

SinoMCU 8 位单片机

MC35P7041

用户手册

V1.6



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.....	8
2.1	极限参数	8
2.2	直流电气特性	8
2.3	交流电气特性	9
2.4	ADC 特性参数	9
3	CPU 及存储器	11
3.1	指令集.....	11
3.2	程序存储器.....	13
3.3	数据存储器.....	14
3.4	堆栈	15
3.5	控制寄存器.....	15
3.6	用户配置字.....	18
4	系统时钟.....	20
4.1	内部高频 RC 振荡器	20
4.2	内部低频 RC 振荡器	21
4.3	外部晶体振荡器.....	21
4.4	外部 RC 振荡器	22
4.5	外部时钟输入	22
5	系统工作模式.....	23
5.1	高速模式	25
5.2	低速模式	25
5.3	休眠模式	25
5.4	空闲模式（绿色模式）	26
6	复位.....	27
6.1	复位条件	27
6.2	上电复位	28
6.3	外部复位	28
6.4	低电压复位.....	28
6.5	看门狗复位.....	28
7	I/O 端口	29
7.1	通用 I/O 功能.....	29
7.2	内部上拉电阻	30
7.3	端口模式控制	31
7.4	端口电平唤醒	32
8	定时器 TIMER.....	33
8.1	看门狗定时器 WDT.....	33

8.2	定时器 T0.....	33
8.3	定时器 T1.....	34
9	脉宽调制模块 PWM.....	38
9.1	PWM 概述.....	38
9.2	PWM 相关寄存器.....	38
10	蜂鸣器输出 BUZZER.....	41
10.1	BUZZER 概述.....	41
10.2	BUZZER 相关寄存器.....	41
11	模数转换器 ADC.....	42
11.1	ADC 概述.....	42
11.2	ADC 相关寄存器.....	43
11.3	ADC 操作步骤.....	46
11.4	ADC 零点偏移修调流程.....	46
12	中断.....	47
12.1	外部中断.....	47
12.2	定时器中断.....	47
12.3	ADC 中断.....	48
12.4	PWM 中断.....	48
12.5	中断相关寄存器.....	48
13	特性曲线.....	50
13.1	I/O 特性.....	50
13.2	功耗特性.....	54
13.3	模拟电路特性.....	60
14	封装尺寸.....	64
14.1	SOP20.....	64
14.2	DIP20.....	64
14.3	SOP16.....	65
14.4	DIP16.....	65
14.5	SOP8.....	66
14.6	DIP8.....	66
14.7	TSSOP20.....	67
14.8	SOP14.....	67
15	修订记录.....	68

1 产品概要

1.1 产品特性

- 8 位 CPU 内核
 - ◇ 精简指令集，8 级缓存寄存器型堆栈
 - ◇ 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 2/4/8/16/32/64/128/256 分频
 - ◇ 系统低频时钟下 F_{CPU} 固定为 4 分频
- 程序存储器
 - ◇ 2K×16 位 OTP 型程序存储器（烧录 1 次）
 - ◇ 1K×16 位 OTP 型程序存储器（烧录 2 次）
 - ◇ 可通过间接寻址读取程序存储器内容
- 数据存储器
 - ◇ 128 字节 SRAM 型通用数据存储器，支持直接寻址、间接寻址等多种寻址方式
- 3 组共 18 个 I/O
 - ◇ P0 (P00~P07), P4 (P40~P44), P5 (P50~P54)
 - ◇ P03 为输入/开漏输出口，可复用为外部复位 RST 输入，编程时为高压 VPP 输入
 - ◇ P01/P02 可复用为外部时钟振荡器输入/输出
 - ◇ 所有端口均内置上拉电阻，均可单独使能
 - ◇ P0 和 P5 所有端口均支持输入电平变化唤醒功能，并可单独使能
 - ◇ 端口输出灌电流 2 级可配置
- 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
 - ◇ 支持外接晶体振荡器 (32768Hz/455KHz/1MHz~16MHz)，可用作系统高频时钟源
 - ◇ 支持外接 RC 振荡器 (0~8MHz)，可用作系统高频时钟源
 - ◇ 支持外部时钟输入 (0~4MHz)，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
- 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
 - ◇ 空闲模式：CPU 停止运行，高频时钟源可选停止或工作，低频时钟源工作
 - ◇ 休眠模式：CPU 停止运行，所有时钟源停止工作
- 内部自振式看门狗计数器 (WDT)
 - ◇ 溢出时间：8192 / 内部低频 RC 振荡器频率 (FLIRC)，约为 256ms@VDD=5V
 - ◇ 工作模式可配置：始终开启、始终关闭、低功耗模式下关闭
- 2 个定时器
 - ◇ 8 位定时器 T0，支持空闲模式下溢出唤醒
 - ◇ 8 位定时器 T1，可实现外部计数、BUZ 和 PWM 功能
- 1 个 8 位脉宽调制模块 PWM
 - ◇ 支持共周期 3 路 PWM 输出，共用 1 个计数器和 1 个周期寄存器
 - ◇ 每路 PWM 输出均可单独使能/禁止，并可分别设置占空比

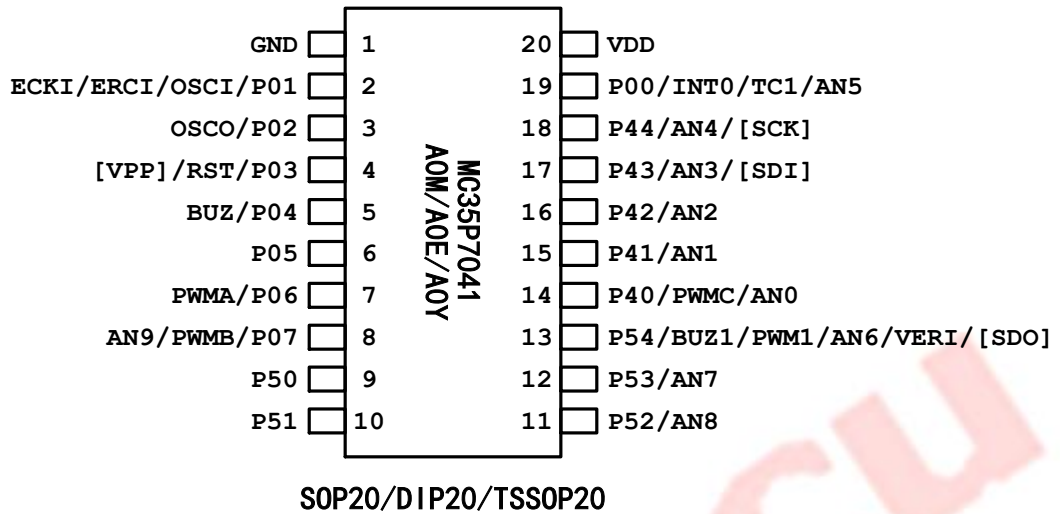
- ◇ 共用 1 个 PWM 溢出中断
- 1 路蜂鸣器输出
 - ◇ 占空比固定为 50%
 - ◇ 输出频率可设
- 1 个 12 位高精度逐次逼近型 ADC
 - ◇ 10 路外部通道：AN0~AN9；2 路内部通道：GND、VDD/4
 - ◇ 参考电压可选：VDD、内部参考电压 V_{IR} （2V/3V/4V）、外部参考电压 V_{ER} （VERI 输入）
 - ◇ ADC 时钟：F_{cpu} 的 4/8/32/64 分频
 - ◇ 支持零点校准
- 中断
 - ◇ 外部中断（INT0）
 - ◇ 定时器中断（T0~T1），PWM 中断
 - ◇ ADC 中断
- 低电压复位 LVR：2.2V/2.7V/3.2V/3.6V
- 低电压检测 LVD：2.4V/3.6V
- 工作电压
 - ◇ $V_{LVR32} \sim 5.5V @ F_{cpu} = 0\sim 8MHz$
 - ◇ $V_{LVR27} \sim 5.5V @ F_{cpu} = 0\sim 4MHz$
 - ◇ $V_{LVR22} \sim 5.5V @ F_{cpu} = 0\sim 2MHz$
 - ◇ $V_{LVR22} \sim 5.5V @ F_{cpu} = 0\sim 1MHz$
 - ◇ $V_{LVR22} \sim 5.5V @ F_{cpu} = 0\sim 455KHz/2$
 - ◇ $V_{LVR22} \sim 5.5V @ F_{cpu} = 0\sim 32KHz/4$
- 封装形式
 - ◇ SOP20/DIP20/SOP16/DIP16/SOP8/DIP8/TSSOP20/SOP14

1.2 订购信息

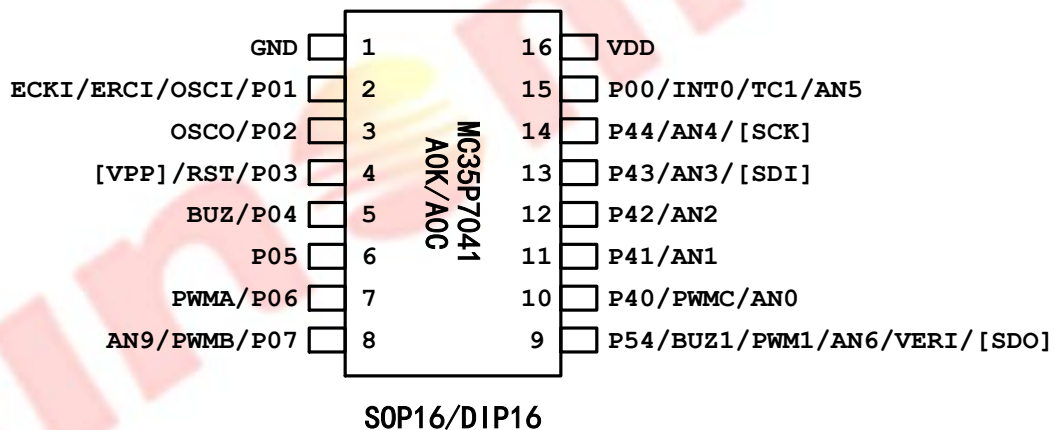
产品名称	封装形式	备注
MC35P7041A0M	SOP20	
MC35P7041A0E	DIP20	
MC35P7041A0K	SOP16	
MC35P7041A0C	DIP16	
MC35P7041A0H	SOP8	
MC35P7041A0A	DIP8	
MC35P7041A0Y	TSSOP20	
MC35P7041A0J	SOP14	
MC35P7041A1J	SOP14	

1.3 引脚排列

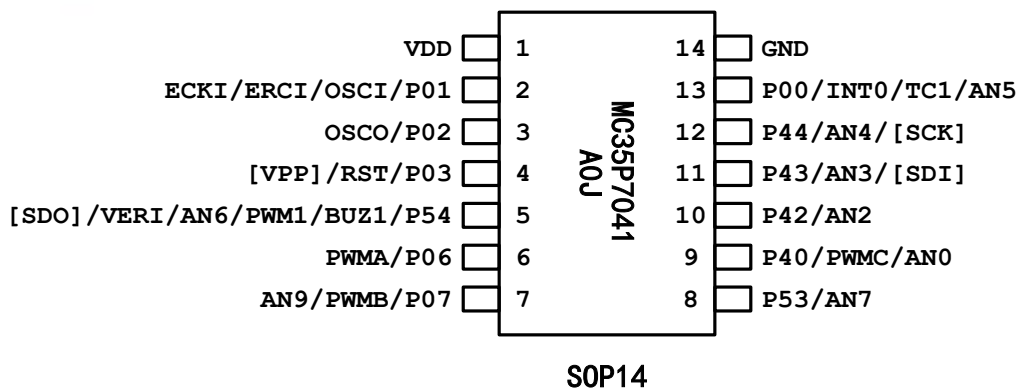
MC35P7041A0M/A0E/A0Y



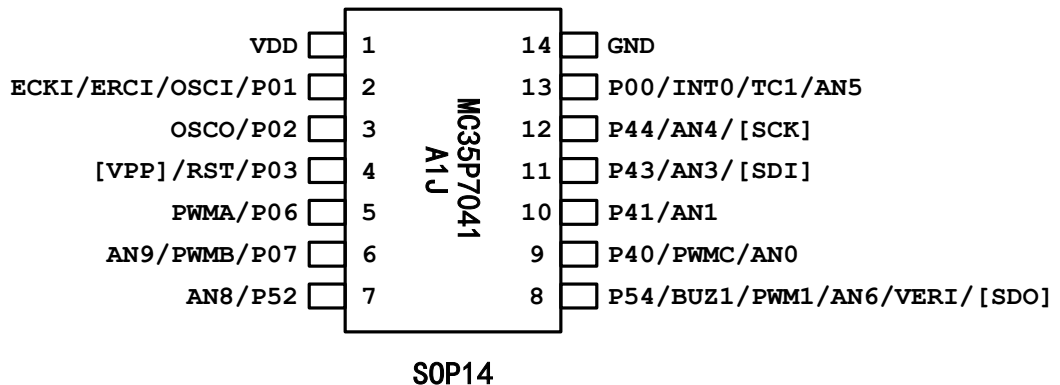
MC35P7041A0K/A0C



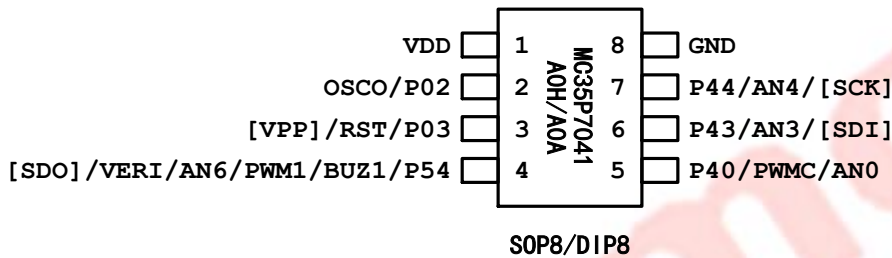
MC35P7041A0J



MC35P7041A1J



MC35P7041A0H/A0A



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 P03), P4, P5	D	GPIO (推挽输出), 内部上拉
P03	D	GPIO (开漏输出), 内部上拉
INT0	DI	外部中断输入
TC1	DI	定时器 T1 的外部计数输入
PWM1/BUZ1	DO	定时器 T1 的 PWM/BUZ 输出
PWMA~PWMC	DO	PWM 的 3 路输出
BUZ	DO	BUZZER 输出
AN0~AN9	AI	ADC 模拟输入通道
VERI	AI	ADC 外部参考电压输入
OSCI, OSCO	A	外部时钟振荡器输入/输出
ERCI	AI	外部 RC 振荡器输入
ECKI	DI	外部时钟信号输入
RST	DI	外部复位输入
SCK, SDI, SDO	D	编程时钟/数据输入/数据输出接口
VPP	P	编程高压输入

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

2 电气特性

2.1 极限参数

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

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

2.2 直流电气特性

VDD=5V, T=25°C

特性	符号	端口	条件	最小	典型	最大	单位
工作电压	VDD	VDD	Fcpu=8MHz@FHIRC/4	VLVR32		5.5	V
			Fcpu=4MHz@FHIRC/8	VLVR27		5.5	
			Fcpu=2MHz@FHIRC/16	VLVR22		5.5	
			Fcpu=1MHz@FHIRC/32	VLVR22		5.5	
			Fcpu=500KHz@FHIRC/64	VLVR22		5.5	
			Fcpu=8KHz@FLIRC/4	VLVR22		5.5	
输入漏电流	Ileak	所有输入脚	VDD=5V	-1		1	uA
输入高电平	Vih	所有输入脚		0.8VDD			V
输入低电平	Vil	所有输入脚				0.2VDD	V
输出拉电流	Ioh	推挽输出脚	VDD=5V, Voh=0.9VDD		16		mA
输出灌电流	Iol	所有输出脚 (除 P03)	VDD=5V, Vol=0.1VDD, DRVS 配置		16		mA
			VDD=5V, Vol=0.1VDD, DRVS 配置	13	25	50	mA
		P03	VDD=5V, Vol=0.1VDD		25		mA
上拉电阻	Rpu	所有输入脚	VDD=5V, Vin=0	40	85	160	KΩ
高速模式功耗	Ihigh	VDD	VDD=5V, Fcpu=1MHz@HIRC		1.3		mA
			VDD=3V, Fcpu=1MHz@HIRC		0.8		mA
低速模式功耗	Ilow	VDD	VDD=5V, Fcpu=FLIRC/4, HIRC 开		500		uA
			VDD=5V, Fcpu=FLIRC/4, HIRC 关		5	10	uA
			VDD=3V, Fcpu=FLIRC/4, HIRC 关		2	4	uA
空闲模式功耗	Ihold	VDD	VDD=5V, CPU 停, HIRC 关, LIRC 开		3	6	uA
			VDD=3V, CPU 停, HIRC 关, LIRC 开		1	3	uA

休眠模式功耗	Istop	VDD	VDD=5V, CPU 停, HIRC/LIRC 关		1	3	uA
低压检测电压	VLVD	VDD		-15%	2.4	+15%	V
				-15%	3.6	+15%	
低压复位电压	VLVR22	VDD		-15%	2.2	+15%	V
	VLVR27			-15%	2.7	+15%	
	VLVR32			-15%	3.2	+15%	
	VLVR36			-15%	3.6	+15%	
LVD/LVR 回滞电压		VDD			6%	12%	

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

2.3 交流电气特性

特性	符号	条件	最小	典型	最大	单位
HIRC 振荡频率	FHIRC	VDD=5V, T=25°C	-2%	32	+2%	MHz
		VDD=2.0V~5.5V, T=-40°C~85°C	-6%		+6%	
LIRC 振荡频率	FLIRC	VDD=5V, T=25°C	-50%	32	+50%	KHz
		VDD=3V, T=25°C	-50%	24	+50%	
16M 晶振起振电压		T=25°C	2.7			V
8M 晶振起振电压		T=25°C	2.4			V
4M 晶振起振电压		T=25°C	2.2			V
455K 晶振起振电压		T=25°C	2.2			V
32768 晶振起振电压		T=25°C	2.2			V
32768 晶振起振时间		VDD=5V, T=25°C		1		s
外部 RC 振荡频率	FERC1	VDD=5V, T=25°C, R=1KΩ, C=100pF		5.0		MHz
	FERC2	VDD=5V, T=25°C, R=3KΩ, C=100pF		2.0		MHz
	FERC3	VDD=5V, T=25°C, R=1KΩ, C=1000pF		650		KHz

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	VADC		2.7		5.5	V
积分非线性误差	INL	VREF=VDD, FADC=1MHz, Tcon=27us			±4	LSB
微分非线性误差	DNL	VREF=VDD, FADC=1MHz, Tcon=27us			±2	LSB
零点偏移误差	EZ	VREF=VDD, FADC=1MHz, Tcon=27us			±4	LSB
增益误差	ET	VREF=VDD, FADC=1MHz, Tcon=27us			±4	LSB

转换时钟	FADC	VDD=5V			1	MHz
转换时间	Tcon		20		27	1/FADC
ADC 输入电压	VAIN		GND		VREF	V
ADC 输入阻抗	RAIN		2			MΩ
ADC 输入电流	IAIN				2	uA
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	uA
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°C	-1.5%	2/3/4	+1.5%	
		选择内部参考电压 VIR, T=-40°C~85°C	-3%		+3%	
		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

3 CPU 及存储器

3.1 指令集

芯片的指令集为精简指令集。所有指令均为单字指令，即指令码只占用 1 个程序存储器地址空间。

指令汇总表

助记符	说明	操作	周期	标志
ADDAR R	ACC 和 R 相加, 结果存入 ACC	ACC+R→ACC	1	C,DC,Z
ADDRA R	ACC 和 R 相加, 结果存入 R	ACC+R→R	1+M	C,DC,Z
ADCAR R	ACC 和 R 相加 (带 C 标志), 结果存入 ACC	ACC+R+C→ACC	1	C,DC,Z
ADCRA R	ACC 和 R 相加 (带 C 标志), 结果存入 R	ACC+R+C→R	1+M	C,DC,Z
ASUBAR R	ACC 和 R 相减, 结果存入 ACC	ACC-R→ACC	1	C,DC,Z
ASUBRA R	ACC 和 R 相减, 结果存入 R	ACC-R→R	1+M	C,DC,Z
ASBCAR R	ACC 和 R 相减 (带 C 标志), 结果存入 ACC	ACC-R-/C→ACC	1	C,DC,Z
ASBCRA R	ACC 和 R 相减 (带 C 标志), 结果存入 R	ACC-R-/C→R	1+M	C,DC,Z
ANDAR R	ACC 和 R 与操作, 结果存入 ACC	ACC and R→ACC	1	Z
ANDRA R	ACC 和 R 与操作, 结果存入 R	ACC and R→R	1+M	Z
ORAR R	ACC 和 R 或操作, 结果存入 ACC	ACC or R→ACC	1	Z
ORRA R	ACC 和 R 或操作, 结果存入 R	ACC or R→R	1+M	Z
XORAR R	ACC 和 R 异或操作, 结果存入 ACC	ACC xor R→ACC	1	Z
XORRA R	ACC 和 R 异或操作, 结果存入 R	ACC xor R→R	1+M	Z
RLAR R	R 循环左移 (带 C 标志), 结果存入 ACC	R[7]→C R[6:0]→ACC[7:1] C→ACC[0]	1	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+M	C
RRAR R	R 循环右移 (带 C 标志), 结果存入 ACC	R[0]→C R[7:1]→ACC[6:0] C→ACC[7]	1	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+M	C
SWAPAR R	交换 R 的高低半字节, 结果存入 ACC	R[7:4]→ACC[3:0] R[3:0]→ACC[7:4]	1	-
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+M	-
MOVRA R	将 ACC 存入 R	ACC→R	1	-
MOVAR R	将 R 存入 ACC	R→ACC	1	Z
XCH R	ACC 和 R 交换	ACC↔R	1+M	-

