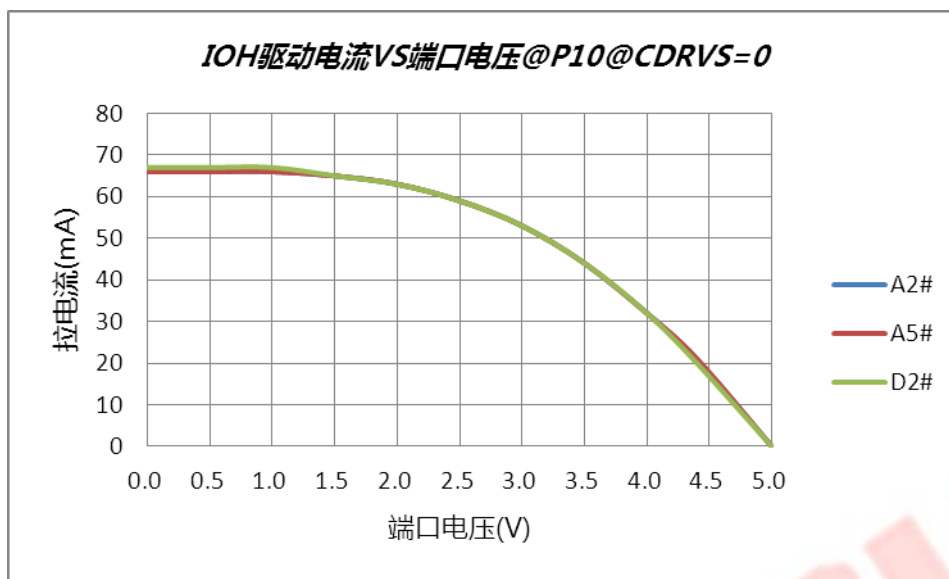
- 1、本章节所列特性曲线图为抽样实测数据，仅作为应用参考，部分数据因生产工艺偏差，可能与实际芯片不符；为保证芯片能正常工作，请确保其工作条件符合电气特性参数说明；
- 2、若图文中无特别说明，则电压特性曲线的温度条件为 $T=25^{\circ}\text{C}$ ，温度特性曲线的电压条件为 $V_{DD}=5\text{V}$ ；

11.1 I/O 特性

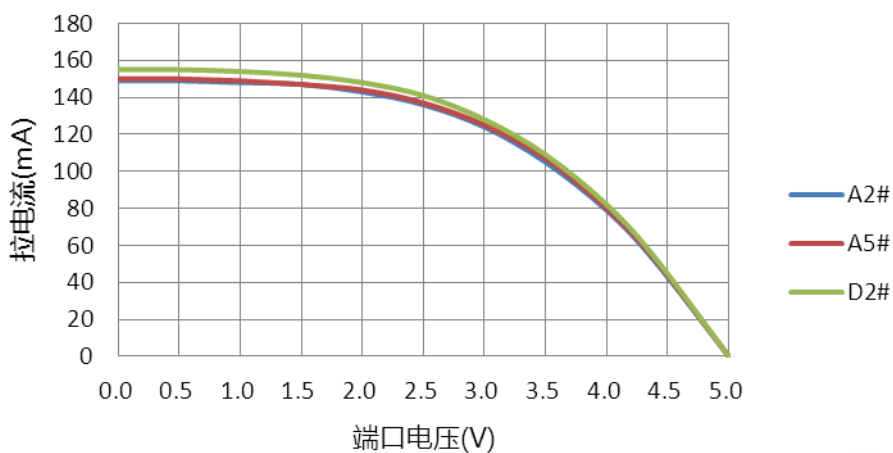
输入 SMT 阈值电压 VS 电源电压



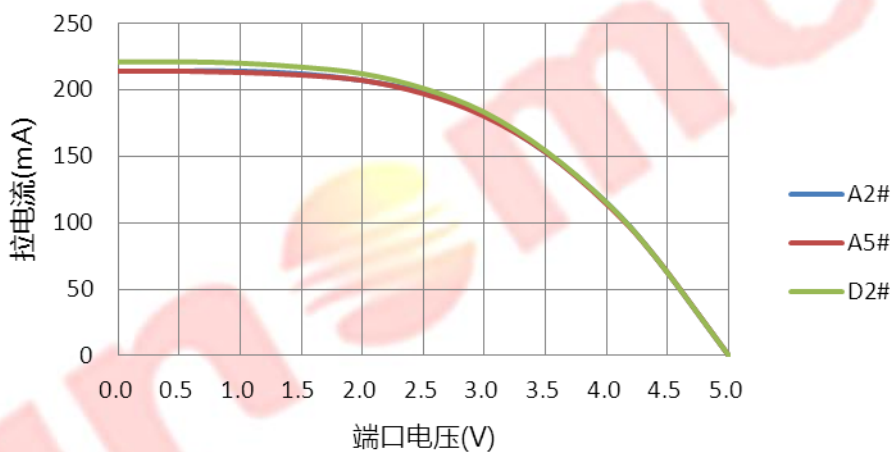
I/O 输出 驱动电流 VS 端口电压 (VDD=5V)



