SinoMCU 8 位单片机

MC32F7062

用户手册

V1.3

本产品为广东晟矽微电子有限公司研制并销售，公司保留对产品在可靠性、功能和设计方面的改进作进一步说明的权利。本文档的更改，恕不另行通知。

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 8
2 电气特性 10
2.1 极限参数 10
2.2 直流电气特性 10
2.3 交流电气特性 11
2.4 ADC 特性参数 12
2.5 CMP 特性参数 12
2.6 OPA 特性参数 13
2.7 EEPROM 特性参数 13
3 CPU 与存储器 14
3.1 指令集 14
3.2 程序存储器 16
3.3 数据存储器 17
3.4 堆栈 18
3.5 控制寄存器 18
3.6 用户配置字 22
4 系统时钟 23
4.1 内部高频 RC 振荡器 23
4.2 内部低频 RC 振荡器 24
4.3 外部晶体振荡器 24
4.4 系统工作模式 25
4.5 低功耗模式 26
5 复位 27
5.1 复位条件 27
5.2 上电复位 27
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
7 定时器 TIMER 33
7.1 看门狗定时器 WDT 33
7.2 定时器 T0 33
7.3 定时器 T1 35
7.4 定时器 T2 39

7.5 定时器 T3 44
8 模数转换器 ADC 51
8.1 ADC 概述 51
8.2 ADC 相关寄存器 52
8.3 ADC 操作步骤 55
8.4 ADC 零点偏移修调流程 55
9 模拟比较器 CMP 56
9.1 CMP 概述 56
9.2 CMP 相关寄存器 56
9.3 CMP 失调电压调校流程 58
10 运算放大器 OPA 59
10.1 OPA 概述 59
10.2 OPA 相关寄存器 59
11 低电压检测 LVD 61
12 EEPROM 存储器 62
12.1 EEPROM 概述 62
12.2 EEPROM 相关寄存器 62
12.3 EEPROM 操作示例 63
13 FLASH 烧录编程 64
13.1 FLASH 在板编程 64
14 中断 65
14.1 外部中断 65
14.2 定时器中断 65
14.3 T1 捕捉中断 65
14.4 键盘中断 66
14.5 ADC 中断 66
14.6 比较器中断 66
14.7 LVD 中断 66
14.8 中断相关寄存器 67
15 特性曲线 70
15.1 I/O 特性 70
15.2 功耗特性 73
15.3 模拟电路特性 78
16 封装尺寸 85
16.1 SOP20 85
16.2 DIP20 85
16.3 TSSOP20 86
16.4 SOP16 86
16.5 DIP16 87
16.6 SOP14 87
16.7 SOP8 88
16.8 DIP8 88
17 修订记录 89

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 2/4/8/16/32/64 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
  - ✧ 4K×16 位 FLASH 型程序存储器，可通过间接寻址读取程序存储器内容
  - ✧ 支持在板不带电烧录编程
- 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
  - ✧ 256 字节 EEPROM 型数据存储器，支持单独烧录和软件读写，擦写次数至少 10 万次
- 3 组共 18 个 I/O
  - ✧ P0（P00~P07），P1（P10~P17），P2（P20~P21）
  - ✧ 所有端口均支持施密特输入，均支持推挽输出
  - ✧ P17 可复用为外部复位 RST 输入，P10/P11 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
  - ✧ P20/P21 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/2MHz），可用作系统高频时钟源，支持软件微调
  - ✧ 支持外接高频晶体振荡器（455KHz/4MHz~16MHz），可用作系统高频时钟源
  - ✧ 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
  - ✧ HOLD 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
  - ✧ 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，可实现外部计数和 BUZ 功能
  - ✧ 16 位定时器 T1，可实现外部计数、BUZ 和内/外部信号输入捕捉功能
  - ✧ 8 位定时器 T2，可实现 1 对 8+3 模式的带死区互补 PWM
  - ✧ 8 位定时器 T3，可实现 3 路共周期独立占空比的 PWM（其中 1 路可扩展为 1 对 8+3 模式的带死区互补 PWM，且可从 3 组端口输出并支持输出保护功能）
- 1 个 12 位高精度 SAR 型 ADC

- ✧ 14 路外部通道：AN0~AN13；2 路内部通道：GND、VDD/4
- ✧ 参考电压可选：VDD、内部参考电压 VIR（2V/3V/4V）、外部参考电压 VER（VERI 输入）
- ✧ ADC 时钟：FCPU 的 2/4/8/16 分频
- ✧ 支持零点校准

■ 1 个模拟比较器 CMP
- ✧ 输入共模 0 ~（VDD-1.4V），支持失调电压自消除调校（调校精度±2mV），输出无回滞且支持去抖处理
- ✧ 正端输入可选择外部输入电压、或 VDD 内部分压电压
- ✧ 负端输入可选择外部输入电压
- ✧ 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

■ 1 个运算放大器 OPA
- ✧ 输入共模 0～（VDD-1.4V），开环放大倍数 60dB

■ 中断
- ✧ 外部中断（INT0~INT1），键盘中断（P10~P17）
- ✧ 定时器中断（T0~T3），T1 捕捉中断
- ✧ ADC 中断，CMP 中断，LVD 中断

■ 低电压检测 LVD：2.0V/2.2V/2.4V/2.6V/2.8V/3.2V/3.6V/4.0V

■ 低电压复位 LVR：2.0V/2.4V/2.7V/3.2V

■ 工作电压（@HIRC）
- ✧ VLVR32 ~ 5.5V @ Fcpu = 0~16MHz
- ✧ VLVR24 ~ 5.5V @ Fcpu = 0~8MHz
- ✧ VLVR20 ~ 5.5V @ Fcpu = 0~4MHz

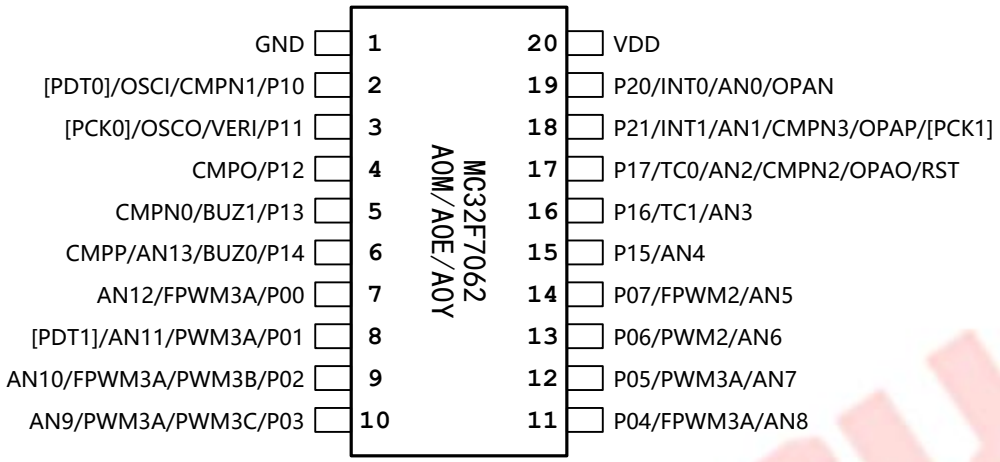
■ 封装形式：SOP20/DIP20/TSSOP20/SOP16/DIP16/SOP14/SOP8/DIP8

1.2 订购信息

产品名称	封装形式	备注
MC32F7062A0M	SOP20	
MC32F7062A0E	DIP20	
MC32F7062A0Y	TSSOP20	
MC32F7062A1M	SOP20	
MC32F7062A0K	SOP16	
MC32F7062A0C	DIP16	
MC32F7062A1K	SOP16	
MC32F7062A2K	SOP16	
MC32F7062A0J	SOP14	
MC32F7062A1J	SOP14	
MC32F7062A0H	SOP8	
MC32F7062A0A	DIP8	

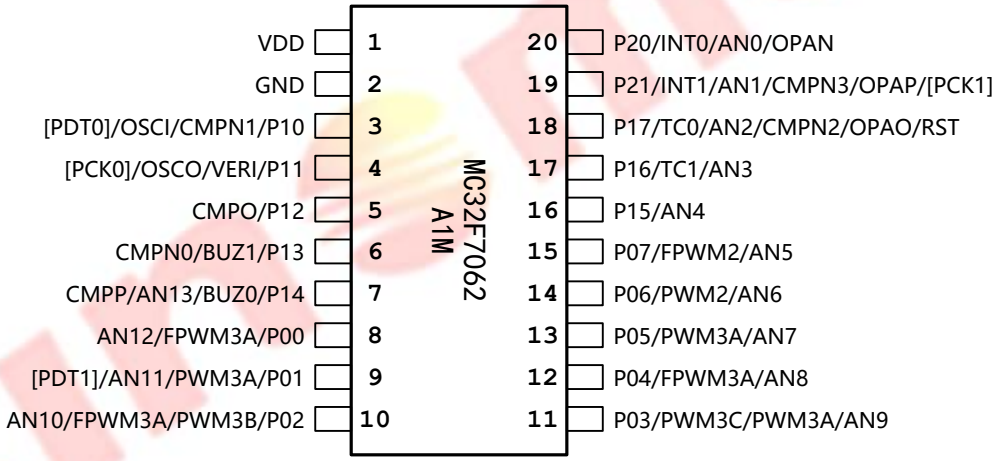
1.3 引脚排列

MC32F7062A0M/A0E/A0Y



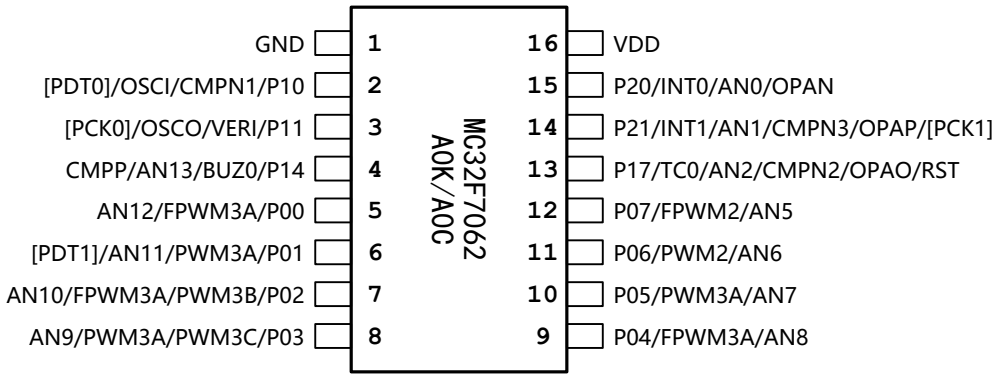
SOP20/DIP20/TSSOP20

MC32F7062A1M



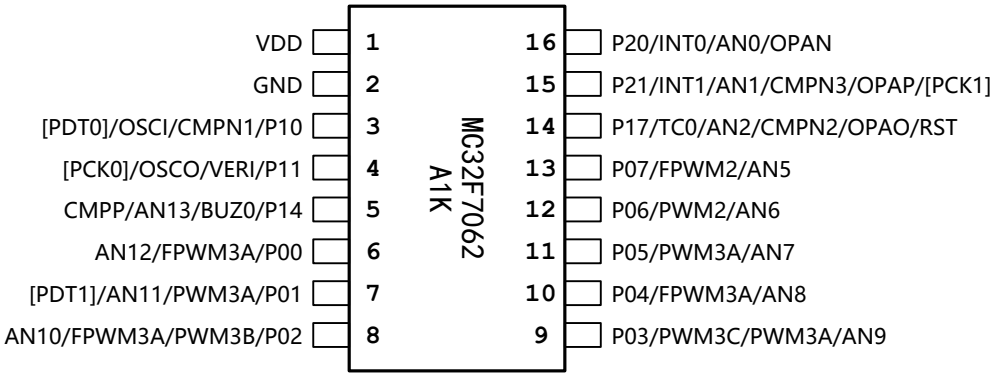
SOP20

MC32F7062A0K/A0C



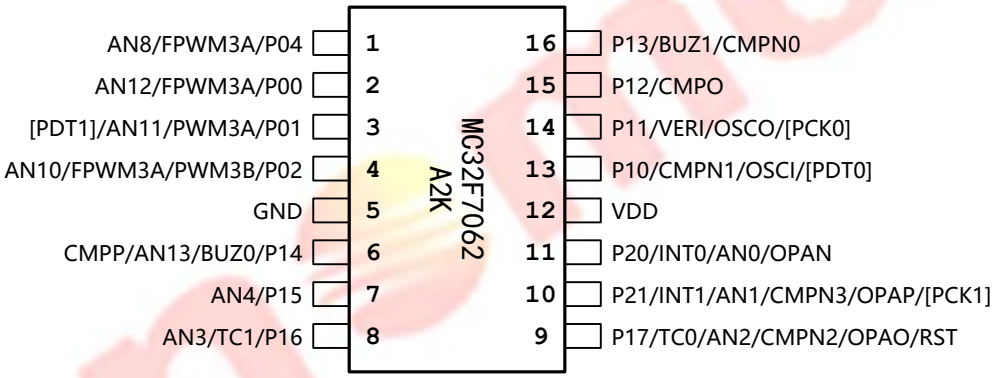
SOP16/DIP16

MC32F7062A1K



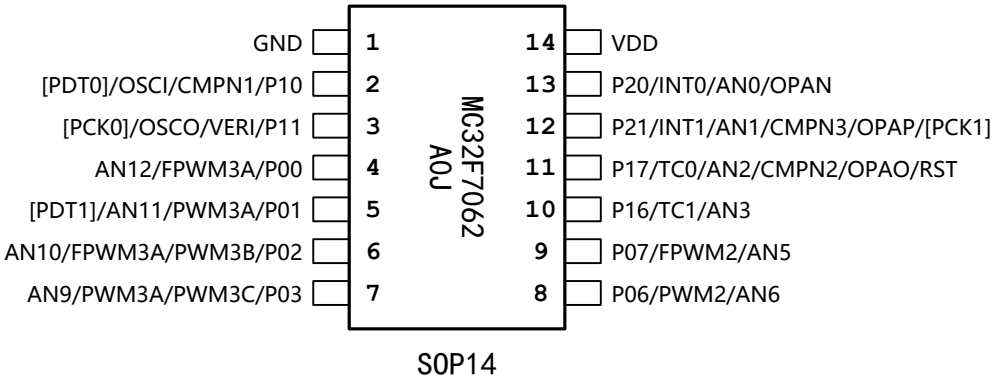
SOP16

MC32F7062A2K



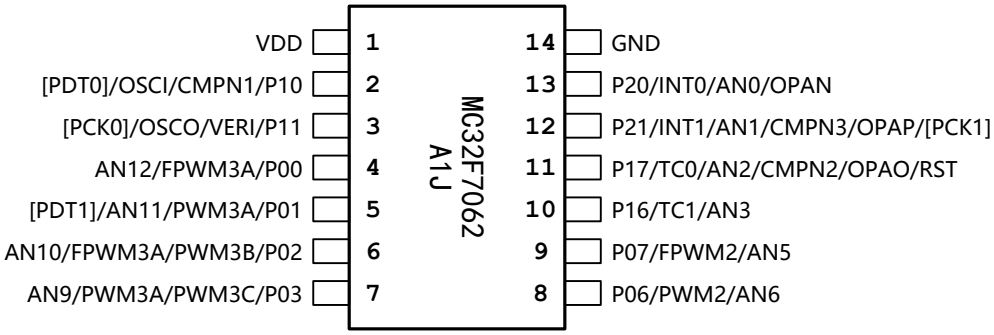
SOP16

MC32F7062A0J



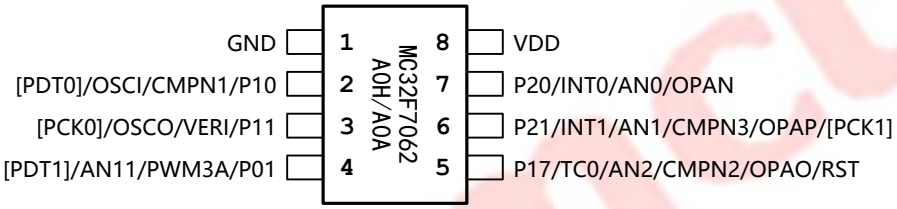
SOP14

MC32F7062A1J



SOP14

MC32F7062A0H/A0A



SOP8/DIP8

1.4 端口说明

端口名称	类型	功能说明
VDD	P	电源
GND	P	地
P0，P1，P2	D	GPIO（推挽输出），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
BUZ0	DO	定时器 T0 的 BUZ 输出
BUZ1	DO	定时器 T1 的 BUZ 输出
PWM2，FPWM2	DO	定时器 T2 的 PWM 及其互补输出
PWM3A~PWM3C	DO	定时器 T3 的 3 路 PWM 输出
FPWM3A	DO	定时器 T3 中 PWM3A 的互补 PWM 输出
AN0~AN13	AI	ADC 外部输入通道
VERI	AI	ADC 外部参考电压输入
CMPP	AI	CMP 正端外部输入
CMPN0~CMPN3	AI	CMP 负端外部输入通道
CMPO	DO	CMP 输出

OPAP，OPAN	AI	OPA 正/负端外部输入
OPAO	AO	OPA 输出
OSCI，OSCO	A	外部时钟振荡器输入/输出
RST	DI	外部复位输入
PCK0/PDT0，PCK1/PDT1	D	编程时钟/数据接口 在线仿真接口

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

2 电气特性

2.1 极限参数

参数	符号	值	单位
电源电压	VDD	-0.3~6.0	V
I/O 输入电压	Vin	-0.3~VDD+0.3	V
工作温度	Ta	-40~85	℃
储存温度	Tstg	-65~150	℃
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，则将影响其可靠性。

2.2 直流电气特性

VDD=5V，T=25℃

特性	符号	端口	条件	最小	典型	最大	单位
工作电压	VDD	VDD	Fcpu=16MHz@FHIRC(32M)/2	VLVR32		5.5	V
			Fcpu=8MHz@FHIRC(32M)/4	VLVR24		5.5	
			Fcpu=4MHz@FHIRC(32M)/8	VLVR20		5.5	
			Fcpu=2MHz@FHIRC(32M)/16	VLVR20		5.5	
			Fcpu=1MHz@FHIRC(32M)/32	VLVR20		5.5	
			Fcpu=500KHz@FHIRC(32M)/64	VLVR20		5.5	
			Fcpu=16KHz@FLIRC(32K)/2	VLVR20		5.5	
输入漏电流	Ileak	所有输入脚	VDD=5V	-1		1	μA
输入高电平	Vih	所有输入脚		0.8VDD			V
输入低电平	Vil	所有输入脚				0.2VDD	V
上拉电阻	Rpu	P0, P1, P2	VDD=5V, Vin=0		16		KΩ
下拉电阻	Rpd	P0, P1, P2	Vin=VDD=5V		16		KΩ
输出源电流	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
运行模式功耗	Irun	VDD	VDD=5V,Fcpu=16MHz@HIRC(32M)		6.4		mA
			VDD=5V, Fcpu=8MHz@HIRC(32M)		3.5		mA
			VDD=5V, Fcpu=4MHz@HIRC(32M)		2.1		mA
			VDD=5V, Fcpu=2MHz@HIRC(32M)		1.3		mA
			VDD=5V, Fcpu=1MHz@HIRC(32M)		0.9		mA

			VDD=5V, Fcpu=1MHz@HIRC(2M)		460		μA
			VDD=3V, Fcpu=1MHz@HIRC(2M)		280		μA
			VDD=5V, Fcpu=500KHz@HIRC(2M)		270		μA
			VDD=3V, Fcpu=500KHz@HIRC(2M)		170		μA
			VDD=5V, Fcpu=16KHz@LIRC(32K)		22		μA
HOLD 功耗	Ihold	VDD	CPU 停, HIRC 关, LIRC 开		3		μA
			CPU 停, HIRC 关, LEXT 开		6		μA
休眠模式功耗	Istop	VDD	休眠模式, WDT/LVR 关		1	3	μA
			休眠模式, WDT 开, LVR 关		3	6	μA
			休眠模式, WDT 关, LVR 开		11	25	μA
低压检测电压	VLVD	VDD	LVDVS 选择	-10%		+10%	V
LVD 响应时间	TLVD					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	FHIRC1	VDD=5V, T=25℃	-1.5%	32	+1.5%	MHz
		VDD=2.2V~5.5V, T=-20℃~70℃	-3%		+3%	
		VDD=2.2V~5.5V, T=-40℃~85℃	-5%		+5%	
	FHIRC2	VDD=3V, T=25℃	-1.5%	2	+1.5%	MHz
		VDD=2.2V~3.6V, T=-20℃~70℃	-3%		+3%	
		VDD=2.2V~3.6V, T=-40℃~85℃	-5%		+5%	
LIRC 振荡频率	FLIRC	VDD=5V, T=25℃	-50%	32	+50%	KHz
16M 晶振起振电压		T=25℃	3.2			V
8M 晶振起振电压		T=25℃	2.4			V
4M 晶振起振电压		T=25℃	2.0			V
455K 晶振起振电压		T=25℃	2.0			V
32768 晶振起振电压		T=25℃	2.0			V
32768 晶振起振时间		VDD=5V, T=25℃		1		s
OSCI 内部对地电容	CG			10		pF
OSCO 内部对地电容	CD			10		pF

2.4 ADC 特性参数

VDD=5V，T=25℃

特性	符号	条件	最小	典型	最大	单位
ADC 有效工作电压	VADC	T=-40℃~85℃	2.7		5.5	V
积分非线性误差	INL	VREF=VDD, FADC=1MHz, Tcon=27μs			±4	LSB
微分非线性误差	DNL	VREF=VDD, FADC=1MHz, Tcon=27μs			±2	LSB
零点偏移误差	EZ	VREF=VDD, FADC=1MHz, Tcon=27μs			±4	LSB
增益误差	ET	VREF=VDD, FADC=1MHz, Tcon=27μs			±4	LSB
转换时钟	FADC	VDD=5V			1	MHz
转换时间	Tcon		14		27	1/FADC
ADC 输入电压	VAIN	AN0~AN13	GND		VREF	V
		AN12, AN13	GND		VDD-1.4	
ADC 输入阻抗	RAIN		2			MΩ
ADC 输入电流	IAIN				2	μA
ADC 动态电流	IADD	VDD=5V, AD 转换中		1	3	mA
ADC 静态电流	IADS	VDD=5V, ADC 关闭		0.1	1	μA
模拟信号源推荐阻抗	ZAIN				10	KΩ
ADC 参考电压	VREF	选择 VDD		VDD		V
		选择内部参考电压 VIR, T=25℃	-1.5%	2/3/4	+1.5%	
		选择内部参考电压 VIR, T=-20℃~70℃	-3%		+3%	
		选择内部参考电压 VIR, T=-40℃~85℃	-5%		+5%	
		选择外部参考电压 VER	2		VDD	
VIR 有效工作电压	VVIR	选择内部参考电压 VIR	VIR+0.5		VDD	V

2.5 CMP 特性参数

VDD=5V，T=25℃

特性	符号	条件	最小	典型	最大	单位
比较器有效工作电压	VCMP	T=-40℃~85℃	2.7		5.5	V
工作电流	ICMP			90		μA
输入失调电压	Voffset	已调校, CMPOE=0	-2		+2	mV
		已调校, CMPOE=1	-5		+5	
		未调校	-15		+15	
输入共模电压	Vcom		0		VDD-1.4	V
响应时间	TRESP				2	μs
比较器启动时间					200	μs
通道切换等待时间	TCCW		2			μs

2.6 OPA 特性参数

VDD=5V，T=25℃

特性	符号	条件	最小	典型	最大	单位
OPA 有效工作电压	VOPA	T=-40℃~85℃	2.7		5.5	V
OPA 启动时间					200	μs
输入失调电压	Voffset		-15		+15	mV
输入共模电压	Vcom		0		VDD-1.4	V
输出电压摆幅	Vovs		0	0.2	0.3	V
			4.7	4.8	5.0	
电压摆率	SR		0.1	0.2		V/μs
开环增益	AVOL		60	80		dB
电源抑制比	PSRR		60	80		dB
共模抑制比	CMRR		60	80		dB
正端输入内部电阻	ROPAP			6		KΩ

2.7 EEPROM 特性参数

VDD=5V，T=25℃

特性	符号	条件	最小	典型	最大	单位
EEPROM 读操作电压	VEERD	T=-40℃~85℃	2.0		5.5	V
EEPROM 写操作电压	VEEWR	T=-40℃~85℃	2.0		5.5	V
EEPROM 写操作电流	IEEWR	T=-40℃~85℃		2		mA
EEPROM 单字节写入时间	TEEWR	VDD=2.0V~5.5V, T=-40℃~85℃		5		ms
EEPROM 擦写次数		VDD=5V, T=25℃	100000			cycle
EEPROM 数据保存时间			10			year