CLRR	R	清零 R	$0 \rightarrow R$	1	Z
JZR	R	R 自加 1; 结果为 0 则跳过下一条指令	$R+1 \rightarrow R$; 结果为 0 则 $PC+2 \rightarrow PC$	$1+J+M$	-
JZAR	R	R 加 1, 结果存入 ACC; 结果为 0 则跳过下一条指令	$R+1 \rightarrow ACC$; 结果为 0 则 $PC+2 \rightarrow PC$	$1+J$	-
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 $PC+2 \rightarrow PC$	$1+J+M$	-
DJZAR	R	R 减 1, 结果存入 ACC; 结果为 0 则跳过下一条指令	$R-1 \rightarrow ACC$; 结果为 0 则 $PC+2 \rightarrow PC$	$1+J$	-
BCLR	R, b	将 R 的第 b 位清 0	$0 \rightarrow R[b]$	$1+M$	-
BSET	R, b	将 R 的第 b 位置 1	$1 \rightarrow R[b]$	$1+M$	-
JBCLR	R, b	若 R 的第 b 位为 0, 则跳过下一条指令	若 $R[b]=0$, 则 $PC+2 \rightarrow PC$	$1+J$	-
JBSET	R, b	若 R 的第 b 位为 1, 则跳过下一条指令	若 $R[b]=1$, 则 $PC+2 \rightarrow PC$	$1+J$	-
CMPR	R	ACC 和 R 比较; 相等则跳过下一条指令	$ACC-R$; 结果为 0 则 $PC+2 \rightarrow PC$	$1+J$	C,Z
CMPI	K	ACC 和 K 比较; 相等则跳过下一条指令	$ACC-K$; 结果为 0 则 $PC+2 \rightarrow PC$	$1+J$	C,Z
ADDAI	K	ACC 和 K 相加, 结果存入 ACC	$ACC+K \rightarrow ACC$	1	C,DC,Z
ASUBAI	K	ACC 和 K 相减, 结果存入 ACC	$ACC-K \rightarrow ACC$	1	C,DC,Z
ANDAI	K	ACC 和 K 与操作, 结果存入 ACC	$ACC \text{ and } K \rightarrow ACC$	1	Z
ORAI	K	ACC 和 K 或操作, 结果存入 ACC	$ACC \text{ or } K \rightarrow ACC$	1	Z
XORAI	K	ACC 和 K 异或操作, 结果存入 ACC	$ACC \text{ xor } K \rightarrow ACC$	1	Z
MOVAI	K	将 K 存入 ACC	$K \rightarrow ACC$	1	-
CALL	K	子程序调用	$PC+1 \rightarrow TOS$ $K \rightarrow PC[10:0]$	2	-
GOTO	K	无条件跳转	$K \rightarrow PC[10:0]$	2	-
RETURN		从子程序返回	$TOS \rightarrow PC$	2	-
RETIE		从中断返回	$TOS \rightarrow PC$ $1 \rightarrow GIE$	2	-
NOP		空操作	空操作	1	-
MULAR	R	ACC 和 R 相乘, 结果存入[HIBYTE:ACC]	$ACC \times R \rightarrow [HIBYTE:ACC]$	1	-
PUSH		缓存 ACC 和 C,DC,Z	$ACC \text{ 和 } C,DC,Z \rightarrow BUF$	1	-
POP		恢复 ACC 和 C,DC,Z	$BUF \rightarrow ACC \text{ 和 } C,DC,Z$	1	C,DC,Z
MOVC		读取程序存储器内容	$ROM(@[FSR1:FSR0]) \rightarrow [HIBYTE:ACC]$	2	-

注:

1. ACC-算术逻辑单元累加器, R-数据存储器, K-立即数;
2. 对于条件跳转类指令, 若跳转条件成立, 则 $J=1$, 否则 $J=0$;
3. 部分访问数据存储器的指令, 若目的寄存器为 GPR, 则 $M=1$, 若目的寄存器为 SFR, 则 $M=0$;
4. PUSH/POP 指令涉及的缓存器 BUF 仅有 1 层, 所以 PUSH/POP 必须成对使用, 否则将导致数据错误;

3.2 程序存储器

芯片的程序存储器为 OTP 型存储器，可通过配置字 PAGE 选择存储器的地址空间范围。

2K×16 位的地址空间范围为 0000H-07FFH，仅能烧录 1 次，地址分配如下图所示：

复位起始地址 (0000H)
通用程序区 (0001H - 0007H)
中断入口地址 (0008H)
通用程序区 (0009H - 07FFH)

1K×16 位的地址空间范围为 0000H~03FFH，可以烧录 2 次，地址分配如下图所示：

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

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

```

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

3.3 数据存储器

数据存储器包括通用数据存储器 GPR 和特殊功能寄存器 SFR，具体地址分配参照下表。GPR/SFR 可直接寻址或通过 INDF 间接寻址。

数据存储器区地址映射表

地址	类型	0/8	1/9	2/A	3/B	4/C	5/D	6/E	7/F	
00H-7FH	GPR	通用数据存储器区								
80H-87H	SFR			HIBYTE	FSR0	FSR1		PFLAG		
88H-8FH								ZEROTR	OSCCAL	
90H-97H										
98H-9FH										
A0H-A7H		WKCR0						WKCR5		
A8H-AFH								ADIOS0	ADIOS1	ADIOS2
B0H-B7H			ADCR0	ADCR1	ADRH	ADRL	ADCR2			
B8H-BFH		OEP0							EINTCR	
C0H-C7H						OEP4	OEP5			
C8H-CFH		INTF	INTE	OSCM		WDTCR	T1LDR	PCL	PCH	
D0H-D7H		IOP0				IOP4	IOP5			
D8H-DFH		T0CR	T0CNT	T1CR	T1CNT	BUZCR			MCR	
E0H-E7H		PUP0				PUP4	PUP5		INDF	
E8H-EFH		PWMCR0	PWMCR1	PWMCNT	PWMADT	PWMBDT	PWMCDDT	PWMPD		
F0H-F7H		STKR0L	STKR0H	STKR1L	STKR1H	STKR2L	STKR2H	STKR3L	STKR3H	
F8H-FFH		STKR4L	STKR4H	STKR5L	STKR5H	STKR6L	STKR6H	STKR7L	STKR7H	

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

直接寻址模式，是以指令的低 8 位为数据存储器地址，通过指令访问，寻址范围为 00H~FFH。例：通过直接寻址模式将数据 55H 写入数据存储器 10H 地址

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

间接寻址模式，是以 [FSR1:FSR0] 为数据存储器地址指针，通过 INDF 访问，寻址范围为 0000H~FFFFH。例：通过间接寻址模式将数据 55H 写入数据存储器 0010H 地址

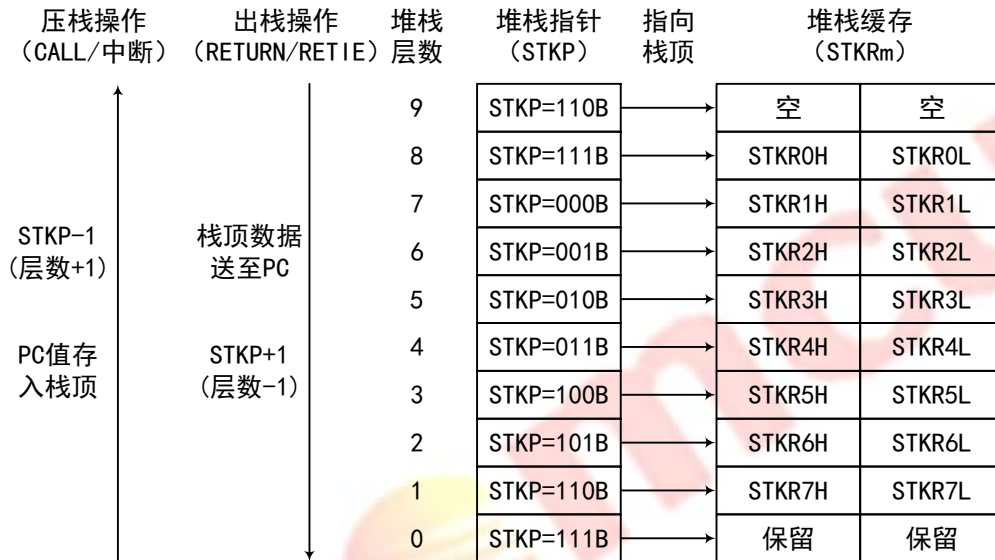
```
MOVAI    00H
MOVRA    FSR1
MOVAI    10H
MOVRA    FSR0
MOVAI    55H
MOVRA    INDF           ; 将数据 55H 写入 FSR1×256+FSR0 指向的数据存储器中
```

注：间接寻址模式可寻址 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，写入数据可能会更改其他地址中的数据。

3.4 堆栈

8 级 11 位堆栈缓存寄存器 STKRm (m=7-0)，位于特殊功能寄存器 SFR 区，3 位堆栈指针 STKP 位于杂项控制寄存器 MCR 中，指向堆栈栈顶。CPU 在响应中断或执行子程序调用指令时，先将 STKP 减 1，然后将 PC 值压入 STKP 所指栈顶的堆栈缓存寄存器中；当 CPU 执行中断或子程序返回指令时，先将栈顶数据送入 PC 中，再将 STKP 加 1 指向新的栈顶地址。若在堆栈满时压栈、或堆栈空时出栈，会导致数据错误。

堆栈操作示意



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: 间接寻址模式的指针低 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: 间接寻址模式的指针高 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
INDF	INDF7	INDF6	INDF5	INDF4	INDF3	INDF2	INDF1	INDF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	X	X	X	X	X	X	X	X

BIT[7:0] **INDF[7:0]** – 间接寻址寄存器

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

字操作高字节寄存器

	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]** – 字操作高字节寄存器

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

程序指针计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PCH	-	-	-	-	-	PC10	PC9	PC8
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2:0] **PC[10:8]** – 程序指针计数器高 3 位

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

BIT[7:0] **PC[7:0]** – 程序指针计数器低 8 位

程序指针计数器 (PC) 有以下几种操作模式:

- ◇ 顺序运行指令: $PC = PC + 1$;
- ◇ 分支指令 GOTO/CALL: $PC =$ 指令码低 11 位;
- ◇ 返回指令 RETIE/RETURN: $PC =$ 堆栈栈顶 (TOS);

对 PCL 操作指令:

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

注: 以 PCL 为目的寄存器的算术/逻辑操作指令中, 仅有加法指令产生进位时会影响 PCH 的值, 而其他指令均不会影响 PCH 的值。

CPU 状态寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PFLAG		-	LVD36	LVD24		C	DC	Z
R/W		-	R	R		R/W	R/W	R/W
初始值		-	0	0		0	0	0

BIT[5] **LVD36** – 3.6V 检测状态标志位（仅配置字 LVDVS 选择 3.6V 时有效）
 0: VDD 电压高于 3.6V（有回滞）；
 1: VDD 电压低于 3.6V；

BIT[4] **LVD24** – 2.4V 检测状态标志位（仅配置字 LVDVS 选择 2.4V 时有效）
 0: VDD 电压高于 2.4V（有回滞）；
 1: VDD 电压低于 2.4V；

注：因 LVD 电路有回滞特性（回滞电压典型值为 6%），所以 VDD 需上升至检测量值+6%后标志位才被清 0。

BIT[2] **C** – 进/借位标志位
 0: 加法运算时无进位；减法运算时有借位；移位后移出逻辑 0；比较运算结果 < 0；
 1: 加法运算时有进位；减法运算时无借位；移位后移出逻辑 1；比较运算结果 ≥ 0；

BIT[1] **DC** – 半字节进/借位标志位
 0: 加法运算时半字节无进位；减法运算时半字节有借位；
 1: 加法运算时半字节有进位；减法运算时半字节无借位；

BIT[0] **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；

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	-				STKP2	STKP1	STKP0
R/W	R/W	-				R/W	R/W	R/W
初始值	0	-				1	1	1

BIT[7] **GIE** – 中断总使能位
 0: 屏蔽所有中断；
 1: 由相应的中断使能位决定 CPU 是否响应中断源所触发的中断；

BIT[2:0] **STKP[2:0]** – 堆栈指针控制位

注：虽然堆栈指针的复位初始值已为 111B，但仍推荐在程序初始部分重新设定 STKP=7 并禁止程序再访问堆栈相关寄存器，以确保芯片能正常使用堆栈。

堆栈缓存寄存器 (m=0-7)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STKRmH	-	-	-	-	-	STKRmB10	STKRmB9	STKRmB8
R/W	-	-	-	-	-	R/W	R/W	R/W
初始值	-	-	-	-	-	0	0	0

BIT[2:0] STKRmB[10:8] – 堆栈缓存寄存器 STKRm 的高 3 位

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
STKRmL	STKRmB7	STKRmB6	STKRmB5	STKRmB4	STKRmB3	STKRmB2	STKRmB1	STKRmB0
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] STKRmB[7:0] – 堆栈缓存寄存器 STKRm 的低 8 位

3.6 用户配置字

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

芯片的用户配置字，定义如下：

符号	功能说明
PAGE	ROM 烧录模式设置： 1K 容量 MTP 模式，第 1 次烧录； 1K 容量 MTP 模式，第 2 次烧录； 2K 容量 OTP 模式；
HOSCS	系统高频时钟选择： FHOSC=内部高频 RC 振荡器时钟 FHIRC； FHOSC=外部 32768Hz 晶体振荡器时钟 FEXT； FHOSC=外部 455KHz 晶体振荡器时钟 FEXT； FHOSC=外部 1MHz 晶体振荡器时钟 FEXT； FHOSC=外部 2MHz 晶体振荡器时钟 FEXT； FHOSC=外部 4MHz 晶体振荡器时钟 FEXT； FHOSC=外部 8MHz 晶体振荡器时钟 FEXT； FHOSC=外部 16MHz 晶体振荡器时钟 FEXT； FHOSC=外部时钟输入 FECK (<1MHz)； FHOSC=外部时钟输入 FECK (<2MHz)； FHOSC=外部时钟输入 FECK (<4MHz)； FHOSC=外部 RC 振荡器时钟 FERC (<1MHz)； FHOSC=外部 RC 振荡器时钟 FERC (<2MHz)； FHOSC=外部 RC 振荡器时钟 FERC (<4MHz)； FHOSC=外部 RC 振荡器时钟 FERC (<8MHz)；

FCPUS	高频时钟下 FCPU 分频选择: (FHOSC 为 32MHz FHIRC 时, FCPU 不支持 2 分频) FCPU= FHOSC/2; FCPU= FHOSC/4; FCPU= FHOSC/8; FCPU= FHOSC/16; FCPU= FHOSC/32; FCPU= FHOSC/64; FCPU= FHOSC/128; FCPU= FHOSC/256;
RSTEN	RST 外部复位设置: P03 为外部复位脚; P03 为输入/输出脚;
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) 2.2V; 2.7V; 3.2V; 3.6V;
LVDVS	LVD 检测电压选择: 2.4V; 3.6V;
WDTM	WDT 模式设置: WDT 始终关闭; WDT 在低功耗模式下关闭; WDT 始终开启;
DRVS	端口驱动灌电流选择: 正常驱动 (Iol=16mA); 增强驱动 (Iol=25mA);
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

芯片为双时钟系统，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，部分模块的时钟还可在 F_{HOSC} 和 F_{LOSC} 之间切换。

系统高频时钟 F_{HOSC} 可通过配置字 $HOSCS$ 选择以下时钟：

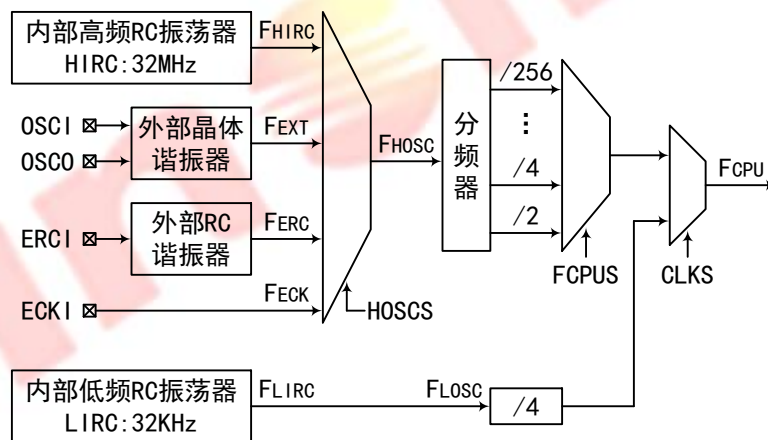
- ◇ 内部高频 RC 振荡器 $HIRC$ (32MHz) 时钟 F_{HIRC} ；
- ◇ 外部晶体振荡器 EXT (32768Hz/455KHz/1MHz~16MHz) 时钟 F_{EXT} ；
- ◇ 外部 RC 振荡器 ERC (0~8MHz) 时钟 F_{ERC} ；
- ◇ 外部时钟输入 ECK (0~4MHz) 时钟 F_{ECK} ；

系统低频时钟 F_{LOSC} 固定为内部低频 RC 振荡器 $LIRC$ (32KHz) 时钟 F_{LIRC} 。

CPU 为双时钟，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。选择 F_{HOSC} 时，CPU 的时钟频率 F_{CPU} 通过配置字 $FCPUS$ 选择；选择 F_{LOSC} 时， F_{CPU} 固定为 F_{LOSC} 的 4 分频。

WDT（看门狗）电路的时钟源固定为内部低频 RC 振荡器。

系统时钟示意图



4.1 内部高频 RC 振荡器

芯片内置 1 个振荡频率为 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 $HIRC$ 的 8 位微调校准寄存器 $OSCCAL$ 中，芯片复位后，寄存器自动加载出厂设定值作为初始值，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
OSCCAL	OSCCAL7	OSCCAL6	OSCCAL5	OSCCAL4	OSCCAL3	OSCCAL2	OSCCAL1	OSCCAL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	U	U	U	U	U	U	U	U

BIT[7:0] OSCCAL[7:0] – HIRC 频率微调校准位（复位初始值为出厂设定值）

注:

1. HIRC 最大调节范围为 (32MHz-4%) ~ (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. 需在 CPU 低速模式下调整 HIRC 频率, 待其稳定后才可切换回高速模式;
3. 步长增减的频率是非线性的, 最大约为 0.5%;
4. OSCCAL 中已存有出厂校准值, 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、WDT 定时器等电路。

4.3 外部晶体振荡器

芯片支持外接 32768Hz/455KHz/1MHz~16MHz 的晶体振荡器作为系统高频时钟源。

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。用户应使晶振离 OSCI/OSCO 引脚的距离尽可能短, 这样有助于振荡器的起振和振荡的稳定性。

下表是典型频率晶振选用电容 CG/CD 的推荐值和相应最低起振电压参考值。

晶振频率 (Hz)	电容 CG/CD (pF)	最低起振电压 (V)
16M	10	2.7
8M	10/20	2.4
4M	10/20	2.2
455K	220/470	2.2
32768	10/20	2.2

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

4.4 外部 RC 振荡器

芯片支持外接 RC 振荡器作为系统高频时钟源。外部 RC 振荡电路只需要和 ERCI 引脚连接，电容值不能低于 100pF，电阻值和电容值共同决定 RC 的振荡频率，最高支持 8MHz。

注：文档中所有提到的外部 RC 频率或 F_{ERC} ，均指外部 RC 振荡器的输出频率，其值为外部 RC 输入端口 ERCI 波形频率的 2 分频。

4.5 外部时钟输入

芯片支持外部 ECKI 引脚输入的数字时钟信号作为系统高频时钟源，外部时钟最高支持 4MHz。

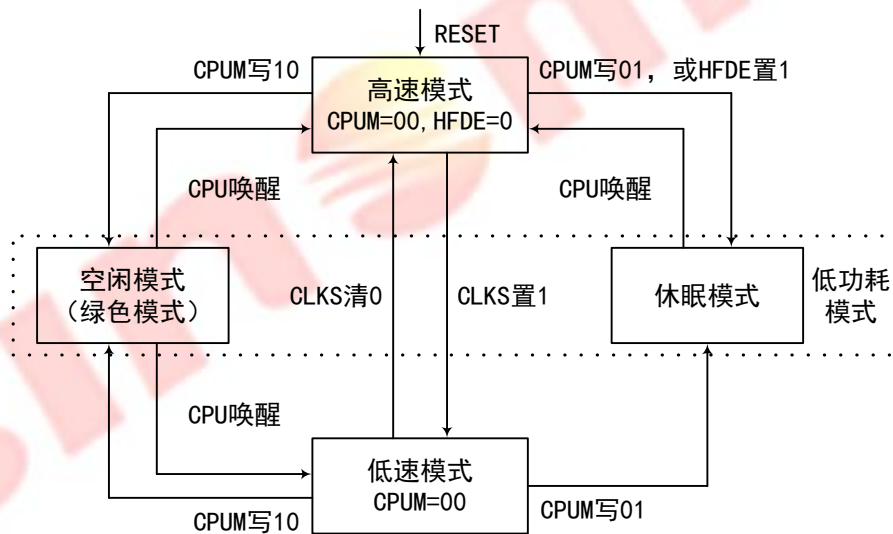
5 系统工作模式

芯片支持高速模式、低速模式、空闲模式和休眠模式等多种系统工作模式，其中高/低速模式为运行模式，CPU 运行；而空闲模式和休眠模式为低功耗模式，CPU 暂停工作。

工作模式	切入条件	系统状态
高速模式	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	
	空闲/休眠模式下，CPU 唤醒	
低速模式	高速模式下，CLKS 置 1	CPU 低速运行，高频时钟源由 HFDE 决定
	空闲/休眠模式下，CPU 唤醒	
空闲模式	高/低速模式下，CPUM 写 10	CPU 暂停，高频时钟源由 HFDE 决定
休眠模式	高速模式下，CPUM 写 01、或 HFDE 置 1	CPU 暂停，高/低频时钟源均停止
	低速模式下，CPUM 写 01	

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



工作模式寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OSCM	-	-	-	CPUM1	CPUM0	CLKS	HFDE	-
R/W	-	-	-	R/W	R/W	R/W	R/W	-
初始值	-	-	-	0	0	0	0	-

BIT[4:3] CPUM[1:0] – 系统工作模式控制位

CPUM[1:0]	系统工作模式
00	高/低速模式（硬件自动清 0）
01	进入休眠模式

10	进入空闲模式
11	保留

BIT[2] **CLKS** – CPU 时钟选择位
 0: F_{HOSC} 的分频时钟作为 F_{CPU};
 1: F_{LOSC} 的 4 分频时钟作为 F_{CPU};

BIT[1] **HFDE** – 高频时钟源控制位
 0: 使能高频时钟源;
 1: 关闭高频时钟源;

系统功能和模块在各工作模式下的状态

功能/模块的工作状态	高速模式	低速模式	空闲模式	休眠模式
内/外部高频时钟源	工作	通过 HFDE 控制	继续原状态	停止
内部低频 RC 振荡器	工作	工作	工作	由 WDTM 决定
CPU	工作	工作	停止	停止
看门狗定时器 WDT	由 WDTM 决定	由 WDTM 决定	由 WDTM 决定	由 WDTM 决定
T0	通过 T0EN 控制	通过 T0EN 控制	继续原状态, 溢出唤醒	暂停
T1	通过 T1EN 控制	通过 T1EN 控制	暂停	暂停
内部中断	全部有效	全部有效	仅 T0 中断有效	全部无效
外部中断	有效	有效	触发仅置标志位	无效
唤醒方法	-	-	P0/P5 电平变化, T0 溢出, 复位	P0/P5 电平变化, 复位

注: 未使用或未封出的引脚, 应将其对应的 IO 端口设置为输出、输入上拉或输入下拉等稳定态, 以免因引脚浮空而产生漏电流或非预期的中断唤醒。

在高/低速模式中, 可通过位操作指令 **BSET** 将 CPUM0 置 1 进入休眠模式、或将 CPUM1 置 1 进入空闲模式。但请注意, 因指令的预读寄存器机制及 CPU 的并行流水线架构, 若唤醒后的第一条指令又立即对寄存器 **OSCM** 进行操作, 则会再次进入休眠或空闲模式。

可按以下例程实现运行-休眠、运行-空闲的模式切换:

```

..... ; 运行模式
BSET      CPUM0 ; CPUM0 置 1, 进入休眠模式
NOP      ; 唤醒后第一条指令
..... ; 运行模式
BSET      CPUM1 ; CPUM1 置 1, 进入空闲模式
NOP      ; 唤醒后第一条指令
..... ; 运行模式
    
```

5.1 高速模式

高速模式下，系统（包括 CPU）的工作时钟由高频时钟 F_{HOSC} 提供。上电复位或其他复位触发后，系统以高速模式工作，当系统从休眠模式唤醒后也进入高速模式。

- ◇ CPU 以 F_{HOSC} 为时钟源，在其分频时钟下执行指令代码；
- ◇ 系统在 F_{HOSC} 下工作；所有模块均可在选定的时钟下工作；
- ◇ 高频时钟源和内部低频 RC 振荡器均正常工作；
- ◇ 系统可从高速模式转入其他任意模式；
- ◇ 系统可从低速模式转入高速模式；
- ◇ 系统从高速模式转入空闲模式后，再唤醒将返回到高速模式；
- ◇ 系统从休眠模式唤醒后，转入高速模式；

5.2 低速模式

低速模式下，系统（包括 CPU）的工作时钟由 F_{LOSC} 提供。高/低速模式的切换由寄存器位 CLKS 控制，CLKS 清 0 后转入高速模式，CLKS 置 1 后转入低速模式。从高速模式转入低速模式时，高频时钟源并不自动停止，可通过寄存器位 HFDE 关闭高频时钟源以降低功耗；而从低速模式转入高速模式时，需先通过 HFDE 清 0 开启高频时钟源，然后再通过 CLKS 清 0 切换模式。

- ◇ CPU 以 F_{LOSC} 的 4 分频时钟执行指令代码；
- ◇ 系统在 F_{LOSC} 的 4 分频时钟下工作；所有模块均可在选定的时钟下工作；
- ◇ 内部低频 RC 振荡器正常工作，高频时钟源由 HFDE 控制；
- ◇ 系统可从低速模式转入其他任意模式；
- ◇ 系统可从高速模式转入低速模式；
- ◇ 系统从低速模式转入空闲模式后，再唤醒将返回到低速模式；

5.3 休眠模式

休眠模式为低功耗模式，系统和 CPU 均停止工作，高频时钟源停止工作，内部低频 RC 振荡器根据 WDT 模式决定（若 WDT 始终工作则 LIRC 工作，否则 LIRC 停止）。休眠模式可由选定端口的电平变化来唤醒，系统从其他任意模式转入到休眠模式后，再唤醒均将以高速模式恢复工作。可通过寄存器位 CPUM 控制是否进入休眠模式，当 CPUM=01 时，系统进入休眠模式，唤醒后 CPUM 自动恢复成 00，CLKS 也自动清为 0，系统进入高速模式。

- ◇ CPU 停止工作，程序代码暂停执行；
- ◇ 包括内部高/低速 RC 振荡器、外部高速振荡器在内的所有时钟源均停止工作。
- ◇ 选定端口的电平变化，将唤醒系统；
- ◇ 系统从休眠模式唤醒后进入高速模式；

注：高速模式下 HFDE 置 1，系统也将进入休眠模式；在低速模式且 HFDE 置 1 时，直接将 CLKS 清 0 会误进入休眠模式。

5.4 空闲模式 (绿色模式)

空闲模式又称为绿色模式，也为低功耗模式，CPU 停止工作，但切入空闲模式前选定的时钟源仍保持工作，系统和内部模块也将在 F_{HOSC} 或 F_{CPU} 下继续工作。空闲模式下，选定端口的电平变化、或支持唤醒功能的定时器的溢出，均将唤醒系统。可通过寄存器位 CPUM 控制是否进入空闲模式，当 CPUM=10 时，系统进入空闲模式，唤醒后 CPUM 自动恢复成 00，系统按之前的模式恢复工作。

- ◇ CPU 停止工作，程序代码暂停执行；
- ◇ 此前作为系统时钟源的振荡器仍正常工作，其他振荡器的状态取决于工作模式的配置；
- ◇ 支持唤醒功能的定时器继续原工作状态；
- ◇ PWM 模块继续原工作状态，但溢出不唤醒系统；
- ◇ 选定端口的电平变化、或支持唤醒功能的定时器的溢出，将唤醒系统；
- ◇ 由高速模式切换到空闲模式，唤醒后系统返回到高速模式；
- ◇ 由低速模式切换到空闲模式，唤醒后系统返回到低速模式；

6 复位

6.1 复位条件

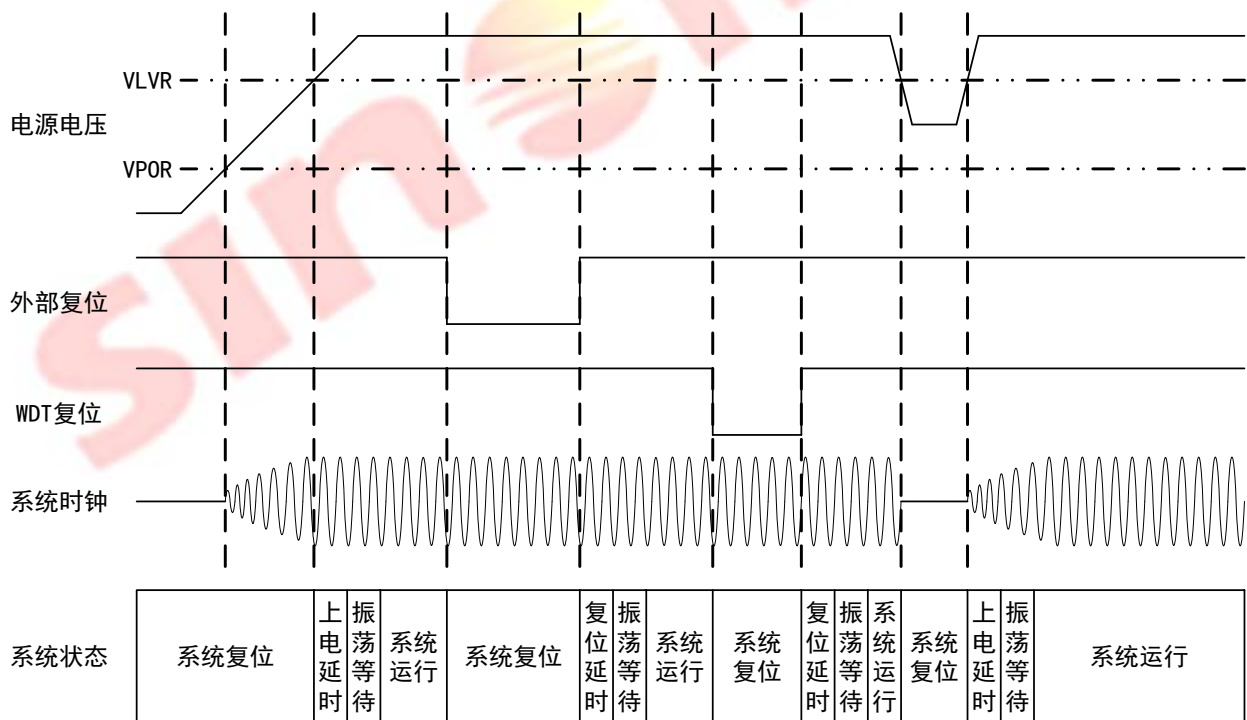
芯片共有如下几种复位方式：

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

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

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

下图是复位产生和系统工作状态之间时序关系的示意图：



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

6.2 上电复位

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

上电复位过程可以概括为以下几个步骤：

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

6.3 外部复位

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

6.4 低电压复位

芯片的低电压复位电压可通过配置字 $LVRVS$ 选择。电压检测电路有一定的回滞特性，回滞电压为 6% 左右（典型值），当电源电压下降至 LVR 电压时 LVR 复位有效，反之则电源电压需上升至 LVR 电压 +6% 后 LVR 复位才解除。

6.5 看门狗复位

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

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

7 I/O 端口

7.1 通用 I/O 功能

芯片的输入/输出端口包括一组 8 位端口 P0，和两组 5 位端口 P4、P5。所有 I/O 端口均支持施密特输入，除 P03 外的 I/O 端口均支持推挽输出。除用作通用数字 I/O 端口外，部分端口还具有外部中断输入、PWM 输出、或 ADC 模拟输入等复用功能。

端口数据寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOP0	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	0	0	0	0	0	0	0	0

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
IOP4	-	-	-	P44D	P43D	P42D	P41D	P40D
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4:0] P4nD – P4n 端口数据位 (n=4-0)

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
IOP5	-	-	-	P54D	P53D	P52D	P51D	P50D
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4:0] P5nD – P5n 端口数据位 (n=4-0)

注：开漏输出口输出高电平时，若端口电压不等于 VDD 电压则会产生漏电流。

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输入电平状态；
- 1: 端口作为输出口，读端口操作将读取端口的数据寄存器值；

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OEP4	-	-	-	P44OE	P43OE	P42OE	P41OE	P40OE

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

BIT[4:0] **P4nOE** – P4n 端口输出使能位 (n=4-0)
 0: 端口作为输入口, 读端口操作将读取端口的输入电平状态;
 1: 端口作为输出口, 读端口操作将读取端口的数据寄存器值;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
OEP5	-	-	-	P54OE	P53OE	P52OE	P51OE	P50OE
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4:0] **P5nOE** – P5n 端口输出使能位 (n=4-0)
