

SinoMCU 8 位单片机

MC32P7351

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	6
1.4	端口说明	7
2	电气特性	9
2.1	极限参数	9
2.2	直流电气特性	9
2.3	交流电气特性	11
2.4	ADC 特性参数	11
2.5	PFRC 特性参数	12
3	CPU 与存储器	13
3.1	指令集	13
3.2	程序存储器	15
3.3	数据存储器	16
3.4	堆栈	17
3.5	控制寄存器	17
3.6	用户配置字	21
4	系统时钟	22
4.1	内部高频 RC 振荡器	22
4.2	内部低频 RC 振荡器	23
4.3	外部晶体振荡器	23
4.4	系统工作模式	24
4.5	低功耗模式	25
5	复位	27
5.1	复位条件	27
5.2	上电复位	28
5.3	外部复位	28
5.4	低电压复位	28
5.5	看门狗复位	28
6	I/O 端口	29
6.1	通用 I/O 功能	29
6.2	内部上/下拉电阻	30
6.3	端口模式控制	31
6.4	端口驱动控制	32
7	可编程 RC 振荡器 PFRC	34
8	定时器 TIMER	36
8.1	看门狗定时器 WDT	36
8.2	定时器 T0	36
8.3	定时器 T1	39
8.4	定时器 T2	42

8.5	定时器 T3.....	45
8.6	PWM 相关寄存器.....	46
9	模数转换器 ADC	49
9.1	ADC 概述	49
9.2	ADC 相关寄存器.....	50
9.3	ADC 操作步骤	52
9.4	ADC 零点偏移修调流程	53
10	低电压检测 LVD.....	54
11	中断.....	55
11.1	外部中断	55
11.2	定时器中断.....	55
11.3	键盘中断	55
11.4	ADC 中断	56
11.5	中断相关寄存器.....	56
12	特性曲线.....	59
12.1	I/O 特性.....	59
12.2	功耗特性	66
12.3	模拟电路特性	70
13	封装尺寸.....	74
13.1	SOP16	74
13.2	DIP16.....	74
13.3	SOP14	75
13.4	DIP14.....	75
13.5	SOP8.....	76
13.6	DIP8.....	76
13.7	MSOP10	77
14	修订记录.....	78

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 F_{HOSC} 的 4/8/16/32/64/128/256 分频
 - ◇ 系统低频时钟下 F_{CPU} 固定为 F_{LOSC} 的 2 分频
- 程序存储器
 - ◇ 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
- 数据存储器
 - ◇ 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)
 - ◇ 所有端口均支持施密特输入，均支持推挽输出
 - ◇ P00/P01/P13 可选推挽或开漏输出，且支持 5V 耐受
 - ◇ P13 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
 - ◇ P16/P17 为大电流端口且输出电流 4 级可选，其余端口输出电流 2 级可选，其中 P00 支持灌电流大电流输出
 - ◇ P11/P12 可复用为外部中断输入，支持外部中断唤醒功能
 - ◇ 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 (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（低功耗模式）：CPU 暂停，高频时钟源工作，低频时钟源可选停止或工作
 - ◇ HOLD2 模式（低功耗模式）：CPU 暂停，高频时钟源停止，低频时钟源工作
 - ◇ 休眠模式（低功耗模式）：CPU 暂停，高/低频时钟源均停止
- 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 和 8+3 模式的 PWM (2 路输出通道可选)
 - ◇ 8 位定时器 T1，可实现外部计数、BUZ 和 8+3 模式的 PWM (可扩展为 1 对带死区的互补 PWM)
 - ◇ 8 位定时器 T2，可实现 BUZ、8+3 模式的 PWM (可扩展为 2 对互反的带死区互补 PWM)

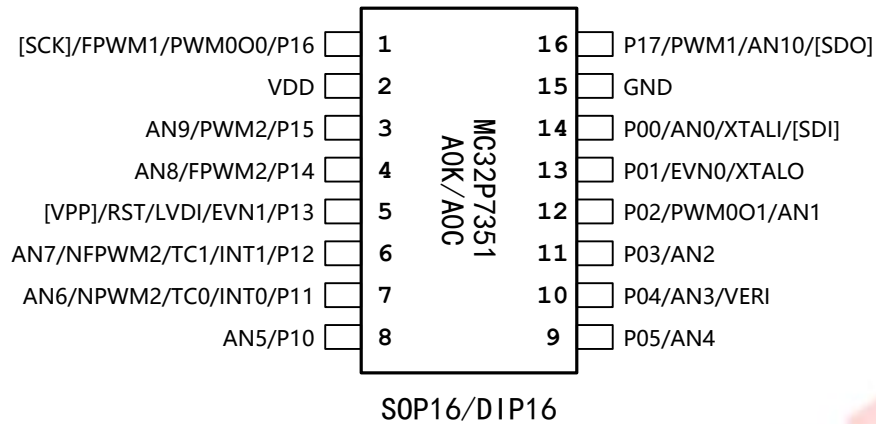
- ◇ 8 位定时器 T3
- 1 个频率可编程 RC 振荡器 PFRC
 - ◇ 振荡频率初始值: 32MHz
 - ◇ 最大调节范围 (以实际芯片为准): (32MHz-10%) ~ (32MHz+4%)
 - ◇ 输出时钟 F_{PFRC} 仅可用作定时器时钟源
- 1 个 12 位高精度 SAR 型 ADC
 - ◇ 11 路外部通道: AN0~AN10; 4 路内部通道: GND、VDD/4、EVN0/4、EVN1/4
 - ◇ 参考电压可选: VDD、内部参考电压 V_{IR} (2V/3V/4V)、外部参考电压 V_{ER} (VERI 输入)
 - ◇ ADC 时钟: F_{HRC} 的 32/64/128/256 分频
 - ◇ 支持零点校准
 - ◇ VERI 端口内部 1.8KΩ 上拉电阻可选
- 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 0.5V 比较
- 低电压复位 LVR
 - ◇ 1.8V/2.0V/2.4V/2.7V/3.6V
- 工作电压
 - ◇ VLVR27 ~ 5.5V @ F_{cpu} = 0~8MHz
 - ◇ VLVR20 ~ 5.5V @ F_{cpu} = 0~4MHz
 - ◇ VLVR18 ~ 5.5V @ F_{cpu} = 0~1MHz
- 封装形式
 - ◇ SOP16/DIP16/SOP14/DIP14/SOP8/DIP8/MSOP10

1.2 订购信息

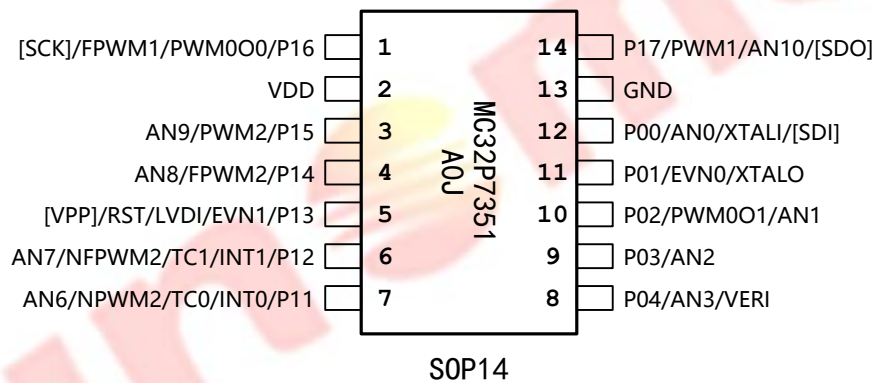
产品名称	封装形式	备注
MC32P7351A0K	SOP16	
MC32P7351A0C	DIP16	
MC32P7351A0J	SOP14	
MC32P7351A1J	SOP14	
MC32P7351A1B	DIP14	
MC32P7351A0H	SOP8	
MC32P7351A0A	DIP8	
MC32P7351A0I	MSOP10	

1.3 引脚排列

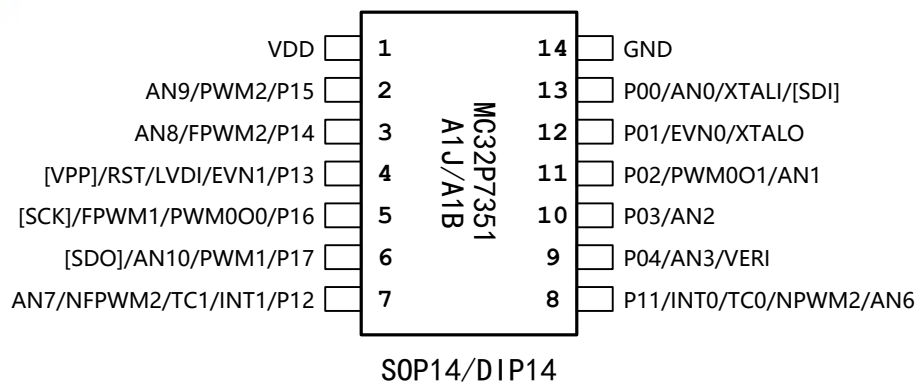
MC32P7351A0K/A0C



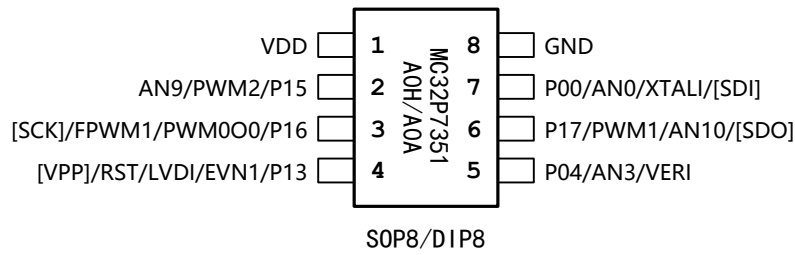
MC32P7351A0J



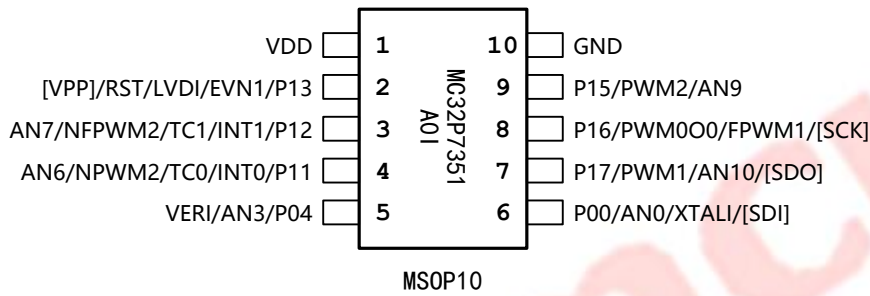
MC32P7351A1J/A1B



MC32P7351A0H/A0A



MC32P7351A0I



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 P00,P01), P1(除 P13)	D	GPIO (推挽输出), 内部上/下拉
P00, P01, P13	D	GPIO (推挽输出或开漏输出), 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
PWM0O0~PWM0O1	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 的互补 PWM 输出
PWM2	DO	定时器 T2 的 PWM/BUZ 输出
FPWM2	DO	定时器 T2 的互补 PWM 输出
NPWM2	DO	定时器 T2 的反向 PWM 输出
NFPWM2	DO	定时器 T2 的反向互补 PWM 输出
AN0~AN10	AI	ADC 外部输入通道
EVN0~EVN1	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-电源端口; D-数字端口, DI-数字输入, DO-数字输出; A-模拟端口, AI-模拟输入, AO-模拟输出。

2 电气特性

2.1 极限参数

参数	符号	值	单位
电源电压	VDD	-0.3~6.0	V
I/O 输入电压	V _{in}	-0.3~VDD+0.3	V
工作温度	T _a	-40~85	°C
储存温度	T _{stg}	-65~150	°C
流入 VDD 最大电流 (除 P16,P17)	IVDDmax	50	mA
流出 GND 最大电流 (除 P00,P16,P17)	IGNDmax	50	mA
流过 P00 最大电流	IP00max	50	mA
流过 P16,P17 最大电流	IP16P17max	120	mA

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

2.2 直流电气特性

VDD=5V, T=25°C

特性	符号	端口	条件	最小	典型	最大	单位
工作电压	VDD	VDD	F _{cpu} =8MHz@F _{HIRC} /4	V _{LVR27}		5.5	V
			F _{cpu} =4MHz@F _{HIRC} /8	V _{LVR20}		5.5	
			F _{cpu} =2MHz@F _{HIRC} /16	V _{LVR20}		5.5	
			F _{cpu} =1MHz@F _{HIRC} /32	V _{LVR18}		5.5	
			F _{cpu} =16KHz@F _{LIRC} /2	V _{LVR18}		5.5	
			F _{cpu} =16384Hz@F _{XTAL} /2	V _{LVR18}		5.5	
定时器 工作电压	V _{TMR}	VDD	定时器计数时钟频率为 64MHz	V _{LVR24}			V
			定时器计数时钟频率为 32MHz	V _{LVR20}			
			定时器计数时钟频率为 16MHz	V _{LVR18}			
输入漏电流	I _{leak}	所有输入脚	VDD=5V	-1		1	μA
输入高电平	V _{ih}	所有输入脚	SMTVS 配置	0.8VDD			V
			SMTVS 配置	2.0			
输入低电平	V _{il}	所有输入脚	SMTVS 配置			0.2VDD	V
			SMTVS 配置			0.8	
上拉电阻	R _{pu1}	P0, P1	VDD=5V, V _{in} =0		50		KΩ
	R _{pu2}	VER1	VDD=2V~5V, V _{in} =0, RESS0=1		1.8		KΩ
下拉电阻	R _{pd}	P0, P1	V _{in} =VDD=5V		50		KΩ

输出源电流	Ioh1	推挽输出脚 (除 P16,P17)	Voh=VDD-0.6V, CDRV=0		20		mA
			Voh=VDD-0.6V, CDRV=1	4	8		mA
	Ioh2	P16, P17	Voh=VDD-0.6V, PDRV=00		25		mA
			Voh=VDD-0.6V, PDRV=01		50		mA
			Voh=VDD-0.6V, PDRV=10		75		mA
			Voh=VDD-0.6V, PDRV=11		100		mA
输出灌电流	Iol1	所有输出脚(除 P00,P16,P17)	Vol=0.6V, CDRV=0		30		mA
			Vol=0.6V, CDRV=1	8	16		mA
	Iol2	P16, P17	Vol=0.6V, PDRV=00		25		mA
			Vol=0.6V, PDRV=01		50		mA
			Vol=0.6V, PDRV=10		75		mA
			Vol=0.6V, PDRV=11		100		mA
	Iol3	P00	Vol=0.6V, LDRV=0		16		mA
			Vol=0.6V, LDRV=1		40		mA
输出驱动管 开关速度		推挽输出脚 (除 P16,P17)	高速输出@空载			16	MHz
			低速输出@空载			2	MHz
		P16, P17	带载 1000pF			4	MHz
运行模式功耗	Irun	VDD	Fcpu=8MHz@HIRC		3.5		mA
			Fcpu=4MHz@HIRC		2.2		mA
			Fcpu=2MHz@HIRC		1.5		mA
			Fcpu=1MHz@HIRC		1.2		mA
			Fcpu=500KHz@HIRC		1.0		mA
			Fcpu=32KHz/2@LIRC		9		μA
			Fcpu=32768KHz/2@XTAL		10		μA
HOLD1 功耗	Ihold1	VDD	CPU 停, HIRC/LIRC 开		700		μA
HOLD2 功耗	Ihold2	VDD	CPU 停, HIRC 关, LIRC 开		2.5		μA
休眠模式功耗	Istop	VDD	休眠模式, WDT/LVR/LVD 关		0.3	1	μA
			休眠模式, WDT 开, LVR/LVD 关		2.5	5	μA
			休眠模式, WDT/LVR 关, LVD 开		10	20	μA
			休眠模式, WDT/LVD 关, LVR 开		10	20	μA
低压检测电压	VLVD	VDD	LVDVS=0001	-10%	0.5	+10%	V
			LVDVS=其他	-10%		+10%	
LVD 响应时间	TLVD				0.2	2	ms
低压复位电压	VLVR	VDD	LVRVS 配置	-5%		+5%	V
LVD/LVR 回滞电压		VDD			6%	12%	

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

2.3 交流电气特性

特性	符号	条件	最小	典型	最大	单位
HIRC 振荡频率	FHIRC	VDD=5V, T=25°C	-1.5%	32	+1.5%	MHz
		VDD=2.0V~5.5V, T=-20°C~70°C	-3%		+3%	
		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
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
XTALI 内部对地电容	CG			10		pF
XTALO 内部对地电容	CD			10		pF

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=-40°C~85°C	2.5		5.5	V
积分非线性误差	INL	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =27μs			±4	LSB
微分非线性误差	DNL	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =27μs			±2	LSB
零点偏移误差	EZ	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =27μs			±4	LSB
增益误差	ET	V _{REF} =VDD, F _{ADC} =1MHz, T _{con} =27μs			±4	LSB
转换时钟	F _{ADC}	VDD=5V			1	MHz
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	μA
ADC 动态电流	I _{ADD}	VDD=5V, AD 转换中		1	3	mA
ADC 静态电流	I _{ADS}	VDD=5V, ADC 关闭		0.1	1	μA
模拟信号源推荐阻抗	Z _{AIN}				10	KΩ
内部 1/4 分压电阻总值	REVI	V _{in} =VDD=2.5V~5.5V		24		KΩ
电阻分压比值			-1%	1/4	+1%	VEVI
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%		+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		5.5	V

2.5 PFRC 特性参数

特性	符号	条件	最小	典型	最大	单位
PFRC 振荡频率	F _{PFRC}	VDD=5V, T=25°C	-1.5%	32	+1.5%	MHz
		VDD=2.0V~5.5V, T=-20°C~70°C	-3%		+3%	
		VDD=2.0V~5.5V, T=-40°C~85°C	-5%		+5%	

3 CPU 与存储器

3.1 指令集

芯片的指令集为精简指令集。

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

指令汇总表

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

MOVRA R	将 A 存入 R	$A \rightarrow R$	1	1	-
MOVAR R	将 R 存入 A	$R \rightarrow A$	1	1	Z
MOVR R	将 R 存入 R	$R \rightarrow R$	1	1	Z
CLRA	将 A 清零	$0 \rightarrow A$	1	1	Z
CLRR R	将 R 清零	$0 \rightarrow R$	1	1	Z
INCA	A 自加 1	$A+1 \rightarrow A$	1	1	-
INCR R	R 自加 1	$R+1 \rightarrow R$	1	1	Z
INCAR R	R 加 1, 结果存入 A	$R+1 \rightarrow A$	1	1	Z
DECA	A 自减 1	$A-1 \rightarrow A$	1	1	-
DECR R	R 自减 1	$R-1 \rightarrow R$	1	1	Z
DECAR R	R 减 1, 结果存入 A	$R-1 \rightarrow A$	1	1	Z
JZA	A 自加 1: 结果为 0 则跳过下一条指令	$A+1 \rightarrow A$: 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
JZR R	R 自加 1: 结果为 0 则跳过下一条指令	$R+1 \rightarrow R$: 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
JZAR R	R 加 1, 结果存入 A: 结果为 0 则跳过下一条指令	$R+1 \rightarrow A$: 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
DJZA	A 自减 1: 结果为 0 则跳过下一条指令	$A-1 \rightarrow A$: 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
DJZR R	R 自减 1: 结果为 0 则跳过下一条指令	$R-1 \rightarrow R$: 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
DJZAR R	R 减 1, 结果存入 A: 结果为 0 则跳过下一条指令	$R-1 \rightarrow A$: 结果为 0 则 $PC+2 \rightarrow PC$	1/2	1	-
BCLR R, b	将 R 的第 b 位清 0	$0 \rightarrow R[b]$	1	1	-
BSET R, b	将 R 的第 b 位置 1	$1 \rightarrow R[b]$	1	1	-
JBCLR R, b	若 R 的第 b 位为 0, 则跳过下一条指令	若 $R[b]=0$, 则 $PC+2 \rightarrow PC$	1/2	1	-
JBSET R, b	若 R 的第 b 位为 1, 则跳过下一条指令	若 $R[b]=1$, 则 $PC+2 \rightarrow PC$	1/2	1	-
ADDAI I	I 和 A 相加, 结果存入 A	$I+A \rightarrow A$	1	1	C,DC,Z
ADCAI I	I 和 A 相加 (带 C 标志), 结果存入 A	$I+A+C \rightarrow A$	1	1	C,DC,Z
ISUBAI I	I 和 A 相减, 结果存入 A	$I-A \rightarrow A$	1	1	C,DC,Z
ISBCAI I	I 和 A 相减 (带 C 标志), 结果存入 A	$I-A-/C \rightarrow A$	1	1	C,DC,Z
ASUBAI I	A 和 I 相减, 结果存入 A	$A-I \rightarrow A$	1	1	C,DC,Z
ASBCAI I	A 和 I 相减 (带 C 标志), 结果存入 A	$A-I-/C \rightarrow A$	1	1	C,DC,Z
ANDAI I	I 和 A 与操作, 结果存入 A	$I \text{ and } A \rightarrow A$	1	1	Z
ORAI I	I 和 A 或操作, 结果存入 A	$I \text{ or } A \rightarrow A$	1	1	Z
XORAI I	I 和 A 异或操作, 结果存入 A	$I \text{ xor } A \rightarrow A$	1	1	Z
MOVAI I	将 I 存入 A	$I \rightarrow A$	1	1	-
CALL K	子程序调用	$PC+1 \rightarrow TOS; K \rightarrow PC[12:0]$	2	1	-
GOTO K	无条件跳转	$K \rightarrow PC[12:0]$	2	1	-
RETURN	从子程序返回	$TOS \rightarrow PC$	2	1	-
RETAI I	从子程序返回, 并将 I 存入 A	$TOS \rightarrow PC; I \rightarrow A$	2	1	-
RETI E	从中断返回	$TOS \rightarrow PC; 1 \rightarrow GIE$	2	1	-
NOP	空操作	空操作	1	1	-
DAA	BCD 码加法操作后, 将 A 的值调整为 BCD 码	$A(\text{HEX 码}) \rightarrow A(\text{BCD 码})$	1	1	C
DSA	BCD 码减法操作后, 将 A 的值调整为 BCD 码	$A(\text{HEX 码}) \rightarrow A(\text{BCD 码})$	1	1	-

