

SinoMCU 8 位单片机

MC30P6201

用户手册

V1.0



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	CMP 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	13
3.4	堆栈	14
3.5	控制寄存器	14
3.6	用户配置字	17
4	系统时钟	19
4.1	内部高频 RC 振荡器	20
4.2	内部低频 RC 振荡器	20
4.3	外部晶体振荡器	20
4.4	外部 RC 振荡器	20
4.5	系统工作模式	21
4.6	低功耗模式	21
5	复位	22
5.1	复位条件	22
5.2	上电复位	23
5.3	外部复位	23
5.4	低电压复位	23
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	25
6.3	端口模式控制	26
7	定时器 TIMER	28
7.1	看门狗定时器 WDT	28
7.2	定时器 T0	28
7.3	定时器 T1	30
7.4	定时器 T2	34
8	比较器 CMP 和电压检测 EVD	39
8.1	CMP 概述	39

8.2	CMP 相关寄存器.....	40
9	中断.....	42
9.1	外部中断.....	42
9.2	定时器中断.....	42
9.3	键盘中断.....	42
9.4	CMP 中断.....	43
9.5	中断相关寄存器.....	43
10	特性曲线.....	46
10.1	I/O 特性.....	46
10.2	功耗特性.....	50
10.3	模拟电路特性.....	55
11	封装尺寸.....	58
11.1	SOP20.....	58
11.2	SSOP20.....	58
11.3	SOP18.....	59
11.4	DIP18.....	59
11.5	SOP16.....	60
11.6	DIP16.....	60
11.7	SOP14.....	61
11.8	DIP14.....	61
12	修订记录.....	62

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} 可配置为 2/4 分频
- 程序存储器
 - ◇ 2K×14 位 OTP 型程序存储器（烧录 1 次）
 - ◇ 1K×14 位 OTP 型程序存储器（烧录 2 次）
- 数据存储器
 - ◇ 96 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
- 3 组共 18 个 I/O
 - ◇ P0 (P00~P07), P1 (P10~P17), P2 (P20~P21)
 - ◇ P05 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
 - ◇ P06/P07 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
 - ◇ P1 可选开漏或推挽输出，P04~P07/P20~P21 可选输入/输出态或高阻态（复位初值）
 - ◇ P0 和 P1 所有端口均支持键盘中断唤醒功能，并可单独使能
- 系统时钟源
 - ◇ 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
 - ◇ 支持外接高频晶体振荡器（455KHz/4MHz~16MHz），可用作系统主时钟源
 - ◇ 支持外接 RC 振荡器（0~4MHz），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、或系统低频时钟源
 - ◇ 支持外接低频晶体振荡器（32768Hz），可用作系统主时钟源、或系统低频时钟源
- 系统工作模式
 - ◇ 运行模式：CPU 在系统主时钟下运行
 - ◇ 休眠模式：CPU 停止运行，系统主时钟源停止工作
- 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：开启 WDT、关闭 WDT，也可软件控制开启或关闭
- 3 个定时器
 - ◇ 8 位定时器 T0，支持系统低频时钟，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、3 路共周期独立占空比的 PWM（可组合成 1 对带死区的互补 PWM）
 - ◇ 11 位定时器 T2，可实现 2 路 11 位共周期独立占空比的 PWM
- 1 个模拟比较器 CMP
 - ◇ 输入共模 0 ~ (VDD-1.4V)，输出可选有/无回滞
 - ◇ 正端输入可选择外部输入电压、或内部参考电压 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

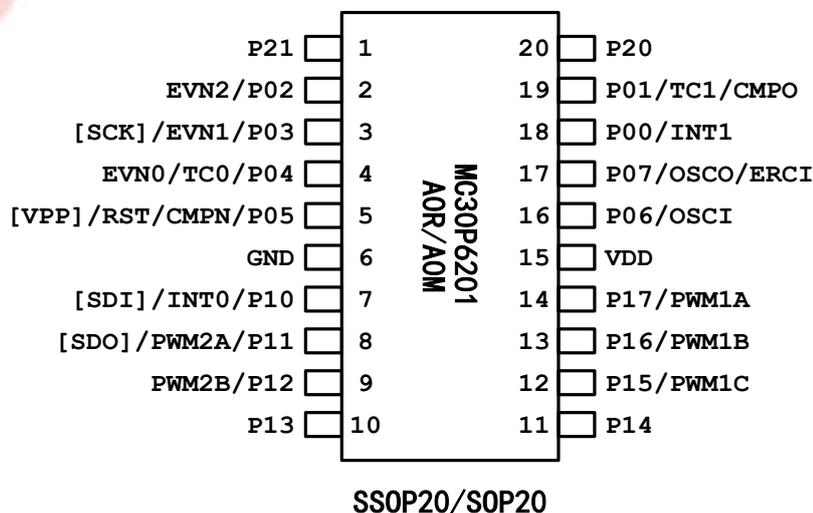
- ◇ 输出端电平可选择上升沿或下降沿触发中断，可从端口输出且支持输出取反
- ◇ 可实现 VDD 或外部输入的电压检测（EVD）功能
- 中断
 - ◇ 外部中断（INT0~INT1），键盘中断（P00~P07，P10~P17）
 - ◇ 定时器中断（T0~T2），CMP 中断
- 低电压复位 LVR：1.8V/2.0V/2.4V/2.7V/3.0V
- 工作电压（@HIRC）
 - ◇ VLVR27 ~ 5.5V @ Fcpu = 0~8MHz
 - ◇ VLVR20 ~ 5.5V @ Fcpu = 0~4MHz
 - ◇ VLVR18 ~ 5.5V @ Fcpu = 0~1MHz
- 封装形式：SSOP20/SOP20/SOP18/DIP18/SOP16/DIP16/SOP14/DIP14

1.2 订购信息

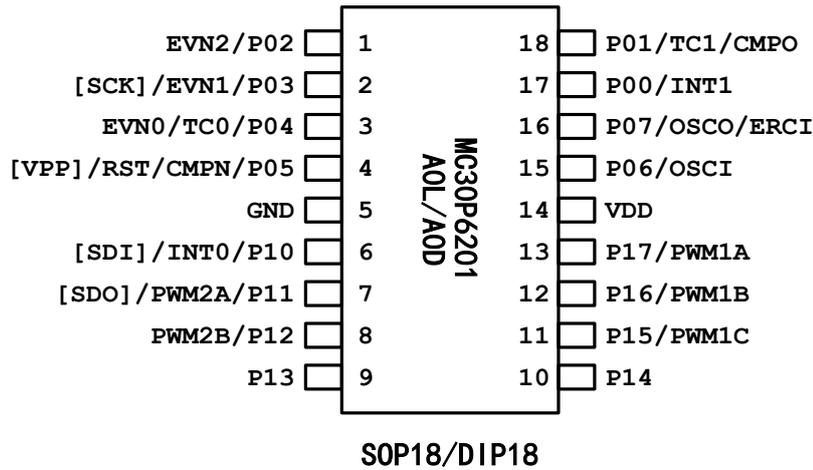
产品名称	封装形式	备注
MC30P6201A0R	SSOP20	
MC30P6201A0M	SOP20	
MC30P6201A0L	SOP18	
MC30P6201A0D	DIP18	
MC30P6201A0K	SOP16	
MC30P6201A0C	DIP16	
MC30P6201A0J	SOP14	
MC30P6201A0B	DIP14	

1.3 引脚排列

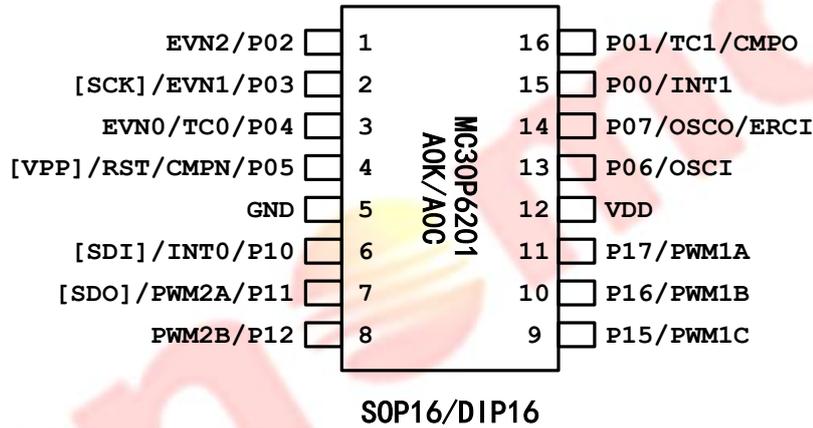
MC30P6201A0R/A0M



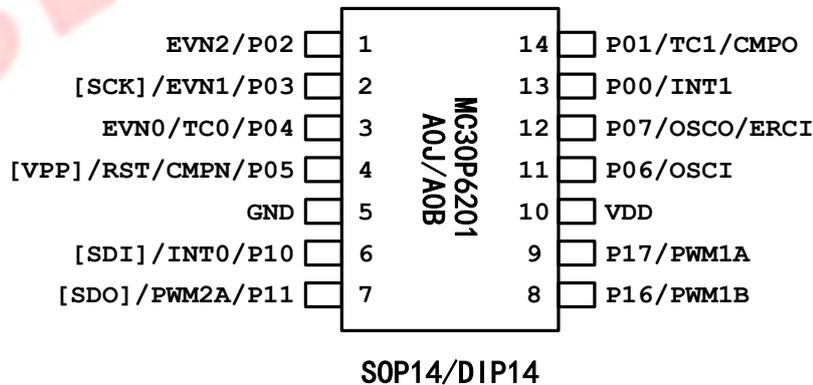
MC30P6201A0L/A0D



MC30P6201A0K/A0C



MC30P6201A0J/A0B



1.4 端口说明

端口名称	类型	功能说明
VDD	P	电源
GND	P	地
P0 (除 P05), P2	D	GPIO (推挽输出), 内部上/下拉
P05	D	GPIO (开漏输出), 内部上/下拉
P1	D	GPIO (可选推挽/开漏输出), 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
PWM1A~PWM1C	DO	定时器 T1 的 3 路 PWM 输出
PWM2A~PWM2B	DO	定时器 T2 的 2 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
OSCI, OSCO	A	外部时钟振荡器输入/输出
ERCI	AI	外部 RC 振荡器输入
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	IVDDmax	60	mA
流出 GND 最大电流	IGNDmax	60	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@HIRC(16M)/2	VLVR27		5.5	V
			Fcpu=4MHz@HIRC(8M)/2	VLVR20		5.5	
			Fcpu=2MHz@HIRC(4M)/2	VLVR20		5.5	
			Fcpu=1MHz@HIRC(2M)/2	VLVR20		5.5	
			Fcpu=500KHz@HIRC(1M)/2	VLVR20		5.5	
			Fcpu=8MHz@HEXT(16M)/2	VLVR30		5.5	
			Fcpu=4MHz@HEXT(8M)/2	VLVR24		5.5	
			Fcpu=2MHz@HEXT(4M)/2	VLVR20		5.5	
			Fcpu=227.5KHz@HEXT(455K)/2	VLVR20		5.5	
			Fcpu=16KHz@LIRC(32K)/2	VLVR18		5.5	
Fcpu=16384Hz@LEXT(32768)/2	VLVR18		5.5				
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	SMTVS 配置	0.8VDD			V
			SMTVS 配置	2.0			V
输入低电平	Vil	所有输入脚	SMTVS 配置			0.2VDD	V
			SMTVS 配置			0.8	V
输出拉电流	Ioh	推挽输出脚	Voh=VDD-0.6V		20		mA
输出灌电流	Iol	所有输出脚	Vol=0.6V		30		mA
上拉电阻	Rpu1	所有输入脚 (除 P05)	VDD=5V, Vin=0	-25%	16	+25%	KΩ
	Rpu2	P05	VDD=5V, Vin=0		50		KΩ

下拉电阻	Rpd1	所有输入脚 (除 P05)	Vin=VDD=5V	-25%	16	+25%	KΩ
	Rpd2	P05	Vin=VDD=5V		80		KΩ
运行模式功耗	Irun	VDD	Fcpu=8MHz@HIRC(16M)/2		2.9		mA
			Fcpu=4MHz@HIRC(8M)/2		1.6		mA
			Fcpu=2MHz@HIRC(4M)/2		0.9		mA
			Fcpu=1MHz@HIRC(2M)/2		580		uA
			Fcpu=500KHz@HIRC(1M)/2		420		uA
			Fcpu=8MHz@HEXT(16M)/2		3.6		mA
			Fcpu=4MHz@HEXT(8M)/2		2.0		mA
			Fcpu=2MHz@HEXT(4M)/2		1.3		mA
			Fcpu=227.5KHz@HEXT(455K)/2		350		uA
			Fcpu=16KHz@LIRC(32K)/2		8.5		uA
			Fcpu=16384Hz@LEXT(32768)/2		10		uA
休眠模式功耗	Istop	VDD	单时钟 HIRC 模式, WDT 关		0.2	1	uA
			双时钟 HIRC+LIRC 模式		3	6	uA
			双时钟 HIRC+LEXT 模式		6.5	13	uA
低压复位电压	VLVR	VDD	LVRVS 配置	-5%		+5%	V
LVR 回滞电压		VDD			6%	12%	

注：条件项中，未注明模块默认关闭，无关端口状态为输出无负载，输入或开漏输出高则端口电压为 VDD/GND。

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=5V, T=25°C	-2%	16	+2%	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
16M 晶振起振电压		T=25°C	3.0			V
8M 晶振起振电压		T=25°C	2.4			V
4M 晶振起振电压		T=25°C	2.0			V
455K 晶振起振电压		T=25°C	2.0			V
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
OSCI 内部对地电容	CG			10		pF
OSCO 内部对地电容	CD			10		pF
外部 RC 振荡频率	FERC1	VDD=5V, T=25°C, R=1KΩ, C=100pF	-50%	3.4	+50%	MHz
	FERC2	VDD=5V, T=25°C, R=3KΩ, C=100pF	-50%	1.3	+50%	MHz
	FERC3	VDD=5V, T=25°C, R=1KΩ, C=1000pF	-50%	590	+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
CMP 有效工作电压	V _{CMP}	T=-40°C~85°C	2.0		5.5	V
比较电路工作电流	I _{CMP}			10		uA
VDD 检测工作电流	I _{EVD}	VDD=5V, EVDIS=00		20	40	uA
输入失调电压	V _{offset}		-15		+15	mV
输入共模电压	V _{com}		0		VDD-1.4	V
输出回滞电压	V _{hys}	CMPPS=0(电压检测)		6%	12%	
		CMPPS=1(电压比较)	5	12	30	mV
响应时间	T _{RESP}	快速响应		2	20	us
		慢速响应		50	2000	
通道切换等待时间	T _{CCW}		200			us
内部参考电压	V _{IR}		-5%	0.5	+5%	V
电阻分压比值			-1%		+1%	

3 CPU 及存储器

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，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→ACC	1	1	C,DC,Z
ADDRA R	R 和 ACC 相加, 结果存入 R	R+ACC→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→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→R	1	1	C,DC,Z
RSUBAR R	R 和 ACC 相减, 结果存入 ACC	R-ACC→ACC	1	1	C,DC,Z
RSUBRA R	R 和 ACC 相减, 结果存入 R	R-ACC→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→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→R	1	1	C,DC,Z
ANDAR R	R 和 ACC 与操作, 结果存入 ACC	R and ACC→ACC	1	1	Z
ANDRA R	R 和 ACC 与操作, 结果存入 R	R and ACC→R	1	1	Z
ORAR R	R 和 ACC 或操作, 结果存入 ACC	R or ACC→ACC	1	1	Z
ORRA R	R 和 ACC 或操作, 结果存入 R	R or ACC→R	1	1	Z
XORAR R	R 和 ACC 异或操作, 结果存入 ACC	R xor ACC→ACC	1	1	Z
XORRA R	R 和 ACC 异或操作, 结果存入 R	R xor ACC→R	1	1	Z
COMAR R	对 R 取反, 结果存入 ACC	R 取反→ACC	1	1	Z
COMR R	对 R 取反, 结果存入 R	R 取反→R	1	1	Z
RLAR R	R 循环左移 (带 C 标志), 结果存入 ACC	R[7]→C R[6:0]→ACC[7:1] C→ACC[0]	1	1	C
RLR R	R 循环左移 (带 C 标志), 结果存入 R	R[7]→C R[6:0]→R[7:1] C→R[0]	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		清零 ACC	0→ACC	1	1	Z
CLRR	R	清零 R	0→R	1	1	Z
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
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
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	-
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
ISUBAI	K	K 和 ACC 相减, 结果存入 ACC	K-ACC→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[10:0]	2	1	-
GOTO	K	无条件跳转	K→PC[10:0]	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		清零看门狗定时器	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-算术逻辑单元累加器, 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，可通过配置字 PAGE 选择存储器的地址空间范围。

2K×14 位的地址空间范围为 0000H~07FFH，仅能烧录 1 次，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 - 07FFH)

1K×14 位的地址空间范围为 0000H~03FFH，可以烧录 2 次，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

数据存储器包括通用数据存储器 GPR 和特殊功能寄存器 SFR，部分地址映射了 2 组 SFR，通过寄存器位 RBS 选择，具体地址分配参照下表。GPR/SFR 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	RBS	类型	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F
00H-5FH	0/1	GPR	通用数据存储器区							
60H-67H	0/1	SFR	INDF	T0CNT	PCL	STATUS	FSR	P0	P1	P2
68H-6FH	0/1		MCR	T0CR	TMRCR	PCLATH	INTE1	INTF1	INTE0	INTF0
70H-77H	0		CMPCR0	CMPCR1	PUCON1	PDCONL	ODCON	DDR0	DDR1	DDR2
	1		PUCON0	PDCONH		PMOD	P0ADCR			
78H-7FH	0		T1DATA	T1DATB	T1DATC	PWM1CR	T2DATAH	T2DATAL	T2DATBH	T2DATBL
	1		T1CR	T1CNT	T1LOAD	T2CR	T2LOADH	T2LOADL	P0KBCR	P1KBCR