 0: 端口作为输入口, 读端口操作将读取端口的输入电平状态;
 1: 端口作为输出口, 读端口操作将读取端口的数据寄存器值;

7.2 内部上拉电阻

所有端口均有内部上拉电阻, 且均有单独的寄存器位控制其上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
PUP4	-	-	-	P44PU	P43PU	P42PU	P41PU	P40PU
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4:0] **P4nPU** – P4n 端口上拉电阻控制位 (n=4-0)
 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
PUP5	-	-	-	P54PU	P53PU	P52PU	P51PU	P50PU
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4:0] **P5nPU** – P5n 端口上拉电阻控制位 (n=4-0)
 0: 端口内部上拉电阻无效;
 1: 端口内部上拉电阻有效;

7.3 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	AN9EN	-	-	-	-	-	-	AN5EN
R/W	R/W	-	-	-	-	-	-	R/W
初始值	0	-	-	-	-	-	-	0

BIT[7,0] **ANnEN** – ANn 端口数字功能控制位 (n=9,5)
 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	-	-	-	AN6EN	AN7EN	AN8EN	-	-
R/W	-	-	-	R/W	R/W	R/W	-	-
初始值	-	-	-	0	0	0	-	-

BIT[4:2] **ANnEN** – ANn 端口数字功能控制位 (n=6-8)
 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
ADIOS2	-	-	-	AN4EN	AN3EN	AN2EN	AN1EN	AN0EN
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4:0] **ANnEN** – ANn 端口数字功能控制位 (n=4-0)
 0: 使能端口的数字 I/O 功能;
 1: 关闭端口的数字 I/O 功能;

7.4 端口电平唤醒

部分端口支持输入电平变化唤醒功能，并可通过寄存器位单独使能或关闭。使能唤醒功能的端口处于输入状态时，端口上的电平变化可将系统从低功耗模式中唤醒。

注：应用端口电平唤醒功能时，端口不能浮空且需外部输入电平稳定，否则易产生误触发。

端口唤醒控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WKCR0	P07WK	P06WK	P05WK	P04WK	P03WK	P02WK	P01WK	P00WK
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] **P0nWK** – P0n 端口电平唤醒使能位 (n=7-0)

- 0: 关闭端口电平变化唤醒功能;
- 1: 使能端口电平变化唤醒功能;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WKCR5	-	-	-	P54WK	P53WK	P52WK	P51WK	P50WK
R/W	-	-	-	R/W	R/W	R/W	R/W	R/W
初始值	-	-	-	0	0	0	0	0

BIT[4:0] **P5nWK** – P5n 端口电平唤醒使能位 (n=4-0)

- 0: 关闭端口电平变化唤醒功能;
- 1: 使能端口电平变化唤醒功能;

8 定时器 TIMER

8.1 看门狗定时器 WDT

看门狗定时器 WDT 的时钟源为内部低频 RC 振荡器，WDT 计数器溢出将复位芯片。

可通过配置字 WDTM 设置 WDT 工作模式。若选择始终开启，则 WDT 一直工作，任意模式下 WDT 溢出均将复位芯片。若选择低功耗模式下关闭，则 WDT 在休眠/空闲模式下自动停止、在其他方式唤醒 CPU 后恢复工作。

WDT 控制寄存器 WDTCR 写 5AH 将清零 WDT 计数器。

WDT 溢出时间 = 8192 / 内部低频 RC 振荡器频率 (FLIRC)。

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

WDT 控制寄存器

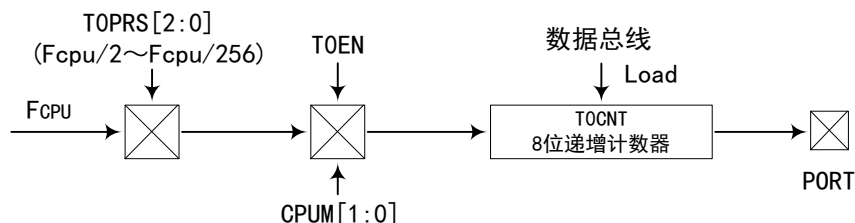
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
WDTCR	WDTCR7	WDTCR6	WDTCR5	WDTCR4	WDTCR3	WDTCR2	WDTCR1	WDTCR0
R/W	W	W	W	W	W	W	W	W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] WDTCR[7:0] – WDT 控制寄存器，写 5AH 将清零 WDT 计数器

8.2 定时器 T0

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

- ◇ 可通过预分频比设置计数频率；
- ◇ 支持溢出中断和溢出唤醒功能；



定时器 T0，时钟源固定为 CPU 时钟，可通过寄存器位 TOPRS 选择预分频比，CPU 时钟通过预分频器后产生 T0 计数器 TOCNT 的计数时钟。预分频比可选择 2~256 分频，对 TOCNT 的写操作将清零预分频计数器，而预分频比保持不变。当 TOEN=1 时，TOCNT 递增计数，从 FFH 溢出到 0 时产生溢出信号并触发中断，中断标志 TOIF 将被置 1。若 T0 在空闲模式下仍继续工作，则溢出时将唤醒系统。

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	TOEN	TOPRS2	TOPRS1	TOPRS0	保留	保留	-	-
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] **TOEN** – 定时器 T0 使能位
 0: 关闭定时器 T0;
 1: 开启定时器 T0;

BIT[6:4] **TOPRS[2:0]** – T0 预分频比选择位

TOPRS[2:0]	T0 时钟频率
000	FCPU/256
001	FCPU/128
010	FCPU/64
011	FCPU/32
100	FCPU/16
101	FCPU/8
110	FCPU/4
111	FCPU/2

BIT[3:2] 保留位，需固定写“00”

定时器 T0 计数器

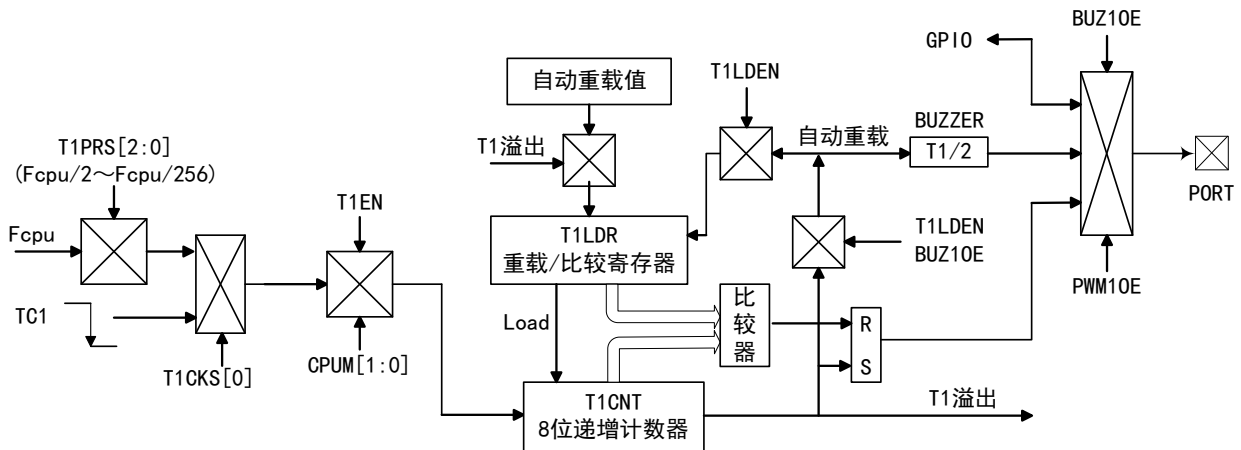
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
TOCNT	TOCNT7	TOCNT6	TOCNT5	TOCNT4	TOCNT3	TOCNT2	TOCNT1	TOCNT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

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

8.3 定时器 T1

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

- ◇ 时钟源可选择 FCPU，或外部 TC1 输入；选择 FCPU 时可通过预分频比设置计数频率；
- ◇ 支持 PWM 或 BUZ 输出；
- ◇ 用于定时/计数/BUZ 时，可通过重载/比较寄存器控制计数周期；
- ◇ 用于 PWM 输出时，可通过重载/比较寄存器设置 PWM 占空比；
- ◇ 支持溢出中断，低功耗模式下暂停工作；



定时器 T1，可通过寄存器位 T1PRS 选择 CPU 时钟预分频比，通过 T1CKS 选择外部时钟、或 CPU 时钟经过预分频器后的内部时钟，作为 T1 计数器 T1CNT 的计数时钟。预分频比可选择 2~256 分频，对 T1CNT 的写操作将清零预分频计数器，而预分频比保持不变。

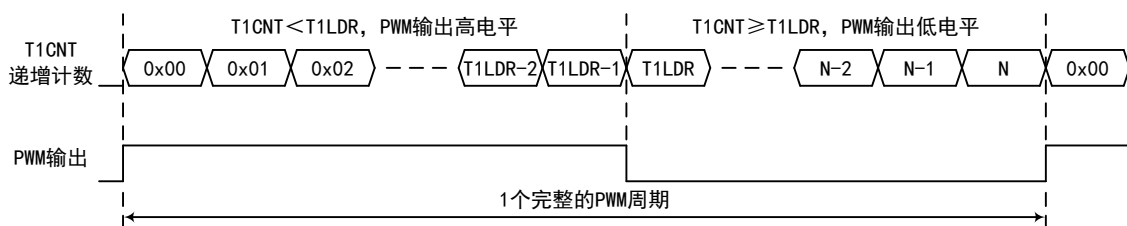
