



**REALMAGIC**  
锐盟半导体

# RM1211A 数据手册

18 路触按和 PWM 增强型 RISC MCU

V1.6  
2023-11-02



深圳锐盟半导体有限公司  
Realmagic Semiconductors (Shenzhen) Co.,Ltd.  
深圳市南山区高新北四道 13 号宇阳大厦 3 层  
[www.realmagicsemi.com](http://www.realmagicsemi.com)



## 目 录

1	产品简介.....	8
1.1	功能特性.....	8
1.2	系统框图.....	10
2	管脚配置.....	11
2.1	管脚排列图.....	11
2.2	脚位定义.....	13
3	CPU.....	17
3.1	CPU 核.....	17
3.2	程序存储器(CODE).....	17
3.3	数据存储器(SRAM).....	17
3.4	SFR.....	18
3.5	状态寄存器.....	27
3.5.1	状态寄存器 0(PFLAG0).....	27
3.5.1	状态寄存器 1(PFLAG1).....	28
3.6	间接寻址(IAR0 和 MP0).....	28
4	时钟系统.....	29
4.1	特性.....	29
4.1.1	系统时钟配置寄存器(OSCM).....	29
5	电源管理.....	31
5.1	IDLE 模式.....	31
5.2	STOP 模式.....	31
5.3	低电压检测.....	31
5.3.1	BOD 配置寄存器(BODCON).....	31
6	复位系统.....	33
7	中断系统.....	34
7.1	特性.....	34
7.2	寄存器.....	35
7.2.1	中断标志位寄存器 0(IF0).....	35
7.2.2	中断使能寄存器 0(IEN0).....	35
7.2.3	中断标志位寄存器 1(IF1).....	36
7.2.4	中断使能寄存器 1(IEN1).....	36
7.2.5	中断标志位寄存器 2(IF2).....	37
7.2.6	中断使能寄存器 2(IEN2).....	38
7.3	外部中断.....	38
7.3.1	外部中断触发方式选择控制寄存器 (EXINTS).....	39
7.4	其他中断.....	39

8	看门狗(WDT).....	40
8.1	特性.....	40
8.2	寄存器.....	40
8.2.1	WDT 配置寄存器(WDTM) .....	40
8.2.2	WDT 定时溢出寄存器(WDTR) .....	41
9	自唤醒定时器(WKT).....	43
9.1	寄存器.....	43
9.1.1	WKT 控制寄存器 (WKTCN) .....	43
9.1.2	WKT 重载数据寄存器 (RWK) .....	44
10	GPIO .....	45
10.1	特性.....	45
10.2	寄存器.....	45
10.2.1	PA/PB/PC 端口数据锁存寄存器 (PORTA / PORTB / PORTC) .....	45
10.2.2	PA/PB/PC 端口控制寄存器 (PAM/PBM/PCM) .....	45
10.2.3	PA/PB/PC 上拉控制寄存器 (PAUR/PBUR/PCUR) .....	46
10.2.4	PA/PB/PC 下拉控制寄存器 (PADR/PBDR/PCDR) .....	46
10.2.5	PA/PB/PC 开漏功能控制寄存器 (PAOD / PBOD / PCOD) .....	47
10.2.6	驱动电流配置寄存器 (PAIO) .....	47
10.2.1	驱动电流配置寄存器 (PPIO) .....	48
10.3	IO 翻转变醒 .....	49
10.3.1	GPIO 翻转变醒使能寄存器 (PWKENx) .....	49
11	Timer0.....	51
11.1	特性.....	51
11.2	寄存器.....	51
11.2.1	TIMER0 配置寄存器(T0M).....	51
11.2.2	TIMER0 计数寄存器(T0C).....	52
11.2.3	TIMER0 重载寄存器(T0R).....	53
11.3	定时器.....	53
11.4	计数器.....	53
11.5	RTC 模式.....	54
12	Timer1.....	55
12.1	特性.....	55
12.2	寄存器.....	55
12.2.1	TIMER1 配置寄存器 0(T1M0).....	55
12.2.2	TIMER1 配置寄存器 1(T1M1).....	56
12.2.3	TIMER1 重载寄存器高 8 位(T1RH).....	57
12.2.4	TIMER1 重载寄存器低 8 位(T1RL) .....	57
12.2.5	T1PWM0 占空比重载寄存器高 8 位(T1D0H) .....	57
12.2.6	T1PWM0 占空比重载寄存器低 8 位(T1D0L).....	57
12.2.7	T1PWM1 占空比重载寄存器高 8 位(T1D1H) .....	58
12.2.8	T1PWM1 占空比重载寄存器低 8 位(T1D1L).....	58

12.2.9	T1PWM0 死区配置寄存器高 8 位(T1PWM0DZH) .....	58
12.2.10	T1PWM0 死区配置寄存器低 8 位(T1PWM0DZL).....	59
12.2.11	T1PWM1 死区配置寄存器高 8 位(T1PWM1DZH) .....	59
12.2.12	T1PWM1 死区配置寄存器低 8 位(T1PWM1DZL).....	59
12.2.13	TIMER1 配置寄存器 2(T1M2).....	60
12.2.14	T1PWM2 占空比重载寄存器高 8 位(T1D2H) .....	60
12.2.15	T1PWM2 占空比重载寄存器低 8 位(T1D2L).....	61
12.2.16	T1PWM3 占空比重载寄存器高 8 位(T1D3H) .....	61
12.2.17	T1PWM3 占空比重载寄存器低 8 位(T1D3L).....	61
12.3	定时器.....	62
12.4	计数器.....	62
12.5	PWM.....	62
12.5.1	PWM 输出.....	63
12.5.2	PWM 互补输出与死区.....	64
<b>13</b>	<b>Timer2.....</b>	<b>67</b>
13.1	特性.....	67
13.2	工作原理.....	67
13.3	寄存器.....	68
13.3.1	TIMER2 配置寄存器 0(T2M0).....	68
13.3.2	TIMER2 配置寄存器 1(T2M1).....	68
13.3.3	T2PWM0 占空比重载寄存器高 8 位(T2D0H) .....	69
13.3.4	T2PWM0 占空比重载寄存器低 8 位(T2D0L).....	69
13.3.5	T1PWM2 占空比重载寄存器高 8 位(T2D1H).....	70
13.3.6	T2PWM1 占空比重载寄存器低 8 位(T2D1L).....	70
13.3.7	TIMER2 计数寄存器高 8 位(T2CH).....	70
13.3.8	TIMER2 计数寄存器低 8 位(T2CL) .....	70
13.3.9	TIMER2 重载寄存器高 8 位(T2RH).....	71
13.3.10	TIMER2 重载寄存器低 8 位(T2RL) .....	71
13.3.11	TIMER2 捕获中断寄存器 (T2CAPINT) .....	71
13.3.12	TIMER2 捕获控制寄存器 (T2CAPCON) .....	72
13.3.13	TIMER2 捕获个数配置寄存器 (T2CAPNUM).....	72
13.3.14	TIMER2 捕获输入滤波时间配置寄存器 (T2CAPNFCNT).....	73
13.4	定时器.....	74
13.5	PWM 输出.....	74
13.5.1	T2PWM 输出配置步骤.....	74
13.6	捕获功能.....	75
13.6.1	软件配置步骤.....	76
13.6.2	脉冲周期测量.....	77
13.6.3	电平宽度测量.....	77
<b>14</b>	<b>Timer3.....</b>	<b>78</b>
14.1	特性.....	78
14.2	寄存器.....	78
14.2.1	TIMER3 配置寄存器 (T3M).....	78

14.2.2	TIMER3 重载寄存器高 8 位(T3RH).....	79
14.2.3	TIMER3 重载寄存器低 8 位(T3RL).....	79
14.3	定时器.....	79
<b>15</b>	<b>ADC .....</b>	<b>80</b>
15.1	概述.....	80
15.2	功能描述.....	81
15.2.1	ADC 工作方式 .....	81
15.2.2	ADC 参考带隙电压应用.....	82
15.3	寄存器.....	82
15.3.1	ADC 控制寄存器 0 (ADCCON0) .....	82
15.3.2	ADC 控制寄存器 1 (ADCCON1) .....	83
15.3.3	ADC 控制寄存器 2 (ADCCON2) .....	83
15.3.4	ADC 控制寄存器 3 (ADCCON3) .....	84
15.3.5	ADC 控制寄存器 4 (ADCCON4) .....	85
15.3.6	ADC 转换结果低位寄存器 (ADCRL) .....	86
15.3.7	ADC 转换结果高位寄存器 (ADCRH) .....	86
15.3.8	ADC 模拟输入通道 (PA) 功能寄存器 (PACHS) .....	86
15.3.9	ADC 模拟输入通道 (PB) 功能寄存器 (PBCHS) .....	87
15.3.10	ADC 模拟输入通道 (PC) 功能寄存器 (PCCHS) .....	87
15.4	ADC 应用注意事项 .....	88
<b>16</b>	<b>比较器/运算放大器 .....</b>	<b>89</b>
16.1	特性.....	89
16.2	寄存器.....	89
16.2.1	比较器 0 控制寄存器 0 (CMP0CON0) .....	89
16.2.2	比较器 0 控制寄存器 1 (CMP0CON1) .....	90
16.2.3	比较器 0 控制寄存器 2 (CMP0CON2) .....	90
16.2.4	比较器 0 控制寄存器 3 (CMP0CON3) .....	91
16.2.5	比较器 1 控制寄存器 0 (CMP1CON0) .....	92
16.2.6	比较器 1 控制寄存器 1 (CMP1CON1) .....	93
16.2.7	比较器 1 控制寄存器 2 (CMP1CON2) .....	93
16.2.8	比较器 1 控制寄存器 3 (CMP1CON3) .....	94
16.2.9	比较器 2 控制寄存器 0 (CMP2CON0) .....	95
16.2.10	比较器 2 控制寄存器 1 (CMP2CON1) .....	95
16.2.11	比较器 2 控制寄存器 2 (CMP2CON2) .....	96
16.2.12	比较器 2 控制寄存器 3 (CMP2CON3) .....	97
16.2.13	比较器 3 控制寄存器 0 (CMP3CON0) .....	97
16.2.14	比较器 3 控制寄存器 1 (CMP3CON1) .....	98
16.2.15	比较器 3 控制寄存器 2 (CMP3CON2) .....	99
16.2.16	比较器 3 控制寄存器 3 (CMP3CON3) .....	100
16.2.17	防水功能控制寄存器 (SHDCON) .....	100
16.3	运算放大器的使用.....	101
16.3.1	OPA 使能 .....	101
16.3.2	OPA 端口选择.....	101

16.3.3	OPA 工作模式.....	101
16.4	比较器的使用.....	102
<b>17</b>	<b>UART.....</b>	<b>103</b>
17.1	概述.....	103
17.2	功能描述.....	103
17.2.1	模式 1 .....	103
17.2.2	模式 2 .....	104
17.2.3	波特率 .....	105
17.3	寄存器描述.....	105
17.3.1	串口控制寄存器 (SCON) .....	105
17.3.2	串口数据缓存寄存器 (SBUF) .....	106
17.3.3	串口波特率低字节 (SOVRL) .....	106
17.3.4	串口波特率高字节 (SOVRH) .....	106
<b>18</b>	<b>触摸按键(TOUCH KEY).....</b>	<b>107</b>
18.1	概述.....	107
<b>19</b>	<b>MTP 和 EEPROM 的自编程 .....</b>	<b>108</b>
19.1	特性.....	108
19.2	寄存器.....	108
19.2.1	IAP 执行寄存器 (IAPTRIG) .....	108
19.2.2	IAP 控制寄存器 (IAPCON) .....	109
19.2.3	IAP 地址高字节 (IAPADRH) .....	110
19.2.4	IAP 地址低字节 (IAPADRL) .....	110
19.2.5	IAP 数据高字节 (IAPDATH) .....	110
19.2.6	IAP 数据低字节 (IAPDATL) .....	110
19.2.7	E2PROM 控制寄存器 (E2CON) .....	111
<b>20</b>	<b>时钟 Trimming .....</b>	<b>113</b>
20.1	ICK.....	113
20.1.1	ICK 配置寄存器(ICK_TRIM) .....	113
20.2	WCK .....	113
20.2.1	WCK 配置寄存器(WCK_TRIM).....	113
<b>21</b>	<b>LCD.....</b>	<b>114</b>
21.1	概述.....	114
21.2	LCD 使用说明.....	114
21.3	寄存器.....	117
21.3.1	LCD 模块控制寄存器 (LCDCON) .....	117
21.3.2	PA 口输出选择寄存器 (PAVO) .....	117
21.3.3	PB 口输出选择寄存器 (PBVO) .....	118
<b>22</b>	<b>1 DAC .....</b>	<b>118</b>
22.1	概述.....	118
22.2	寄存器.....	118
22.2.1	DAC0 控制寄存器 0 (DAC0CON0) .....	118

22.2.2	DAC0 控制寄存器 1 (DAC0CON1) .....	119
22.2.3	DAC1 控制寄存器 0 (DAC1CON0) .....	119
22.2.4	DAC1 控制寄存器 1 (DAC1CON1) .....	120
22.3	DAC 的使用 .....	120
22.3.1	DAC 使能 .....	120
22.3.2	DAC 参考电压来源选择 .....	120
22.3.3	DAC 输出 .....	120
23	烧录 .....	121
24	指令集 .....	122
25	电气特性 .....	124
25.1	极限值 .....	124
25.2	直流特性 .....	124
25.3	内部 16 MHz 高速时钟(HIRC) .....	125
25.4	内部 128 KHz 低速时钟(LIRC) .....	126
25.5	低压复位规格 .....	126
25.6	欠压检测规格 .....	126
25.7	ADC 电气特性 .....	127
25.8	OPA 电气特性 .....	127
25.9	交流电气特性 .....	128
26	封装 .....	129
26.1	SOP20 封装尺寸与外形图 .....	129
26.2	TSSOP20 封装尺寸与外形图 .....	130
26.3	SOP16 封装尺寸与外形图 .....	130
26.4	TSSOP16 封装尺寸图 .....	131
26.5	QFN16 3mm*3mm 封装尺寸图 .....	132
26.6	QFN20 3mm*3mm 封装尺寸图 .....	133
26.7	订购信息 .....	133

## 1 产品简介

### 1.1 功能特性

- ◆ 高性能的精简指令集控制处理器
  - 基于 1T RISC CPU 内核
  - CPU 运行速度最高可达 16MHz
- ◆ 存储器
  - ROM (MTP) : 4K\*16
  - SRAM: 384 \*8
  - EEPROM: 128\*8
- ◆ 时钟系统
  - 内部高速时钟: 16MHz RC
  - 内部低速时钟: 128KHz RC
  - 外部高速晶体振荡器: 16MHz
  - 外部低速晶体振荡器: 32KHz RC
- ◆ 工作模式
  - 普通模式: 高低速时钟同时工作
  - 低速模式: 仅低速时钟工作
  - 休眠模式: 高低速时钟都停止工作
  - 绿色模式: 由定时器周期性唤醒
- ◆ 低电压复位 (BOR)
  - 复位电压有 (1.8~4.0V), 共 8 档可选
  - 缺省值为用户烧写 Code Option 所选值
- ◆ 16 档电平可选的低电压检测模块 (BOD)
- ◆ 中断源 (INT)
  - 19 个内部中断: T0、T1、T2、T3、T1PWM 占空比中断、T2PWM 占空比中断、T2CAP 中断、UART、WKT、ADC、LVD、CMP
  - 2 个外部中断: INT0、INT1
  - 1 个中断入口
- ◆ GPIO
  - 最大 18 个双向可独立控制的 I/O 口, 可独立设定上下拉电阻, 可设置为开漏输出
  - IO 口源驱动能力分两级控制
  - 全部 IO 具有大灌电流驱动能力 (60mA)
  - 所有 IO 可作为 1/2 BIAS 的 LCD COM 输出
- ◆ Timer
  - T0: 基本 8 位定时器/计数器, 支持预分频功能。
  - T1: 16 位可配置定时器/计数器(最高可跑 16MHz)
    - 4 组 (8 通道) PWM, 均支持互补输出和死区可配支持预分频功能