CLRWDT	将看门狗计数器清零	0→WDT CNT	1	1	TO,PD
STOP	进入低功耗模式	0→WDT CNT; CPU 暂停	1	1	TO,PD

注:

- 1、A-算术逻辑单元累加器 ALU, R-数据存储器, I-立即数, K-程序存储器地址, TOS-堆栈栈顶;
- 2、对于条件跳转类指令, 若跳转条件成立, 则执行时间需 2 个指令周期, 否则仅需 1 个指令周期;
- 3、禁止采用对 C,DC,Z 标志有影响的指令访问寄存器 PFLAG;

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)

程序存储器支持间接寻址, 可通过寄存器 INDF3 访问地址为 (FSR1×256+FSR0) 的程序存储器内容, 高 8 位将缓存于寄存器 HIBYTE, 低 8 位将缓存于寄存器 A。

例如, 采用间接寻址读取程序存储器 0155H 地址中内容, 高 8 位存入通用数据存储器 11H 地址中, 低 8 位存入通用数据存储器 10H 地址中:

```

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) 所指地址的程序存储器中内容
                        ; 高 8 位缓存于 HIBYTE, 低 8 位缓存于 A
MOVRA    10H           ; 将 A 中缓存的低 8 位存入通用数据存储器 10H 地址中
MOVAR    HIBYTE        ; 读取 HIBYTE 中缓存的高 8 位
MOVRA    11H           ; 高 8 位存入通用数据存储器 11H 地址中

```


3.3 数据存储器

芯片的数据存储器包括通用数据存储器 GPR（256 字节）和特殊功能寄存器 SFR，地址映射如下表所示。其中 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	PFRCCR	P1KBCR	PFRCCAL
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	OSADJCR	ADRH	ADRL	ADIOS0	ADIOS1
1B8H-1BFH						PMOD	PWMCR3	PWMCR2	
1C0H-1F7H		保留							
1F8H-1FFH					PFRCAUX	HIRCCAL			

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

数据存储器寻址方式地址组成

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	寻址方式
/	/	/	/	/	/	/	取自指令的 9 位地址									直接寻址方式
/	/	/	/	/	/	/	0	FSR0								间接寻址方式 0
/	/	/	/	/	/	/	1	FSR1								间接寻址方式 1
FSR1								FSR0								间接寻址方式 2

直接寻址方式，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 0~1FFH。例如，采用直接寻址方式将数据 55H 写入数据存储器 010H 地址中：

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

间接寻址方式 0，是以 FSR0 为数据存储器地址指针，通过 INDF0 访问，寻址范围 0~0FFH。例如，采用间接寻址方式 0 将数据 55H 写入数据存储器 010H 地址中：

```
MOVAI    10H
MOVRA    FSR0
MOVAI    55H
MOVRA    INDF0          ; 将 55H 写入 FSR0 所指地址的数据存储器中
```


间接寻址方式 1，是以 FSR1 为数据存储器地址指针，通过 INDF1 访问，寻址范围 100H~1FFH。例如，采用间接寻址方式 1 将数据 55H 写入数据存储器 110H 地址中：

```
MOVAI    10H
MOVRA    FSR1
MOVAI    55H
MOVRA    INDF1          ; 将 55H 写入 (FSR1+256) 所指地址的数据存储器中
```

间接寻址方式 2，是以[FSR1:FSR0]为数据存储器地址指针，通过 INDF2 访问，寻址范围 0~FFFFH。例如，采用间接寻址方式 2 将数据 55H 写入数据存储器 0010H 地址中：

```
MOVAI    00H
MOVRA    FSR1
MOVAI    10H
MOVRA    FSR0
MOVAI    55H
MOVRA    INDF2          ; 将 55H 写入 (FSR1×256+FSR0) 所指地址的数据存储器中
```

注：间接寻址方式 2 可寻址 0~FFFFH，但访问数据存储器中未定义的地址时，读出数据不确定，写入操作可能会更改其他地址中的内容。

3.4 堆栈

芯片的堆栈为 8 级深度的硬件堆栈。当 CPU 响应中断或执行子程序调用指令时，会自动将下一条指令的 PC 值压栈保存；当 CPU 执行中断返回或子程序返回指令时，会自动将栈顶内容出栈载入 PC。

3.5 控制寄存器

数据指针寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
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：间接寻址方式 0 的指针，或间接寻址方式 2、3 的指针低 8 位。

	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: 间接寻址方式 1 的指针, 或间接寻址方式 2、3 的指针高 8 位。

间接寻址寄存器

	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 操作实际是对 FSR0 所指向地址的数据存储器进行操作, 从而实现间接寻址功能。

	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 操作实际是对 (FSR1+256) 所指向地址的数据存储器进行操作, 从而实现间接寻址功能。

	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 操作实际是对 (FSR1×256+FSR0) 所指向地址的数据存储器进行操作, 从而实现间接寻址功能。

	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 操作实际是对 (FSR1×256+FSR0) 所指向地址的程序存储器进行操作, 从而实现间接寻址功能。

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

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: 用于缓存通过 INDF3 访问程序存储器时所读取内容的高 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: $PC = PC + 1$;
- ✧ 程序跳转指令 GOTO/CALL: $PC = \text{指令码低 13 位}$;
- ✧ 返回指令 RETIE/RETURN/RETAI: $PC = \text{堆栈栈顶 (TOS)}$;

对 PCL 操作指令:

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

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** – 半字节进位/借位标志位

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

BIT[0] **C** – 进位/借位标志位

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

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 CLRWDT/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 CLRWDT 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
1X	电平变化触发

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
1X	电平变化触发

3.6 用户配置字

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

芯片的用户配置字，定义如下：

符号	功能说明
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	系统低频时钟选择： FLOSC=内部低频 RC 振荡器时钟 FLIRC； FLOSC=外部 32768Hz 晶体振荡器时钟 FXTAL；（支持 32768Hz 晶振）
RSTEN	RST 外部复位端口设置： P13 为外部复位脚； P13 为输入/输出脚；
LVRMD	LVR 模式设置： LVR 始终开启； LVR 在运行模式下开启，在低功耗模式下关闭；
LVRVS	LVR 复位电压选择：（LVR 电压应满足由 FCPU 决定的工作电压特性） 1.8V； 2.0V； 2.4V； 2.7V； 3.6V；
WDTM	WDT 模式设置： WDT 始终关闭； WDT 在运行模式下开启，在低功耗模式下关闭； WDT 始终开启；
WDTT	WDT 溢出时间选择： 16ms； 64ms； 256ms； 1024ms； 2048ms； 4096ms；
SMTVS	端口施密特阈值选择： 2.0V/0.8V； 0.8VDD/0.2VDD；
SPDS	端口输出驱动管开关速度选择（不包括 P16/P17）： 高速输出； 低速输出；
ENCR	程序代码加密设置： 程序代码加密； 程序代码不加密；

4 系统时钟

芯片内部电路均在系统高频时钟 F_{HOSC} 或系统低频时钟 F_{LOSC} 下工作，系统及部分外设模块的时钟源还可在 F_{HOSC} 和 F_{LOSC} 之间切换。

系统高频时钟 F_{HOSC} 固定为内部高频 RC 振荡器 $HIRC$ (32MHz) 时钟 F_{HIRC} ；

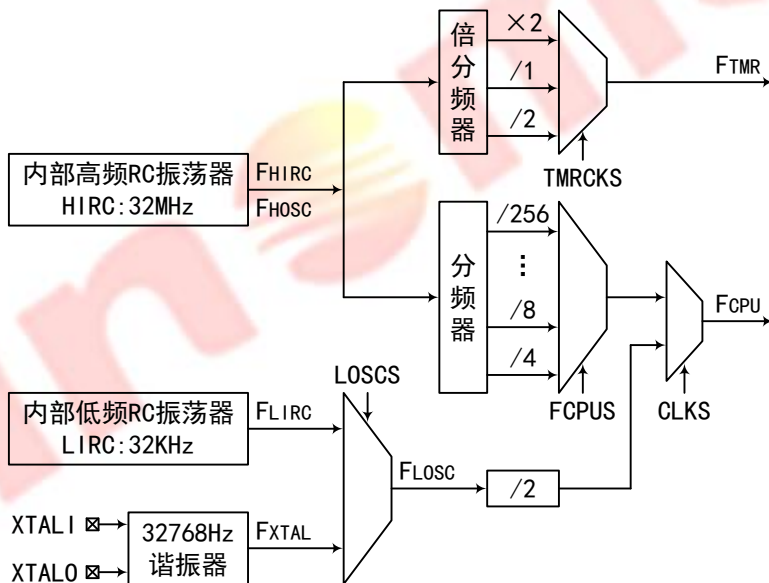
系统低频时钟 F_{LOSC} 可通过配置字 $LOSCS$ 选择以下时钟：

- ✧ 内部低频 RC 振荡器 $LIRC$ (32KHz) 时钟 F_{LIRC} ；
- ✧ 外部晶体振荡器 $XTAL$ (32768Hz) 时钟 F_{XTAL} ；

CPU 的时钟源可在系统高频时钟 F_{HOSC} 和系统低频时钟 F_{LOSC} 之间切换。 F_{HOSC} 下 CPU 的时钟频率 F_{CPU} 通过配置字 $FCPUS$ 选择； F_{LOSC} 下 F_{CPU} 则固定为 F_{LOSC} 的 2 分频。

WDT（看门狗）电路的时钟源固定为内部低频 RC 振荡器 $LIRC$ 。

系统时钟示意图



4.1 内部高频 RC 振荡器

芯片内置 1 个振荡频率为 32MHz 的高精度 $HIRC$ 振荡器，可用作系统高频时钟源。

$HIRC$ 频率可微调校准，校准值保存在 8 位 $HIRC$ 微调校准寄存器 $HIRCCAL$ 中，芯片复位后，寄存器自动加载出厂设置值作为初始值，将 $HIRC$ 频率调整至 32MHz。该寄存器可通过软件进行微调，以获得一定范围内的其他频率。

HIRC 微调校准寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
HIRCCAL	HIRCCAL7	HIRCCAL6	HIRCCAL5	HIRCCAL4	HIRCCAL3	HIRCCAL2	HIRCCAL1	HIRCCAL0
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] HIRCCAL[7:0] – HIRC 频率微调校准位（复位初始值为出厂设置值）

注:

- 1、HIRC 最大调节范围为 (32MHz -10%) ~ (32MHz +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 时推荐从初值 (32MHz) 往低频调节，以免影响正常的工作电压范围；
- 2、校准位 1 个 LSB 所调节的频率变化是非线性的，最大约为 0.5%；
- 3、HIRCCAL 中已有出厂校准值，软件微调前应备份原值以便恢复；

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、看门狗定时器（WDT）等电路。

4.3 外部晶体振荡器

芯片支持外接 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 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)
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。

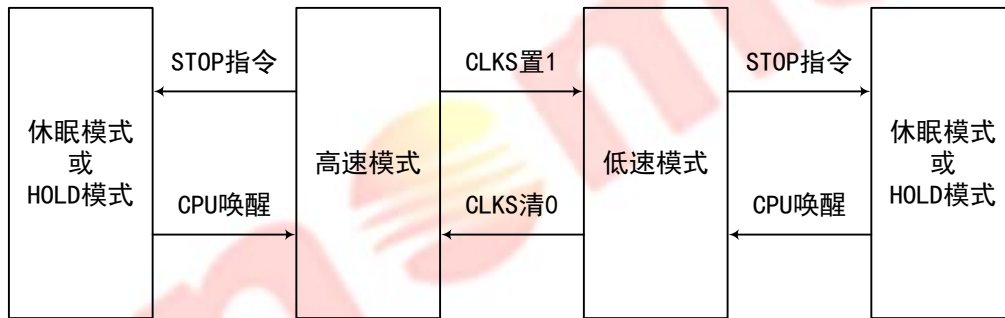
4.4 系统工作模式

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。

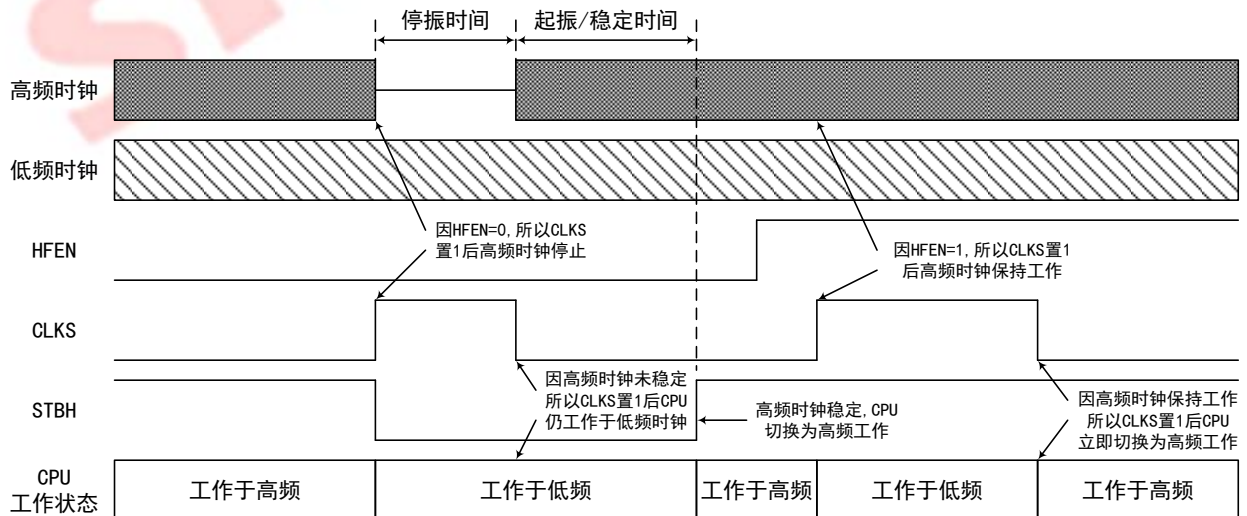
工作模式	模式切换条件	系统工作状态
高速	任意模式下，系统复位	CPU 高速运行，高/低频时钟源均工作
	低速模式下，CLKS 清 0	
	HOLD1/HOLD2/休眠模式下，CPU 唤醒 (@CLKS=0)	
低速	高速模式下，CLKS 置 1	CPU 低速运行，低频时钟源工作，高频时钟源由使能位 HFEN 决定
	HOLD1/HOLD2/休眠模式下，CPU 唤醒 (@CLKS=1)	
HOLD1	高/低速模式下，执行 STOP 指令 (@HFEN=1)	CPU 暂停，高频时钟源工作，低频时钟源由使能位 LFEN 决定
HOLD2	高/低速模式下，执行 STOP 指令 (@HFEN=0,LFEN=1)	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，执行 STOP 指令 (@HFEN=0,LFEN=0)	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



振荡器模式寄存器

	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, 低频时钟源暂停工作;
- 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, 高频时钟源暂停工作;
- 1: 高频时钟源始终工作;

4.5 低功耗模式

芯片的高速模式、低速模式为运行模式, 而休眠模式、HOLD1 模式、HOLD2 模式则为低功耗模式。

执行 STOP 指令可使系统进入低功耗模式, 同时对系统会产生以下影响:

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

以下情况可使系统退出低功耗模式:

- ✧ 芯片复位;
- ✧ WDT 溢出 (若低功耗模式下 WDT 及其时钟源保持继续工作);
- ✧ 外部中断请求发生 (若有外部中断功能并有效);

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

注：

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

5 复位

5.1 复位条件

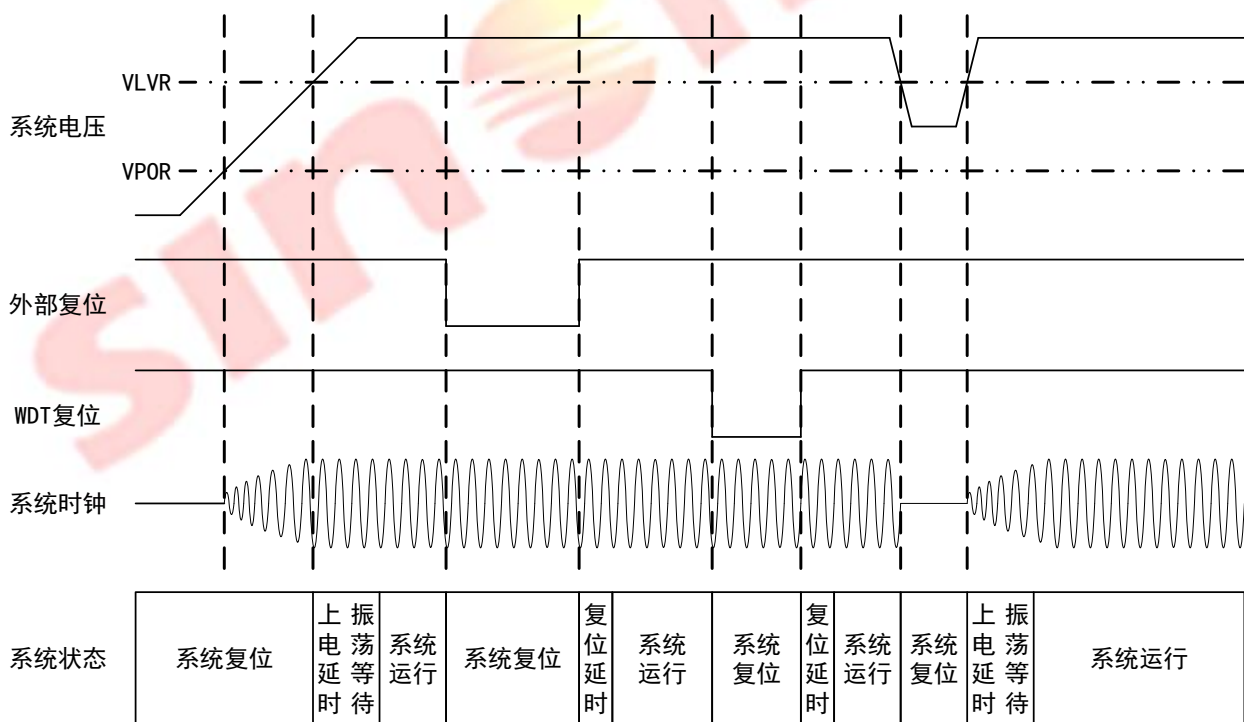
芯片共有如下几种复位方式：

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

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

上电复位 POR 和低电压复位 LVR 会关闭系统主时钟振荡器，复位解除后才重新开启振荡器，因为振荡器起振和稳定需要一定的时间，所以系统将保持一定时间的上电延时（典型值为 16ms）以待振荡器稳定振荡后才开始工作；而外部复位、WDT 复位则不会关闭主时钟振荡器，复位解除后系统将在较短的复位延时后即开始工作。