PWM1OE=0 时定时器 T1 可实现定时/计数功能：当 T1EN=0 时，T1CNT 保持不变，写重载/比较寄存器 T1LDR 将立即载入 T1CNT 中；当 T1EN=1 时，T1CNT 递增计数，从 FFH 溢出到 0 时产生溢出信号并触发中断，中断标志 T1IF 将被置 1，同时若已通过 T1LDEN 使能自动重载功能，则 T1 自动将当前 T1LDR 值载入 T1CNT 中重新开始计数，否则 T1 将重新从 0 开始计数。

T1LDR 配有 1 个 8 位缓冲器，写 T1LDR 会将数据先写入该缓冲器中，在 T1 溢出时才将缓冲器中的值自动载入 T1LDR 和 T1CNT 中。通过 T1LDR 调整 T1 的计数周期，只能从下一个周期开始有效，若要首个计数周期正确，需在使能 T1 前对 T1LDR 和 T1CNT 写同样的值。

PWM1OE=0 时定时器 T1 可实现 BUZ 功能：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。

PWM1OE=1 时定时器 T1 可实现 PWM 功能，端口将输出 PWM 波形。每个 PWM 周期内，T1CNT 从 0 开始递增计数，并与重载/比较寄存器 T1LDR 的值比较，当计数到与 T1LDR 相等时，PWM1 信号变为低电平；当计数溢出时，PWM1 信号变为高电平。写 T1LDR 操作仅将数据写入缓冲器，T1 溢出时才载入 T1LDR 中，所以通过 T1LDR 调整 PWM1 的占空比，只能从下一个周期开始有效。

PWM1OE=1 时 T1LDEN 和 BUZ1EN 用于设置 T1CNT 的最大计数值 N，即设置 PWM1 的计数周期（详见寄存器位说明）。



系统在高/低速模式之间切换时，T1 时钟若选择 CPU 时钟则同样会在系统高/低频时钟源之间自动切换。低功耗模式下，T1 暂停工作。

定时器 T1 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1CR	T1EN	T1PRS2	T1PRS1	T1PRS0	T1CKS0	T1LDEN	BUZ1OE	PWM1OE
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[6:4] **T1PRS[2:0]** – T1 预分频比选择位

T1PRS[2:0]	T1 时钟频率
000	FCPU/256
001	FCPU/128
010	FCPU/64
011	FCPU/32
100	FCPU/16
101	FCPU/8
110	FCPU/4
111	FCPU/2

BIT[3] **T1CKS0** – T1 时钟源选择位
 0: T1 时钟源为 CPU 时钟;
 1: T0 时钟源为外部 TC1 输入下降沿;

BIT[2] **T1LDEN** – T1 自动重载使能位 (仅 PWM1OE=0 时有效)
 0: 禁止 T1 自动重载;
 1: 使能 T1 自动重载;

BIT[1] **BUZ1OE** – BUZ1 端口输出使能位 (仅 PWM1OE=0 时有效)
 0: 禁止端口输出 BUZ 波形;
 1: 允许端口输出 BUZ 波形;

BIT[2,1] **T1LDEN, BUZ1OE** – PWM1 周期选择位 (仅 PWM1OE=1 时有效)

T1LDEN : BUZ1OE	N	T1CNT 范围	T1CNT/T1LDR 有效值 (二进制)	PWM1 周期
00	255	0~255	0000 0000 ~ 1111 1111	256 个时钟周期
01	63	0~63	xx00 0000 ~ xx11 1111	64 个时钟周期
10	31	0~31	xxx0 0000 ~ xxx1 1111	32 个时钟周期
11	15	0~15	xxxx 0000 ~ xxxx 1111	16 个时钟周期

BIT[0] **PWM1OE** – 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 PWM 波形;
 1: 使能 PWM1 功能, 并允许端口输出 PWM 波形;

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	0	0	0	0	0	0	0	0

BIT[7:0] **T1CNT[7:0]** – T1 计数器，为可读写的递增计数器

定时器 T1 重载/比较寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
T1LDR	T1LDR7	T1LDR6	T1LDR5	T1LDR4	T1LDR3	T1LDR2	T1LDR1	T1LDR0
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] **T1LDR[7:0]** – T1 重载/比较寄存器，用于设置 T1 的计数周期或 PWM1 的占空比

9 脉宽调制模块 PWM

9.1 PWM 概述

芯片内置 1 个 8 位脉宽调制模块 PWM，包含 1 个递减计数器、1 个周期寄存器和 3 个比较寄存器。PWM 模块支持共周期的 3 路 8 位 PWM 功能（PWMA/PWMB/PWMC），可通过对应的比较寄存器分别设置每路 PWM 的占空比。

PWM 计数器的时钟源为 F_{CPU}，可设置不同的分频比以获得不同周期的 PWM 波形。

当 PWM 计数器溢出时触发重载动作，自动将周期寄存器和占空比寄存器的值重载至各自的缓存器中。溢出信号也会将 PWM 中断标志 PWMIF 置 1，若 PWM 中断使能则产生 PWM 中断。

3 路 PWM 输出具有独立的电平控制，可分别使能/禁止端口输出 PWM 波形，并可分别设置输出波形为先低后高或先高后低的 PWM 波形。

PWM_n 占空比的计算如下（PWM_nOF=0，PWM_n 正常输出格式，n=C-A）：

- ◇ PWM_n 高电平时间 = (PWMPD+1 - PWMnDT) × PWMCNT 计数时钟周期
- ◇ PWM_n 周期 = (PWMPD+1) × PWMCNT 计数时钟周期
- ◇ PWM_n 占空比 = 高电平时间 / 周期 = (PWMPD+1 - PWMnDT) / (PWMPD+1)

9.2 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	PWMEN	PWMPRS2	PWMPRS1	PWMPRS0	-	PWMCOF	PWMBOF	PWMAOF
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] **PWMEN** – PWM 使能位
 0: 关闭 PWM;
 1: 开启 PWM;

BIT[6:4] **PWMPRS[2:0]** – PWM 预分频比选择位

PWMPRS[2:0]	PWM 时钟频率
000	F _{CPU} /64
001	F _{CPU} /32
010	F _{CPU} /16
011	F _{CPU} /8

100	FCPU/4
101	FCPU/2
110	FCPU
111	保留

BIT[2:0] **PWMnOF** – PWMn 输出格式选择位 (n=C-A)
 0: PWMn 正常输出, 每周先输出高电平再输出低电平;
 1: PWMn 输出取反, 每周先输出低电平再输出高电平;

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCRI	PWMCOE	PWMBOE	PWMAOE	-	-	-	-	-
R/W	R/W	R/W	R/W	-	-	-	-	-
初始值	0	0	0	-	-	-	-	-

BIT[7:5] **PWMnOE** – PWMn 端口输出使能位 (n=C-A)
 0: 禁止端口输出 PWM 波形;
 1: 允许端口输出 PWM 波形;

PWM 计数器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCNT	PWMCNT7	PWMCNT6	PWMCNT5	PWMCNT4	PWMCNT3	PWMCNT2	PWMCNT1	PWMCNT0
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] **PWMCNT[7:0]** – PWM 计数器, 为可读写的递减计数器