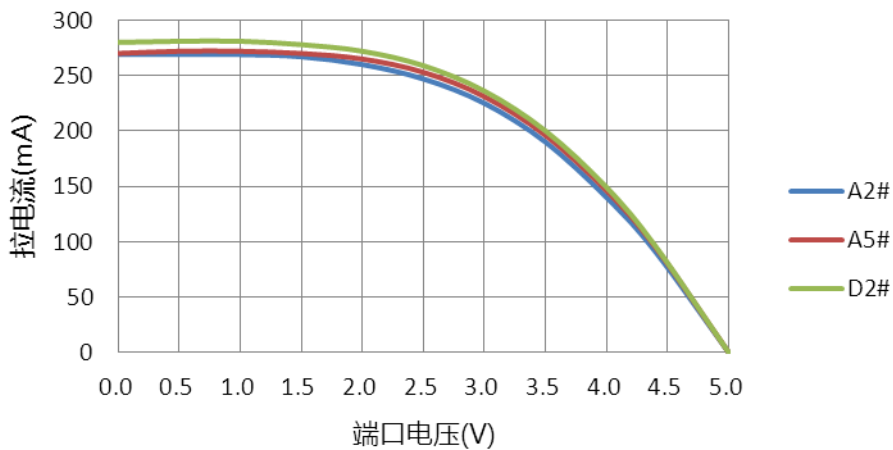
IOH驱动电流VS端口电压@P16@PDRVS=01



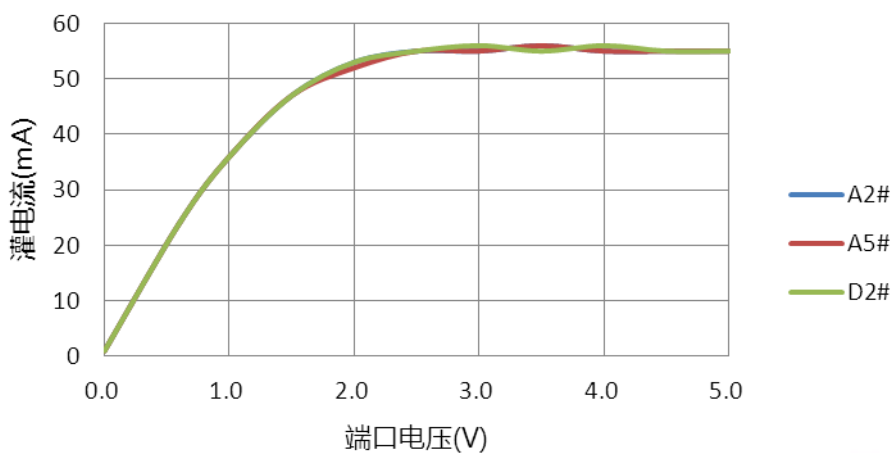
IOH驱动电流VS端口电压@P16@PDRVS=10



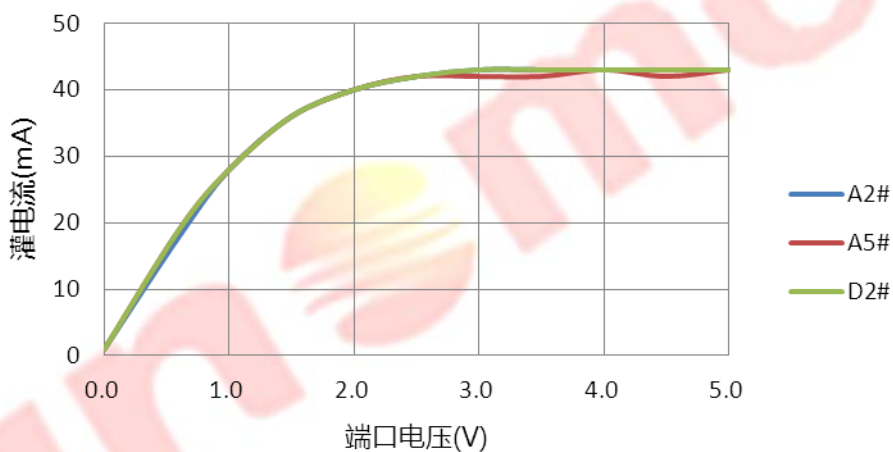
IOH驱动电流VS端口电压@P16@PDRVS=11



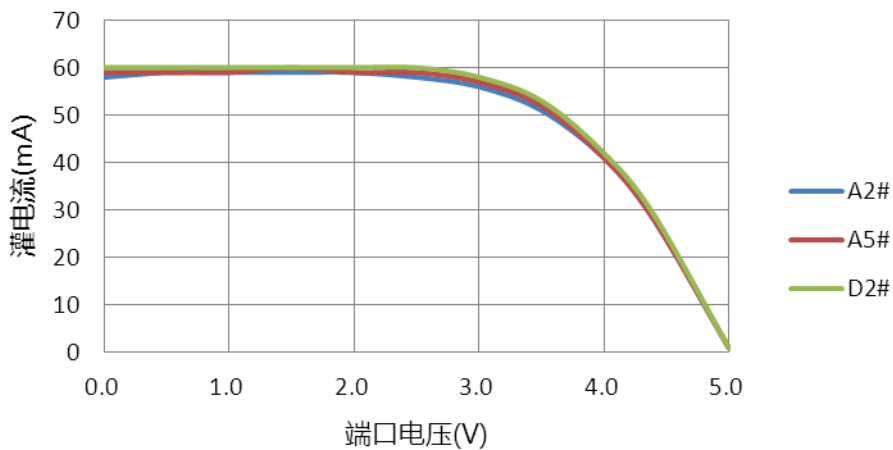
IOL驱动电流VS端口电压@P11@CDRVS=0



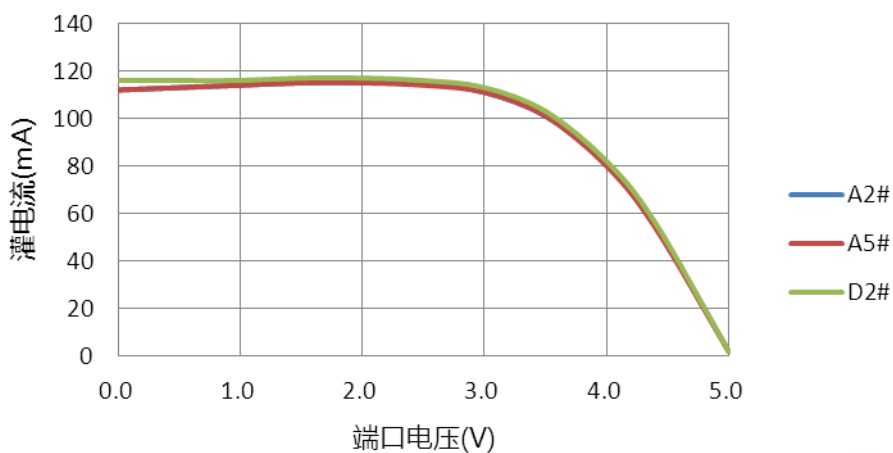
IOL驱动电流VS端口电压@P11@CDRVS=1



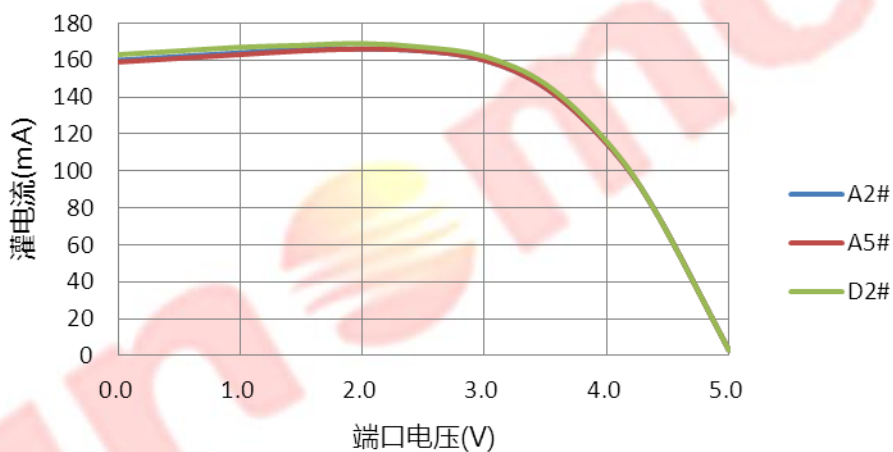
IOL驱动电流VS端口电压@P17@PDRVS=00