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。

数据存储寄存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
/	/	/	/	/	/	/	/	RBS	来自指令的 7 位地址							直接寻址模式
/	/	/	/	/	/	/	/	RBS	FSR							间接寻址模式

注：无论采用哪种寻址模式访问 70H~7FH 地址的 SFR，均需保证 RBS 位选择正确。

直接寻址模式，是以指令的低 7 位为数据存储寄存器地址，通过指令访问，寻址范围为 00H~7FH。例：
通过直接寻址模式将数据 55H 写入数据存储寄存器 10H 地址

```
MOVAI    55H
MOVRA    10H           ; 将数据 55H 写入数据存储寄存器 10H 地址
```

间接寻址模式，是以 FSR 为数据存储寄存器地址指针，通过 INDF 访问，寻址范围为 00H~7FH。例：
通过间接寻址模式将数据 55H 写入数据存储寄存器 10H 地址

```
MOVAI    10H
MOVRA    FSR
MOVAI    55H
MOVRA    INDF         ; 将数据 55H 写入 FSR 指向的数据存储器中
```

3.4 堆栈

5 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

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	1	1	1	1	1	1	1

BIT[6:0] FSR[6:0] – 数据指针寄存器

FSR：间接寻址模式的指针。

汇编模式、C 编译模式：FSR 用于间接寻址时仅低 7 位有效，读 FSR 时高 1 位固定为 1；

兼容模式：FSR 用于间接寻址时仅低 6 位有效，读 FSR 时高 2 位固定为 1。

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							
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 寻址实际是对 FSR 指向的数据存储器地址进行访问, 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							
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寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCLATH	-	-	-	-	-	PCH2	PCH1	PCH0
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2:0] **PCH[2:0]** – 程序指针计数器高 3 位寄存器 (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: $PC = PC + 1$;
- ◇ 分支指令 GOTO/CALL: $PC =$ 指令码低 11 位;
- ◇ 返回指令 RETIE/RETURN/RETAI: $PC =$ 堆栈栈顶 (TOS);

对 PCL 操作指令 (汇编模式、兼容模式):

- ◇ 对 PCL 操作的加法指令: $PC = (PC[10:0] + ALU[7:0])$;
- ◇ 对 PCL 操作的其它指令: $PC = \{PC[10:8]:ALU[7:0](ALU \text{ 运算结果})\}$;

对 PCL 操作指令 (C 编译模式):

- ◇ 对 PCL 操作指令: $PC = \{PCLATH[2:0]:ALU[7:0](ALU \text{ 运算结果})\}$;

CPU 状态寄存器

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

- BIT[7] WKUP** – 唤醒源标志位
0: 芯片通过其他唤醒源唤醒;
1: 芯片通过外部中断或键盘中断唤醒;
- BIT[5] RBS** – 数据寄存器地址映射选择位 (兼容模式下固定为 0)
0: 地址 70H~7FH 映射至第 0 组寄存器;
1: 地址 70H~7FH 映射至第 1 组寄存器;
- 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/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	WDTEN	EIS	-	-	-	-	-	-
R/W	R/W	R/W	-	-	-	-	-	-
初始值	1	0	-	-	-	-	-	-

- BIT[7] WDTEN** – 看门狗使能位
0: 关闭 WDT;
1: 开启 WDT;
- BIT[6] EIS** – INTO 中断外部输入使能位
0: INTO 中断外部输入无效, 端口用作其他功能;
1: INTO 中断外部输入有效, 端口需设为输入;

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。

芯片的用户配置字，定义如下：

符号	功能说明
PAGE	ROM 烧录模式设置： 1K 容量 MTP 模式，第 1 次烧录； 1K 容量 MTP 模式，第 2 次烧录； 2K 容量 OTP 模式；
OSCM	系统时钟振荡模式设置： 单时钟模式：内部高频 RC 振荡器 HIRC； 单时钟模式：内部低频 RC 振荡器 LIRC； 单时钟模式：外部 16M 晶体振荡器；（支持外部 10MHz~16MHz 晶振） 单时钟模式：外部 8M 晶体振荡器；（支持外部 6MHz~10MHz 晶振） 单时钟模式：外部 4M 晶体振荡器；（支持外部 4MHz~6MHz 晶振） 单时钟模式：外部 455K 晶体振荡器；（支持外部 455KHz 晶振） 单时钟模式：外部 32K 晶体振荡器；（支持外部 32768Hz 晶振） 单时钟模式：外部 RC 振荡器 ERC；（支持外部 0~4MHz RC 振荡器） 双时钟模式：内部 HIRC + 内部 LIRC； 双时钟模式：内部 HIRC + 外部 32K 晶体振荡器；
HIRCDS	HIRC 后分频输出选择： $F_{OSC}=F_{HIRC}/1$ ； $F_{HIRC}/2$ ； $F_{HIRC}/4$ ； $F_{HIRC}/8$ ； $F_{HIRC}/16$ ； $F_{HIRC}/32$ ；
FCPU	CPU 时钟频率选择： $F_{CPU}=F_{OSC}/2$ ； $F_{CPU}=F_{OSC}/4$ ；
RSTEN	RST 外部复位设置： P05 为外部复位脚；P05 为输入/输出脚；
LVRVS	LVR 复位电压选择：（LVR 电压应满足由 F_{CPU} 决定的工作电压特性） 1.8V；2.0V；2.4V；2.7V；3.0V；
WDTM	WDT 模式设置： WDT 始终关闭；WDT 始终开启；
WDTT	上电延时及 WDT 溢出时间（无预分频）选择：（时间均为典型值） 上电延时=2.2ms，WDT 溢出时间=4.5ms； 上电延时=9ms，WDT 溢出时间=18ms； 上电延时=36ms，WDT 溢出时间=72ms； 上电延时=144ms，WDT 溢出时间=288ms；
SMTVS	端口施密特阈值选择： 2.0V/0.8V；0.8VDD/0.2VDD；
CPRSPT	比较器响应时间设置： 快速响应（2us）；慢速响应（50us）；
MCUSEL	芯片模式选择： 汇编模式（不支持 C 程序中常量数组的应用）；

4 系统时钟

芯片为双时钟系统（可配置为单时钟模式），内部电路均在系统主时钟 Fosc 或系统低频时钟 FLOsc 下工作，部分模块的时钟还可在 Fosc 和 FLOsc 之间切换。

系统主时钟 Fosc 可通过配置字 OSCM 选择以下时钟：

- ◇ 内部高频 RC 振荡器 HIRC（16MHz）时钟 FHIRC 的 1/2/4/8/16/32 分频；
- ◇ 内部低频 RC 振荡器 LIRC（32KHz）时钟 FLIRC；
- ◇ 外部高频晶体振荡器 HEXT（455KHz/4MHz~16MHz）时钟 FHEXT；
- ◇ 外部低频晶体振荡器 LEXT（32768Hz）时钟 FLEXT；
- ◇ 外部 RC 振荡器 ERC（0~4MHz）时钟 FERC；

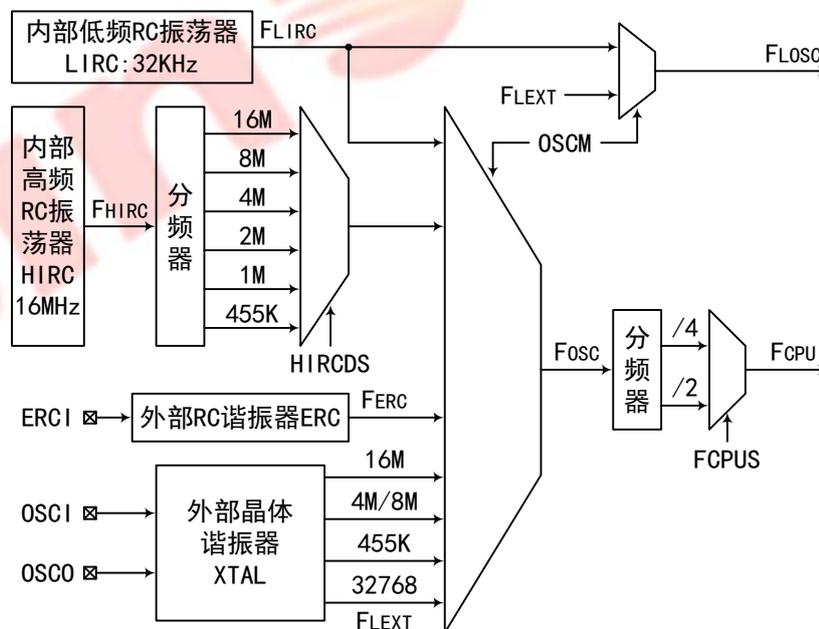
系统低频时钟 FLOsc 可通过配置字 OSCM 在配置为双时钟模式时选择以下时钟：

- ◇ 内部低频 RC 振荡器 LIRC（32KHz）时钟 FLIRC；
- ◇ 外部低频晶体振荡器 LEXT（32768Hz）时钟 FLEXT；

CPU 为单时钟，时钟源固定为系统主时钟 Fosc，CPU 的时钟频率 Fcpu 通过配置字 FCPUS 选择。

WDT（看门狗）电路的时钟源固定为内部低频 RC 振荡器。

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，其后分频输出时钟（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。

注：系统时钟配置为 HIRC+LIRC 双时钟、或系统时钟配置为 LIRC 单时钟后系统处于运行模式、或 WDT 开启，此时 LIRC 才振荡工作。

4.3 外部晶体振荡器

芯片支持外接高频 455KHz/4MHz~16MHz、或低频 32768Hz 的晶体振荡器作为系统主时钟源。还可在系统时钟配置为双时钟模式时，选择外接 32768Hz 晶体振荡器作为系统低频时钟源。

外接晶振的实际应用中，晶振两端的对地电容 CG/CD 是必需的（芯片已在端口内置典型值为 10pF 的对地电容）。用户应使晶振离 OSCI/OSCO 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 CG/CD 的推荐值和相应最低起振电压参考值。

晶振频率 (Hz)	电容 CG/CD (pF)	最低起振电压 (V)
16M	0/10	3.0
8M	10/20	2.4
4M	10/20	2.0
455K	100/220	2.0
32768	0/10	1.8

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

4.4 外部 RC 振荡器

芯片支持外接 RC 振荡器作为系统主时钟源。外部 RC 振荡电路只需要和 ERCI 引脚连接，电容值不能低于 100pF，电阻值和电容值共同决定 RC 的振荡频率，最高支持 4MHz。

4.5 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	切入条件	系统状态
运行模式	系统复位	CPU 运行，主时钟源工作
	休眠模式下，CPU 唤醒	
休眠模式	运行模式下，执行 STOP 指令	CPU 暂停，主时钟源停止

注：系统低频时钟源不受工作模式影响，若系统时钟配置为双时钟模式则选定的低频时钟源将一直工作。

4.6 低功耗模式

芯片的低功耗模式即为休眠模式。

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）；
- ◇ 比较器 CMP 中断发生（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

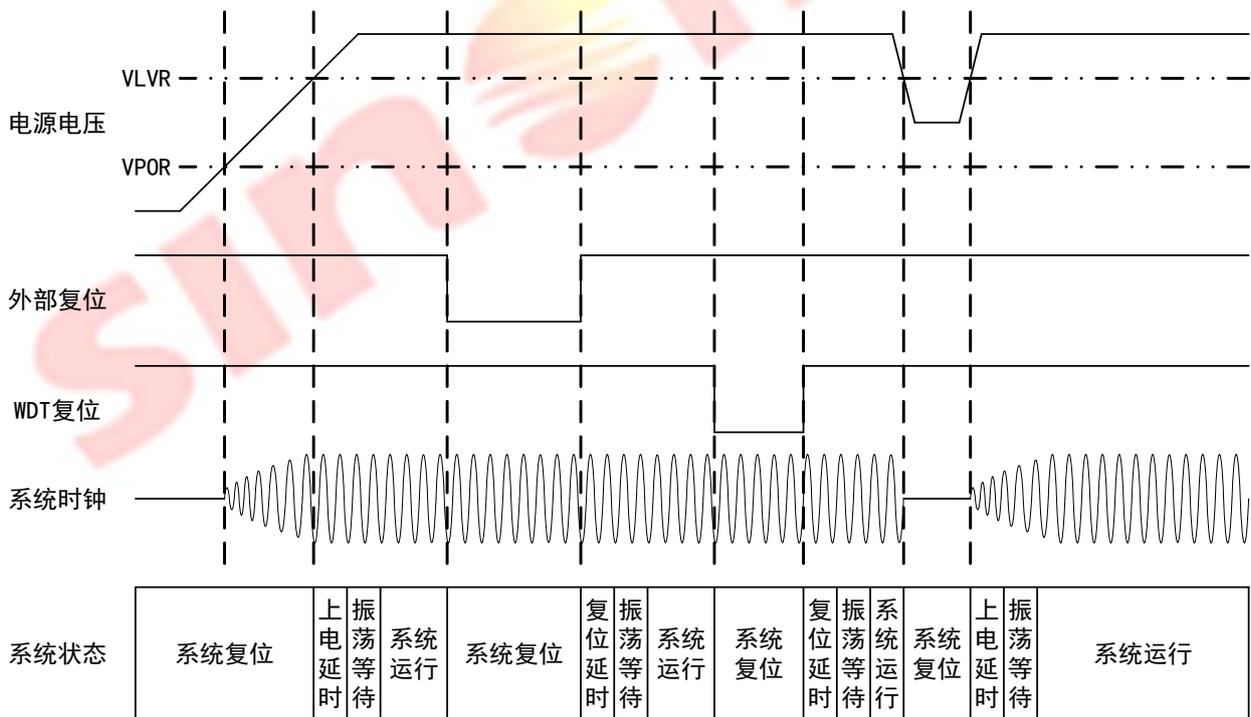
芯片共有如下几种复位方式：

- ◇ 上电复位 POR；
- ◇ 外部复位；
- ◇ 低电压复位 LVR；
- ◇ 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 开始重新从程序存储器 0000H 地址处执行指令。

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时序关系的示意图：



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，则应通过配置字选择较长的上电延时时间，或在复位后进行软件延时，以确保芯片开始工作时 VDD 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快速、慢速上电的情况，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

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开启外部复位功能，端口的内部上拉电阻自动使能。外部复位输入端口 **RST** 为施密特结构，低电平有效。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可通过配置字 **LVRVS** 选择。电压检测电路有一定的回滞特性，回滞电压为 6% 左右（典型值），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 +6% 后 LVR 复位才解除。

LVR 在休眠模式时将自动关闭，在进入运行模式后自动开启。

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。若出现异常情况，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/输出端口包括两组 8 位端口 P0、P1，和一组 2 位端口 P2。所有 I/O 端口均支持施密特输入，除 P05 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 CMP 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
P0	P07D	P06D	P05D	P04D	P03D	P02D	P01D	P00D
R/W	R/W							
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **P0nD** – P0n 端口数据位 (n=7-0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1	P17D	P16D	P15D	P14D	P13D	P12D	P11D	P10D
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)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P2	-	-	-	-	-	-	P21D	P20D
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	X	X

BIT[1:0] **P2nD** – P2n 端口数据位 (n=1-0)

端口方向寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DDR0	DDR07	DDR06	DDR05	DDR04	DDR03	DDR02	DDR01	DDR00
R/W	R/W							
初始值	1	1	1	1	1	1	1	1

BIT[7:0] **DDR0n** – P0n 端口方向控制位 (n=7-0)

0: 端口作为输出口，读端口操作将读取端口的数据寄存器值；

1: 端口作为输入口，读端口操作将读取端口的输入电平状态；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DDR1	DDR17	DDR16	DDR15	DDR14	DDR13	DDR12	DDR11	DDR10
R/W	R/W							
初始值	1	1	1	1	1	1	1	1

- BIT[7:0] **DDR1n – P1n 端口方向控制位 (n=7-0)**
 0: 端口作为输出口, 读端口操作将读取端口的数据寄存器值;
 1: 端口作为输入口, 读端口操作将读取端口的输入电平状态;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
DDR2	-	-	-	-	-	-	DDR21	DDR20
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	1	1

- BIT[1:0] **DDR2n – P2n 端口方向控制位 (n=1-0)**
 0: 端口作为输出口, 读端口操作将读取端口的数据寄存器值;
 1: 端口作为输入口, 读端口操作将读取端口的输入电平状态;

6.2 内部上/下拉电阻

所有端口均有内部上拉和下拉电阻, 且均有单独的寄存器位控制其上/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
PUCON0	P07PU	P06PU	P05PU	P04PU	P03PU	P02PU	P01PU	P00PU
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] **P0nPU – P0n 端口上拉电阻控制位 (n=7-0)**
 0: 端口内部上拉电阻有效;
 1: 端口内部上拉电阻无效;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PUCON1	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	1	1	1	1	1	1	1	1

- 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
PDCONL	P13PD	P12PD	P11PD	P10PD	P03PD	P02PD	P01PD	P00PD
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[7:4] **P1nPD – P1n 端口下拉电阻控制位 (n=3-0)**

- 0: 端口内部下拉电阻有效;
- 1: 端口内部下拉电阻无效;

BIT[3:0] **P0nPD** – P0n 端口下拉电阻控制位 (n=3-0)

- 0: 端口内部下拉电阻有效;
- 1: 端口内部下拉电阻无效;

注: 兼容模式下 P03 下拉电阻及控制位无效。

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDCONH	P17PD	P16PD	P15PD	P14PD	P07PD	P06PD	P05PD	P04PD
R/W	R/W							
初始值	1	1	1	1	1	1	1	1

BIT[7:4] **P1nPD** – P1n 端口下拉电阻控制位 (n=7-4)