## 计数器寄存器可配置

- T2: 自动装载 16 位定时器, 2 通道 PWM 输出 (无互补功能) 或 2 通道捕获输入功能
- T3: 16 位可配置定时器, 支持预分频功能

## ◆ WDT

- 支持预分频功能
- 可配置溢出复位或唤醒

## ◆ WKT

- 支持预分频功能
- 在休眠(STOP)模式下或者绿色(IDLE)模式下, 均可唤醒 CPU

## ◆ UART

- 1 个独立 UART 通信口

## ◆ TK

- 18 通道高灵敏度高可靠性触控电路, 无需外挂电容
- 高灵敏度可适应隔空按键触控、接近感应等对灵敏度要求较高的触控应用
- 高可靠性具有很强的抗干扰性, 可通过 10V 动态 CS 测试
- 可实现 18 路触控按键及衍生功能如滚轮、滑条等多种应用
- 高灵活度开发软件库支持, 低开发难度
- 自动化调试软件支持, 智能化开发
- 低功耗触控模式, 单触控按键唤醒时芯片整体功耗可低至 11uA
- 休眠(STOP)模式下支持最多 4 键硬件自动扫描, 可唤醒芯片
- 内置 Shield Driver

## ◆ ADC

- 18 路 12 位 SAR ADC
- 内建基准的 1.024V、2.048V、3.072V、4.096V 参考电压
- ADC 的参考电压可选 VDD、内部基准、外部参考
- 内部一路 ADC 可直接测量 VDD/4 电压
- 内部一路 ADC 可直接测量 IO/4
- 内部一路 GND 电压检测
- 内部一路内部参考电压检测
- 内部两路运放输出检测
- 可设 ADC 转换完成中断

## ◆ 4 路运放/比较器

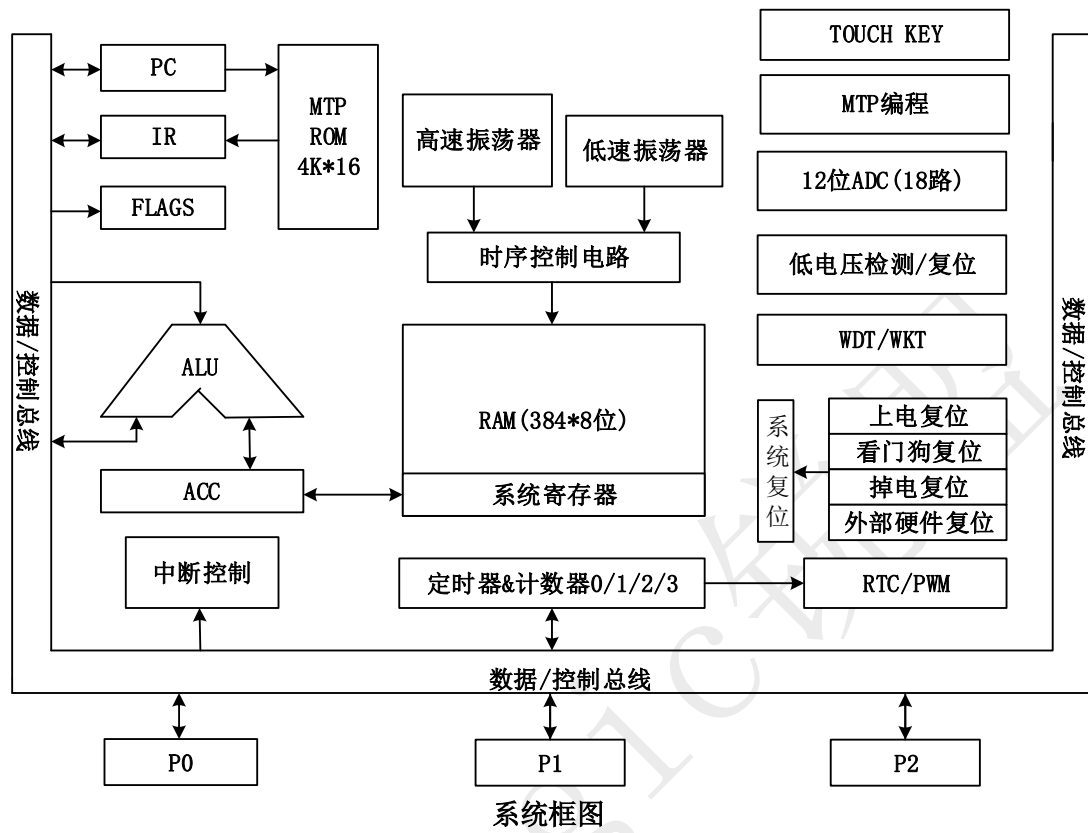
- 支持内部 PGA 16 倍放大
- 支持运放输出接 ADC

## ◆ 低功耗特性

- 休眠待机电流: 电压为 2.5V 时, 典型值 500nA
- 看门狗定时器电流: 电压为 2.5V 时, 典型值为 1uA

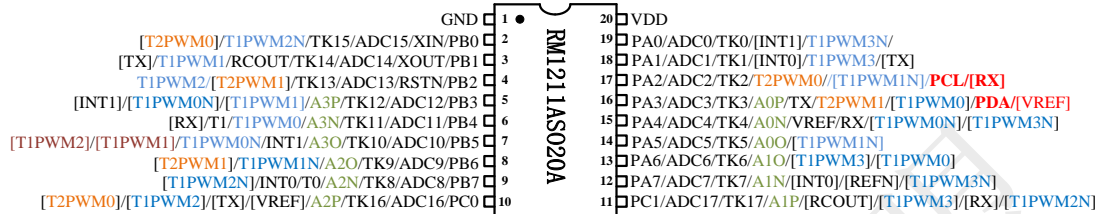
## ◆ 封装: SOP16/SOP20/TSSOP20/QFN20

### 1.2 系统框图

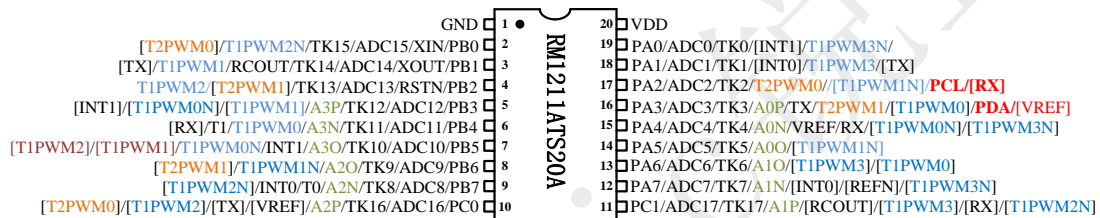


## 2 管脚配置

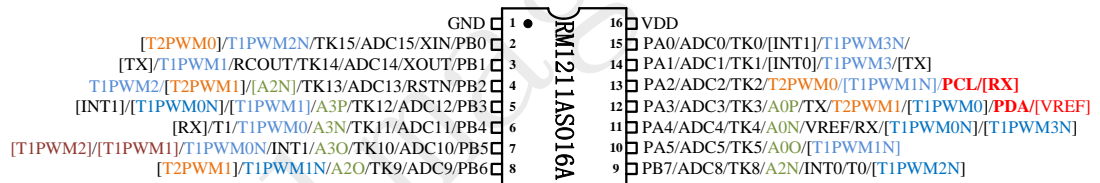
### 2.1 管脚排列图



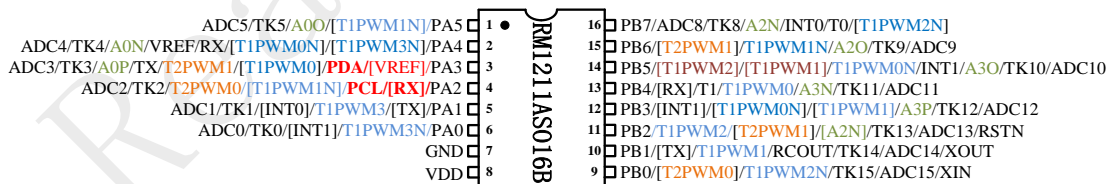
RM1211ASO20A (SOP20)



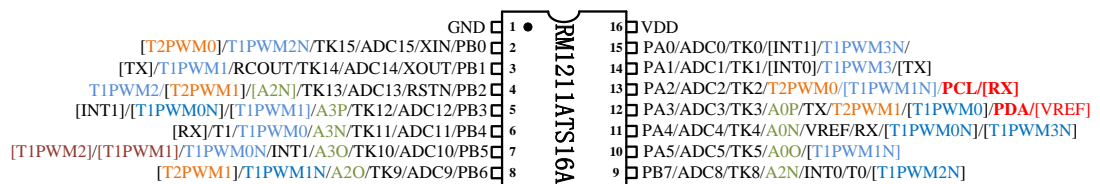
RM1211ATS20A (TSSOP20)



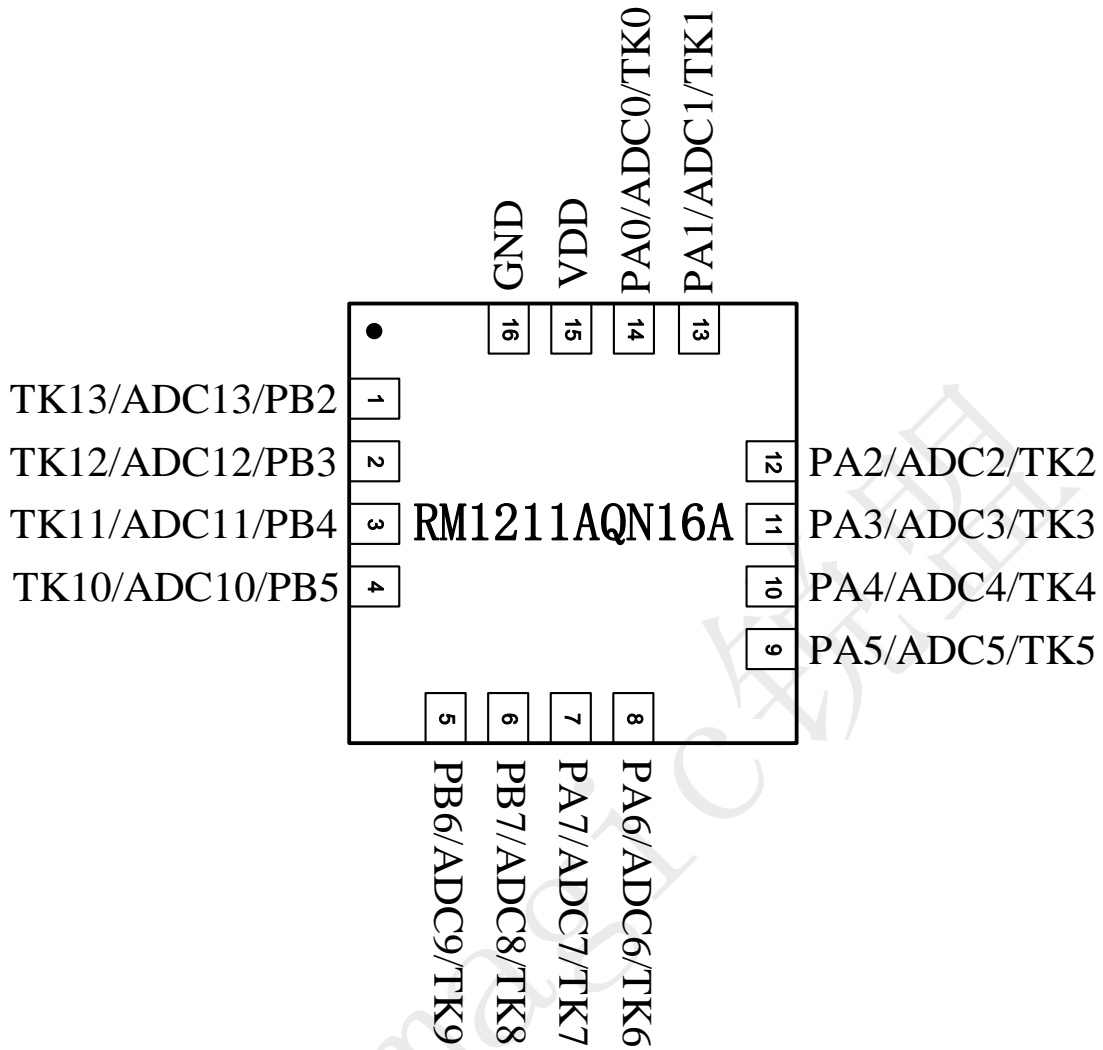
RM1211ASO16A (SOP16)



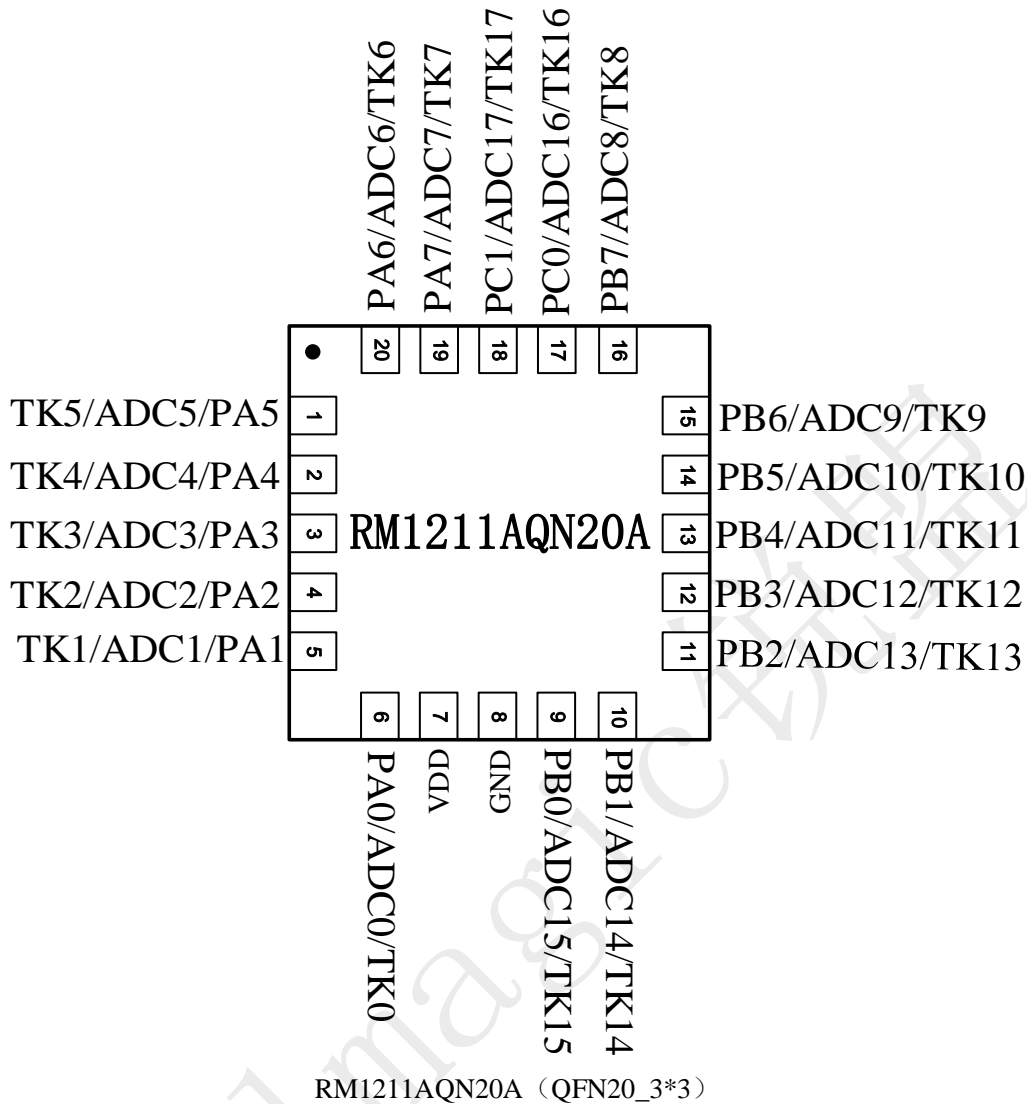
RM1211ASO16B (SOP16)



RM1211ATS16A (TSSOP16)



RM1211AQN16A (QFN16\_3\*3)



## 2.2 脚位定义

管脚号	管脚名称	管脚类型	描述
<b>20-PIN</b>			
1	VSS	P	地
2	PB0	I/O	双向 GPIO 口
	ADC15	AI	ADC 通道 15 输入
	TK15	AI	触摸按键通道 15 输入
	XIN	AI	外置晶振输入脚
	A10	AO	运放/比较器 1 输出
	T1PWM2N	O	Tiemr1 PWM2 反向输出
3	T2PWM0	O	Tiemr2 PWM0 输出
	PB1	I/O	双向 GPIO 口