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→A	1	1	C,DC,Z
ADDRA R	R 和 A 相加，结果存入 R	R+A→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→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→R	1	1	C,DC,Z
RSUBAR R	R 和 A 相减，结果存入 A	R-A→A	1	1	C,DC,Z
RSUBRA R	R 和 A 相减，结果存入 R	R-A→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→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→R	1	1	C,DC,Z
ASUBAR R	A 和 R 相减，结果存入 A	A-R→A	1	1	C,DC,Z
ASUBRA R	A 和 R 相减，结果存入 R	A-R→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→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→R	1	1	C,DC,Z
ANDAR R	R 和 A 与操作，结果存入 A	R and A→A	1	1	Z
ANDRA R	R 和 A 与操作，结果存入 R	R and A→R	1	1	Z
ORAR R	R 和 A 或操作，结果存入 A	R or A→A	1	1	Z
ORRA R	R 和 A 或操作，结果存入 R	R or A→R	1	1	Z
XORAR R	R 和 A 异或操作，结果存入 A	R xor A→A	1	1	Z
XORRA R	R 和 A 异或操作，结果存入 R	R xor A→R	1	1	Z
COMAR R	对 R 取反，结果存入 A	R 取反→A	1	1	Z
COMR R	对 R 取反，结果存入 R	R 取反→R	1	1	Z
RLA	A 循环左移（带 C 标志）	A[7]→C；A[6:0]→A[7:1]；C→A[0]	1	1	C
RLAR R	R 循环左移（带 C 标志），结果存入 A	R[7]→C；R[6:0]→A[7:1]；C→A[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
RRA	A 循环右移（带 C 标志）	A[0]→C；A[7:1]→A[6:0]；C→A[7]	1	1	C
RRAR R	R 循环右移（带 C 标志），结果存入 A	R[0]→C；R[7:1]→A[6:0]；C→A[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 A	R[7:4]→A[3:0]；R[3:0]→A[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 A 存入 R	A→R	1	1	-
MOVAR R	将 R 存入 A	R→A	1	1	Z
MOVR R	将 R 存入 R	R→R	1	1	Z
CLRA	将 A 清零	0→A	1	1	Z
CLRR R	将 R 清零	0→R	1	1	Z
INCA	A 自加 1	A+1→A	1	1	-
INCR R	R 自加 1	R+1→R	1	1	Z
INCAR R	R 加 1，结果存入 A	R+1→A	1	1	Z
DECA	A 自减 1	A-1→A	1	1	-
DECR R	R 自减 1	R-1→R	1	1	Z
DECAR R	R 减 1，结果存入 A	R-1→A	1	1	Z
JZA	A 自加 1：结果为 0 则跳过下一条指令	A+1→A：结果为 0 则 PC+2→PC	1/2	1	-
JZR R	R 自加 1：结果为 0 则跳过下一条指令	R+1→R：结果为 0 则 PC+2→PC	1/2	1	-
JZAR R	R 加 1，结果存入 A：结果为 0 则跳过下一条指令	R+1→A：结果为 0 则 PC+2→PC	1/2	1	-
DJZA	A 自减 1：结果为 0 则跳过下一条指令	A-1→A：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 A：结果为 0 则跳过下一条指令	R-1→A：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 I	I 和 A 相加，结果存入 A	I+A→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→A	1	1	C,DC,Z
ISUBAI I	I 和 A 相减，结果存入 A	I-A→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→A	1	1	C,DC,Z
ASUBAI I	A 和 I 相减，结果存入 A	A-I→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→A	1	1	C,DC,Z
ANDAI I	I 和 A 与操作，结果存入 A	I and A→A	1	1	Z
ORAI I	I 和 A 或操作，结果存入 A	I or A→A	1	1	Z
XORAI I	I 和 A 异或操作，结果存入 A	I xor A→A	1	1	Z
MOVAI I	将 I 存入 A	I→A	1	1	-
CALL K	子程序调用	PC+1→TOS；K→PC[12:0]	2	1	-
GOTO K	无条件跳转	K→PC[12:0]	2	1	-
RETURN	从子程序返回	TOS→PC	2	1	-
RETAI I	从子程序返回，并将 I 存入 A	TOS→PC；I→A	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 码加法操作后，将 A 的值调整为 BCD 码	A(HEX 码)→A(BCD 码)	1	1	C
DSA	BCD 码减法操作后，将 A 的值调整为 BCD 码	A(HEX 码)→A(BCD 码)	1	1	-

CLRWDT	将看门狗计数器清零	0→WDTCNT	1	1	TO,PD
STOP	进入低功耗模式	0→WDTCNT；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 FLASH 型存储器，4K×16 位的地址空间范围为 0000H~0FFFH。程序存储器地址分配如下图所示：



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。

数据存储器还包括掉电非易失的 EEPROM 型数据存储器（256 字节），需通过 SFR 进行读写操作，详细说明请参见后续章节。

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	INTE0	INTF0	OSCMR	HIRCCAL	INTE1	INTF1
190H-197H		IOP0	OEP0	PUP0	PDP0	IOP1	OEP1	PUP1	PDP1
198H-19FH		IOP2	OEP2	PUP2	PDP2	P1KBCR	OPACR0	OPACR1	
1A0H-1A7H		T0CR	T0CNT	T0LOAD		LVDCR	CMPCR0	CMPCR1	CMPCLR
1A8H-1AFH		T1CR0	T1CR1	T1CNTH	T1CNTL	T1LOADH	T1LOADL		
1B0H-1B7H		T2CR	T2CNT	T2LOAD	T2DATA	PWM2CR0	PWM2CR1	PWM2CR2	
1B8H-1BFH		T3CR	T3CNT	T3LOAD	T3DATA	T3DATB	T3DATC	PWM3CR0	PWM3CR1
1C0H-1C7H		PWM3CR2	PWM3CR3	PWM3CR4		EECR	EEPR	EEAR	EEDR
1C8H-1CFH		ADCR0	ADCR1	ADRH	ADRL	OSADJCR	P0ADCR	P1ADCR	P2ADCR
1D0H-1FFH	保留								

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

数据存储器寻址方式地址组成

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	寻址方式
/	/	/	/	/	/	/	取自指令的 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 0

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR0	FSR07	FSR06	FSR05	FSR04	FSR03	FSR02	FSR01	FSR00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **FSR0[7:0]** – 数据指针寄存器 0

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

数据指针寄存器 1

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
FSR1	FSR17	FSR16	FSR15	FSR14	FSR13	FSR12	FSR11	FSR10
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **FSR1[7:0]** – 数据指针寄存器 1

FSR1：间接寻址方式 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 0

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF0	INDF07	INDF06	INDF05	INDF04	INDF03	INDF02	INDF01	INDF00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **INDF0[7:0]** – 间接寻址寄存器 0

INDF0：INDF0 不是物理寄存器，对 INDF0 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。

间接寻址寄存器 1

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF1	INDF17	INDF16	INDF15	INDF14	INDF13	INDF12	INDF11	INDF10
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **INDF1[7:0]** – 间接寻址寄存器 1

INDF1：INDF1 不是物理寄存器，对 INDF1 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。

间接寻址寄存器 2

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF2	INDF27	INDF26	INDF25	INDF24	INDF23	INDF22	INDF21	INDF20
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **INDF2[7:0]** – 间接寻址寄存器 2

INDF2：INDF2 不是物理寄存器，对 INDF2 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。

间接寻址寄存器 3

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INDF3	INDF37	INDF36	INDF35	INDF34	INDF33	INDF32	INDF31	INDF30
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

初始值	X	X	X	X	X	X	X	X

BIT[7:0] **INDF3[7:0]** – 间接寻址寄存器 3

INDF3：INDF3 不是物理寄存器，对 INDF3 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：仅可对寄存器 INDF3 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 = 指令码低 13 位；
- ✧ 返回指令 RETIE/RETURN/RETAI：PC = 堆栈栈顶（TOS）；

对 PCL 操作指令：

- ✧ 对 PCL 操作的加法指令：PC = (PC[12:0]+ALU[7:0])；
- ✧ 对 PCL 操作的其他指令：PC = (PC[12:8]:ALU[7:0](ALU 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： 加法运算后半字节无进位；减法运算后半字节有借位；
1： 加法运算后半字节有进位；减法运算后半字节无借位；

BIT[0] **C** – 进位/借位标志位
0： 加法运算后无进位；减法运算后有借位；移位操作后移出位为 0；
1： 加法运算后有进位；减法运算后无借位；移位操作后移出位为 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
OSCM	系统时钟振荡模式设置： 内部高频 RC 振荡器 HIRC + 内部低频 RC 振荡器 LIRC； 内部高频 RC 振荡器 HIRC + 外部 32768Hz 晶体振荡器 LEXT； 外部 16MHz 晶体振荡器 HEXT + 内部低频 RC 振荡器 LIRC； 外部 8MHz 晶体振荡器 HEXT + 内部低频 RC 振荡器 LIRC； 外部 4MHz 晶体振荡器 HEXT + 内部低频 RC 振荡器 LIRC； 外部 455KHz 晶体振荡器 HEXT + 内部低频 RC 振荡器 LIRC；
HIRCFS	HIRC 振荡频率选择： FHIRC=32MHz； FHIRC=2MHz；
FCPUS	高频时钟下 FCPU 分频选择： FCPU=FHOSC/2； FCPU=FHOSC/4； FCPU=FHOSC/8； FCPU=FHOSC/16； FCPU=FHOSC/32； FCPU=FHOSC/64；
RSTEN	RST 外部复位端口设置： P17 为外部复位脚； P17 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） 2.0V；2.4V；2.7V；3.2V；
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；
ENCR	程序代码加密设置： 程序代码加密； 程序代码不加密；
DBGPIN0	PCK0/PDT0 编程扩展设置： 端口仅在复位时用作编程端口，芯片仅在复位时才可进入编程/仿真模式； 端口始终用作编程端口，芯片随时可借助编程器进入编程/仿真模式；
DBGPIN1	PCK1/PDT1 编程扩展设置： 端口仅在复位时用作编程端口，芯片仅在复位时才可进入编程/仿真模式； 端口始终用作编程端口，芯片随时可借助编程器进入编程/仿真模式；

4 系统时钟

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

系统高频时钟 FHOSC 可通过配置字 OSCM 和 HIRCFS 选择以下时钟：

- ✧ 内部高频 RC 振荡器 HIRC（32MHz/2MHz）时钟 FHIRC；
- ✧ 外部高频晶体振荡器 HEXT（455KHz/4MHz~16MHz）时钟 FHEXT；

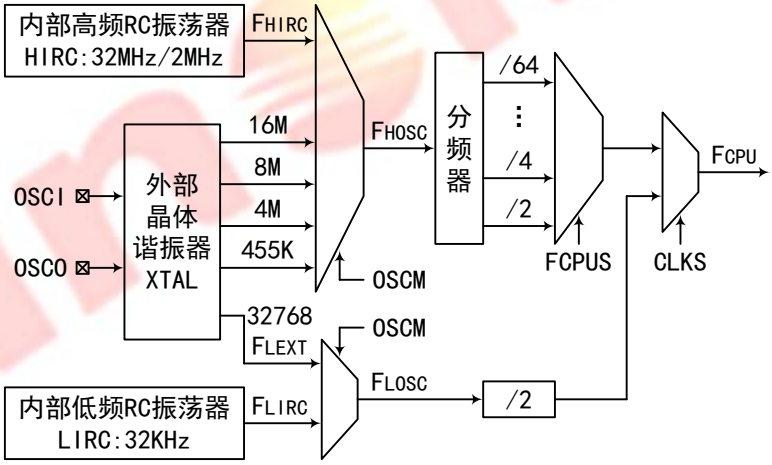
系统低频时钟 FLOSC 可通过配置字 OSCM 选择以下时钟：

- ✧ 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（仅在 FHOSC 为 FHIRC 时可选）；

CPU 的时钟源可在系统高频时钟 FHOSC 和系统低频时钟 FLOSC 之间切换。FHOSC 下 CPU 的时钟频率 FCPU 通过配置字 FCPUS 选择；FLOSC 下 FCPU 则固定为 FLOSC 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 个振荡频率可通过配置字 HIRCFS 选择（32MHz/2MHz）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/2MHz。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/2MHz -10%）~（32MHz/2MHz +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/2MHz）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 455KHz/4MHz~16MHz 的晶体振荡器作为系统高频时钟源，还可外接 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	3.2
8M	10/20	2.4
4M	10/20	2.0
455K	100/220	2.0
32768	0/10	2.0

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、HOLD 模式和休眠模式等多种系统工作模式。

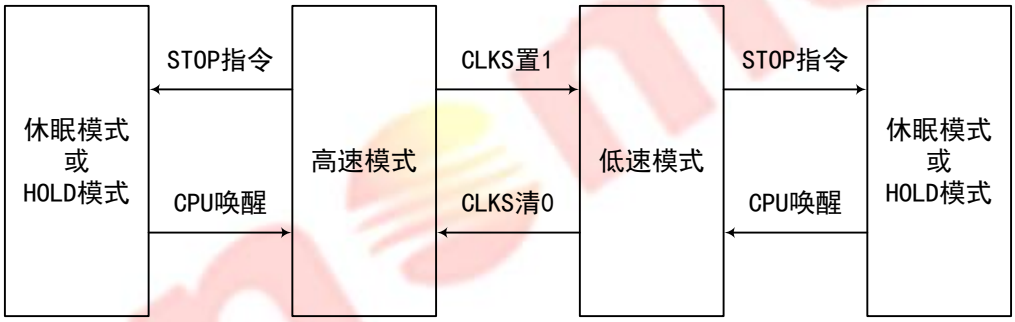
工作模式	模式切换条件	系统工作状态
高速	任意模式下，系统复位	CPU 高速运行，高/低频时钟源均工作
	低速模式下，CLKS 清 0	
	HOLD/休眠模式下，CPU 唤醒（@CLKS=0）	
低速	高速模式下，CLKS 置 1	CPU 低速运行，低频时钟源工作，高频时钟源由使能位 HFEN 决定
	HOLD/休眠模式下，CPU 唤醒（@CLKS=1）	
HOLD	高/低速模式下，执行 STOP 指令（@HFEN=0,LFEN=1）	CPU 暂停，高频时钟源停止，低频时钟源工作
休眠	高/低速模式下，执行 STOP 指令（@HFEN=0,LFEN=0）	CPU 暂停，高/低频时钟源均停止

注：

1、 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；

2、 禁止在 HFEN=1 时执行 STOP 指令，否则 CPU 将无法正常唤醒；

工作模式切换示意图



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、HOLD 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）；
- ✧ 比较器 CMP 中断请求发生（若低功耗模式下比较器保持继续工作）；

注：

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

5 复位

5.1 复位条件

芯片共有如下几种复位方式：

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

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

芯片发生复位后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复位延时（典型值为 16ms）以待振荡器稳定振荡后才开始工作。

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 FCPU 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 VPOR 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 VLVR 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 Vih；
(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/输出端口包括两组 8 位端口 P0、P1，和一组 2 位端口 P2。所有端口均支持施密特输入，均支持推挽输出。

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。

端口数据寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOP0	P07D	P06D	P05D	P04D	P03D	P02D	P01D	P00D
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] **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
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）

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOP2	-	-	-	-	-	-	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
OEP0	P07OE	P06OE	P05OE	P04OE	P03OE	P02OE	P01OE	P00OE
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] **P0nOE** – P0n 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；

	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；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OEP2	-	-	-	-	-	-	P21OE	P20OE
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

BIT[1:0] **P2nOE** – P2n 端口输出使能位（n=1-0）

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。

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	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	0	0	0	0	0	0	0	0

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；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PUP2	-	-	-	-	-	-	P21PU	P20PU

R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

BIT[1:0] **P2nPU** – P2n 端口上拉电阻控制位（n=1-0）
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	P07PD	P06PD	P05PD	P04PD	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	0	0	0	0	0	0	0	0

BIT[7:0] **P0nPD** – P0n 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；

	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；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PDP2	-	-	-	-	-	-	P21PD	P20PD
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

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
P0ADCR	P07DC	P06DC	P05DC	P04DC	P03DC	P02DC	P01DC	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[7:0] **P0nDC** – P0n 端口数字功能控制位（n=7-0）

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
P1ADCR	P17DC	P16DC	P15DC	P14DC	P13DC	-	P11DC	P10DC
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[7:3,1:0] **P1nDC** – P1n 端口数字功能控制位（n=7-3,1-0）

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
P2ADCR	-	-	-	-	-	-	P21DC	P20DC
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

BIT[1:0] **P2nDC** – P2n 端口数字功能控制位（n=1-0）

0： 使能端口的数字 I/O 功能；

1： 关闭端口的数字 I/O 功能；

7 定时器 TIMER

7.1 看门狗定时器 WDT

看门狗定时器 WDT 的时钟源为内部低频 RC 振荡器 LIRC，WDT 溢出将复位芯片或唤醒 CPU。

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

执行 CLRWDT 指令或 STOP 指令将清零 WDT 计数器。

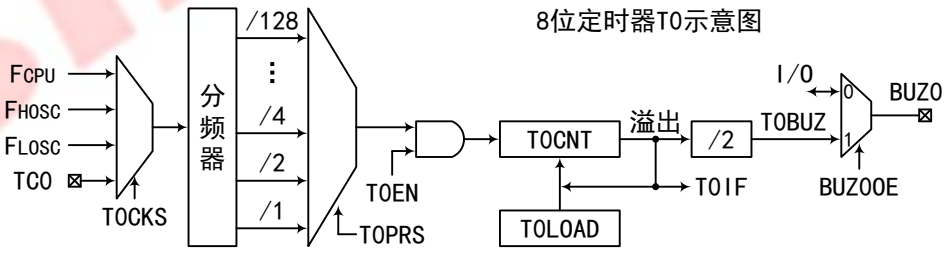
WDT 溢出时间可配置为 16ms/64ms/256ms/1024ms。

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

7.2 定时器 T0

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

- ✧ 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 BUZ 输出；
- ✧ 支持溢出中断和溢出唤醒功能；



定时器 T0，可通过寄存器位 T0CKS 选择时钟源，通过 T0PRS 选择时钟预分频比，所选时钟源通过预分频器后产生 T0 计数器 T0CNT 的计数时钟（上升沿计数）。写 T0CNT 将清零预分频计数器，而预分频比保持不变。

T0EN=0 时，T0CNT 保持不变，写重载寄存器 T0LOAD 将立即载入 T0CNT；T0EN=1 时，T0CNT 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 T0IF 将被置 1，同时 T0 自动将当前 T0LOAD 值载入 T0CNT 并重新开始计数。

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。

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	-	BUZ0OE	T0CKS1	T0CKS0	T0PRS2	T0PRS1	T0PRS0
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] **T0EN** – 定时器 T0 使能位

0： 关闭定时器 T0；

1： 开启定时器 T0；

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；

BIT[4:3] **T0CKS[1:0]** – T0 时钟源选择位

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

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

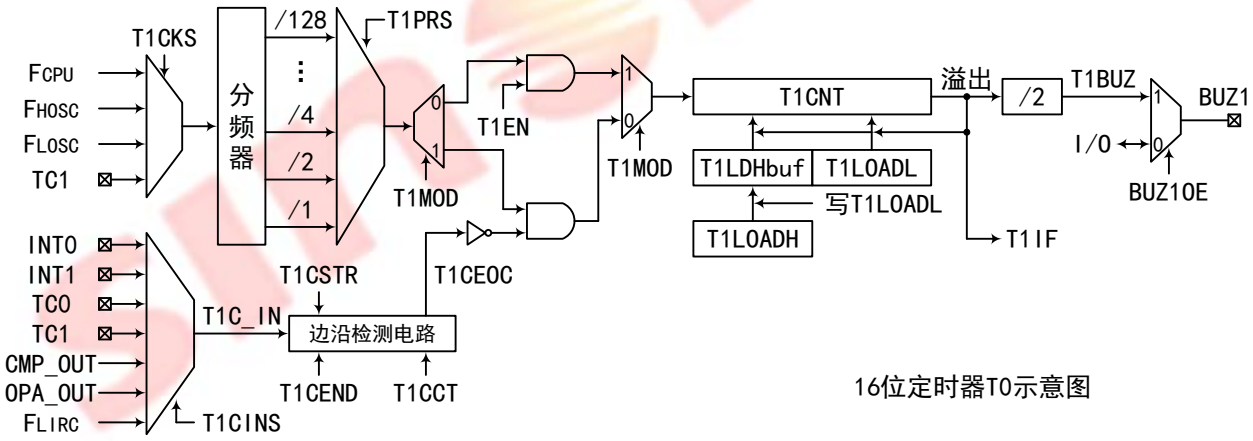
BIT[7:0] **T0LOAD[7:0]** – T0 重载寄存器，用于设置 T0 的计数周期

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。

7.3 定时器 T1

定时器 T1 为 16 位定时/计数器，包含 1 个 16 位递减计数器、可编程预分频器、控制寄存器、16 位重载寄存器。

- ✧ 可通过预分频器设置时钟频率，可通过重载寄存器控制计数周期；
- ✧ 支持内/外部信号的上升/下降沿输入捕捉功能，支持 T1 捕捉中断功能；
- ✧ 支持 BUZ 输出；
- ✧ 支持溢出中断和溢出唤醒功能；



16位定时器T0示意图

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

16 位 T1CNT 的高字节 T1CNTH 配有缓冲器，读写 T1CNTH 实际为读写缓冲器中内容，T1CNTH 实际内容仅在读写 T1CNTL 时才自动与缓冲器交互，因此写 T1CNT 需先写 T1CNTH 再写 T1CNTL（硬件同时自动将缓存器内容载入 T1CNTH），而读 T1CNT 则需先读 T1CNTL（硬件同时自动将 T1CNTH 内容载入缓冲器中）再读 T1CNTH。

T1EN=0 时，T1CNT 保持不变，写重载寄存器 T1LOAD 将立即载入 T1CNT；T1EN=1 时，T1CNT

递减计数，计数到 0 的时钟结束后产生溢出信号并触发中断，中断标志 T1IF 将被置 1，同时 T1 自动将当前 T1LOAD 值载入 T1CNT 并重新开始计数。

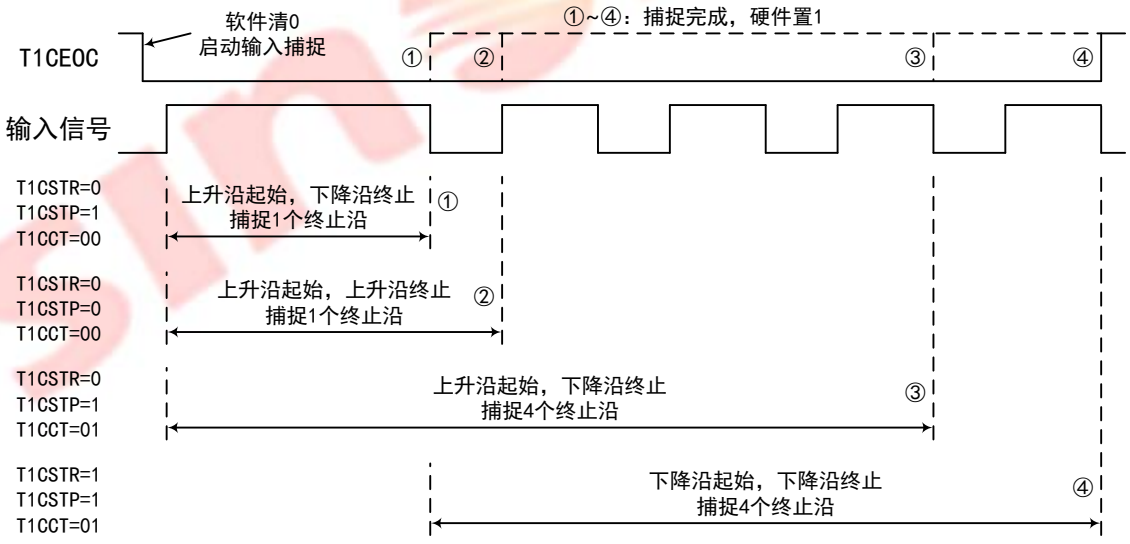
16 位 T1LOAD 的高字节 T1LOADH 配有缓冲器（T1LDHbuf），写 T1LOADL 时会同时将 T1LOADH 内容载入该缓冲器中，因此调整 T1LOAD 值时需先写 T1LOADH 再写 T1LOADL。此时若 T1EN=0，则会同时再将[缓冲器:T1LOADL]载入 T1CNT；若 T1EN=1，则会在 T1 溢出后才将[缓冲器:T1LOADL]载入 T1CNT。