下图是复位产生和系统工作状态之间的时序关系示意图：



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

5.2 上电复位

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

上电复位过程主要包括以下几个步骤：

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

5.3 外部复位

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

5.4 低电压复位

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

5.5 看门狗复位

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

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

6 I/O 端口

6.1 通用 I/O 功能

芯片的输入/输出端口包括一组 6 位端口 P0 和一组 8 位端口 P1。所有端口均支持施密特输入，均支持推挽输出，P00/P01/P13 还可选开漏输出。除用作通用数字 I/O 端口外，部分端口还可复用为外部中断输入、PWM 输出、或 ADC 模拟输入等功能。

注：开漏输出口输出高电平时，若端口电压低于 VDD 电压则将产生额外的漏电流。

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位值；

	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位值；

6.2 内部上/下拉电阻

所有端口均具有内部上拉和下拉电阻，且均可单独控制其上/下拉电阻在端口处于输入状态时是否有效。端口处于输出状态时，上/下拉电阻及其控制位无效（但端口处于开漏输出高电平状态时上拉电阻及其控制位依然有效）。

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
ADIOS0	P12DC	P11DC	P10DC	P05DC	P04DC	P03DC	P02DC	P00DC
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] **P1nDC** – P1n 端口数字功能控制位（n=2-0）

0: 使能端口的数字 I/O 功能；

1: 关闭端口的数字 I/O 功能；

BIT[4:0] **P0nDC** – P0n 端口数字功能控制位（n=5-2,0）

0: 使能端口的数字 I/O 功能；

1: 关闭端口的数字 I/O 功能；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADIOS1	-	-	-	-	-	-	P15DC	P14DC
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

BIT[1:0] **P1nDC** – P1n 端口数字功能控制位（n=5-4）

0: 使能端口的数字 I/O 功能；

1: 关闭端口的数字 I/O 功能；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PMOD	P17DC	P13DC	P01DC	-	-	P13OM	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] **P17DC** – P17 端口数字功能控制位

0: 使能端口的数字 I/O 功能；

1: 关闭端口的数字 I/O 功能；

BIT[6] **P13DC** – P13 端口数字功能控制位

0: 使能端口的数字 I/O 功能；

1: 关闭端口的数字 I/O 功能；

BIT[5] **P01DC** – P01 端口数字功能控制位

0: 使能端口的数字 I/O 功能；

1: 关闭端口的数字 I/O 功能；

BIT[2] **P13OM** – P13 端口输出模式选择位
0: 端口输出时为开漏输出;
1: 端口输出时为推挽输出;

BIT[1] **P01OM** – P01 端口输出模式选择位
0: 端口输出时为开漏输出;
1: 端口输出时为推挽输出;

BIT[0] **P00OM** – P00 端口输出模式选择位
0: 端口输出时为开漏输出;
1: 端口输出时为推挽输出;

6.4 端口驱动控制

P16/P17 为大电流端口且输出电流 4 级可选，其他普通端口输出电流 2 级可选，其中 P00 支持灌电流大电流输出。

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	CDRV	LDRV	PDRV1	PDRV0	PDLY1	PDLY0	PSPD1	PSPD0
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] **CDRV** – 除 P16/P17 外其他普通端口输出驱动电流选择位
0: 增强驱动电流输出（典型值：Ioh=20mA, Iol=30mA）;
1: 普通驱动电流输出（典型值：Ioh=8mA, Iol=16mA）;

BIT[6] **LDRV** – P00 端口输出灌电流选择位
0: 普通灌电流输出（典型值：Iol=16mA）;
1: 增强灌电流输出（典型值：Iol=40mA）;

注：P00 端口的输出源电流由 CDRV 决定，输出灌电流则由 LDRV 决定。

BIT[5:4] **PDRV[1:0]** – P16/P17 端口输出驱动电流选择位

PDRV[1:0]	P16/P17 输出驱动电流
00	25mA
01	50mA
10	75mA
11	100mA

BIT[3:2] **PDLY[1:0]** – P16/P17 端口驱动前级非叠时间选择位

PDLY[1:0]	P16/P17 驱动前级非叠时间
00	10ns
01	50ns
10	禁用
11	禁用

BIT[1:0] **PSPD[1:0]** – P16/P17 端口驱动管开关时间选择位

PSPD[1:0]	P16/P17 驱动管开关时间
00	10ns
01	50ns
10	禁用
11	禁用

7 可编程 RC 振荡器 PFRC

芯片内置 1 个频率可编程的高频 RC 振荡器 PFRC，输出时钟 F_{PFRC} 仅可用作定时器时钟源。

PFRC 振荡频率可通过 8 位微调校准寄存器 PFRCCAL 和 3 位微调辅助寄存器 PFRCAUX 进行微调校准。芯片复位后，寄存器 PFRCCAL 自动加载出厂设置值，将 F_{PFRC} 初始值调整至 $32\text{MHz} \pm 0.5\%$ ，而寄存器 PFRCAUX 初始值为 0，可将 F_{PFRC} 调整至 $\pm 0.2\%$ 以内。也可通过 PFRCCAL 和 PFRCAUX 微调以使 PFRC 生成一定范围内的其他频率。

PFRC 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PFRCCR	PFCEN	-	T2CKS3	T2CKS2	T1CKS3	T1CKS2	T0CKS3	T0CKS2
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] **PFCEN** – PFRC 使能位

0: 关闭 PFRC;

1: 开启 PFRC;

BIT[5:4] **T2CKS[3:2]** – T2 时钟源选择位

T2CKS[3:2]	T2 时钟源
00	T2 时钟源由 T2CKS[1:0]决定
01	$F_{PFRC}/2$
10	F_{PFRC}
11	$F_{PFRC} \times 2$

BIT[3:2] **T1CKS[3:2]** – T1 时钟源选择位

T1CKS[3:2]	T1 时钟源
00	T1 时钟源由 T1CKS[1:0]决定
01	$F_{PFRC}/2$
10	F_{PFRC}
11	$F_{PFRC} \times 2$

BIT[1:0] **T0CKS[3:2]** – T0 时钟源选择位

T0CKS[3:2]	T0 时钟源
00	T0 时钟源由 T0CKS[1:0]决定
01	$F_{PFRC}/2$
10	F_{PFRC}
11	$F_{PFRC} \times 2$

PFRC 微调校准寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PFRCAL	PFRCAL7	PFRCAL6	PFRCAL5	PFRCAL4	PFRCAL3	PFRCAL2	PFRCAL1	PFRCAL0
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] **PFRCAL[7:0]** – PFRC 频率微调校准位（复位初始值为出厂设置值）

PFRC 微调辅助寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PFRCAX	-	-	-	-	-	PFRCAX2	PFRCAX1	PFRCAX0
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2:0] **PFRCAX[2:0]** – PFRC 频率微调辅助校准位

注:

- 1、PFRC 开启后需延时一段时间 ($> 20\mu s$) 以待其振荡稳定, 才可正常应用;
- 2、PFRC 最大调节范围为 (32MHz -10%) ~ (32MHz +4%) (以实际芯片为准);
- 3、校准位 1 个 LSB 所调节的频率变化是非线性的, PFRCAL 最大约为 0.5%, PFRCAX 最大约为 0.2%;
- 4、PFRCAL 中已有出厂校准值, 软件微调前应备份原值以便恢复;

如图所示，定时器 T0 可实现 BUZ 功能（BUZ0）。当 BUZ0OE=1 时，端口将输出频率为 T0 溢出频率 2 分频的蜂鸣器驱动信号（需 PWM0EC=0）。

如图所示，定时器 T0 可实现 PWM 功能（PWM0），可通过寄存器位使能或关闭 PWM 功能，并控制端口是否输出 PWM 波形。PWM0 关闭时 T0PWM 信号为低电平。PWM0 使能后 T0CNT 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 T0DATA 相等时，T0PWM 变为高电平；当计数溢出时，T0PWM 变为低电平。

T0DATA 配有 1 个 8 位比较缓冲器（T0DT_buf）用于与 T0CNT 比较，PWM0 关闭时写 T0DATA 将立即载入缓冲器中，而 PWM0 使能后写 T0DATA 则将在 T0 溢出时才载入缓冲器中。若要首个 PWM 周期和占空比准确，需先写重载寄存器和比较寄存器，再使能 PWM，最后开启定时器。

T0PWM 信号的占空比计算如下：

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

PWM0 支持 8+3 模式，即在普通模式 PWM 的时钟下，以每 8 个 PWM 周期作为一个大周期，其中部分周期进行占空比延展（即延展高电平时间），这些延展周期内 T0PWM 信号将比普通周期提前 1 个计数时钟变为高电平，即 T0PWM 高电平时间延展为 (T0DATA+1)。

8+3 模式提高的是整体上的 PWM 占空比调节精度，而不是单个 PWM 周期的占空比调节精度。占空比延展控制位 PWM0D[2:0] 决定每 8 个周期内哪几个周期为普通周期，哪几个周期为延展周期。

PWM0 的 8+3 模式一个大周期的占空比计算如下（需 $0 < T0DATA < T0LOAD$ ）：

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

举例说明：

若 PWM0D[2:0]=B'000，则 8 个 PWM 周期均不进行占空比延展，一个大周期 PWM 的占空比与普通模式 PWM 的占空比一样，为 [(T0DATA) / (T0LOAD+1)]；

若 PWM0D[2:0]=B'101，则每 8 个 PWM 周期中的第 1、2、3、5、7 个周期（共 5 个周期）进行占空比延展，一个大周期的占空比为 [(T0DATA) + (5/8)] / (T0LOAD+1)；

PWM0/BUZ0 波形可通过寄存器位 PWM0CHS 选择从端口 PWM0O0 (P16) 或端口 PWM0O1 (P02) 输出，而寄存器位 PWM0EC/BUZ0OE 仅对选定的输出通道有效。

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	PWM0EC	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** – 定时器 T0 使能位

0: 关闭定时器 T0;

1: 开启定时器 T0;

BIT[6] **PWM0EC** – PWM0 使能位及端口输出控制位
0: 关闭 PWM0 功能, 并禁止端口输出脉宽调制波形;
1: 使能 PWM0 功能, 并允许端口输出脉宽调制波形;

BIT[5] **BUZ0OE** – BUZ0 端口输出使能位
0: 禁止端口输出蜂鸣器驱动波形;
1: 允许端口输出蜂鸣器驱动波形 (仅 PWM0EC=0 时有效);

BIT[4:3] **T0CKS[1:0]** – T0 时钟源选择位 (仅 T0CKS[3:2]=00 时有效)

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

注: T0 时钟源还可通过 T0CKS[3:2] 选择内部时钟 *FPFRC* (详见 *PFRC* 章节)。

BIT[2:0] **T0PRS[2:0]** – T0 时钟预分频比选择位

T0PRS[2:0]	T0 时钟预分频比
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

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 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

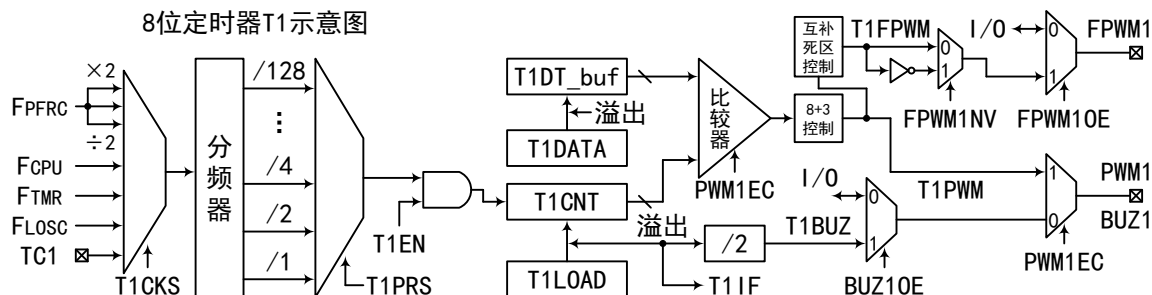
BIT[2:0] PWM0D[2:0] – T0PWM 占空比延展控制位（仅 PWM0MD=1 时有效）

PWM0D[2:0]	每 8 个周期中延展周期选择
000	0 个周期延展
001	1 个周期（第 1 个周期）延展，7 个周期（第 2,3,4,5,6,7,8 个周期）不延展
010	2 个周期（第 1,5 个周期）延展，6 个周期（第 2,3,4,6,7,8 个周期）不延展
011	3 个周期（第 1,3,5 个周期）延展，5 个周期（第 2,4,6,7,8 个周期）不延展
100	4 个周期（第 1,3,5,7 个周期）延展，4 个周期（第 2,4,6,8 个周期）不延展
101	5 个周期（第 1,2,3,5,7 个周期）延展，3 个周期（第 4,6,8 个周期）不延展
110	6 个周期（第 1,2,3,5,6,7 个周期）延展，2 个周期（第 4,8 个周期）不延展
111	7 个周期（第 1,2,3,4,5,6,7 个周期）延展，1 个周期（第 8 个周期）不延展

8.3 定时器 T1

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

- ✧ 可通过预分频器设置时钟频率，可通过重载寄存器控制计数周期；
- ✧ 支持 8+3 模式 PWM 输出，可通过比较寄存器设置 PWM 占空比；
- ✧ 可扩展为 1 对 8+3 模式的带死区互补 PWM 输出；
- ✧ 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、BUZ 功能与定时器 T0 完全相同。

定时器 T1 可实现 PWM 功能 (PWM1)，并支持 8+3 模式，其功能与定时器 T0 的 8+3 模式 PWM 功能完全相同。如图所示，PWM1 可从 T1PWM 信号衍生 1 路带死区（2 路互补信号高电平非交叠时间）控制的互补 PWM 信号 T1FPWM，从而扩展为 1 对 8+3 模式的带死区互补 PWM 输出。

注：首个 PWM 周期内，互补死区控制无效，T1FPWM 信号固定为高电平。

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	PWM1EC	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** – 定时器 T1 使能位

- 0: 关闭定时器 T1;
- 1: 开启定时器 T1;

BIT[6] **PWM1EC** – PWM1 使能位及端口输出控制位

- 0: 关闭 PWM1 功能，并禁止端口输出脉宽调制波形;
- 1: 使能 PWM1 功能，并允许端口输出脉宽调制波形;

BIT[5] **BUZ1OE** – BUZ1 端口输出使能位

- 0: 禁止端口输出蜂鸣器驱动波形;
- 1: 允许端口输出蜂鸣器驱动波形（仅 PWM1EC=0 时有效）;

BIT[4:3] **T1CKS[1:0]** – T1 时钟源选择位（仅 T1CKS[3:2]=00 时有效）

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

注：T1 时钟源还可通过 T1CKS[3:2] 选择内部时钟 FPFRC (详见 PPFRC 章节)。

BIT[2:0] T1PRS[2:0] – T1 时钟预分频比选择位

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

定时器 T1 计数器

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

BIT[7:0] T1CNI[7:0] – T1 计数器，为可读写的递减计数器

定时器 T1 重载寄存器

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

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

注：定时器重载寄存器的值禁止为 0，否则定时器将无法正常工作。

定时器 T1 比较寄存器

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

BIT[7:0] T1DATA[7:0] – T1 比较寄存器，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] – T1PWM/T1FPWM 占空比延展控制位（仅 PWM1MD=1 时有效）

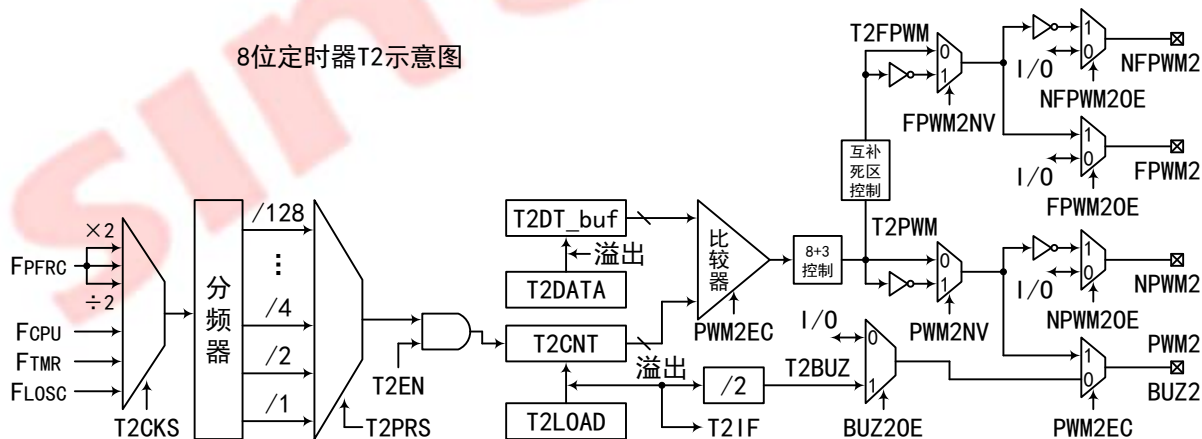
PWM1D[2:0]	每 8 个周期中延展周期选择
000	0 个周期延展
001	1 个周期（第 1 个周期）延展，7 个周期（第 2,3,4,5,6,7,8 个周期）不延展
010	2 个周期（第 1,5 个周期）延展，6 个周期（第 2,3,4,6,7,8 个周期）不延展
011	3 个周期（第 1,3,5 个周期）延展，5 个周期（第 2,4,6,7,8 个周期）不延展
100	4 个周期（第 1,3,5,7 个周期）延展，4 个周期（第 2,4,6,8 个周期）不延展
101	5 个周期（第 1,2,3,5,7 个周期）延展，3 个周期（第 4,6,8 个周期）不延展
110	6 个周期（第 1,2,3,5,6,7 个周期）延展，2 个周期（第 4,8 个周期）不延展
111	7 个周期（第 1,2,3,4,5,6,7 个周期）延展，1 个周期（第 8 个周期）不延展

8.4 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 位重载寄存器和 8 位比较寄存器。

- ◇ 可通过预分频器设置时钟频率，可通过重载寄存器控制计数周期；
- ◇ 支持 8+3 模式 PWM 输出，可通过比较寄存器设置 PWM 占空比；
- ◇ 可扩展为 2 对互反的 8+3 模式带死区互补 PWM 输出；
- ◇ 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、BUZ 功能与定时器 T0 完全相同。

定时器 T2 可实现 PWM 功能 (PWM2)，并支持 8+3 模式，其功能与定时器 T0 的 8+3 模式 PWM 功能完全相同。如图所示，PWM2 可从 T2P20E 信号衍生 1 路带死区（2 路互补信号高电平非交叠时间）控制的互补 PWM 信号 T2FP20E，并可将 T2P20E/T2FP20E 信号的反向信号从端口输出，从而扩展为 2 对互反的 8+3 模式带死区互补 PWM 输出。

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	PWM2EC	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** – 定时器 T2 使能位

0: 关闭定时器 T2;

1: 开启定时器 T2;

BIT[6] **PWM2EC** – PWM2 使能位及端口输出控制位

0: 关闭 PWM2 功能, 并禁止端口输出脉宽调制波形;

1: 使能 PWM2 功能, 并允许端口输出脉宽调制波形;

BIT[5] **BUZ2OE** – BUZ2 端口输出使能位

0: 禁止端口输出蜂鸣器驱动波形;

1: 允许端口输出蜂鸣器驱动波形 (仅 PWM2EC=0 时有效);

BIT[4:3] **T2CKS[1:0]** – T2 时钟源选择位 (仅 T2CKS[3:2]=00 时有效)

T2CKS[1:0]	T2 时钟源
00	FCPU
01	FTMR (详见 TMRCKS[1:0])
10	FLOSC
11	-

注: T2 时钟源还可通过 T2CKS[3:2] 选择内部时钟 *FPFRC* (详见 *PFRC* 章节)。

BIT[2:0] **T2PRS[2:0]** – T2 时钟预分频比选择位

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

定时器 T2 计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
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] T2CNI[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

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
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