	ADC14	AI	ADC 通道 14 输入
	TK14	AI	触摸按键通道 14 输入
	XOUT	AI	外置晶振输出脚
	VREF	AO	内部参考电压输出
	RCOUT	O	芯片内部主时钟输出
	T1PWM1	O	Tiemr1 PWM1 正向输出
	TX	O	UART 输出
4	PB2	I/O	双向 GPIO 口
	ADC13	AI	ADC 通道 13 输入
	TK13	AI	触摸按键通道 13 输入
	RSTN	I	外部复位输入口
	A2N	AI	运放/比较器 2 负端输入
	T1PWM2	O	Tiemr1 PWM2 正向输出
	T2PWM1	O	Tiemr2 PWM1 输出
5	PB3	I/O	双向 GPIO 口
	ADC12	AI	ADC 通道 12 输入
	TK12	AI	触摸按键通道 12 输入
	A3P	AI	运放/比较器 3 正端输入
	INT1	I	外部中断 1 输入
	T1PWM1	O	Tiemr1 PWM1 正向输出
	T1PWM0N	O	Tiemr1 PWM0 反向输出
6	PB4	I/O	双向 GPIO 口
	ADC11	AI	ADC 通道 11 输入
	TK11	AI	触摸按键通道 11 输入
	A3N	AI	运放/比较器 3 负端输入
	RX	I	UART 输入
	T1	I	Timer1 计数输入
	T1PWM0	O	Tiemr1 PWM0 正向输出
7	PB5	I/O	双向 GPIO 口
	ADC10	AI	ADC 通道 10 输入
	TK10	AI	触摸按键通道 10 输入
	A3O	AO	运放/比较器 3 输出
	INT1	I	外部中断 1 输入
	T1PWM0N	O	Tiemr1 PWM0 反向输出
	T1PWM1	O	Tiemr1 PWM1 正向输出
T1PWM2	O	Tiemr1 PWM2 输出	
8	PB6	I/O	双向 GPIO 口
	ADC9	AI	ADC 通道 9 输入
	TK9	AI	触摸按键通道 9 输入
	A2O	AO	运放/比较器 2 输出
	T1PWM1N	O	Tiemr1 PWM1 反向输出
	T2PWM1	O	Tiemr2 PWM1 输出

9	PB7	I/O	双向 GPIO 口
	ADC8	AI	ADC 通道 8 输入
	TK8	AI	触摸按键通道 8 输入
	A2N	AI	运放/比较器 2 负端输入
	T0	I	Timer0 计数输入
	T1PWM2N	O	Tiemr1 PWM2 反向输出
	INT0	I	外部中断 0 输入
10	PC0	I/O	双向 GPIO 口
	ADC16	AI	ADC 通道 16 输入
	TK16	AI	触摸按键通道 16 输入
	A2P	AI	运放/比较器 2 正端输入
	VREF	AO	内部参考电压输出
	TX	O	UART 输出
	T2PWM0	O	Tiemr2 PWM0 输出
T1PWM2	O	Tiemr1 PWM2 正向输出	
11	PC1	I/O	双向 GPIO 口
	ADC17	AI	ADC 通道 17 输入
	TK17	AI	触摸按键通道 17 输入
	A1P	AI	运放/比较器 1 正端输入
	RCOUT	O	芯片内部主时钟输出
	RX	I	UART 输入
	T1PWM3	O	Tiemr1 PWM3 正向输出
12	PA7	I/O	双向 GPIO 口
	ADC7	AI	ADC 通道 7 输入
	TK7	AI	触摸按键通道 7 输入
	A1N	AI	运放/比较器 1 负端输入
	T1PWM3N	O	Tiemr1 PWM3 反向输出
13	PA6	I/O	双向 GPIO 口
	ADC6	AI	ADC 通道 6 输入
	TK6	AI	触摸按键通道 6 输入
	A1O	AO	运放/比较器 1 输出
	T1PWM3	O	Tiemr1 PWM3 正向输出
	T1PWM0	O	Tiemr1 PWM0 正向输出
14	PA5	I/O	双向 GPIO 口
	ADC5	AI	ADC 通道 5 输入
	TK5	AI	触摸按键通道 5 输入
	A0O	AO	运放/比较器 0 输出
	T1PWM1N	O	Tiemr1 PWM1 反向输出
15	PA4	I/O	双向 GPIO 口
	ADC4	AI	ADC 通道 4 输入
	TK4	AI	触摸按键通道 4 输入
	A0N	AI	运放/比较器 0 负端输入

	VREF	AO	内部参考电压输出
	RX	I	UART 输入
	T1PWM0N	O	Tiemr1 PWM0 反向输出
16	PA3	I/O	双向 GPIO 口
	PDA	I/O	串行编程和调试数据输入输出
	ADC3	AI	ADC 通道 3 输入
	TK3	AI	触摸按键通道 3 输入
	A0P	AI	运放/比较器 0 正端输入
	TX	O	UART 输出
	T2PWM1	O	Tiemr2 PWM1 正向输出
	T1PWM0	O	Tiemr1 PWM0 正向输出
17	PA2	I/O	双向 GPIO 口
	PCL	I/O	串行编程和调试时钟输入
	ADC2	AI	ADC 通道 2 输入
	TK2	AI	触摸按键通道 2 输入
	A2P	AI	运放/比较器 2 正端输入
	T2PWM0	O	Tiemr2 PWM0 输出
	T1PWM1N	O	Tiemr1 PWM1 反向输出
18	PA1	I/O	双向 GPIO 口
	ADC1	AI	ADC 通道 1 输入
	TK1	AI	触摸按键通道 1 输入
	VREF	AO	内部参考电压输出
	A1P	AI	运放/比较器 1 正端输入
	INT0	I	外部中断 0 输入
	TX	O	UART 输出
	T1PWM3	O	Tiemr1 PWM3 正向输出
19	PA0	I/O	双向 GPIO 口
	ADC0	AI	ADC 通道 0 输入
	TK0	AI	触摸按键通道 0 输入
	A1N	AI	运放/比较器 1 负端输入
	INT1	I	外部中断 1 输入
	RX	I	UART 输入
	T1PWM3N	O	Tiemr1 PWM3 反向输出
20	VDD	P	电源

注： 引脚类型 I = 数字输入, O = 数字输出; AI = 模拟输入; P = 电源;



### 3 CPU

#### 3.1 CPU 核

自研的 1T RISC core, 指令效率高, 运行速度达到 16MHz。支持 C 编译器, IDE 用户界面友好, 使用简单。

#### 3.2 程序存储器(CODE)

##### (1) 程序存储器 ROM

程序存储器主要用于指令的存储, 在 RM1211A 中, 该程序存储器是 4K\*16bit 的程序 MTP, 该存储器可以提供 1000 次的烧写, 具体操作请参考后面的章节<MTP 自编程模块>。系统的 reset 地址为 0x000。

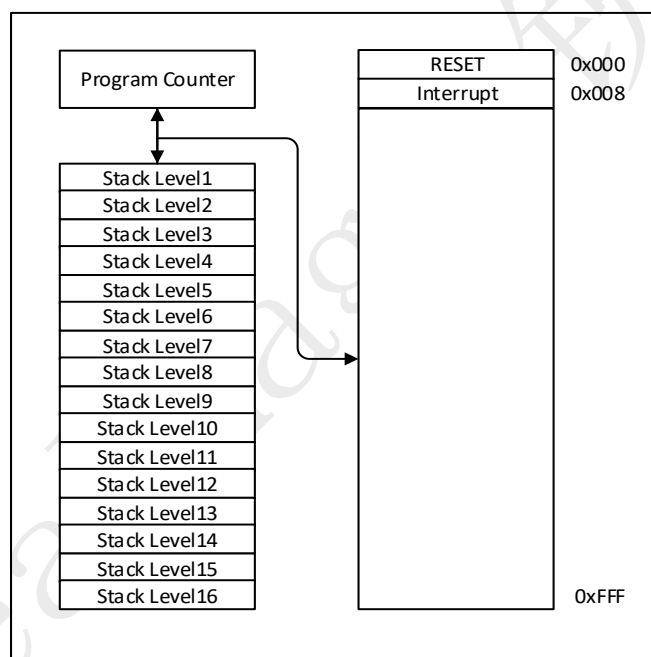


图1 程序存储器

#### 3.3 数据存储器(SRAM)

数据存储器主要用于程序运行过程中, 全局以及中间变量的存储。在 RM1211A 中, 数据存储器一共有 384 个 Byte, 其中堆栈用到 32 个 Byte, 所以可供用户使用的 SRAM 大小为 384-32=352 Byte。这可用的 352Byte 被划分为 4 个区域。

- 不分页数据存储器: 0x60 ~ 0x7f (共 32Byte, 不管 PAGE 为何值, 都可以访问此区间)
- 分页 0 数据存储器: 0x80 ~ 0xff (当 PAGE=0 时)
- 分页 1 数据存储器: 0x80 ~ 0xff (当 PAGE=1 时)
- 分页 2 数据存储器: 0x80 ~ 0xbf (当 PAGE=2 时)

分页的选择由特殊功能寄存器 PFLAG0 的 PAGE 位来指定。PAGE 为 0 时，选择的是分页 0 数据存储区。PAGE 为 1 时，选择的是分页 1 数据存储区。PAGE 为 2 时，选择的是分页 2 数据存储区。分页 2 数据存储区的寻址范围是 0x80 ~ 0xbb，一共只有 64 个 byte，超出此范围为无效的地址。不分页数据存储区的访问不受 PAGE 的限制，不管 PAGE 为 0 或者 1 或其他值，对不分页数据的地址段 0x60 ~ 0x7f 的访问都是有效的。

进入中断服务程序时，PFLAG0 会被压入堆栈区；退出中断服务程序时 PFLAG0 恢复为对应的值。因此，位于 PFLAG0 的 PAGE 位也会被进栈保护和出栈恢复。客户无需操心当前设定的分页是否被改变。

<b>0x60~0x7F</b>	公共 SRAM 区(32Byte)				
<b>0x80~0xFF</b>	SRAM_PAGE0 (128Byte)	<b>0x80~0xFF</b>	SRAM_PAGE1 (128Byte)	<b>0x80~0xBF</b>	SRAM_PAGE2 (64Byte)

### 3.4 SFR

特殊功能寄存器(SFR)包含系统专用寄存器和辅助专用寄存器。芯片内部的 SFR 和 SRAM 分成三页，分别为 page0 和 page1 及 page2 这三页。软件可以通过配置状态寄存器 0(PFLAG0)的 PAGE[1:0](第 4-5 bit)来选择当前访问的 page。

其中地址 0x00~0x0F 的 SFR 不分页，也就是不管当前的 PAGE[1:0]为何值，都可以对这些地址进行读写访问。

下表是芯片内部所有 SFR 的总表和各个 SFR 每个 bit 的明细。

PAGE 0		PAGE 1		PAGE 2	
偏移地址 (0X)	名称	偏移地址 (0X)	名称	偏移地址 (0X)	名称
0	IAR0	0	IAR0	0	IAR0
1		1		1	
2	MP0	2	MP0	2	MP0
3		3		3	
4	PFLAG0	4	PFLAG0	4	PFLAG0
5	ACC	5	ACC	5	ACC
6	IF0	6	IF0	6	IF0
7	IEN0	7	IEN0	7	IEN0
8	PFLAG1	8	PFLAG1	8	PFLAG1
9	PCL	9	PCL	9	PCL
A	IF1	A	IF1	A	IF1
B	IEN1	B	IEN1	B	IEN1
C	IF2	C	IF2	C	IF2
D	IEN2	D	IEN2	D	IEN2
E		E		E	
F		F		F	
10	OSCM	10	SCON	10	IAPTRIG
11	WDTM	11	SBUF	11	IAPCON
12	WDTR	12	SOVRL	12	IAPADRH
13	WKTCON	13	SOVRH	13	IAPADRL
14	RWK	14		14	IAPDATH
15		15		15	IAPDATL
16		16		16	E2CON
17	T0M	17		17	
18	T0C	18		18	
19	T0R	19		19	T3M
1A		1A		1A	T3RH
1B	T1M0	1B		1B	T3RL
1C	T1M1	1C		1C	
1D		1D		1D	
1E		1E		1E	
1F	T1RH	1F		1F	
20	T1RL	20		20	EXINTS
21	T1D0H	21		21	PWKEN0
22	T1D0L	22		22	PWKEN1
23	T1D1H	23		23	PWKEN2
24	T1D1L	24		24	METCH
25	T1PWM0DZH	25		25	BODCON
26	T1PWM0DZL	26		26	
27	T1PWM1DZH	27		27	
28	T1PWM1DZL	28		28	ICK_TRIM
29	T1M2	29		29	WCK_TRIM
2A	T1D2H	2A		2A	
2B	T1D2L	2B		2B	
2C	T1D3H	2C		2C	
2D	T1D3L	2D		2D	
2E		2E		2E	
2F	T2M0	2F		2F	

30	T2M1	30		30	CMP0CON0
31	T2D0H	31		31	CMP0CON1
32	T2D0L	32		32	CMP0CON2
33	T2D1H	33		33	CMP0CON3
34	T2D1L	34		34	CMP1CON0
35	T2CH	35		35	CMP1CON1
36	T2CL	36		36	CMP1CON2
37	T2RH	37		37	CMP1CON3
38	T2RL	38		38	CMP2CON0
39	T2CAPINT	39		39	CMP2CON1
3A	T2CAPCON	3A		3A	CMP2CON2
3B	T2CAPNUM	3B		3B	CMP2CON3
3C	T2CAPNFCNT	3C		3C	CMP3CON0
3D		3D		3D	CMP3CON1
3E		3E		3E	CMP3CON2
3F	ADCCON0	3F		3F	CMP3CON3
40	ADCCON1	40		40	
41	ADCCON2	41		41	
42	ADCCON3	42		42	LCDCON
43	ADCCON4	43		43	PAVO
44		44		44	PBVO
45		45		45	DAC0CON0
46	ADCRL	46		46	DAC0CON1
47	ADCRH	47		47	DAC1CON0
48		48		48	DAC1CON1
49	PACHS	49		49	
4A	PBCHS	4A		4A	SHDCON
4B	PCCHS	4B		4B	
4C		4C		4C	
4D		4D		4D	
4E	PORTA	4E		4E	
4F	PAM	4F		4F	
50	PAUR	50		50	
51	PADR	51		51	
52	PAOD	52		52	
53	PORTB	53		53	
54	PBM	54		54	
55	PBUR	55		55	
56	PBDR	56		56	
57	PBOD	57		57	
58	PORTC	58		58	
59	PCM	59		59	
5A	PCUR	5A		5A	
5B	PCDR	5B		5B	
5C	PCOD	5C		5C	
5D	PAIO	5D		5D	
5E	PBIO	5E		5E	
5F		5F		5F	
60~7F	公共SRAM区	60~7F	公共SRAM区	60~7F	公共SRAM区
80~FF	SRAM0(128Byte)	80~FF	SRAM1(128Byte)	80~BF	SRAM2(64Byte)

Addr	PAGE 0								
	NAME	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	IAR0	IAR0[7:0]							
01									
02	MP0	MP0[7:0]							
03									
04	PFLAG0	—	PAGE[1:0]			OV	Z	AC	C
05	ACC	ACC[7:0]							
06	IF0	—	CMP0F	ADCF	BODF	EXT1F	T1F	T0F	EXT0F
07	IEN0	GIE	ECMP0	EADC	EBOD	EXT1	ET1	ET0	EXT0
08	PFLAG1	—	—	—	—	—	BODOUT	PD	—
09	PCL	PCL[7:0]							
0A	IF1	CMP3F	CMP2F	CMP1F	T3F	T1PWM3F	T1PWM2F	T1PWM1F	T1PWM0F
0B	IEN1	ECMP3	ECMP2	ECMP1	ET3	ET1PWM3	ET1PWM2	ET1PWM1	ET1PWM0
0C	IF2	—	—	—	—	—	T2PWM1F	T2PWM0F	T2F
0D	IEN2	—	—	—	EWKT	EUART	ET2PWM1	ET2PWM0	ET2
0E									
0F									
10	OSCM	C2EN	SYSCOEN		PCKEN	RUNWCK	WCKEN	WCKS	SYCKEN
11	WDTM	WDTRF	WDTSEL	WDTEN	WDTCLR	WDTIS[3:0]			
12	WDTR	WDTR[7:0]							
13	WKTCON	—	—	—	WFTF	WKTR	WKPS[2:0]		
14	RWK	RWK[7:0]							
15									
16									
17	T0M	T0CLR	T0M	—	T0EG	T0PSC[2:0]		T0EN	
18	T0C	T0C[7:0]							
19	T0R	T0R[7:0]							
1A									
1B	T1M0	T1CLR	T1M	—	T1EG	T1PSC[2:0]		T1EN	
1C	T1M1	T1PWM1NS	T1PWM1S	T1PWM0NS	T1PWM0S	T1PWM1NEN	T1PWM1EN	T1PWM0NEN	T1PWM0EN
1D									
1E									
1F	T1RH	T1R[15:8]							
20	T1RL	T1R[7:0]							
21	T1D0H	T1D0[15:8]							
22	T1D0L	T1D0[7:0]							
23	T1D1H	T1D1[15:8]							
24	T1D1L	T1D1[7:0]							
25	T1PWM0DZH	T1PWM0DZ [15:8]							
26	T1PWM0DZL	T1PWM0DZ [7:0]							
27	T1PWM1DZH	T1PWM1DZ [15:8]							
28	T1PWM1DZL	T1PWM1DZ [7:0]							
29	T1M2	T1PWM3NS	T1PWM3S	T1PWM2NS	T1PWM2S	T1PWM3NEN	T1PWM3EN	T1PWM2NEN	T1PWM2EN
2A	T1D2H	T1D2[15:8]							
2B	T1D2L	T1D2[7:0]							
2C	T1D3H	T1D3[15:8]							
2D	T1D3L	T1D3[7:0]							
2E									
2F	T2M0	T2CLR	—	—	—	T2PSC[2:0]		T2EN	