- 0: 端口内部下拉电阻有效;
- 1: 端口内部下拉电阻无效;

BIT[3:0] **P0nPD** – P0n 端口下拉电阻控制位 (n=7-4)

- 0: 端口内部下拉电阻有效;
- 1: 端口内部下拉电阻无效;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMOD	P0HST	P2ST	-	-	P21PU	P20PU	P21PD	P20PD
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
初始值	0	0	-	-	1	1	1	1

BIT[7] **P0HST** – P04~P07 端口状态选择位

BIT[6] **P2ST** – P20/P21 端口状态选择位

BIT[3:2] **P2nPU** – P2n 端口上拉电阻控制位 (n=1-0)

- 0: 端口内部上拉电阻有效;
- 1: 端口内部上拉电阻无效;

BIT[1:0] **P2nPD** – P2n 端口下拉电阻控制位 (n=1-0)

- 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

用作数字输出口时, P1 可选择推挽输出或开漏输出。用作数字口时, P04~P07/P20/P21 可选择输入/输出态或高阻态 (复位初值), 高阻态时端口内部上/下拉电阻及其控制位无效。

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	P17OD	P16OD	P15OD	P14OD	P13OD	P12OD	P11OD	P10OD
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

BIT[7:0] **P1nOD** – P1n 端口输出模式选择位 (n=7-0)

0: 输出时端口为推挽输出;

1: 输出时端口为开漏输出;

注: 开漏输出口输出高电平时, 若端口电压低于 VDD 电压则会产生漏电流。

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMOD	P0HST	P2ST	-	-	P21PU	P20PU	P21PD	P20PD
R/W	R/W	R/W	-	-	R/W	R/W	R/W	R/W
初始值	0	0	-	-	1	1	1	1

BIT[7] **P0HST** – P04~P07 端口状态选择位

0: 端口为高阻态;

1: 端口为输入/输出态;

BIT[6] **P2ST** – P20/P21 端口状态选择位

0: 端口为高阻态;

1: 端口为输入/输出态;

BIT[3:2] **P2nPU** – P2n 端口上拉电阻控制位 (n=1-0)

BIT[1:0] **P2nPD** – P2n 端口下拉电阻控制位 (n=1-0)

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
P0ADCR	-	-	P05DC	P04DC	P03DC	P02DC	-	-
R/W	-	-	R/W	R/W	R/W	R/W	-	-
初始值	-	-	0	0	0	0	-	-

BIT[5:2] **P0nDC** – P0n 端口数字功能控制位 (n=5-2)

0: 使能端口的数字 I/O 功能;

1: 关闭端口的数字 I/O 功能;

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，可通过预分频器选择不同的计数时钟频率。WDT 计数器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。当 WDTM 选择始终关闭或 WDTEN 为 0 时 WDT 定时器被关闭；当 WDTM 选择始终开启且 WDTEN 为 1 时 WDT 定时器才开启。若 WDT 定时器开启，则在休眠模式下 WDT 依然工作，溢出时将唤醒 CPU；而在 CPU 运行时 WDT 溢出，则将复位芯片。

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）。

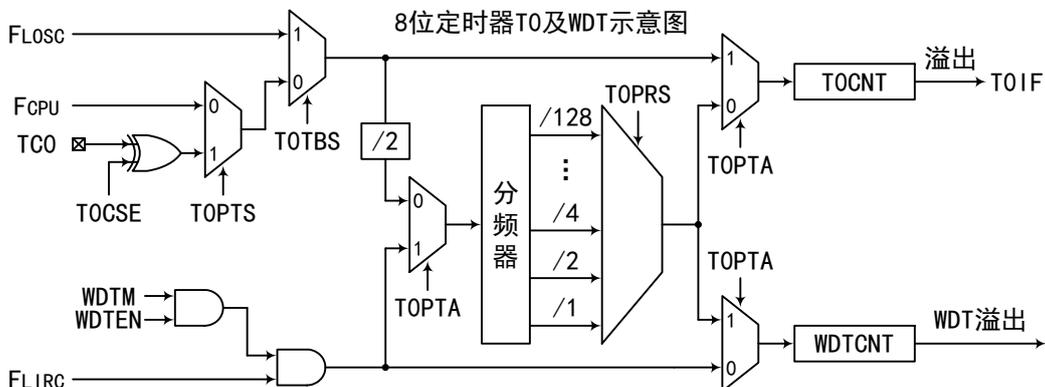
WDT 的基本溢出时间（即无预分频的时间）可配置为 4.5ms/18ms/72ms/288ms。

注：WDT 溢出时间为典型值，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 位递增计数器、可编程预分频器、控制寄存器。

- ◇ 可通过预分频比设置计数频率；
- ◇ 时钟源可选：FLOSC、FCPU、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 计数周期 = 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动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。

当系统时钟通过配置字 OSCM 配置为双时钟模式时，通过 TOTBS 选择 FLOSC 作为 T0 时钟，则在低功耗模式下 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
TOCR	-	TOTBS	TOPTS	TOCSE	TOPTA	TOPRS2	TOPRS1	TOPRS0
R/W	-	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	0	1	1	1	1	1	1

BIT[6] **TOTBS** – T0 时钟源选择位
 0: T0 时钟源由 TOPTS 决定;
 1: T0 时钟源为系统低频时钟 FLOSC (兼容模式下无效);

BIT[5] **TOPTS** – T0 时钟源选择位
 0: T0 时钟源为 FCPU;
 1: T0 时钟源为 TC0 输入的外部时钟 (兼容模式下无效);

BIT[4] **TOCSE** – T0 外部时钟计数沿选择位
 0: 外部时钟上升沿计数;
 1: 外部时钟下降沿计数;

BIT[3] **TOPTA** – 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] **TOPRS[2:0]** – 预分频比选择位

TOPRS[2:0]	T0 时钟预分频比 (TOPTA=0)	WDT 时钟预分频比 (TOPTA=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

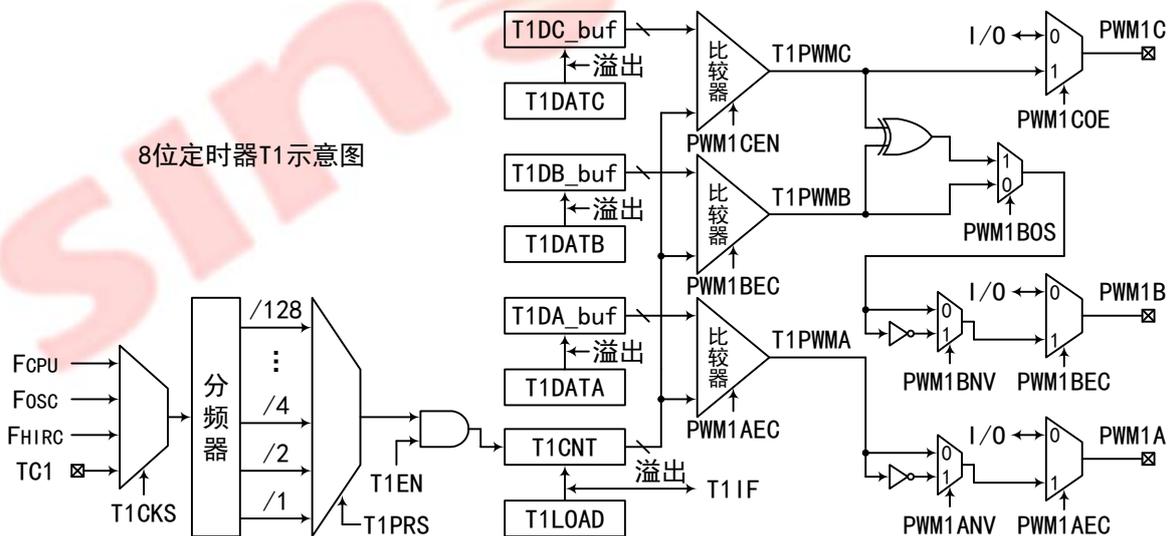
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TOCNT	TOCNT7	TOCNT6	TOCNT5	TOCNT4	TOCNT3	TOCNT2	TOCNT1	TOCNT0
R/W	R/W							
初始值	X	X	X	X	X	X	X	X

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

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 3 个 8 位比较寄存器。

- ◇ 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；
- ◇ 支持共周期 3 路 8 位 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 1 对带死区的互补 PWM；
- ◇ 支持溢出中断和溢出唤醒功能；



定时器 T1，可通过寄存器位 T1CKS 选择时钟源，通过 T1PRS 选择预分频比，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）。预分频比可选择 1~128 分频，对 T1CNT 的写操作将清零预分频计数器，而预分频比保持不变。

当 T1EN=0 时，T1CNT 保持不变，写重载寄存器 T1LOAD 将立即载入 T1CNT 中；当 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 中重新开始计数。

如图所示，定时器 T1 可实现共周期的 3 路 PWM 功能（PWM1x, x=A,B,C, 下同），且可分别设置每路 PWM 的占空比。可通过寄存器位使能/关闭 PWM 功能，或控制端口是否输出 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。

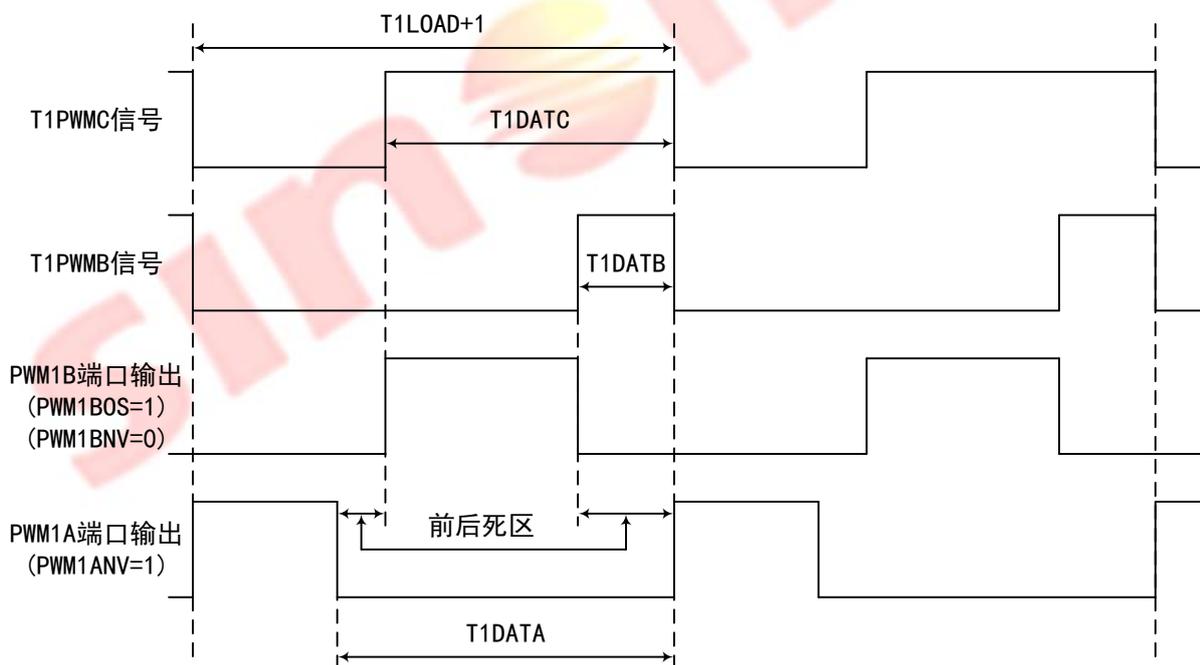
T1DATx 均配有 1 个 8 位的比较缓冲器 (T1Dx_buf) 用于与 T1CNT 比较，PWM1x 关闭时写 T1DATx 将立即载入比较缓冲器中，而 PWM1x 使能后写 T1DATx 则将在 T1 溢出时才载入比较缓冲器中。若要首个 PWM 周期和占空比准确，需先写重载寄存器和比较寄存器，再使能 PWM，最后开启定时器。

T1PWMx 信号 (x=A,B,C, 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 的溢出周期) = (T1LOAD+1) × T1CNT 计数时钟周期
- ◇ 占空比 = 高电平时间 / 周期 = (T1DATx) / (T1LOAD+1)

如图所示，端口 PWM1B 可通过 PWM1BOS 选择输出 T1PWMB 和 T1PWMC 逻辑异或后的 PWM 波形，从而与端口 PWM1A 输出的 PWM 波形组成 1 对带死区的互补 PWM。

PWM1 互补波形示意



定时器控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMCR	T2EN	PWM2AEC	PWM2BEC	T1EN	PWM1AEC	PWM1BEC	PWM1CEN	PWM1COE
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** – 定时器 T2 使能位
- BIT[6] **PWM2AEC** – PWM2A 使能位及端口输出控制位
- BIT[5] **PWM2BEC** – PWM2B 使能位及端口输出控制位
- BIT[4] **T1EN** – 定时器 T1 使能位
0: 关闭定时器 T1;
1: 开启定时器 T1;
- BIT[3] **PWM1AEC** – PWM1A 使能位及端口输出控制位
0: 关闭 PWM1A 功能, 并禁止端口输出 PWM 波形;
1: 使能 PWM1A 功能, 并允许端口输出 PWM 波形;
- BIT[2] **PWM1BEC** – PWM1B 使能位及端口输出控制位
0: 关闭 PWM1B 功能, 并禁止端口输出 PWM 波形;
1: 使能 PWM1B 功能, 并允许端口输出 PWM 波形;
- BIT[1] **PWM1CEN** – PWM1C 使能位
0: 关闭 PWM1C 功能;
1: 使能 PWM1C 功能;
- BIT[0] **PWM1COE** – PWM1C 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;

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	PWM1ANV	PWM1BNV	PWM1BOS	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] **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[6] **PWM1BNV** – PWM1B 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[5] **PWM1BOS** – PWM1B 端口输出信号选择位
0: 输出 T1PWMB 信号;
1: 输出 T1PWMB 和 T1PWMC 的逻辑异或信号;
- BIT[4:3] **T1CKS[1:0]** – T1 时钟源选择位

T1CKS[1:0]	T1 时钟源
00	FCPU
01	FOSC
10	FHIRC
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

PWM1 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM1CR	PWM1AMD	PWM1BMD	PWM1CMD	-	PWM1ACE	PWM1BCE	PWM1CCE	-
R/W	R/W	R/W	R/W	-	R/W	R/W	R/W	-
初始值	0	0	0	-	1	1	1	-

BIT[7:5] **PWM1xMD** – PWM1x 模式控制位 (x=A,B,C)

- 0: T1 计数溢出时 T1PWMx 信号变为低电平;
- 1: T1 计数溢出时 T1PWMx 信号变为高电平;

BIT[3:1] **PWM1xCE** – PWM1x 比较使能位 (x=A,B,C)

- 0: 下一个计数周期, T1CNT 计数到与 T1DATx 相等时, T1PWMx 信号保持不变;
- 1: 下一个计数周期, T1CNT 计数到与 T1DATx 相等时, T1PWMx 信号变为高电平;

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							
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							

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

7.4 定时器 T2

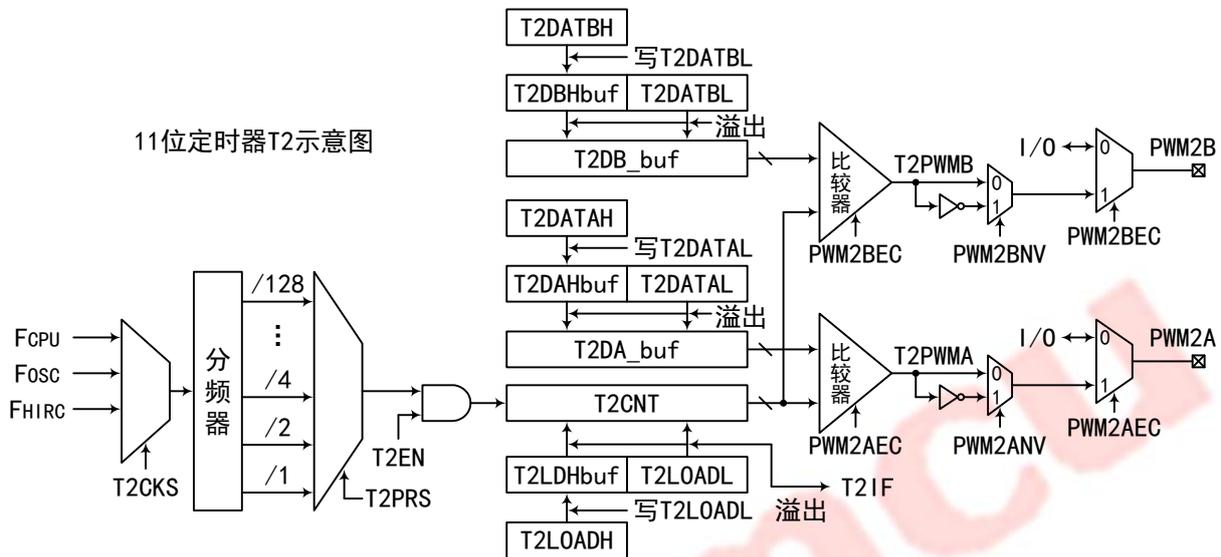
定时器 T2 为 11 位定时器，包含 1 个 11 位递减计数器、可编程预分频器、控制寄存器、11 位重载寄存器及 2 个 11 位比较寄存器。

- ◇ 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；
- ◇ 支持共周期 2 路 11 位 PWM 输出，可通过对应的比较寄存器分别设置每路 PWM 占空比；
- ◇ 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）。预分频比可选择 1~128 分频，对 T2CNT 的写操作将清零预分频计数器，而预分频比保持不变。11 位 T2CNT 的高字节 T2CNTH 配有读/写缓冲器，写 T2CNT 时需先写 T2CNTH 再写 T2CNTL，而读 T2CNT 时则需先读 T2CNTL 再读 T2CNTH。