如图所示，定时器 T1 可实现 BUZ 功能（BUZ1）。当 BUZ1OE=1 时，端口将输出频率为 T1 溢出频率 2 分频的蜂鸣器驱动信号。

如图所示，定时器 T1 可实现内/外部信号上升沿或下降沿的输入捕捉功能。通过寄存器位 T1MOD 选择输入捕捉模式，当 T1CEOC 为 1 时写 0 启动 T1 捕捉功能，若 T1CINS 选定的输入信号产生起始沿（T1CSTR 选择），则 T1 开始计数（若计数溢出则自动重载），当输入信号产生终止沿（T1CSTP 选择）时，T1 停止计数且 T1CNT 保持当前值，T1CEOC 自动置 1，同时中断标志 T1CIF 置 1 触发 T1 捕捉中断。通过 T1CNT 前后计数的差值，即可获得输入信号起始沿与终止沿之间的时间宽度。T1 捕捉功能还可通过 T1CCT 选择捕捉沿的个数，以获得 1 个高/低电平脉冲信号的时间宽度（起始沿和终止沿方向相反）或 1/4/8/16 个周期信号的时间宽度（起始沿和终止沿方向相同）。

输入捕捉模式中，在捕捉完成后 T1CEOC 自动置 1，需再次写 0 才能重新启动捕捉功能。在终止沿捕捉完成停止计数到起始沿再次触发恢复计数的期间，T1CNT 将一直保持原值，若要调整则需重新写 T1LOAD 或 T1CNT。

T1 输入捕捉示意



定时器 T1 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CR0	T1EN	T1MOD	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] **T1MOD** – T1 工作模式选择位
0： 定时/计数/BUZ 模式；
1： 输入捕捉模式；

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；

BIT[4:3] **T1CKS[1:0]** – T1 时钟源选择位

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

BIT[2:0] **T1PRS[2:0]** – T1 时钟预分频比选择位

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

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CR1	T1CEOC	T1CINS2	T1CINS1	T1CINS0	T1CSTR	T1CSTP	T1CCT1	T1CCT0
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] **T1CEOC** – T1 输入捕捉控制位
0： 输入捕捉等待起始沿触发或正在计数中，终止沿捕捉完成后自动置 1；
1： 输入捕捉未启动或已完成，写 0 启动输入捕捉；

BIT[6:4] **T1CINS[2:0]** – T1 输入捕捉信号源选择位

T1CINS[2:0]	T1 输入捕捉信号源
000	INT0

001	INT1
010	TC0
011	TC1
100	CMP_OUT
101	-
110	FLIRC
111	-

BIT[3] **T1CSTR** – T1 输入捕捉起始沿选择位
0： 上升沿触发捕捉计数；
1： 下降沿触发捕捉计数；

BIT[2] **T1CSTP** – T1 输入捕捉终止沿选择位
0： 上升沿终止捕捉计数；
1： 下降沿终止捕捉计数；

BIT[1:0] **T1CCT[1:0]** – T1 捕捉沿个数选择位

T1CCT[1:0]	T1 捕捉沿个数
00	1 个
01	4 个
10	8 个
11	16 个

定时器 T1 计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CNTH	T1CNT15	T1CNT14	T1CNT13	T1CNT12	T1CNT11	T1CNT10	T1CNT9	T1CNT8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

BIT[7:0] **T1CNT[15:8]** – T1 计数器高 8 位，为可读写的递减计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CNTL	T1CNT7	T1CNT6	T1CNT5	T1CNT4	T1CNT3	T1CNT2	T1CNT1	T1CNT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

BIT[7:0] **T1CNT[7:0]** – T1 计数器低 8 位，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
T1LOADH	T1LOAD15	T1LOAD14	T1LOAD13	T1LOAD12	T1LOAD11	T1LOAD10	T1LOAD9	T1LOAD8
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[15:8]** – T1 重载寄存器高 8 位，用于设置 T1 的计数周期

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1LOADL	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

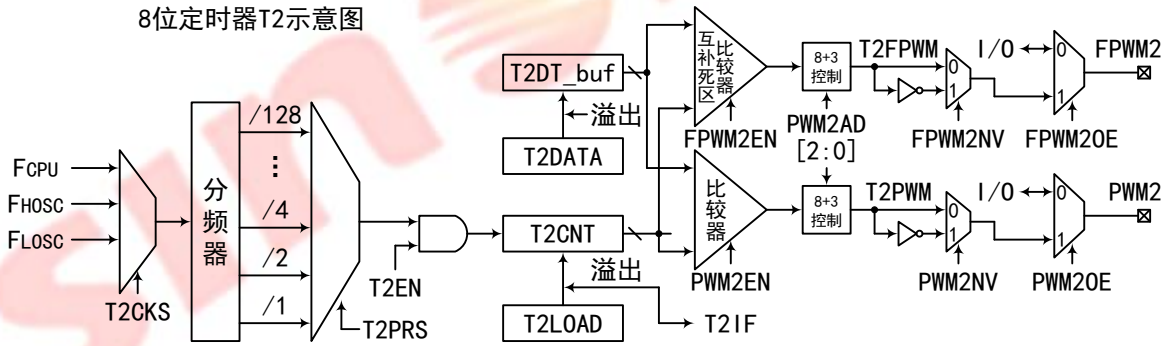
BIT[7:0] **T1LOAD[7:0]** – T1 重载寄存器低 8 位，用于设置 T1 的计数周期

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。

7.4 定时器 T2

定时器 T2 为 8 位定时器，包含 1 个 8 位递减计数器、可编程预分频器、控制寄存器、8 位重载寄存器和 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 1 对 8+3 模式的带死区互补 PWM 输出，可通过比较寄存器设置 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 选择时钟预分频比，所选时钟源通过预分频器后产生 T2 计数器 T2CNT 的计数时钟（上升沿计数）。写 T2CNT 将清零预分频计数器，而预分频比保持不变。

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

如图所示，定时器 T2 可实现 1 对互补且带死区（2 路互补信号高电平非交叠时间）控制的 PWM 功能（PWM2/FPWM2），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。PWM2 关闭时 T2PWM 信号为低电平，FPWM2 关闭时 T2FPWM 信号为高电平。PWM2/FPWM2 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 T2DATA 相等时，

T2PWM 变为高电平（T2FPWM 提前下降沿非交叠时间变为低电平）；当计数溢出时，T2PWM 变为低电平（T2FPWM 滞后上升沿非交叠时间变为高电平）。

T2DATA 配有 1 个 8 位比较缓冲器（T2DT_buf）用于与 T2CNT 比较，PWM2 和 FPWM2 均关闭时写 T2DATA 将立即载入缓冲器中，而 PWM2 或 FPWM2 使能后写 T2DATA 则将在 T2 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。

T2PWM 信号的占空比计算如下：

- ✧ 高电平时间 =（T2DATA）× T2CNT 计数时钟周期
- ✧ 周期（T2 溢出时间）=（T2LOAD + 1）× T2CNT 计数时钟周期
- ✧ 占空比（高电平时间/周期）=（T2DATA）/（T2LOAD + 1）

PWM2/FPWM2 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 T2PWM 信号将比普通周期提前半个计数时钟变为高电平，即 T2PWM 高电平时间延展为（T2DATA+0.5）。

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 PWM2D[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。

PWM2/FPWM2 的占空比可通过寄存器位 PWM2DB 微调半个计数时钟周期。PWM2DB=1 时，普通周期内 T2PWM 信号在计数时钟上升沿变为高电平，而延展周期内则提前半个时钟周期在时钟下降沿翻转；PWM2DB=0 时，普通周期内 T1PWM 信号滞后半个时钟周期在时钟下降沿变为高电平，而延展周期内则变为正常的时钟上升沿翻转。T2FPWM 信号变为低电平的时间则在叠加下降沿死区时间后进行相应微调。

PWM2 的 8+3 模式一个大周期的占空比计算如下（需 0＜T2DATA＜T2LOAD）：

- ✧ 高电平时间 = [(T2DATA+(PWM2DB-1)/2) ×8 + (PWM2D[2:0])/2] × T2CNT 计数时钟周期
- ✧ 周期 = (T2LOAD+1) ×8 × T2CNT 计数时钟周期
- ✧ 占空比（高电平时间/周期）= [T2DATA+(PWM2DB-1)/2+(PWM2D[2:0]/16)] / (T2LOAD+1)

举例说明（PWM2DB=1）：

若 PWM2D[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[(T2DATA) / (T2LOAD+1)]；

若 PWM2D[2:0]=B’001，则每 8 个 PWM 周期中的第 1 个周期进行占空比延展，一个大周期的占空比为[(T2DATA) + (1/16)] / (T2LOAD+1)；

若 PWM2D[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[(T2DATA) + (5/16)] / (T2LOAD+1)；

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	PWM2EN	FPWM2EN	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] **PWM2EN** – PWM2 使能位
0： 关闭 PWM2 功能；
1： 使能 PWM2 功能；

BIT[5] **FPWM2EN** – FPWM2 使能位
0： 关闭 FPWM2 功能；
1： 使能 FPWM2 功能；

BIT[4:3] **T2CKS[1:0]** – T2 时钟源选择位

T2CKS[1:0]	T2 时钟源
00	FCPU
01	FHOSC
10	FLOSC
11	-

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

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

定时器 T2 计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2CNT	T2CNT7	T2CNT6	T2CNT5	T2CNT4	T2CNT3	T2CNT2	T2CNT1	T2CNT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	1	1	1	1	1	1	1

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

定时器 T2 重载寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T2LOAD	T2LOAD7	T2LOAD6	T2LOAD5	T2LOAD4	T2LOAD3	T2LOAD2	T2LOAD1	T2LOAD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

初始值	1	1	1	1	1	1	1	1

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

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 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2CR0	-	-	-	-	FPWM2NV	FPWM2OE	PWM2NV	PWM2OE
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	0	0	0	0

BIT[3] **FPWM2NV** – FPWM2 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[2] **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；

BIT[1] **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[0] **PWM2OE** – PWM2 端口输出使能位
0： 禁止端口输出脉宽调制波形；
1： 允许端口输出脉宽调制波形；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM2CR1	-	-	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 1110	63 个计数时钟周期	63 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
PWM2CR2	-	-	-	-	PWM2DB	PWM2D2	PWM2D1	PWM2D0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	1	0	0	0

BIT[3] **PWM2DB** – T2PWM/T2FPWM 匹配翻转时钟沿选择位

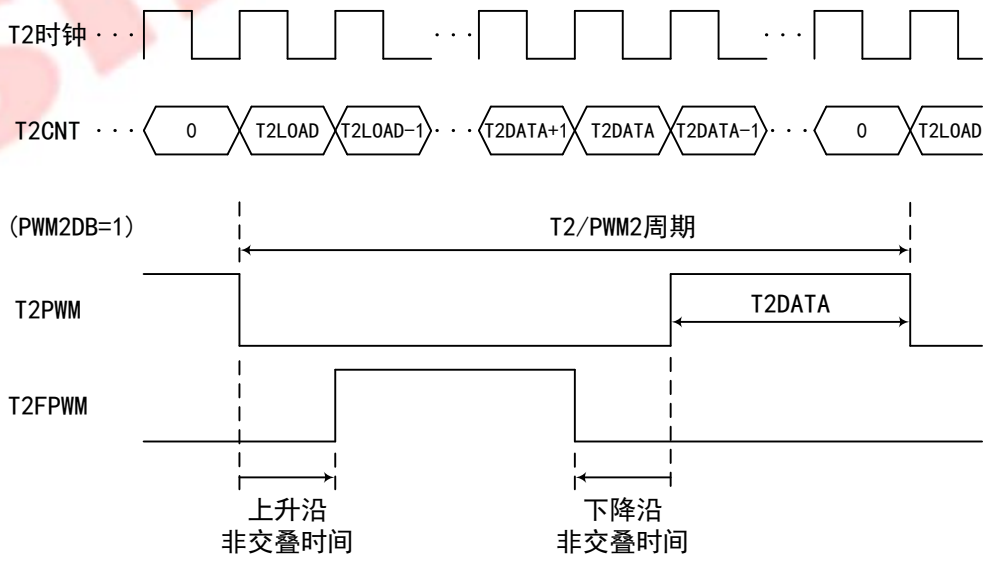
0： 占空比匹配时，脉宽调制信号电平滞后半个时钟周期翻转；

1： 占空比匹配时，脉宽调制信号电平与计数时钟的有效沿同步翻转；

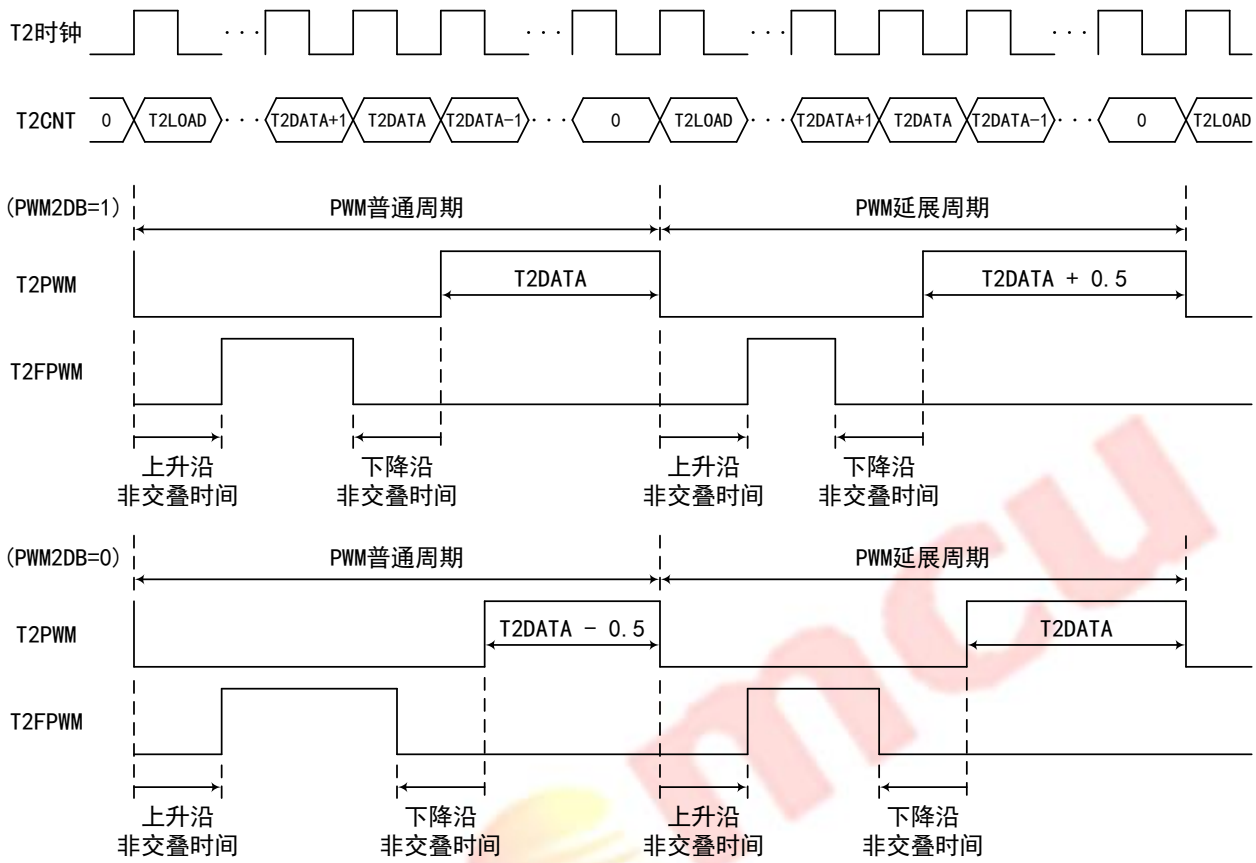
BIT[2:0] **PWM2D[2:0]** – T2PWM/T2FPWM 占空比延展控制位

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

PWM 互补及死区波形示意



PWM2DB 及 8+3 模式波形示意



注:

1. ***应用互补 PWM 时，前后死区的总时间应小于 T2PWM 低电平时间，以确保 T2FPWM 能正常生成高电平;***
2. ***不可在 PWM 工作时调整 PWM 周期（即定时器周期）和死区时间;***
3. ***8+3 模式下，当 PWM2DB=1 时，正常周期（不延展的周期）的占空比（高电平时间）为 T2DATA，而延展周期的占空比则延展半个时钟周期（即为 T2DATA+0.5）；当 PWM2DB=0 时，所有周期（包括正常周期和延展周期）的占空比均缩减半个时钟周期;***
4. ***若 T2PWM 或 T2FPWM 在 PWM2DB 为 1 时高电平时间仅为 1 个时钟周期，则 PWM2DB 清 0 操作并不会将高电平时间缩减为半个时钟周期;***

7.5 定时器 T3

定时器 T3 为 8 位定时器，包含 1 个 8 位递减计数器、可编程预分频器、控制寄存器、8 位重载寄存器和 3 个 8 位比较寄存器。

- ✧ 可通过预分频器设置时钟频率，可通过重载寄存器控制计数周期;
- ✧ 支持 3 路 8 位共周期 PWM 输出，可通过对应的比较寄存器分别设置每路 PWM 占空比;
- ✧ 其中 1 路 PWM 可扩展为 1 对 8+3 模式的带死区互补 PWM 输出，且可从 3 组端口输出并支持输出保护功能;
- ✧ 支持溢出中断和溢出唤醒功能;

定时器 T3，可通过寄存器位 T3CKS 选择时钟源，通过 T3PRS 选择时钟预分频比，所选时钟源通过预分频器后产生 T3 计数器 T3CNT 的计数时钟（上升沿计数）。写 T3CNT 将清零预分频计数器，而预分频比保持不变。

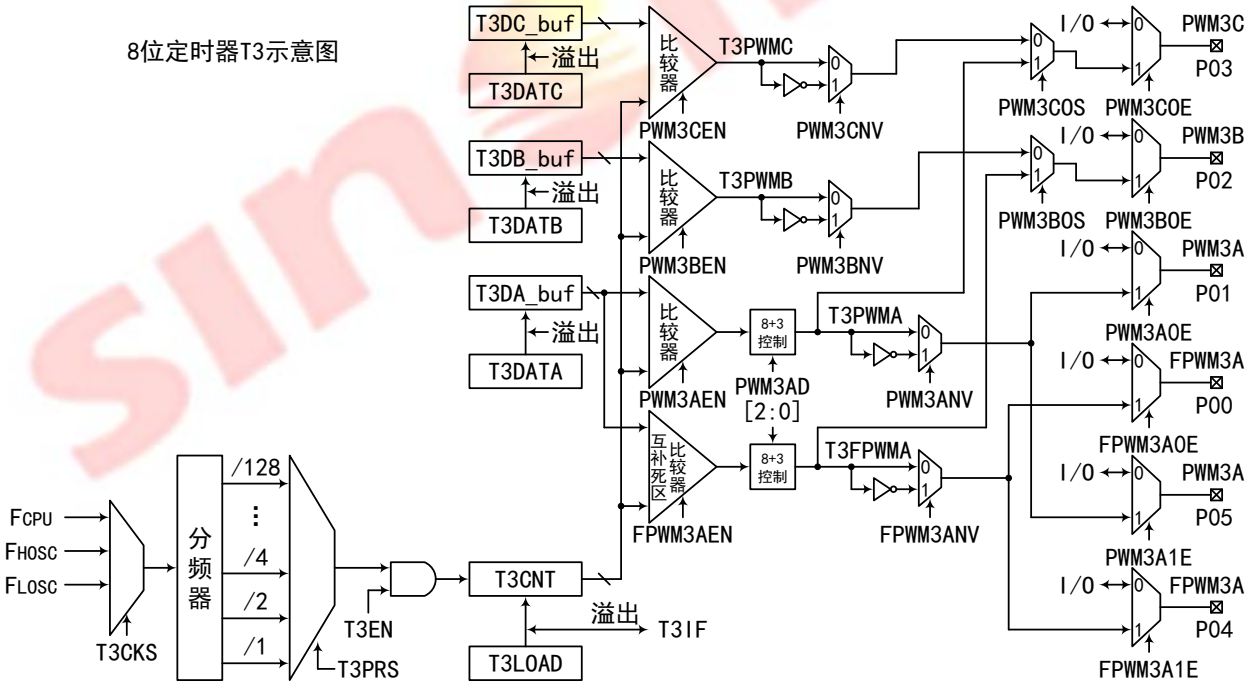
T3EN=0 时，T3CNT 保持不变，写重载寄存器 T3LOAD 将立即载入 T3CNT；T3EN=1 时，T3CNT 递减计数，计数到 0 的时钟结束后产生溢出信号并触发中断，中断标志 T3IF 将被置 1，同时 T3 自动将当前 T3LOAD 值载入 T3CNT 并重新开始计数。

如图所示，定时器 T3 可实现 3 路共周期的 PWM 功能（PWM3x，x=A,B,C，下同），可分别设置每路 PWM 占空比，可通过寄存器位使能或关闭 PWM 功能，并控制端口是否输出 PWM 波形。PWM3x 关闭时 T3PWMx 信号为低电平。PWM3x 使能后 T3CNT 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 T3DATx 相等时，T3PWMx 变为高电平；当计数溢出时，T3PWMx 变为低电平。

T3DATx 均配有 1 个 8 位比较缓冲器（T3Dx_buf）用于与 T3CNT 比较，PWM3x 关闭时写 T3DATx 将立即载入缓冲器中，而 PWM3x 使能后写 T3DATx 则将在 T3 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。

T3PWMx 信号（x=A,B,C，下同）的占空比计算如下：

- ✧ 高电平时间 =（T3DATx）× T3CNT 计数时钟周期
- ✧ 周期（T3 溢出时间）=（T3LOAD + 1）× T3CNT 计数时钟周期
- ✧ 占空比（高电平时间/周期）=（T3DATx）/（T3LOAD + 1）



如图所示，定时器 T3 还可实现 1 路与 PWM3A 互补且带死区控制的互补 PWM 功能（FPWM3A），可通过单独的寄存器位使能互补 PWM 功能，并控制端口是否输出互补 PWM 波形。PWM3A/FPWM3A 支持 8+3 模式和半个计数时钟周期的微调功能，其工作方法与 PWM2/FPWM2 完全相同。

PWM3A/FPWM3A 不仅可从 3 组端口同时输出，还支持输出保护功能，即当比较器 CMP 的输出电平发生变化时触发 PWM 输出保护（可选择变为高电平或变为低电平时保护），3 组端口将立即停止输出 PWM 波形，输出保护控制位 PWM3APF 自动置 1，端口转为通用 I/O 端口。只有 CMP 的输出电平恢复为低电平（或高电平）后，PWM3APF 才可清 0，允许 3 组 PWM3A/FPWM3A 端口从下一个 PWM 周期开始重新输出 PWM 波形。

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	PWM3AEN	FPWM3AEN	T3CKS1	T3CKS0	T3PRS2	T3PRS1	T3PRS0
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] **T3EN** – 定时器 T3 使能位
0： 关闭定时器 T3；
1： 开启定时器 T3；

BIT[6] **PWM3AEN** – PWM3A 使能位
0： 关闭 PWM3A 功能；
1： 使能 PWM3A 功能；

BIT[5] **FPWM3AEN** – FPWM3A 使能位
0： 关闭 FPWM3A 功能；
1： 使能 FPWM3A 功能；

BIT[4:3] **T3CKS[1:0]** – T3 时钟源选择位

T3CKS[1:0]	T3 时钟源
00	FCPU
01	FHOSC
10	FLOSC
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。

定时器 T3 比较寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3DATA	T3DATA7	T3DATA6	T3DATA5	T3DATA4	T3DATA3	T3DATA2	T3DATA1	T3DATA0
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] **T3DATA[7:0]** – T3 比较寄存器 A，用于设置 PWM3A 的占空比

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3DATB	T3DATB7	T3DATB6	T3DATB5	T3DATB4	T3DATB3	T3DATB2	T3DATB1	T3DATB0
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] **T3DATB[7:0]** – T3 比较寄存器 B，用于设置 PWM3B 的占空比

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T3DATC	T3DATC7	T3DATC6	T3DATC5	T3DATC4	T3DATC3	T3DATC2	T3DATC1	T3DATC0
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] **T3DATC[7:0]** – T3 比较寄存器 C，用于设置 PWM3C 的占空比

PWM3 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM3CR0	PWM3CEN	PWM3BEN	PWM3COS	PWM3BOS	PWM3CNV	PWM3BNV	PWM3ANV	FPWM3ANV
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] **PWM3CEN** – PWM3C 使能位
0： 关闭 PWM3C 功能；
1： 使能 PWM3C 功能；