Addr	PAGE 0									
	NAME	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	
30	T2M1	—	T2PWM1S	—	T2PWM0S	—	T2PWM1EN	—	T2PWM0EN	
31	T2D0H	T2D0[15:8]								
32	T2D0L	T2D0[7:0]								
33	T2D1H	T2D1[15:8]								
34	T2D1L	T2D1[7:0]								
35	T2CH	T2C[15:8]								
36	T2CL	T2C[7:0]								
37	T2RH	T2R[15:8]								
38	T2RL	T2R[7:0]								
39	T2CAPINT	—	—	T2C1F	T2C0F	—	—	ET2C1	ET2C0	
3A	T2CAPCON	—	—	—	—	T2C1MOD[1:0]		T2C0MOD[1:0]		
3B	T2CAPNUM	—	T2C1NUM[2:0]			—	T2C0NUM[2:0]			
3C	T2CAPNFCNT	T2CAPNFCNT[7:0]								
3D										
3E										
3F	ADCCON0	ADCF	ADCS	—	—	—	—	—	—	
40	ADCCON1	CHOP	ADVREFS[2:0]			—	—	—	ADCEN	
41	ADCCON2	—	—	—	—	—	ADCKS[2:0]			
42	ADCCON3	ADTS[1:0]		VCMS	ADT[4:0]					
43	ADCCON4	DFILEN	DFILNUM[1:0]		ADCHS[4:0]					
44										
45										
46	ADCR1	BGVS[1:0]		TSAMPS[1:0]		ADCR[7:0]				
47	ADCRH	ADCR[11:4]								
48										
49	PACHS	PACHS[7:0]								
4A	PBCHS	PBCHS[7:0]								
4B	PCCHS	—	—	—	—	—	—	PCCHS[1:0]		
4C										
4D										
4E	PORTA	PORTA[7:0]								
4F	PAM	PAM[7:0]								
50	PAUR	PAUR[7:0]								
51	PADR	PADR[7:0]								
52	PAOD	PAOD[7:0]								
53	PORTB	PORTB[7:0]								
54	PBM	PBM[7:0]								
55	PBUR	PBUR[7:0]								
56	PBDR	PBDR[7:0]								
57	PBOD	PBOD[7:0]								
58	PORTC	—						PORTC[1:0]		
59	PCM	—						PCM[1:0]		
5A	PCUR	—						PCUR[1:0]		
5B	PCDR	—						PCDR[1:0]		
5C	PCOD	—						PCOD[1:0]		
5D	PAIO	PBIOL1	PBIOL0	PBIOH1	PBIOH0	PAIOL1	PAIOL0	PAIOH1	PAIOH0	
5E	PBIO	—	IOVTS[2:0]			PB4HDR	PB3HDR	PCIOLO	PCIOHO	
5F										

Addr	PAGE1								
	NAME	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
00	IAR0	IAR0[7:0]							
01									
02	MP0	MP0[7:0]							
03									
04	STATUS	–	PAGE[1:0]			OV	Z	AC	C
05	ACC	ACC[7:0]							
06	IF0	–	CMP0F	ADCF	LVDF	EXT1F	T1F	T0F	EXT0F
07	IEN0	EA	ECMP0	EADC	ELVD	EXT1	ET1	ET0	EXT0
08	STA1	–	–	–	–	–	BODOUT	PD	–
09	PCL	PCL[7:0]							
0A	IF1	CMP3F	CMP2F	CMP1F	T3F	T1PWM3F	T1PWM2F	T1PWM1F	T1PWM0F
0B	IEN1	ECMP3	ECMP2	ECMP1	ET3	ET1PWM3	ET1PWM2	ET1PWM1	ET1PWM0
0C	IF2	–	–	–	–	–	T2PWM1F	T2PWM0F	T2F
0D	IEN2	–	–	–	EWKT	EUART	ET2PWM1	ET2PWM0	ET2
0E									
0F									
10	SCON	–	–	B8EN	REN	TB8	RB8	TI	RI
11	SBUF	SBUF[7:0]							
12	SOVRL	SOVRL[7:0]							
13	SOVRH	–	–	–	–	–	SOVR[10:8]		
14									
15									
16									
17									
18									
19									
1A									
1B									
1C									
1D									
1E									
1F									
20									
21									
22									
23									
24									
25									
26									
27									
28									
29									
2A									
2B									
2C									
2D									
2E									
2F									

Addr	PAGE 1								
	NAME	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
30									
31									
32									
33									
34									
35									
36									
37									
38									
39									
3A									
3B									
3C									
3D									
3E									
3F									
40									
41									
42									
43									
44									
45									
46									
47									
48									
49									
4A									
4B									
4C									
4D									
4E									
4F									
50									
51									
52									
53									
54									
55									
56									
57									
58									
59									
5A									
5B									
5C									
5D									
5E									
5F									



Addr	PAGE2									
	NAME	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	
00	IAR0	IAR0[7:0]								
01										
02	MP0	MP0[7:0]								
03										
04	STATUS	–	PAGE[1:0]			OV	Z	AC	C	
05	ACC	ACC[7:0]								
06	IF0	–	CMP0F	ADCF	LVDF	EXT1F	T1F	T0F	EXT0F	
07	IEN0	EA	ECMP0	EADC	ELVD	EXT1	ET1	ET0	EXT0	
08	STA1	–	–	–	–	–	BODOUT	PD	–	
09	PCL	PCL[7:0]								
0A	IF1	CMP3F	CMP2F	CMP1F	T3F	T1PWM3F	T1PWM2F	T1PWM1F	T1PWM0F	
0B	IEN1	ECMP3	ECMP2	ECMP1	ET3	ET1PWM3	ET1PWM2	ET1PWM1	ET1PWM0	
0C	IF2	–	–	–	–	–	T2PWM1F	T2PWM0F	T2F	
0D	IEN2	–	–	–	EWKT	EUART	ET2PWM1	ET2PWM0	ET2	
0E										
0F										
10	IAPTRIG	IAPTRIG [7:0]								
11	IAPCON	–	IAPLOCK	WAITEN	STATICEN	WAITSEL[1:0]		IAPRD	IAPWR	
12	IAPADRH	–	–	–	–	IAPADRH [11:8]				
13	IAPADRL	IAPADRL [7:0]								
14	IAPDATH	IAPDATH [15:8]								
15	IAPDATL	IAPDATL [7:0]								
16	E2CON	–	–	–	–	–	–	E2RD	E2WR	
17										
18										
19	T3M	–	–	–	–	T3PSC[2:0]			T3EN	
1A	T3RH	T3R [15:8]								
1B	T3RL	T3R [7:0]								
1C										
1D										
1E										
1F										
20	EXINTS	–	–	–	–	E1EG[1:0]		E0EG[1:0]		
21	PWKEN0	PWKEN0 [7:0]								
22	PWKEN1	PWKEN1 [7:0]								
23	PWKEN2	–	–	–	–	–	–	PWKEN2 [1:0]		
24	METCH	METCH [7:0]								
25	BODCON	BODOF	LVREN	–	BODSEL[3:0]				BODEN	
26										
27										
28	ICK_TRIM	ICK_TRIM [7:0]								
29	WCK_TRIM	WCK_TRIM [7:0]								
2A										
2B										
2C										
2D										
2E										
2F										

Addr	PAGE 2								
	NAME	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
30	CMP0CON0	CMP0O	CMP0HYS	SF0EN	OPA0CRS	OPA0COFM	CMP0OEN	CMP0MODE	OPA0EN
31	CMP0CON1	A0P[2:0]			OPA0ADJ[4:0]				
32	CMP0CON2	A0N[2:0]			A0SW4	A0SW3	A0SW2	A0SW1	A0SW0
33	CMP0CON3	A0GS[2:0]			CMP0INV	CMP0DBS[1:0]		INT0SEL[1:0]	
34	CMP1CON0	CMP1O	CMP1HYS	SF1EN	OPA1CRS	OPA1COFM	CMP1OEN	CMP1MODE	OPA1EN
35	CMP1CON1	A1P[2:0]			OPA1ADJ[4:0]				
36	CMP1CON2	A1N[2:0]			A1SW4	A1SW3	A1SW2	A1SW1	A1SW0
37	CMP1CON3	A1GS[2:0]			CMP1INV	CMP1DBS[1:0]		INT1SEL[1:0]	
38	CMP2CON0	CMP2O	CMP2HYS	SF2EN	OPA2CRS	OPA2COFM	CMP2OEN	CMP2MODE	OPA2EN
39	CMP2CON1	A2P[2:0]			OPA2ADJ[4:0]				
3A	CMP2CON2	A2N[2:0]			A2SW4	A2SW3	A2SW2	A2SW1	A2SW0
3B	CMP2CON3	A2GS[2:0]			CMP2INV	CMP2DBS[1:0]		INT2SEL[1:0]	
3C	CMP3CON0	CMP3O	CMP3HYS	SF3EN	OPA3CRS	OPA3COFM	CMP3OEN	CMP3MODE	OPA3EN
3D	CMP3CON1	A3P[2:0]			OPA3ADJ[4:0]				
3E	CMP3CON2	A3N[2:0]			A3SW4	A3SW3	A3SW2	A3SW1	A3SW0
3F	CMP3CON3	A3GS[2:0]			CMP3INV	CMP3DBS[1:0]		INT3SEL[1:0]	
40									
41									
42	LDCON	—	—	LCDEN	VBIASS	VOIRS[1:0]		PCVO[1:0]	
43	PAVO	PAVO[7:0]							
44	PBVO	PBVO[7:0]							
45	DAC0CON0	SVL0[7:0]							
46	DAC0CON1	—	—	DAC0EN	SVRF0	—	—	SLV0[9:8]	
47	DAC1CON0	SVL1[7:0]							
48	DAC1CON1	—	—	DAC1EN	SVRF1	—	—	SLV1[9:8]	
49									
4A	SHDCON	—	—	OPAIS[1:0]		SHDEN[3:0]			
4B									
4C									
4D									
4E									
4F									
50									
51									
52									
53									
54									
55									
56									
57									
58									
59									
5A									
5B									
5C									
5D									
5E									
5F									

## 3.5 状态寄存器

状态寄存器包含 ALU 的算术状态及复位状态。状态寄存器类似于其它寄存器，可以作为任何指令的目标寄存器。如果状态寄存器是某条指令的目标寄存器，而且影响到 Z, AC 或 C 位，那么对这三个位只可写 0。

### 3.5.1 状态寄存器 0(PFLAG0)

寄存器	地址	R/W	描述	复位后的值
<b>PFLAG0</b>	0x04	R/W	状态寄存器 0	0000 0000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留		PAGE[1:0]		OV	Z	AC	C

位	描述	
[7:6]	保留	保留
[5:4]	PAGE[1:0]	<b>SRAM 和 SFR 页选择位</b> 00: 选择页 0 01: 选择页 1 10: 选择页 2 11: 保留
[3]	OV	<b>溢出标志（用于有符号数运算时才有意义）</b> 0: 有符号数算术结果是错误 1: BIT7 有进位（或借位），但 BIT6 没有进位（或没有借位）
[2]	Z	<b>零标志</b> 0: 算术或逻辑操作结果不为 0 1: 算术或逻辑操作结果为 0
[1]	AC	<b>半进位标志/借位标志</b> 用于借位时，极性相反 0: 结果的最高位不出现进位溢出 1: 结果的最高位出现进位溢出
[0]	C	<b>进位标志/借位标志</b> 用于借位时，极性相反 0: 结果的最高位不出现进位溢出 1: 结果的最高位出现进位溢出

### 3.5.1 状态寄存器 1(PFLAG1)

寄存器	地址	R/W	描述	复位后的值
<b>PFLAG1</b>	0x08	R/W	状态寄存器 0	0000 0000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留					<b>BODOUT</b>	<b>PD</b>	保留

位	描述	
[7:3]	保留	保留
[2]	<b>BODOUT</b>	低电压检测 (BOD) 状态标志位 0: 工作电压高于检测电压阈值 1: 工作电压低于检测电压阈值
[1]	<b>PD</b>	掉电标志位, STOP 命令后为 1, 由软件清 0 0: 上电复位后或硬件复位后 1: 执行 STOP 指令后
[0]	保留	保留

### 3.6 间接寻址(IAR0 和 MP0)

通过 IAR 寄存器可以实现间接寻址。IAR 是虚拟寄存器，其对应的真实寄存器是 MSR 寄存器数据所指向的寄存器。比如对地址 80h 的 RAM 写入数据 0x55，可以如下操作：

```
Mov a, #0x80
Mov MP0, a
Mov a, #0x55
Mov IAR0, a
```

## 4 时钟系统

### 4.1 特性

RM1211A 有两个时钟源。一个是内部集成的时钟 ICK(16MHz)，另一个是外部时钟。由代码选项选择 CPU 跑内部或外部时钟。

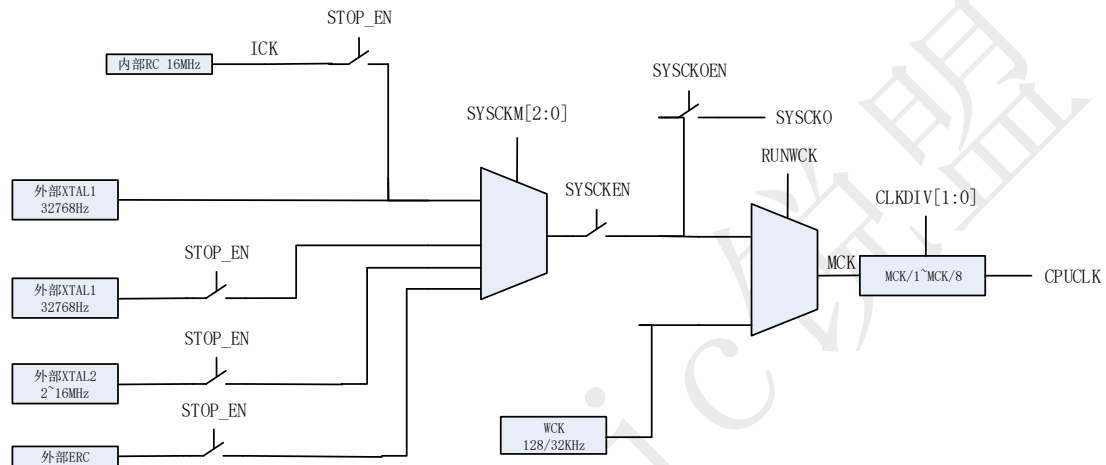


图2 RM1211A 振荡器状态框图

#### 4.1.1 系统时钟配置寄存器(OSCM)

寄存器	地址	R/W	描述	复位后的值
<b>OSCM</b>	0x10	R/W	系统时钟配置寄存器	1000 0011

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>C2EN</b>	<b>SYSCKOEN</b>	保留	<b>PCKEN</b>	<b>RUNWCK</b>	<b>WCKEN</b>	<b>WCKS</b>	<b>SYSCKEN</b>

位	描述
[7]	<b>C2EN</b> <b>带电烧录管脚使能位</b> 0: PA2/PA3 为普通 GPIO 口，禁止随时进烧录模式 1: PA2/PA3 为烧录口，可以随时进烧录模式
[6]	<b>SYSCKOEN</b> <b>系统时钟 PB1/PC1 口输出控制位</b> 0: 禁止系统时钟输出 1: 允许系统时钟输出
[5]	保留
[4]	<b>PCKEN</b> <b>Timer1 PWM 的定时时钟，也就是芯片内部 16M 时钟的使能信号。</b> 0: 无效。 1: 使能