当 T2EN=0 时，T2CNT 保持不变，写重载寄存器 T2LOAD 将立即载入 T2CNT 中；当 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 中重新开始计数。

11 位 T2LOAD 配有 1 个 3 位的高位缓冲器 (T2LDHbuf), 写 T2LOADL 时会同时将 T2LOADH 值载入该缓冲器中。此时若 T2EN=0, 则会同时再将[缓冲器:T2LOADL]载入 T2CNT; 若 T2EN=1, 则需在 T2 溢出后才将[缓冲器:T2LOADL]的值载入 T2CNT。调整 T2LOAD 值时需先写 T2LOADH, 再写 T2LOADL。



如图所示, 定时器 T2 可实现共周期的 2 路 PWM 功能 (PWM2x, x=A,B, 下同), 且可分别设置每路 PWM 的占空比。可通过寄存器位使能/关闭 PWM 功能, 或控制端口是否输出 PWM 波形。PWM2x 关闭时 T2PWMx 信号为低电平。PWM2x 使能后, T2CNT 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 T2DATx 相等时, T2PWMx 变为高电平; 当计数溢出时, T2PWMx 变为低电平。

11 位 T2DATx 均配有 1 个 3 位的高位缓冲器 (T2DxHbuf) 和 1 个 11 位的比较缓冲器 (T2Dx_buf)。写 T2DATxL 时会同时将 T2DATxH 值载入高位缓冲器中, 此时若 PWM2x 关闭, 则会同时再将[高位缓冲器:T2DATxL]载入比较缓冲器中, 若 PWM2x 已使能, 则将在 T2 溢出后才将[高位缓冲器:T2DATxL]载入比较缓冲器中。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。

T2PWMx 信号 (x=A,B, 下同) 的占空比计算如下:

- ✧ 高电平时间 = (T2DATx) × T2CNT 计数时钟周期
- ✧ 周期 (T2 的溢出周期) = (T2LOAD+1) × T2CNT 计数时钟周期
- ✧ 占空比 = 高电平时间 / 周期 = (T2DATx) / (T2LOAD+1)

定时器控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TMCR	T2EN	PWM2AEC	PWM2BEC	T1EN	PWM1AEC	PWM1BEC	PWM1CEN	PWM1COE
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 – 定时器 T2 使能位

- 0: 关闭定时器 T2;
- 1: 开启定时器 T2;

- BIT[6] **PWM2AEC** – PWM2A 使能位及端口输出控制位
 0: 关闭 PWM2A 功能, 并禁止端口输出 PWM 波形;
 1: 使能 PWM2A 功能, 并允许端口输出 PWM 波形;
- BIT[5] **PWM2BEC** – PWM2B 使能位及端口输出控制位
 0: 关闭 PWM2B 功能, 并禁止端口输出 PWM 波形;
 1: 使能 PWM2B 功能, 并允许端口输出 PWM 波形;
- BIT[4] **T1EN** – 定时器 T1 使能位
- BIT[3] **PWM1AEC** – PWM1A 使能位及端口输出控制位
- BIT[2] **PWM1BEC** – PWM1B 使能位及端口输出控制位
- BIT[1] **PWM1CEN** – PWM1C 使能位
- BIT[0] **PWM1COE** – PWM1C 端口输出使能位

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	PWM2ANV	PWM2BNV	-	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] **PWM2ANV** – PWM2A 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[6] **PWM2BNV** – PWM2B 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[4:3] **T2CKS[1:0]** – T2 时钟源选择位

T2CKS[1:0]	T2 时钟源
00	FCPU
01	FOSC
10	FHIRC
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
T2CNTH	-	-	-	-	-	T2CNT10	T2CNT9	T2CNT8
R/W	-	-	-	-	-	-	-	-
初始值	-	-	-	-	-	1	1	1

BIT[2:0] T2CNT[10:8] – T2 计数器高 3 位，为不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，为不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	-	-	-	-	-	T2LOAD10	T2LOAD9	T2LOAD8
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	1	1	1

BIT[2:0] T2LOAD[10:8] – T2 重载寄存器高 3 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。

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
T2DATAH	-	-	-	-	-	T2DATA10	T2DATA9	T2DATA8
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2:0] T2DATA[10:8] – T2 比较寄存器 A 高 3 位，用于设置 PWM2A 的占空比

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							
初始值	0	0	0	0	0	0	0	0

BIT[7:0] **T2DATA[7:0]** – T2 比较寄存器 A 低 8 位，用于设置 PWM2A 的占空比

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2DATBH	-	-	-	-	-	T2DATB10	T2DATB9	T2DATB8
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2:0] **T2DATB[10:8]** – T2 比较寄存器 B 高 3 位，用于设置 PWM2B 的占空比

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2DATBL	T2DATB7	T2DATB6	T2DATB5	T2DATB4	T2DATB3	T2DATB2	T2DATB1	T2DATB0
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

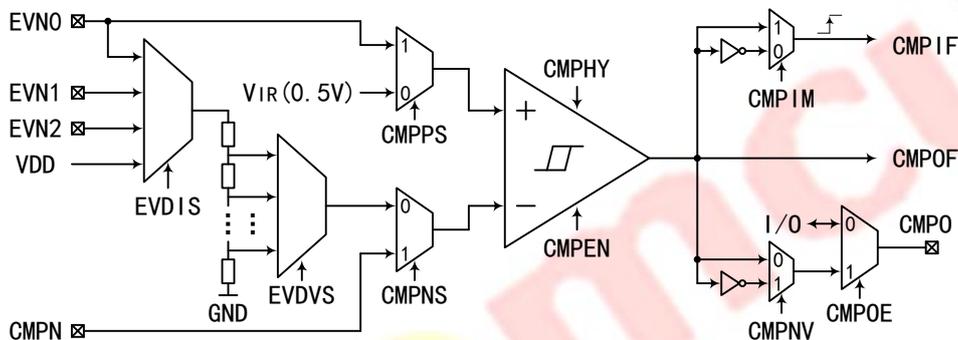
BIT[7:0] **T2DATB[7:0]** – T2 比较寄存器 B 低 8 位，用于设置 PWM2B 的占空比

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，包括内部参考电压 V_{IR} (0.5V) 电路和内部电阻分压电路。

- ◇ 输入共模 0 ~ (VDD-1.4V)，输出可选有/无回滞；
- ◇ 正端输入可选择外部输入电压、或内部参考电压 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；



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 与内部参考电压 V_{IR} 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 VDD 或外部输入的电压检测 (EVD) 功能。通过寄存器位 EVDVS 选择电压检测量值，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参考电压 V_{IR} 的比较结果，即为 VDD 或外部输入电压与电压检测量值的比较结果，从而实现电压检测功能。

CMP 外部输入为模拟信号，相应端口应通过数模控制寄存器关闭数字 I/O 功能，以防止漏电。

注：

1. 开启 CMP、切换输入通道或电压检测量值等操作，需待电路稳定 (时间 > 200us) 后 CMP 输出才有效；
2. CMP 选择快速响应或无回滞输出时，比较器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，即按照电压检测量值换算出的电阻分压比，从 600 个 3KΩ 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. 仅在 CMPO 端口设为输出口时，CMPOE 置 1 将比较结果输出至端口的操作才有效；

8.2 CMP 相关寄存器

CMP 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPCR0	CM PEN	CM PHY	CM PPS	CM PNS	CM PNV	CM POE	CM PIM	CM POF
R/W	R/W	R						
初始值	0	0	0	0	0	0	0	X

BIT[7] **CM PEN** – 比较器 CMP 使能位

- 0: 关闭 CMP;
- 1: 开启 CMP;

BIT[6] **CM PHY** – CMP 输出回滞控制位

- 0: CMP 输出无回滞;
- 1: CMP 输出有回滞;

BIT[5] **CM PPS** – CMP 正端输入选择位

- 0: 正端输入为内部参考电压 V_{IR} ;
- 1: 正端输入为端口 EVN0 输入电压;

BIT[4] **CM PNS** – CMP 负端输入选择位

- 0: 负端输入为内部电阻分压电压;
- 1: 负端输入为端口 CMPN 输入电压;

BIT[3] **CM PNV** – 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[2] **CM POE** – 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 (仅在端口为输出口时有效);

BIT[1] **CM PIM** – CMP 中断触发方式选择位

- 0: CMP 输出的下降沿触发;
- 1: CMP 输出的上升沿触发;

BIT[0] **CM POF** – CMP 输出状态标志位

- 0: CMP 输出为低, 即其正端输入电压低于负端输入电压; 或 CMP 关闭;
- 1: CMP 输出为高, 即其正端输入电压高于负端输入电压;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPCR1	EVDIS1	EVDIS0	EVDVS5	EVDVS4	EVDVS3	EVDVS2	EVDVS1	EVDVS0
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

BIT[7:6] EVDIS[1:0] – EVD 被检测电压选择位

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

BIT[5:0] EVDVS[5:0] – EVD 电压检测量值选择位

EVDVS[5:0]	电压检测量值	电阻分压比	EVDVS[5:0]	电压检测量值	电阻分压比
00 0000	0.56V	536R/600R	10 0000	2.80V	107R/600R
00 0001	0.63V	476R/600R	10 0001	2.87V	105R/600R
00 0010	0.70V	429R/600R	10 0010	2.94V	102R/600R
00 0011	0.77V	390R/600R	10 0011	3.01V	100R/600R
00 0100	0.84V	357R/600R	10 0100	3.08V	97R/600R
00 0101	0.91V	330R/600R	10 0101	3.15V	95R/600R
00 0110	0.98V	306R/600R	10 0110	3.22V	93R/600R
00 0111	1.05V	286R/600R	10 0111	3.29V	91R/600R
00 1000	1.12V	268R/600R	10 1000	3.36V	89R/600R
00 1001	1.19V	252R/600R	10 1001	3.43V	87R/600R
00 1010	1.26V	238R/600R	10 1010	3.50V	86R/600R
00 1011	1.33V	226R/600R	10 1011	3.57V	84R/600R
00 1100	1.40V	214R/600R	10 1100	3.64V	82R/600R
00 1101	1.47V	204R/600R	10 1101	3.71V	81R/600R
00 1110	1.54V	195R/600R	10 1110	3.78V	79R/600R
00 1111	1.61V	186R/600R	10 1111	3.85V	78R/600R
01 0000	1.68V	179R/600R	11 0000	3.92V	77R/600R
01 0001	1.75V	171R/600R	11 0001	3.99V	75R/600R
01 0010	1.82V	165R/600R	11 0010	4.06V	74R/600R
01 0011	1.89V	159R/600R	11 0011	4.13V	73R/600R
01 0100	1.96V	153R/600R	11 0100	4.20V	71R/600R
01 0101	2.03V	148R/600R	11 0101	4.27V	70R/600R
01 0110	2.10V	143R/600R	11 0110	4.34V	69R/600R
01 0111	2.17V	138R/600R	11 0111	4.41V	68R/600R
01 1000	2.24V	134R/600R	11 1000	4.48V	67R/600R
01 1001	2.31V	130R/600R	11 1001	4.55V	66R/600R
01 1010	2.38V	126R/600R	11 1010	4.62V	65R/600R
01 1011	2.45V	122R/600R	11 1011	4.69V	64R/600R
01 1100	2.52V	119R/600R	11 1100	4.76V	63R/600R
01 1101	2.59V	116R/600R	11 1101	4.83V	62R/600R
01 1110	2.66V	113R/600R	11 1110	4.90V	61R/600R
01 1111	2.73V	110R/600R	11 1111	4.97V	60R/600R

9 中断

芯片的中断源包括外部中断 (INT0~INT1)、定时器中断 (T0~T2)、比较器 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，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。

注：要使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。

9.1 外部中断

芯片有 2 路外部中断源 INT0/INT1，可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 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。

注：当 P10 端口通过寄存器位 EIS 置 1 复用为 INT0 时，端口的键盘中断唤醒功能无效。

9.2 定时器中断

定时器 Tn (n=0-2) 在计数溢出时触发定时器中断，中断标志 TnIF (n=0-2) 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=0-2) 为 1，则产生定时器中断。

9.3 键盘中断

芯片有 16 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
P0KBCR	P07KE	P06KE	P05KE	P04KE	P03KE	P02KE	P01KE	P00KE
R/W	R/W							
初始值	0	0	0	0	0	0	0	0

BIT[7:0] **P0nKE** – P0n 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;

	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							
初始值	1	1	1	1	1	1	1	1

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;

9.4 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 CMP 中断，中断标志 CMPIF 将被置 1，若中断总使能位 GIE 为 1 且 CMP 中断使能位 CMPIE 为 1，则产生 CMP 中断。

9.5 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
INTE1	GIE	INT1M	INT0M	-	CMPIE	T2IE	T1IE	INT1IE
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] **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[6] **INT1M** – 外部中断 INT1 触发方式选择位

- 0: 下降沿触发;
- 1: 上升沿触发;

BIT[5] **INT0M** – 外部中断 INT0 触发方式选择位
0: 下降沿触发;
1: 上升沿触发;

BIT[3] **CMPIE** – CMP 中断使能位
0: 屏蔽 CMP 中断;
1: 使能 CMP 中断;

BIT[2] **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[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] **INT1IE** – INT1 中断使能位
0: 屏蔽 INT1 中断;
1: 使能 INT1 中断;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE0	-	-	-	-	-	INTOIE	KBIE	TOIE
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2] **INTOIE** – INTO 中断使能位
0: 屏蔽 INTO 中断;
1: 使能 INTO 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
INTF1	-	-	-	-	CMPIF	T2IF	T1IF	INT1IF
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	0	0	0	0

- BIT[3] **CMPIF** – 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;
- 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] **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;
- BIT[0] **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 0;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTF0	-	-	-	-	-	INTOIF	KBIF	TOIF
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

- BIT[2] **INTOIF** – INTO 中断标志位
0: 未触发 INTO 中断;
1: 已触发 INTO 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] **TOIF** – 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;

注: 兼容模式下, 读 INTF0 操作的结果, 不是 INTF0 的实际寄存器值, 而是 INTF0 与 INTE0 的逻辑与的值。

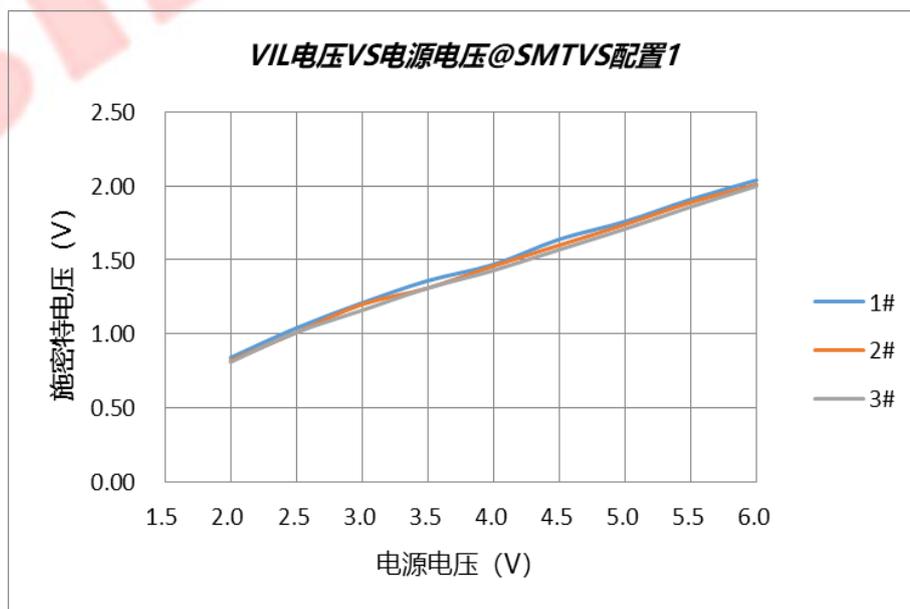
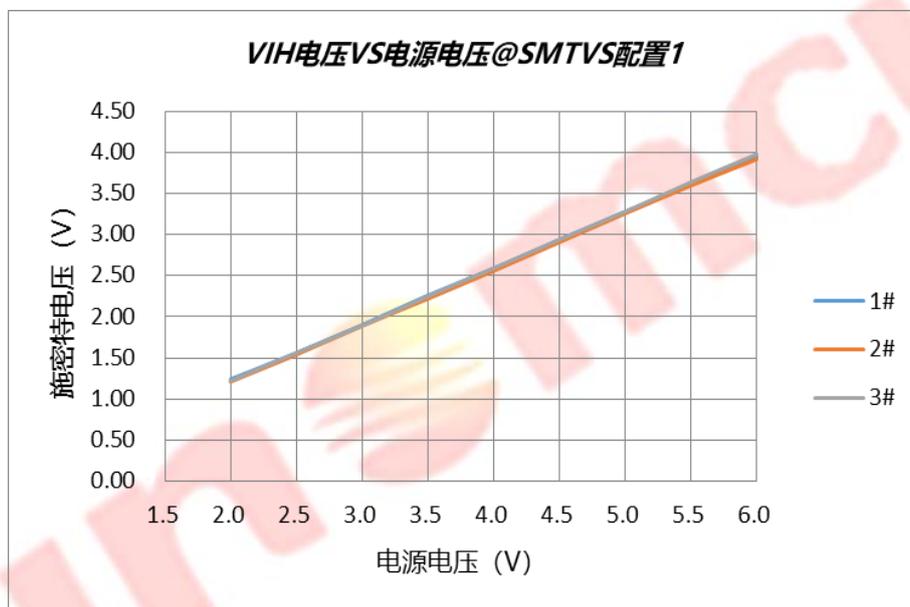
10 特性曲线