BIT[6] **PWM3BEN** – PWM3B 使能位
0： 关闭 PWM3B 功能；
1： 使能 PWM3B 功能；

BIT[5] **PWM3COS** – PWM3C 端口输出信号源选择位
0： 端口输出信号源为 T3PWMC；
1： 端口输出信号源为 T3PWMA；

BIT[4] **PWM3BOS** – PWM3B 端口输出信号源选择位
0： 端口输出信号源为 T3PWMB；
1： 端口输出信号源为 T3FPWMA；

BIT[3] **PWM3CNV** – T3PWMC 信号取反控制位
0： 脉宽调制信号电平不取反；
1： 脉宽调制信号电平取反；

BIT[2] **PWM3BNV** – T3PWMB 信号取反控制位
0： 脉宽调制信号电平不取反；
1： 脉宽调制信号电平取反；

BIT[1] **PWM3ANV** – T3PWMA 信号取反控制位
0： 脉宽调制信号电平不取反；
1： 脉宽调制信号电平取反；

注：控制位 PWM3ANV 仅对从 P01,P05 输出的 T3PWMA 信号有效，而对从 P03（端口 PWM3C）输出的 T3PWMA 信号无效。

BIT[0] **FPWM3ANV** – T3FPWMA 信号取反控制位
0： 脉宽调制信号电平不取反；
1： 脉宽调制信号电平取反；

注：控制位 FPWM3ANV 仅对从 P00,P04 输出的 T3FPWMA 信号有效，而对从 P02（端口 PWM3B）输出的 T3FPWMA 信号无效。

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM3CR1	-	-	PWM3A1E	FPWM3A1E	PWM3COE	PWM3BOE	PWM3A0E	FPWM3A0E
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] **PWM3A1E** – PWM3A（P05）端口输出使能位
0： 禁止端口输出脉宽调制波形；
1： 允许端口输出脉宽调制波形；

BIT[4] **FPWM3A1E** – FPWM3A（P04）端口输出使能位
0： 禁止端口输出脉宽调制波形；
1： 允许端口输出脉宽调制波形；

BIT[3] **PWM3COE** – PWM3C（P03）端口输出使能位
0： 禁止端口输出脉宽调制波形；
1： 允许端口输出脉宽调制波形；

BIT[2] **PWM3BOE** – PWM3B（P02）端口输出使能位
0： 禁止端口输出脉宽调制波形；
1： 允许端口输出脉宽调制波形；

BIT[1] **PWM3A0E** – PWM3A（P01）端口输出使能位
0： 禁止端口输出脉宽调制波形；
1： 允许端口输出脉宽调制波形；

BIT[0] **FPWM3A0E** – FPWM3A（P00）端口输出使能位
0： 禁止端口输出脉宽调制波形；
1： 允许端口输出脉宽调制波形；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM3CR2	-	-	-	PWM3APF	保留	保留	PWM3AP0E	PWM3AP0T
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4] **PWM3APF** – PWM3A/FPWM3A 输出保护控制位
0： PWM3A/FPWM3A 正常输出中，输出保护触发后自动置 1；
1： PWM3A/FPWM3A 输出保护中，写 0 恢复正常输出（需保护条件解除）；

BIT[3,2] 保留位，需固定写“0”

BIT[1] **PWM3AP0E** – CMP 输出保护触发使能位
0： 禁止 CMP 触发输出保护；
1： 允许 CMP 触发输出保护；

BIT[0] **PWM3AP0T** – CMP 输出保护触发电平选择位

0： CMP_OUT 变为高电平时（上升沿）触发输出保护，变为低电平后解除保护；

1： CMP_OUT 变为低电平时（下降沿）触发输出保护，变为高电平后解除保护；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWM3CR3	-	-	FPWM3AE5	FPWM3AE4	FPWM3AE3	FPWM3AE2	FPWM3AE1	FPWM3AE0
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] **FPWM3AE[5:0]** – T3FPWMA 上升/下降沿非交叠时间选择位

FPWM3AE[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
PWM3CR4	-	-	-	-	PWM3ADB	PWM3AD2	PWM3AD1	PWM3AD0
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	1	0	0	0

BIT[3] **PWM3ADB** – T3PWMA/T3FPWMA 匹配翻转时钟沿选择位

0： 占空比匹配时，脉宽调制信号电平滞后半个时钟周期翻转；

1： 占空比匹配时，脉宽调制信号电平与计数时钟的有效沿同步翻转；

BIT[2:0] **PWM3AD[2:0]** – T3PWMA/T3FPWMA 占空比延展控制位

PWM3AD[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 模数转换器 ADC

8.1 ADC 概述

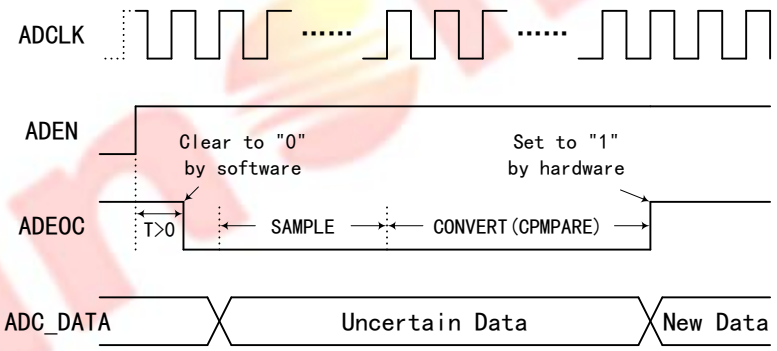
芯片内置 1 个 12 位高精度逐次逼近型的模数转换器 ADC。

- ✧ 14 路外部通道：AN0~AN13；2 路内部通道：GND、VDD/4；
- ✧ 参考电压可选：VDD、内部参考电压 VIR（2V/3V/4V）、外部参考电压 VER（VERI 输入）；
- ✧ ADC 时钟：FCPU 的 2/4/8/16 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）时间可选择 2/4/8/15 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 14/16/20/27 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 VIR，则需保证 VDD > （VIR+0.5V），否则 VIR 实际电压将降为（VDD-0.5V）；***
3. ***使能 ADC 模块、或切换参考电压等操作后，需延时（时间 > 200μs）以待电路稳定后才可启动 AD 转换；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；***
4. ***AD 转换精度受参考电压精度的影响，且内部参考电压下的转换精度，比外部参考电压下略低 2 个 LSB 左右；***
5. ***转换时钟越慢、采样时间越长，则越能过滤外部输入的波动，越能保证 AD 转换的精度；***

8.2 ADC 相关寄存器

ADC 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCR0	ADEN	ADEOC	ADCKS1	ADCKS0	ADCHS3	ADCHS2	ADCHS1	ADCHS0
R/W	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	1

BIT[7] **ADEN** – ADC 使能位

0： 关闭 ADC；

1： 开启 ADC；

BIT[6] **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[5:4] **ADCKS[1:0]** – ADC 转换时钟选择位

ADCKS[1:0]	ADC 转换时钟 FADC
00	FCPU/2
01	FCPU/4
10	FCPU/8
11	FCPU/16

注：ADC 转换时钟不能大于 1MHz。

BIT[3:0] **ADCHS[3:0]** – ADC 模拟输入通道选择位

ADCHS[3:0]	ADC 模拟输入通道	ADCHS[3:0]	ADC 模拟输入通道
0000	AN0	1000	AN8
0001	AN1	1001	AN9
0010	AN2	1010	AN10
0011	AN3	1011	AN11
0100	AN4	1100	AN12
0101	AN5	1101	AN13
0110	AN6	1110	VDD/4
0111	AN7	1111	GND

注：AN12,AN13 的输入电压不仅不能高于最大量程 VREF，还因端口内部电路的限制而不能高于（VDD-1.4V)。

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCR1	ADRSEL	ADVRS2	ADVRS1	ADVRS0	-	-	ADSPS1	ADSPS0
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] **ADRSEL** – ADC 转换结果数据格式选择位

0： 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]；

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

BIT[6] **ADVRS2** – ADC 参考电压选择位

0： ADC 参考电压由 ADVRS[1:0]决定；

1： ADC 参考电压为外部参考电压 VER（端口 VERI 输入电压）；

BIT[5:4] **ADVRS[1:0]** – ADC 参考电压选择位

ADVRS[1:0]	ADC 参考电压
00	VDD
01	内部 4.0V
10	内部 3.0V
11	内部 2.0V

BIT[1:0] **ADSPS[1:0]** – ADC 采样时间选择位

ADSPS[1:0]	ADC 采样时间
00	15 个 ADCLK
01	8 个 ADCLK
10	4 个 ADCLK
11	2 个 ADCLK

ADC 转换结果寄存器

ADRSEL=0：

	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]** – 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

ADRSEL=1：

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRH	-	-	-	-	ADR11	ADR10	ADR9	ADR8

R/W	-	-	-	-	R	R	R	R
初始值	-	-	-	-	X	X	X	X

BIT[3:0] **ADR[11:8]** – 12 位 ADC 转换结果高 4 位

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRL	ADR7	ADR6	ADR5	ADR4	ADR3	ADR2	ADR1	ADR0
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[7:0]** – 12 位 ADC 转换结果低 8 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×VREF/4096
00 0010	2×VREF/4096
…	…
00 1111	15×VREF/4096
01 0000	16×VREF/4096
…	…
11 1110	62×VREF/4096
11 1111	63×VREF/4096

8.3 ADC 操作步骤

模数转换操作步骤：

(1) 设置相应端口为输入端口，关闭端口的内部上/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；

8.4 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。

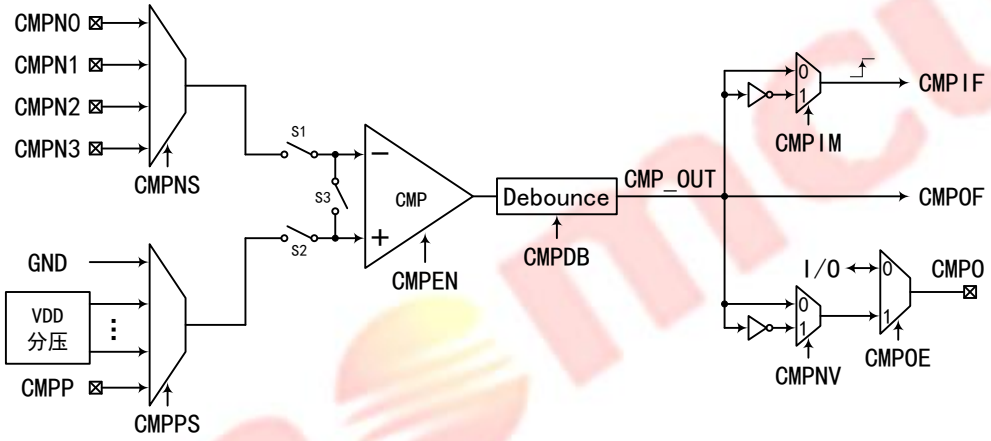
9 模拟比较器 CMP

9.1 CMP 概述

芯片内置 1 个模拟比较器 CMP，共模输入范围为 0~（VDD-1.4V），输入端失调电压可通过自消除寄存器进行调校，精度为±2mV，输出无回滞且支持去抖处理。

CMP 的正端输入可通过 CMPPS 选择外部输入电压、GND、或 VDD 内部分压电压；负端输入可通过 CMPNS 选择多路外部输入电压。

CMP 的输出端电平可通过 CMPDB 选择去抖处理，去抖后的输出信号 CMP_OUT 可从端口输出且支持输出取反，并可选择上升沿或下降沿触发 CMP 中断。



9.2 CMP 相关寄存器

比较器控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPCR0	CMPEN	CMPDB2	CMPDB1	CMPDB0	CMPNV	CMPOE	CMPIM	CMPOF
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
初始值	0	0	0	0	0	0	0	X

BIT[7] **CMPEN** – 比较器 CMP 使能位

0： 关闭 CMP；

1： 开启 CMP；

BIT[6:4] **CMPDB[2:0]** – CMP 输出去抖时间选择位

CMPDB[2:0]	CMP 输出去抖时间
000	0（不去抖）

001	1 个 FCPU 时钟周期
010	2 个 FCPU 时钟周期
011	4 个 FCPU 时钟周期
100	8 个 FCPU 时钟周期
101	12 个 FCPU 时钟周期
110	16 个 FCPU 时钟周期
111	32 个 FCPU 时钟周期

BIT[3] **CMPNV** – CMPO 端口输出取反控制位
0： 端口输出 CMP 比较结果的正向电平；
1： 端口对 CMP 比较结果取反后输出；

BIT[2] **CMPOE** – CMPO 端口输出使能位
0： 禁止端口输出 CMP 比较结果；
1： 允许端口输出 CMP 比较结果（仅在端口为输出口时有效）；

BIT[1] **CMPIM** – CMP 中断触发方式选择位
0： CMP 输出信号 CMP_OUT 的上升沿触发中断；
1： CMP 输出信号 CMP_OUT 的下降沿触发中断；

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

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPCR1	-	-	CMPPS3	CMPPS2	CMPPS1	CMPPS0	CMPNS1	CMPNS0
R/W	-	-	R/W	R/W	R/W	R/W	R/W	R/W
初始值	-	-	0	0	0	0	0	0

BIT[5:2] **CMPPS[3:0]** – CMP 正端输入选择位

CMPPS[3:0]	CMP 正端输入	CMPPS[3:0]	CMP 正端输入
0000	GND	1000	0.40×VDD
0001	0.05×VDD	1001	0.45×VDD
0010	0.10×VDD	1010	0.50×VDD
0011	0.15×VDD	1011	0.55×VDD
0100	0.20×VDD	1100	0.60×VDD
0101	0.25×VDD	1101	0.65×VDD
0110	0.30×VDD	1110	0.70×VDD
0111	0.35×VDD	1111	外部 CMPP 输入

BIT[1:0] **CMPNS[1:0]** – CMP 负端输入选择位

CMPNS[1:0]	CMP 负端输入
00	外部 CMPN0 输入
01	外部 CMPN1 输入
10	外部 CMPN2 输入
11	外部 CMPN3 输入

比较器自消除寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CMPCLR	CMPCLRE	CMPCLRS	CMPCLR5	CMPCLR4	CMPCLR3	CMPCLR2	CMPCLR1	CMPCLR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	1	0	0	0	0	0

BIT[7] **CMPCLRE** – CMP 自消除模式使能位
0： 关闭 CMP 自消除模式；
1： 使能 CMP 自消除模式；

BIT[6] **CMPCLRS** – CMP 自消除参考源选择位
0： CMP 自消除参考源为其负端输入；
1： CMP 自消除参考源为其正端输入；

注：自消除参考源应选择电压较为稳定的一路输入端，且参考源输入端的正、负极性不影响调校效果。

BIT[5:0] **CMPCLR[5:0]** – CMP 自消除校准位

9.3 CMP 失调电压调校流程

(1) 设置 CMPCLRE=1，使能失调电压自消除模式；
(2) 设置 CMPCLRS，选择比较器自消除参考源；
(3) 设置 CMPCLR[5:0]=00H，等待至少 100μs 后读取标志位 CMPOF；
(4) CMPCLR[5:0]自加 1，等待至少 100μs 后重新读取 CMPOF：
- ✧ 若 CMPOF 发生翻转，则表示已调校到位；
- ✧ 若 CMPCLR[5:0]=3FH 而 CMPOF 仍未翻转，则表示共模输入超出范围，调校失败；

(5) 循环执行（4），直到调校到位或调校失败；
(6) 判断当前调校输出方向：
- ✧ 若 CMPCLR[5]=1，则 CMPCLR[4:0]保持当前值不变；
- ✧ 若 CMPCLR[5]=0，则 CMPCLR[4:0]减 1 作为校准值；

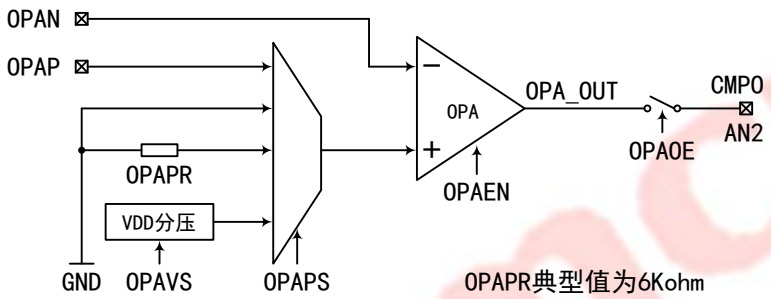
(7) 设置 CMPCLRE=0，关闭自消除模式；

10 运算放大器 OPA

10.1 OPA 概述

芯片内置 1 个运算放大器 OPA，开环放大倍数为 60dB，共模输入范围为 0~（VDD-1.4V）。

OPA 的正端输入可通过 OPAPS 选择外部输入电压、GND、或 VDD 内部分压电压；负端输入固定为端口 OPAN 的外部输入电压。



10.2 OPA 相关寄存器

OPA 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPACR0	OPAEN	保留	保留	保留	保留	OPAOE	保留	保留
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
初始值	0	0	0	0	0	0	0	X

BIT[7] **OPAEN** – 运算放大器 OPA 使能位
0： 关闭 OPA；
1： 开启 OPA；

BIT[6:3] 保留位，需固定写“0”

BIT[2] **OPAOE** – OPAO 端口输出使能位
0： 禁止端口输出 OPA_OUT 信号；
1： 允许端口输出 OPA_OUT 信号；

BIT[1:0] 保留位，需固定写“0”

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OPACR1	-	-	保留	OPAPS1	OPAPS0	OPAVS2	OPAVS1	OPAVS0
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] 保留位，需固定写“0”

BIT[4:3] **OPAPS[1:0]** – OPA 正端输入选择位

OPAPS[1:0]	OPA 正端输入
00	外部 OPAP 输入
01	GND
10	内部接地电阻 OPAPR
11	VDD 内部分压

BIT[2:0] **OPAVS[2:0]** – OPA 正端内部分压电压选择位

OPAVS[2:0]	OPA 正端内部分压电压
000	0.05×VDD
001	0.10×VDD
010	0.20×VDD
011	0.30×VDD
100	0.40×VDD
101	0.50×VDD
110	0.60×VDD
111	0.70×VDD

11 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 选择电压检测阀值。当 VDD 电压降至电压检测阀值以下时检测状态标志位 LVDF 将被置 1，同时中断标志 LVDIF 置 1 触发 LVD 中断；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阀值+6%后 LVDF 才被清 0（但中断标志不会自动清 0）。

注：开启 LVD 或切换电压检测阀值等操作，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	-	LVDVS2	LVDVS1	LVDVS0	-	-	LVDF
R/W	R/W	-	R/W	R/W	R/W	-	-	R
初始值	0	-	0	0	0	-	-	X

BIT[7] **LVDEN** – 低电压检测 LVD 使能位

0： 关闭 LVD；

1： 开启 LVD；

BIT[5:3] **LVDVS[2:0]** – LVD 电压检测阀值选择位

LVDVS[2:0]	LVD 电压检测阀值
000	2.0V
001	2.2V
010	2.4V
011	2.6V
100	2.8V
101	3.2V
110	3.6V
111	4.0V

BIT[0] **LVDF** – LVD 检测状态标志位

0： VDD 电压高于电压检测阀值，或 LVD 关闭；

1： VDD 电压低于电压检测阀值；

12 EEPROM 存储器

12.1 EEPROM 概述

芯片内置 256 字节的 EEPROM 型数据存储器，支持用户程序在带电运行中实时地读写数据。对 EEPROM 中数据的读写操作需通过控制寄存器 EECR、保护寄存器 EEPR、地址寄存器 EEAR 和数据寄存器 EEDR 进行。

写操作控制位 EEWRITE 置 1 将启动 EEPROM 写操作，EEDR 中的数据将被写入 EEAR 指向的 EEPROM 地址中，完成后 EEWRITE 自动清 0。为防止误触发 EEPROM 写操作，需先对 EEPR 写 5AH 再立即写 A5H，EEWRITE 才能置 1，中间不能插入其他操作（包括 NOP 操作），否则 EEWRITE 将无法置 1。在写 EEPR 前需先屏蔽中断，否则可能会因系统响应中断而导致 EEWRITE 无法置 1。

读操作控制位 EEREAD 置 1 将启动 EEPROM 读操作，EEAR 所指 EEPROM 地址中的数据将被读出并缓存于 EEDR，完成后 EEREAD 自动清 0。读操作需 2 个指令周期，且在 EEWRITE 为 1 时无效。

注：若在 EEPROM 的读/写操作未完成时执行 STOP 指令，则操作将被终止，控制位自动清 0。

12.2 EEPROM 相关寄存器

EEPROM 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EECR	-	-	-	-	-	-	EEREAD	EEWRITE
R/W	-	-	-	-	-	-	R/W	R/W
初始值	-	-	-	-	-	-	0	0

BIT[1] **EEREAD** – EEPROM 读操作控制位
0： 操作未开始或已完成，写 1 开始 EEPROM 读操作；
1： EEPROM 读操作中，完成后自动清 0；

BIT[0] **EEWRITE** – EEPROM 写操作控制位
0： 操作未开始或已完成，写 1 开始 EEPROM 写操作；
1： EEPROM 写操作中，完成后自动清 0；

EEPROM 保护寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EEPR	EEP7	EEP6	EEP5	EEP4	EEP3	EEP2	EEP1	EEP0
R/W	W	W	W	W	W	W	W	W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] **EEP[7:0]** – EEPROM 写操作保护控制位，需先写 5AH 再立即写 A5H，EEWRITE 才能置 1

EEPROM 地址寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EEAR	EEA7	EEA6	EEA5	EEA4	EEA3	EEA2	EEA1	EEA0
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] **EEA[7:0]** – EEPROM 读写操作的 8 位地址

EEPROM 数据寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EEDR	EED7	EED6	EED5	EED4	EED3	EED2	EED1	EED0
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] **EED[7:0]** – EEPROM 读写操作的 8 位数据

12.3 EEPROM 操作示例

例如，先将数据 55H 写入 EEPROM 存储器 10H 中，再读取 EEPROM 存储器 11H 地址中内容：

```
MOVAI   10H
MOVRA   EEAR        ; 将 10H 写入 EEAR
MOVAI   55H
MOVRA   EEDR        ; 将 55H 写入 EEDR
BCLR    GIE         ; 屏蔽中断
MOVAI   5AH
MOVRA   EEPR        ; 使能 EE 操作，第 1 步：EEPR 写 5AH
MOVAI   A5H
MOVRA   EEPR        ; 使能 EE 操作，第 2 步：EEPR 写 A5H
MOVAI   01H
MOVRA   EECR        ; 启动 EE 写操作，将数据 55H 写入 EEPROM 地址 10H 中
NOP                 ; 防止时序错误，CPU 必须冗余 2-4 个指令周期
NOP
BSET    GIE         ; 允许中断
JBCLR   EECR,0      ; 检查 EE 写操作是否完成
GOTO    $-1
MOVAI   11H
MOVRA   EEAR        ; 将 11H 写入 EEAR
MOVAI   02H
MOVRA   EECR        ; 启动 EE 读操作，读取 EEPROM 地址 11H 中内容
NOP                 ; 防止时序错误，CPU 必须冗余 2-4 个指令周期
NOP
MOVAR   EEDR        ; 从 EEDR 中读取数据
```

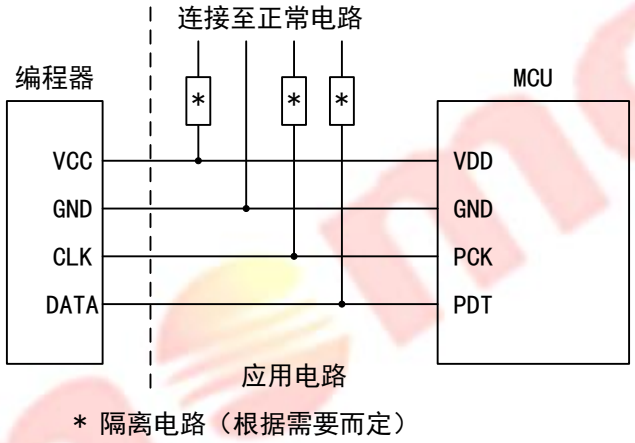
13 FLASH 烧录编程

13.1 FLASH 在板编程

芯片支持编程工具对芯片中程序存储器的在板不带电烧录编程，即在未上电的系统电路板上，借助编程工具，通过芯片的串行编程接口将用户程序代码烧录进芯片的程序存储器中。在板编程功能，可让用户先采用未编程的空芯片制造电路板而仅在产品交付前才将程序代码烧录进芯片，也方便用户直接在电路板上升级 FLASH 存储器中的程序代码。

芯片也支持对 EEPROM 型数据存储器的在板编程。

芯片的在板编程通过引脚 VDD、GND、PCK、PDT 实现，这些编程引脚的外围电路需进行针对性设计，以保证外围电路不会影响在板编程时端口上的电压/电流/时序等特性。下图是典型的在板编程连接示意图：



14 中断

芯片的中断源包括外部中断（INT0~INT1）、定时器中断（T0~T3）、T1 捕捉中断、ADC 中断、比较器中断、LVD 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。

14.1 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。

14.2 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。

14.3 T1 捕捉中断

定时器 T1 的输入捕捉模式，捕捉完成时将触发 T1 捕捉中断，中断标志 T1CIF 将被置 1，若 GIE 为 1 且 T1 捕捉中断使能位 T1CIE 为 1，则产生 T1 捕捉中断。

14.4 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；

14.5 ADC 中断

ADC 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。

14.6 比较器中断

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

14.7 LVD 中断

当 VDD 电压降至电压检测阀值以下时，将触发 LVD 中断，中断标志 LVDIF 将被置 1，若 GIE 为 1 且 LVD 中断使能位 LVDIE 为 1，则产生 LVD 中断。