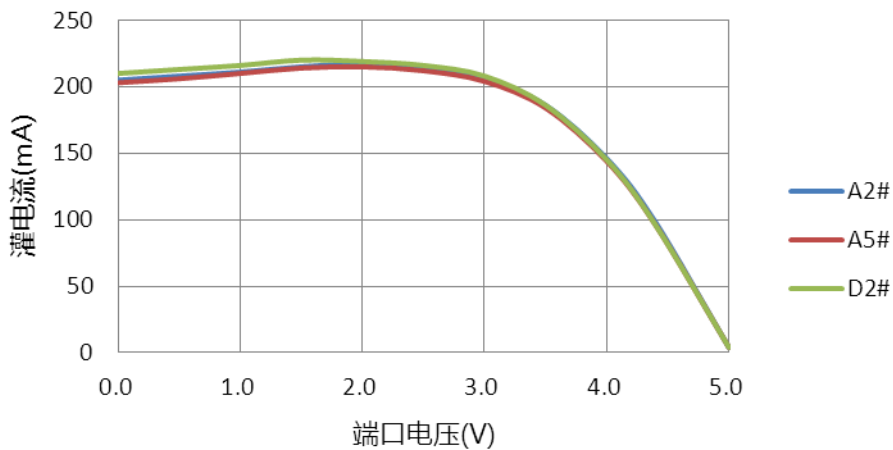
IOL驱动电流VS端口电压@P17@PDRVS=01



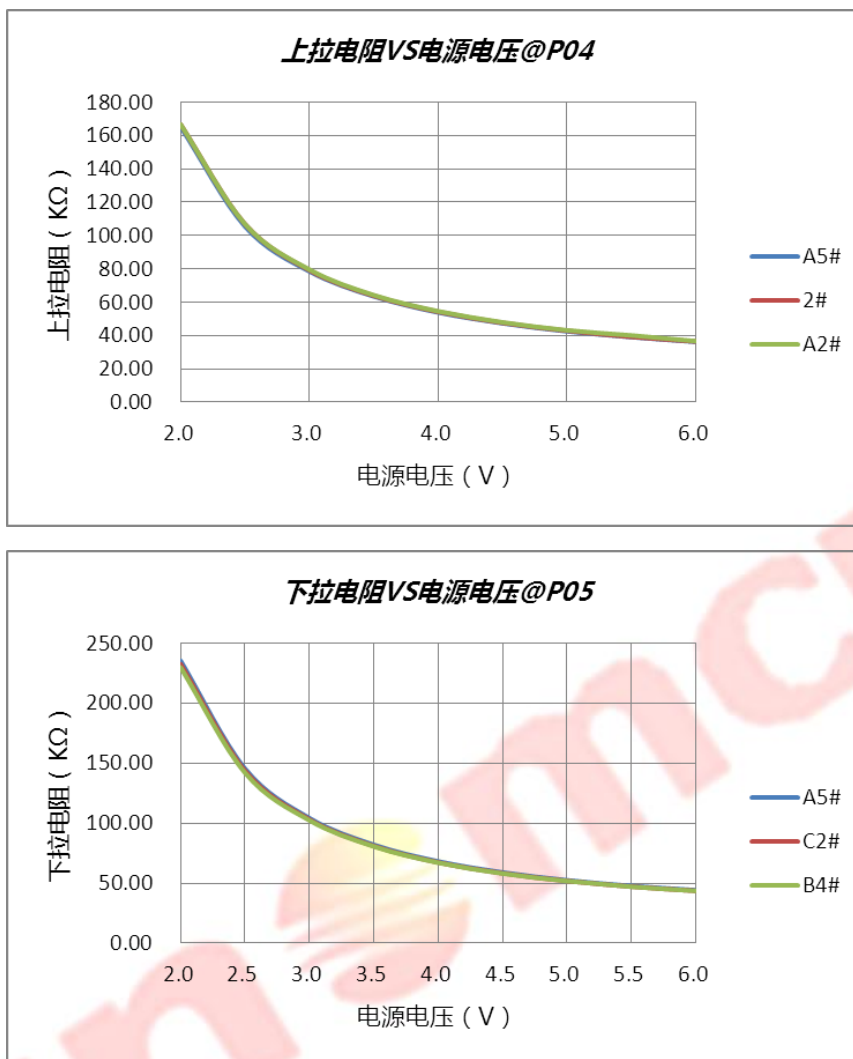
IOL驱动电流VS端口电压@P17@PDRVS=10



IOL驱动电流VS端口电压@P17@PDRVS=11

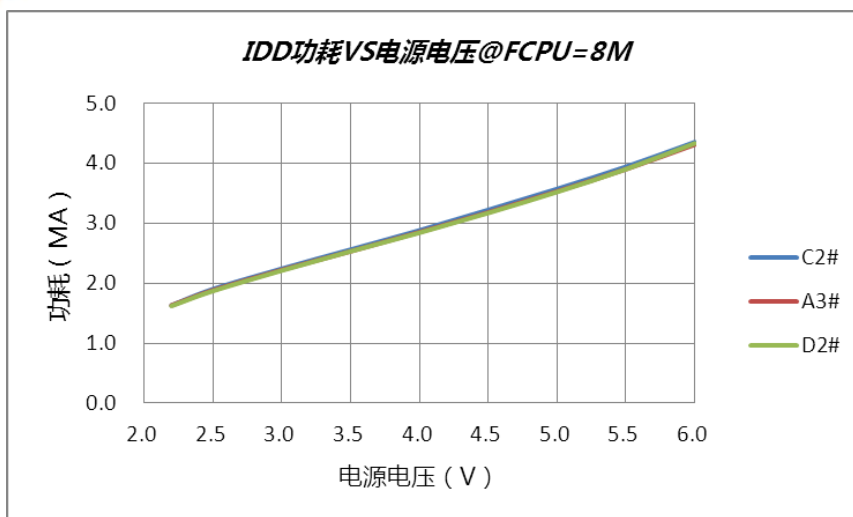


上/下拉电阻值 VS 电源电压

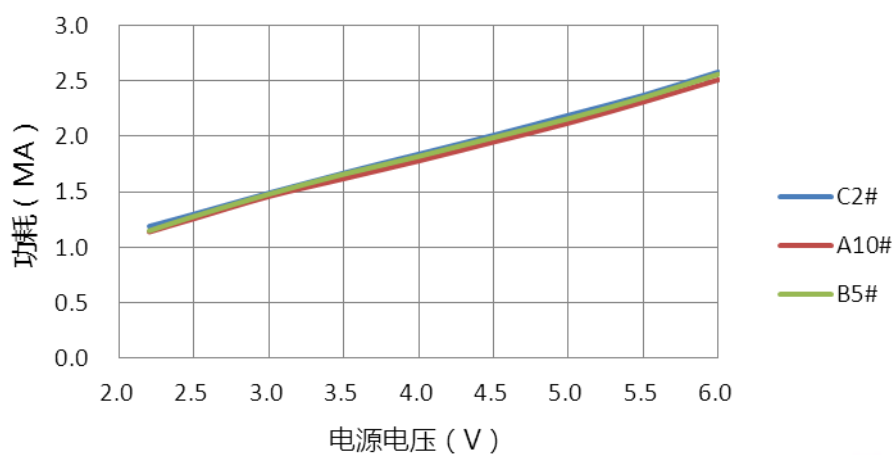


11.2 功耗特性

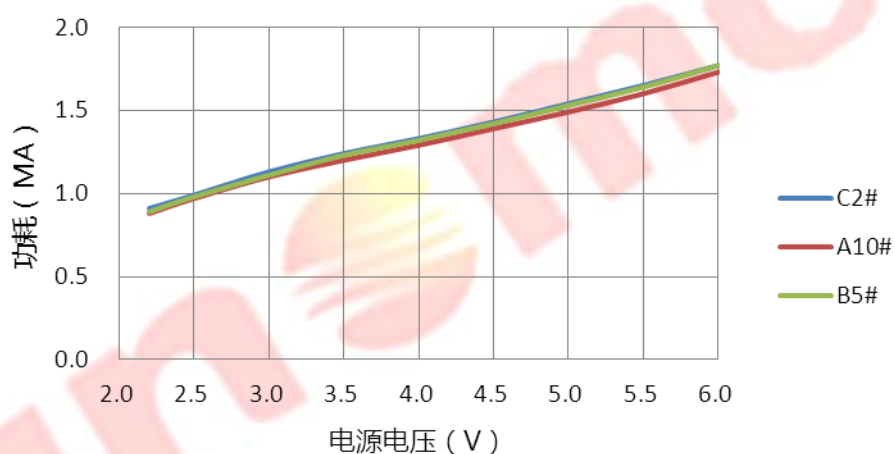
运行模式 功耗 VS 电源电压



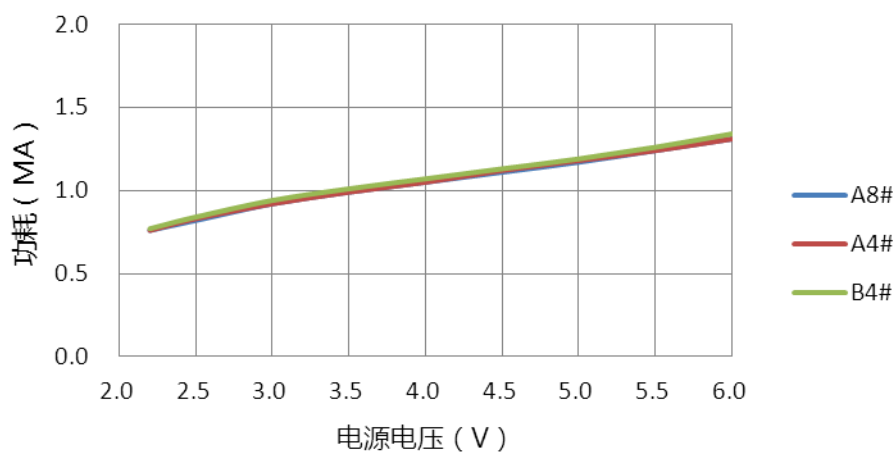
IDD功耗VS电源电压@FCPU=4M



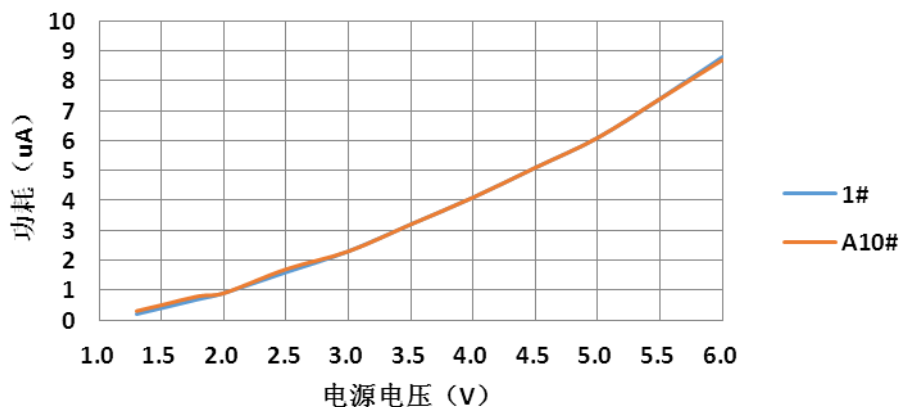