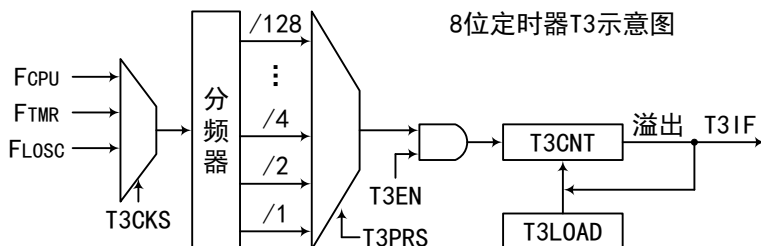
BIT[2:0] PWM2D[2:0] – T2PWM/T2FPWM 占空比延展控制位（仅 PWM2MD=1 时有效）

PWM2D[2:0]	每 8 个周期中延展周期选择
000	0 个周期延展
001	1 个周期（第 1 个周期）延展，7 个周期（第 2,3,4,5,6,7,8 个周期）不延展
010	2 个周期（第 1,5 个周期）延展，6 个周期（第 2,3,4,6,7,8 个周期）不延展
011	3 个周期（第 1,3,5 个周期）延展，5 个周期（第 2,4,6,7,8 个周期）不延展
100	4 个周期（第 1,3,5,7 个周期）延展，4 个周期（第 2,4,6,8 个周期）不延展
101	5 个周期（第 1,2,3,5,7 个周期）延展，3 个周期（第 4,6,8 个周期）不延展
110	6 个周期（第 1,2,3,5,6,7 个周期）延展，2 个周期（第 4,8 个周期）不延展
111	7 个周期（第 1,2,3,4,5,6,7 个周期）延展，1 个周期（第 8 个周期）不延展

8.5 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。

- ✧ 可通过预分频器设置时钟频率，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 T3 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** – 定时器 T3 使能位

0: 关闭定时器 T3;

1: 开启定时器 T3;

BIT[4:3] **T3CKS[1:0]** – T3 时钟源选择位

T3CKS[1:0]	T3 时钟源
00	F_CPU
01	F_TMR (详见 TMRCKS[1:0])
10	F_LOSC
11	-

BIT[2:0] **T3PRS[2:0]** – T3 时钟预分频比选择位

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

注：定时器重载寄存器的值禁止为 0，否则定时器将无法正常工作。

8.6 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: 禁止端口输出脉宽调制波形；

1: 允许端口输出脉宽调制波形（仅 PWM1EC=1 时有效）；

BIT[6] **PWM2NV** – PWM2 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] **FPWM1NV** – FPWM1 端口输出取反控制位（PWM1 端口输出电平不受影响）

0: 端口输出正向波形；

1: 端口对电平取反后输出；

BIT[4] **PWM0NV** – PWM0 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[3:2] **FPWM1R[1:0]** – T1FPWM 上升沿非交叠时间选择位

FPWM1R[1:0]	上升沿非交叠时间
00	1 个计数时钟周期
01	2 个计数时钟周期
10	3 个计数时钟周期
11	4 个计数时钟周期

BIT[1:0] **FPWM1F[1:0]** – T1FPWM 下降沿非交叠时间选择位

FPWM1F[1:0]	下降沿非交叠时间
00	2 个计数时钟周期
01	4 个计数时钟周期
10	6 个计数时钟周期
11	8 个计数时钟周期

	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 FTMR 频率选择位

TMRCKS[1:0]	FTMR 频率
0X	FHOSC/2
10	FHOSC
11	FHOSC×2

BIT[2] **PWM2MD** – PWM2 模式选择位

- 0: PWM2 工作于普通模式;
1: PWM2 工作于 8+3 模式;

BIT[1] **PWM1MD** – PWM1 模式选择位

- 0: PWM1 工作于普通模式;
1: PWM1 工作于 8+3 模式;

BIT[0] **PWM0MD** – PWM0 模式选择位

- 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	-	-	FPWM2E5	FPWM2E4	FPWM2E3	FPWM2E2	FPWM2E1	FPWM2E0
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] **FPWM2E[5:0]** – T2FPWM 上升/下降沿非交叠时间选择位

FPWM2E[5:0]	上升沿非交叠时间	下降沿非交叠时间
00 0000	1 个计数时钟周期	1 个计数时钟周期
00 0001	2 个计数时钟周期	2 个计数时钟周期
...	...	...
11 1111	64 个计数时钟周期	64 个计数时钟周期

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCR3	FPWM2NV	FPWM2OE	NPWM2OE	NFPWM2OE	-	PWM0CHS	-	-
R/W	R/W	R/W	R/W	R/W	-	R/W	-	-
初始值	0	0	0	0	-	0	-	-

BIT[7] **FPWM2NV** – FPWM2 端口输出取反控制位（PWM2 端口输出电平不受影响）

- 0: 端口输出正向波形；
- 1: 端口对电平取反后输出；

BIT[6] **FPWM2OE** – FPWM2 端口输出使能位

- 0: 禁止端口输出脉宽调制波形；
- 1: 允许端口输出脉宽调制波形（仅 PWM2EC=1 时有效）；

BIT[5] **NPWM2OE** – NPWM2 端口输出使能位

- 0: 禁止端口输出脉宽调制波形；
- 1: 允许端口输出脉宽调制波形（仅 PWM2EC=1 时有效）；

BIT[4] **NFPWM2OE** – NFPWM2 端口输出使能位

- 0: 禁止端口输出脉宽调制波形；
- 1: 允许端口输出脉宽调制波形（仅 PWM2EC=1 时有效）；

BIT[2] **PWM0CHS** – PWM0/BUZ0 输出通道选择位（PWM0/BUZ0 功能仅对被选定通道有效）

- 0: PWM0/BUZ0 输出通道为 PWM0O0（P16）；
- 1: PWM0/BUZ0 输出通道为 PWM0O1（P02）；

9 模数转换器 ADC

9.1 ADC 概述

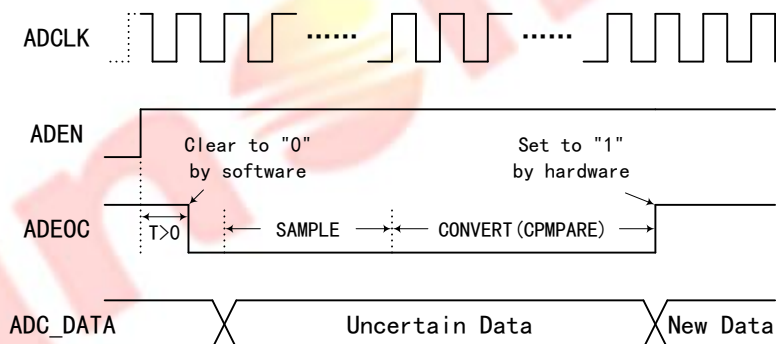
芯片内置 1 个 12 位高精度逐次逼近型的模数转换器 ADC。

- ✧ 11 路外部通道：AN0~AN10；4 路内部通道：GND、VDD/4、EVN0/4、EVN1/4；
- ✧ 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：FHIRC 的 32/64/128/256 分频；
- ✧ 支持零点校准；

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 AD 转换状态。当 ADEOC 为 1 时写 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。

ADC 的采样 (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 转换的时间为 27 个 ADCLK。

ADC 转换时序如下图所示：



注：

1. AD 转换过程中或 ADEN 未使能时，ADRH/ADRL 中的数据未知，应在 AD 转换完成且 ADEN 使能的情况下读取 AD 转换结果数据；
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} 实际电压将降为 $(VDD - 0.5V)$ ；
3. 使能 ADC 模块 (等待时间 $> 200\mu s$)、切换参考电压 (等待时间 $> 40\mu s$) 等操作后，需延时以待电路稳定后才可启动 AD 转换；
4. 因采样保持电路的电容效应，切换输入通道后的前几次转换结果将会有偏差，建议舍弃；
5. 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，比外部参考电压下略低 2 个 LSB 左右；
6. 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；

9.2 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 模拟输入通道选择位

ADCHS[3:0]	ADC 模拟输入通道
0000	AN0
0001	AN1
0010	AN2
0011	AN3
0100	AN4
0101	AN5
0110	AN6
0111	AN7
1000	AN8
1001	AN9
1010	VDD/4
1011	EVN0/4
1100	EVN1/4
1101	-
1110	GND
1111	AN10

BIT[3] **ADRSEL** – ADC 转换结果数据格式选择位

0: ADC 转换结果为 8 位数据, 存入 ADRH[7:0];

1: ADC 转换结果为 12 位数据, 高 8 位存入 ADRH[7:0]、低 4 位存入 ADRL[3:0];

BIT[1] **ADEOC** – AD 转换控制位

0: AD 转换中, 完成后自动置 1;

1: 转换未开始或已完成, 写 0 开始 AD 转换;

BIT[0] **ADEN** – ADC 使能位

0: 关闭 ADC;

1: 开启 ADC;

	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}
000~011	禁用
100	F _{HIRC} /32
101	F _{HIRC} /64
110	F _{HIRC} /128
111	F _{HIRC} /256

BIT[4] 保留位，需固定写“0”

BIT[3] **RESS0** – VERI 端口内部 1.8K Ω 上拉电阻使能位

- 0: 关闭 VERI 内部 1.8K Ω 上拉电阻;
1: 使能 VERI 内部 1.8K Ω 上拉电阻;

BIT[2:0] **ADVRS[2:0]** – ADC 参考电压选择位

ADVRS[1:0]	ADC 参考电压
000	内部 2.0V
001	内部 3.0V
010	内部 4.0V
011	VDD
100~110	禁用
111	外部 VERI 输入电压

注：应用外部参考电压时，需先关闭相应端口的数字 I/O 功能及其内部上/下拉电阻。

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
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，需固定写“1111”

ADSPS[3:0]	ADC 采样时间
0000~1110	禁用
1111	15 个 ADCLK

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]** – 8 位 ADC 转换结果，或 12 位 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]** – 12 位 ADC 转换结果低 4 位

ADC 零点偏移修调控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSADJCR	OSADJEN	OSADJTD	OSADJT5	OSADJT4	OSADJT3	OSADJT2	OSADJT1	OSADJT0
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] **OSADJEN** – ADC 零点偏移修调使能位

0: ADC 零点偏移修调无效;

1: ADC 零点偏移修调有效;

BIT[6] **OSADJTD** – ADC 零点偏移修调方向选择位

0: 负向修调，即根据修调电压减小转换值（转换结果大于理论值时应选择负向修调）;

1: 正向修调，即根据修调电压增加转换值（转换结果小于理论值时应选择正向修调）;

BIT[5:0] **OSADJT[5:0]** – ADC 零点偏移修调电压选择位

OSADJT[5:0]	修调电压（典型值）
00 0000	0
00 0001	$1 \times V_{REF}/4096$
00 0010	$2 \times V_{REF}/4096$
...	...
00 1111	$15 \times V_{REF}/4096$
01 0000	$16 \times V_{REF}/4096$
...	...
11 1110	$62 \times V_{REF}/4096$
11 1111	$63 \times V_{REF}/4096$

9.3 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，关闭端口的内部上/下拉电阻;

- (2) 通过端口数模控制寄存器，关闭相应端口的数字 I/O 功能；
- (3) 若转换时钟可选，则设置 ADCKS，选择适当的转换时钟；
- (4) 若采样时间可选，则设置 ADSPS，选择适当的采样时间；
- (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；

9.4 ADC 零点偏移修调流程

- (1) 设置 ADC 输入通道为 GND，设置 ADC 时钟、采样时间等参数，设置 OSADJEN=1；
- (2) 设置 OSADJTD=0、OSADJT=00H，进行 ADC 转换：
 - ✧ 若转换结果为 0，则执行（4）；
 - ✧ 若转换结果非 0，则执行（3）；
- (3) OSADJT 自加 1 后进行 ADC 转换：
 - ✧ 若转换结果为 0，则跳至（6）；
 - ✧ 若转换结果非 0，则循环执行（3），直到结果为 0 或 OSADJT=3FH 后，跳至（6）；
- (4) 设置 OSADJTD=1、OSADJT=3FH，进行 ADC 转换：
 - ✧ 若转换结果为 0，则跳至（6）；
 - ✧ 若转换结果非 0，则执行（5）；
- (5) OSADJT 自减 1 后进行 ADC 转换：
 - ✧ 若转换结果为 0，则跳至（6）；
 - ✧ 若转换结果非 0，则循环执行（5），直到结果为 0 或 OSADJT=00H 后，跳至（6）；
- (6) OSADJTD 及 OSADJT[5:0]的值即为零点偏移最佳修调结果，修调流程结束，后续 ADC 工作时直接应用，无需再次修调。

注：上述每一步修调中的 ADC 转换，建议转换多次，并依据转换值为 0/非 0 出现的次数判定 ADC 转换结果为 0 或非 0。例如转换 5 次，其中 3 次及以上的转换结果为 0 则判定当前 ADC 转换结果为 0，否则判定为转换结果不为 0。

10 低电压检测 LVD

芯片内置低电压检测模块 LVD，可通过寄存器位 LVDEN 开启，通过 LVDVS 选择电压检测阈值。当 VDD 电压降至电压检测阈值以下时检测状态标志位 LVDF 将被置 1；因 LVD 电路的回滞特性（回滞电压典型值为 6%），VDD 电压需恢复至电压检测阈值+6%后 LVDF 才被清 0。

注：开启 LVD 或切换电压检测阈值等操作，需待电路稳定（时间 > 2ms）后 LVD 输出才有效。

LVD 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
LVDCR	LVDEN	LVDVS3	LVDVS2	LVDVS1	LVDVS0			LVDF
R/W	R/W	R/W	R/W	R/W	R/W			R
初始值	0	0	0	0	0			0

BIT[7] **LVDEN** – 低电压检测 LVD 使能位

0： 关闭 LVD；

1： 开启 LVD；

BIT[6:3] **LVDVS[3:0]** – LVD 电压检测阈值选择位

LVDVS[3:0]	LVD 电压检测阈值
0000	1.8V
0001	LVDI 输入电压与内部 0.5V 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] **LVDF** – LVD 检测状态标志位

0： VDD 电压高于电压检测阈值，或 LVD 关闭；

1： VDD 电压低于电压检测阈值；

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

注：应用外部中断功能或键盘中断功能，需将相应端口设为输入状态。

11.1 外部中断

芯片具有 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 GIE 为 1 且相应的外部中断使能位 INTnIE（n=0-1）为 1，则产生外部中断。

11.2 定时器中断

定时器 Tn（n=0-3）在计数溢出时将触发定时器中断，中断标志 TnIF（n=0-3）将被置 1，若 GIE 为 1 且相应的定时器中断使能位 TnIE（n=0-3）为 1，则产生定时器中断。

11.3 键盘中断

芯片具有 8 路键盘中断源，均可单独使能或关闭端口的键盘中断功能。任意一路使能键盘中断功能的端口，其输入电平发生变化时均将触发键盘中断，中断标志 KBIF 将被置 1，若 GIE 为 1 且键盘中断使能位 KBIE 为 1，则产生键盘中断。

键盘中断控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
P1KBCR	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 端口键盘中断功能使能位 (n=7-0)

0: 关闭端口的键盘中断功能;

1: 使能端口的键盘中断功能;

11.4 ADC 中断

AD 转换完成时将触发 ADC 中断, 中断标志 ADIF 将被置 1, 若 GIE 为 1 且 ADC 中断使能位 ADIE 为 1, 则产生 ADC 中断。

11.5 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 0;
- 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 0;
- 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 0;
- 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 0;
- 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 0;

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

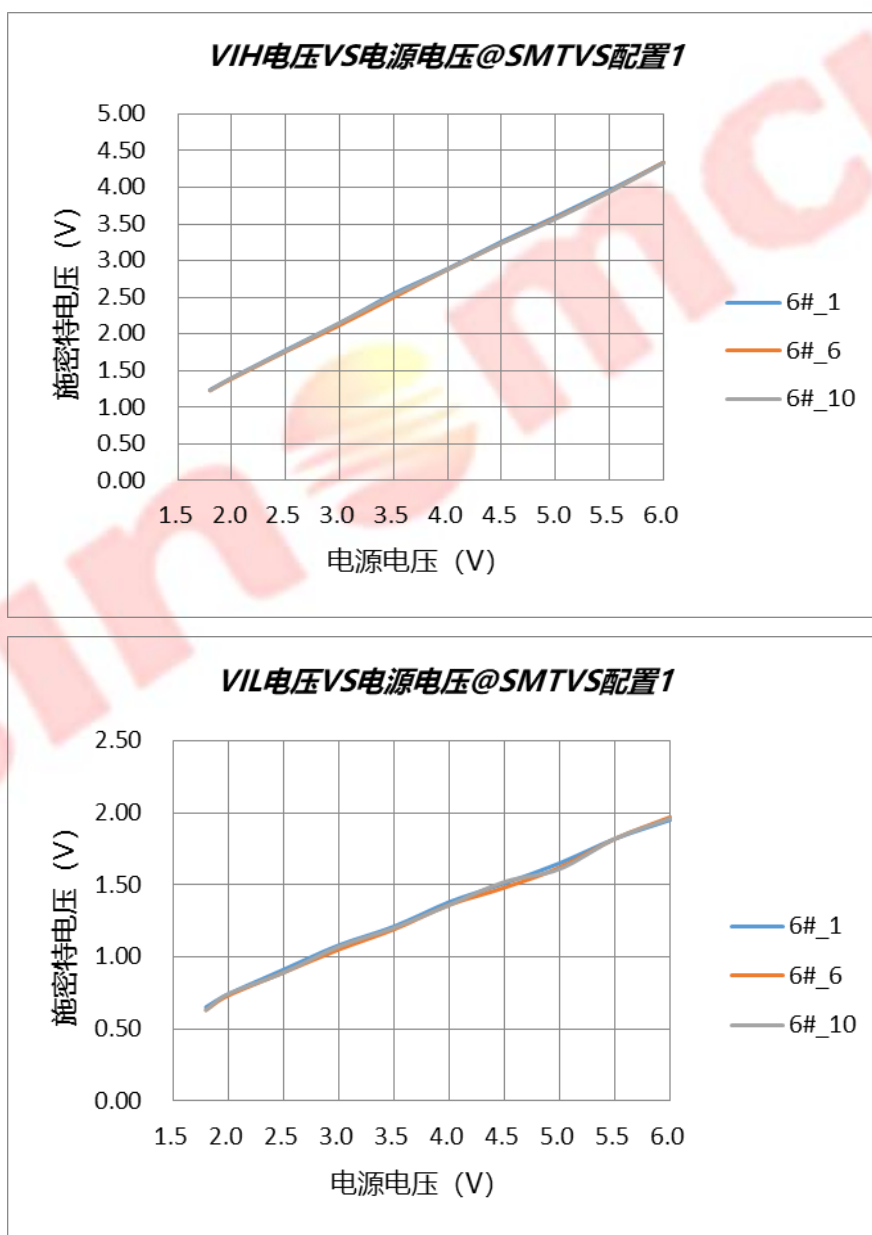
12 特性曲线

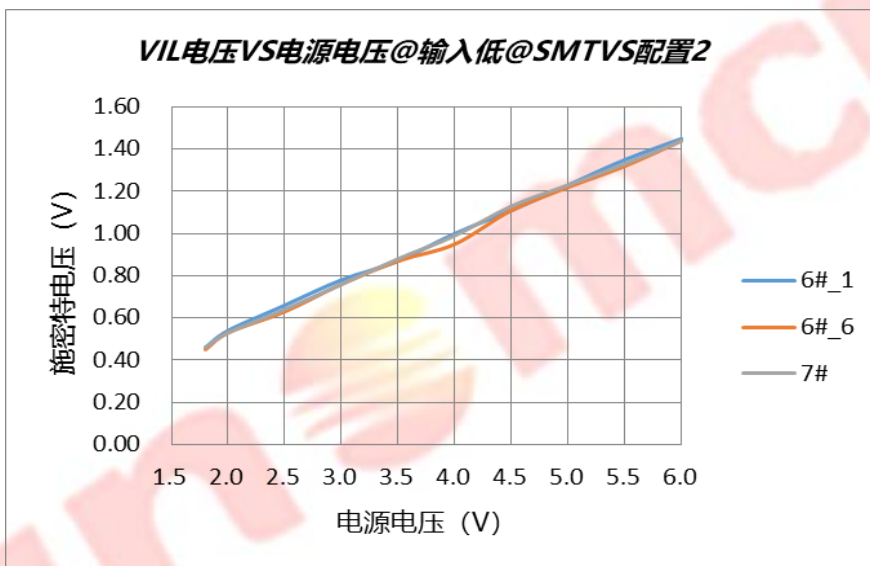
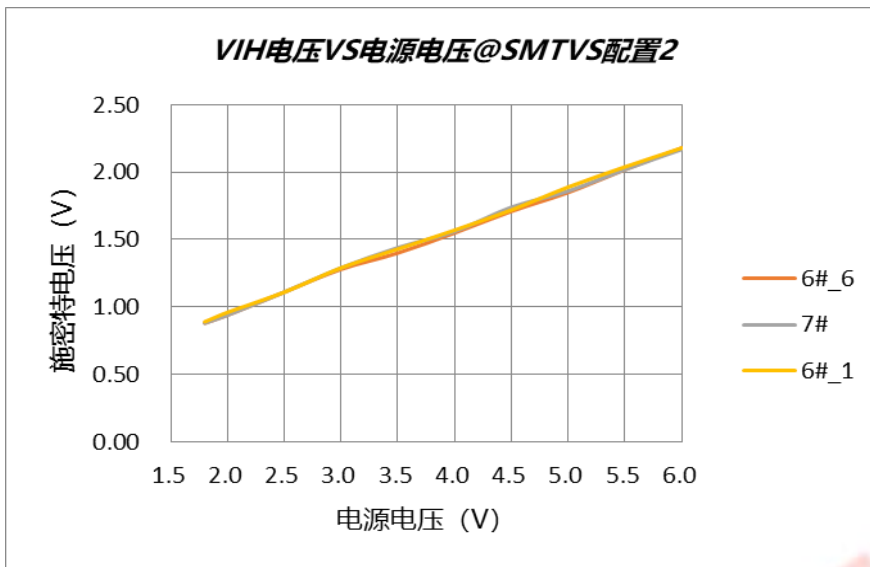
注:

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

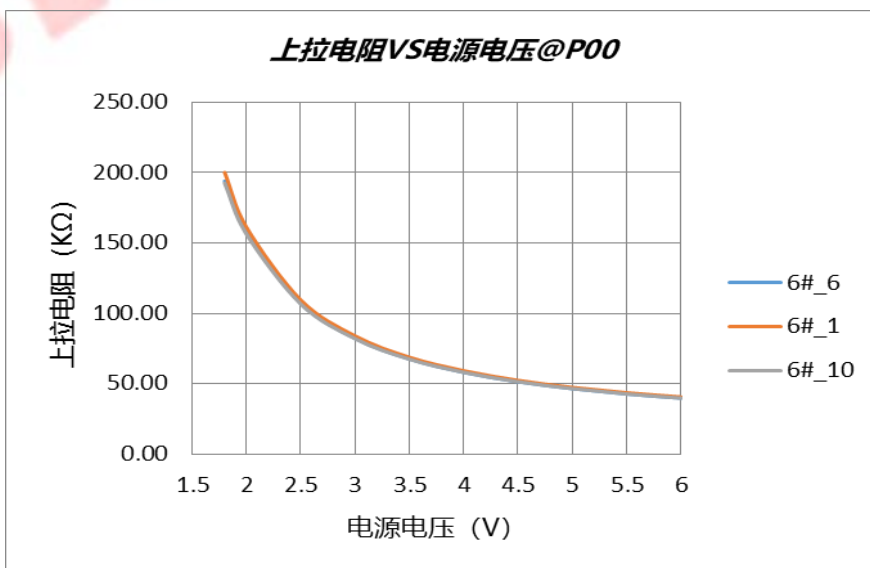
12.1 I/O 特性

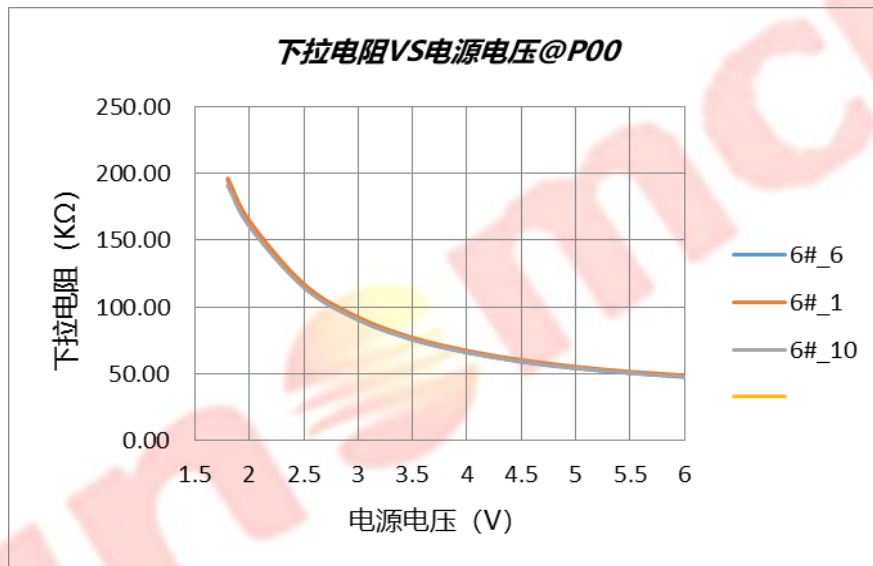
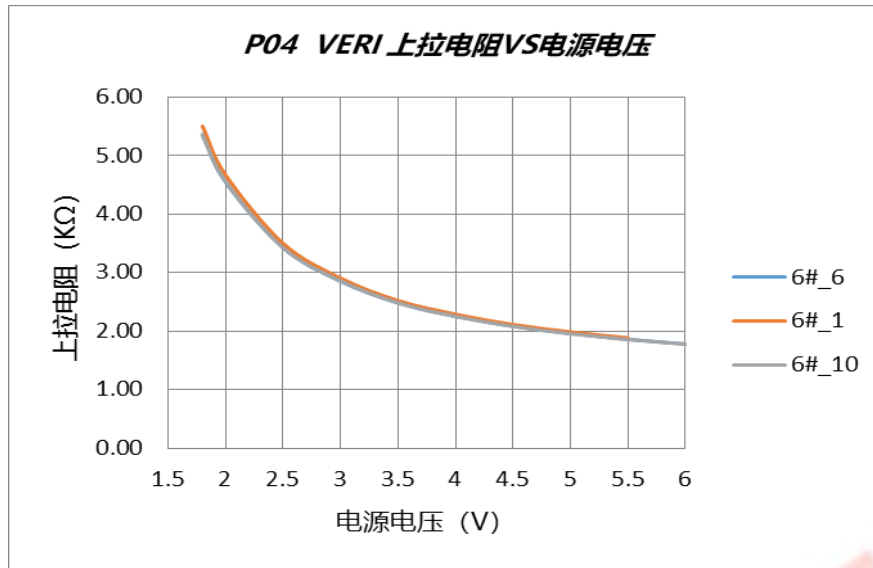
输入 SMT 阈值电压 VS 电源电压



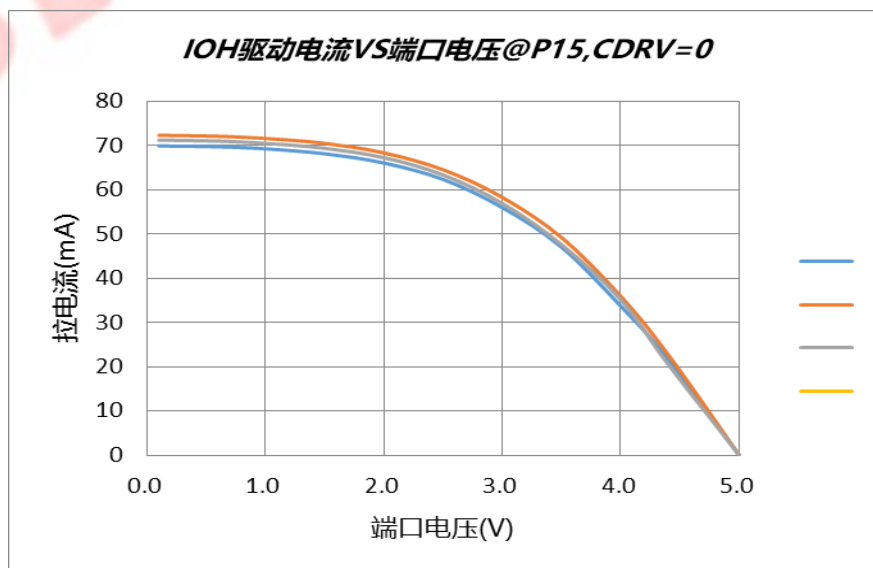


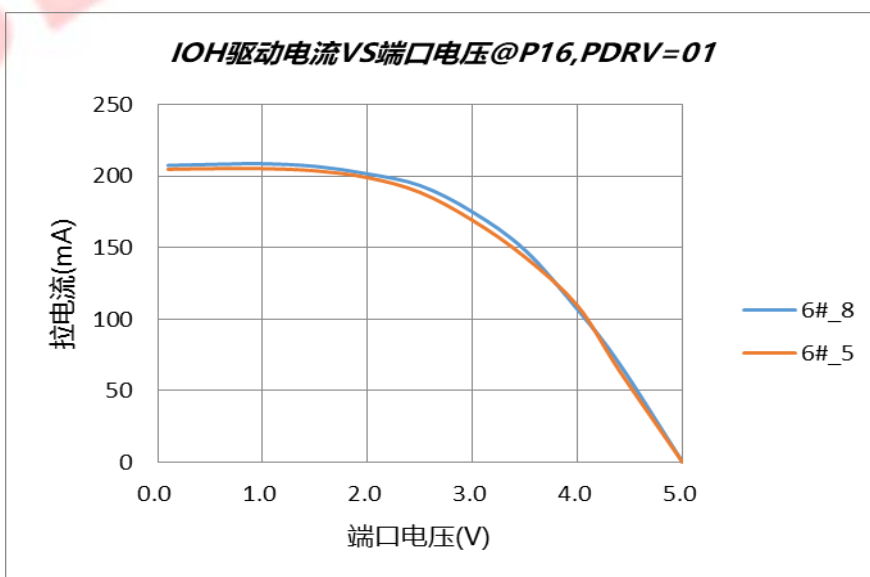
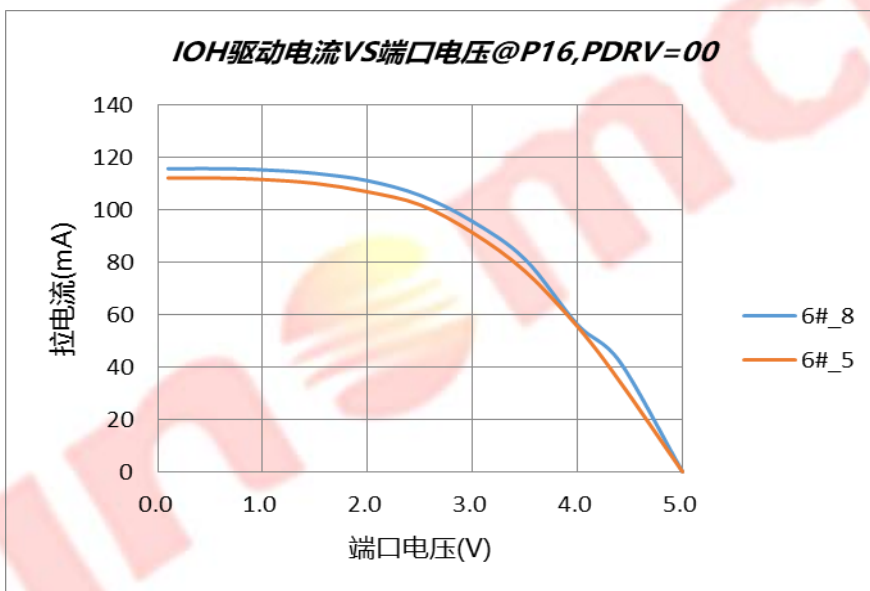
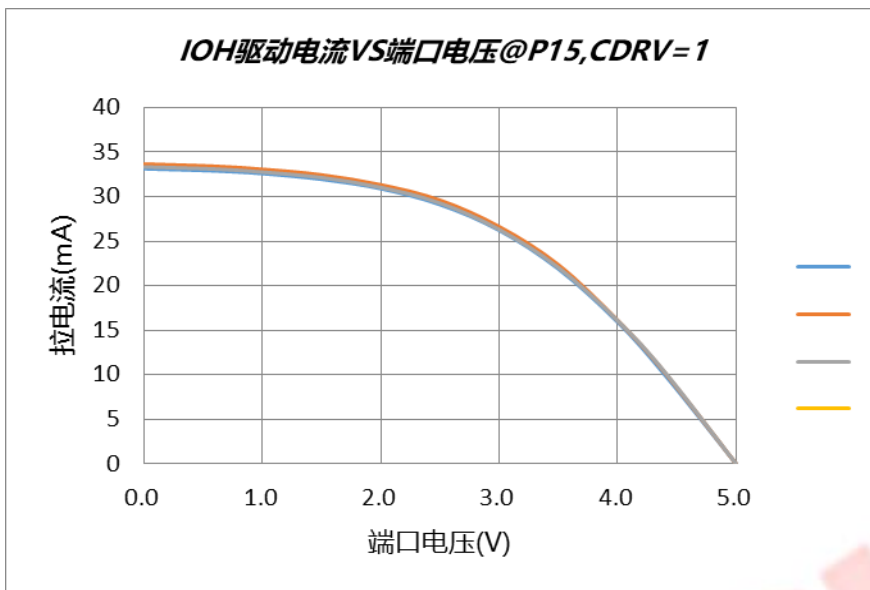
上/下拉电阻值 VS 电源电压

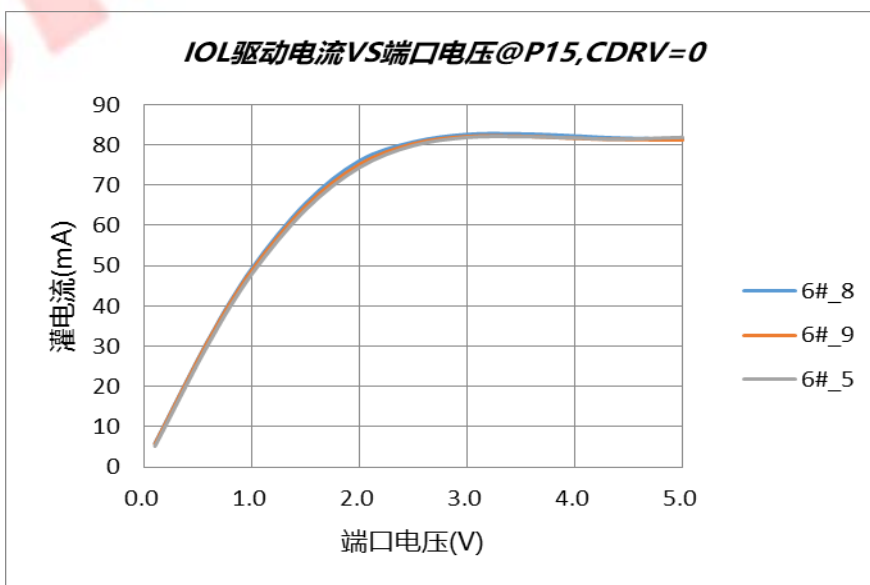
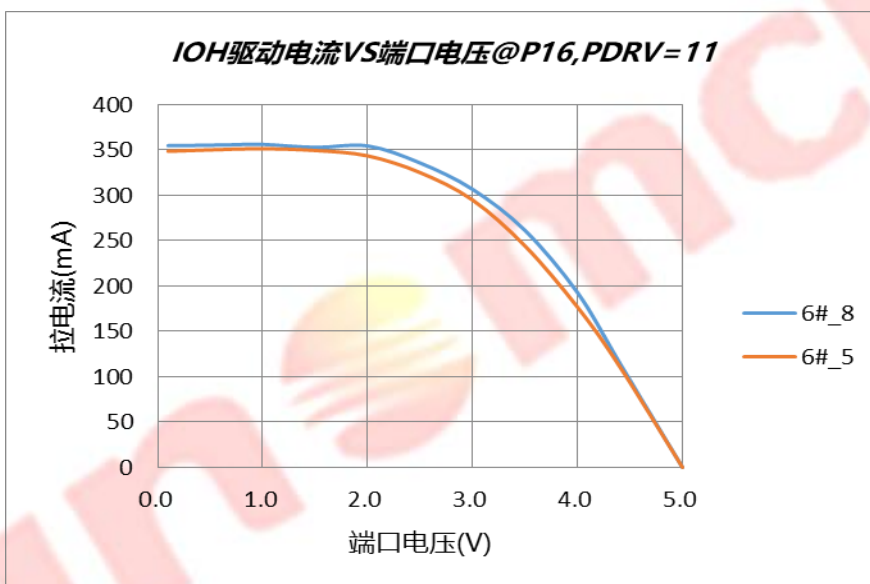
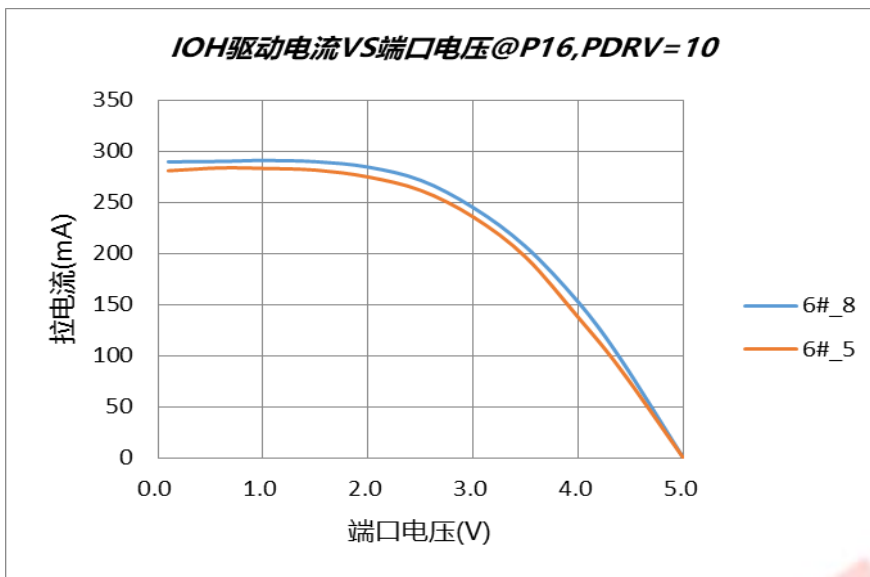