14.8 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
INTE0	LVDIE	ADIE	T2IE	KBIE	INT1IE	INT0IE	T1IE	T0IE
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] **LVDIE** – LVD 中断使能位
0： 屏蔽 LVD 中断；
1： 使能 LVD 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；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTE1	-	-	-	-	T1CIE	T3IE	保留	CMPIE
R/W	-	-	-	-	R/W	R/W	R/W	R/W

初始值	-	-	-	-	0	0	0	0

BIT[3] **T1CIE** – T1 捕捉中断使能位
0： 屏蔽 T1 捕捉中断；
1： 使能 T1 捕捉中断；

BIT[2] **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[1] 保留位，需固定写“0”

BIT[0] **CMPIE** – 比较器 CMP 中断使能位
0： 屏蔽 CMP 中断；
1： 使能 CMP 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
INTF0	LVDIF	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] **LVDIF** – LVD 中断标志位
0： 未触发 LVD 中断；
1： 已触发 LVD 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；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
INTF1	-	-	-	-	T1CIF	T3IF	保留	CMPIF
R/W	-	-	-	-	R/W	R/W	R/W	R/W
初始值	-	-	-	-	0	0	0	0

BIT[3] **T1CIF** – T1 捕捉中断标志位
0： 未触发 T1 捕捉中断；
1： 已触发 T1 捕捉中断，需软件清 0；

BIT[2] **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[1] 保留位，需固定写“0”

BIT[0] **CMPIF** – 比较器 CMP 中断标志位
0： 未触发 CMP 中断；
1： 已触发 CMP 中断，需软件清 0；

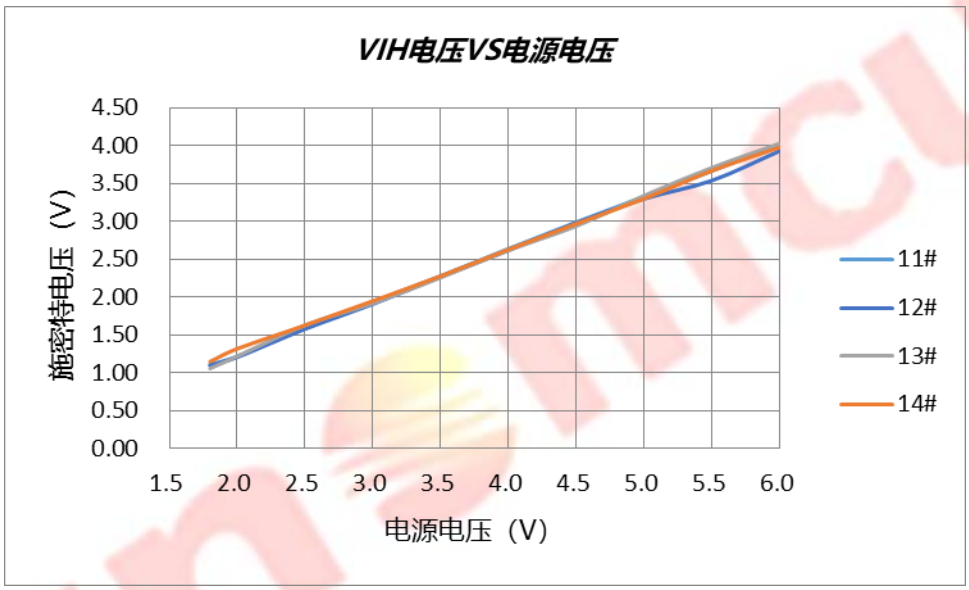
15 特性曲线

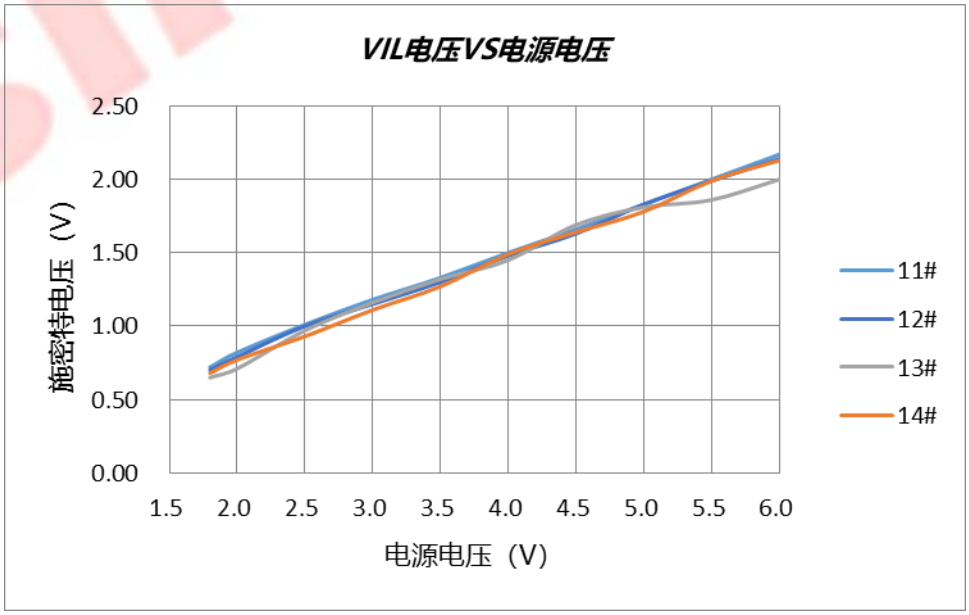
注:

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℃，温度特性曲线的电压条件为 VDD=5V;***

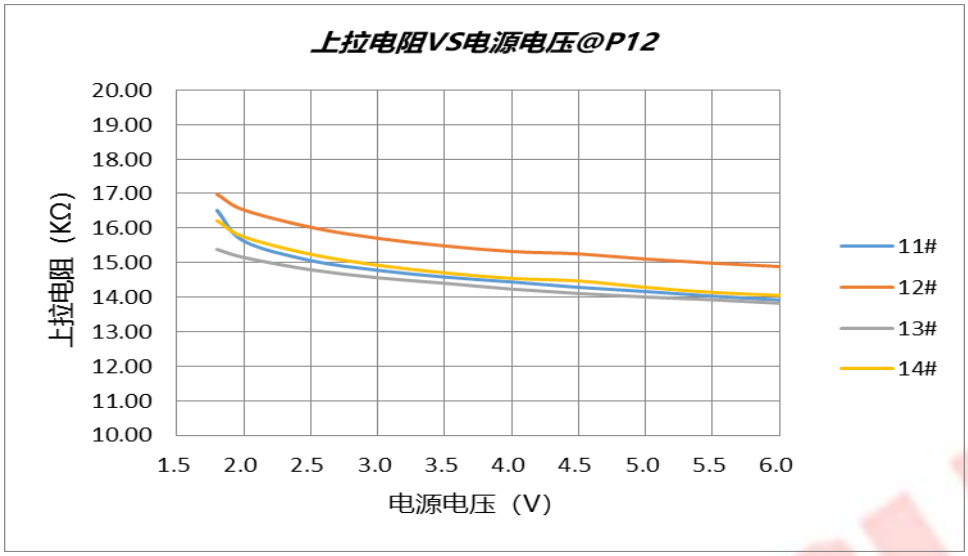
15.1 I/O 特性

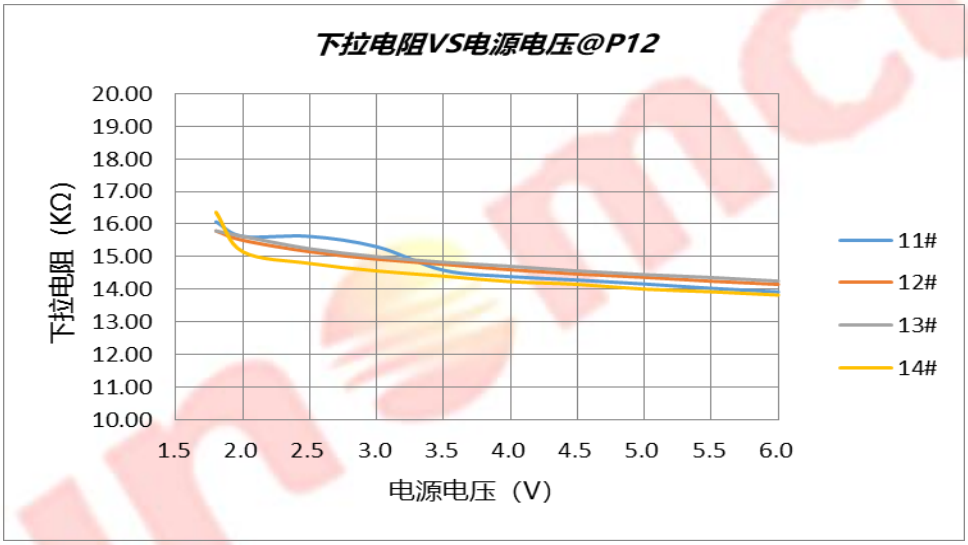
输入 SMT 阈值电压 VS 电源电压



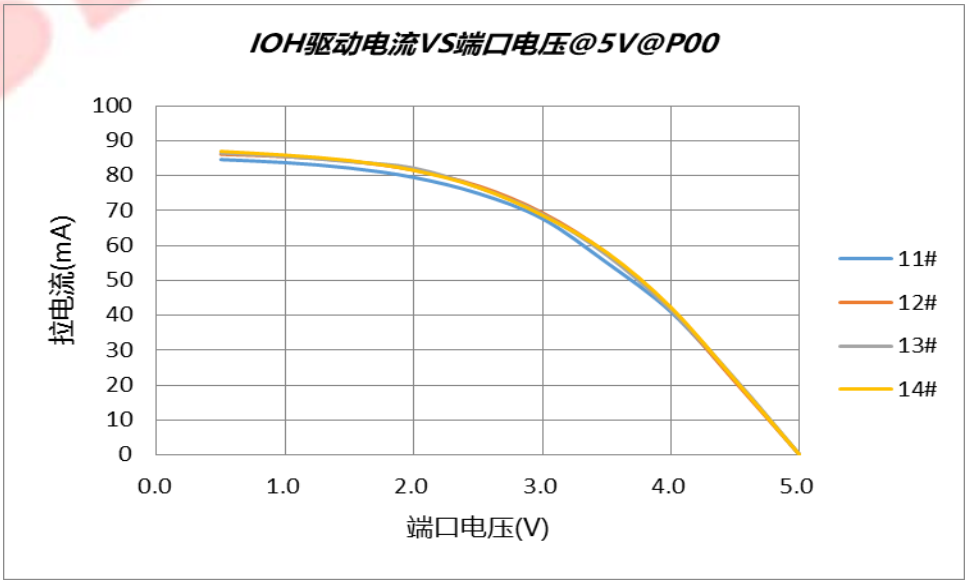


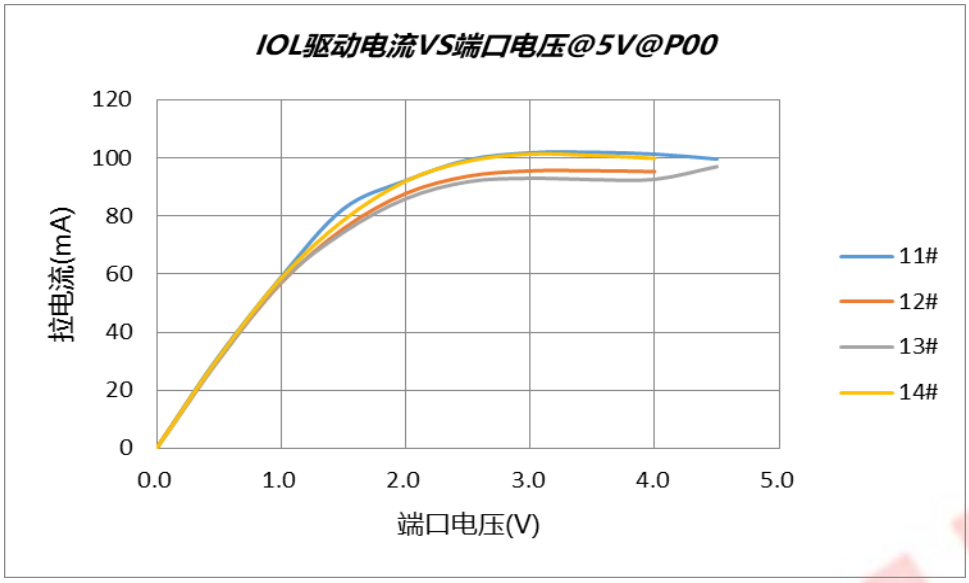
上/下拉电阻值 VS 电源电压

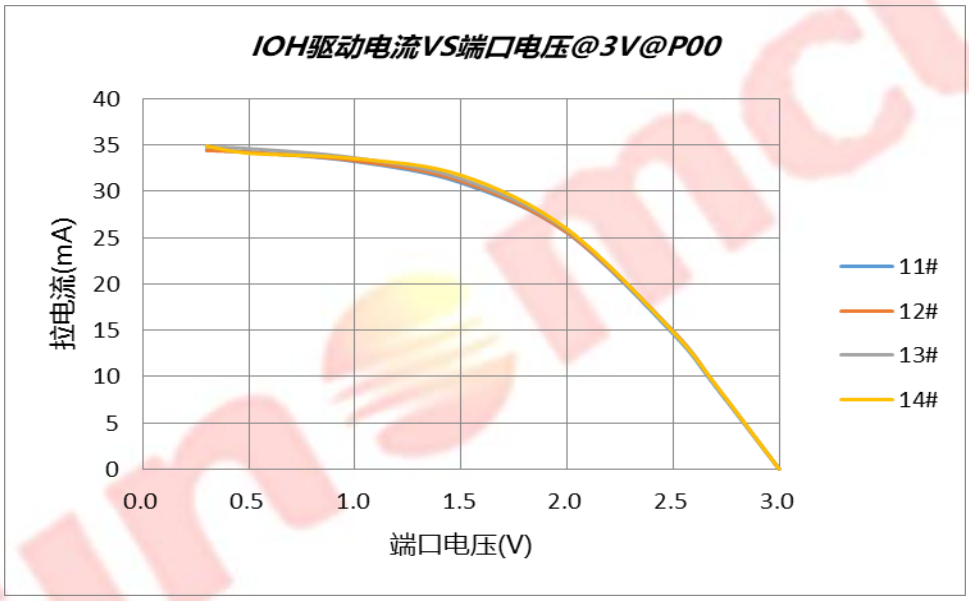


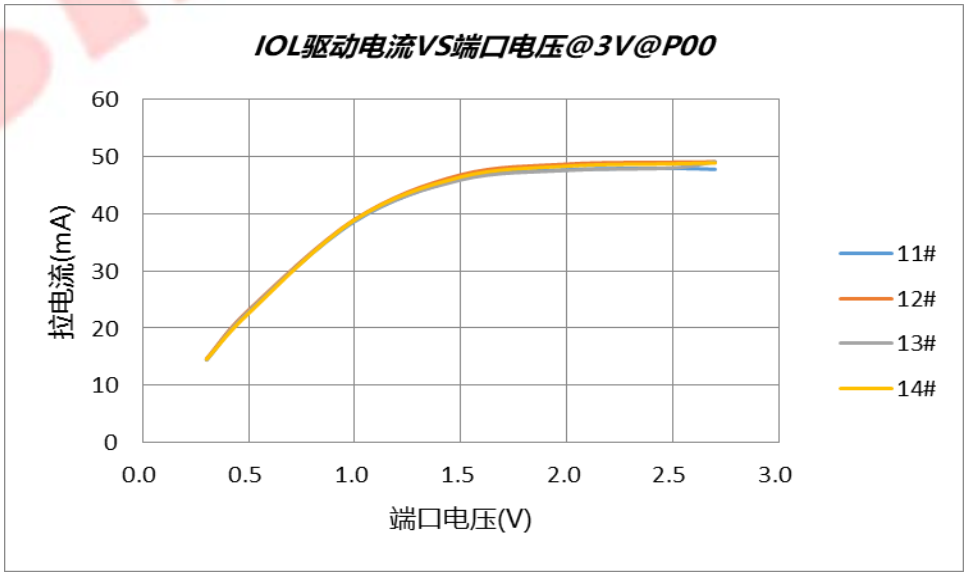


I/O 输出电流 VS 端口电压（VDD=5V）



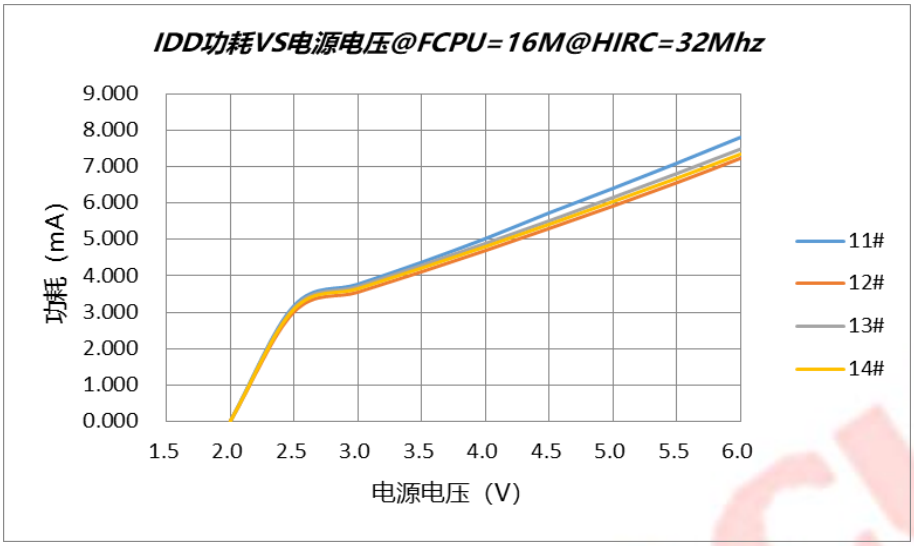

IOL驱动电流VS端口电压@5V@P00
拉电流(mA)
120
100
80
60
40
20
0
0.0
1.0
2.0
3.0
4.0
5.0
端口电压(V)
11#
12#
13#
14#

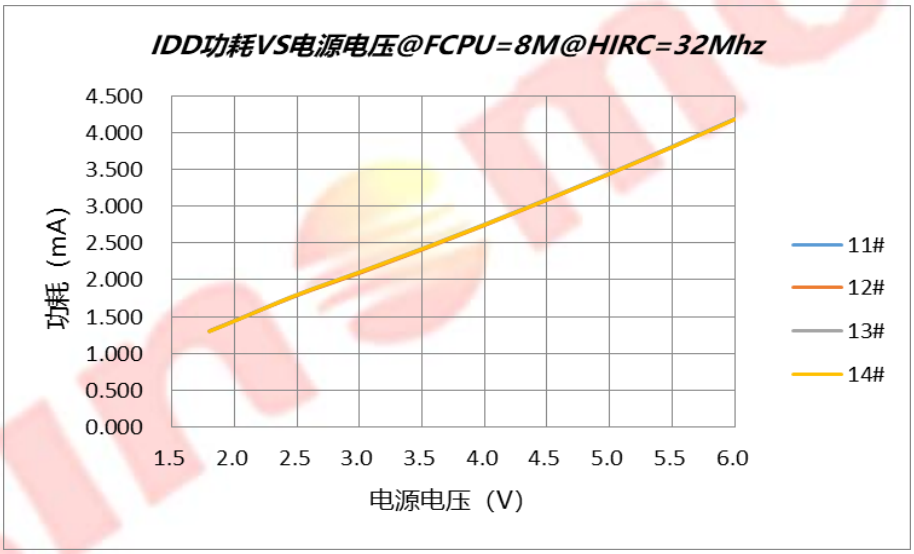

IOH驱动电流VS端口电压@3V@P00
拉电流(mA)
40
35
30
25
20
15
10
5
0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
端口电压(V)
11#
12#
13#
14#

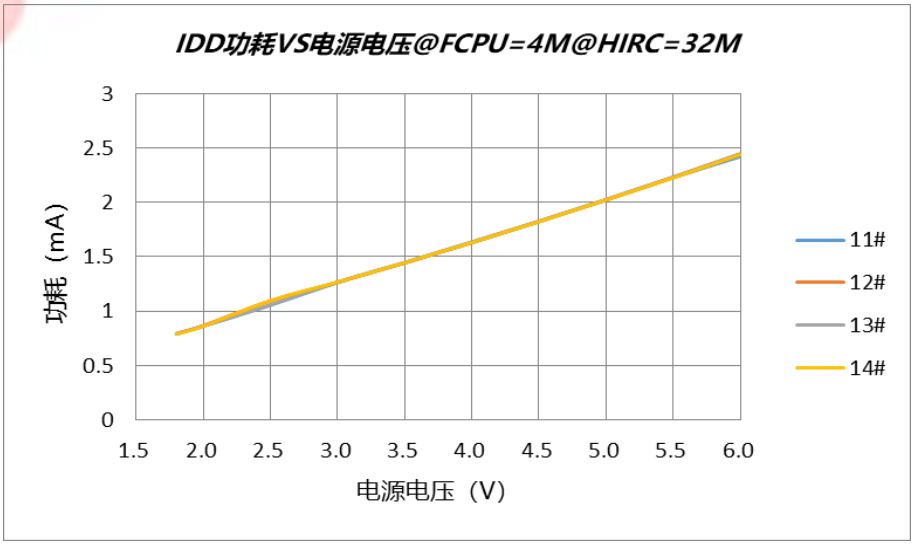

IOL驱动电流VS端口电压@3V@P00
拉电流(mA)
60
50
40
30
20
10
0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
端口电压(V)
11#
12#
13#
14#