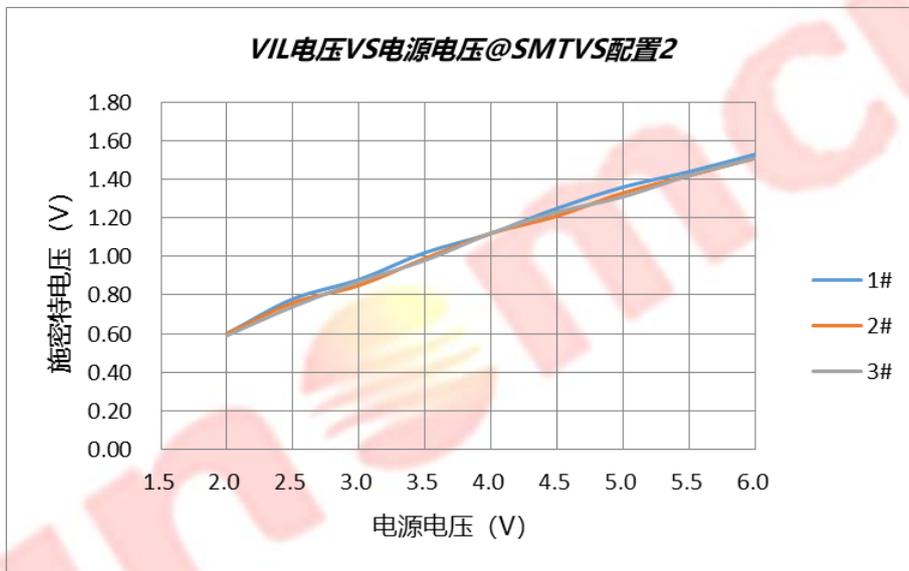
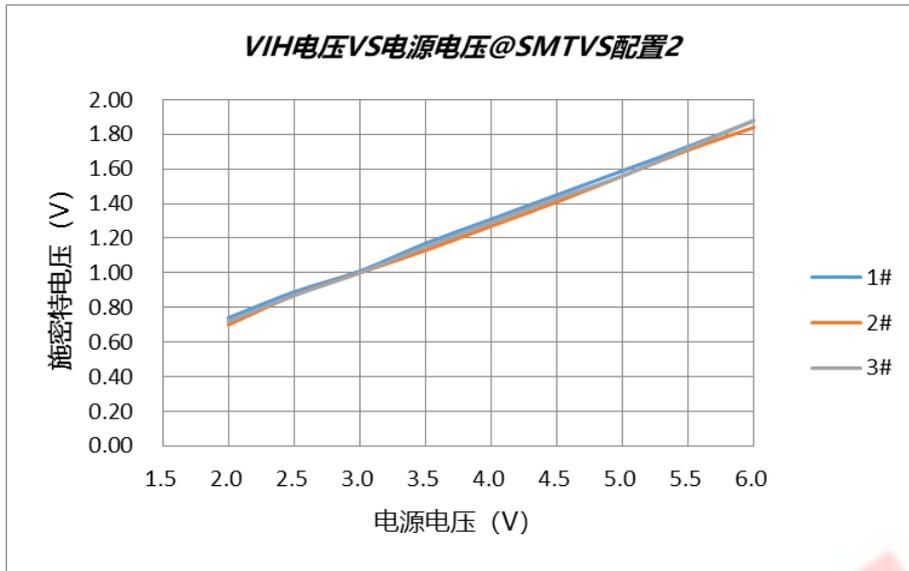
注:

1. 特性曲线图中数据均来自抽样实测, 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. 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 $VDD=5\text{V}$;

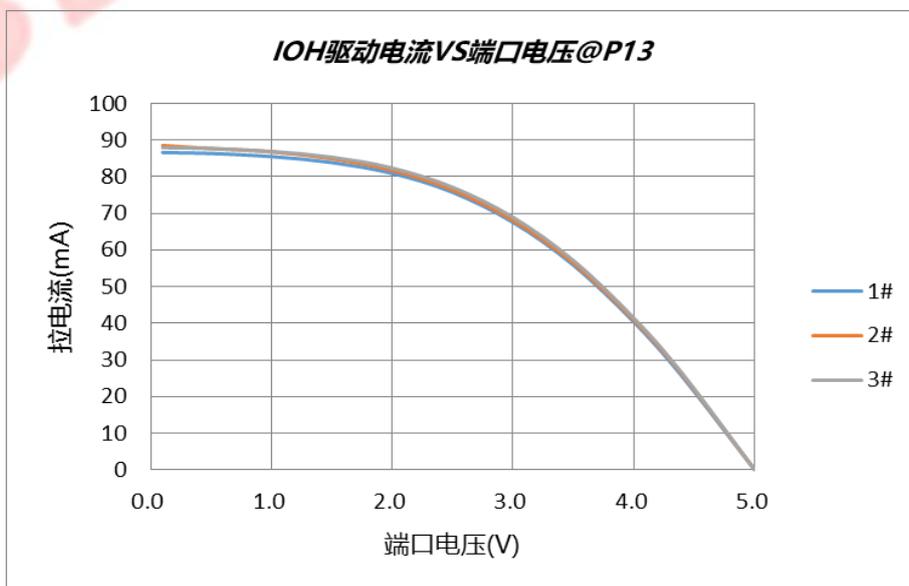
10.1 I/O 特性

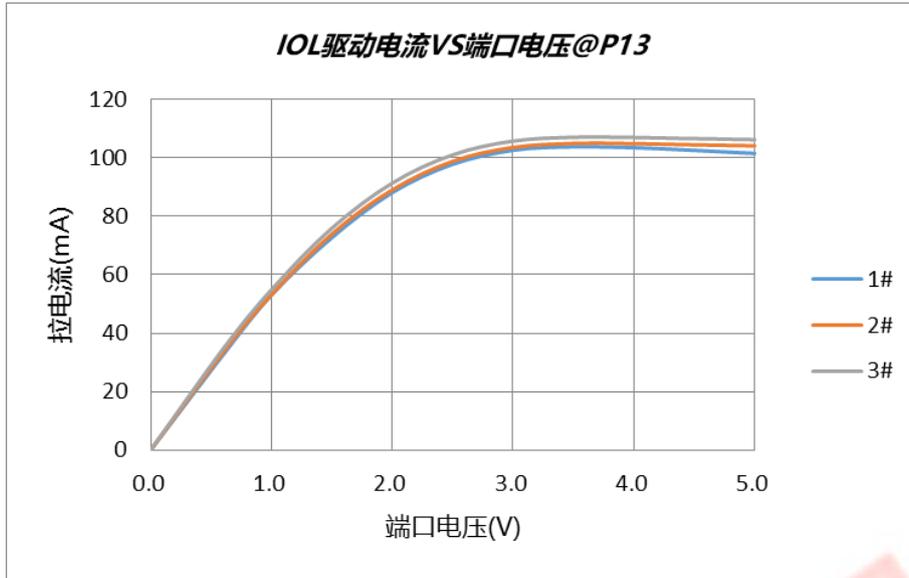
输入 SMT 阈值电压 VS 电源电压



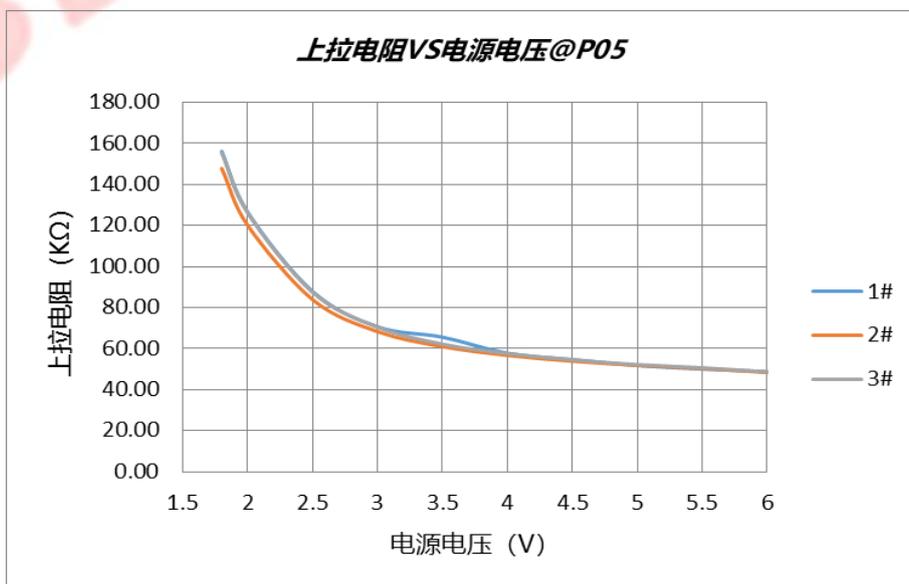
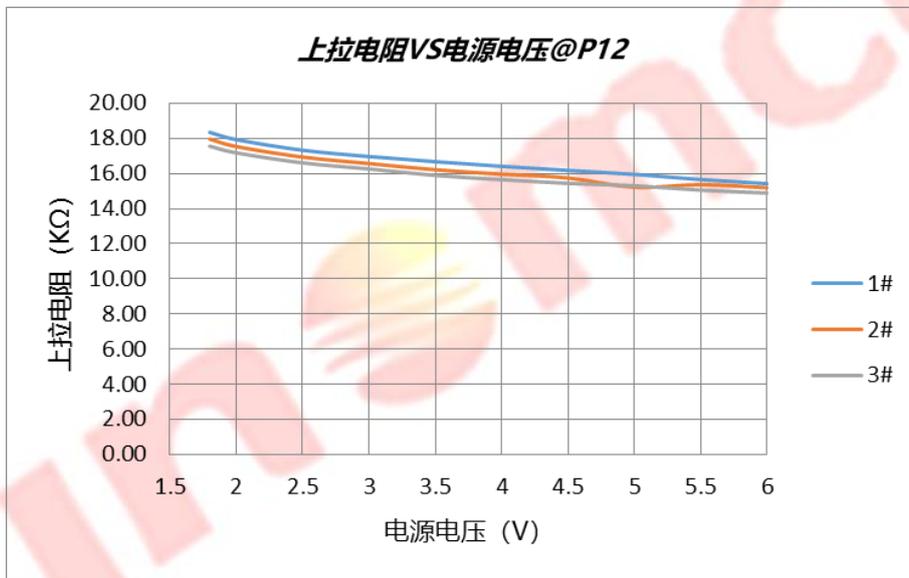


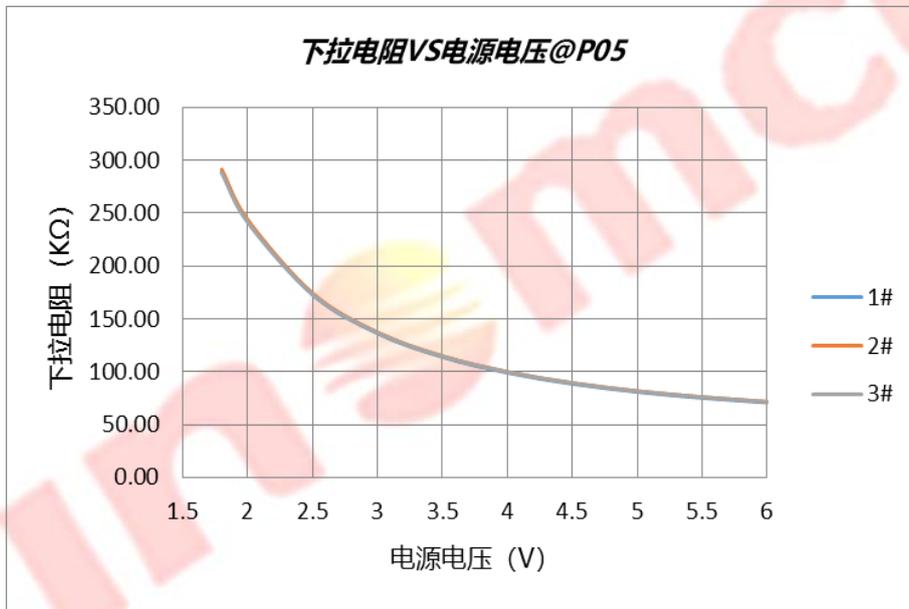
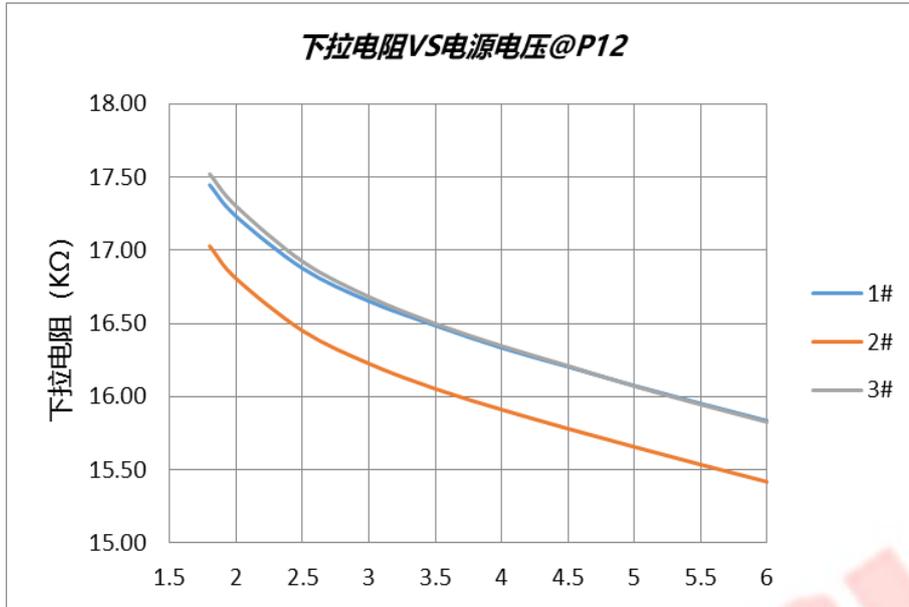
I/O 输出 驱动电流 VS 端口电压 (VDD=5V)





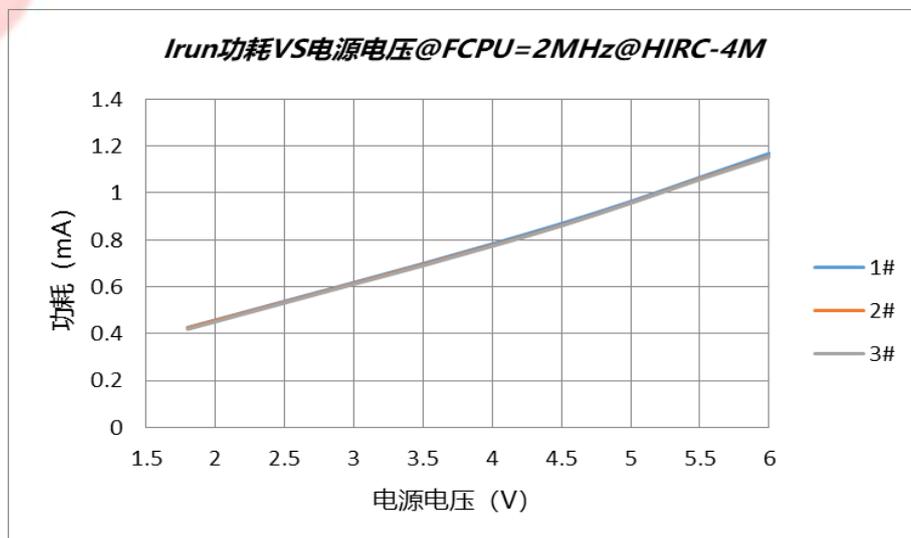
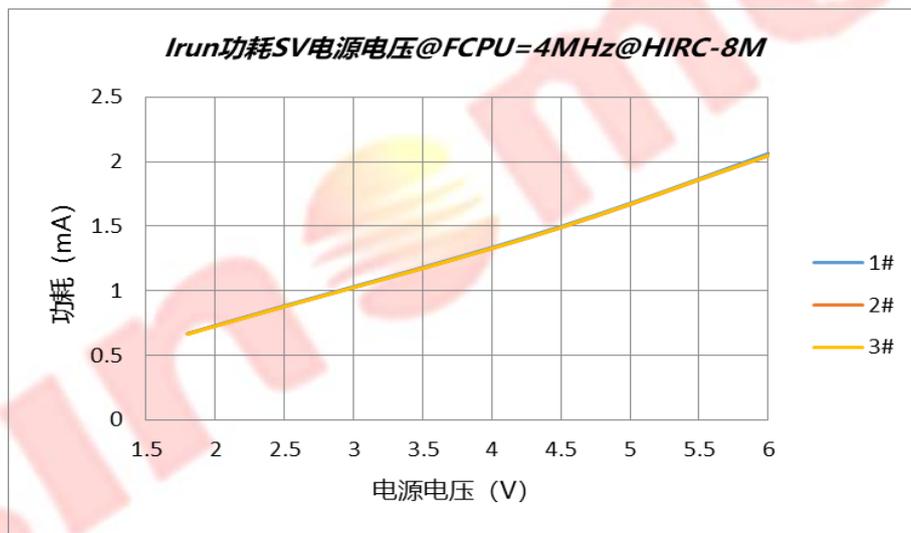
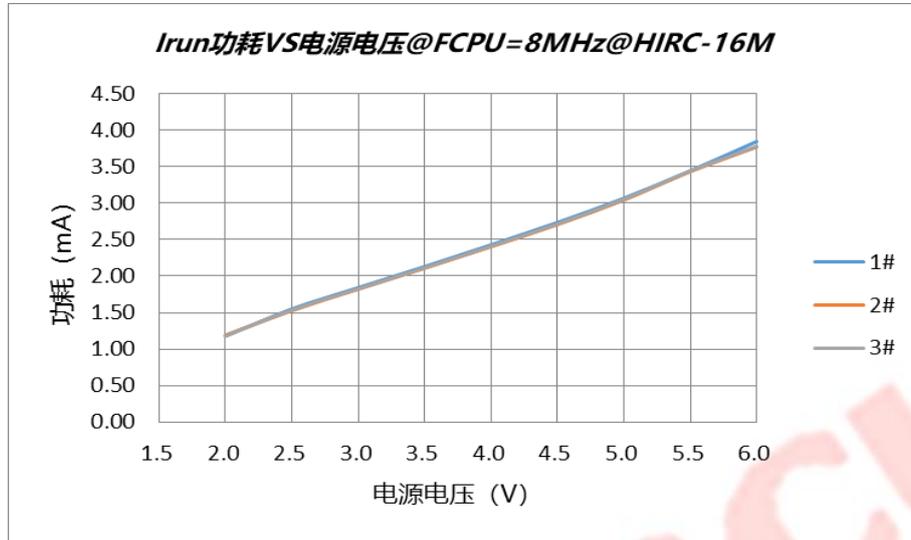
上/下拉电阻值 VS 电源电压