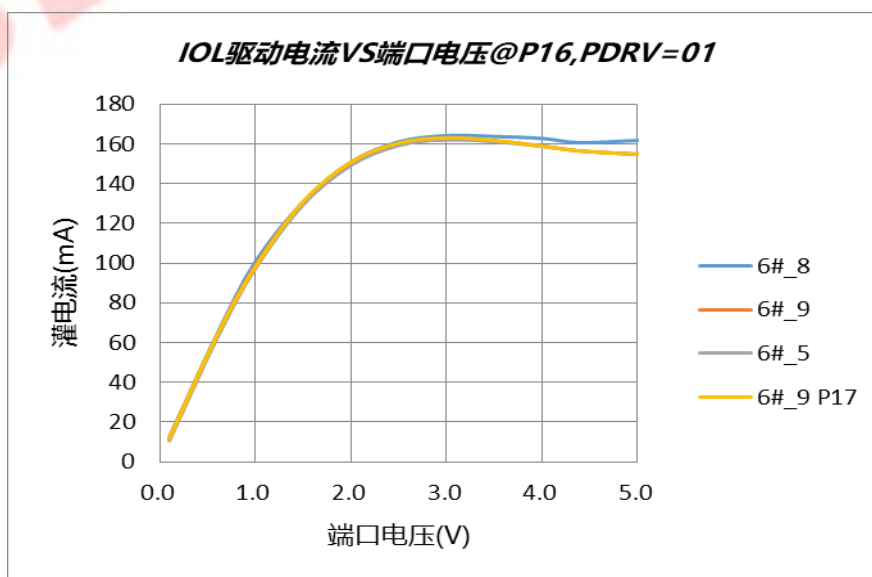
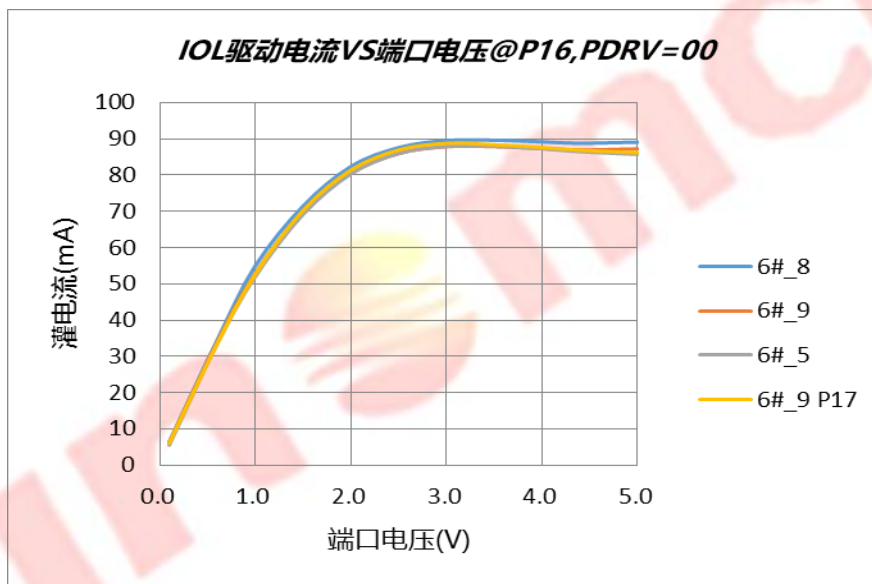
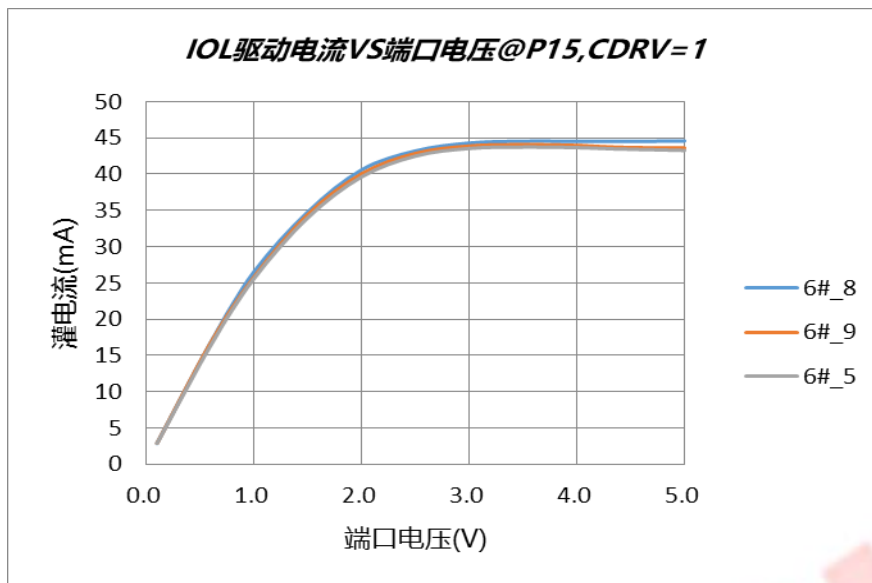


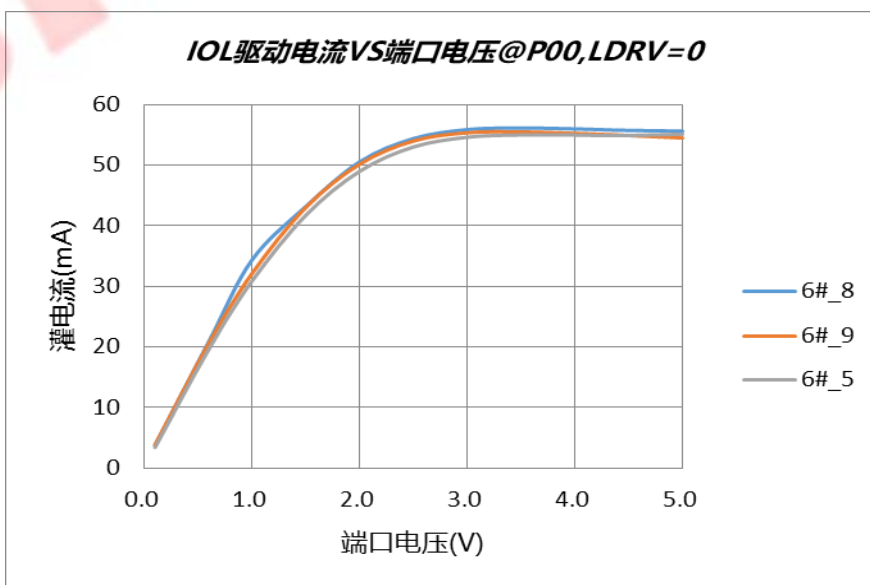
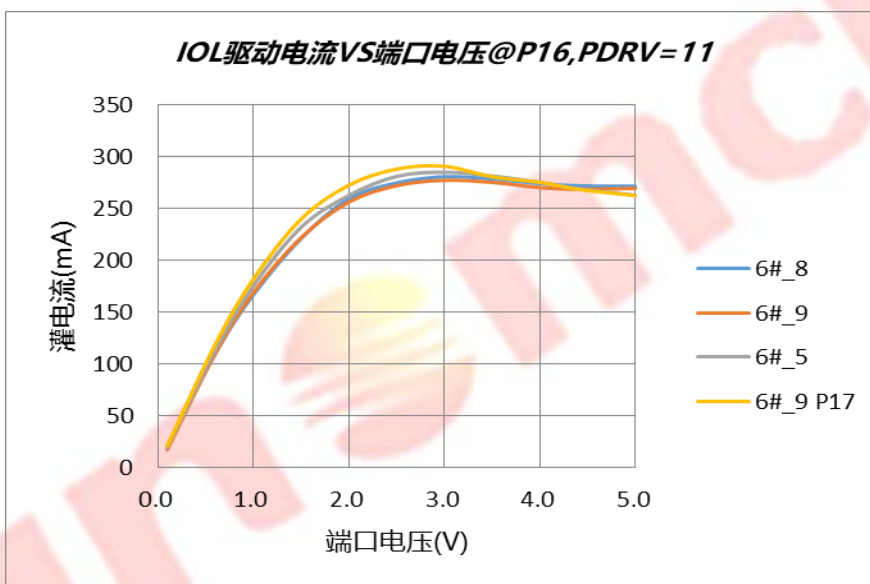
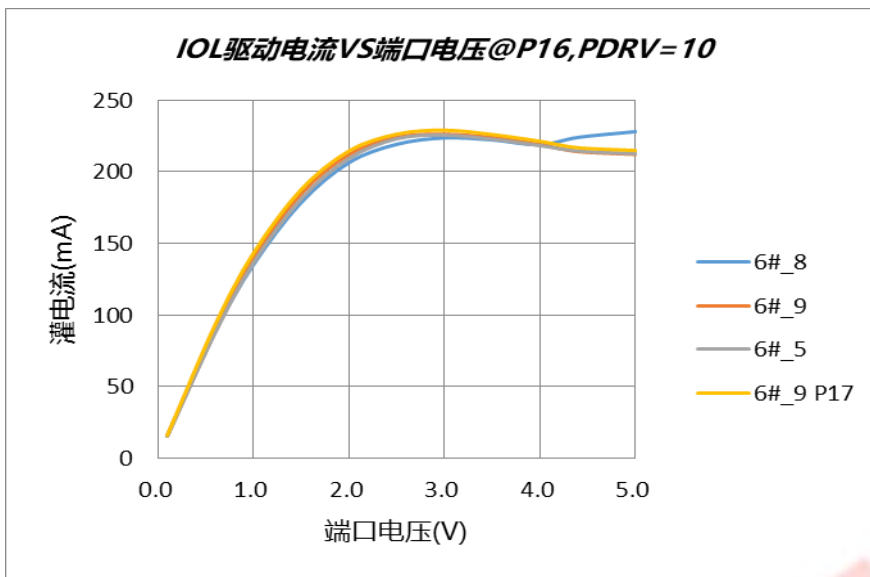
I/O 输出电流 VS 端口电压 (VDD=5V)

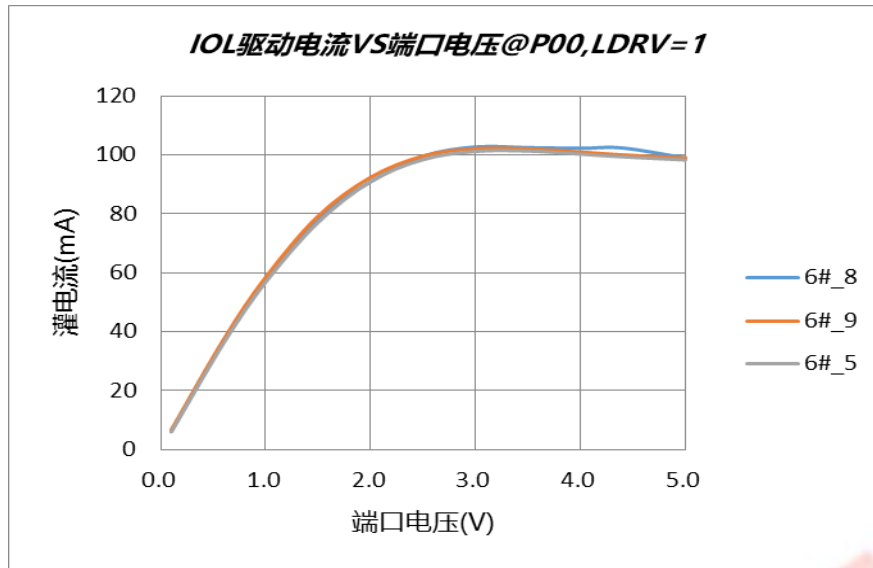






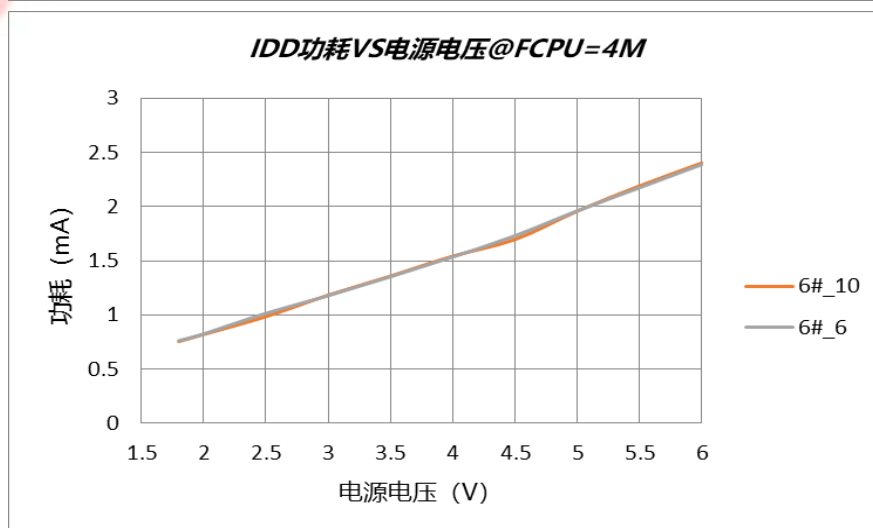
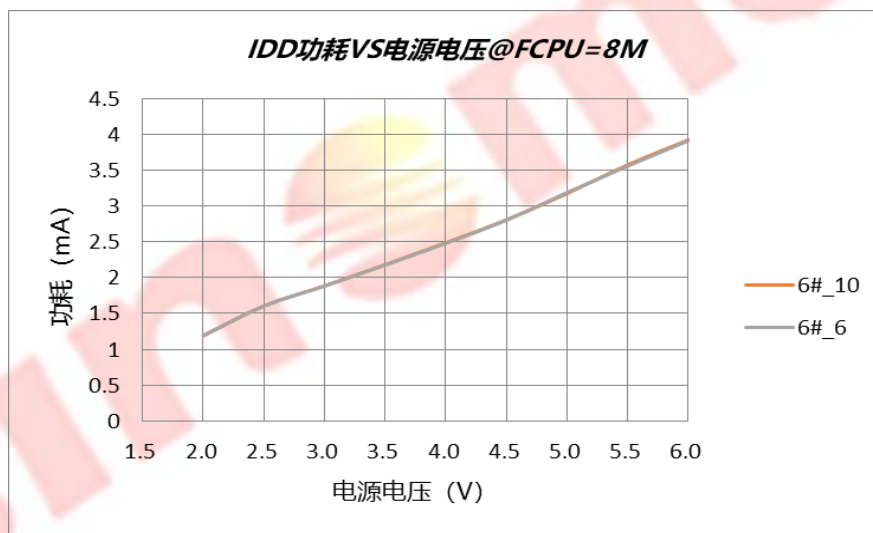




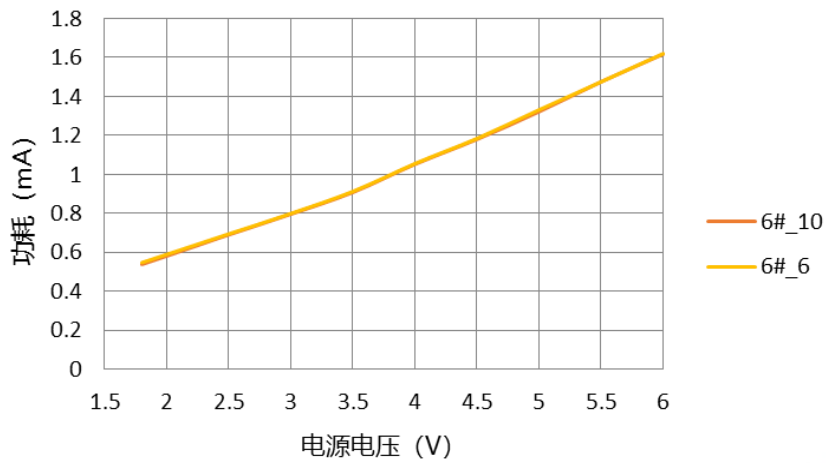


12.2 功耗特性

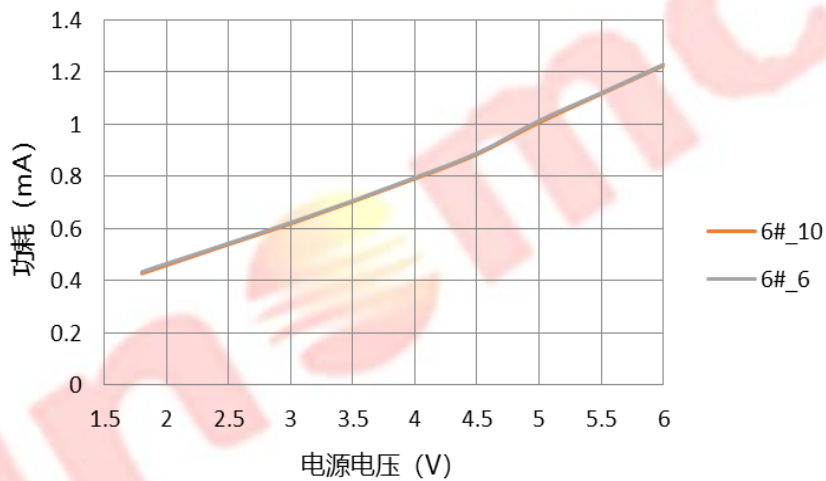
运行模式 功耗 VS 电源电压



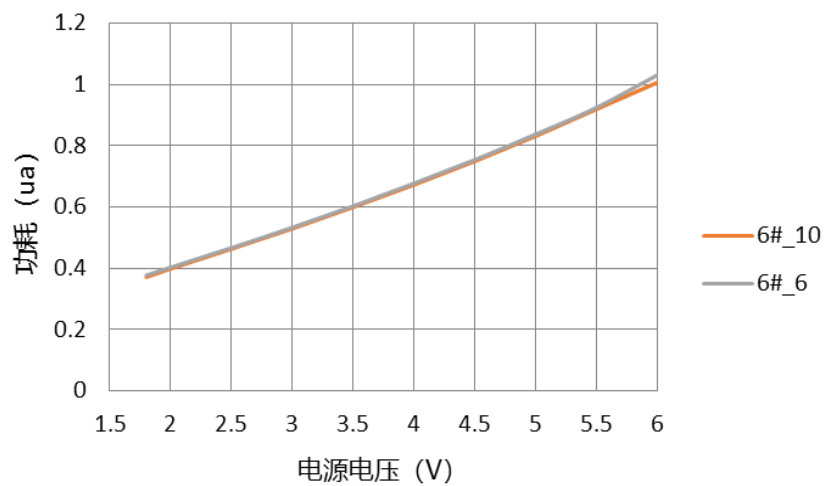
IDD功耗SV电源电压@FCPU=2M

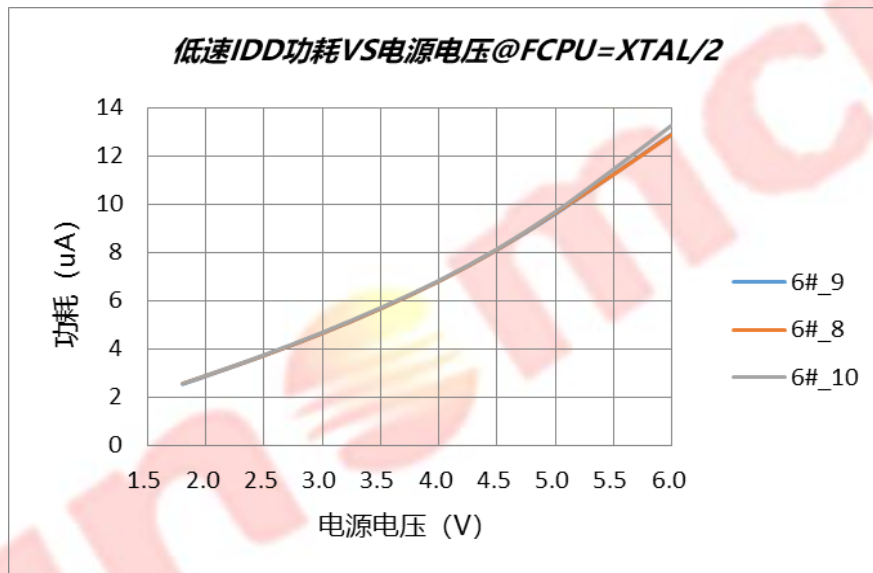
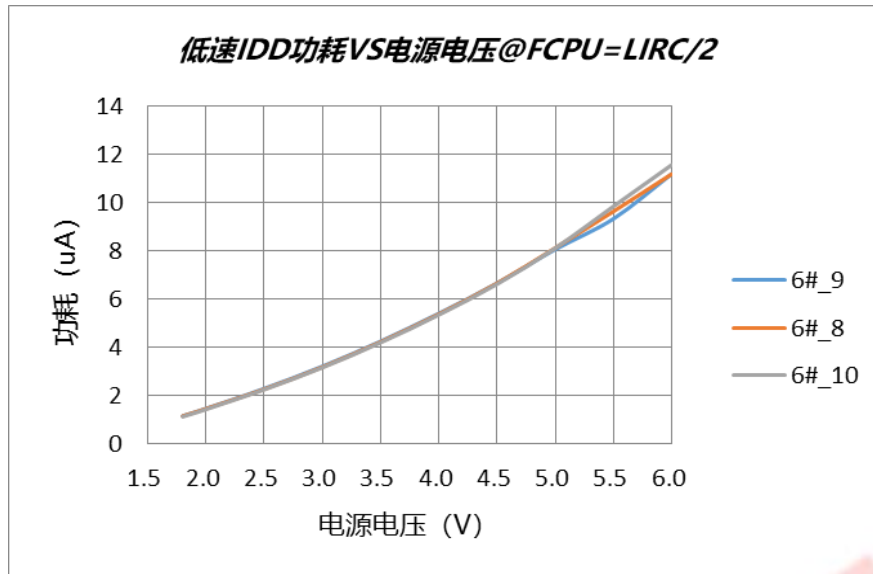