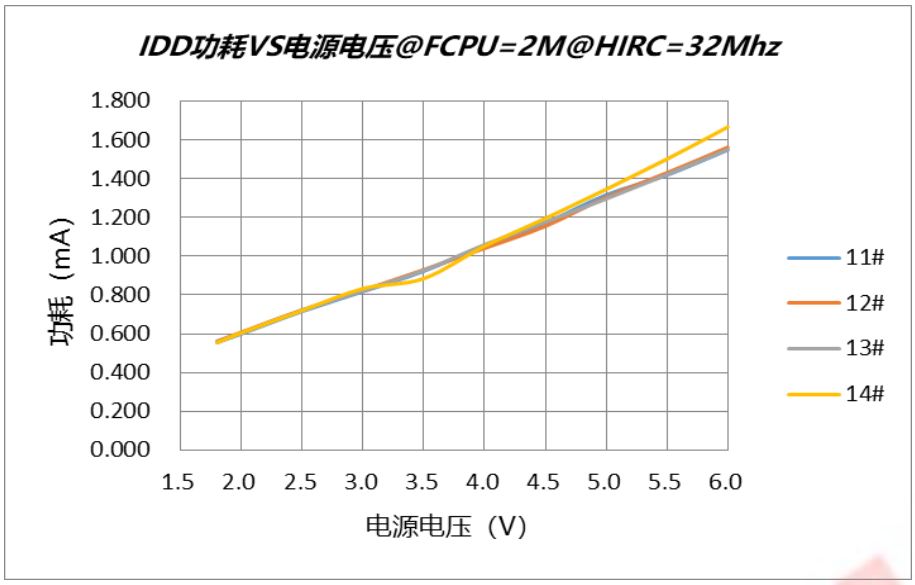
15.2 功耗特性

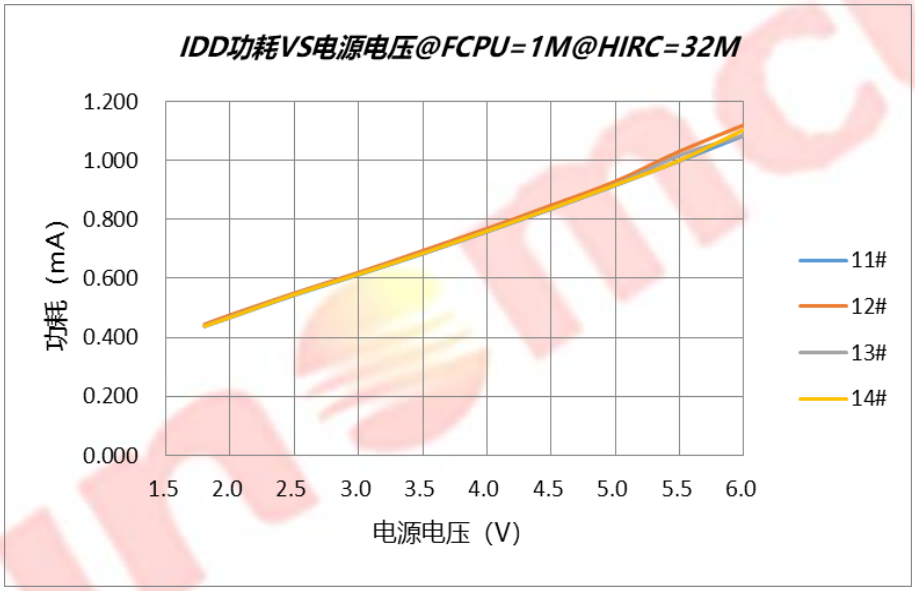
运行模式 功耗 VS 电源电压

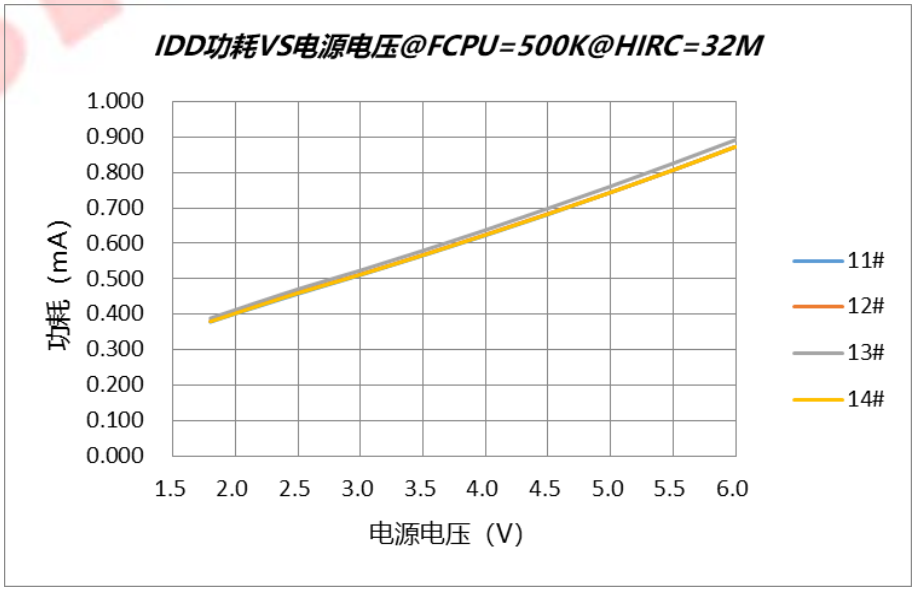


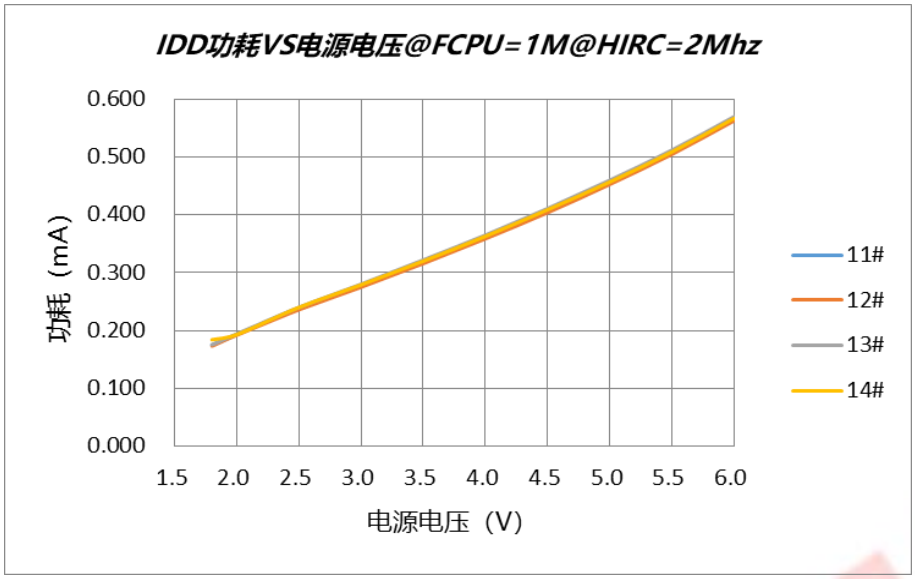


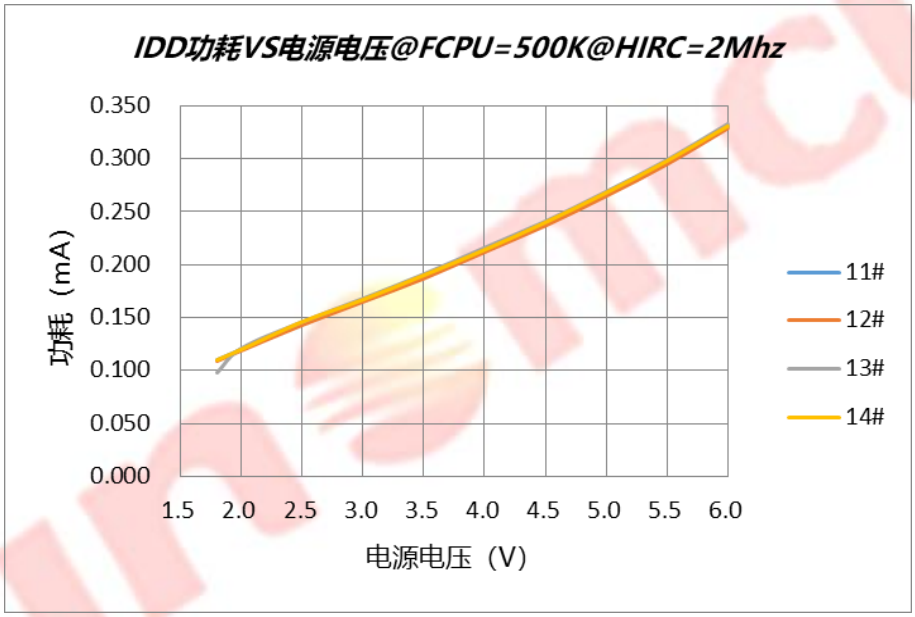


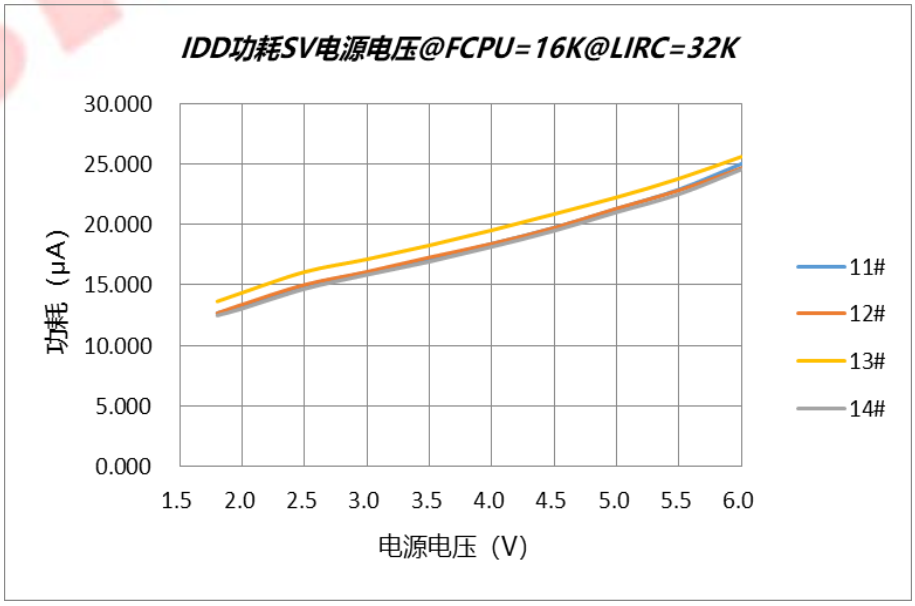

IDD功耗VS电源电压@FCPU=2M@HIRC=32Mhz
功耗（mA）
1.800
1.600
1.400
1.200
1.000
0.800
0.600
0.400
0.200
0.000
1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0
电源电压（V）
11#
12#
13#
14#


IDD功耗VS电源电压@FCPU=1M@HIRC=32M
功耗（mA）
1.200
1.000
0.800
0.600
0.400
0.200
0.000
1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0
电源电压（V）
11#
12#
13#
14#

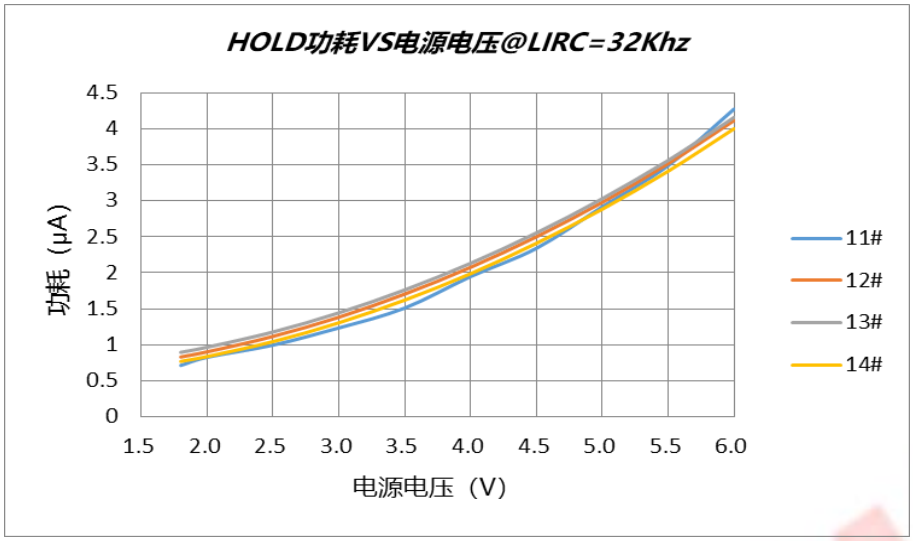

IDD功耗VS电源电压@FCPU=500K@HIRC=32M
功耗（mA）
1.000
0.900
0.800
0.700
0.600
0.500
0.400
0.300
0.200
0.100
0.000
1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0
电源电压（V）
11#
12#
13#
14#


IDD功耗VS电源电压@FCPU=1M@HIRC=2Mhz
功耗（mA）
0.600
0.500
0.400
0.300
0.200
0.100
0.000
1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0
电源电压（V）
11#
12#
13#
14#

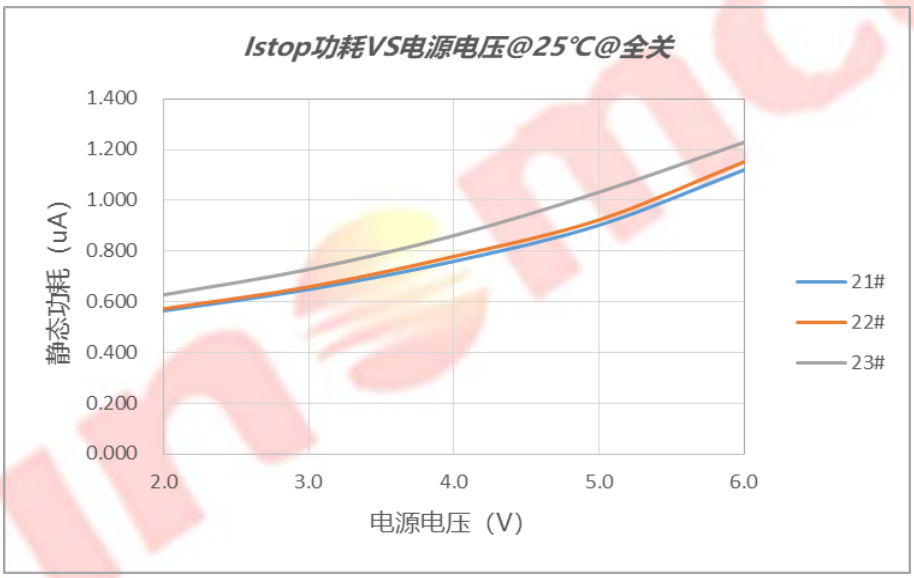

IDD功耗VS电源电压@FCPU=500K@HIRC=2Mhz
功耗（mA）
0.350
0.300
0.250
0.200
0.150
0.100
0.050
0.000
1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0
电源电压（V）
11#
12#
13#
14#

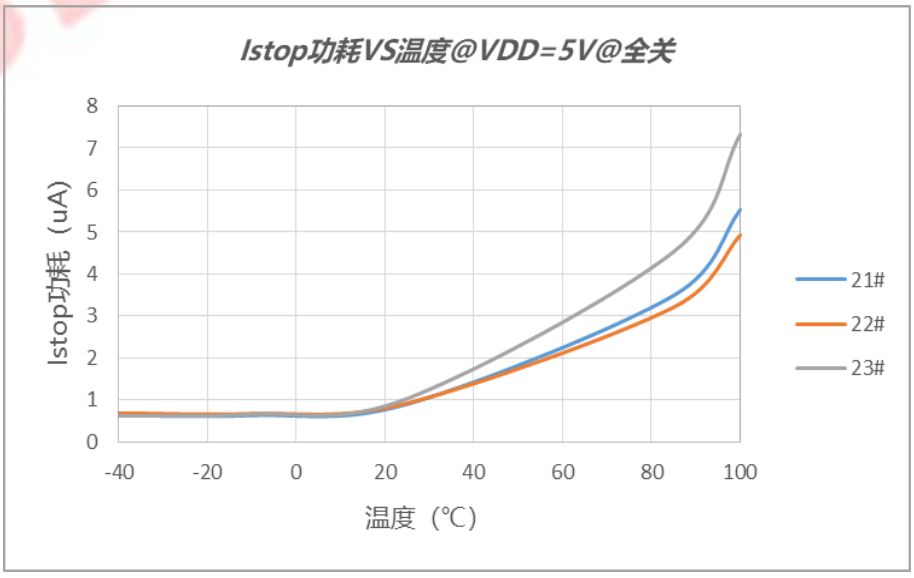

IDD功耗SV电源电压@FCPU=16K@LIRC=32K
功耗（μA）
30.000
25.000
20.000
15.000
10.000
5.000
0.000
1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0
电源电压（V）
11#
12#
13#
14#

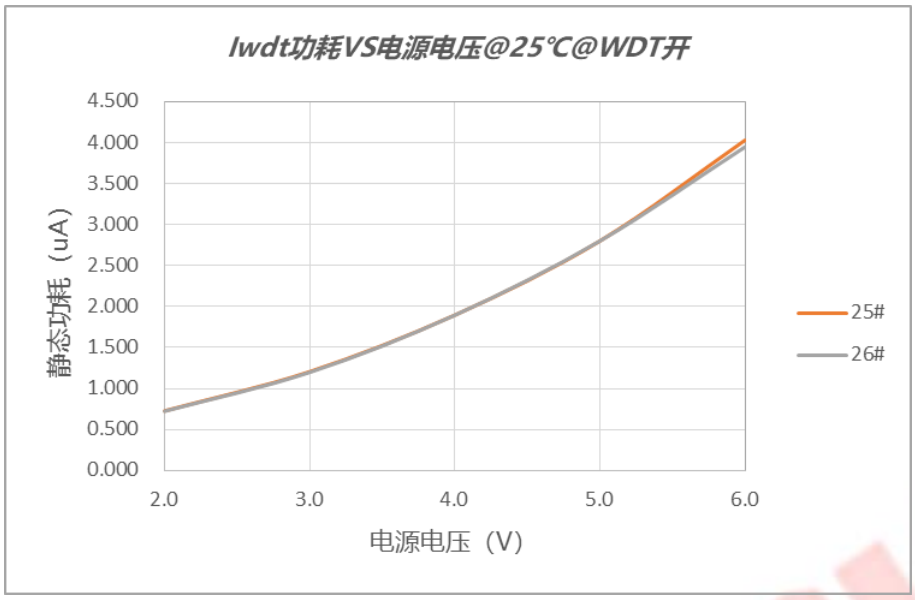
HOLD 模式 功耗 VS 电源电压

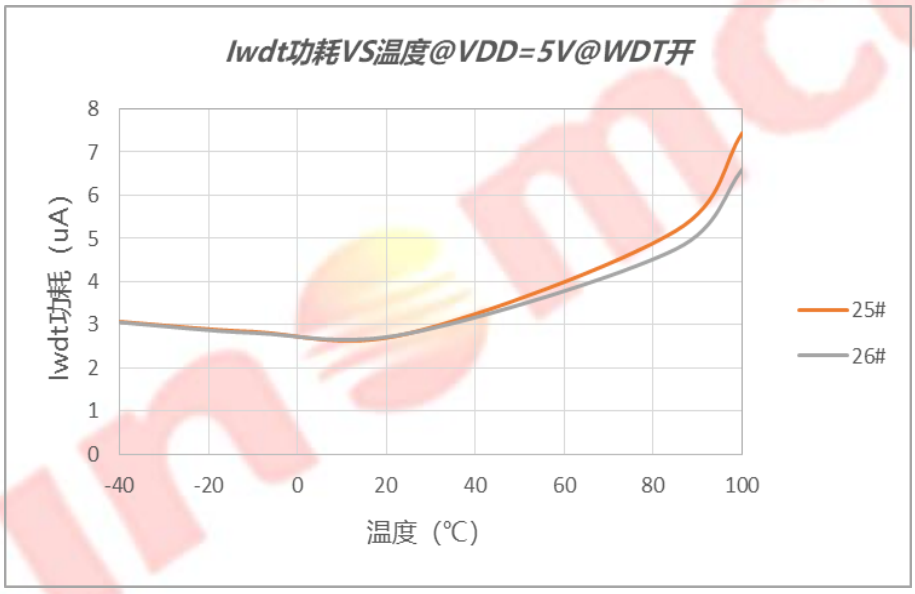


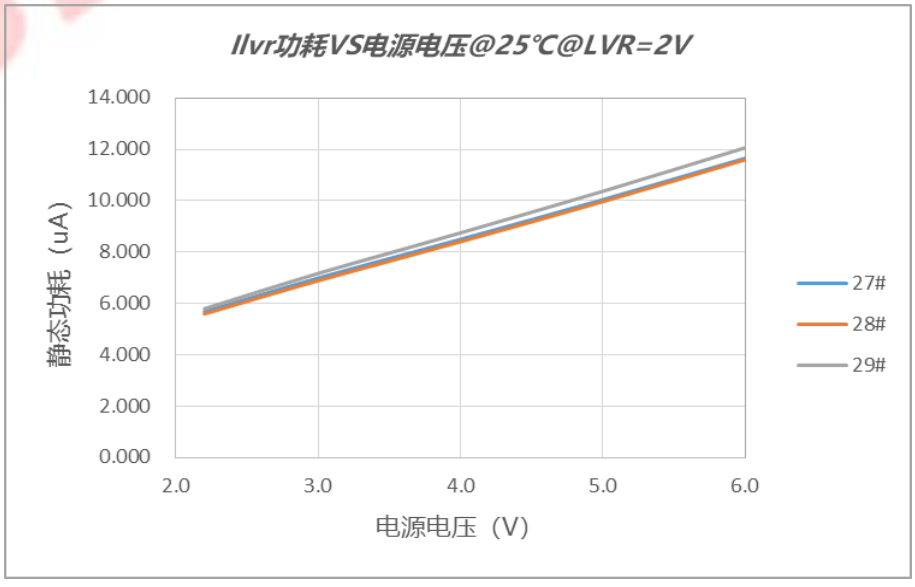
休眠模式 功耗 VS 电源电压

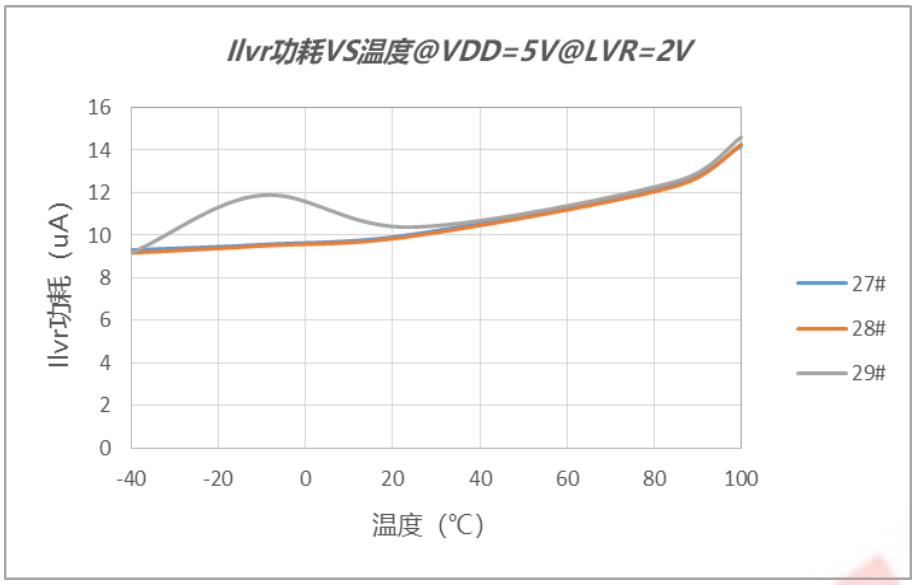





Iwdt功耗VS电源电压@25℃@WDT开
静态功耗（uA）
4.500
4.000
3.500
3.000
2.500
2.000
1.500
1.000
0.500
0.000
2.0
3.0
4.0
5.0
6.0
电源电压（V）
25#
26#

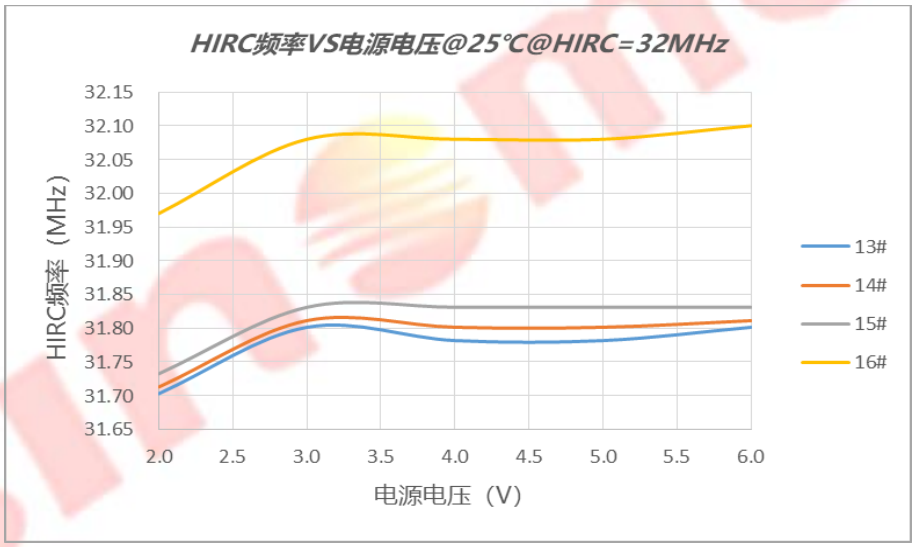

Iwdt功耗VS温度@VDD=5V@WDT开
Iwdt功耗（uA）
8
7
6
5
4
3
2
1
0
-40
-20
0
20
40
60
80
100
温度（℃）
25#
26#

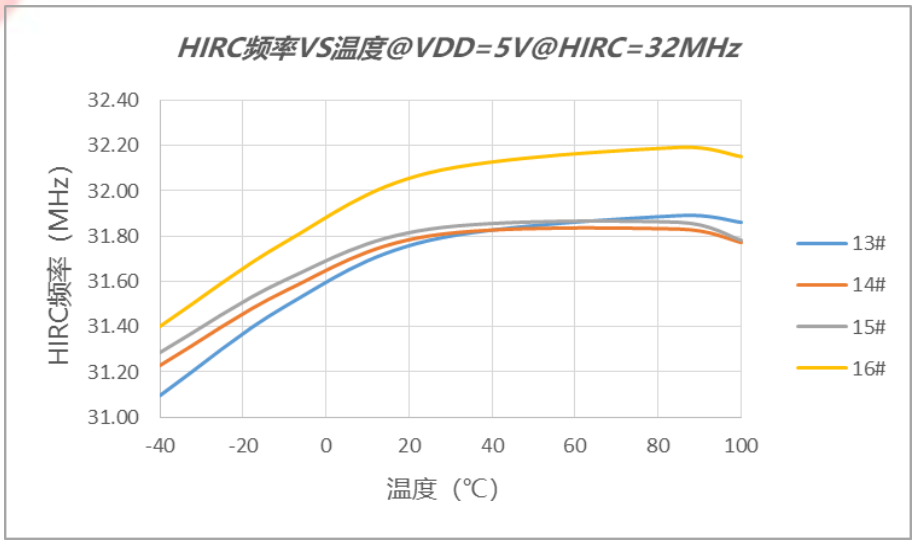

Ilvr功耗VS电源电压@25℃@LVR=2V
静态功耗（uA）
14.000
12.000
10.000
8.000
6.000
4.000
2.000
0.000
2.0
3.0
4.0
5.0
6.0
电源电压（V）
27#
28#
29#