[3]	<b>RUNWCK</b>	<b>看门狗时钟 WCK 作为系统时钟使能位</b> 0: CLKCPU 为系统时钟 1: 系统时钟为 WCK (若切换之前 WCK 为关闭状态, CPU 会自动打开内置 128K 振荡器, 然后将主时钟切换到 WCK)
[2]	<b>WCKEN</b>	<b>内部低速时钟 WCK 使能位</b> 0: 禁止内部低速时钟 WCK 1: 使能内部低速时钟 WCK
[1]	<b>WCKS</b>	<b>内部低速时钟 WCK 选择位</b> 0: 选择 WCK 为 32KHz 1: 选择 WCK 为 128KHz
[0]	<b>SYSCKEN</b>	<b>系统时钟使能位</b> 0: 禁止系统时钟 SYSCK 1: 使能系统时钟 SYSCK

## 5 电源管理

RM1211A 支持低电压工作模式。为了使 RM1211A 处于待机状态，可以让 CPU 停止工作，使 RM1211A 进行 STOP 或 IDLE 模式，减低功耗。这两种模式描述如下：

### 5.1 IDLE 模式

IDLE 模式能够降低系统功耗，在此模式下振荡器等模块不会自动关闭，程序计数器停止运行，CPU 时钟停止，但外设时钟继续运行。CPU 进入 IDLE 时，之前所有状态都被保存，如 PC、PFLAG0、SFR 等等。

任何一个中断、WKT、IO 翻转唤醒、复位信号（外部硬件复位、WDT 复位、LVR 复位）等均能退出 IDLE 模式。

### 5.2 STOP 模式

STOP 模式（掉电模式）能够达到系统最低功耗。CPU 进入 STOP 时，之前所有状态都被保存，如 PC、PFLAG0、SFR 等等。STOP 模式典型情况下，主振荡器模块将自动关闭（WCK 假如开启了 WDT 或者 WKT 功能，需软件关闭），如果软件也禁止了其他外设功能，所有输出引脚无负载，输入引脚不翻转，系统将达到约 1uA 的待机电流。

IO 翻转、外部中断（高电平，低电平）、RTC 定时和看门狗溢出以及 WKT，TK 等都可以唤醒 STOP 模式。

外部硬件复位、WKT、WDT 复位（如果被允许）、LVR 复位（如果被允许）均能退出 STOP 模式。

**注意：** STOP 模式下：外部中断只能由高/低电平唤醒。

**IDLE 模式下：**外部中断可以由高/低电平，上/下降沿唤醒。

### 5.3 低电压检测

RM1211A 内部提供 16 档电压检测，BODSEL[3:0]为选择电压检测阈值，由 BODOF 标志位看电压检测结果。

#### 5.3.1 BOD 配置寄存器(BODCON)

寄存器	地址	R/W	描述	复位后的值
BODCON	0x25	R/W	低电压检测配置寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
BODOF	LVREN	保留	BODSEL[3:0]				BODEN

位	描述	
[7]	<b>BODOF</b>	<b>BOD 状态标志位, 1 表示芯片为低电压状态, 只读</b>
[6]	<b>LVREN</b>	<b>LVR 模块的使能信号</b> 0: 关闭 1: 使能
[5]	<b>保留</b>	<b>保留</b>
[4:1]	<b>BODSEL[3:0]</b>	<b>BOD 电压阈值选择</b> 0000: 1.9V 0001: 2.0V 0010: 2.1V 0011: 2.2V 0100: 2.3V 0101: 2.4V 0110: 2.5V 0111: 2.7V 1000: 3.0V 1001: 3.3V 1010: 3.6V 1011: 3.7V 1100: 3.9V 1101: 4.1V 1110: 4.3V 1111: 4.5V
[0]	<b>BODEN</b>	<b>低电压检测使能位</b> 0: 禁止低电压检测 1: 使能低电压检测



## 6 复位系统

RM1211A 有以下方式复位:

- 1) 上电复位
- 2) 低电压复位
- 3) WDT 复位 (正常操作)
- 4) WDT 复位 (从 STOP 模式)

RealMagic 锐盟

## 7 中断系统

### 7.1 特性

RM1211A 有 21 个中断源，只有一个中断入口地址 0x08H。响应中断之后需要软件将中断标志位清除，否则会不停地响应中断。

中断请求源	中断使能位	中断标志位	软件清除
EXT0	IEN0[0]	IF0[0]	写0清
TO/RTC	IEN0[1]	IF0[1]	写0清
T1	IEN0[2]	IF0[2]	写0清
EXT1	IEN0[3]	IF0[3]	写0清
LVDF	IEN0[4]	IF0[4]	写0清
SAR	IEN0[5]	IF0[5]	写0清
CMPO	IEN0[6]	IF0[6]	写0清
T1PWM0	IEN1[0]	IF1[0]	写0清
T1PWM1	IEN1[1]	IF1[1]	写0清
T1PWM2	IEN1[2]	IF1[2]	写0清
T1PWM3	IEN1[3]	IF1[3]	写0清
T3	IEN1[4]	IF1[4]	写0清
CMP1	IEN1[5]	IF1[5]	写0清
CMP2	IEN1[6]	IF1[6]	写0清
CMP3	IEN1[7]	IF1[7]	写0清
T2	IEN2[0]	IF2[0]	写0清
T2PWM0	IEN2[1]	IF2[1]	写0清
T2PWM1	IEN2[2]	IF2[2]	写0清
T2CAP	T2CAPINT[1:0]	T2CAPINT[5:4]	写0清
UART	IEN2[3]	SOCON[1:0]	写0清
WKT	IEN2[4]	WKCON[4]	写0清

## 7.2 寄存器

### 7.2.1 中断标志位寄存器 0(IF0)

寄存器	地址	R/W	描述	复位后的值
IF0	06H	R/W	中断标志位寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	CMP0F	ADCF	BODF	EXT1F	T1F	T0F	EXT0F

位	描述	
[7]	保留	保留
[6]	CMP0F	比较器 0 中断标志位
[5]	ADCF	ADC 中断标志位
[4]	BODF	低电压检测中断标志位
[3]	EXT1F	外部中断 1 标志位
[2]	T1F	定时器 1 溢出中断标志位
[1]	T0F	定时器 0 溢出中断标志位
[0]	EXT0F	外部中断 0 标志位

### 7.2.2 中断使能寄存器 0(IEN0)

寄存器	地址	R/W	描述	复位后的值
IEN0	07H	R/W	中断使能寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
GIE	ECMP0	EADC	EBOD	EXT1	ET1	ET0	EXT0

位	描述	
[7]	GIE	中断总使能位 0: 禁止所有中断 1: 使能中断总使能
[6]	ECMP0	比较器 0 中断使能位 0: 禁止比较器 0 中断 1: 使能比较器 0 中断
[5]	EADC	ADC 中断使能位 0: 禁止比较器 0 中断 1: 使能比较器 0 中断
[4]	EBOD	低电压检测中断使能位

		0: 禁止低电压检测中断 1: 使能低电压检测中断
[3]	<b>EXT1</b>	<b>外部中断 1 使能位</b> 0: 禁止外部中断 1 中断 1: 使能外部中断 1 中断
[2]	<b>ET1</b>	<b>定时器 1 溢出中断使能位</b> 0: 禁止定时器 1 中断 1: 使能定时器 1 中断
[1]	<b>ET0</b>	<b>定时器 0 溢出中断使能位</b> 0: 禁止定时器 0 中断 1: 使能定时器 0 中断
[0]	<b>EXT0</b>	<b>外部中断 0 使能位</b> 0: 禁止外部中断 0 中断 1: 使能外部中断 0 中断

### 7.2.3 中断标志位寄存器 1(IF1)

寄存器	地址	R/W	描述	复位后的值
<b>IF1</b>	0AH	R/W	中断标志位寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>CMP3F</b>	<b>CMP2F</b>	<b>CMP1F</b>	<b>T3F</b>	<b>T1PWM3F</b>	<b>T1PWM2F</b>	<b>T1PWM1F</b>	<b>T1PWM0F</b>

位	描述	
[7]	<b>CMP3F</b>	比较器 3 中断标志位
[6]	<b>CMP2F</b>	比较器 2 中断标志位
[5]	<b>CMP1F</b>	比较器 1 中断标志位
[4]	<b>T3F</b>	定时器 3 溢出中断标志位
[3]	<b>T1PWM3F</b>	T1PWM3 中断标志位
[2]	<b>T1PWM2F</b>	T1PWM2 中断标志位
[1]	<b>T1PWM1F</b>	T1PWM1 中断标志位
[0]	<b>T1PWM0F</b>	T1PWM0 中断标志位

### 7.2.4 中断使能寄存器 1(IEN1)

寄存器	地址	R/W	描述	复位后的值
<b>IEN1</b>	0BH	R/W	中断使能寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

ECMP3	ECMP2	ECMP1	ET3	ET1PWM3	ET1PWM2	ET1PWM1	ET1PWM0
-------	-------	-------	-----	---------	---------	---------	---------

位	描述	
[7]	<b>ECMP3</b>	<b>比较器 3 中断使能位</b> 0: 禁止比较器 3 中断 1: 使能比较器 3 中断
[6]	<b>ECMP2</b>	<b>比较器 2 中断使能位</b> 0: 禁止比较器 2 中断 1: 使能比较器 2 中断
[5]	<b>ECMP1</b>	<b>比较器 1 中断使能位</b> 0: 禁止比较器 1 中断 1: 使能比较器 1 中断
[4]	<b>ET3</b>	<b>定时器 3 溢出中断使能位</b> 0: 禁止定时器 3 中断 1: 使能定时器 3 中断
[3]	<b>ET1PWM3</b>	<b>T1PWM3 中断使能位</b> 0: 禁止 T1PWM3 中断中断 1: 使能 T1PWM3 中断中断
[2]	<b>ET1PWM2</b>	<b>T1PWM2 中断使能位</b> 0: 禁止 T1PWM2 中断中断 1: 使能 T1PWM2 中断中断
[1]	<b>ET1PWM1</b>	<b>T1PWM1 中断使能位</b> 0: 禁止 T1PWM1 中断中断 1: 使能 T1PWM1 中断中断
[0]	<b>ET1PWM0</b>	<b>T1PWM0 中断使能位</b> 0: 禁止 T1PWM0 中断中断 1: 使能 T1PWM0 中断中断

### 7.2.5 中断标志位寄存器 2(IF2)

寄存器	地址	R/W	描述	复位后的值
<b>IF2</b>	0CH	R/W	中断标志位寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留					<b>T2PWM1F</b>	<b>T2PWM0F</b>	<b>T2F</b>

位	描述	
[7:3]	保留	保留
[2]	<b>T2PWM1F</b>	<b>T2PWM1 中断标志位</b>
[1]	<b>T2PWM0F</b>	<b>T2PWM0 中断标志位</b>
[0]	<b>T2F</b>	<b>定时器 2 溢出中断标志位</b>

## 7.2.6 中断使能寄存器 2(IEN2)

寄存器	地址	R/W	描述	复位后的值
IEN2	0DH	R/W	中断使能寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	保留	保留	EWKT	EUART	ET2PWM1	ET2PWM0	ET2

位	描述	
[7]	保留	保留
[6]	保留	保留
[5]	保留	保留
[4]	<b>EWKT</b>	<b>WKT 中断使能位</b> 0: 禁止 WKT 中断 1: 使能 WKT 中断
[3]	<b>EUART</b>	<b>串口使能位</b> 0: 禁止串口中断 1: 使能串口中断
[2]	<b>ET2PWM1</b>	<b>T2PWM1 中断使能位</b> 0: 禁止 T2PWM1 中断 1: 使能 T2PWM1 中断
[1]	<b>ET2PWM0</b>	<b>T2PWM0 中断使能位</b> 0: 禁止 T2PWM0 中断 1: 使能 T2PWM0 中断
[0]	<b>ET2</b>	<b>定时器 2 溢出中断使能位</b> 0: 禁止定时器 2 中断 1: 使能定时器 2 中断

## 7.3 外部中断

芯片内部有两个外部中断，分别为外部中断 0 和外部中断 1。

PB.7(PA.1 和 PA.7 可选)可作为外部中断 0 的输入端。触发方式由寄存器 EXINTS 决定。IEN0 寄存器中的 IEN0[0]为外部中断 0 的使能位。外部中断 0 可唤醒 STOP 或 IDLE 模式。

PB.5(PB.3 和 PA.0 可选)可作为外部中断 1 的输入端。触发方式由寄存器 EXINTS 决定。IEN0 寄存器中的 IEN0[3]为外部中断 1 的使能位。外部中断 1 可唤醒 STOP 或 IDLE 模式。

### 7.3.1 外部中断触发方式选择控制寄存器 (EXINTS)

寄存器	地址	R/W	描述	复位后的值
EXINTS	0x20	R/W	外部中断触发方式选择控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留				E1EG[1:0]		E0EG[1:0]	

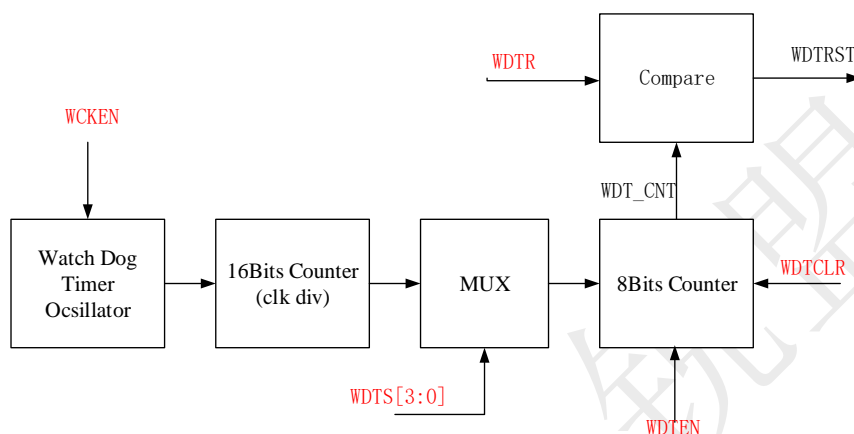
位	描述	
[7:4]	保留	保留
[3:2]	E1EG[1:0]	外部中断 1 边沿触发选择位 00: 低电平触发 01: 高电平触发 10: 下降沿触发 11: 上升沿触发
[1:0]	E0EG[1:0]	外部中断 0 边沿触发选择位 00: 低电平触发 01: 高电平触发 10: 下降沿触发 11: 上升沿触发

## 7.4 其他中断

见后续章节介绍。

## 8 看门狗(WDT)

### 8.1 特性



看门狗定时器功能框图

看门狗定时器 (WDT) 用于防止程序由于某些不确定因素而失去控制。当 WDT 启动时, WDT 计时超时后将使 CPU 复位。在运行的程序一般在 WDT 复位 CPU 之前先复位 WDT。当出现某些故障时, 程序会被 WDT 复位到正常状态下, 但程序不会复位 WDT。

当用户把 WCKEN 置 1 时, 则内部的看门狗定时器振荡器 (128KHz) 将会启动, 产生的时钟被送到“16 bits 计数器”来进行时钟分频。用户根据 WDS[3: 0]来选择不同的时钟分频系数来送给后面的“8 bits 计数器”。当用户置位 WDTEN 时, “8 bits 计数器”开始计数, “8 bits 计数器”的输出是内部信号 WDT\_CNT[7:0]。当 WDT\_CNT 的值与 WDR 数值相等时看门狗便溢出, 溢出时它会发送 WDRF 信号复位 CPU 及置位 WDRF 标志位。用户可以通过设置 WDTM 的 WDTCLR 位来清除。

### 8.2 寄存器

#### 8.2.1 WDT 配置寄存器(WDTM)

寄存器	地址	R/W	描述	复位后的值
WDTM	0x11	R/W	WDT 配置寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
WDRF	WDTSEL	WDTEN	WDTCLR	WDS[3:0]			

位	描述



[7]	<b>WDTRF</b>	看门狗复位标志位，看门狗产生复位时，此位置 1，软件清 0
[6]	<b>WDTSEL</b>	看门狗功能选择位 0: 复位功能 1: 唤醒功能
[5]	<b>WDTEN</b>	WDT 使能位 0: 禁止看门狗 1: 使能看门狗
[4]	<b>WDTCLR</b>	WDT 清狗 0: 不清零看门狗 1: 清零看门狗(此 bit 软件置 1，硬件清零)
[3:0]	<b>WDTS[3:0]</b>	看门狗定时分频 0000: $F_{wck}/65536$ 0001: $F_{wck}/32768$ 0010: $F_{wck}/16384$ 0011: $F_{wck}/8192$ 0100: $F_{wck}/4096$ 0101: $F_{wck}/2048$ 0110: $F_{wck}/1024$ 0111: $F_{wck}/512$ 1000: $F_{wck}/256$ 1001: $F_{wck}/128$ 1010: $F_{wck}/64$ 1011: $F_{wck}/32$ 1100: $F_{wck}/16$ 1101: $F_{wck}/8$ 1110: $F_{wck}/4$ 1111: $F_{wck}/2$