PWM 周期寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMPD	PWMPD7	PWMPD6	PWMPD5	PWMPD4	PWMPD3	PWMPD2	PWMPD1	PWMPD0
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] **PWMPD[7:0]** – PWM 周期寄存器, 用于设置 PWM 的计数周期

PWM 比较寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMADT	PWMADT7	PWMADT6	PWMADT5	PWMADT4	PWMADT3	PWMADT2	PWMADT1	PWMADT0
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] **PWMADT[7:0]** – PWM 比较寄存器 A, 用于设置 PWMA 的占空比

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMBDT	PWMBDT7	PWMBDT6	PWMBDT5	PWMBDT4	PWMBDT3	PWMBDT2	PWMBDT1	PWMBDT0
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] **PWMBDT[7:0]** – PWM 比较寄存器 B，用于设置 PWMB 的占空比

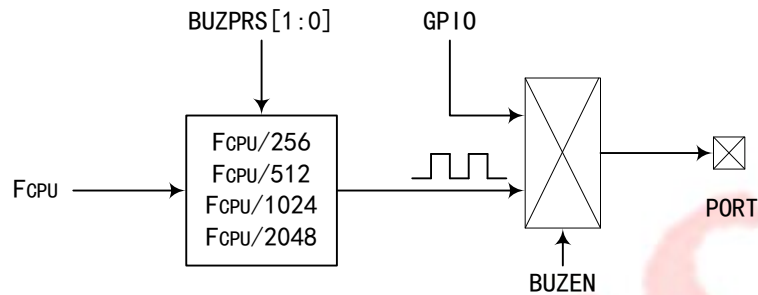
	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
PWMCDT	PWMCDT7	PWMCDT6	PWMCDT5	PWMCDT4	PWMCDT3	PWMCDT2	PWMCDT1	PWMCDT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	0	0	0	0	0	0	0	0

BIT[7:0] **PWMCDT[7:0]** – PWM 比较寄存器 C，用于设置 PWMC 的占空比

10 蜂鸣器输出 BUZZER

10.1 BUZZER 概述

芯片内置 1 个具有独立计数器且频率可设置的蜂鸣器输出电路 BUZZER，与 I/O 端口复用引脚。BUZZER 输出波形的占空比固定为 50%，其时钟源为 F_{CPU}，可设置不同的分频比以获得不同的输出频率。BUZEN=1 时，端口输出蜂鸣器驱动波形；BUZEN=0 时，端口恢复为 I/O 端口状态。



10.2 BUZZER 相关寄存器

BUZ 控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
BUZCR	BUZEN	BUZPRS1	BUZPRS0	-	-	-	-	-
R/W	R/W	R/W	R/W	-	-	-	-	-
初始值	0	0	0	-	-	-	-	-

BIT[7] **BUZEN** – BUZZER 使能位

0: 关闭 BUZZER;

1: 开启 BUZZER;

BIT[6:5] **BUZPRS[1:0]** – BUZ 预分频比选择位

BUZPRS[1:0]	BUZ 时钟频率	BUZZER 输出频率		
		F _{CPU} =1MHz	F _{CPU} =2MHz	F _{CPU} =4MHz
00	F _{CPU} /256	4KHz	8KHz	16KHz
01	F _{CPU} /512	2KHz	4KHz	8KHz
10	F _{CPU} /1024	1KHz	2KHz	4KHz
11	F _{CPU} /2048	0.5KHz	1KHz	2KHz

11 模数转换器 ADC

11.1 ADC 概述

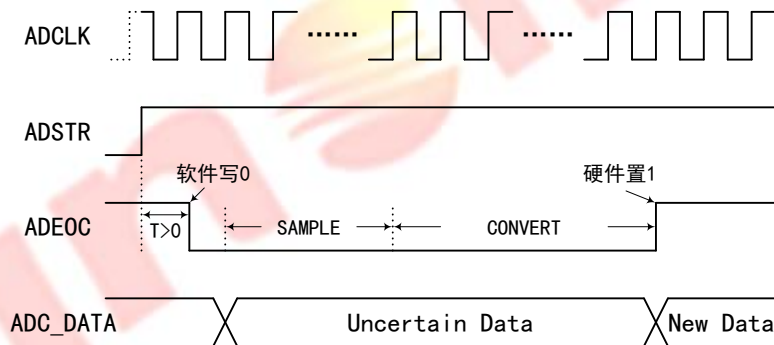
芯片内置 1 个 12 位高精度逐次逼近型的模数转换器 ADC。

- ◇ 10 路外部通道：AN0~AN9；2 路内部通道：GND、VDD/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：FCPU 的 4/8/32/64 分频；
- ◇ 支持零点校准；

ADC 模块可通过寄存器位 ADEN 开启，通过 ADCHS 选择转换的模拟通道，通过 GCHS 控制转换输入通道的开/关，通过 ADCKS 选择转换时钟，ADSPS 选择采样时间。ADSTR 为转换启动控制位，ADEOC 为转换状态标志位。ADSTR 写 1 将启动模数转换；转换完成后 ADSTR 自动清 0、ADEOC 自动置 1，结果存入 ADRH/ADRL 中，同时中断标志 ADIF 置 1 触发 ADC 中断。

ADC 的采样 (SAMPLE) 时间可选择 8/15 个 ADCLK (即 ADC 时钟周期)，转换 (CONVERT) 时间固定为 12 个 ADCLK，一次 ADC 转换为 20/27 个 ADCLK。

ADC 转换时序如下图所示：



注：

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

11.2 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
ADCRO	ADEN	ADSTR	ADEOC	GCHS	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	0	0	0	0	0	0	0

BIT[7] **ADEN** – ADC 使能位

0: 关闭 ADC;

1: 开启 ADC;

BIT[6] **ADSTR** – ADC 启动控制位

0: AD 转换结束后自动清 0;

1: 写 1 启动 AD 转换;

BIT[5] **ADEOC** – ADC 转换状态标志位

0: AD 转换中;

1: AD 转换结束;

BIT[4] **GCHS** – ADC 转换输入通道控制位

0: 关闭输入通道;

1: 开启输入通道;

BIT[3:0] **ADCHS[3:0]** – ADC 输入通道选择位

ADCHS[3:0]	ADC 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	GND
1011	-
1100	-
1101	VDD/4
1110	-
1111	-

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCR1	ADRSEL	ADCKS2	ADCKS1	ADCKS0	ADSPS3	ADSPS2	ADSPS1	ADSPS0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
初始值	1	0	0	0	1	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 8 位数据, 存入 ADRH[7:0];
 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[6:4] **ADCKS[2:0]** – ADC 转换时钟选择位

ADCKS[2:0]	ADC 转换时钟 F_{ADC}
000	F _{CPU} /64
001	F _{CPU} /32
010	F _{CPU} /4
011	F _{CPU} /8
其他	保留

BIT[3:0] **ADSPS[3:0]** – ADC 采样时间选择位

ADSPS[3:0]	ADC 采样时间
1000	8 个 ADCLK
1111	15 个 ADCLK
其他	保留

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADCR2	EVRS	-	-	-	-	-	VRS1	VRS0
R/W	R/W	-	-	-	-	-	R/W	R/W
初始值	0	-	-	-	-	-	0	0

BIT[7] **EVRS** – ADC 外部参考电压控制位
 0: ADC 参考电压由 VRS[1:0]决定;
 1: ADC 参考电压为外部参考电压 V_{ER} (端口 VER1 的输入电压);

BIT[1:0] **VRS[1:0]** – ADC 参考电压选择位

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

ADC 转换结果寄存器

ADRSEL=0 时, 转换结果为 8 位:

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