IDD功耗VS电源电压@FCPU=2M



IDD功耗VS电源电压@FCPU=1M

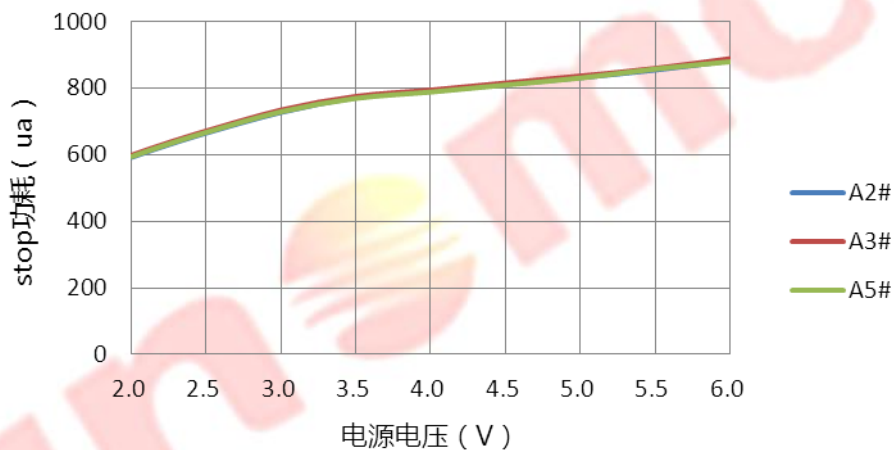


IDD功耗VS电源电压@FCPU=16K

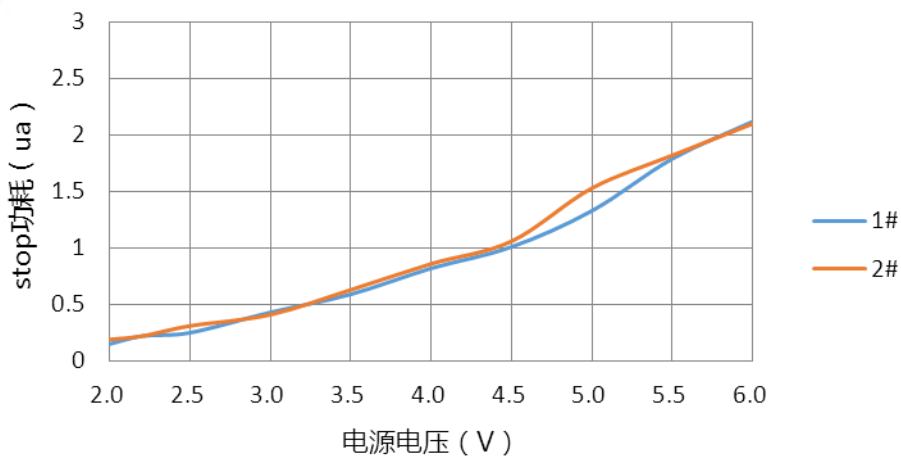


HOLD 模式 功耗 VS 电源电压

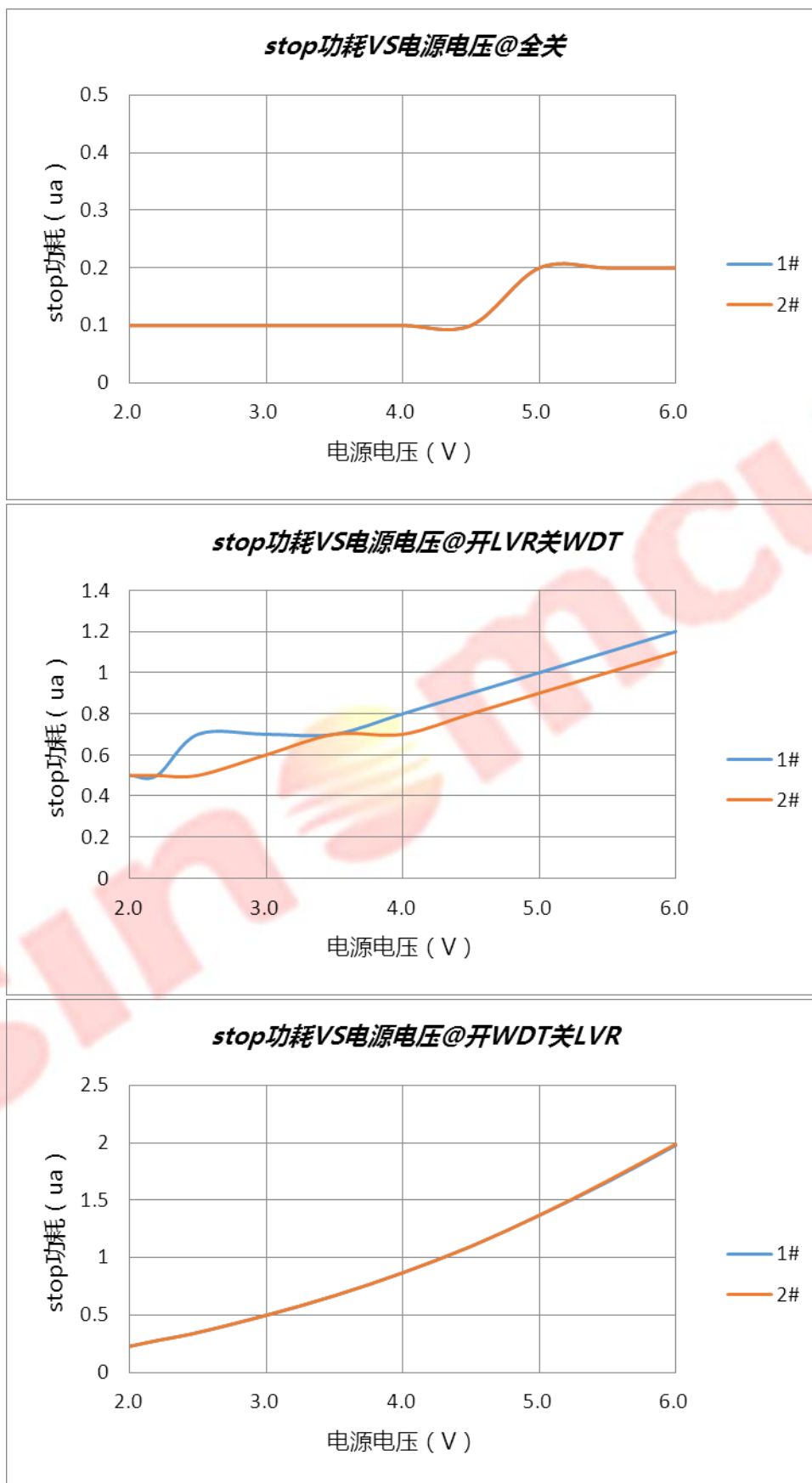
stop功耗VS电源电压@hold1



stop功耗VS电源电压@hold2

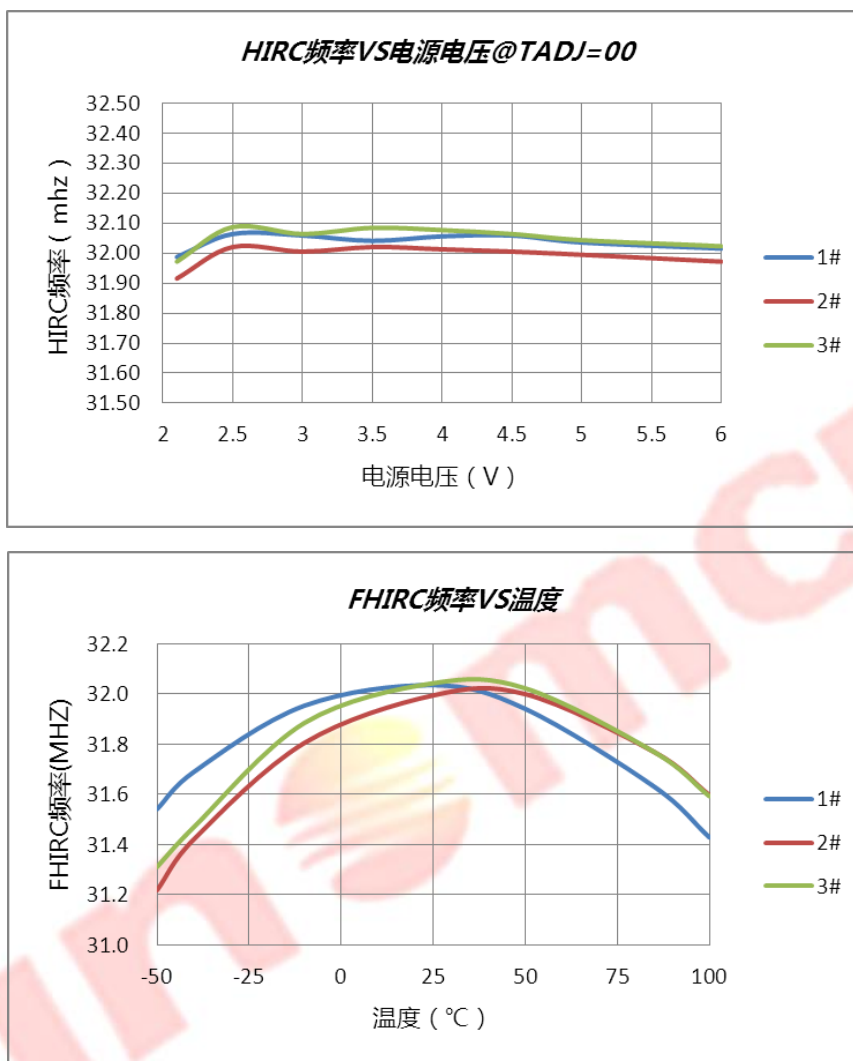


休眠模式 功耗 VS 电源电压

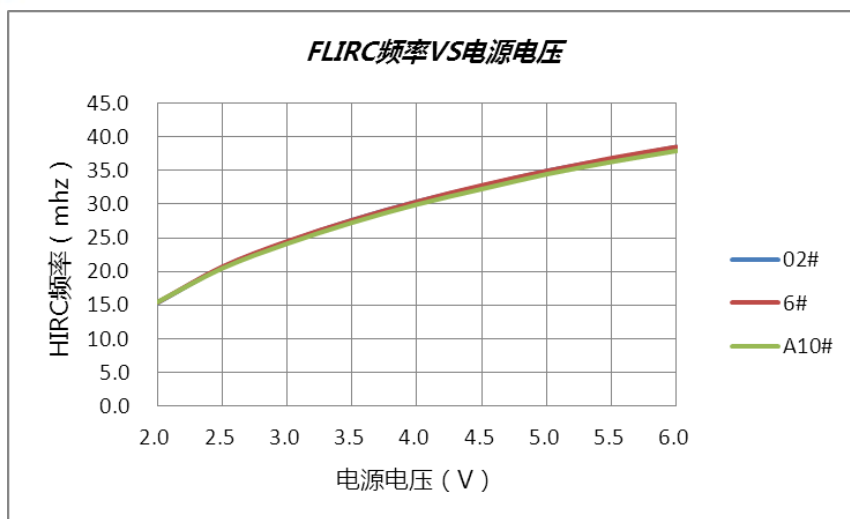


11.3 模拟电路特性

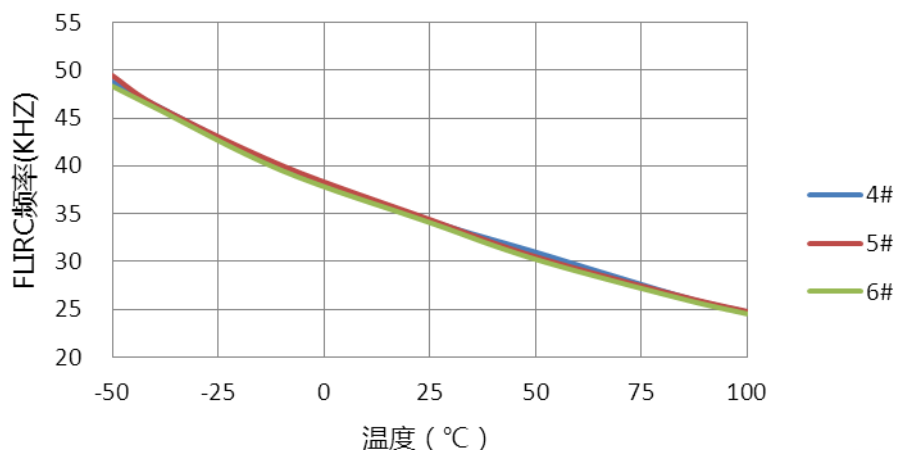
HIRC 频率 VS 电源电压/温度