### 8.2.2 WDT 定时溢出寄存器(WDTR)

寄存器	地址	R/W	描述	复位后的值
<b>WDTR</b>	0x12	R/W	WDT 定时溢出寄存器	11111111

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>WDTR[7:0]</b>							

位	描述
[7:0]	<b>WDTR[7:0]</b> 看门狗定时溢出时间配置寄存器

操作：

1. 设置 WDTS[3:0]，选择 WDT 时钟频率。
2. 设置 WDTR，选择不同的溢出时间值。
3. 开启 WDT 时钟，把 OSCM 的 BIT2 (WCKEN) 置 1。
4. 选择 WDT 为复位功能：WDTSEL=0 作为复位；WDTSEL=1 时作为唤醒功能用；
5. 使能 WDT：WDTEN=1。
7. 在产生复位前，清狗，即设置寄存器的 WDTM 的 WDTCLR 位为 1。

**WDT 溢出时间计算公式：**

$$\text{溢出时间} = \frac{2^{(16-\text{WDTS}[3:0])}}{128k} * (\text{WDTR}[7:0] + 1)$$

WDTS[3:0]范围为 0~15，WDTR[7:0]范围为 0~255。

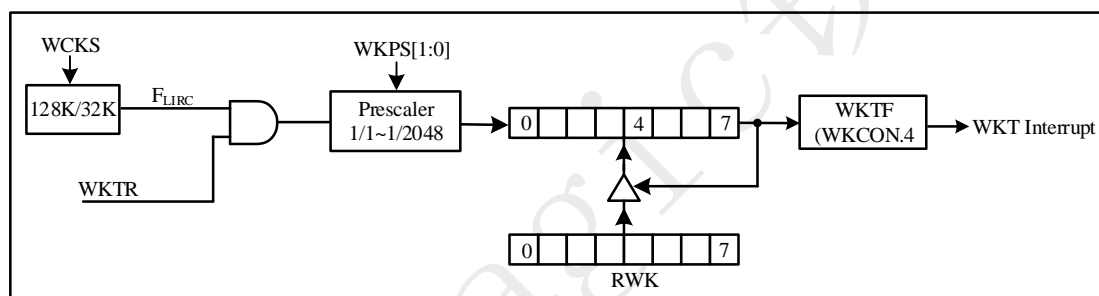
WDTS[3:0]	计数器时钟	时间 (当 WDTR=FFH)
0000	WDTCLK /65536	131072ms
0001	WDTCLK /32768	65536ms
0010	WDTCLK /16384	32768ms
0011	WDTCLK /8192	16384ms
0100	WDTCLK /4096	8192ms
0101	WDTCLK /2048	4096ms
0110	WDTCLK /1024	2048ms
0111	WDTCLK /512	1024ms
1000	WDTCLK /256	512ms
1001	WDTCLK /128	256ms
1010	WDTCLK /64	128ms
1011	WDTCLK /32	64ms
1100	WDTCLK /16	32ms
1101	WDTCLK /8	16ms
1110	WDTCLK /4	8ms
1111	WDTCLK /2	4ms

## 9 自唤醒定时器(WKT)

RM1211A 有一个专用的自唤醒定时器 (WKT)，用于低功耗模式下的周期唤醒芯片，也可用作通用定时器。WKT 保持计数在空闲或掉电模式。当 WKT 用作唤醒定时器时，WKT 要在进入省电模式之前开启。WKT 只能配置片内 128KHz/32KHz 时钟源 LIRC。如果 WKT 开始计数，在设备进入空闲或掉电模式下，选择的时钟源也要保持工作。注意选择的 WKT 时钟源不会连同 WKT 的配置自动使能，用户应该手动使能选择的时钟源并等待它稳定确保操作的成功。

WKT 配备了一个简单的 8 位自动重载向上计数定时器。它的预分频可选择从 1/1 到 1/2048，通过 WKPS[2:0] (WKTCON[2:0])来设置。用户填重装载值到 RWK 寄存器来决定它的溢出速率。WKTR(WKTCON.3)置位开始计数。当计数器溢出，WKTIF(WKTCON.4)置为 1，并重载 RWK 寄存器的值到内部 8 位计数器。如果 EWKT(IEN2[4])置为 1，WKT 中断服务程序将被执行。

WKT 溢出可唤醒 IDLE 或者 STOP 模式。



自唤醒定时器结构图

### 9.1 寄存器

#### 9.1.1 WKT 控制寄存器 (WKTCON)

寄存器	地址	R/W	描述	复位后的值
WKTCON	0x13	R/W	WKT 控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留			WKTIF	WKTR	WKPS[2:0]		

位	描述	
[7:5]	保留	保留
[4]	WKTIF	<b>WKT 溢出标志</b> 当 WKT 溢出，该位置位。如果 WKT 中断和全局中断使能，置位该位会使 CPU 执行 WKT 中断服务程序。该位不会被硬件自动清零，应该通过软件清零。

[3]	<b>WKTR</b>	<b>WKT 运行控制</b> 0: WKT 停止 1: WKT 开始运行 注意重载寄存器 RWK 仅在 WKT 停止的时候可以写入 (WKTR 位为 0)。否则结果是不可预知的。
[2:0]	<b>WKPS[2:0]</b>	<b>WKT 预分频</b> 这些位决定 WKT 时钟的预分频 000: 1/1 001: 1/4 010: 1/16 011: 1/64 100: 1/256 101: 1/512 110: 1/1024 111: 1/2048

### 9.1.2 WKT 重载数据寄存器 (RWK)

寄存器	地址	R/W	描述	复位后的值
<b>RWK</b>	0x14	R/W	WKT 重载数据寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>RWK [7:0]</b>							

位	描述
[7:0]	<b>RWK [7:0]</b> <b>WKT 重载字节</b> 用以保存 WKT 的 8 位重载值。注意如果预分频是 1/1, RWK 限制不能是 FFH。

## 10 GPIO

### 10.1 特性

RM1211A 提供 18 个 I/O 端口：所有引脚都是可编程双向 I/O 端口。18 个 IO 口都可以配置唤醒 STOP 模式和 IDLE 模式。所有引脚均内置上拉和下拉电阻或者开漏输出。

### 10.2 寄存器

#### 10.2.1 PA/PB/PC 端口数据锁存寄存器 (PORTA / PORTB / PORTC)

寄存器	地址	R/W	描述	复位后的值
PORTA	0x4E	R/W	PA 端口数据锁存寄存器	00000000
PORTB	0x53	R/W	PB 端口数据锁存寄存器	00000000
PORTC	0x58	R/W	PC 端口数据锁存寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PORTx[7:0] x=A,B,C							

位	描述	
[7:0]	PORTx x=A,B,C	Px 输入输出数据

#### 10.2.2 PA/PB/PC 端口控制寄存器 (PAM/PBM/PCM)

寄存器	地址	R/W	描述	复位后的值
PAM	0x4F	R/W	PA 端口控制寄存器	00000000
PBM	0x54	R/W	PB 端口控制寄存器	00000000
PCM	0x59	R/W	PC 端口控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PxM[7:0] x=A,B,C							

位	描述	
[7:0]	<b>PxM</b> x=A,B,C	Px 输入输出控制位 0: 输入 1: 输出

### 10.2.3 PA/PB/PC 上拉控制寄存器 (PAUR/PBUR/PCUR)

寄存器	地址	R/W	描述	复位后的值
<b>PAUR</b>	0x50	R/W	PA 上拉控制寄存器	00000000
<b>PBUR</b>	0x55	R/W	PB 上拉控制寄存器	00000000
<b>PCUR</b>	0x5A	R/W	PC 上拉控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>PxUR[7:0]</b> x=A,B,C							

位	描述	
[7:0]	<b>PxUR</b> x=A,B,C	Px 上拉电阻控制位 0: 禁止上拉电阻 1: 使能上拉电阻

### 10.2.4 PA/PB/PC 下拉控制寄存器 (PADR/PBDR/PCDR)

寄存器	地址	R/W	描述	复位后的值
<b>PADR</b>	0x51	R/W	PA 下拉控制寄存器	00000000
<b>PBDR</b>	0x56	R/W	PB 下拉控制寄存器	00000000
<b>PCDR</b>	0x5B	R/W	PC 下拉控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>PADR[7:0]</b> x=A,B,C							

位	描述	
[7:0]	<b>PxDR</b> x=A,B,C	Px 下拉电阻控制位 0: 禁止下拉电阻 1: 使能下拉电阻

### 10.2.5 PA/PB/PC 开漏功能控制寄存器 (PAOD /PBOD /PCOD)

寄存器	地址	R/W	描述	复位后的值
<b>PAOD</b>	0x52	R/W	PA 开漏功能控制寄存器	00000000
<b>PBOD</b>	0x57	R/W	PB 开漏功能控制寄存器	00000000
<b>PCOD</b>	0x5C	R/W	PC 开漏功能控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>PxOD[7:0]</b> x=A,B,C							

位	描述	
[7:0]	<b>PxOD</b> x=A,B,C	Px 开漏功能控制位 0: 禁止开漏 1: 使能开漏

### 10.2.6 驱动电流配置寄存器 (PAIO)

寄存器	地址	R/W	描述	复位后的值
<b>PAIO</b>	0x5D	R/W	PA 驱动电流配置寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PBIO1	PBIO0	PBIOH1	PBIOH0	PAIOL1	PAIOL0	PAIOH1	PAIOH0

位	描述	
7	<b>PBIO1</b>	<b>PB 高四位 IOL 设置</b> 0: 设置 PB 高四位 IOL 等级 0 (最大) 20mA@Vpin=0.8V@VDD=5V 1: 设置 PB 高四位 IOL 等级 2 60mA@Vpin=0.8V@VDD=5V
6	<b>PBIO0</b>	<b>PB 低四位 IOL 设置</b> 0: 设置 PB 高四位 IOL 等级 0 (最大) 20mA@Vpin=0.8V@VDD=5V 1: 设置 PB 高四位 IOL 等级 2 60mA@Vpin=0.8V@VDD=5V
5	<b>PBIOH1</b>	<b>PB 高四位 IOL 设置</b> 0: 设置 PB 高四位 IOL 等级 0 (最大) 20mA@Vpin=0.8V@VDD=5V 1: 设置 PB 高四位 IOL 等级 2 60mA@Vpin=0.8V@VDD=5V

4	<b>PBIOH0</b>	<b>PB 低四位 IOH 设置</b> 0: 设置 PB 低四位 IOH 等级 0 (最大) 20mA@Vpin=4.3V@VDD=5V 1: 设置 PB 低四位 IOH 等级 2 5mA@Vpin=4.3V@VDD=5V
3	<b>PAIOL1</b>	<b>PA 高四位 IOL 设置</b> 0: 设置 PA 高四位 IOL 等级 0 (最大) 20mA@Vpin=0.8V@VDD=5V 1: 设置 PA 高四位 IOL 等级 2 60mA@Vpin=0.8V@VDD=5V
2	<b>PAIOL0</b>	<b>PA 低四位 IOL 设置</b> 0: 设置 PA 高四位 IOL 等级 0 (最大) 20mA@Vpin=0.8V@VDD=5V 1: 设置 PA 高四位 IOL 等级 2 60mA@Vpin=0.8V@VDD=5V
1	<b>PAIOH1</b>	<b>PA 高四位 IOL 设置</b> 0: 设置 PA 高四位 IOL 等级 0 (最大) 20mA@Vpin=0.8V@VDD=5V 1: 设置 PA 高四位 IOL 等级 2 60mA@Vpin=0.8V@VDD=5V
0	<b>PAIOH0</b>	<b>PA 低四位 IOH 设置</b> 0: 设置 PA 低四位 IOH 等级 0 (最大) 20mA@Vpin=4.3V@VDD=5V 1: 设置 PA 低四位 IOH 等级 2 5mA@Vpin=4.3V@VDD=5V

### 10.2.1 驱动电流配置寄存器 (PBIO)

寄存器	地址	R/W	描述	复位后的值
<b>PBIO</b>	0x5E	R/W	PB 驱动电流配置寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	IOVTS[2:0]			PB4HDR	PB3HDR	PCIOLO	PCIOH0

位	描述	
7	保留	保留



6:4	<b>IOVTS[2:0]</b>	<b>IO 斯密特阈值选择 IOVTS[x]</b> 0: $V_{IL}=0.2V_{DD}$ $V_{IH}=0.4V_{DD}$ 1: $V_{IL}=0.3V_{DD}$ $V_{IH}=0.7V_{DD}$ 作为外部硬件复位口时强制为 $V_{IL}=0.3V_{DD}$ $V_{IH}=0.7V_{DD}$ 其中, IOVTS[0]控制 PA 口; IOVTS[1]控制 PB 口; IOVTS[2]控制 PC 口。
3	<b>PB4HDR</b>	<b>PB4 IOH 大驱动配置</b> 0: 5mA/20mA (pb_ioh0=0 或 1 有效) 1: 40mA (pb_ioh1=1 无效)
2	<b>PB3HDR</b>	<b>PB3 IOH 大驱动配置</b> 0: 5mA/20mA (pb_ioh0=0 或 1 有效) 1: 40mA (pb_ioh0=1 无效)
1	<b>PCIOLO</b>	<b>PC 低四位 IOL 设置</b> 0: 设置 PC 高四位 IOL 等级 0 (最大) 20mA@Vpin=0.8V@VDD=5V 1: 设置 PC 高四位 IOL 等级 2 60mA@Vpin=0.8V@VDD=5V
0	<b>PCIOH0</b>	<b>PC 低四位 IOH 设置</b> 0: 设置 PC 低四位 IOH 等级 0 (最大) 20mA@Vpin=4.3V@VDD=5V 1: 设置 PC 低四位 IOH 等级 2 5mA@Vpin=4.3V@VDD=5V

## 10.3 IO 翻转换醒

芯片的所有 IO 口都可以配置成唤醒功能, 当使能了此唤醒功能后, 只要 IO 口的电平出现变化, 都可以唤醒 IDLE 或者 STOP, CPU 醒来后, 会继续往下运行程序。

### 10.3.1 GPIO 翻转换醒使能寄存器 (PWKEN<sub>x</sub>)

寄存器	地址	R/W	描述	复位后的值
<b>PWKEN0</b>	0x21	R/W	GPIO 翻转换醒使能寄存器 0	00000000
<b>PWKEN1</b>	0x22	R/W	GPIO 翻转换醒使能寄存器 1	00000000
<b>PWKEN2</b>	0x23	R/W	GPIO 翻转换醒使能寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>Px[7:0]</b> x=A,B,C							

位	描述	
[7:0]	<b>Px[7:0]</b> <b>x=A,B,C</b>	<b>Px.y (x=A,B,C y=0~7) 翻转换醒使能位</b> 0: 禁止 Px.y 翻转换醒 1: 使能 Px.y 翻转换醒

RealMagiC 锐盟

## 11 Timer0

### 11.1 特性

TIMER0 为 8 位 Basic Timer。可以工作于定时器模式、计数器模式。

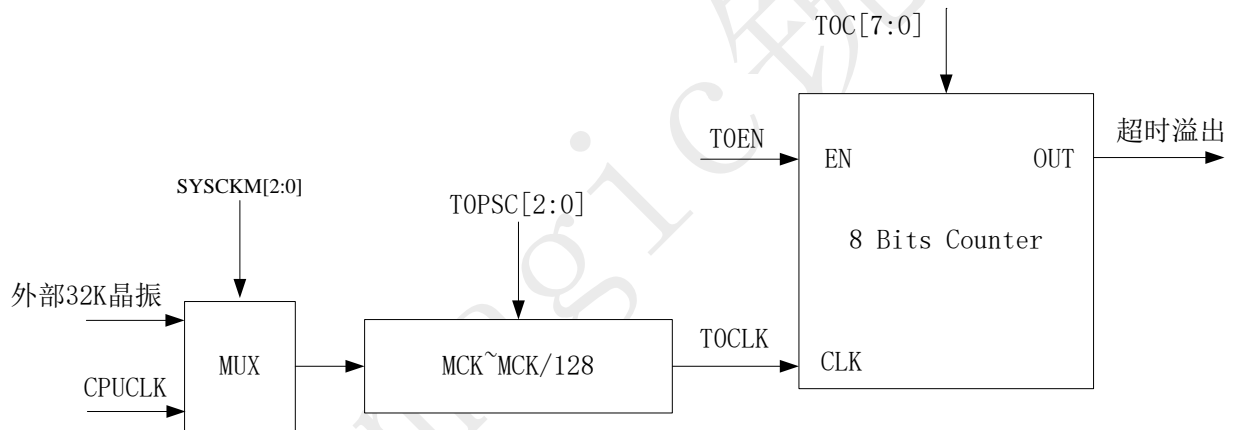
工作于定时器模式时：定时时间由 TOPSC[2:0]配置预分频比来控制，定时器从 T0C 的配置值（默认值为 0x00）开始向上计数，计数到 TOR 时产生溢出中断。