IDD功耗VS电源电压@FCPU=1M

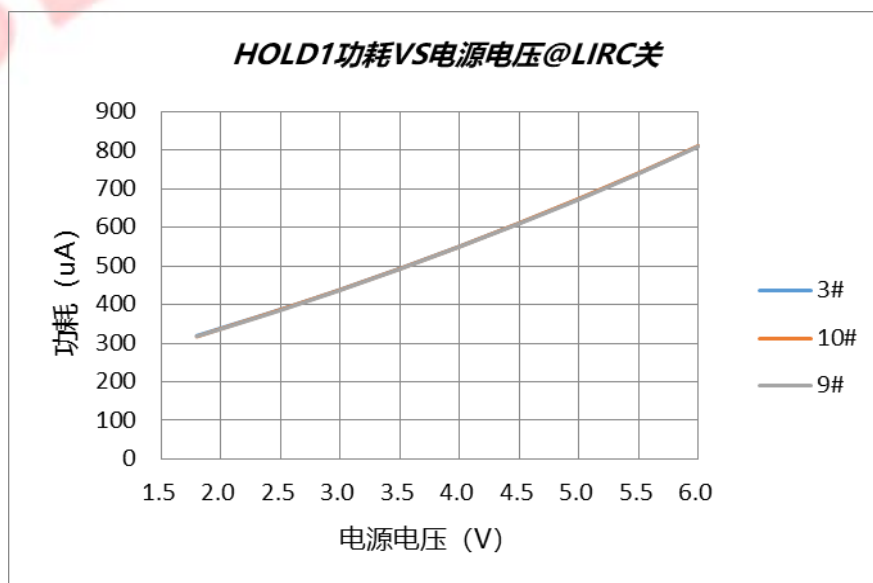


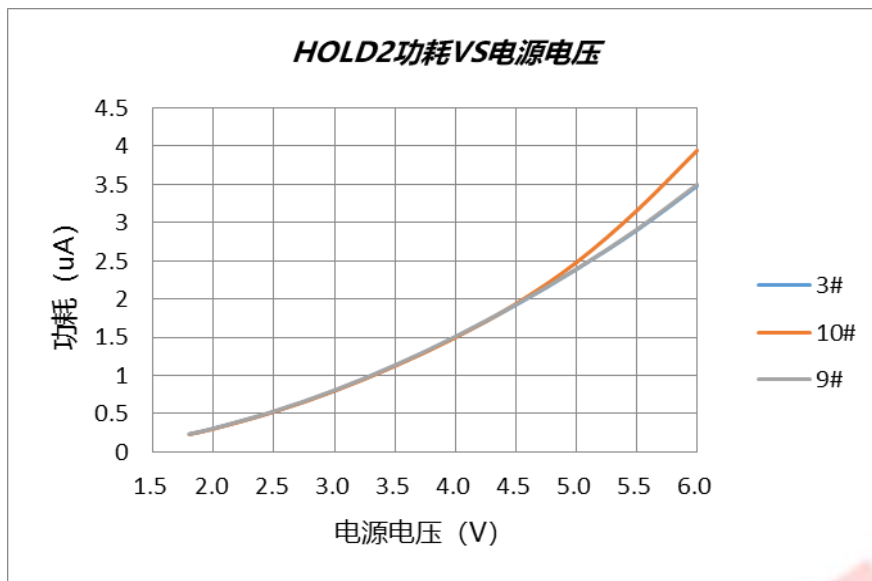
IDD功耗VS电源电压@FCPU=500K



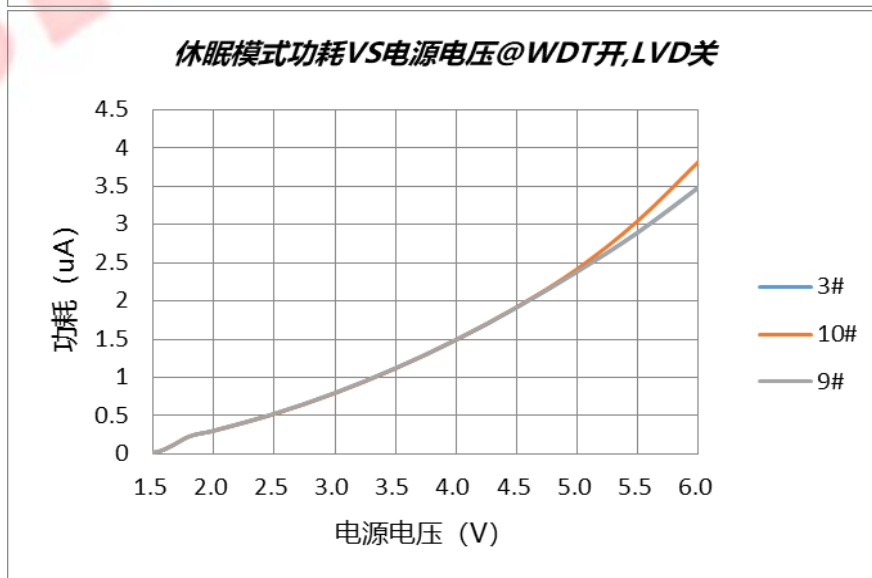
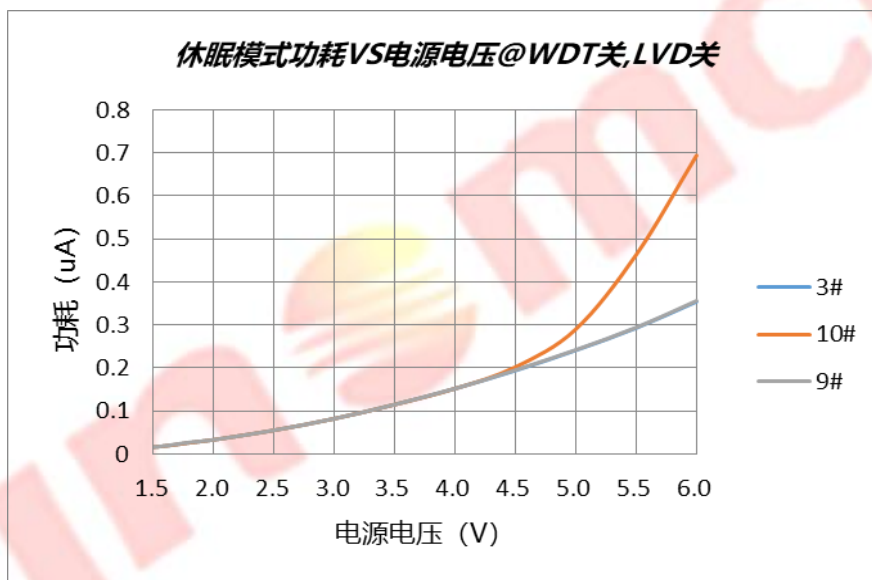


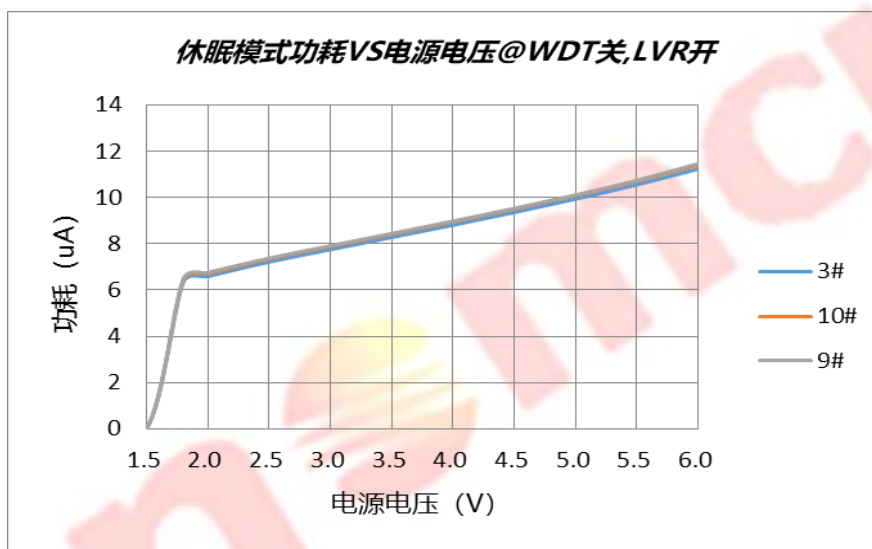
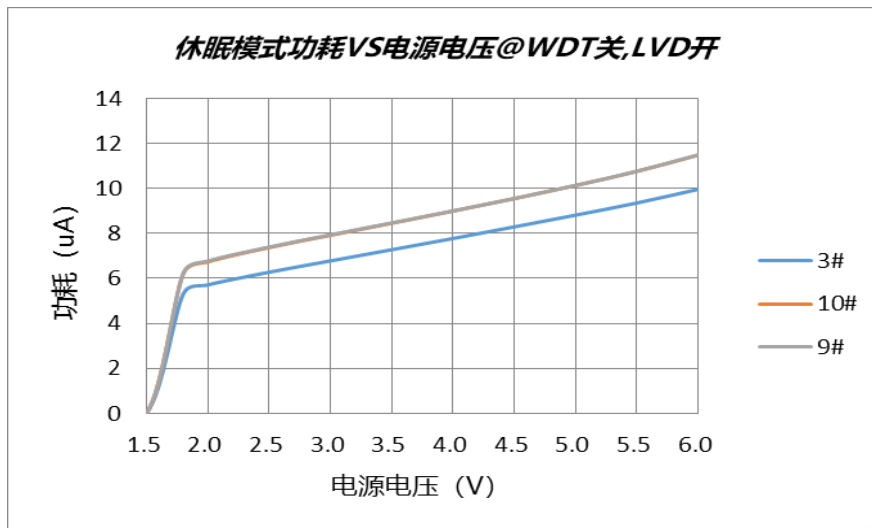
HOLD 模式 功耗 VS 电源电压





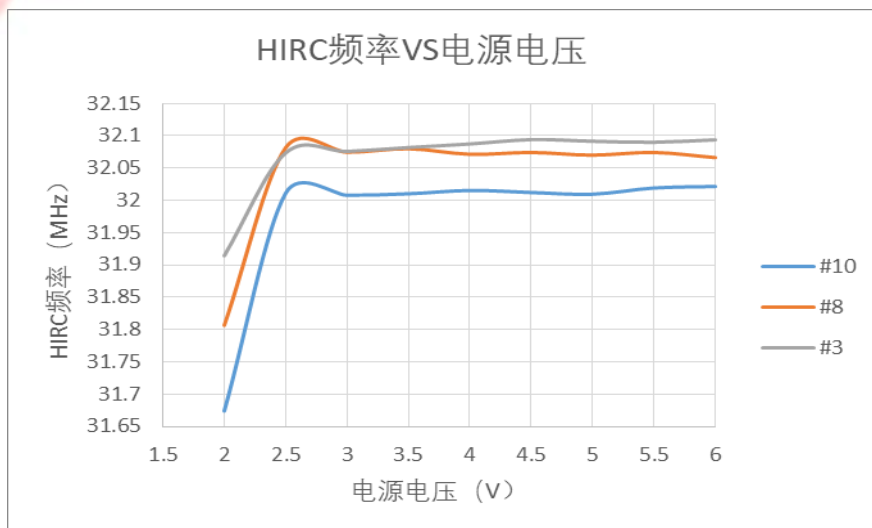
休眠模式 功耗 VS 电源电压

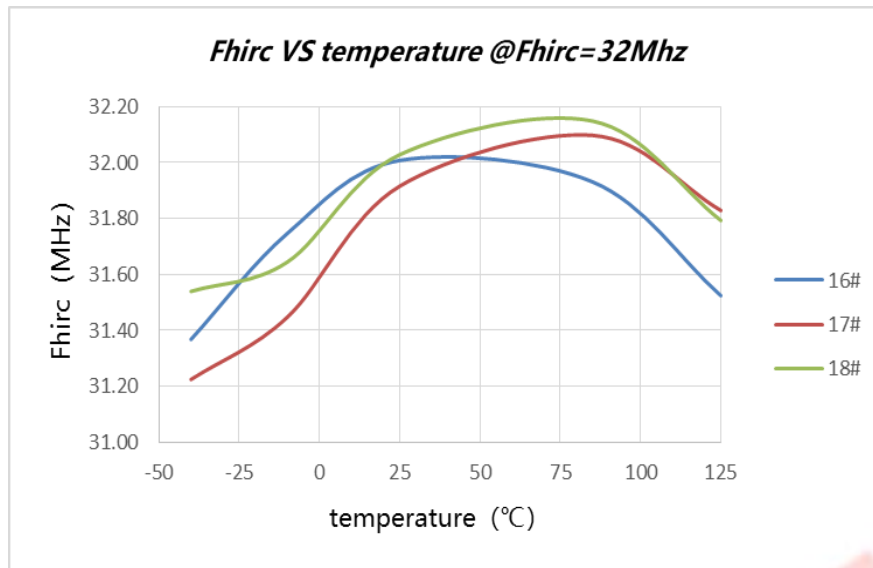




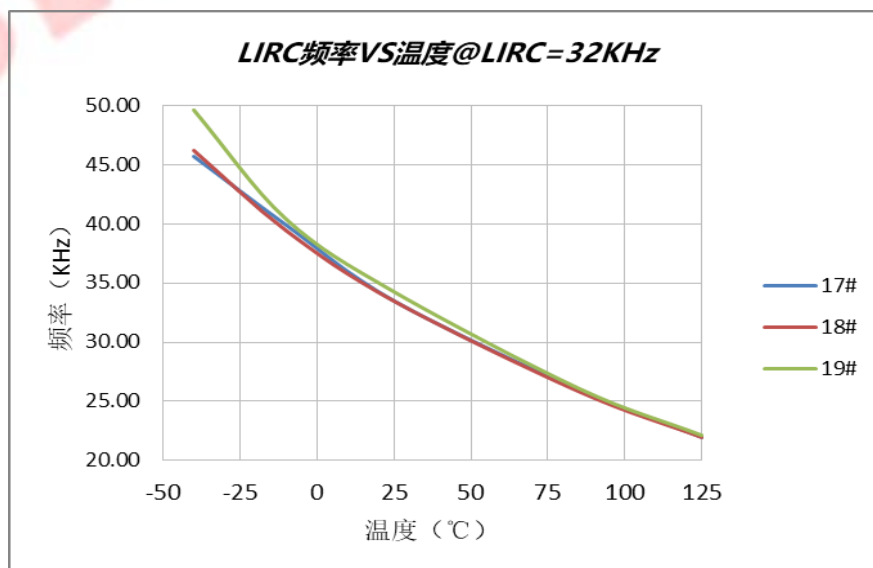
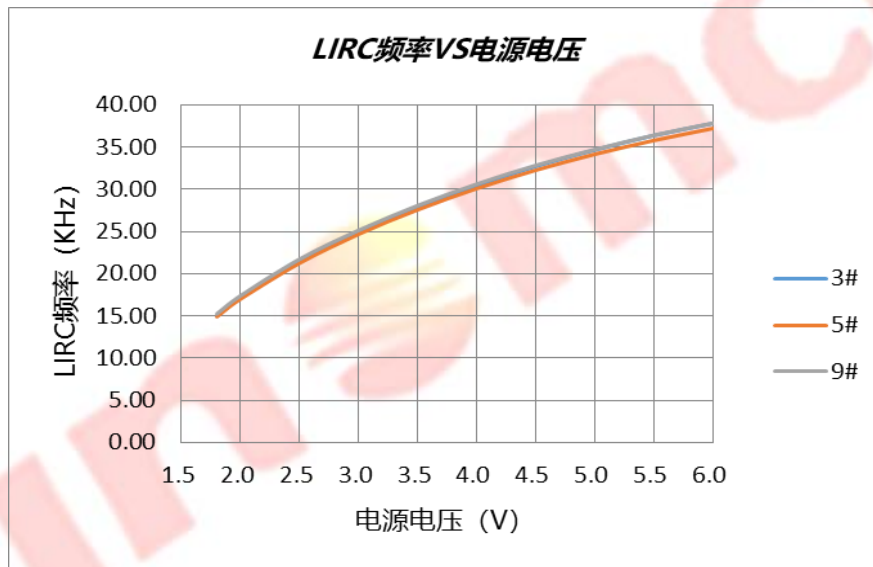
12.3 模拟电路特性