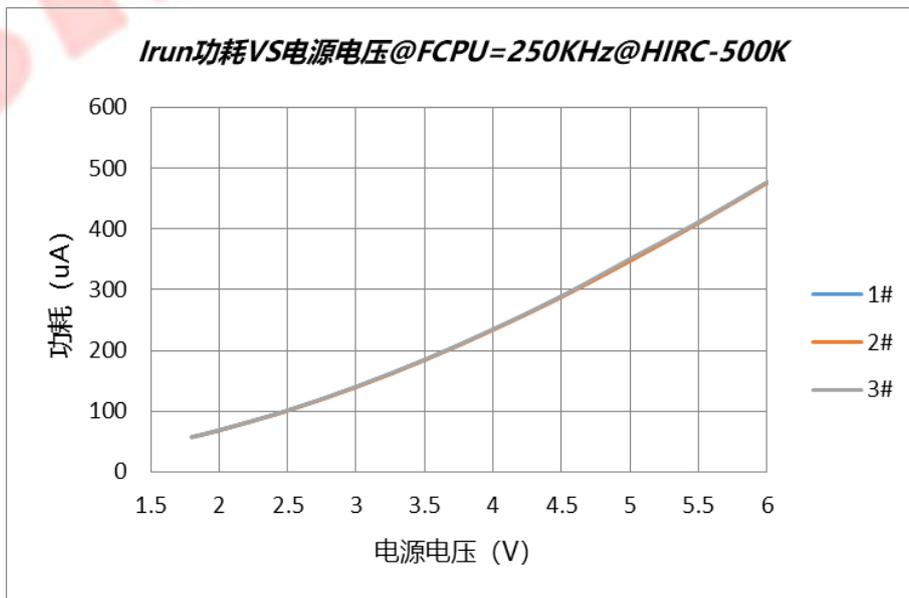
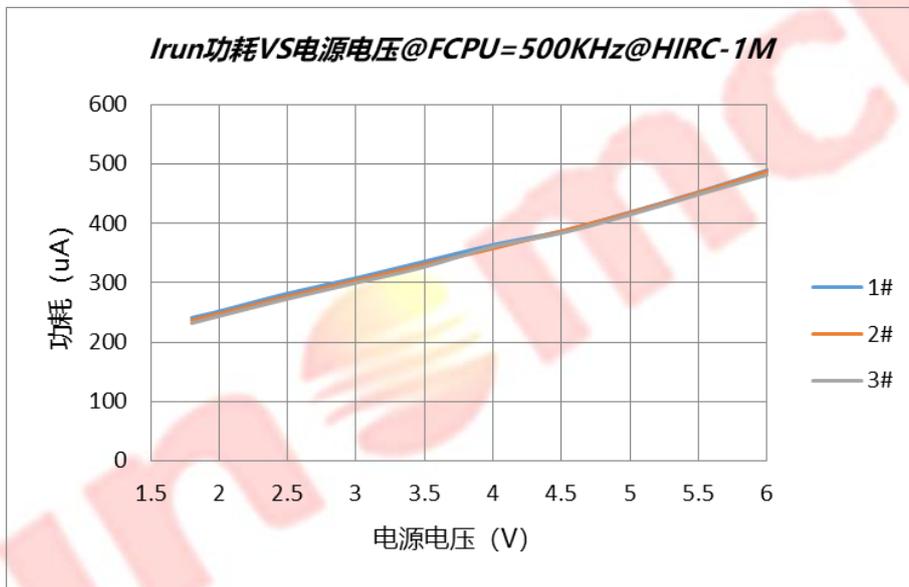
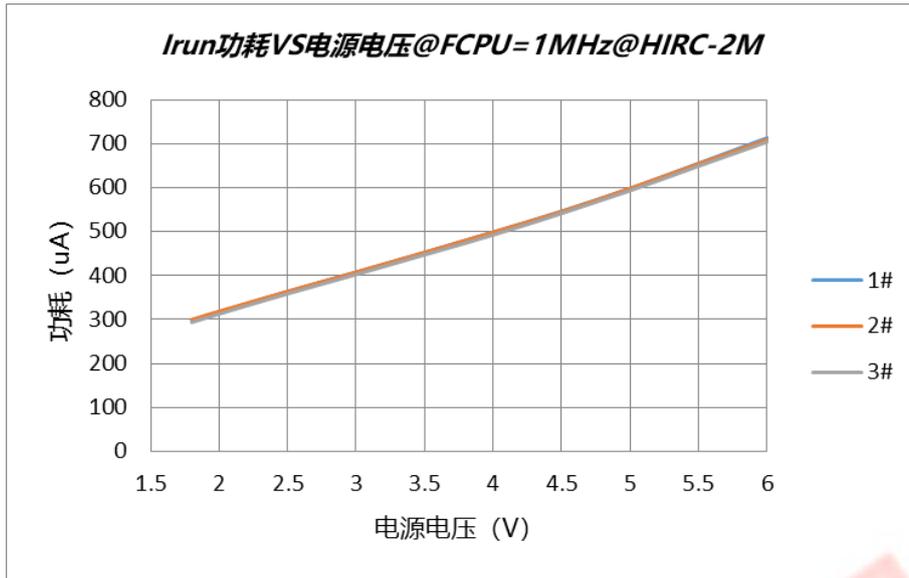




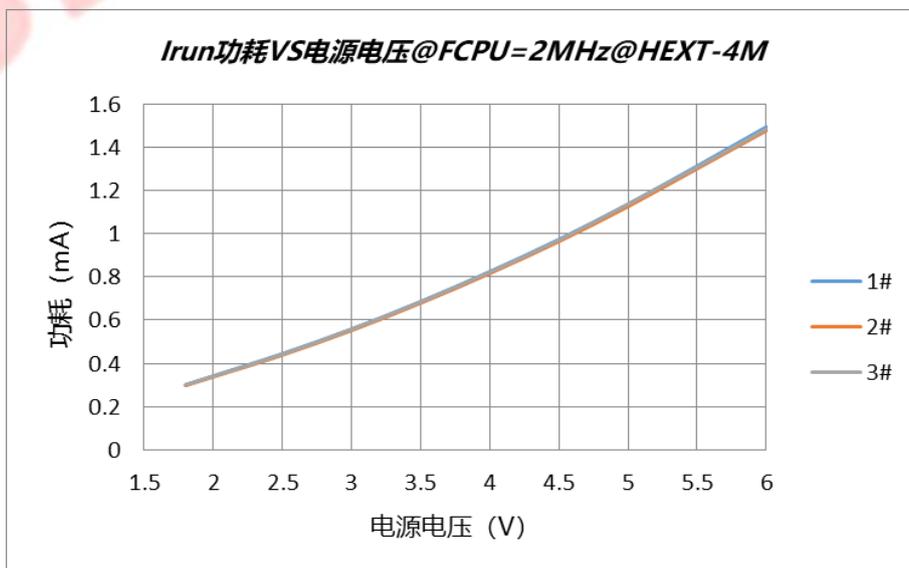
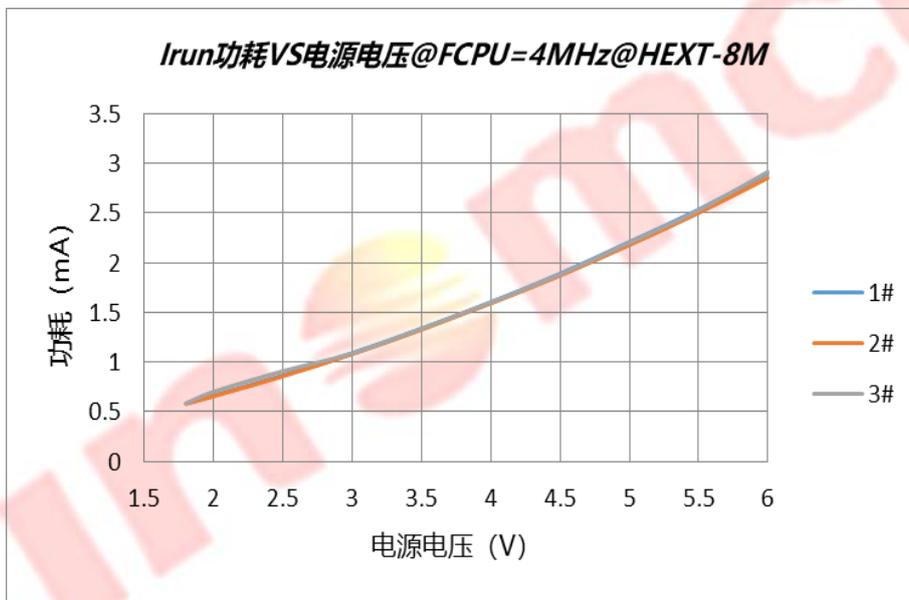
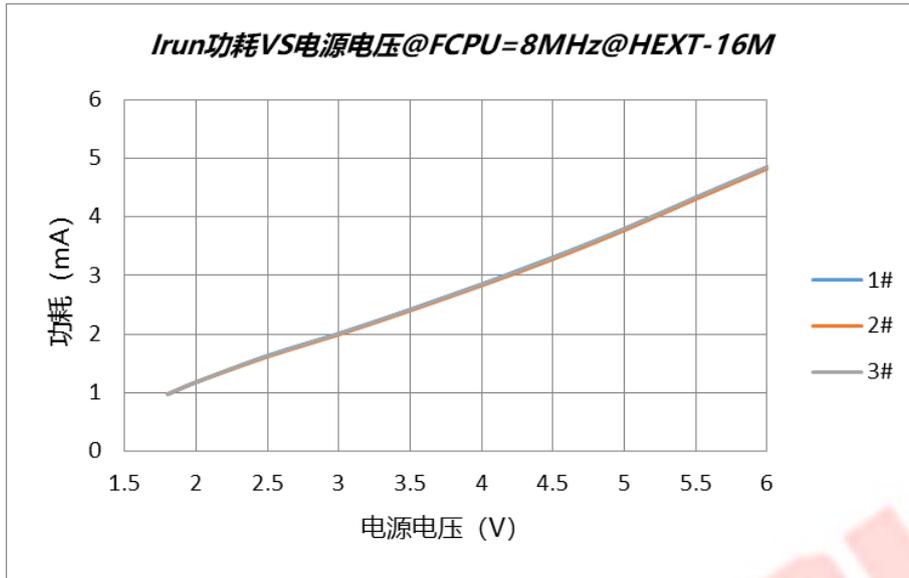
10.2 功耗特性

运行模式@HIRC 功耗 VS 电源电压

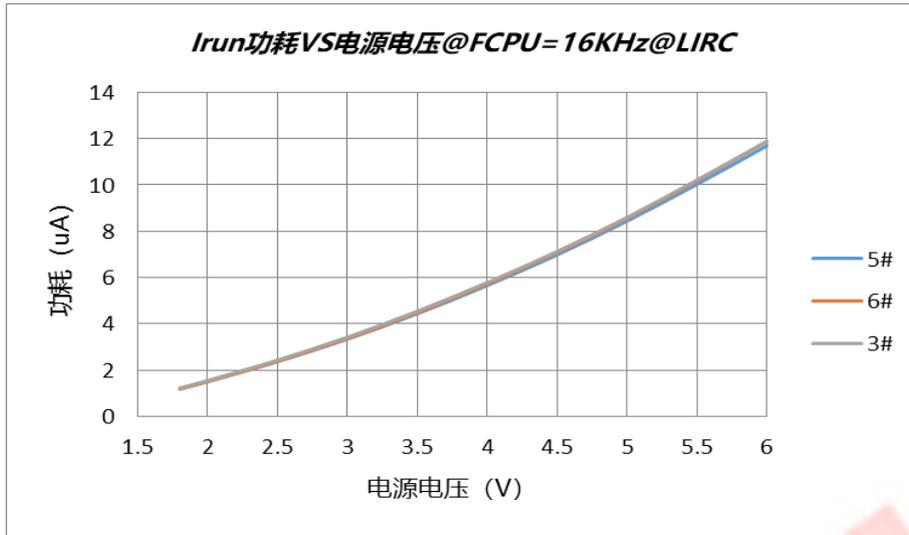




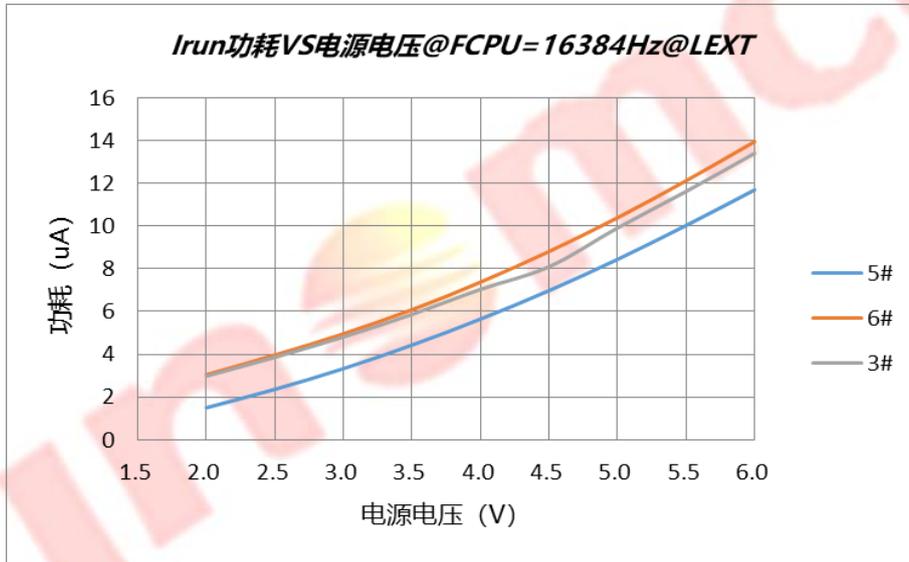
运行模式@HEXT 功耗 VS 电源电压



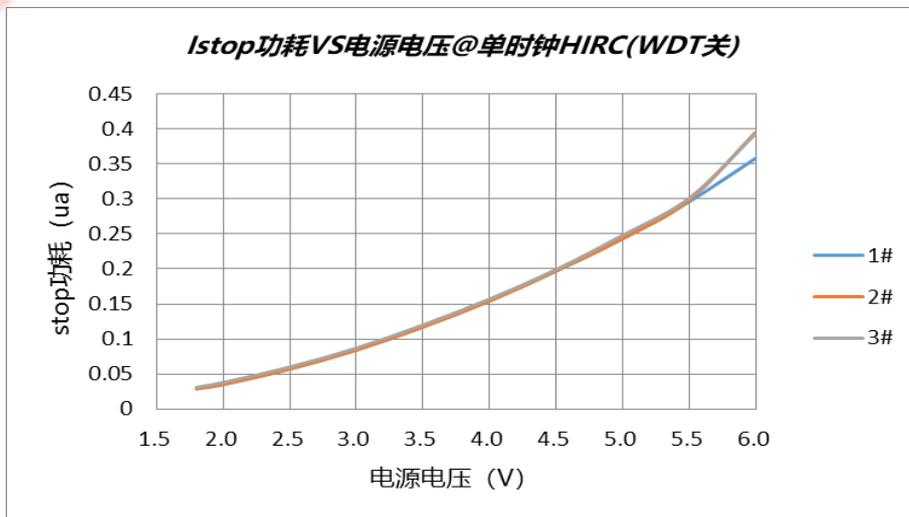
运行模式@LIRC 功耗 VS 电源电压

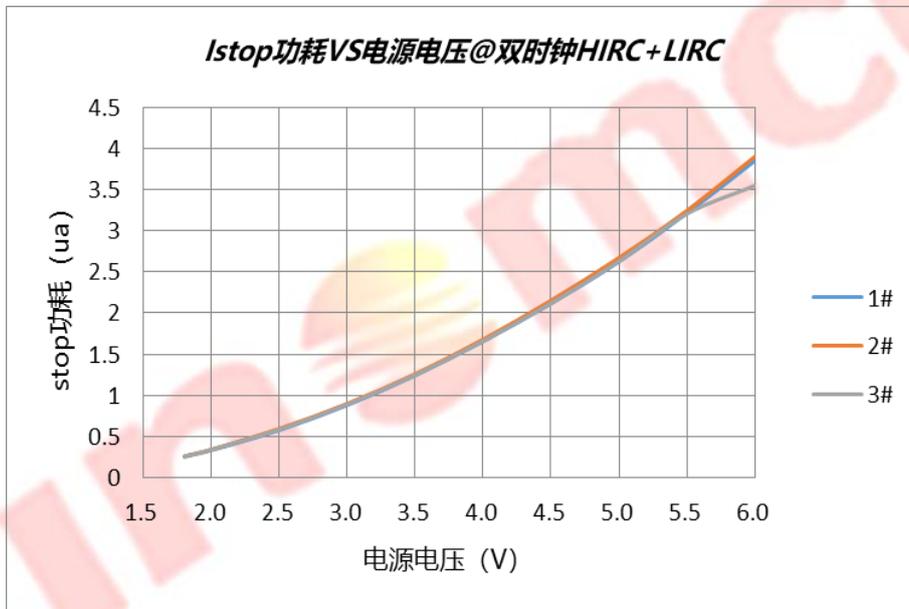
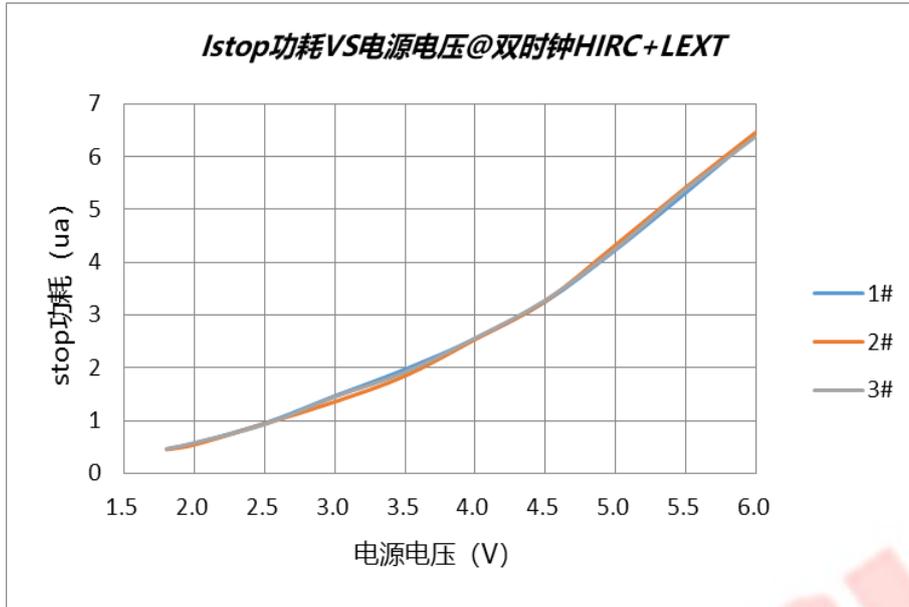