工作于计数器模式时：根据 TOEG 的配置值，对端口 PB.7 的输入信号的上升沿或下降沿进行计数。计数器从 0 开始向上计数，计数到 TOR 值时产生溢出中断。

TIMER0 中断服务程序入口地址为 008H。

TIMER0 对应的中断标志位(TOF)为中断标志寄存器 IF0 (SFR: 0x06)的 bit1。

TIMER0 对应的中断使能位(ETO)为中断使能寄存器 IEN0 (SFR: 0x07)的 bit1。



定时/计数模块的功能框图

### 11.2 寄存器

#### 11.2.1 TIMER0 配置寄存器(T0M)

寄存器	地址	R/W	描述	复位后的值
T0M	0x17	R/W	TIMER0 配置寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
TOCLR	T0M	保留	TOEG	TOPSC[2:0]			TOEN

位	描述
[7]	TOCLR 计数寄存器清 0 使能位，置 1，T0C 清 0，TOCLR 硬件自动清 0

[6]	<b>T0M</b>	定时/计数器模式选择位 0: 定时器模式 1: 计数器模式, PB7 为计数时钟输入, 上升沿计数
[5]	保留	保留
[4]	<b>T0EG</b>	计数器输入有效沿选择位 0: 上升沿计数 1: 下降沿计数
[3:1]	<b>T0PSC[2:0]</b>	定时/计数器 0 时钟分频 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128  RTC 模式下, T0R 不为 0 时, 此三位不起作用, RTC 根据 T0R 设置的值为溢出值来计数定时; T0R 为 0 时, RTC 定时时间选择 x00: RTC 定时时间为 0.25s x01: RTC 定时时间为 0.125s x10: RTC 定时时间为 0.5s x11: RTC 定时时间为 1s
[0]	<b>T0EN</b>	定时器 0 使能位 0: 禁止定时器 0/RTC 1: 使能定时器 0/RTC

### 11.2.2 TIMER0 计数寄存器(T0C)

寄存器	地址	R/W	描述	复位后的值
<b>T0C</b>	0x18	R/W	TIMER0 计数寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>T0C[7:0]</b>							

位	描述
[7:0]	<b>T0C[7:0]</b> TIMER0 计数寄存器

### 11.2.3 TIMER0 重载寄存器(T0R)

寄存器	地址	R/W	描述	复位后的值
<b>T0R</b>	0x19	R/W	TIMER0 重载寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>T0R[7:0]</b>							

位	描述	
[7:0]	<b>T0R[7:0]</b>	TIMER0 重载寄存器

## 11.3 定时器

定时器模块的输入为 **CLKCPU**，受烧录选项的时钟分频影响。在定时器模块集成了一个分频器，分频的时钟 **T0CLK** 作为 8 bits 计数器的输入时钟。当用户设置了定时器模块的使能标志，8 bits 计数器将启动，将会从 0 递增到 **T0R**。当用户开启定时器 **T0** 中断。当定时超时发生时，中断标志位会置 1，程序计数器会跳转到 08h 以执行中断服务程序。

操作：

- 1) 设置 **T0CLK**，选择定时器/计数器时钟。
- 2) 设置 **T0R**，选择定时器溢出值。
- 3) 设置 **T0M**，设置为定时器模式。
- 4) 使能定时器中断：**ET0** 与 **EA**。
- 5) 使能定时器：**TOEN**。
- 6) 当定时超时发生时，程序计数器会跳转到 008H。

**定时器 0 溢出时间= (T0R[7:0]+1) /T0CLK。**

## 11.4 计数器

计数器时钟模块的输入为 **PB.7** 口。分频器不对计数模式产生影响。当用户设置了定时/计数器模块的使能标志，8 bits 计数器将启动，将会从 0 递增到 **T0R**。用户需要设置 **T0R**（定时器模块中断信号选择器）以选择定时超时中断信号。当定时超时发生时，中断标志位会自设置，程序计数器会跳转到 08h 以执行中断服务程序。

操作：

- 1) 设置 **T0R**，选择计数器溢出值。计数次数=**T0R+1**；
- 2) 设置 **T0M**，设置为计数器模式。
- 3) 设置 **T0EG**，配置计数器触发沿。
- 4) 使能计数器中断：**ET0** 与 **EA**。

- 5) 使能计数器: TOEN。
- 6) 当计数溢出时, 程序计数器会跳转到 008H。  
计数次数=  $T0R[7:0]+1$ ;

## 11.5 RTC 模式

当通过工具将 Timer0 配置为 RTC 模式时, 则为 RTC 功能, 否则为 Timer0 定时器/计数器功能。RTC 模式下定时器 0 时钟为外部 32768HZ 时钟。

当配置为 RTC 模式后, 使能 TOEN 就可以开启 RTC 模块的工作。可通过配置 TOPSC 和 TOR 来选择 RTC 的定时时间。当 TOR 不为 0 时, TOPSC 不起作用, RTC 根据 TOR 设置的值为溢出值来计数定时; 当 TOR 为 0 时, RTC 定时时间则根据 TOPSC 来选择, 具体可见 TOPSC 的描述。

RTC 定时溢出时, 会进入中断, 此中断使能位和中断标志位同定时器一样, 具体可查看中断系统章节。

RTC 定时溢出时, 可唤醒 IDLE 或者 STOP 模式。

## 12 Timer1

### 12.1 特性

TIMER1 为 16 位 Basic Timer。可以工作于定时器模式、计数器模式、PWM 模式。TIMER1 有 4 组互补输出的 PWM。可通过配置周期，占空比，死区等参数，得到不同的 PWM 输出。

TIMER1 模块的输入时钟为 16MHz 的 PCK，在使用此模块时，需要先设置 PCKEN=1(OSCM[4])开启此 16MHz 的时钟。

### 12.2 寄存器

#### 12.2.1 TIMER1 配置寄存器 0(T1M0)

寄存器	地址	R/W	描述	复位后的值
T1M0	0x1B	R/W	TIMER1 配置寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1CLR	T1M	保留	T1EG	T1PSC[2:0]			T1EN

位	描述
[7]	<b>T1CLR</b> 计数器寄存器清 0 使能位，置 1，T1C 清 0，T1CLR 硬件自动清 0
[6]	<b>T1M</b> 定时/计数器模式选择位 0: 定时器模式 1: 计数器模式，PB4 为计数时钟输入，上升沿计数
[5]	保留
[4]	<b>T1EG</b> 计数器输入有效沿选择位 0: 上升沿计数 1: 下降沿计数
[3:1]	<b>T1PSC[2:0]</b> 定时/计数器 1 时钟分频 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128

[0]	<b>T1EN</b>	<b>定时器 1 使能位</b> 0: 禁止定时器 1 1: 使能定时器 1
-----	-------------	--

### 12.2.2 TIMER1 配置寄存器 1(T1M1)

寄存器	地址	R/W	描述	复位后的值
<b>T1M1</b>	0x1C	R/W	TIMER1 配置寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>T1PWM1NS</b>	<b>T1PWM1S</b>	<b>T1PWM0NS</b>	<b>T1PWM0S</b>	<b>T1PWM1NEN</b>	<b>T1PWM1EN</b>	<b>T1PWM0NEN</b>	<b>T1PWM0EN</b>

位	描述
[7]	<b>T1PWM1NS</b> <b>T1PWM1 补波输出有效电平选择位</b> 0: T1PWM1N 初始电平为 1, 占空比期间为低电平 1: T1PWM1N 初始电平为 0, 占空比期间为高电平
[6]	<b>T1PWM1S</b> <b>T1PWM1 输出有效电平选择位</b> 0: T1PWM1 初始电平为 0, 占空比期间为高电平 1: T1PWM1 初始电平为 1, 占空比期间为低电平
[5]	<b>T1PWM0NS</b> <b>T1PWM0 补波输出有效电平选择位</b> 0: T1PWM0N 初始电平为 1, 占空比期间为低电平 1: T1PWM0N 初始电平为 0, 占空比期间为高电平
[4]	<b>T1PWM0S</b> <b>T1PWM0 输出有效电平选择位</b> 0: T1PWM0 初始电平为 0, 占空比期间为高电平 1: T1PWM0 初始电平为 1, 占空比期间为低电平
[3]	<b>T1PWM1NEN</b> <b>T1PWM1 补波使能位</b> 0: 关闭 T1PWM1 补波输出 1: 使能 T1PWM1 补波输出
[2]	<b>T1PWM1EN</b> <b>T1PWM1 使能位</b> 0: 关闭 T1PWM1 输出 1: 使能 T1PWM1 输出
[1]	<b>T1PWM0NEN</b> <b>T1PWM0 补波使能位</b> 0: 关闭 T1PWM0 补波输出 1: 使能 T1PWM0 补波输出
[0]	<b>T1PWM0EN</b> <b>T1PWM0 使能位</b> 0: 关闭 T1PWM0 输出 1: 使能 T1PWM0 输出



### 12.2.3 TIMER1 重载寄存器高 8 位(T1RH)

寄存器	地址	R/W	描述	复位后的值
<b>T1RH</b>	0x1F	R/W	TIMER1 重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1R[15:8]							

位	描述	
[7:0]	<b>T1R[15:8]</b>	TIMER1 重载寄存器高 8 位

### 12.2.4 TIMER1 重载寄存器低 8 位(T1RL)

寄存器	地址	R/W	描述	复位后的值
<b>T1RL</b>	0x20	R/W	TIMER1 重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1R[7:0]							

位	描述	
[7:0]	<b>T1R[7:0]</b>	TIMER1 重载寄存器低 8 位

### 12.2.5 T1PWM0 占空比重载寄存器高 8 位(T1D0H)

寄存器	地址	R/W	描述	复位后的值
<b>T1D0H</b>	0x21	R/W	T1PWM0 占空比重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1D0[15:8]							

位	描述	
[7:0]	<b>T1D0[15:8]</b>	T1PWM0 占空比重载寄存器高 8 位

### 12.2.6 T1PWM0 占空比重载寄存器低 8 位(T1D0L)

寄存器	地址	R/W	描述	复位后的值
-----	----	-----	----	-------

<b>T1D0L</b>	0x22	R/W	T1PWM0 占空比重载寄存器低 8 位	00000000
--------------	------	-----	----------------------	----------

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1D0[7:0]							

位	描述	
[7:0]	<b>T1D0[7:0]</b>	T1PWM0 占空比重载寄存器低 8 位

### 12.2.7 T1PWM1 占空比重载寄存器高 8 位(T1D1H)

寄存器	地址	R/W	描述	复位后的值
<b>T1D1H</b>	0x23	R/W	T1PWM1 占空比重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1D1[15:8]							

位	描述	
[7:0]	<b>T1D1[15:8]</b>	T1PWM1 占空比重载寄存器高 8 位

### 12.2.8 T1PWM1 占空比重载寄存器低 8 位(T1D1L)

寄存器	地址	R/W	描述	复位后的值
<b>T1D1L</b>	0x24	R/W	T1PWM1 占空比重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1D1[7:0]							

位	描述	
[7:0]	<b>T1D1[7:0]</b>	T1PWM1 占空比重载寄存器低 8 位

### 12.2.9 T1PWM0 死区配置寄存器高 8 位(T1PWM0DZH)

寄存器	地址	R/W	描述	复位后的值
<b>T1PWM0DZH</b>	0x25	R/W	T1PWM0 死区配置寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1PWM0DZ[15:8]							

位	描述	
[7:0]	T1PWM0DZ[15:8]	T1PWM0 死区配置寄存器高 8 位

### 12.2.10 T1PWM0 死区配置寄存器低 8 位(T1PWM0DZL)

寄存器	地址	R/W	描述	复位后的值
T1PWM0DZL	0x26	R/W	T1PWM0 死区配置寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1PWM0DZ[7:0]							

位	描述	
[7:0]	T1PWM0DZ[7:0]	T1PWM0 死区配置寄存器低 8 位

### 12.2.11 T1PWM1 死区配置寄存器高 8 位(T1PWM1DZH)

寄存器	地址	R/W	描述	复位后的值
T1PWM1DZH	0x27	R/W	T1PWM1 死区配置寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1PWM1DZ[15:8]							

位	描述	
[7:0]	T1PWM1DZ[15:8]	T1PWM1 死区配置寄存器高 8 位

### 12.2.12 T1PWM1 死区配置寄存器低 8 位(T1PWM1DZL)

寄存器	地址	R/W	描述	复位后的值
T1PWM1DZL	0x28	R/W	T1PWM1 死区配置寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T1PWM1DZ[7:0]							

位	描述	
[7:0]	T1PWM1DZ[7:0]	T1PWM1 死区配置寄存器低 8 位

### 12.2.13 TIMER1 配置寄存器 2(T1M2)

寄存器	地址	R/W	描述	复位后的值
<b>T1M2</b>	0x29	R/W	TIMER1 配置寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>PWM3NS</b>	<b>PWM3S</b>	<b>PWM2NS</b>	<b>PWM2S</b>	<b>PWM3NEN</b>	<b>PWM3EN</b>	<b>PWM2NEN</b>	<b>PWM2EN</b>

位	描述	
[7]	<b>PWM3NS</b>	<b>T1PWM3 补波输出有效电平选择位</b> 0: T1PWM3N 初始电平为 1, 占空比期间为低电平 1: T1PWM3N 初始电平为 0, 占空比期间为高电平
[6]	<b>PWM3S</b>	<b>T1PWM3 输出有效电平选择位</b> 0: T1PWM3 初始电平为 0, 占空比期间为高电平 1: T1PWM3 初始电平为 1, 占空比期间为低电平
[5]	<b>PWM2NS</b>	<b>T1PWM2 补波输出有效电平选择位</b> 0: T1PWM2N 初始电平为 1, 占空比期间为低电平 1: T1PWM2N 初始电平为 0, 占空比期间为高电平
[4]	<b>PWM2S</b>	<b>T1PWM2 输出有效电平选择位</b> 0: T1PWM2 初始电平为 0, 占空比期间为高电平 1: T1PWM2 初始电平为 1, 占空比期间为低电平
[3]	<b>PWM3NEN</b>	<b>T1PWM3 补波使能位</b> 0: 关闭 T1PWM3 补波输出 1: 使能 T1PWM3 补波输出
[2]	<b>PWM3EN</b>	<b>T1PWM3 使能位</b> 0: 关闭 T1PWM3 输出 1: 使能 T1PWM3 输出
[1]	<b>PWM2NEN</b>	<b>T1PWM2 补波使能位</b> 0: 关闭 T1PWM2 补波输出 1: 使能 T1PWM2 补波输出
[0]	<b>PWM2EN</b>	<b>T1PWM2 使能位</b> 0: 关闭 T1PWM2 输出 1: 使能 T1PWM2 输出

### 12.2.14 T1PWM2 占空比重载寄存器高 8 位(T1D2H)

寄存器	地址	R/W	描述	复位后的值
<b>T1D2H</b>	0x2A	R/W	T1PWM2 占空比重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
------	------	------	------	------	------	------	------

<b>T1D2[15:8]</b>
-------------------

位	描述	
[7:0]	<b>T1D2[15:8]</b>	T1PWM2 占空比重载寄存器高 8 位

### 12.2.15 T1PWM2 占空比重载寄存器低 8 位(T1D2L)

寄存器	地址	R/W	描述	复位后的值
<b>T1D2L</b>	0x2B	R/W	T1PWM2 占空比重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>T1D2[7:0]</b>							

位	描述	
[7:0]	<b>T1D2[7:0]</b>	T1PWM2 占空比重载寄存器低 8 位

### 12.2.16 T1PWM3 占空比重载寄存器高 8 位(T1D3H)

寄存器	地址	R/W	描述	复位后的值
<b>T1D3H</b>	0x2C	R/W	T1PWM3 占空比重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>T1D3[15:8]</b>							