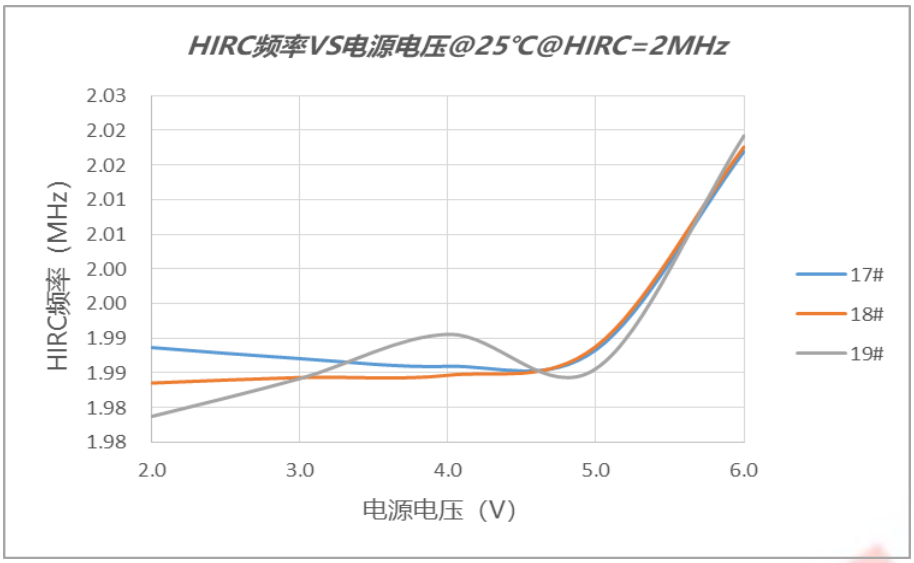


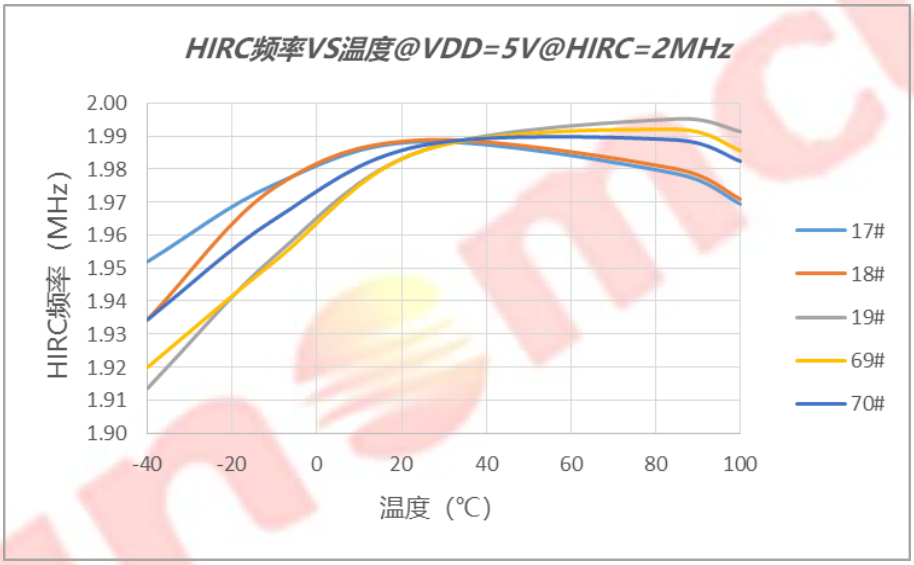
15.3 模拟电路特性

HIRC 频率 VS 电源电压/温度

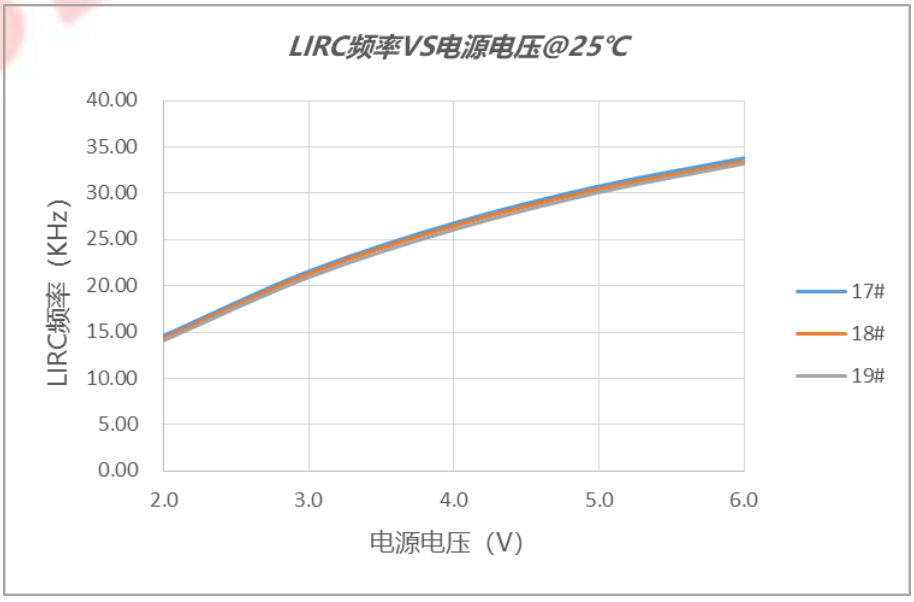


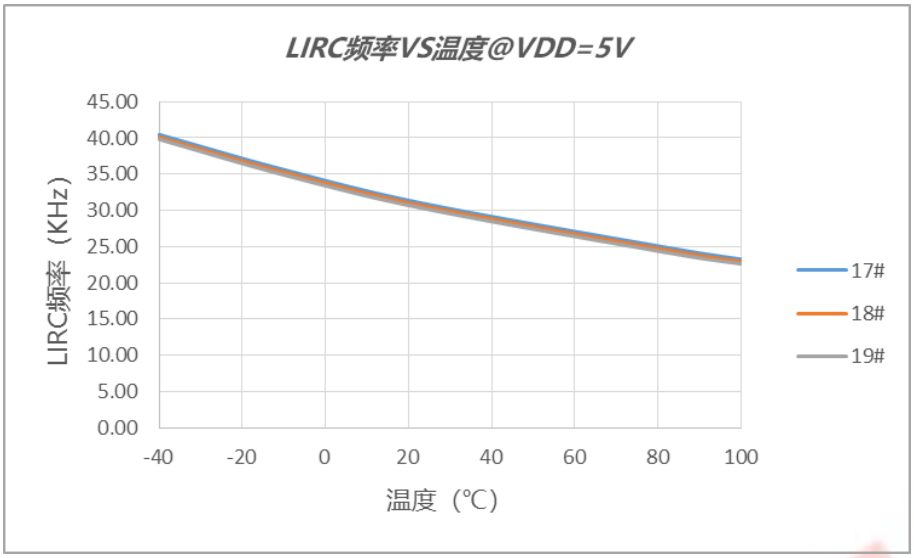




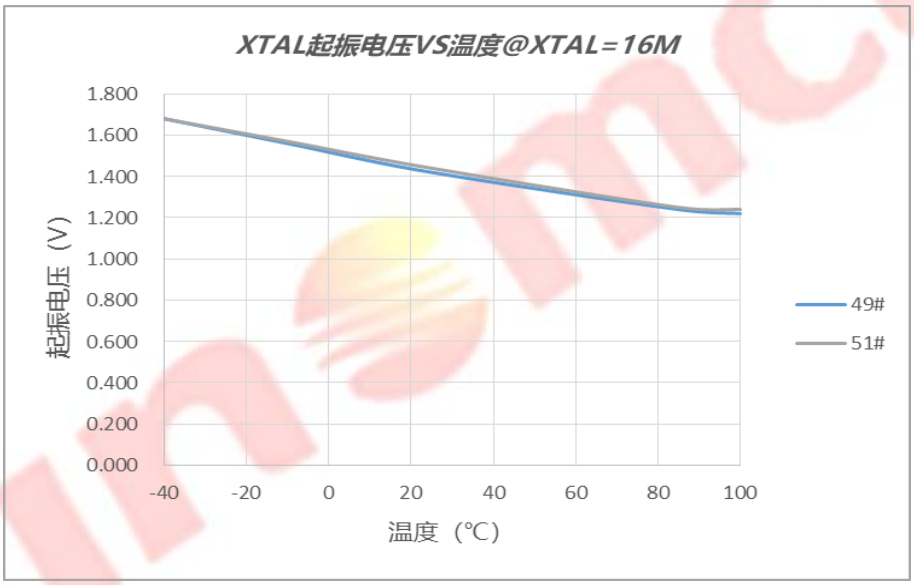


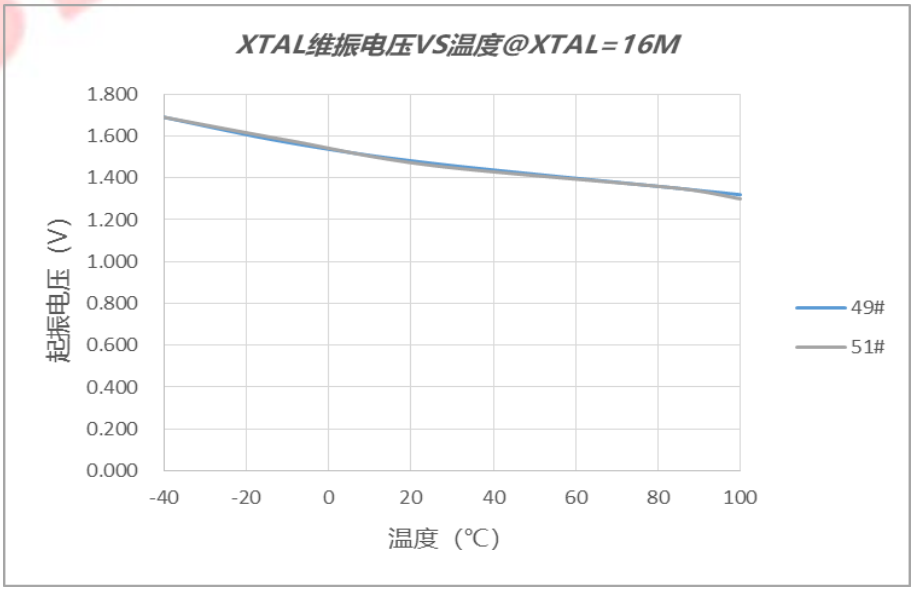
LIRC 频率 VS 电源电压/温度

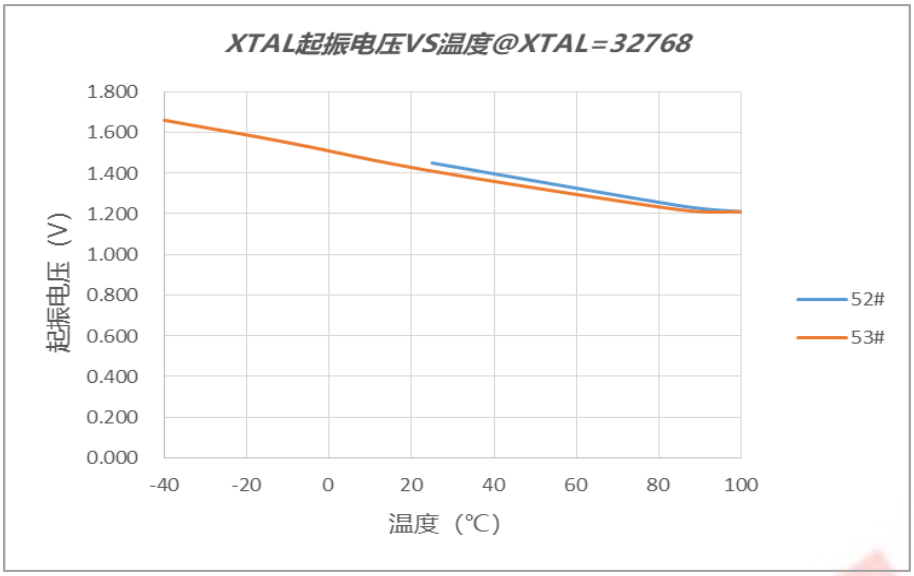


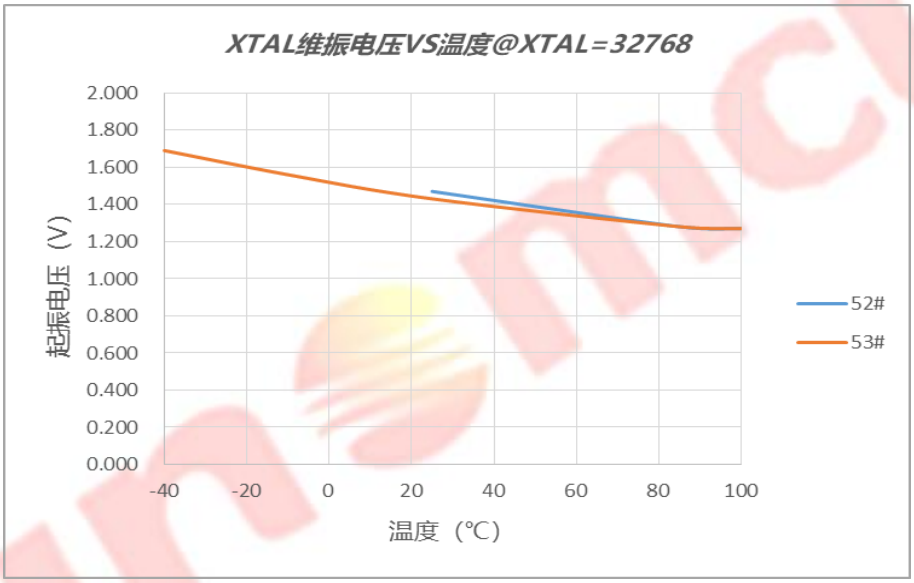


XTAL 起振/维振电压 VS 温度

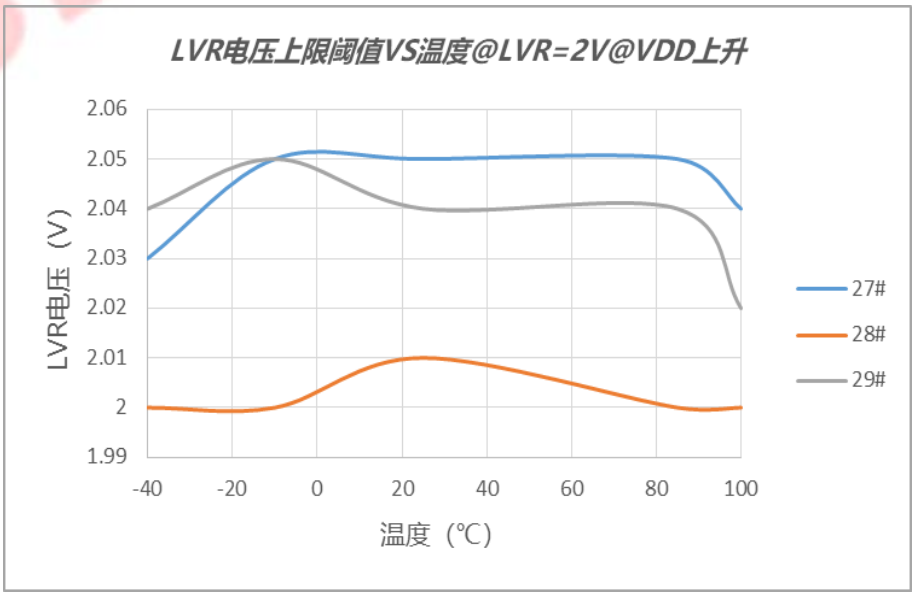


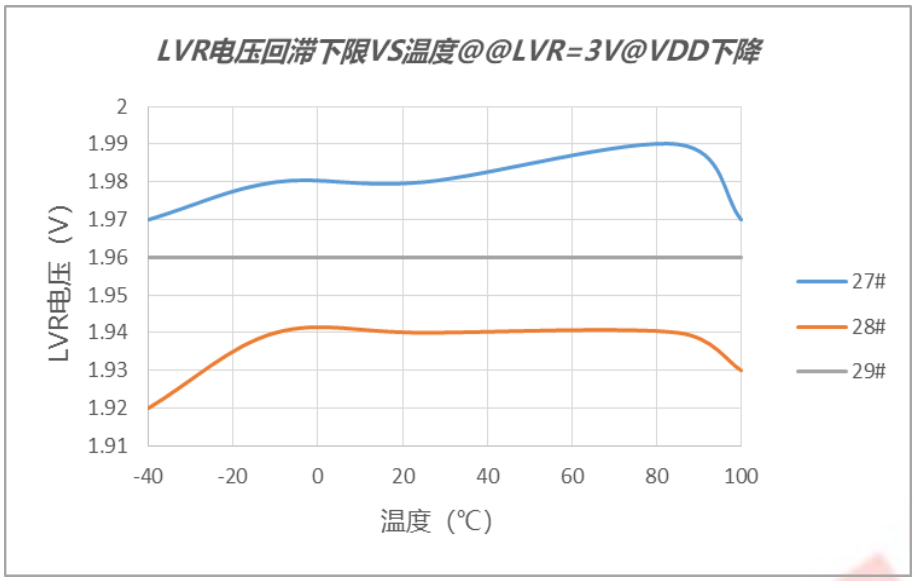




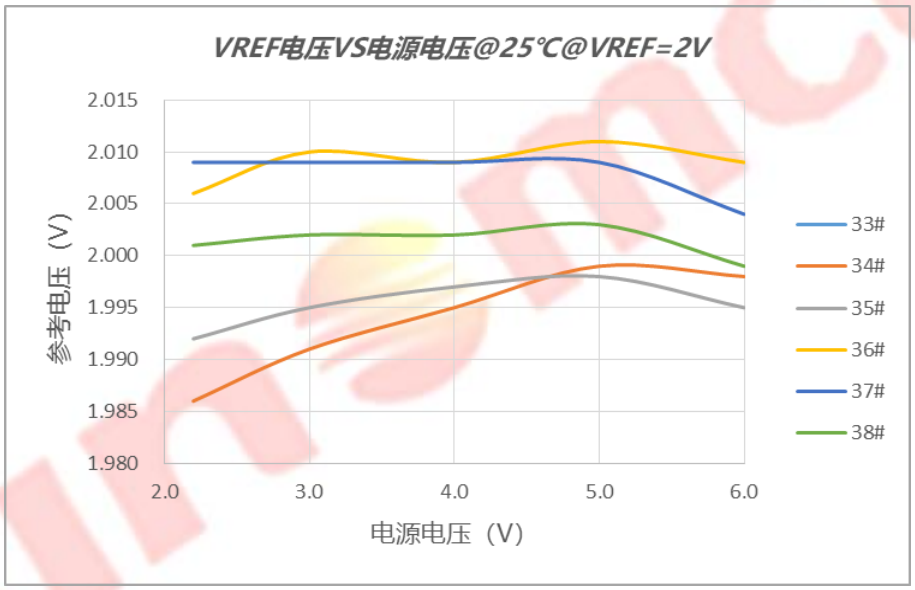


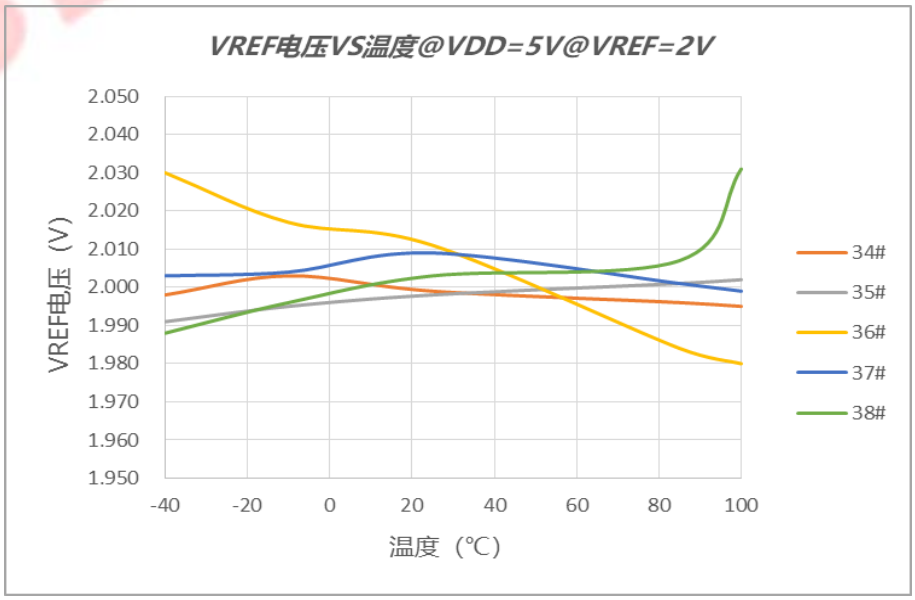
LVR 阈值电压 VS 温度

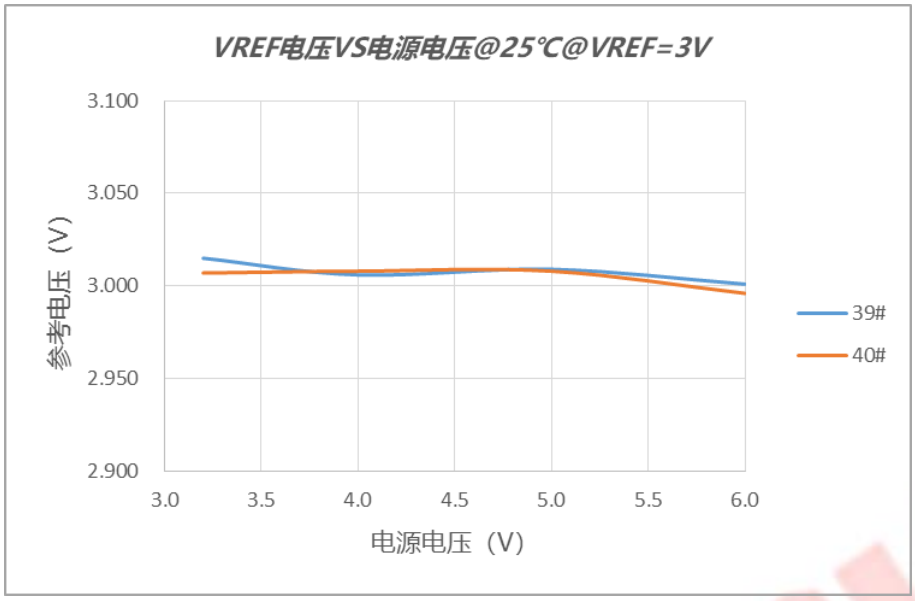


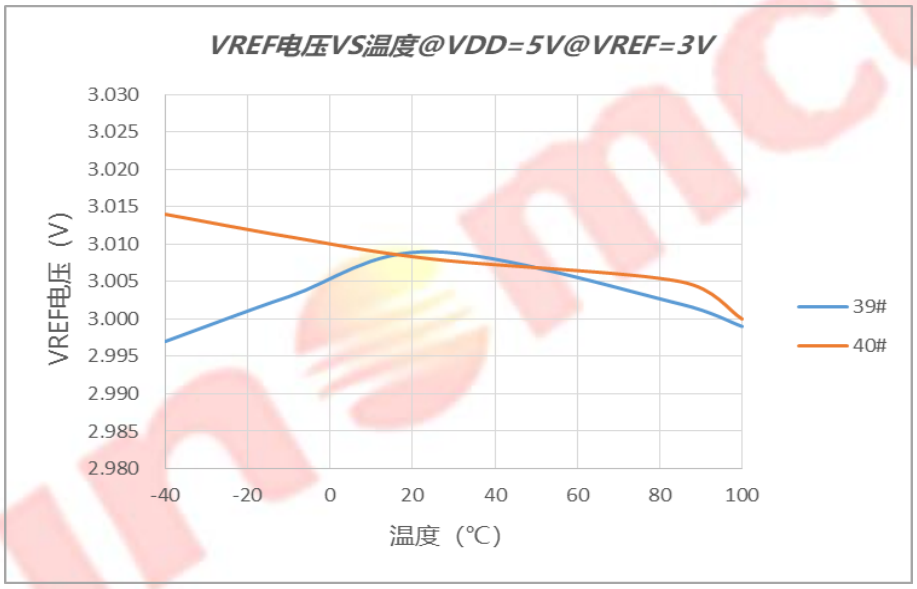


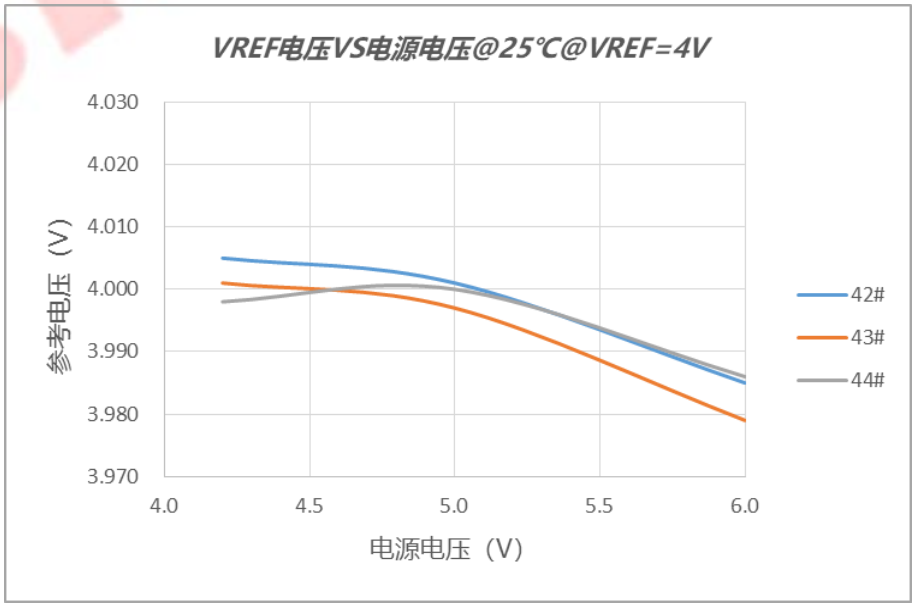
VIR 电压 VS 电源电压/温度

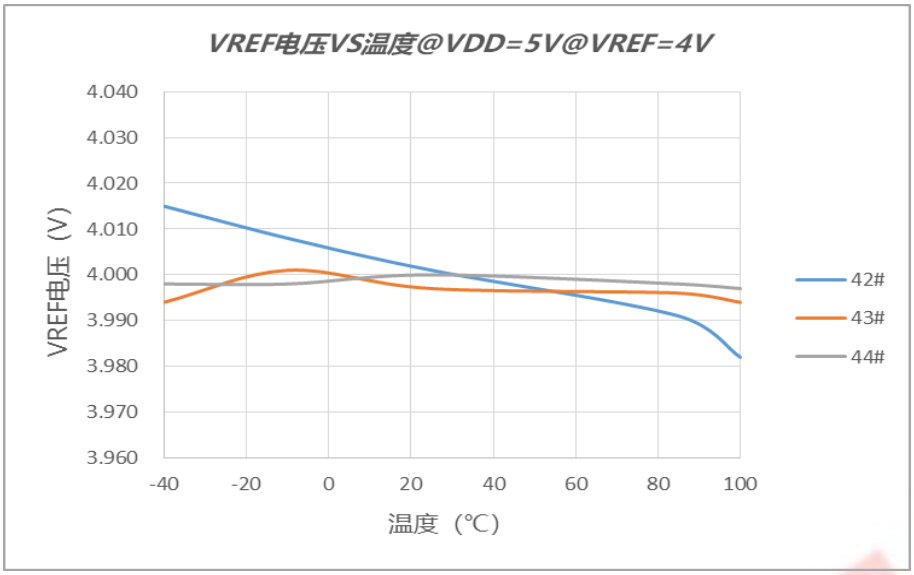




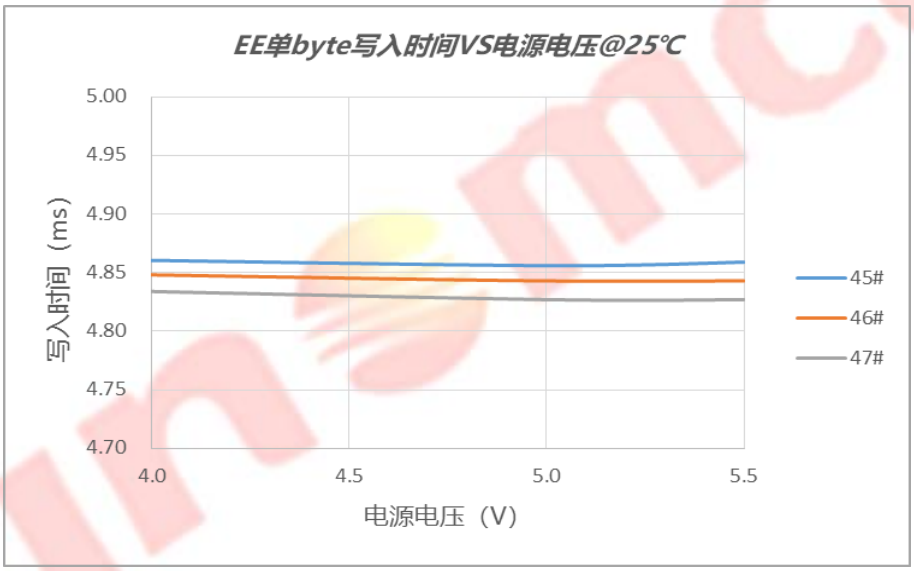

VREF电压VS电源电压@25℃@VREF=3V
参考电压（V）
3.100
3.050
3.000
2.950
2.900
3.0
3.5
4.0
4.5
5.0
5.5
6.0
电源电压（V）
39#
40#

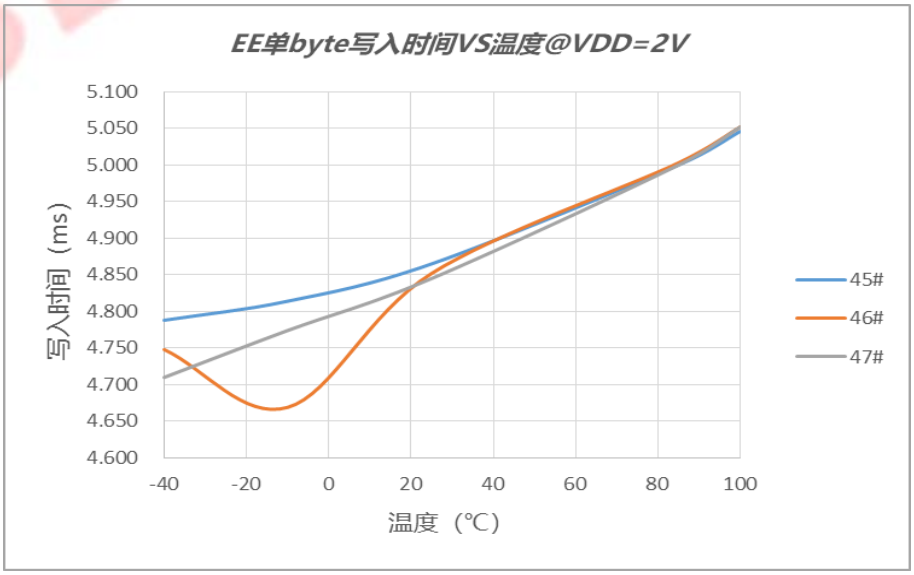

VREF电压VS温度@VDD=5V@VREF=3V
VREF电压（V）
3.030
3.025
3.020
3.015
3.010
3.005
3.000
2.995
2.990
2.985
2.980
-40
-20
0
20
40
60
80
100
温度（℃）
39#
40#


VREF电压VS电源电压@25℃@VREF=4V
参考电压（V）
4.030
4.020
4.010
4.000
3.990
3.980
3.970
4.0
4.5
5.0
5.5
6.0
电源电压（V）
42#
43#
44#



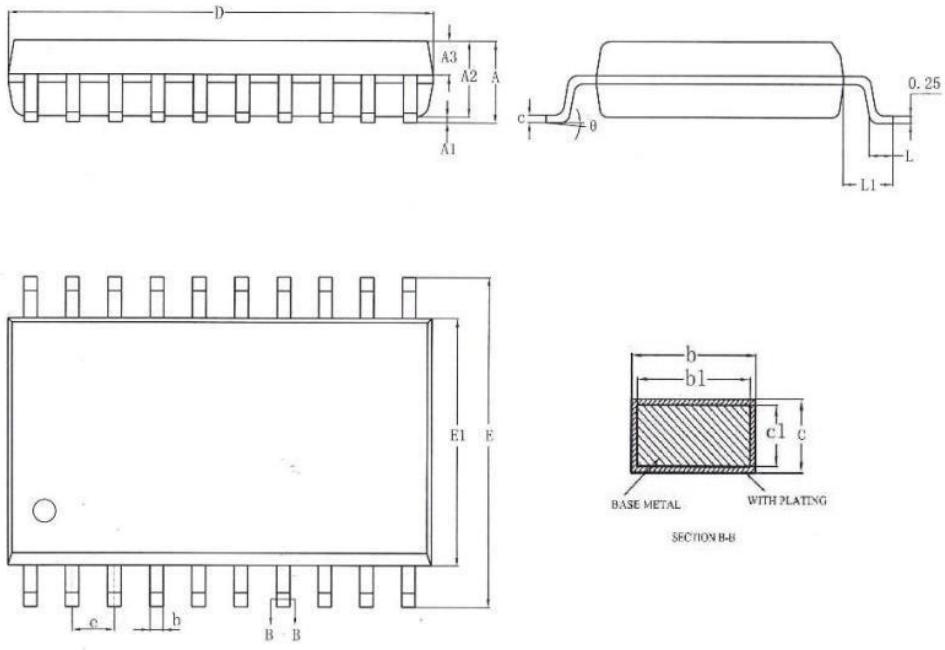
EEPROM 单字节写入时间 VS 电源电压/温度





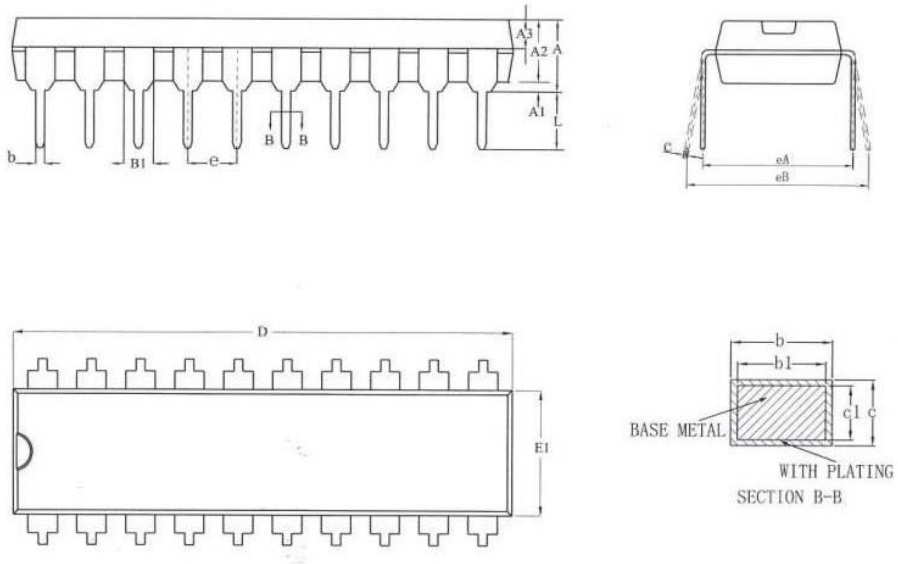
16 封装尺寸

16.1 SOP20



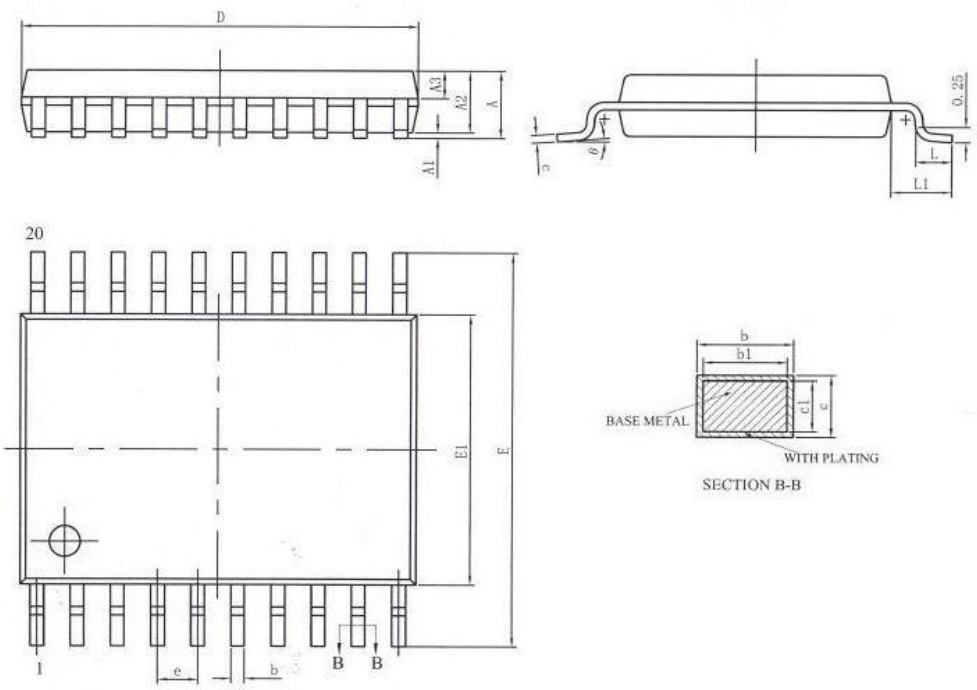
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	2.65
A1	0.10	—	0.30
A2	2.25	2.30	2.35
A3	0.97	1.02	1.07
b	0.35	—	0.43
b1	0.34	0.37	0.40
c	0.25	—	0.29
c1	0.24	0.25	0.26
D	12.70	12.80	12.90
E	10.10	10.30	10.50
E1	7.40	7.50	7.60
e	1.27BSC		
L	0.70	—	1.00
L1	1.40REF		
θ	0	—	8°

16.2 DIP20



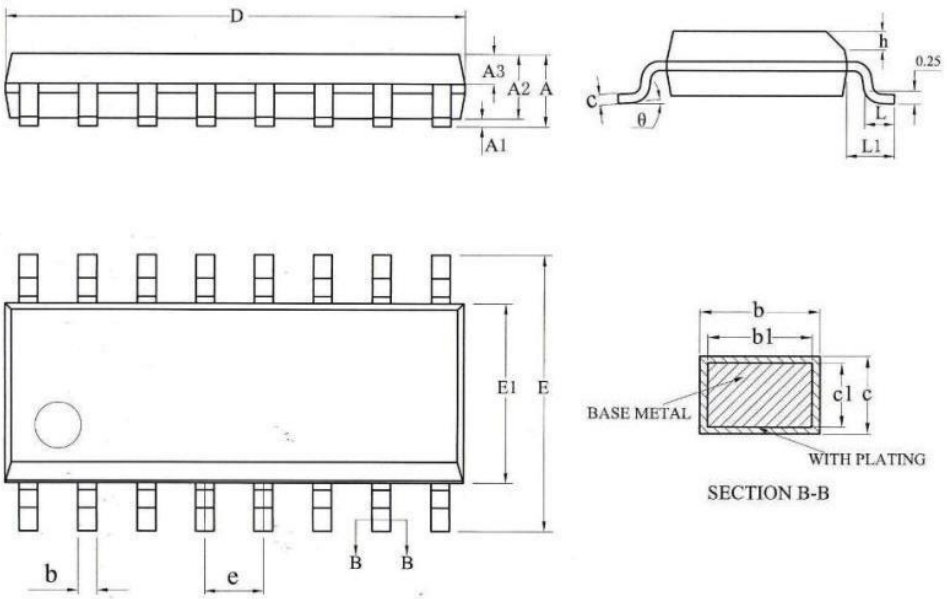
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	3.60	3.80	4.00
A1	0.51	—	—
A2	3.20	3.30	3.40
A3	1.47	1.52	1.57
b	0.44	—	0.52
b1	0.43	0.46	0.49
B1	1.52REF		
c	0.25	—	0.29
c1	0.24	0.25	0.26