运行模式@LEXT 功耗 VS 电源电压



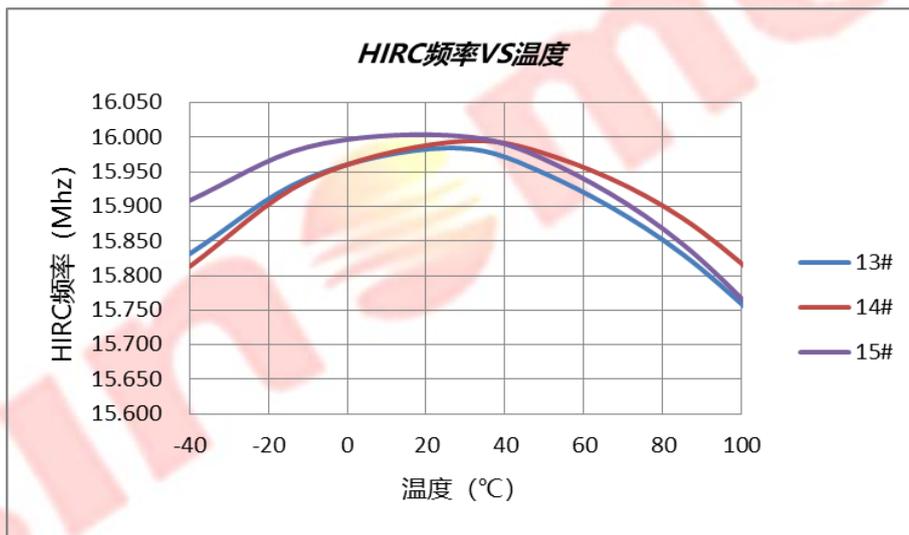
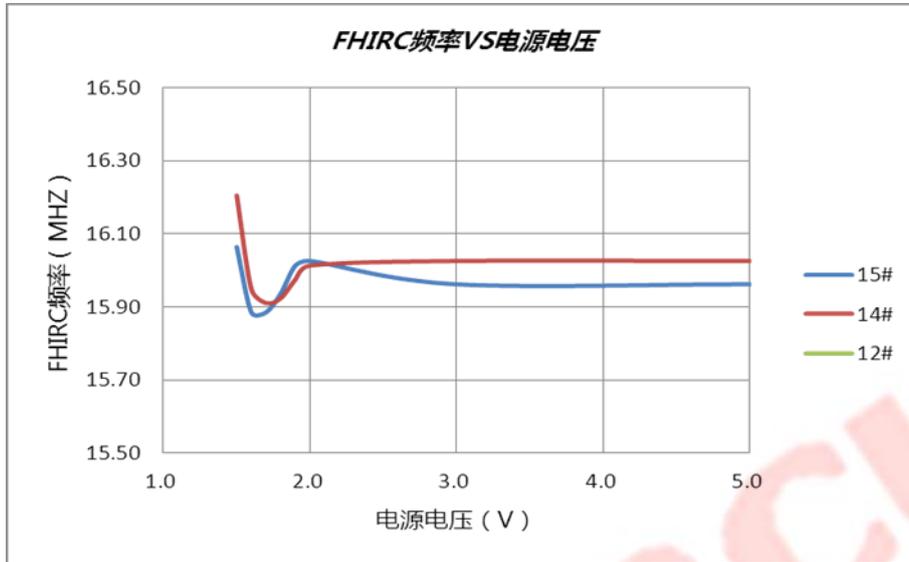
休眠模式 功耗 VS 电源电压



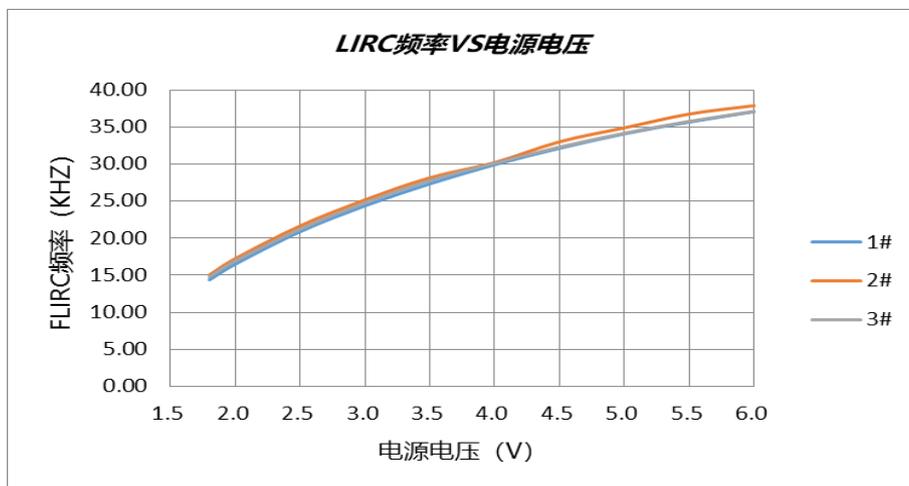


10.3 模拟电路特性

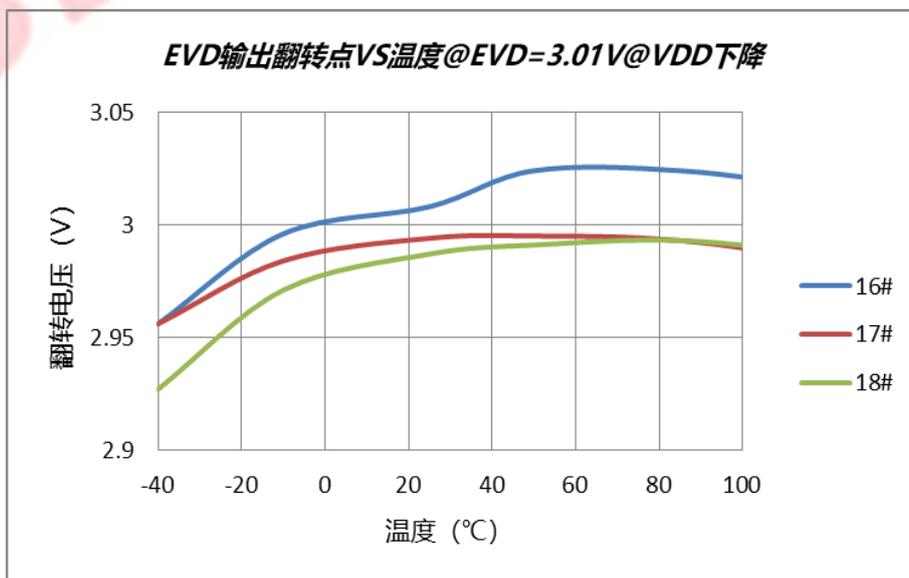
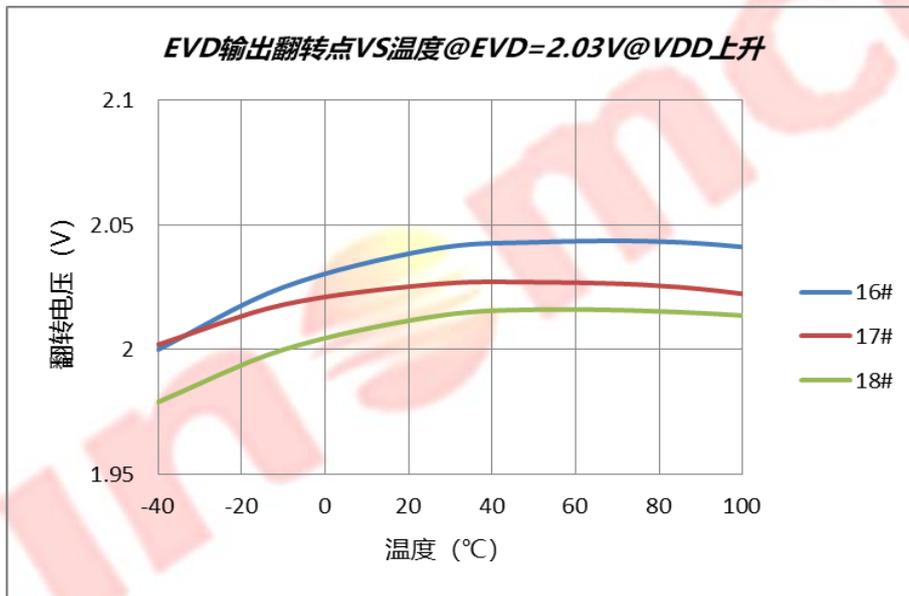
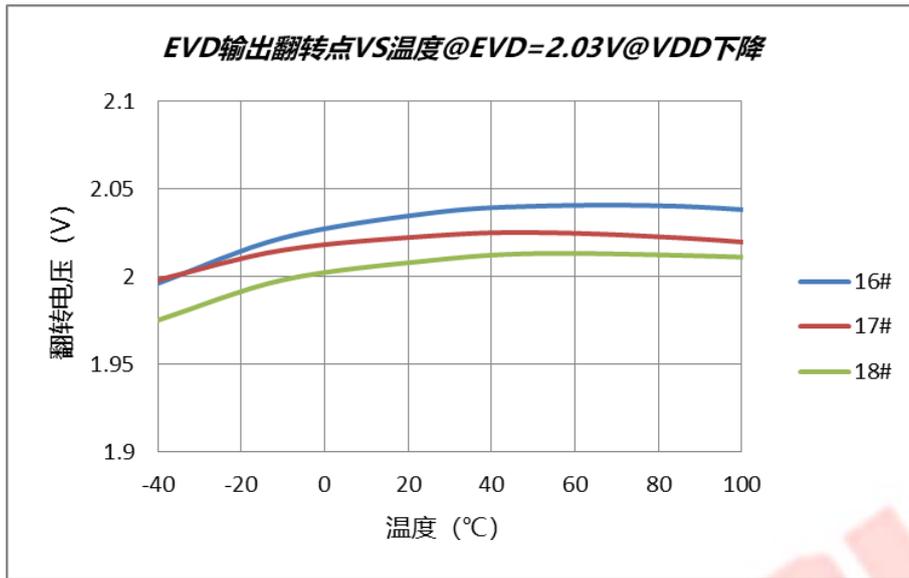
HIRC 频率 VS 电源电压/温度

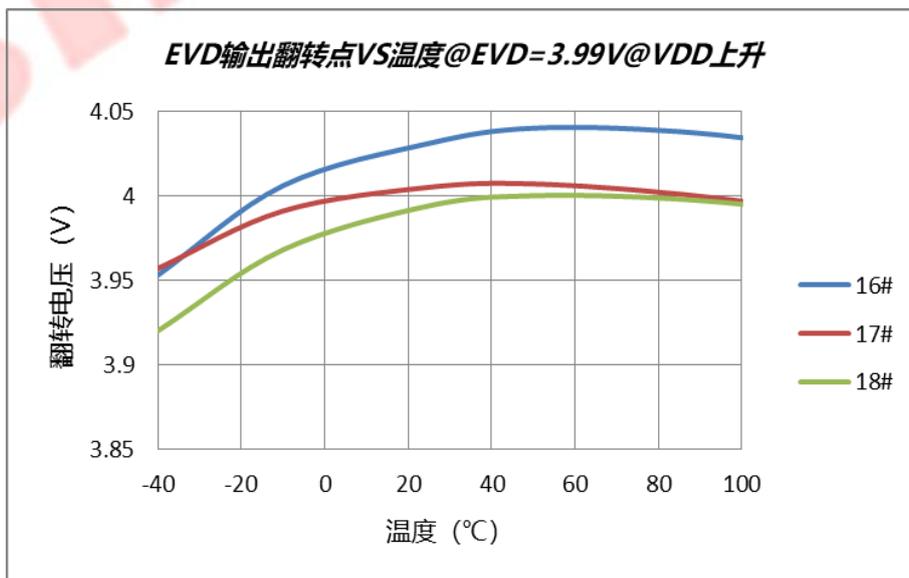
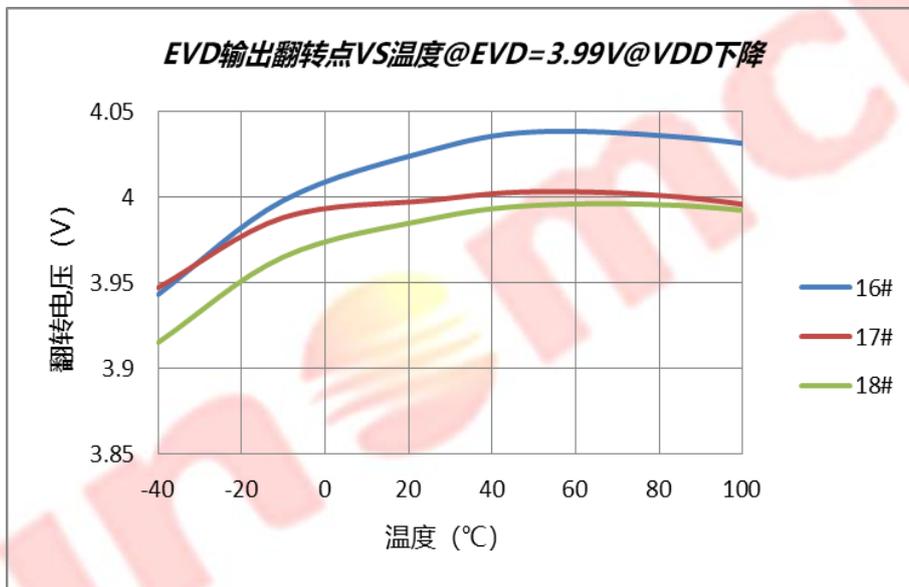
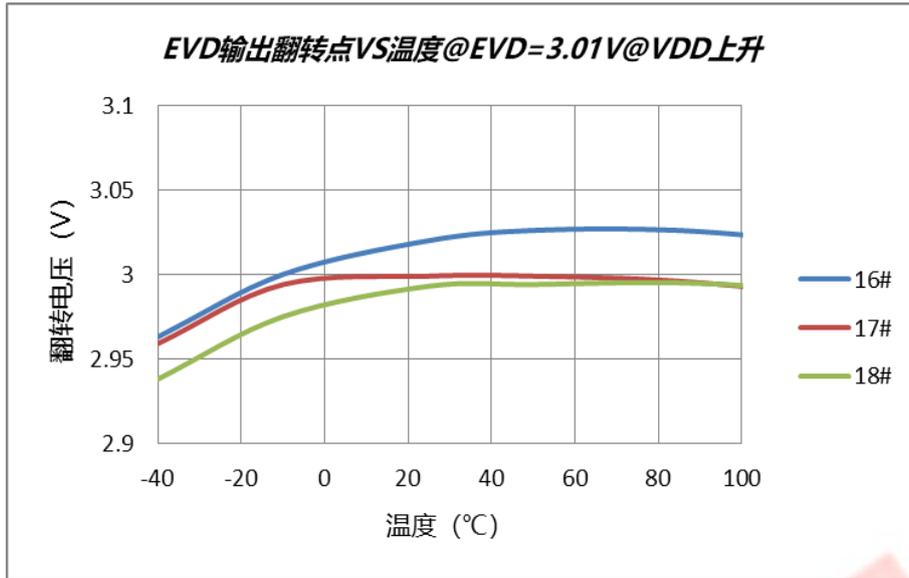