HIRC 频率 VS 电源电压/温度

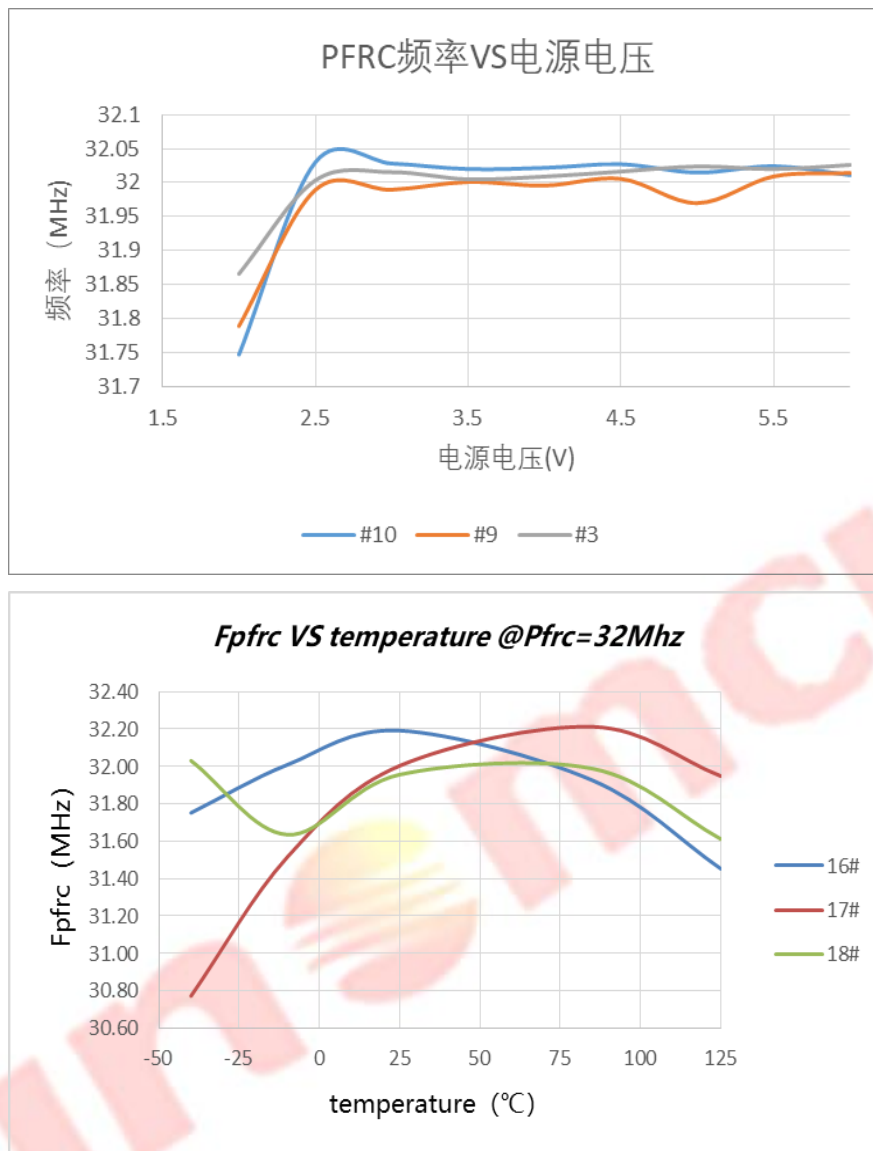




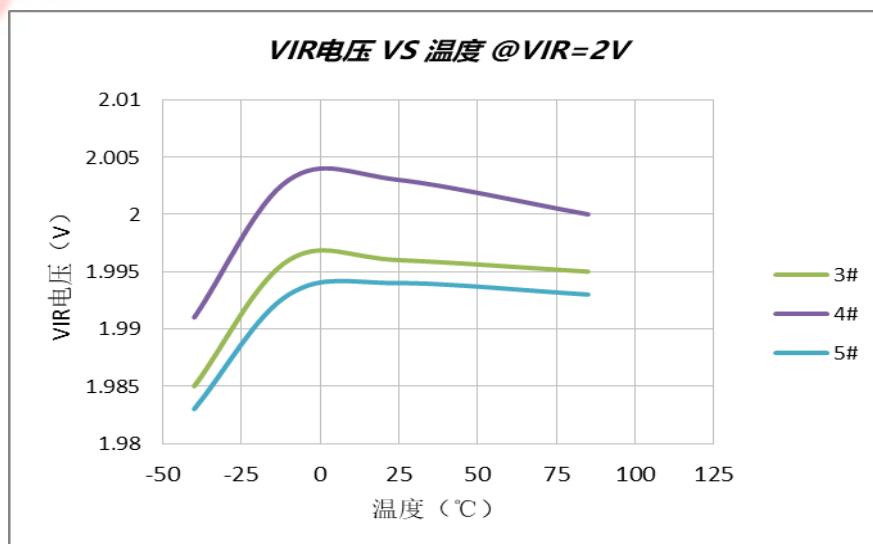
LIRC 频率 VS 电源电压/温度

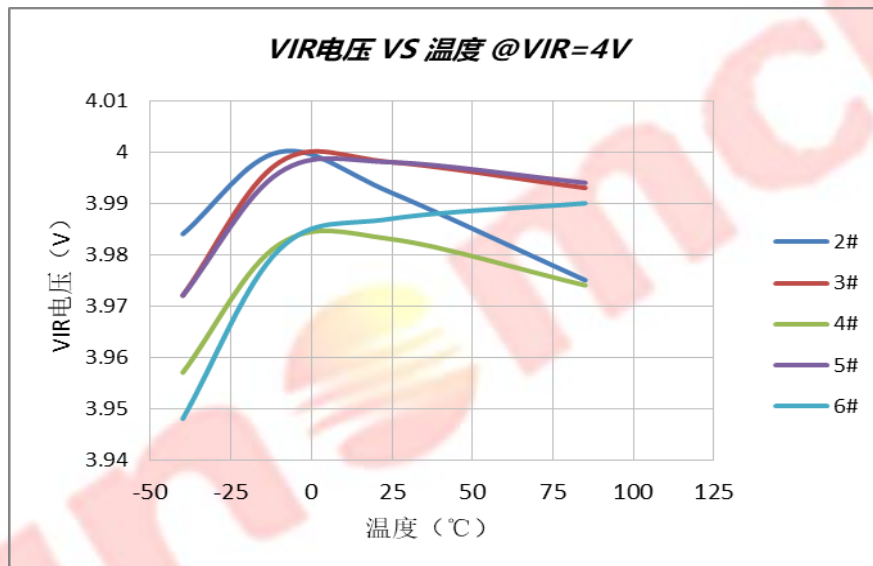
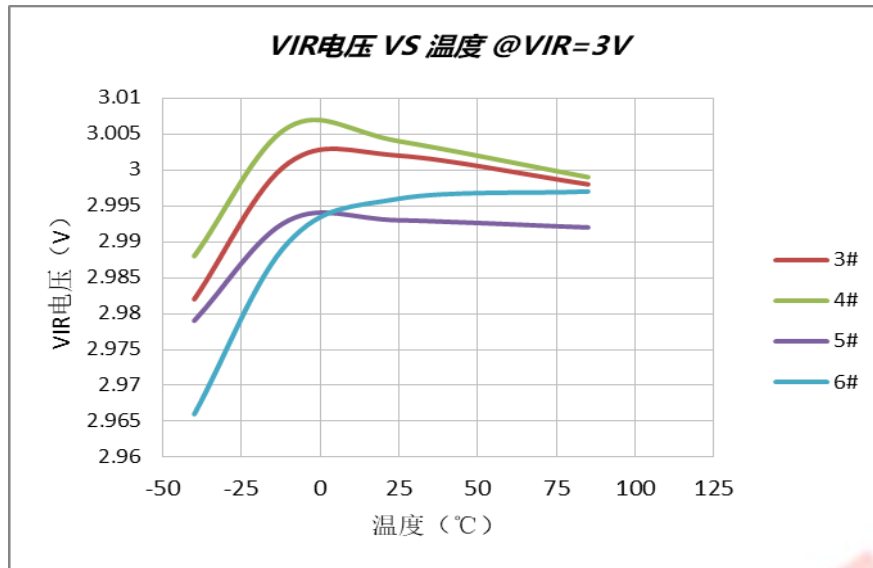


PFRC 频率 VS 电源电压/温度



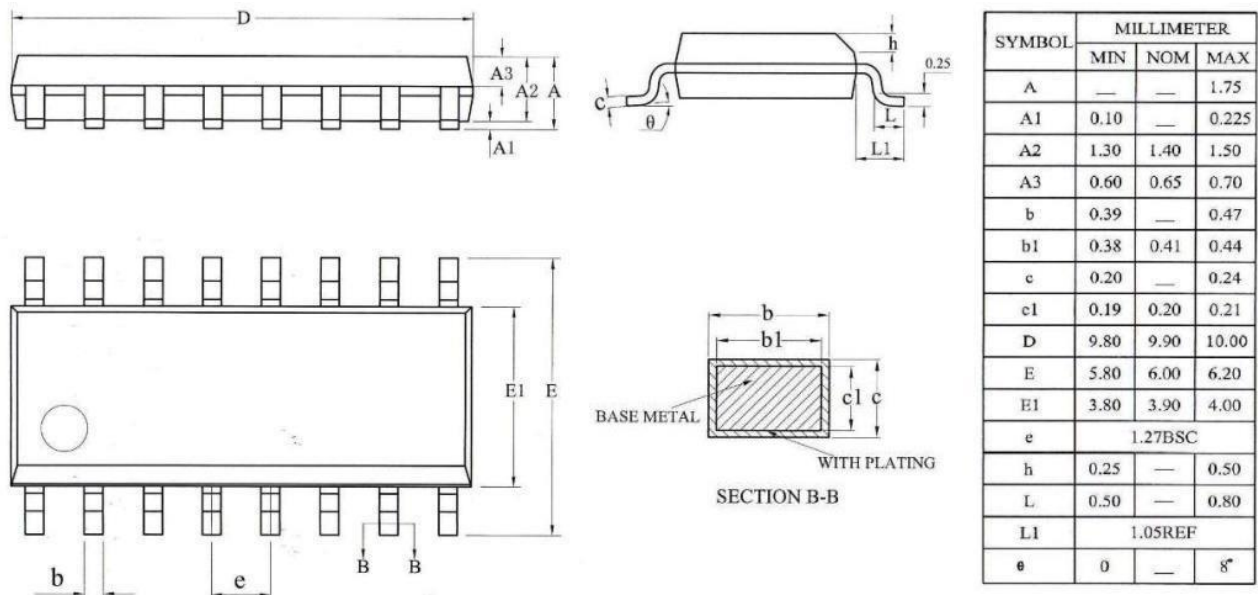
VIR 电压 VS 温度



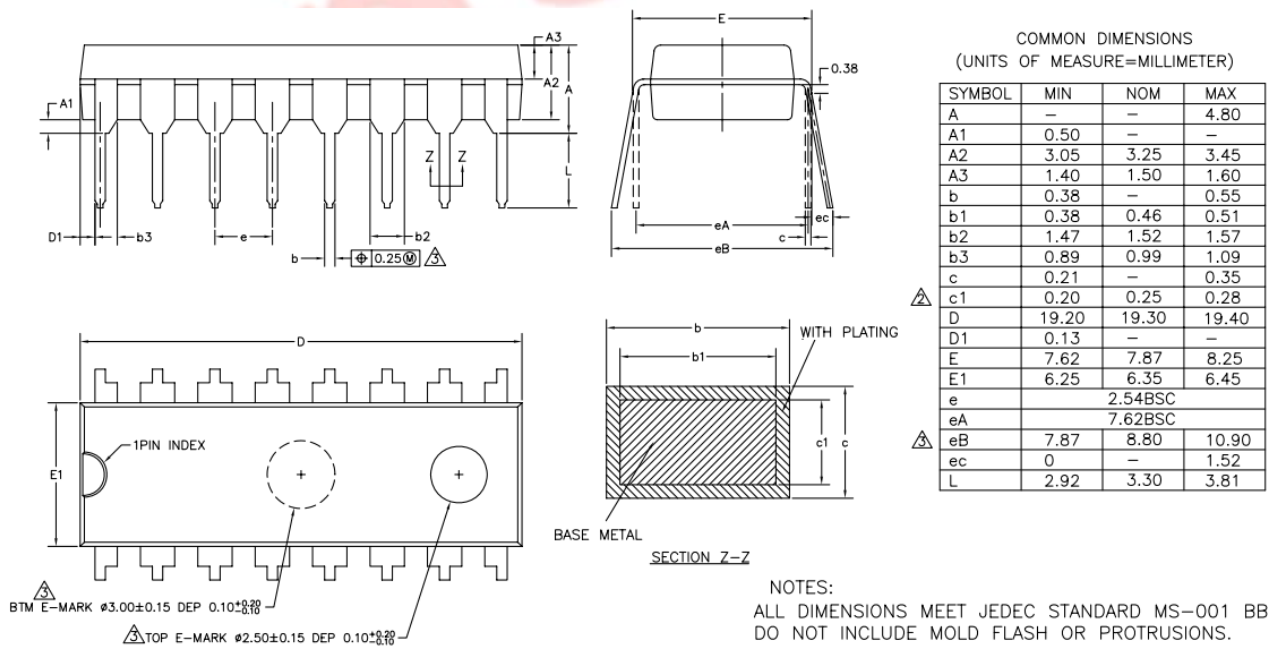


13 封装尺寸

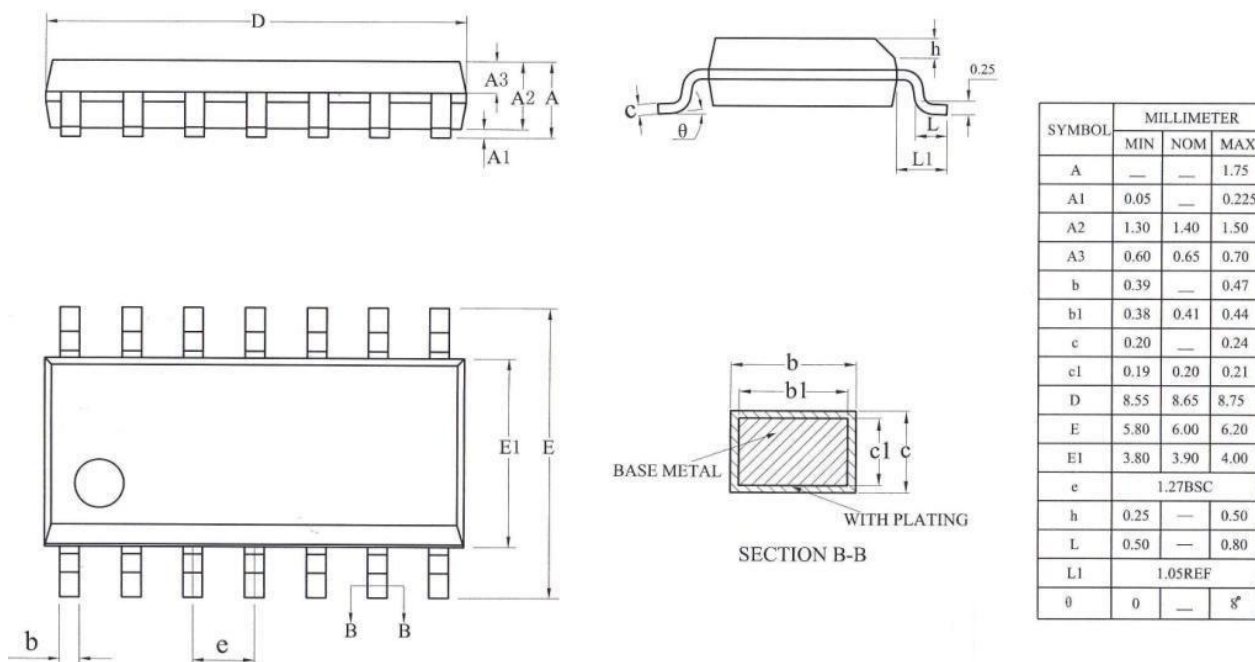
13.1 SOP16



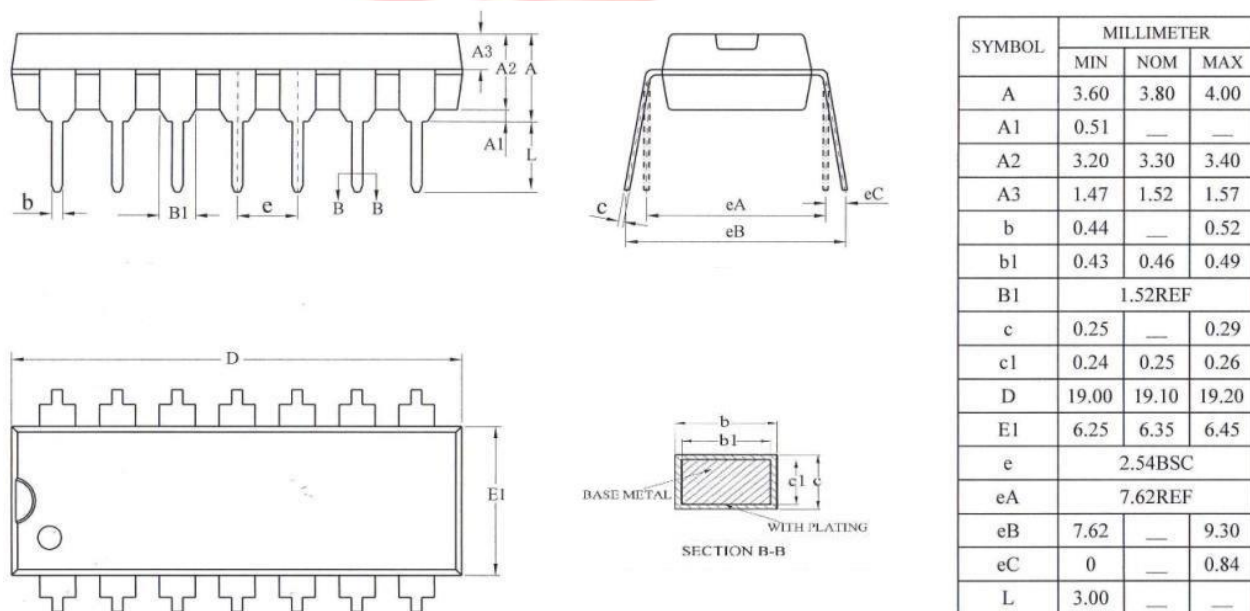
13.2 DIP16



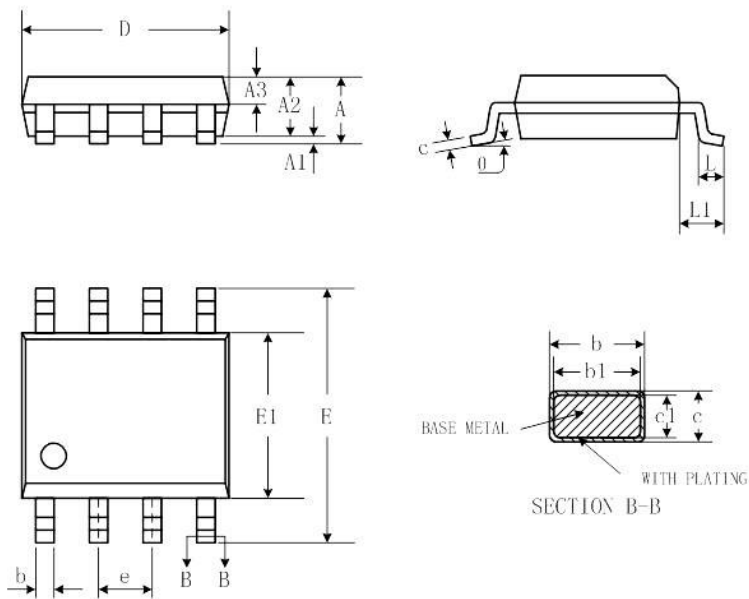
13.3 SOP14



13.4 DIP14

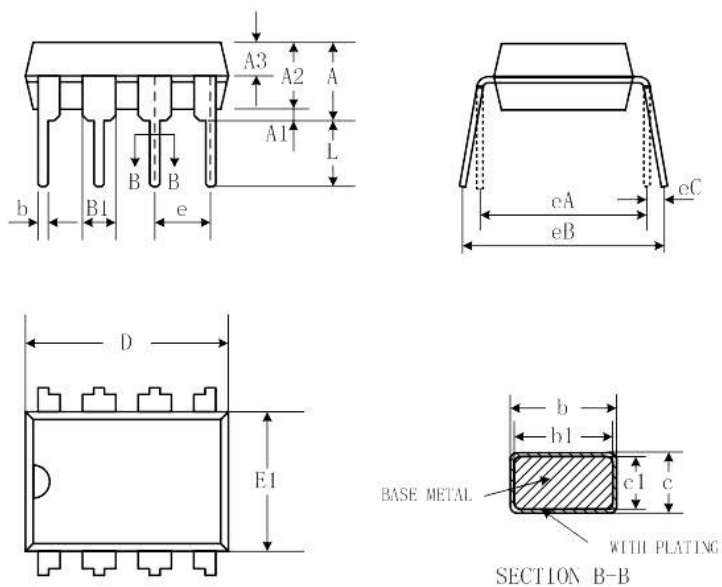


13.5 SOP8



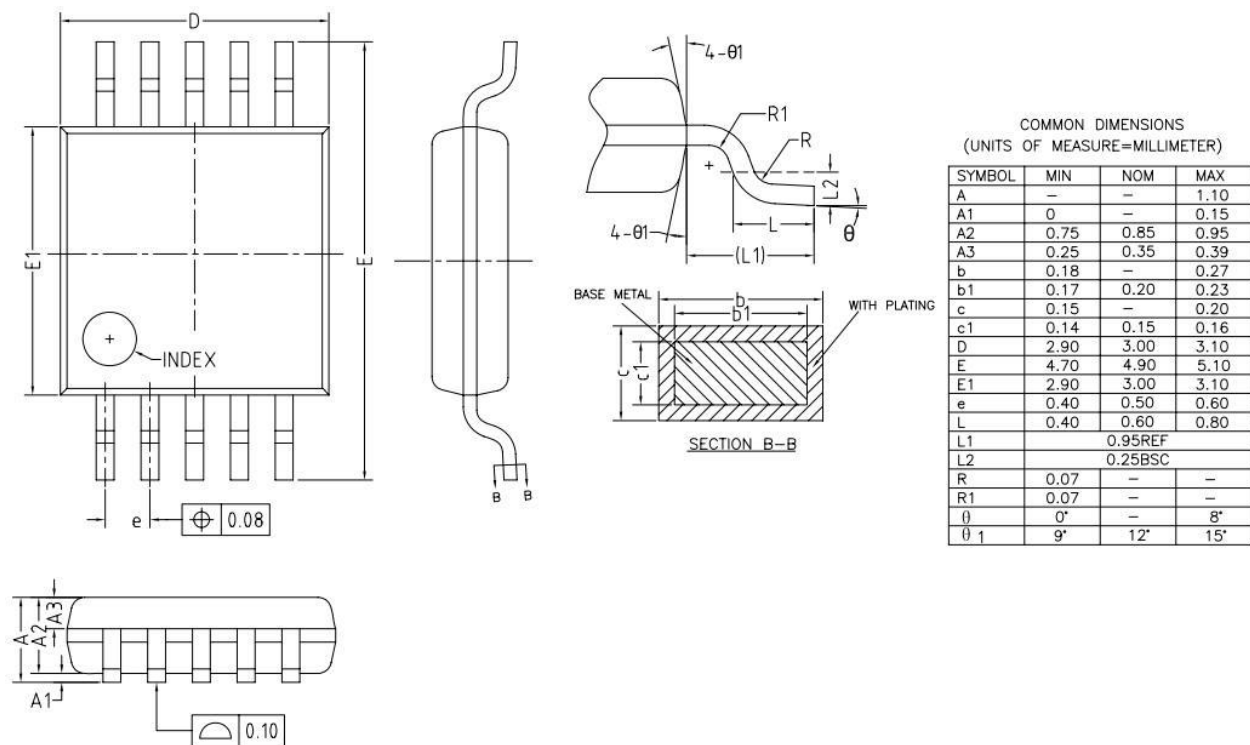
SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	-	-	1.77
A1	0.08	0.18	0.28
A2	1.20	1.40	1.60
A3	0.55	0.65	0.75
b	0.39	-	0.48
b1	0.38	0.41	0.43
c	0.21	-	0.26
c1	0.19	0.20	0.21
D	4.70	4.90	5.10
E	5.80	6.00	6.20
E1	3.70	3.90	4.10
e	1.27BSC		
L	0.50	0.65	0.80
L1	1.05BSC		
θ	0	-	8°

13.6 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	-	-

13.7 MSOP10



14 修订记录

版本	日期	修订内容
V1.0	2021-12-22	发布初版;