LIRC 频率 VS 电源电压/温度

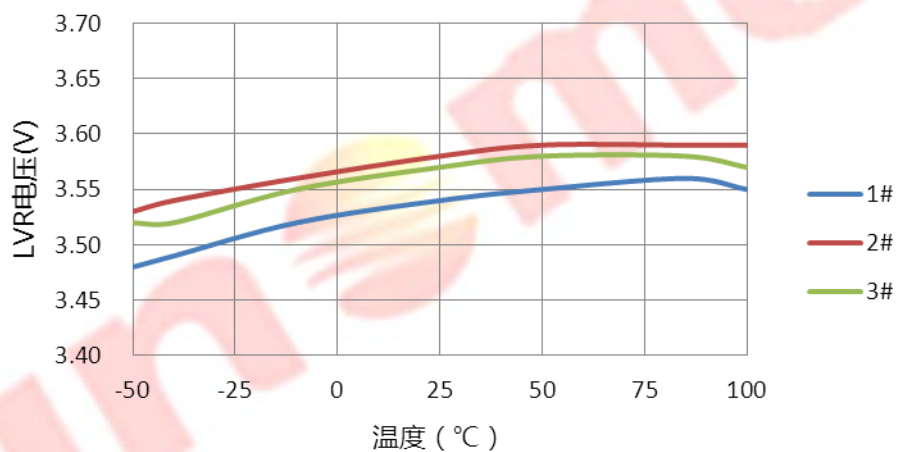


FLIRC频率VS温度

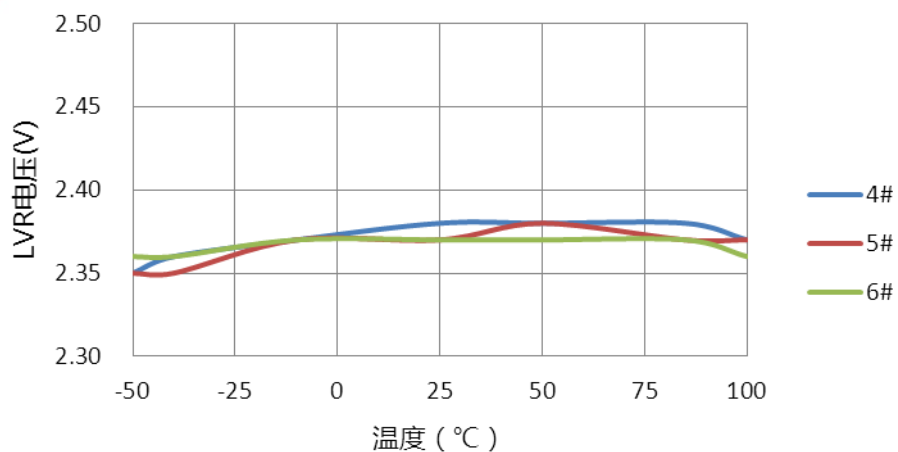


LVR 电压 VS 温度

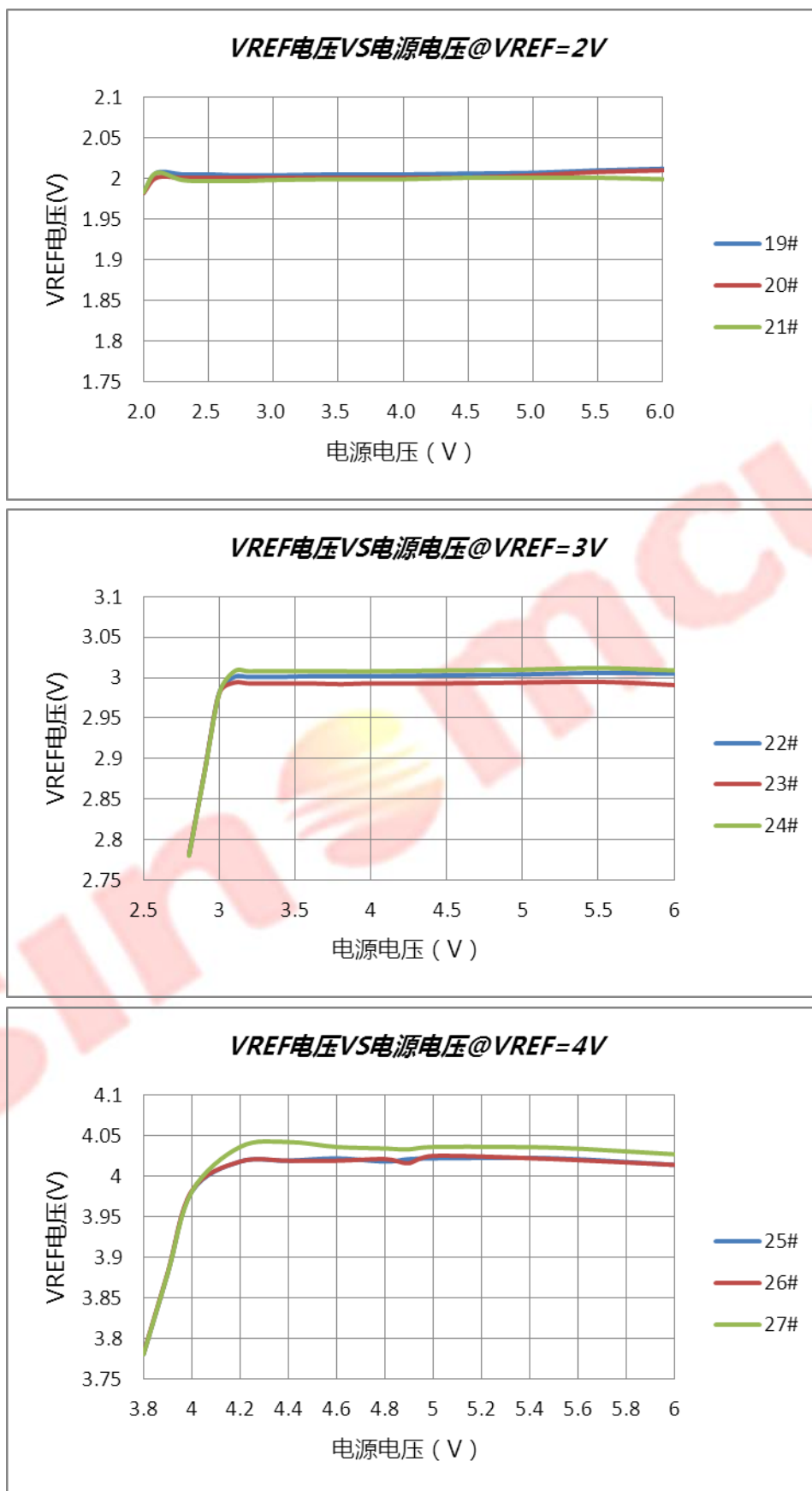
LVR电压VS温度@3.6V



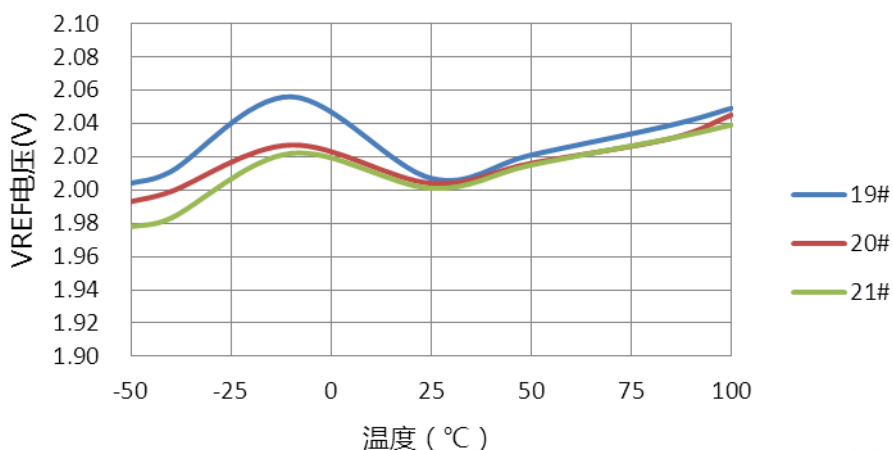
LVR电压VS温度@2.4V



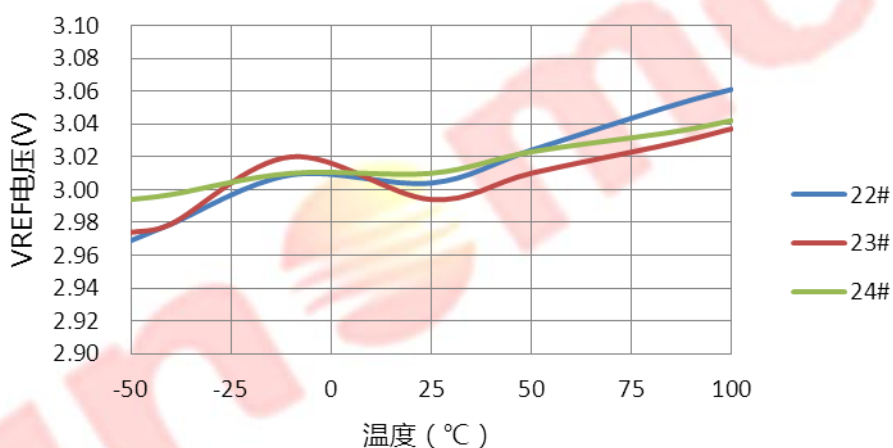
VREF 电压 VS 电源电压/温度



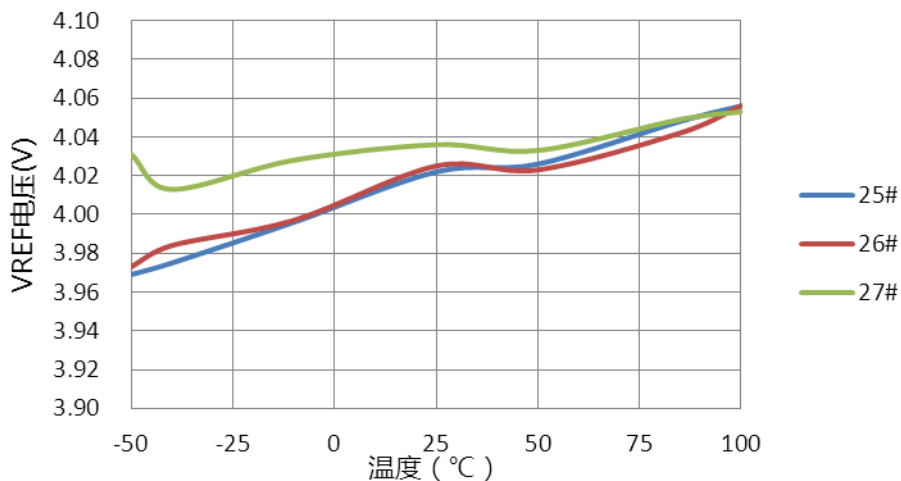
VREF电压VS温度@VREF=2V@工作状态



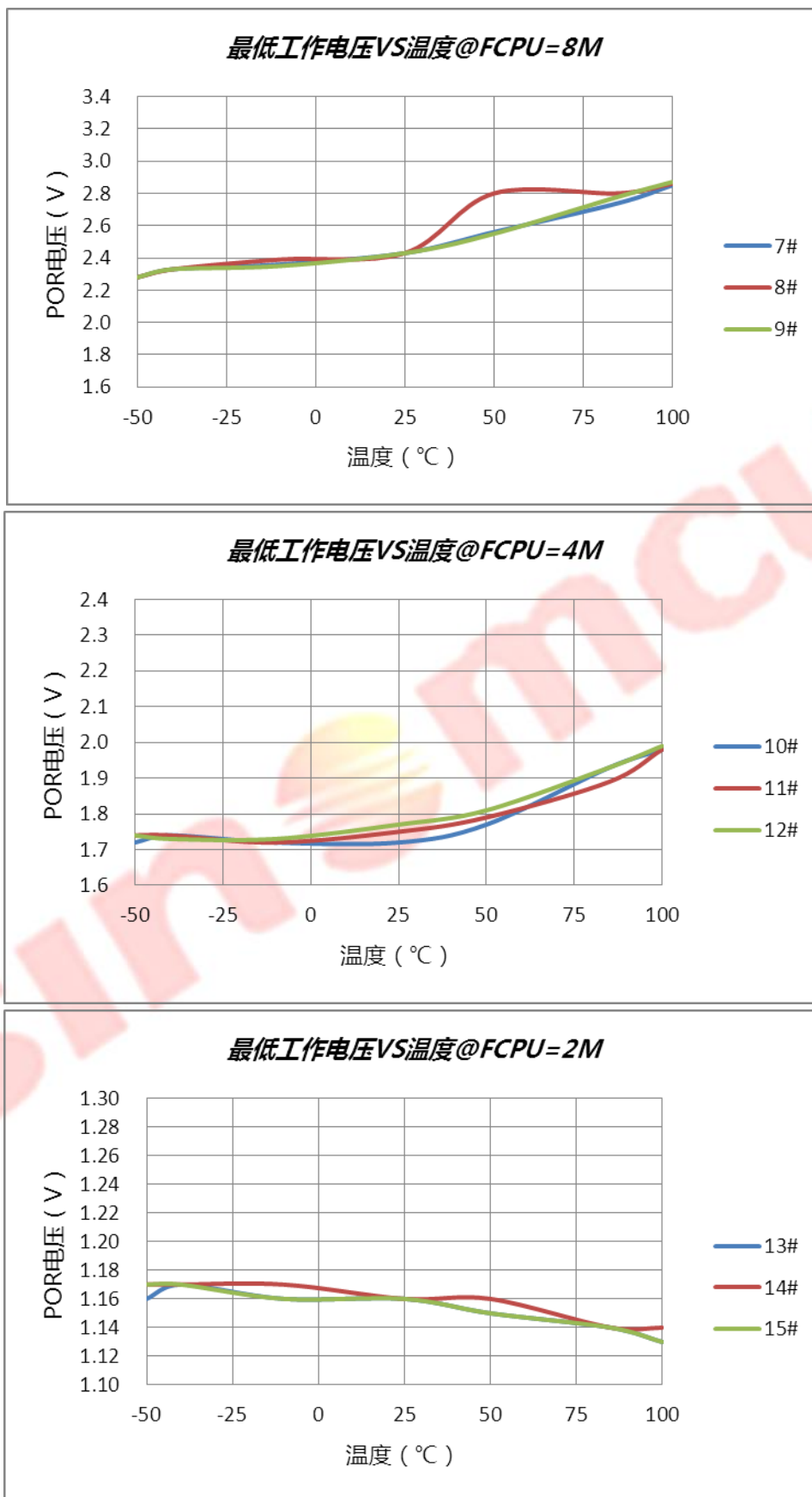
VREF电压VS温度@VREF=3V@工作状态