D	25.80	25.90	26.00
E1	6.45	6.55	6.65
e	2.54BSC		
eA	7.62REF		
eB	7.62	—	9.30
eC	0	—	0.84
L	3.00	—	—

16.3 TSSOP20



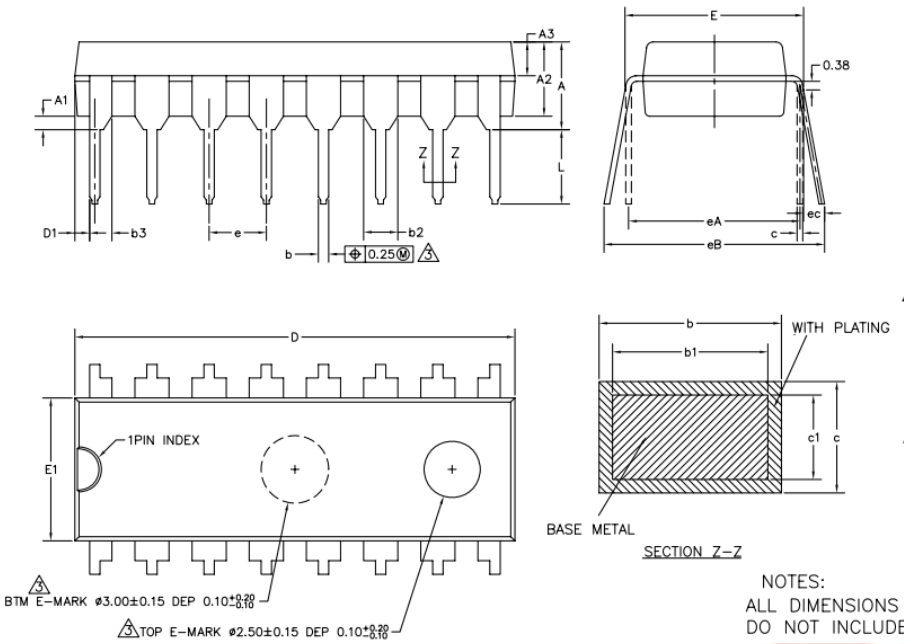
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.20
A1	0.05	—	0.15
A2	0.80	1.00	1.05
A3	0.39	0.44	0.49
b	0.20	—	0.29
b1	0.19	0.22	0.25
c	0.13	—	0.18
c1	0.12	0.13	0.14
D	6.40	6.50	6.60
E1	4.30	4.40	4.50
E	6.20	6.40	6.60
e	0.65BSC		
L	0.45	0.60	0.75
L1	1.00BSC		
θ	0	—	8°

16.4 SOP16



SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.10	—	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	—	0.47
b1	0.38	0.41	0.44
c	0.20	—	0.24
c1	0.19	0.20	0.21
D	9.80	9.90	10.00
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	1.27BSC		
h	0.25	—	0.50
L	0.50	—	0.80
L1	1.05REF		
θ	0	—	8°

16.5 DIP16

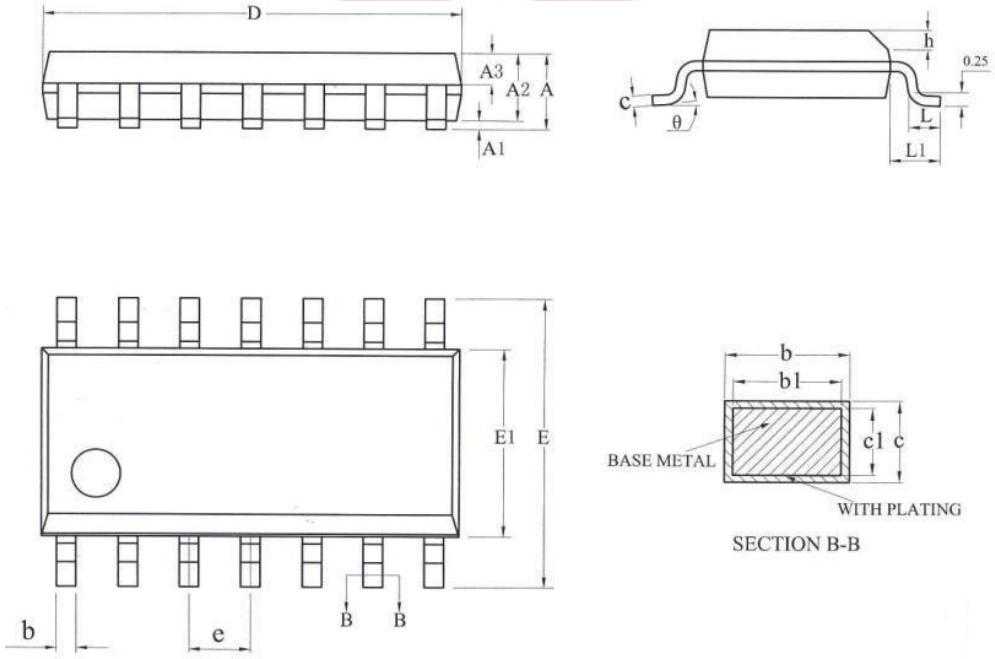


COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	–	–	4.80
A1	0.50	–	–
A2	3.05	3.25	3.45
A3	1.40	1.50	1.60
b	0.38	–	0.55
b1	0.38	0.46	0.51
b2	1.47	1.52	1.57
b3	0.89	0.99	1.09
c	0.21	–	0.35
2 c1	0.20	0.25	0.28
D	19.20	19.30	19.40
D1	0.13	–	–
E	7.62	7.87	8.25
E1	6.25	6.35	6.45
e	2.54BSC		
eA	7.62BSC		
3 eB	7.87	8.80	10.90
ec	0	–	1.52
L	2.92	3.30	3.81

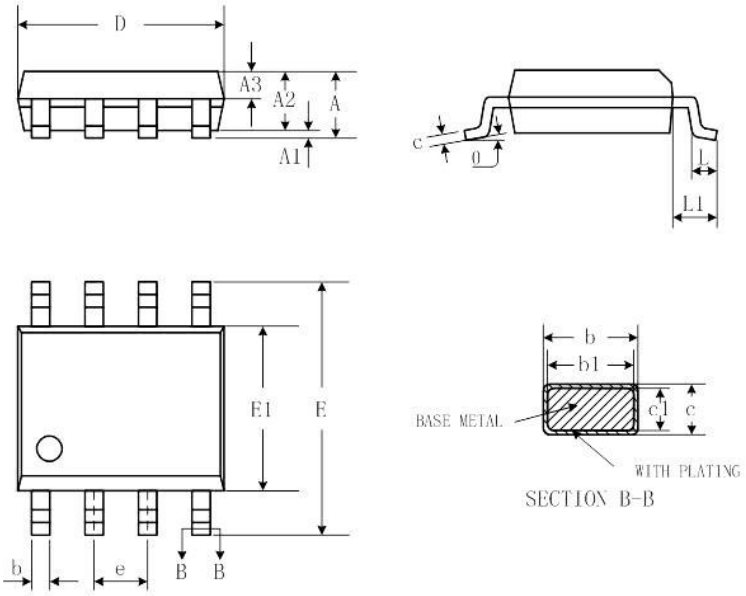
NOTES:
ALL DIMENSIONS MEET JEDEC STANDARD MS-001 BB
DO NOT INCLUDE MOLD FLASH OR PROTRUSIONS.

16.6 SOP14



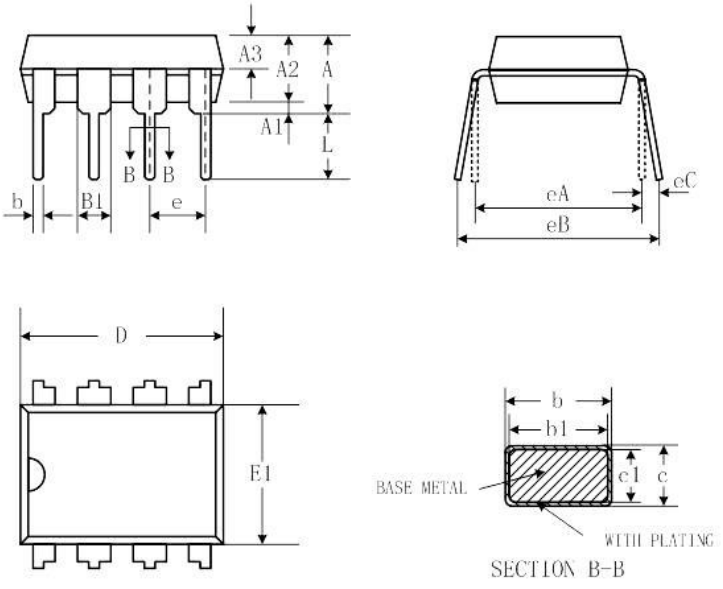
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	1.75
A1	0.05	—	0.225
A2	1.30	1.40	1.50
A3	0.60	0.65	0.70
b	0.39	—	0.47
b1	0.38	0.41	0.44
c	0.20	—	0.24
c1	0.19	0.20	0.21
D	8.55	8.65	8.75
E	5.80	6.00	6.20
E1	3.80	3.90	4.00
e	1.27BSC		
h	0.25	—	0.50
L	0.50	—	0.80
L1	1.05REF		
θ	0	—	8°

16.7 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°

16.8 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	-	-

17 修订记录

版本	日期	修订内容
V1.0	2021-04-08	发布初版；
V1.1	2021-04-13	修订定时器 T3 示意图，并增加寄存器位 PWM3ANV/FPWM3ANV 相关注释；
V1.2	2021-04-21	调整 ADC 输入电压特性参数，并增加 AN12,AN13 输入电压相关注释；
V1.3	2021-05-07	新增 DIP8 封装 A0A；