LIRC 频率 VS 电源电压



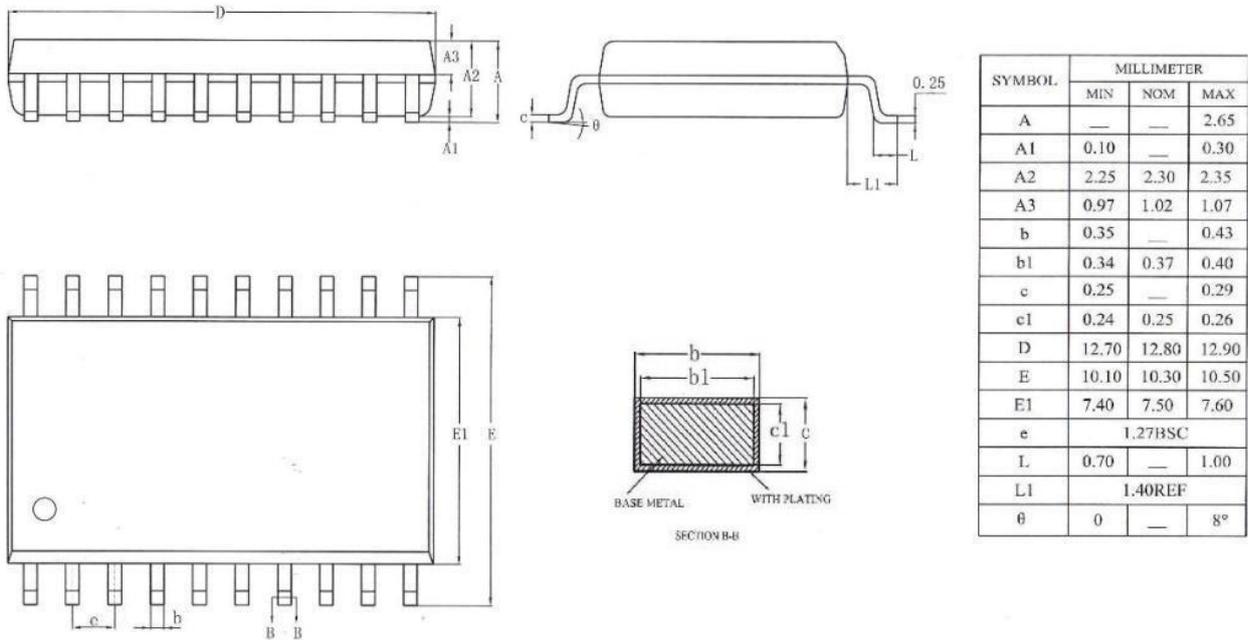
CMP 电压检测(EVD) VS 温度



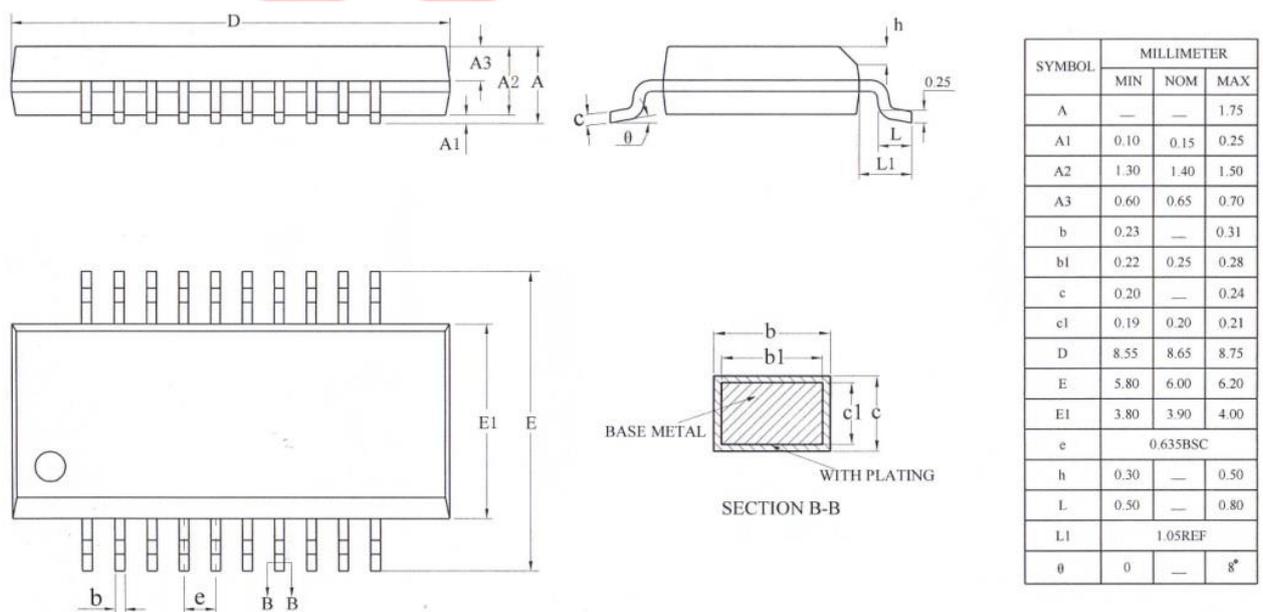


11 封装尺寸

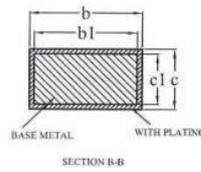
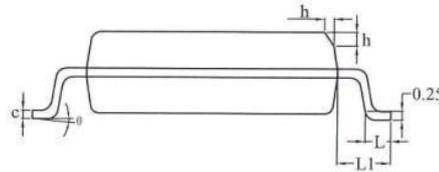
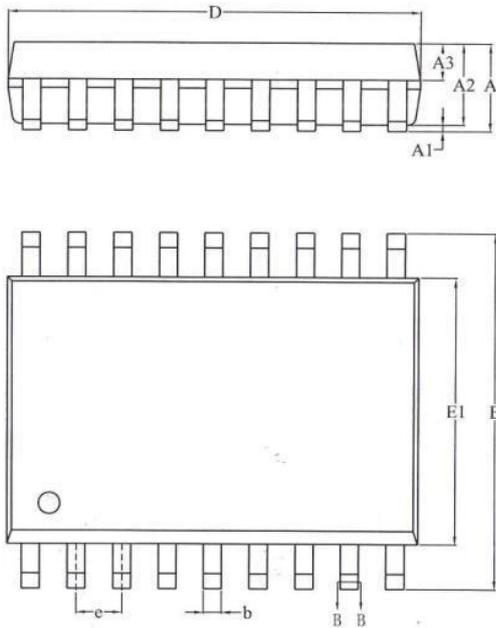
11.1 SOP20



11.2 SSOP20

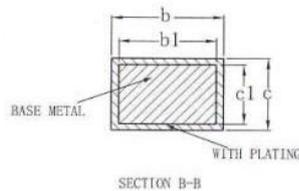
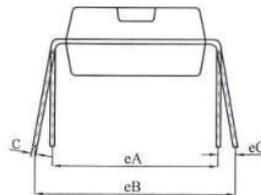
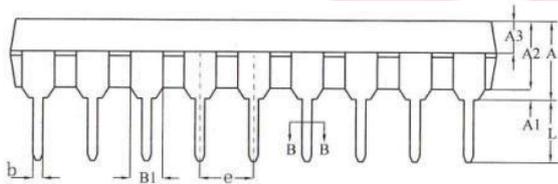


11.3 SOP18



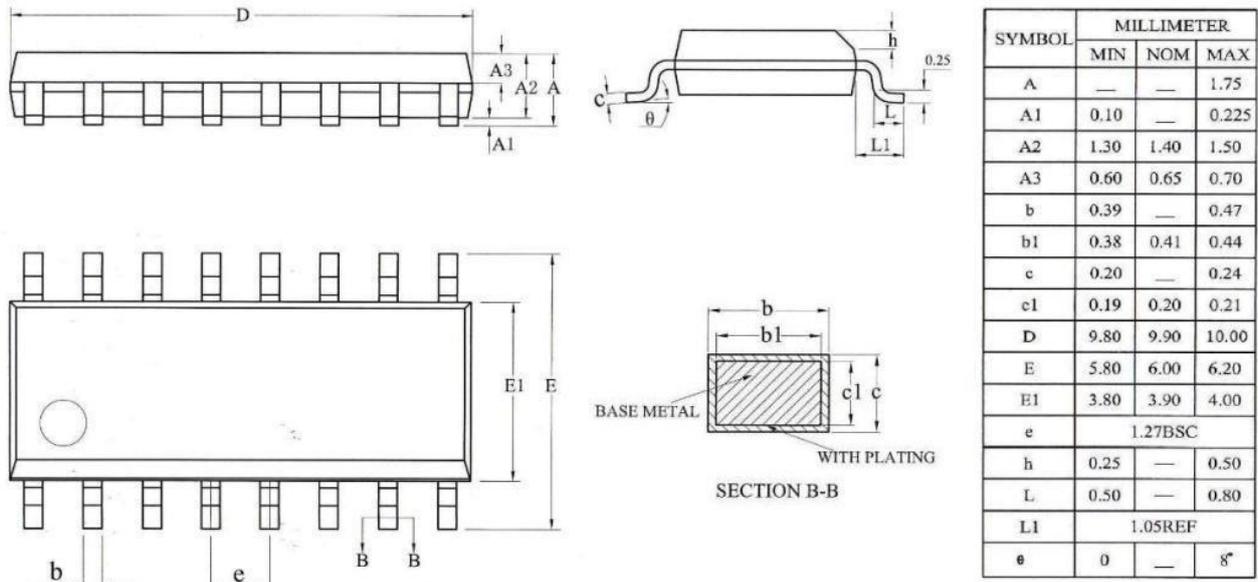
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	2.65
A1	0.10	—	0.30
A2	2.20	2.30	2.40
A3	0.97	1.02	1.07
b	0.35	—	0.43
b1	0.34	0.37	0.40
c	0.25	—	0.29
c1	0.24	0.25	0.26
D	11.35	11.45	11.55
E	10.10	10.30	10.50
E1	7.40	7.50	7.60
e	1.27BSC		
L	0.70	—	1.00
L1	1.40REF		
h	0.25	—	0.75
θ	0	—	8°

11.4 DIP18

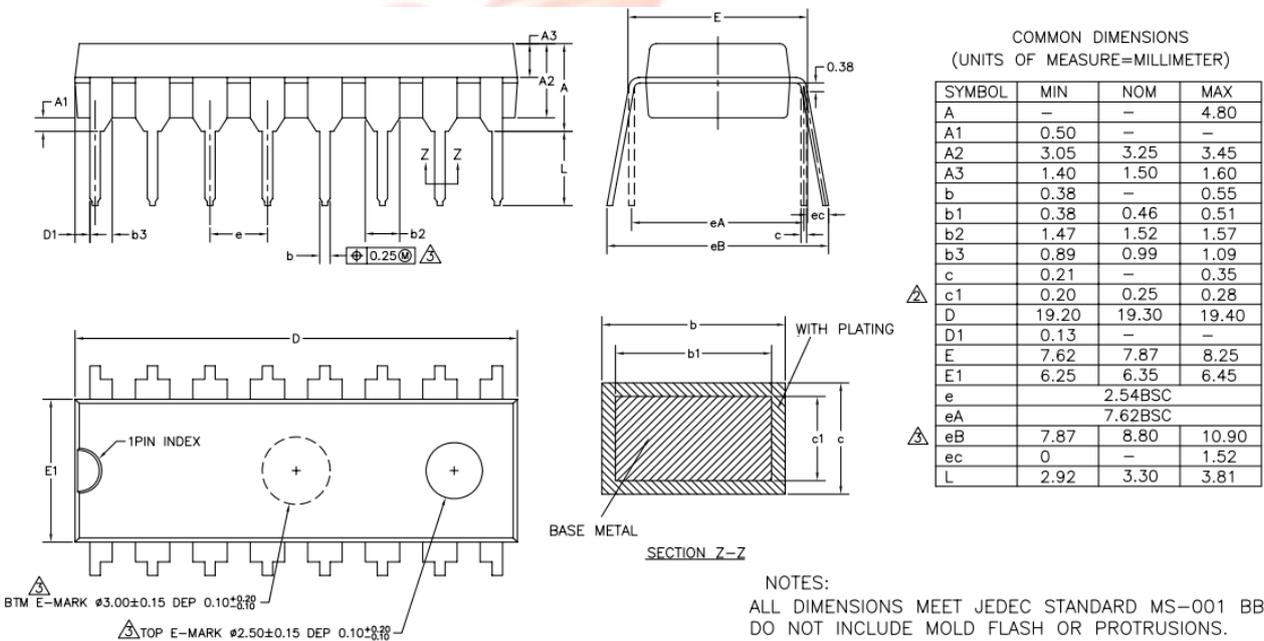


SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	3.60	3.80	4.00
A1	0.51	—	—
A2	3.20	3.30	3.40
A3	1.47	1.52	1.57
b	0.44	—	0.52
b1	0.43	0.46	0.49
B1	1.52REF		
c	0.25	—	0.29
c1	0.24	0.25	0.26
D	22.80	22.90	23.00
E1	6.45	6.55	6.65
e	2.54BSC		
eA	7.62REF		
eB	7.62	—	9.30
eC	0	—	0.84
L	3.00	—	—

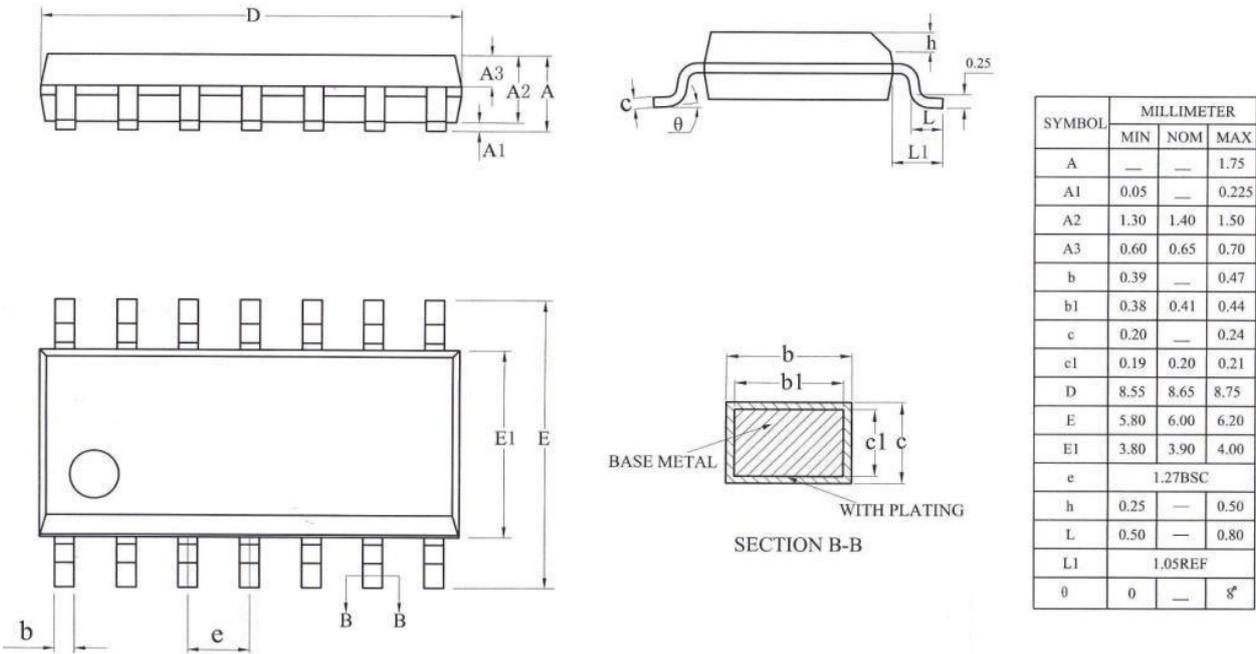
11.5 SOP16



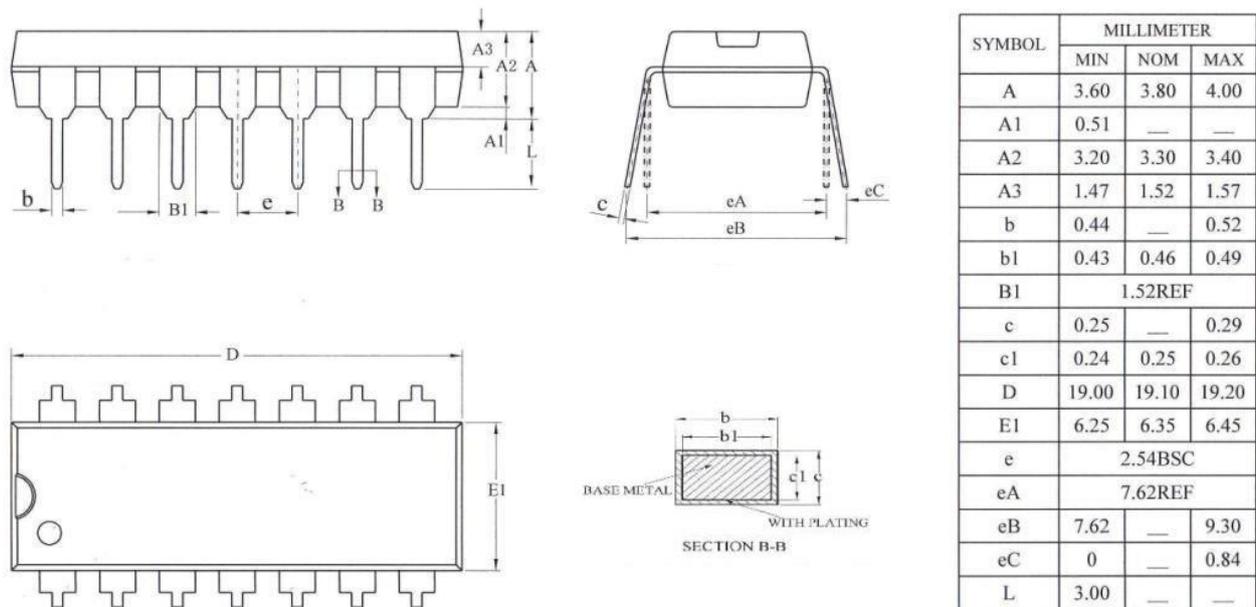
11.6 DIP16



11.7 SOP14



11.8 DIP14



12 修订记录

版本	修订日期	修订内容
V1.0	2020-08-27	初版发布;