ADRH	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]** – 8 位 ADC 转换结果

ADRSEL=1 时，转换结果为 12 位：

	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
ZEROTR	ZOTEN1	ZOTEN0	-	ZEROTR4	ZEROTR3	ZEROTR2	ZEROTR1	ZEROTR0
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

BIT[7:6] **ZOTEN[1:0]** – ADC 零点偏移修调使能位（复位初始值为出厂设定值）

ZOTEN[1:0]	ADC 零点偏移修调控制
01	ADC 零点偏移修调有效
其他	ADC 零点偏移修调无效

BIT[4] **ZEROTR[4]** – ADC 零点偏移修调方向选择位（复位初始值为出厂设定值）

- 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[3:0] **ZEROTR[3:0]** – ADC 零点偏移修调电压选择位（复位初始值为出厂设定值）

ZEROTR[3:0]	修调电压 (典型值)
0000	0mV
0001	1mV
0010	2mV
...	...
1110	14mV
1111	15mV

11.3 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 ADC 转换时钟；
- (4) 若采样时间可选，则设置 ADSPS，选择适当的 ADC 采样时间；
- (5) 若参考电压可选，则设置 EVRS/VRS，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；GCHS 置 1，开启转换输入通道；
- (8) 设置 ADCHS，选择 ADC 转换通道；
- (9) 延时等待电路稳定后，ADSTR 写 1，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；

11.4 ADC 零点偏移修调流程

- (1) 设置 ADC 输入通道为内部接地，设置 ADC 时钟、采样时间等参数，设置 ZOTEN=01；
- (2) 设置 ZERO TR[4]=0、ZERO TR[3:0]=00H，进行 ADC 转换：
 - ✧ 若 ADC 结果为 0，则执行（4）；
 - ✧ 若 ADC 结果不为 0，则执行（3）；
- (3) ZERO TR[3:0]加 1 后进行 ADC 转换：
 - ✧ 若 ADC 结果为 0，则跳至（6）；
 - ✧ 若 ADC 结果不为 0，则循环执行（3），直到结果为 0 或 ZERO TR[3:0]=0FH，跳至（6）；
- (4) 设置 ZERO TR[4]=1、ZERO TR[3:0]=0FH，进行 ADC 转换：
 - ✧ 若 ADC 结果为 0，则跳至（6）；
 - ✧ 若 ADC 结果不为 0，则执行（5）；
- (5) ZERO TR[3:0]减 1 后进行 ADC 转换：
 - ✧ 若 ADC 结果为 0，则跳至（6）；
 - ✧ 若 ADC 结果不为 0，则循环执行（5），直到结果为 0 或 ZERO TR[3:0]=00H，跳至（6）；
- (6) ZERO TR[4:0]的值即为零点偏移最佳修调结果，修调流程结束，后续 ADC 工作时直接使用，不需要再次修调。

12 中断

芯片的中断源包括外部中断（INT0）、定时器中断（T0~T1）、ADC 中断和 PWM 中断等。可通过中断总使能位 GIE 屏蔽所有中断。

CPU 响应中断的过程如下：

- ◇ CPU 响应中断源触发的中断请求时，自动将当前指令的下一条要执行指令的地址压栈保存，自动清 0 中断总使能位 GIE 以暂停响应后续中断。与复位不同，硬件中断不停止当前指令的执行，而是暂时挂起中断直到当前指令执行完成。
- ◇ CPU 响应中断后，程序跳到中断入口地址（0008H）开始执行中断服务程序，中断服务程序应先通过 PUSH 指令保存累加器 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，应先通过 POP 指令恢复累加器 A 和状态寄存器 PFLAG，然后执行 RETIE 返回主程序。此时芯片将自动恢复 GIE 为 1，然后从堆栈取出 PC 值，从中断产生时当前指令的下一条指令继续执行。

12.1 外部中断

芯片有 1 路外部中断源 INT0，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 INTOIF 将被置 1，若中断总使能位 GIE 为 1 且外部中断使能位 INTOIE 为 1，则产生外部中断。

外部中断控制寄存器

	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
EINTCR	-	-	-	INTOM1	INTOM0	-	-	-
R/W	-	-	-	R/W	R/W	-	-	-
初始值	-	-	-	0	0	-	-	-

BIT[4:3] INTOM[1:0] – 外部中断 INT0 触发方式选择位

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

12.2 定时器中断

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

12.3 ADC 中断

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

12.4 PWM 中断

PWM 模块在计数溢出时触发 PWM 中断，中断标志 PWMIF 将被置 1，若中断总使能位 GIE 为 1 且 PWM 中断使能位 PWMIE 为 1，则产生 PWM 中断。

12.5 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	ADIE	-	T1IE	TOIE	PWMIE	-	-	INTOIE
R/W	R/W	-	R/W	R/W	R/W	-	-	R/W
初始值	0	-	0	0	0	-	-	0

BIT[7] ADIE – ADC 中断使能位
 0: 屏蔽 ADC 中断;
 1: 使能 ADC 中断;

BIT[5] 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[4] TOIE – 定时器 T0 中断使能位
 0: 屏蔽定时器 T0 中断;
 1: 使能定时器 T0 中断;

BIT[3] PWMIE – PWM 中断使能位
 0: 屏蔽 PWM 中断;
 1: 使能 PWM 中断;

BIT[0] INTOIE – INTO 中断使能位
 0: 屏蔽 INTO 中断;
 1: 使能 INTO 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	ADIF	-	T1IF	T0IF	PWMIF	-	-	INTOIF
R/W	R/W	-	R/W	R/W	R/W	-	-	R/W
初始值	0	-	0	0	0	-	-	0

BIT[7] 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] 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[4] 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[3] PWMIF – PWM 中断标志位
 0: 未触发 PWM 中断;
 1: 已触发 PWM 中断, 需软件清 0;

BIT[0] INTOIF – INTO 中断标志位
 0: 未触发 INTO 中断;
 1: 已触发 INTO 中断, 需软件清 0;

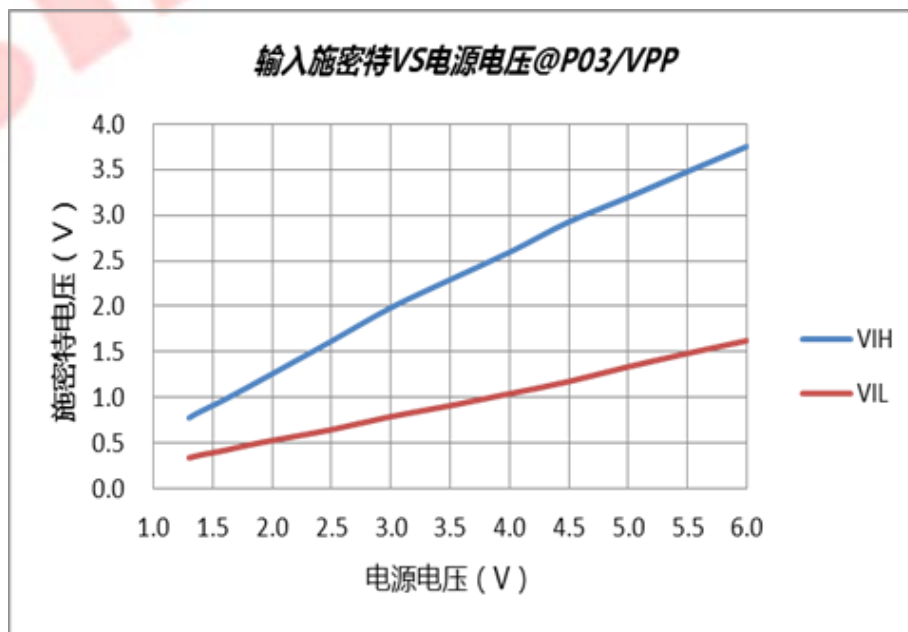
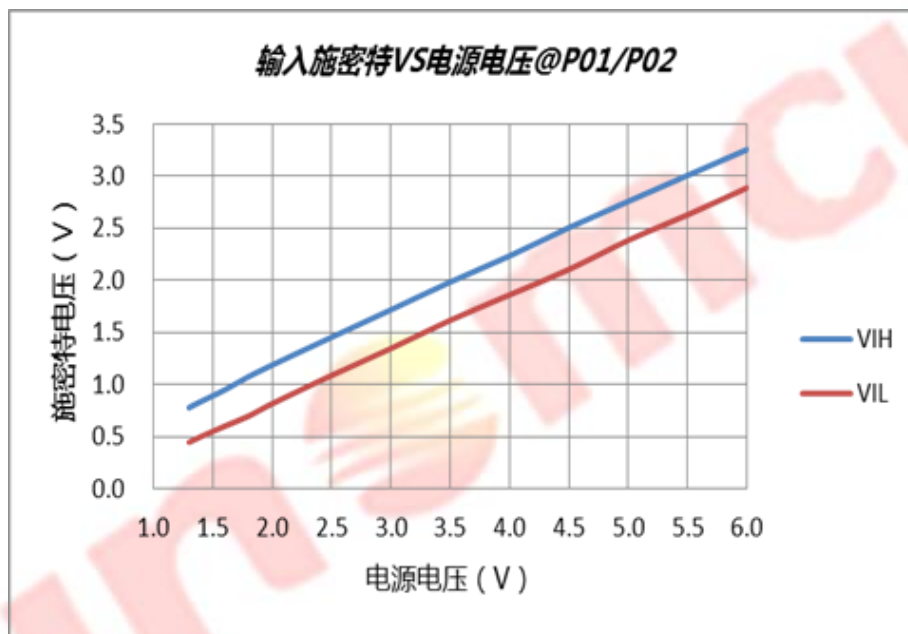
13 特性曲线

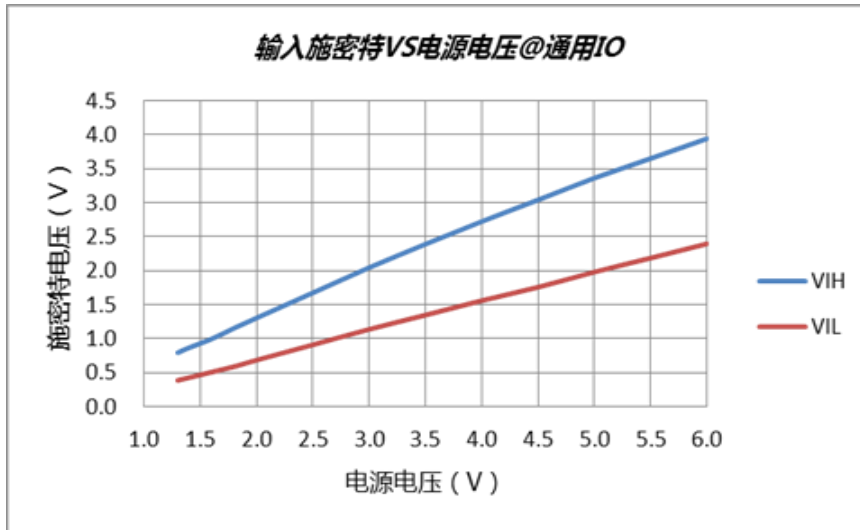
注:

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

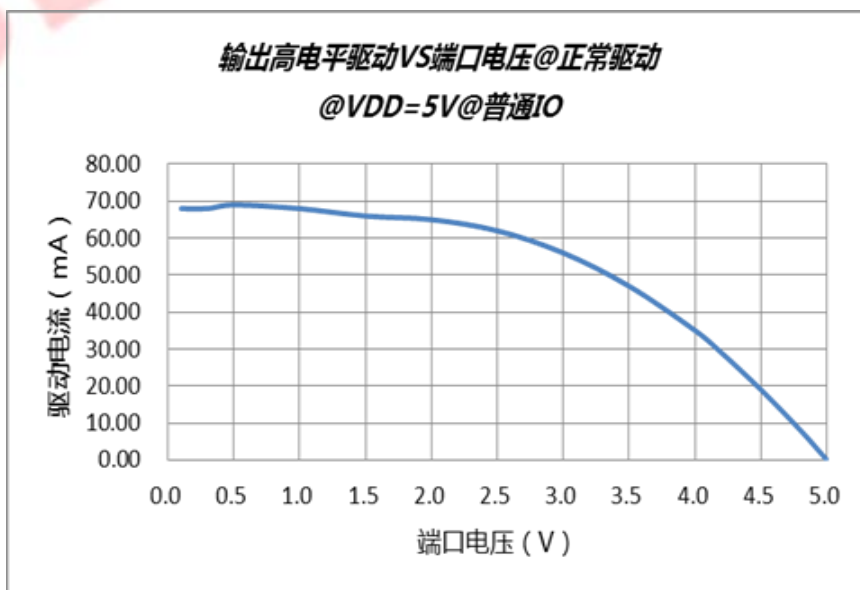