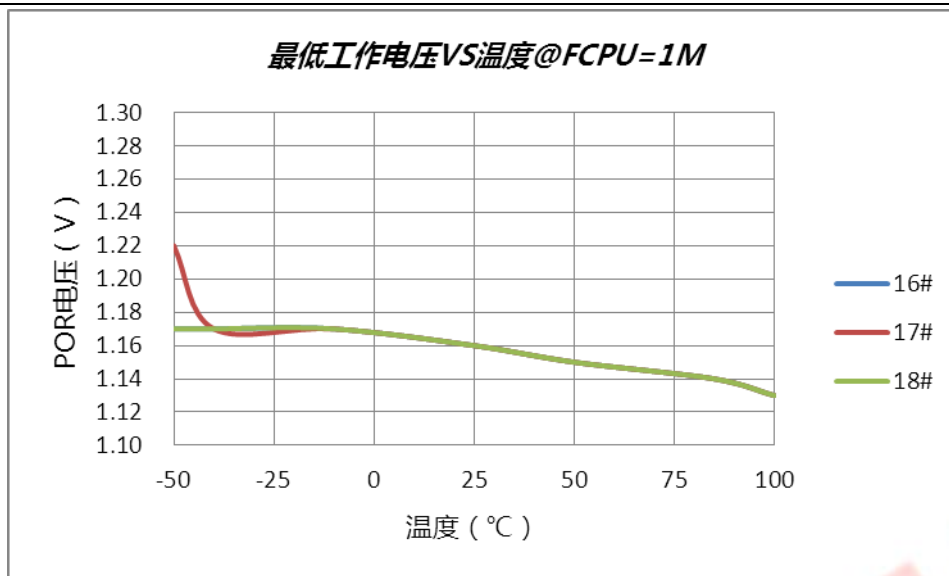


VREF电压VS温度@VREF=4V@工作状态



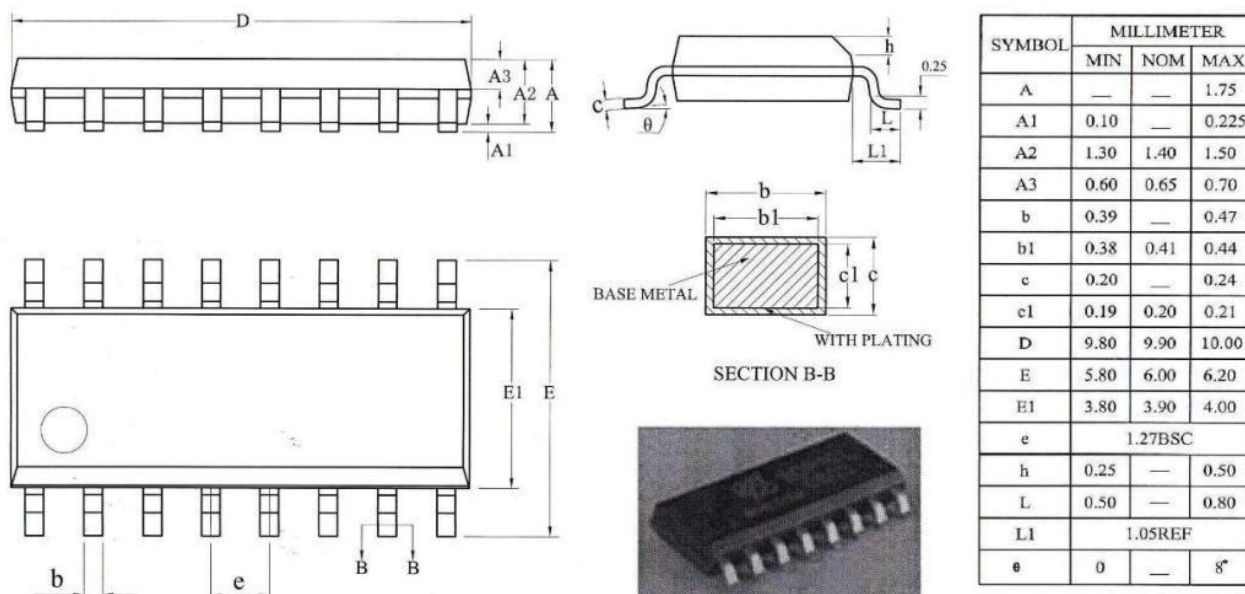
最低工作电压 VS 温度



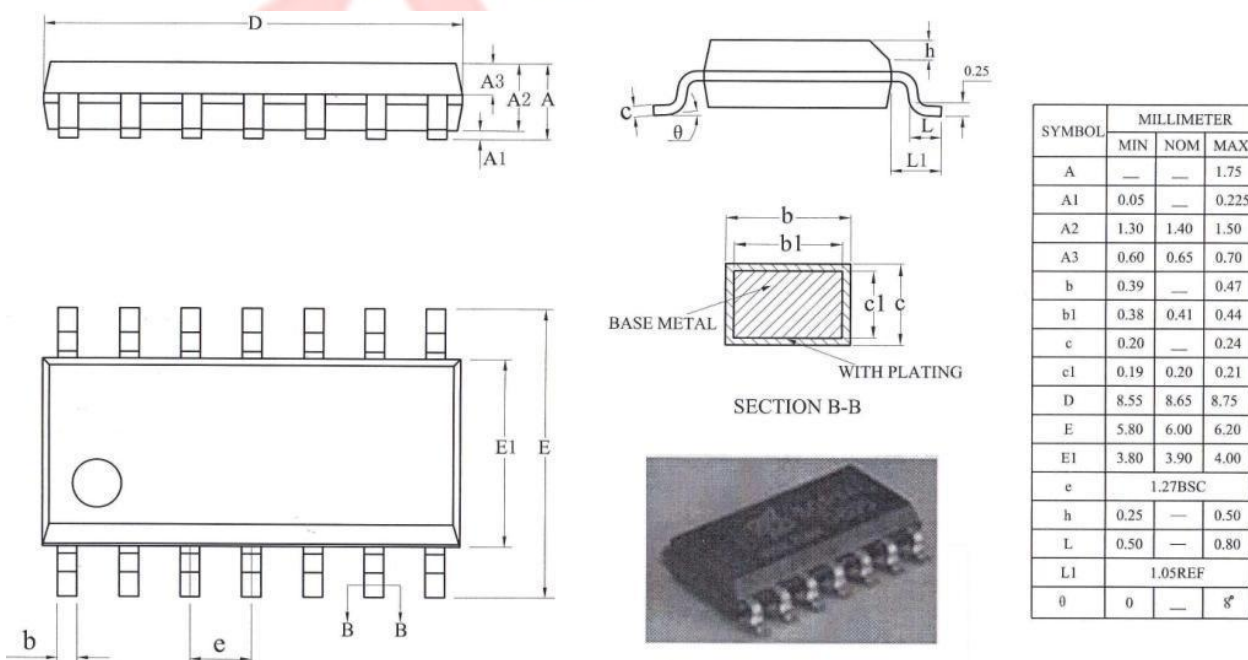


12 封装尺寸

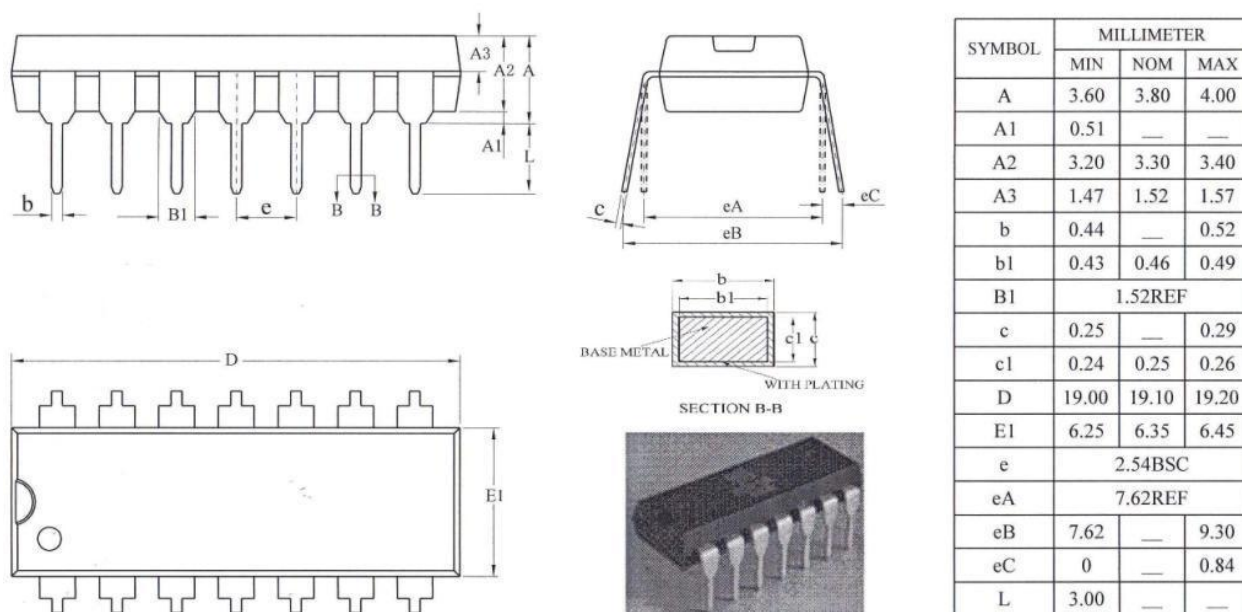
12.1 SOP16



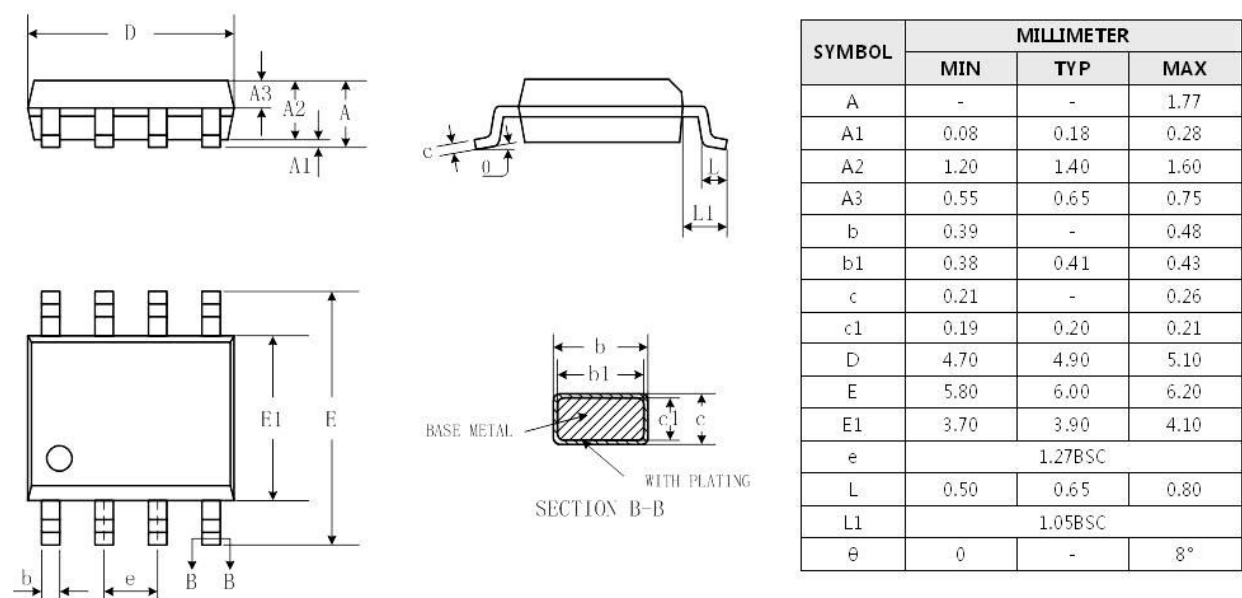
12.2 SOP14



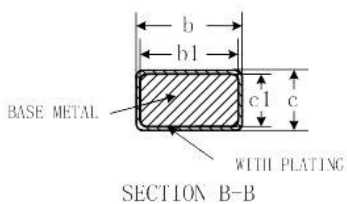
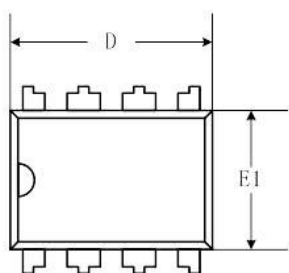
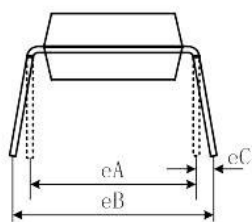
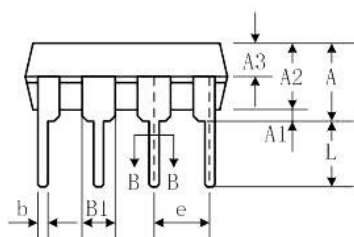