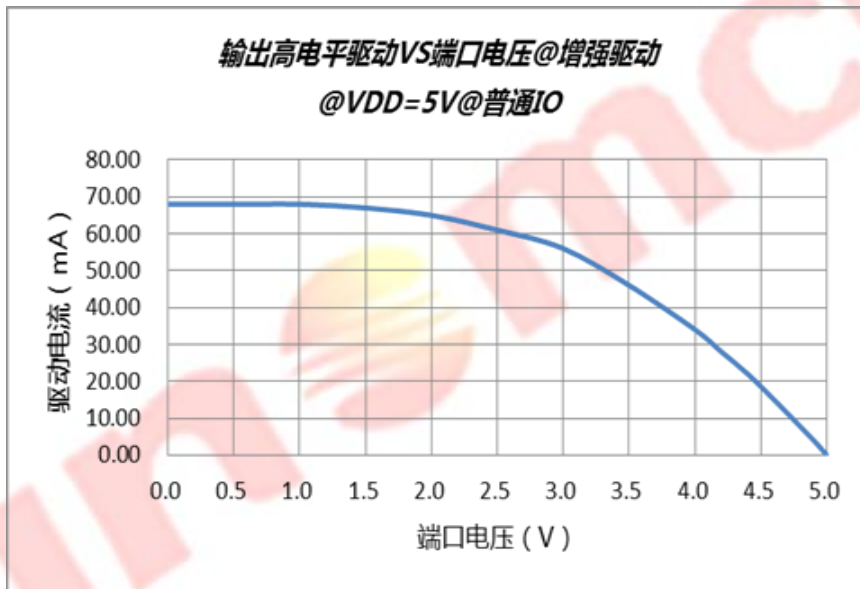
13.1 I/O 特性

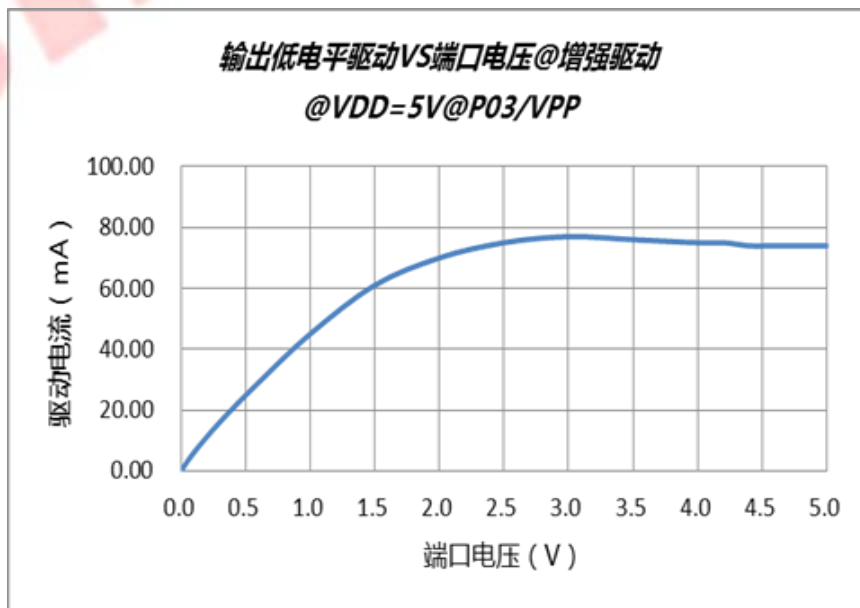
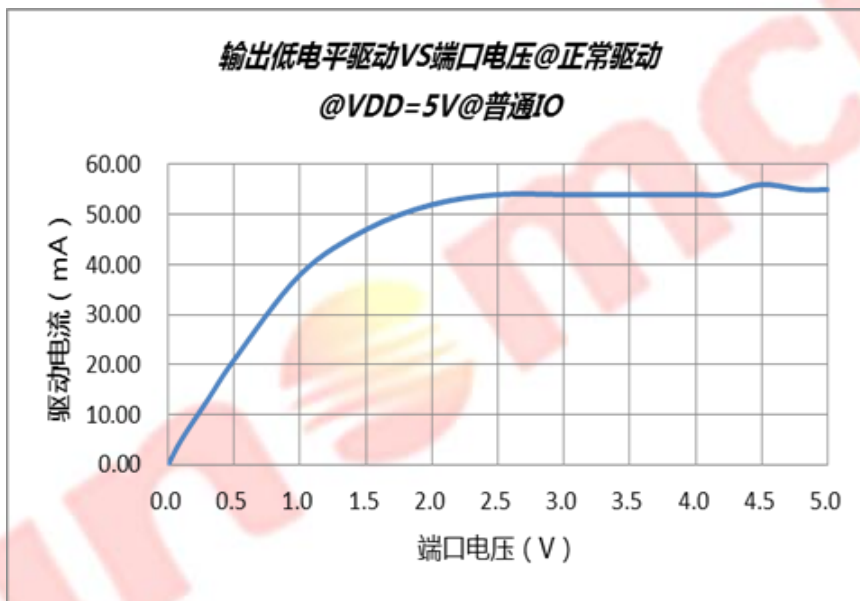
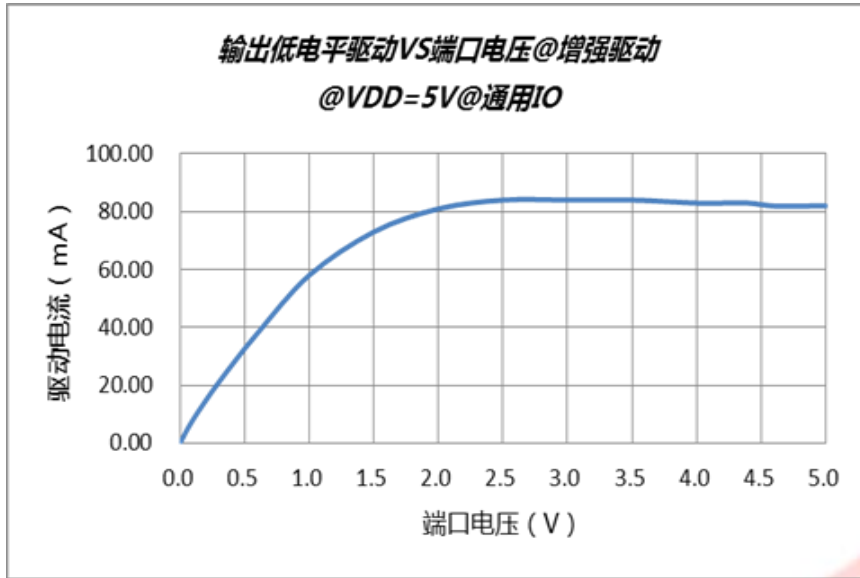
输入 SMT 阈值电压 VS 电源电压

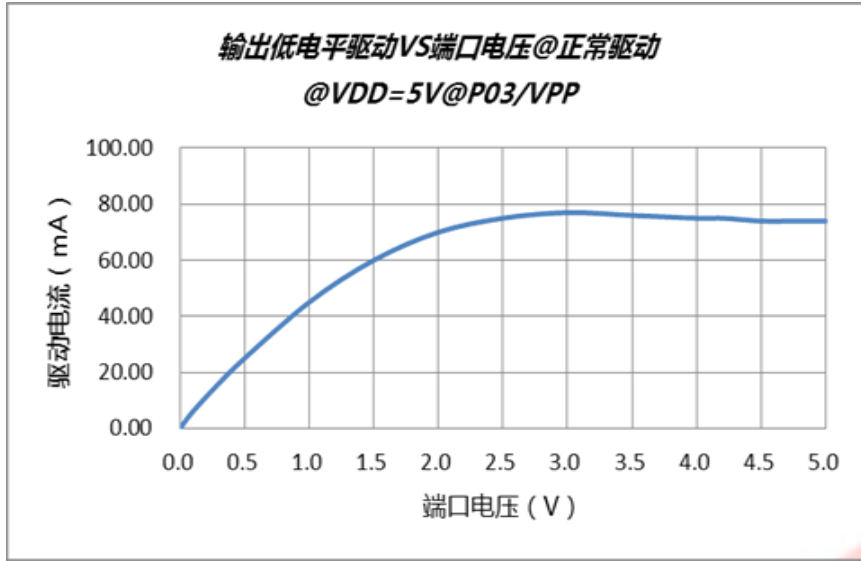




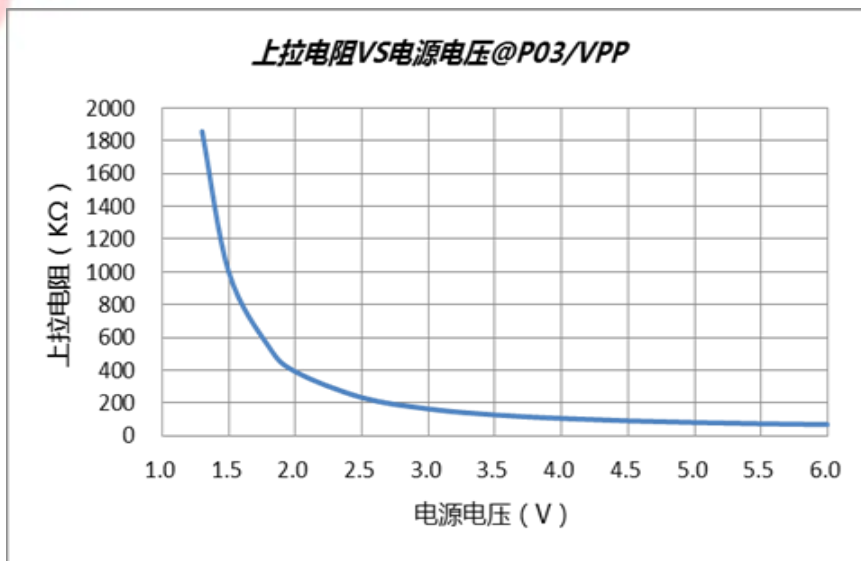
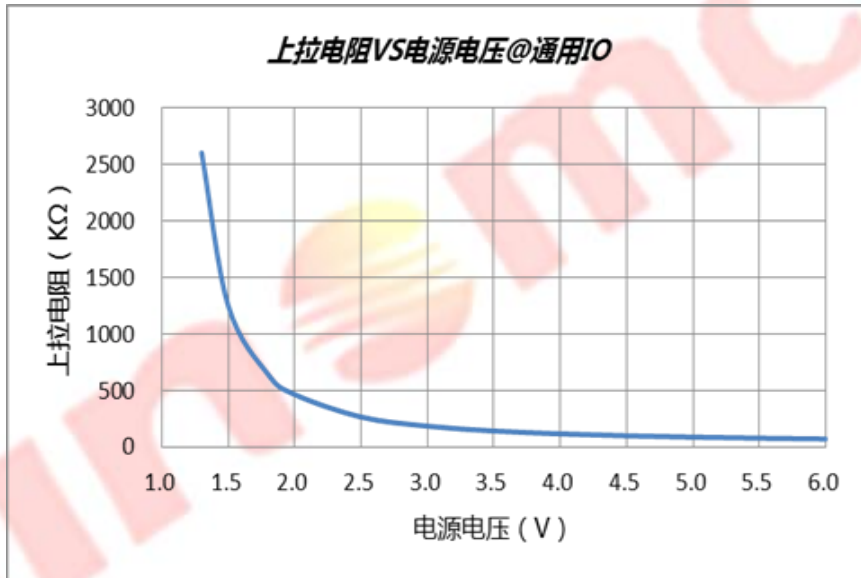
I/O 输出 驱动电流 VS 端口电压 (VDD=5V)





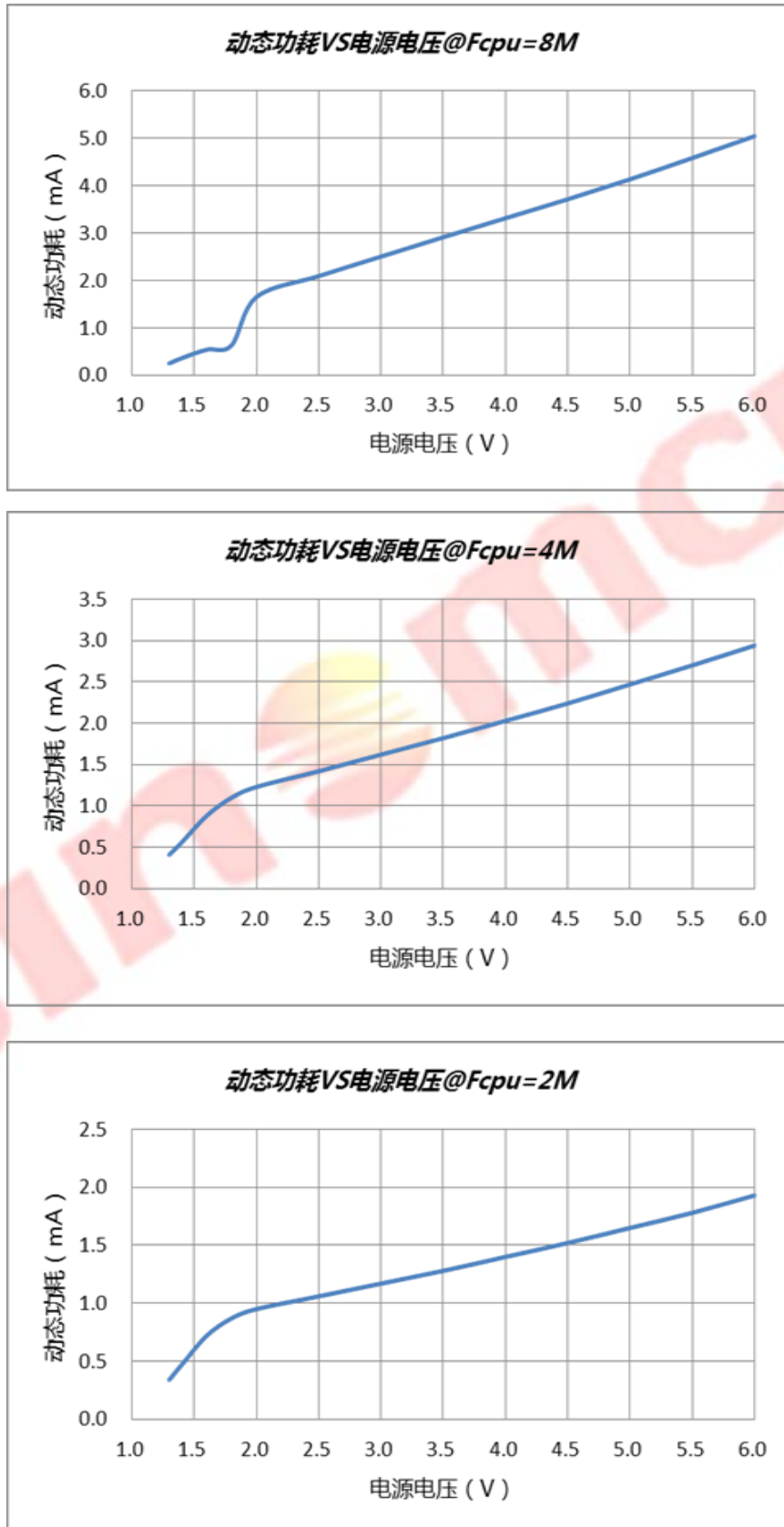


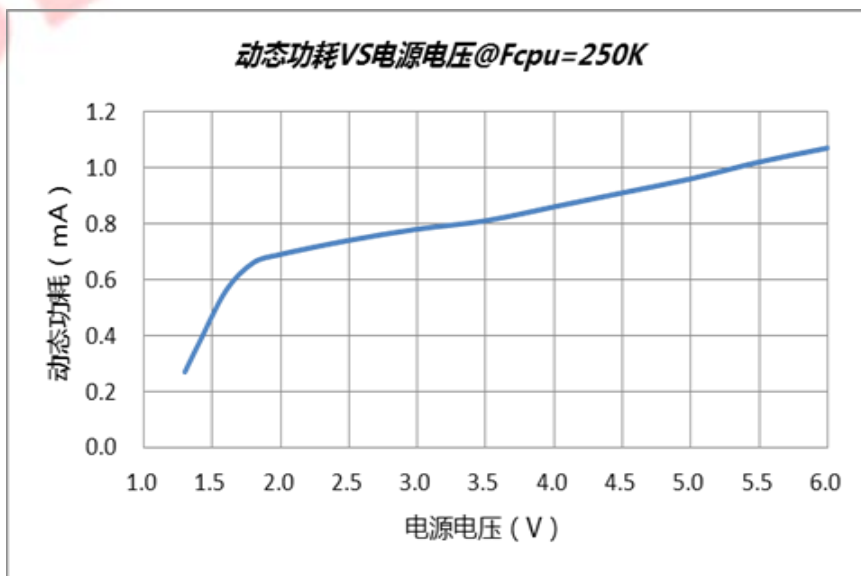
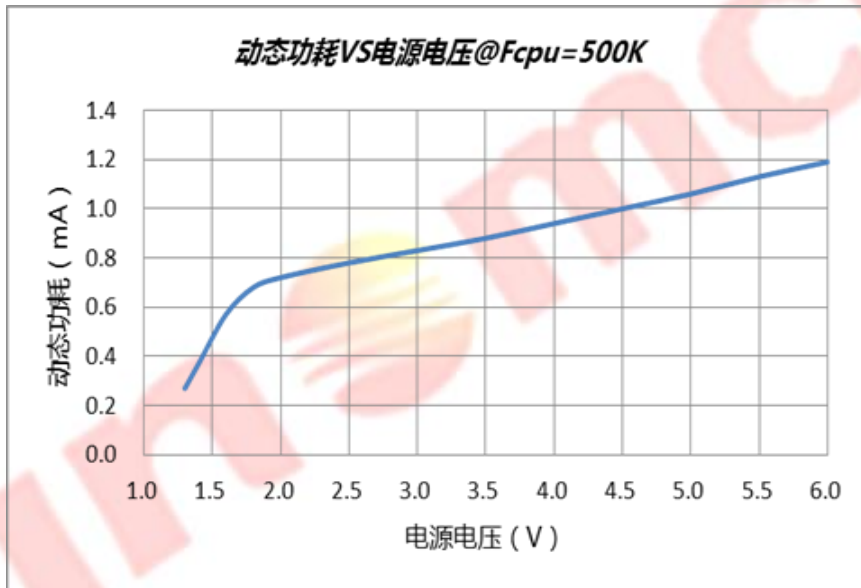
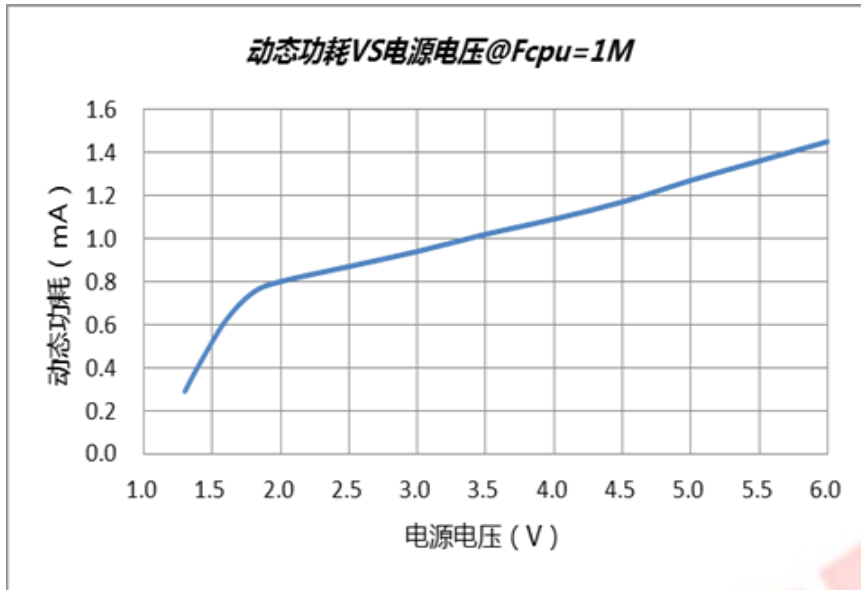
上拉电阻值 VS 电源电压

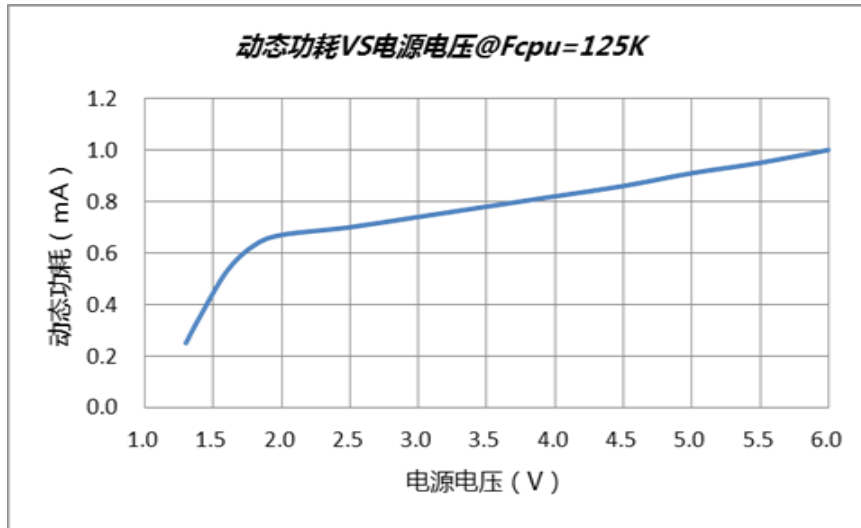


13.2 功耗特性

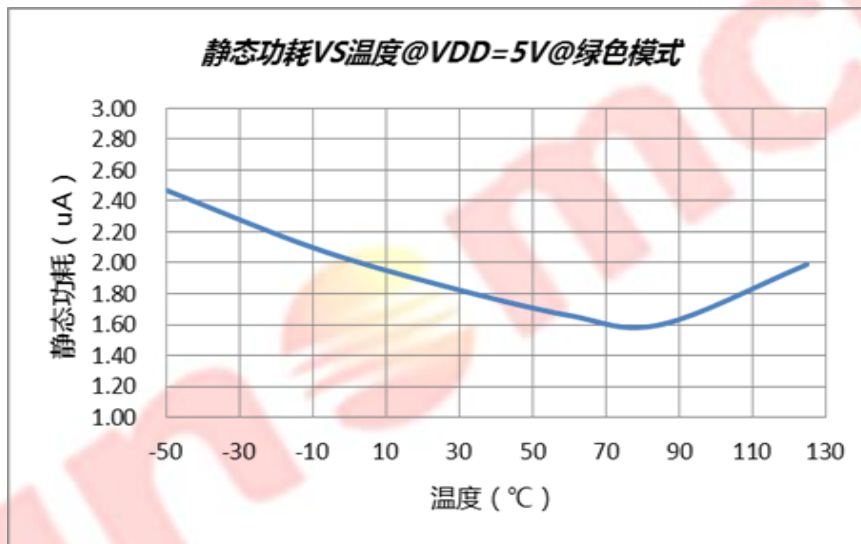
运行模式 功耗 VS 电源电压



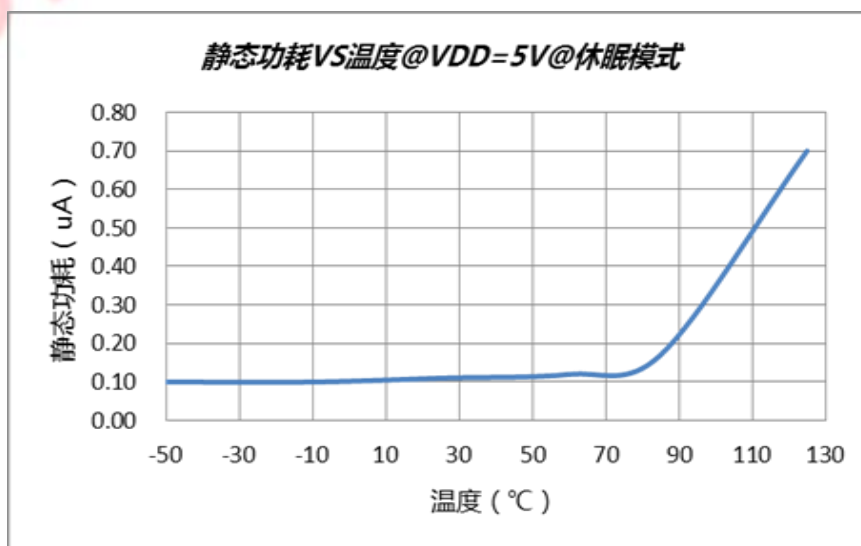




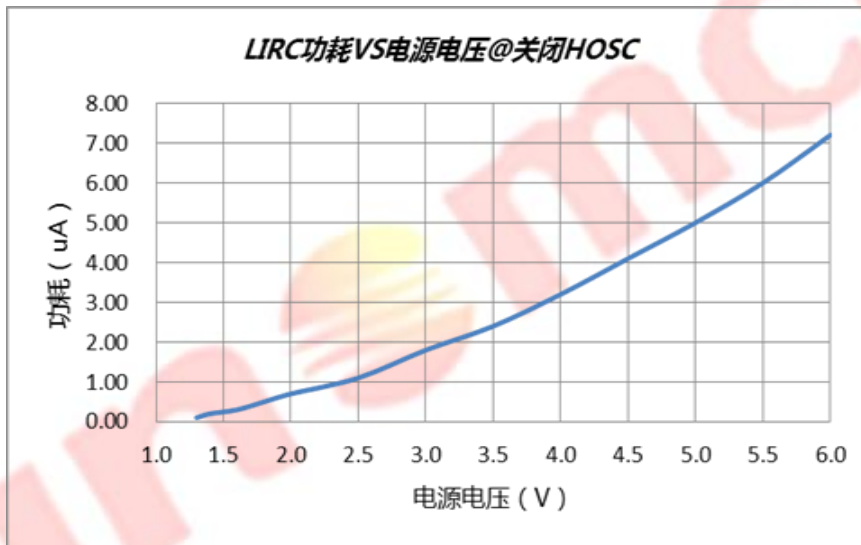
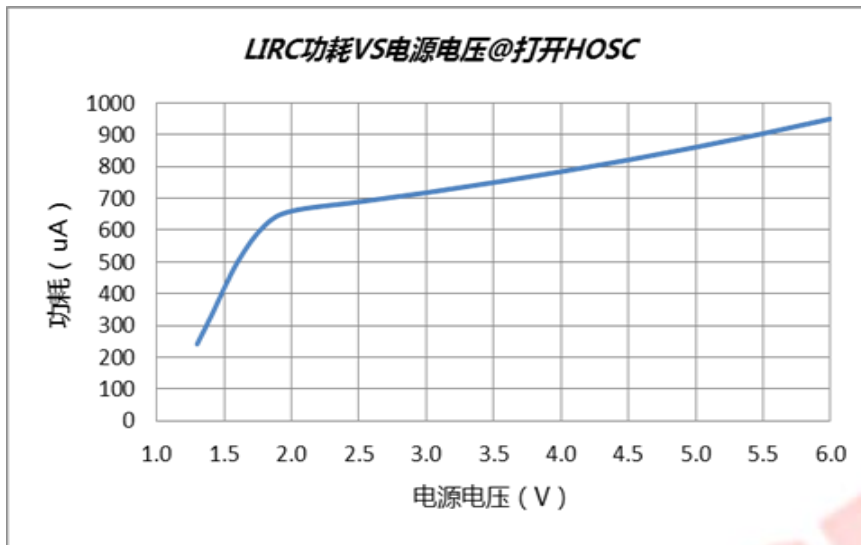
空闲模式 功耗 VS 温度



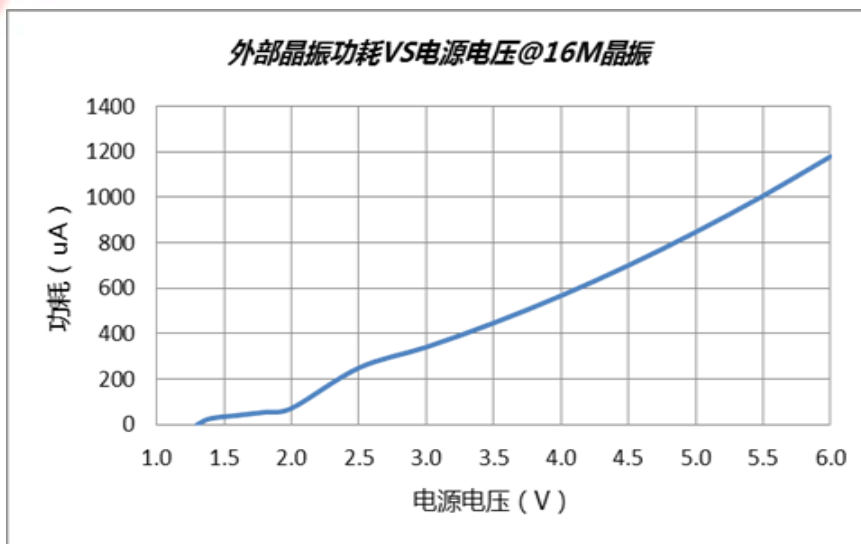
休眠模式 功耗 VS 温度

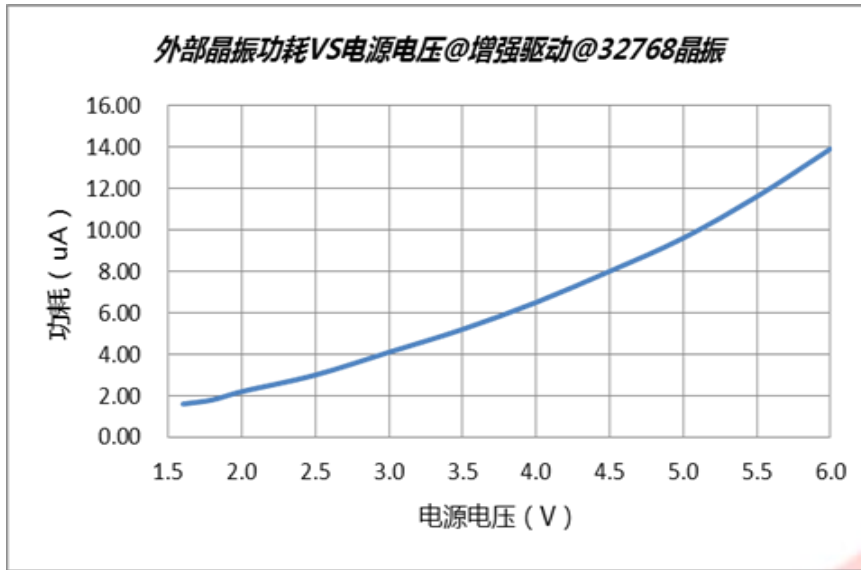


LIRC 功耗 VS 电源电压

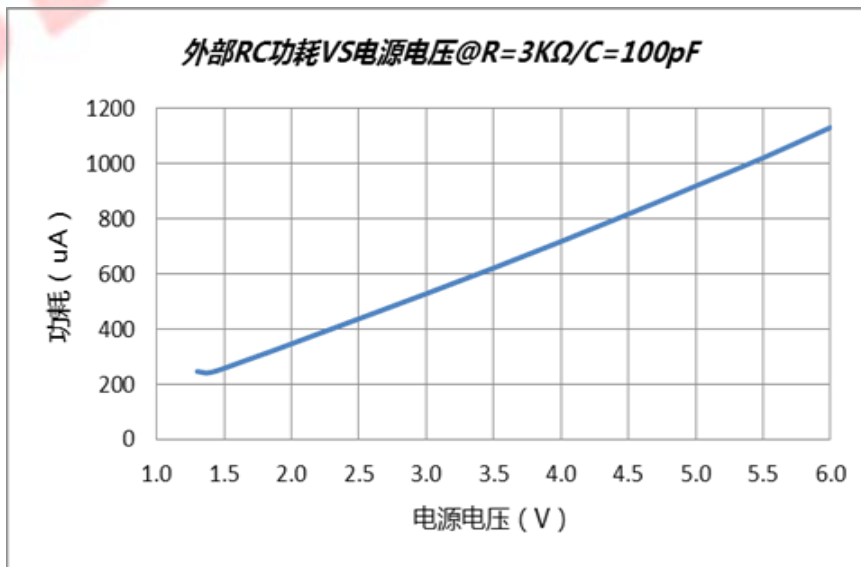
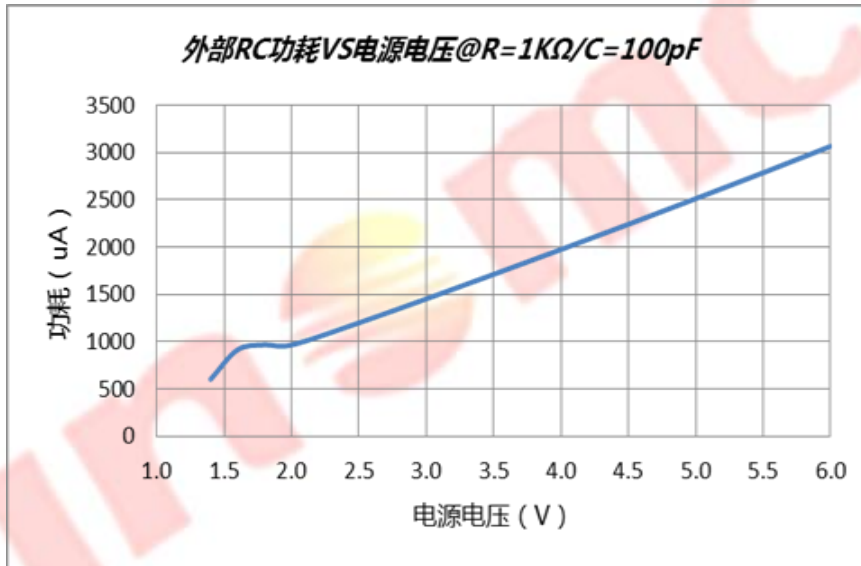


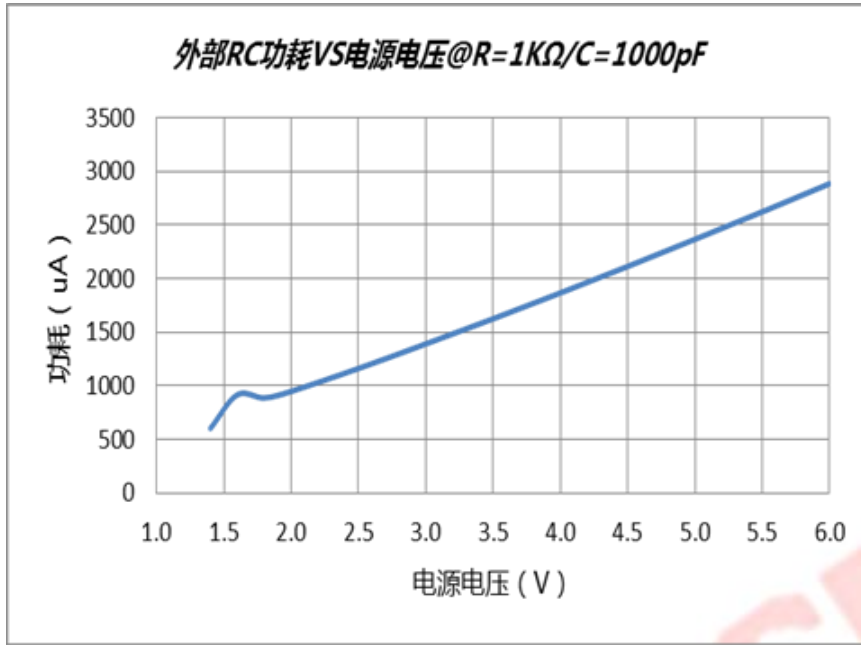
EXT 功耗 VS 电源电压



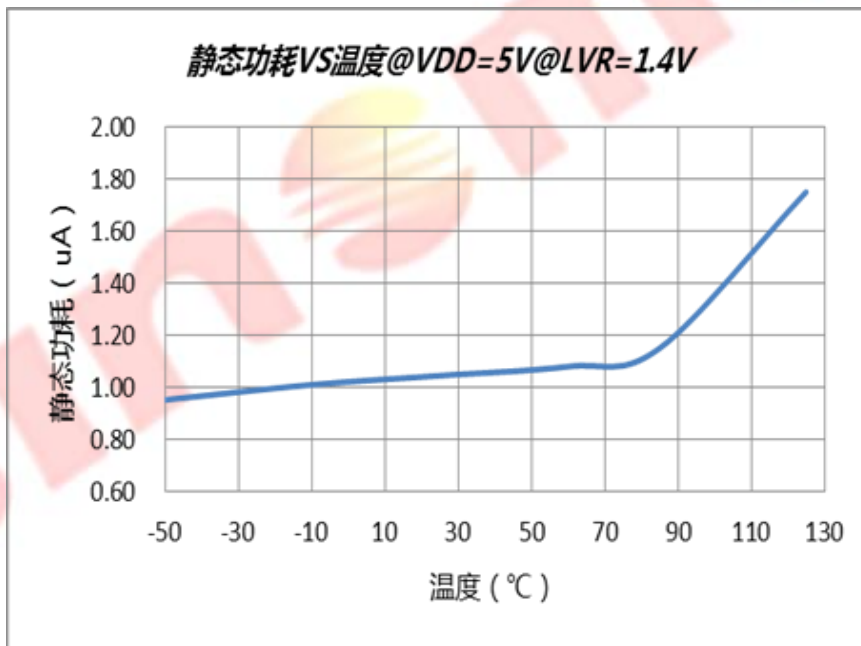


ERC 功耗 VS 电源电压



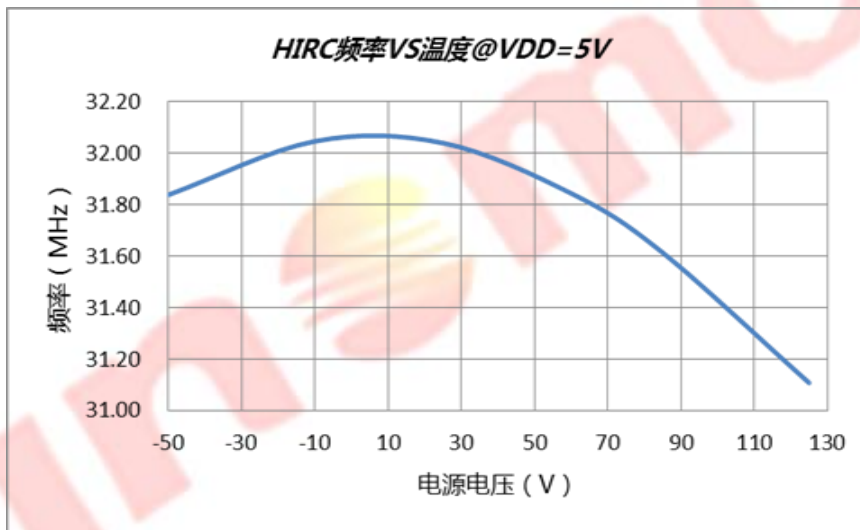
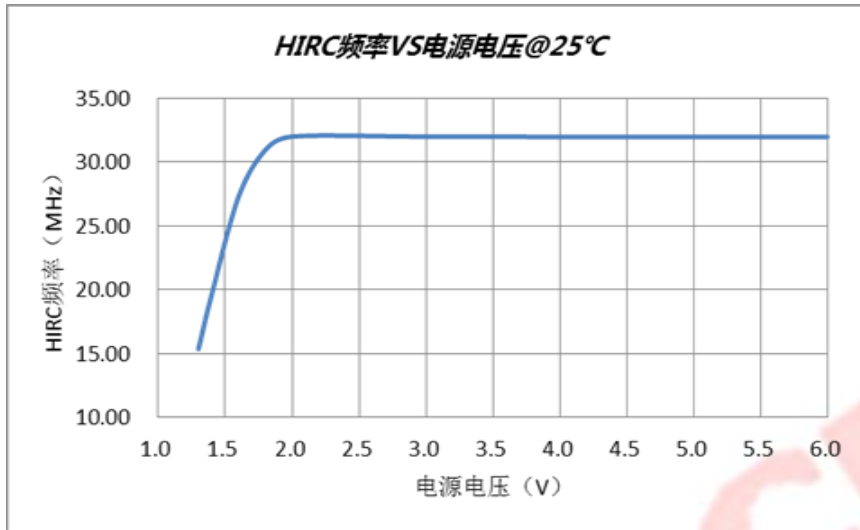


LVR 功耗 VS 温度

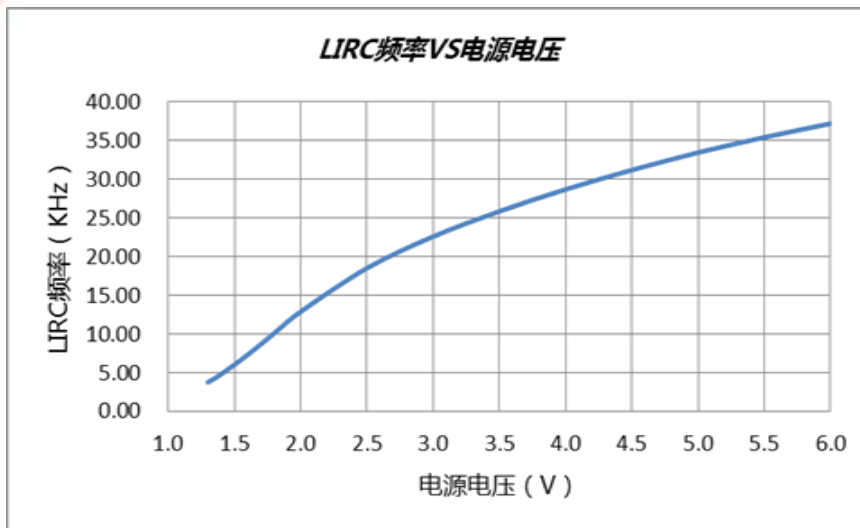


13.3 模拟电路特性

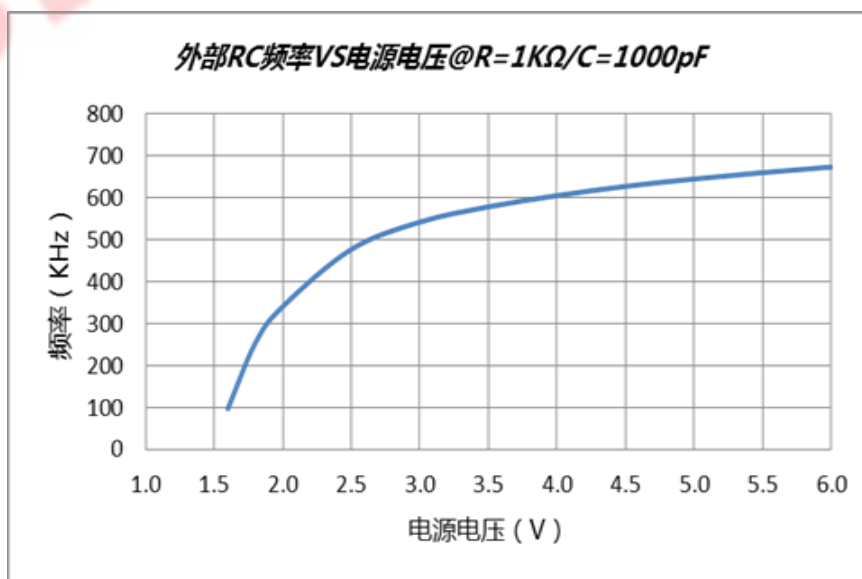
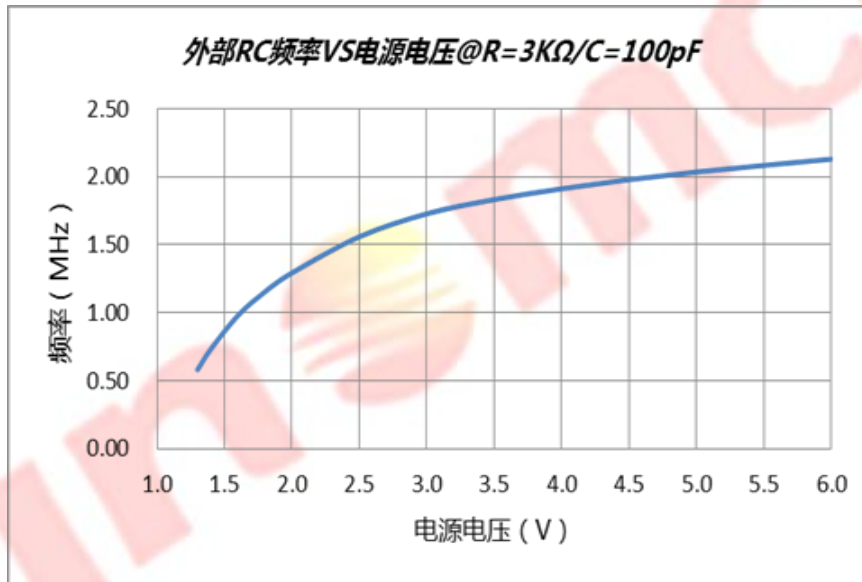
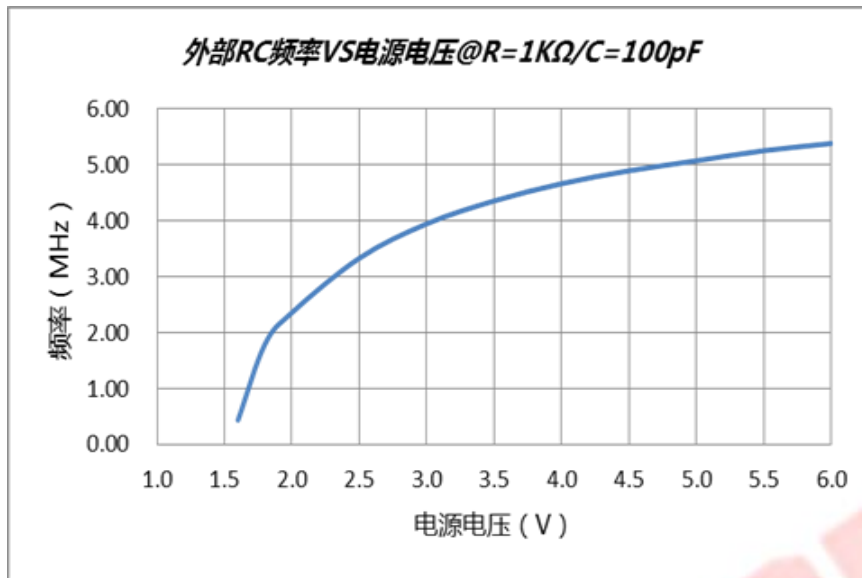
HIRC 频率 VS 电源电压/温度



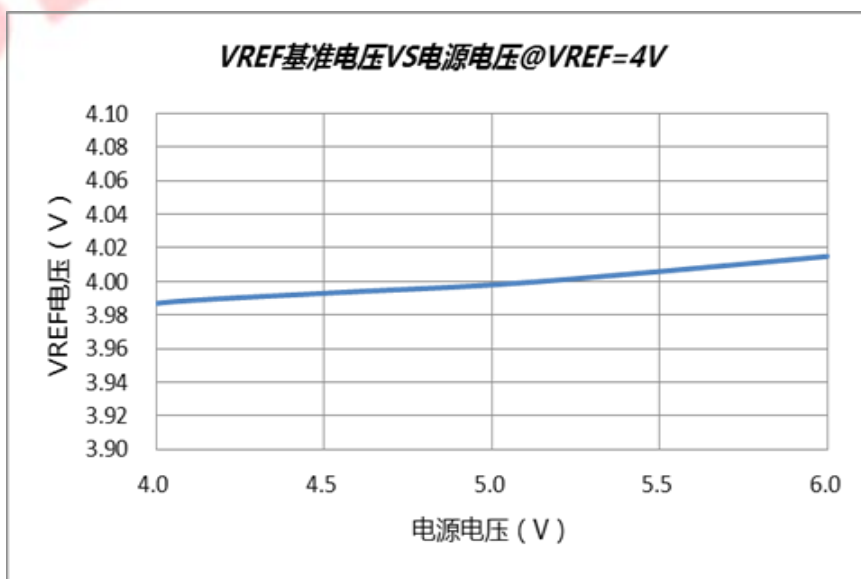
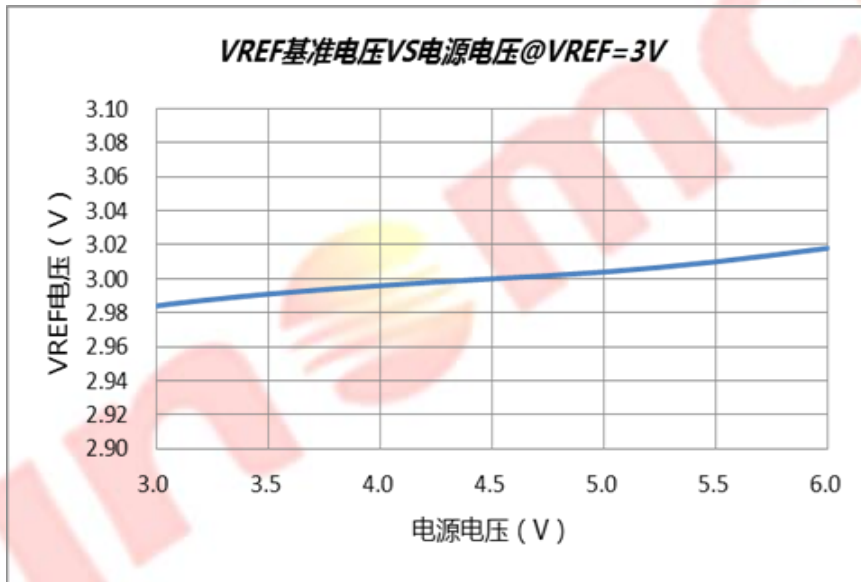
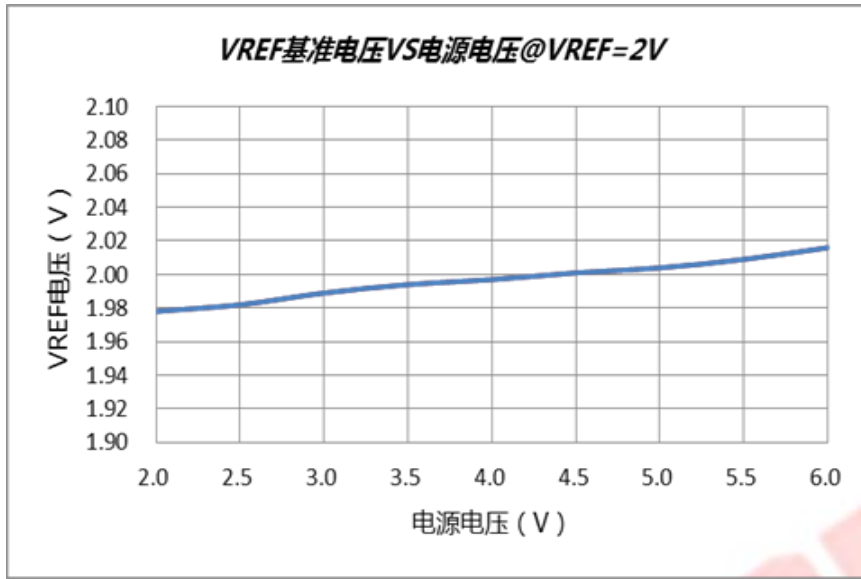
LIRC 频率 VS 电源电压

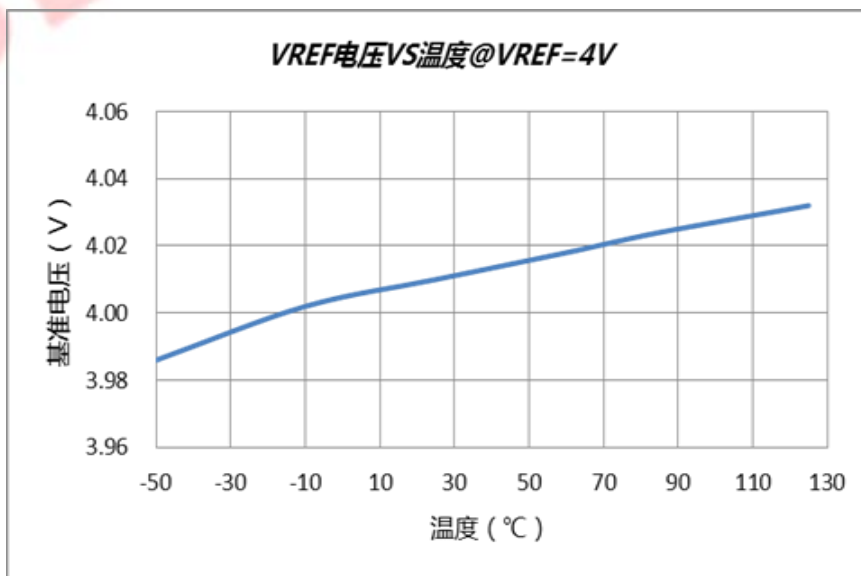
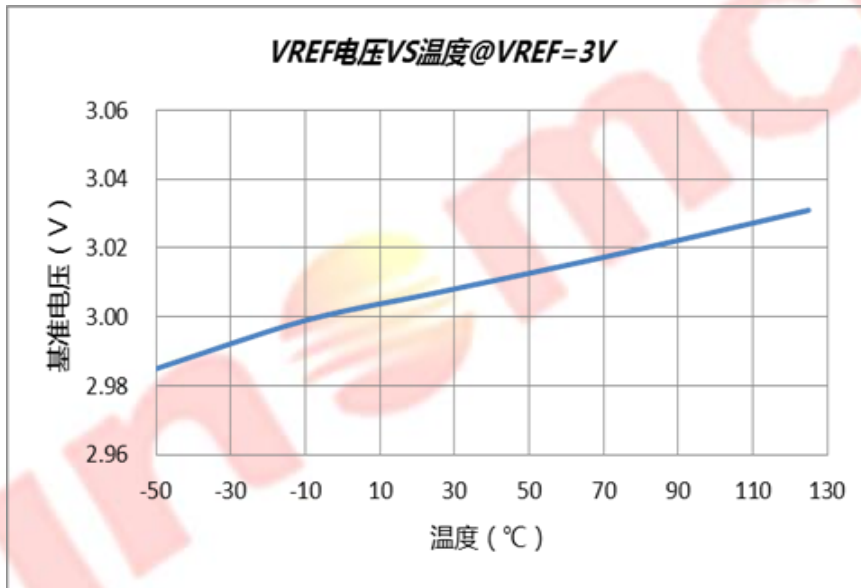
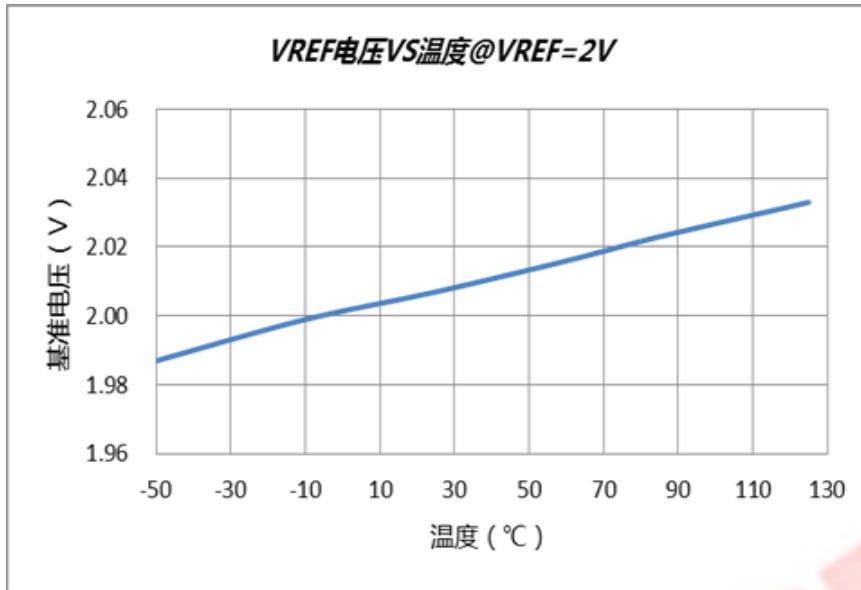


ERC 频率 VS 电源电压



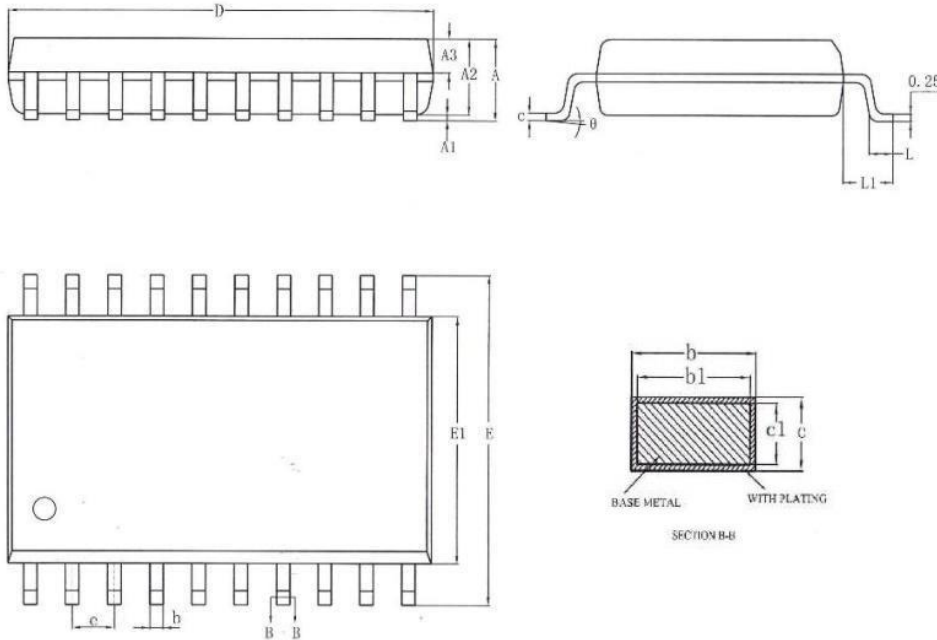
VIR 电压 VS 电源电压/温度





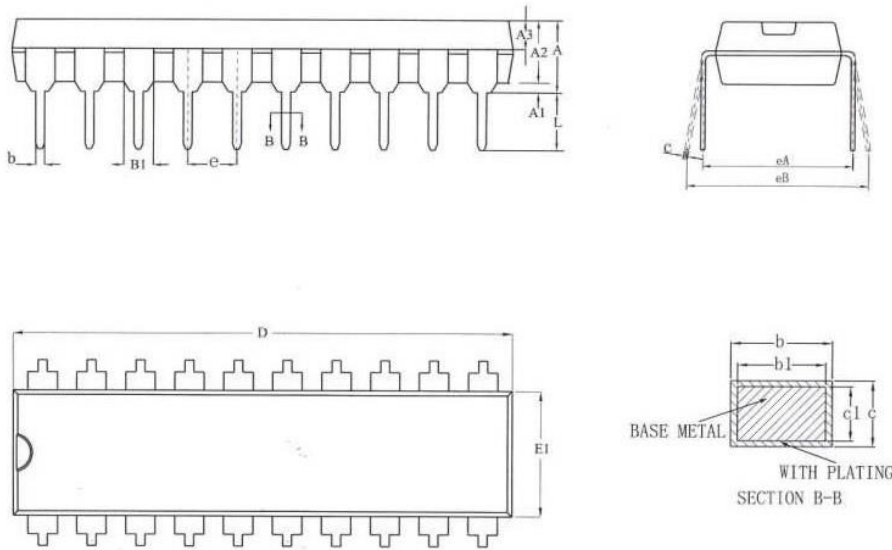
14 封装尺寸

14.1 SOP20



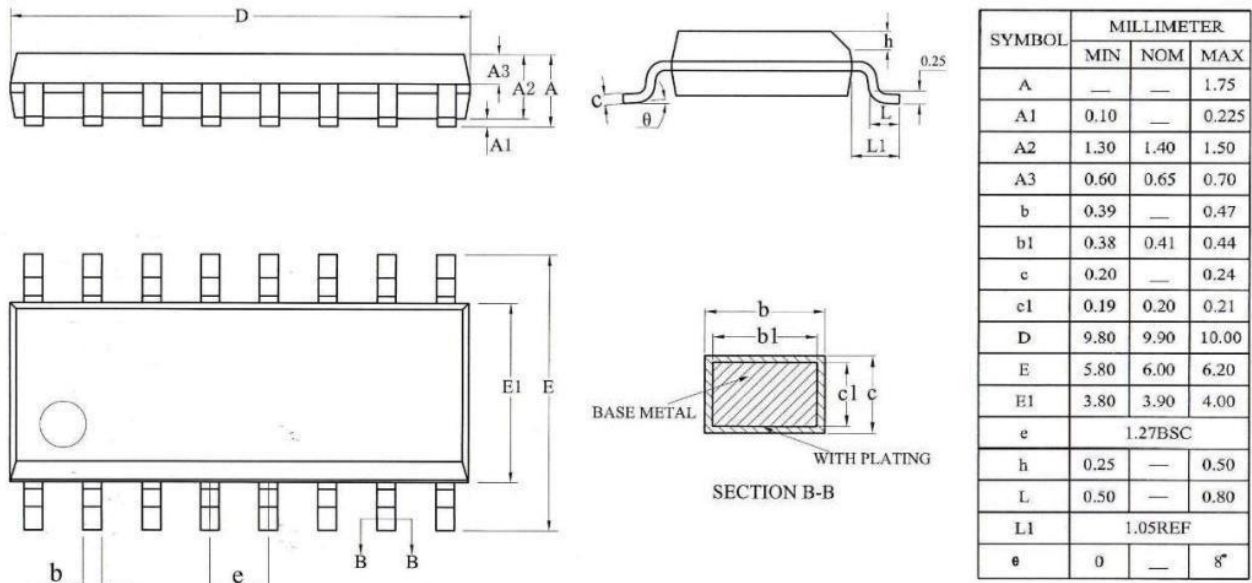
SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	—	—	2.65
A1	0.10	—	0.30
A2	2.25	2.30	2.35
A3	0.97	1.02	1.07
b	0.35	—	0.43
b1	0.34	0.37	0.40
c	0.25	—	0.29
c1	0.24	0.25	0.26
D	12.70	12.80	12.90
E	10.10	10.30	10.50
E1	7.40	7.50	7.60
e	1.27BSC		
L	0.70	—	1.00
L1	1.40REF		
θ	0	—	8°

14.2 DIP20

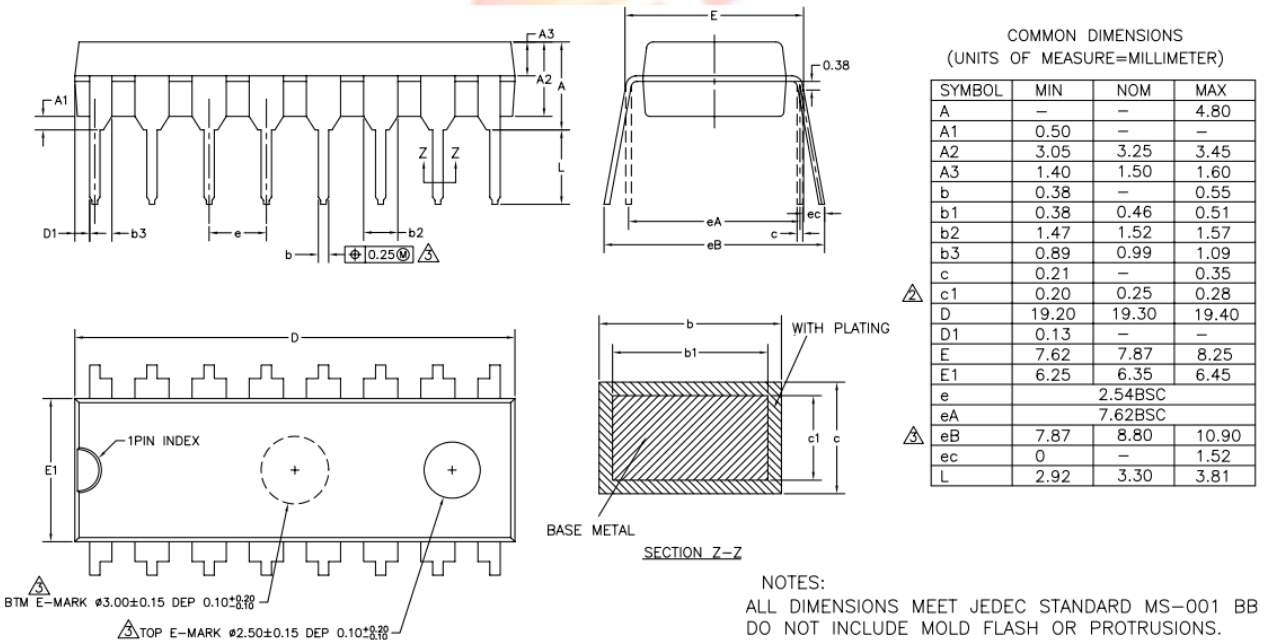


SYMBOL	MILLIMETER		
	MIN	NOM	MAX
A	3.60	3.80	4.00
A1	0.51	—	—
A2	3.20	3.30	3.40
A3	1.47	1.52	1.57
b	0.44	—	0.52
b1	0.43	0.46	0.49
B1	1.52REF		
c	0.25	—	0.29
c1	0.24	0.25	0.26
D	25.80	25.90	26.00
E1	6.45	6.55	6.65
e	2.54BSC		
eA	7.62REF		
eB	7.62	—	9.30
eC	0	—	0.84
L	3.00	—	—

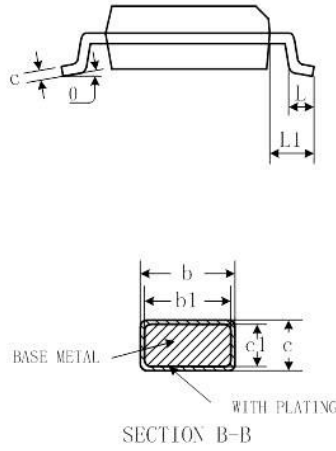
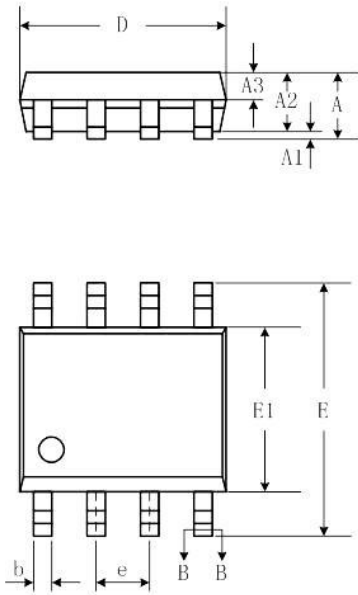
14.3 SOP16



14.4 DIP16

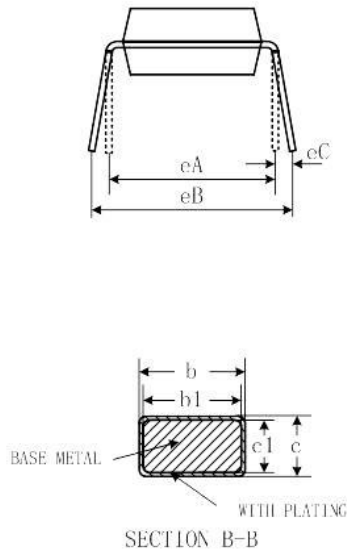
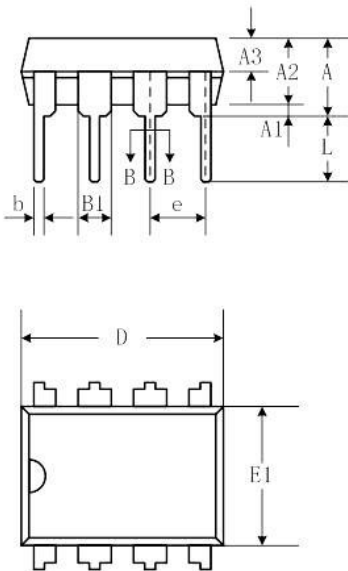


14.5 SOP8



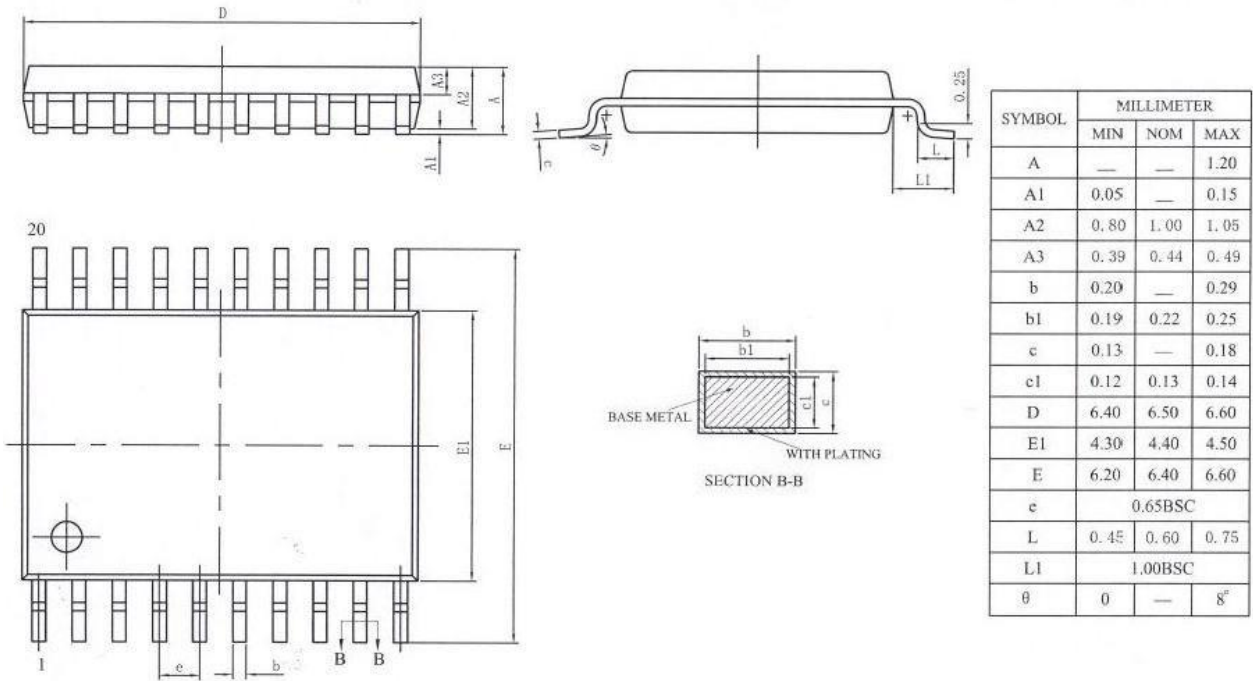
SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	-	-	1.77
A1	0.08	0.18	0.28
A2	1.20	1.40	1.60
A3	0.55	0.65	0.75
b	0.39	-	0.48
b1	0.38	0.41	0.43
c	0.21	-	0.26
c1	0.19	0.20	0.21
D	4.70	4.90	5.10
E	5.80	6.00	6.20
E1	3.70	3.90	4.10
e	1.27BSC		
L	0.50	0.65	0.80
L1	1.05BSC		
θ	0	-	8°

14.6 DIP8

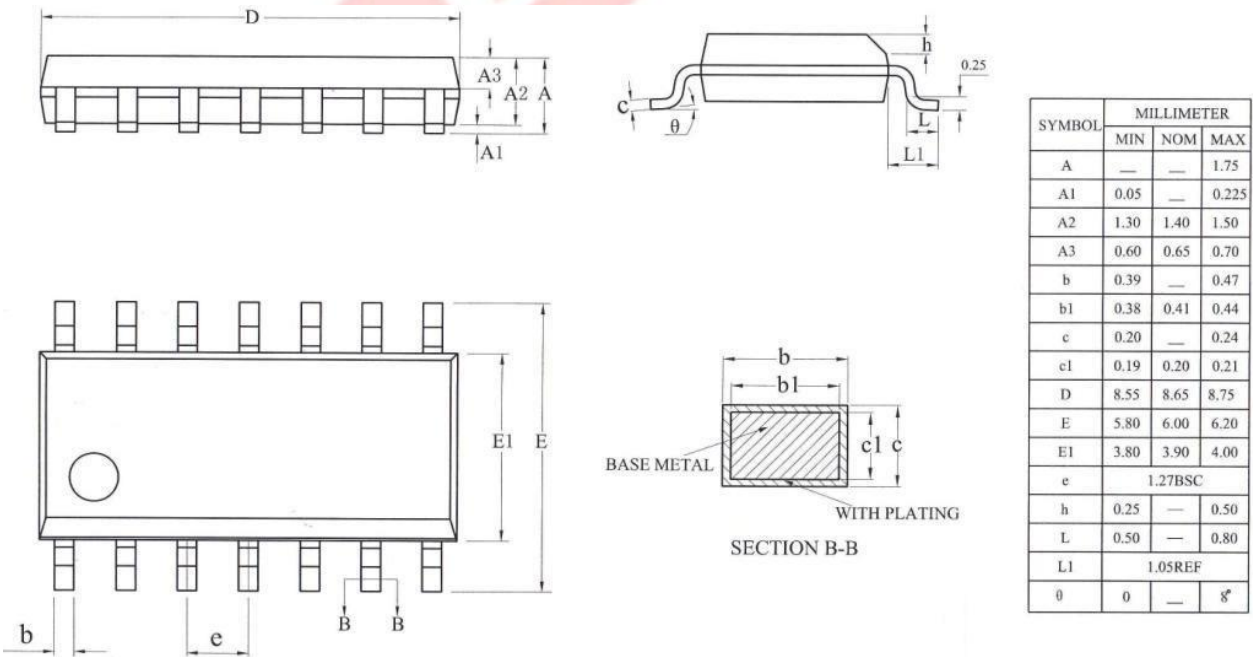


SYMBOL	MILLIMETER		
	MIN	TYP	MAX
A	3.60	3.80	4.00
A1	0.51	-	-
A2	3.10	3.30	3.50
A3	1.50	1.60	1.70
b	0.44	-	0.53
b1	0.43	0.46	0.48
B1	1.52BSC		
c	0.25	-	0.31
c1	0.24	0.25	0.26
D	9.05	9.25	9.45
E1	6.15	6.35	6.55
e	2.54BSC		
eA	7.62BSC		
eB	7.62	-	9.50
eC	0	-	0.94
L	3.00	-	-

14.7 TSSOP20



14.8 SOP14



15 修订记录

版本	修订日期	修订内容
V1.0	2017-07-28	初版发布;
V1.1	2019-03-25	新增 TSSOP20 封装 A0Y; 增加封装尺寸说明;
V1.2	2019-04-12	新增 DIP8 封装 A0A;
V1.3	2019-05-06	新增 LVR/LVD 关于回滞特性的说明及相关特性参数;
V1.4	2019-07-19	新增 SOP14 封装 A0J/A1J;
V1.5	2020-06-17	新增通过 CPUM 切换系统工作模式的说明及例程; 修订笔误; 新增低功耗模式未用引脚设置、复位时 VDD 上电慢的应对方法等相关的应用注释;
V1.6	2020-09-17	修正 SOP14 封装 A0J/A1J 端口名称;