12.3 DIP14



12.4 SOP8

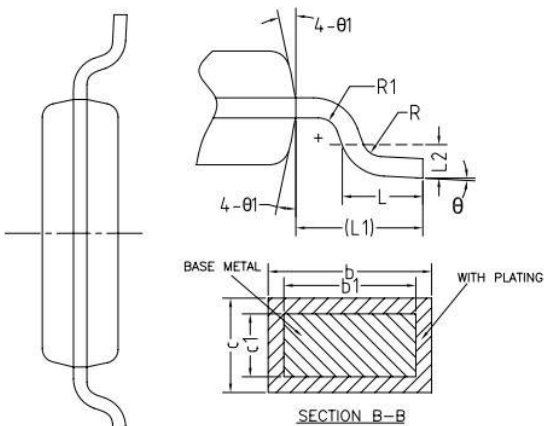
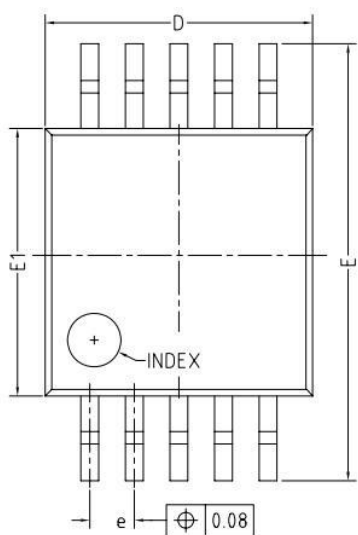


12.5 DIP8



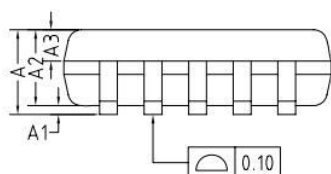
SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	3.60	3.80	4.00
A1	0.51	-	-
A2	3.10	3.30	3.50
A3	1.50	1.60	1.70
b	0.44	-	0.53
b1	0.43	0.46	0.48
B1	1.52BSC		
c	0.25	-	0.31
c1	0.24	0.25	0.26
D	9.05	9.25	9.45
E1	6.15	6.35	6.55
e	2.54BSC		
eA	7.62BSC		
eB	7.62	-	9.50
eC	0	-	0.94
L	3.00	-	-

12.6 MSOP10



COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	-	-	1.10
A1	0	-	0.15
A2	0.75	0.85	0.95
A3	0.25	0.35	0.39
b	0.18	-	0.27
b1	0.17	0.20	0.23
c	0.15	-	0.20
c1	0.14	0.15	0.16
D	2.90	3.00	3.10
E	4.70	4.90	5.10
E1	2.90	3.00	3.10
e	0.40	0.50	0.60
L	0.40	0.60	0.80
L1	0.95REF		
L2	0.25BSC		
R	0.07	-	-
R1	0.07	-	-
θ	0°	-	8°
θ 1	9°	12°	15°



13 修订记录

版本	修订日期	修订内容
V1.0	2019-03-11	初版发布；