位	描述	
[7:0]	<b>T1D3[15:8]</b>	T1PWM3 占空比重载寄存器高 8 位

### 12.2.17 T1PWM3 占空比重载寄存器低 8 位(T1D3L)

寄存器	地址	R/W	描述	复位后的值
<b>T1D3L</b>	0x2D	R/W	T1PWM3 占空比重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>T1D3[7:0]</b>							

位	描述	
[7:0]	<b>T1D3[7:0]</b>	T1PWM3 占空比重载寄存器低 8 位

## 12.3 定时器

在定时器模块集成了一个分频器，分频的时钟 T1CLK 作为 16 bits 计数器的输入时钟。当用户设置了定时器模块的使能标志，16 bits 计数器将启动，将会从 0 递增到 T1R。用户需要设置 T1R（定时器模块中断信号选择器）以选择定时超时中断信号。当定时超时发生时，中断标志位会自设置，程序计数器会跳转到 08h 以执行中断服务程序。

操作：

- 1) 设置 T1CLK，选择定时器/计数器时钟。
- 2) 设置 T1R，选择定时器溢出值。
- 3) 设置 T1M，配置为定时器模式。
- 4) 使能定时器 1 中断：ET1 与 EA。
- 5) 使能定时器 1：T1EN。
- 6) 当定时超时发生时，程序计数器会跳转到 008H。

**定时器 1 溢出时间= (T1R[15:0]+1) /T1CLK.**

## 12.4 计数器

工作于 16 位重加载计数器模式时，根据 T1EG 的配置值，对端口 PB.4 的输入信号的上升沿或下降沿进行计数。定时器分频对计数模式没影响。TIMER1 开始工作于 16 位重加载计数器模式，计数器由 0 开始计数到 T1R[15:0]溢出值产生中断。

操作：

- 1) 设置 T1R，选择定时器溢出值。
- 2) 设置 T1M，配置为计数器模式。
- 3) 设置 T1EG，选择上升沿还是下降沿进行计数。
- 4) 使能定时器 1 中断：ET1 与 EA。
- 5) 使能定时器 1：T1EN。
- 6) 当定时超时发生时，程序计数器会跳转到 008H。

**计数次数= T1R[15:0]+1;**

## 12.5 PWM

定时器 1 有 4 路互补的 PWM 输出，4 路 PWM 的周期大小通过配置 T1PSC[2:0]和 T1R[15:0]来实现，PWM0 占空比由 T1D0[15:0]来调整，PWM0NEN 配置 PWM0 互补输出，T1PWM0DZ[15:0]配置互补输出的死区大小。PWM1/2/3 占空比由 T1D1/2/3[15:0]来调整，PWM1/2/3NEN 配置 PWM1/2/3 互补输出，T1PWM1DZ[15:0]配置 PWM1/2/3 互补输出的死区大小。

### 12.5.1 PWM 输出

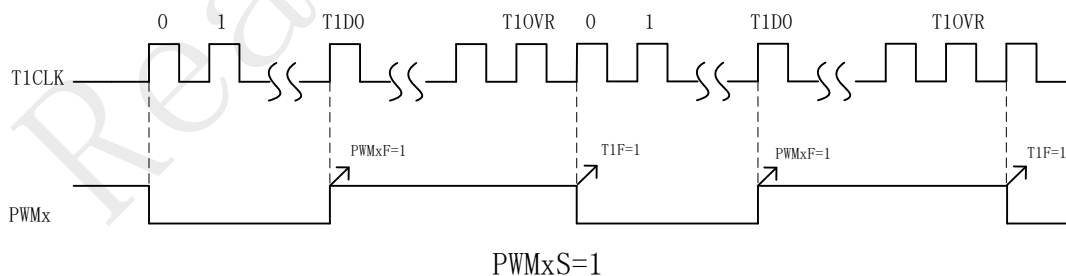
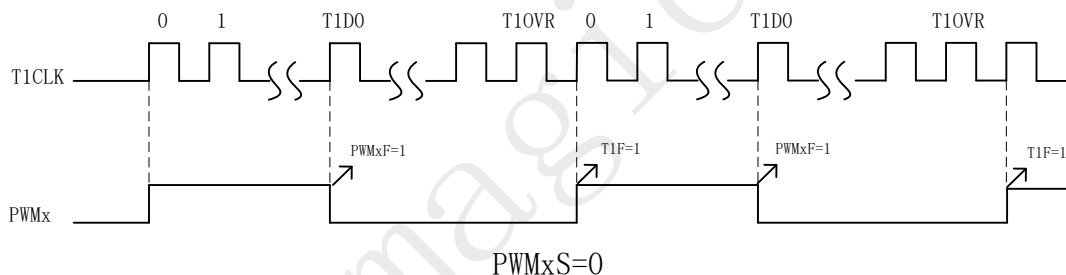
PWM 的周期 =  $(T1R[15:0]+1) / T1CLK$ .

PWM<sub>x</sub> 的有效电平宽度比 =  $T1Dx[15:0] / (T1R[15:0]+1)$

操作:

- 1) 设置 T1PSC[2:0], 配置 PWM 时钟。
- 2) 设置 T1R, 设置 PWM 周期。
- 3) 设置 T1Dx, 设置 PWM<sub>x</sub> 有效电平宽度。
- 4) 设置 PWM<sub>x</sub>S, 设置 PWM<sub>x</sub> 输出有效电平。
- 5) 设置寄存器标志位: PWM<sub>x</sub>EN, 使能 PWM<sub>x</sub> 输出。
- 6) 设置寄存器标志位: T1EN, 使能定时器。
- 7) 当 PWM<sub>x</sub> 内部计数器计数等于 T1Dx 时, 占空比中断标志会置位, 如此时占空比中断使能打开, 程序计数器会跳转到 008H。
- 8) 当 T1C 计数等于 T1R 时, 中断标志会置位。定时器 1 中断使能位如打开的话, 程序计数器会跳转到 008H。

周期为 T1R+1, PWM0 高电平脉宽为 T1D0。PWM0 波形输出如下:



## 12.5.2 PWM 互补输出与死区

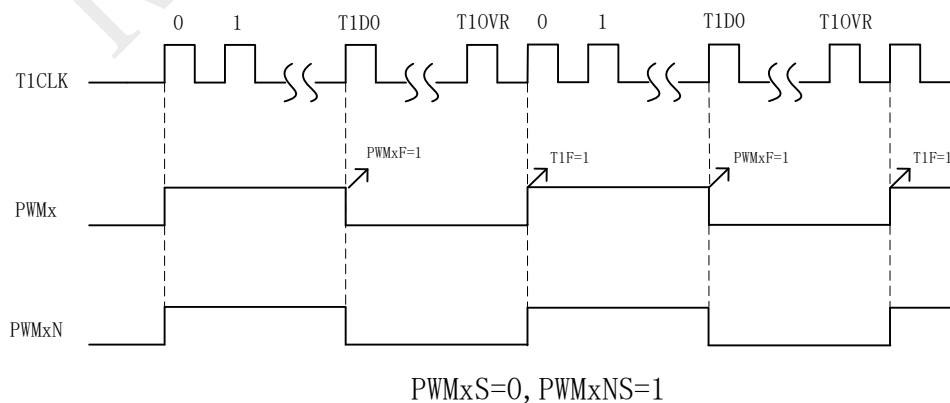
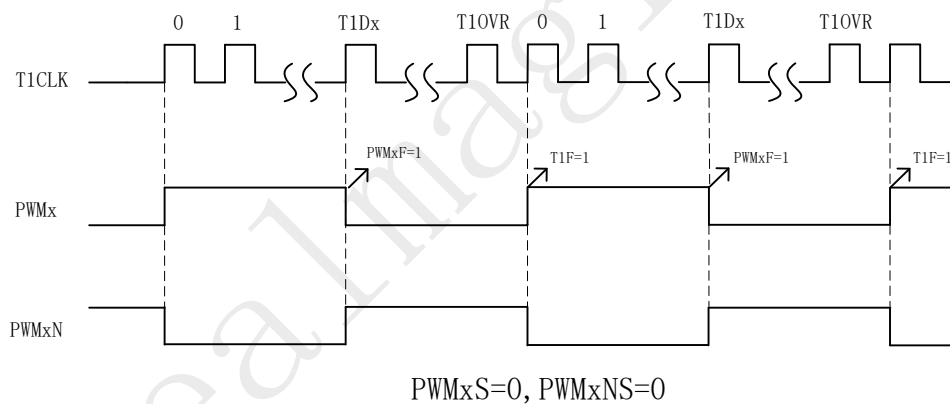
PWM 的周期 =  $(T1R[15:0]+1) / T1CLK$ .

PWMx 的有效电平宽度比 =  $T1Dx[15:0] / (T1R[15:0]+1)$

操作:

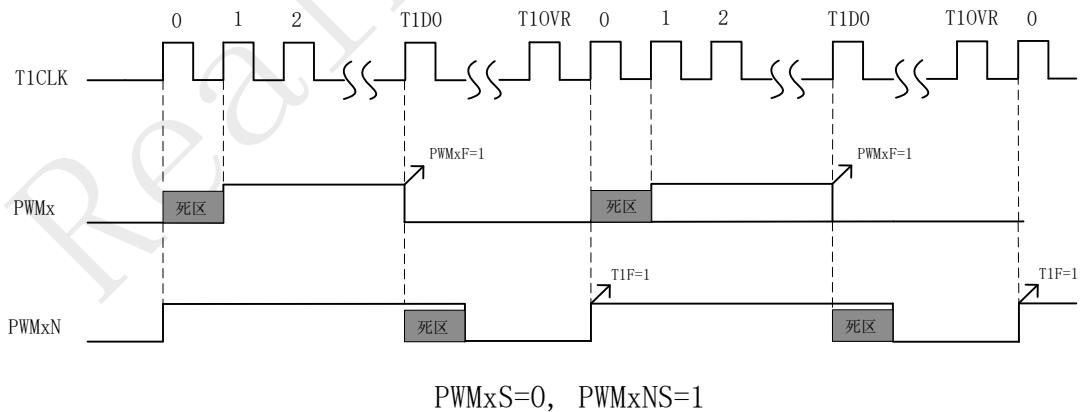
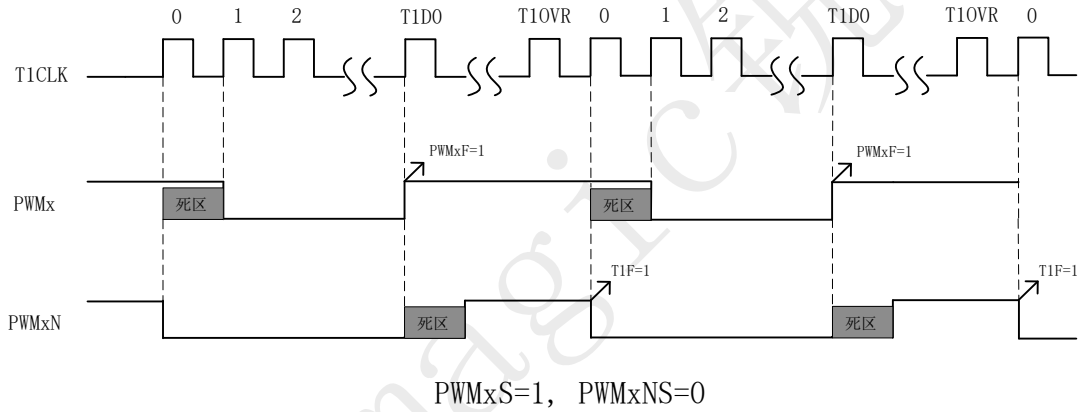
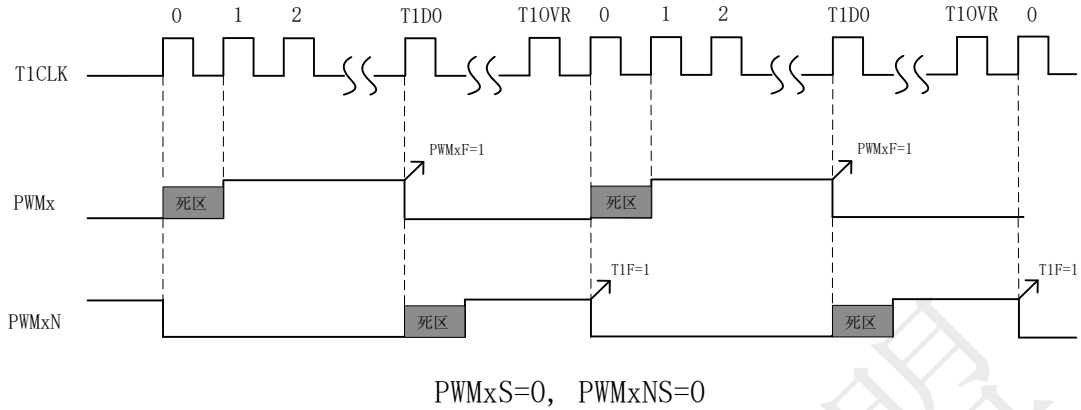
- 1) 设置 T1PSC[2:0], 配置 PWM 时钟。
- 2) 设置 T1R, 设置 PWM 周期。
- 3) 设置 T1Dx, 设置 PWMx 有效电平宽度。
- 4) 设置 T1PWMxDZ[15:0], 设置 PWMx 死区。
- 5) 设置 PWMxS 和 PWMxNS, 设置 PWMx 和 PWMx 互补输出有效电平。
- 6) 设置寄存器标志位: PWMxEN 和 PWMxNEN, 使能 PWMx 和 PWMx 互补输出。
- 7) 设置寄存器标志位: T1EN, 使能定时器。
- 8) 当 PWM 内部计数器计数等于 T1Dx 时, 占空比中断标志会置位, 如此时占空比中断使能打开, 程序计数器会跳转到 008H。
- 9) 当 TIC 计数等于 T1R 时, 中断标志会置位。定时器 1 中断使能位如打开的话, 程序计数器会跳转到 008H。

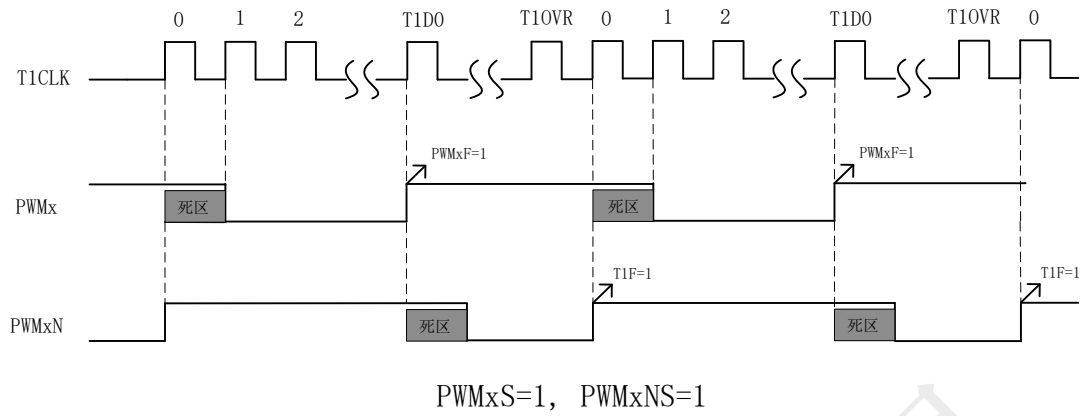
周期为  $T1R+1$ , PWMx 高电平脉宽为 T1Dx。T1PWMxDZ=0, 即死区为 0。PWMx 波形输出如下:





周期为  $T1R+1$ ， $PWM_x$  高电平脉宽为  $T1D_x$ ，死区为  $T1PWM_xDZ=1$ 。 $PWM_x$  波形输出如下：





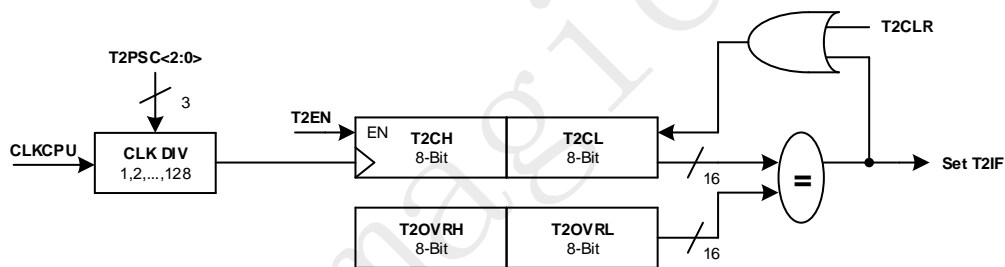
RealMagiC 锐盟

## 13 Timer2

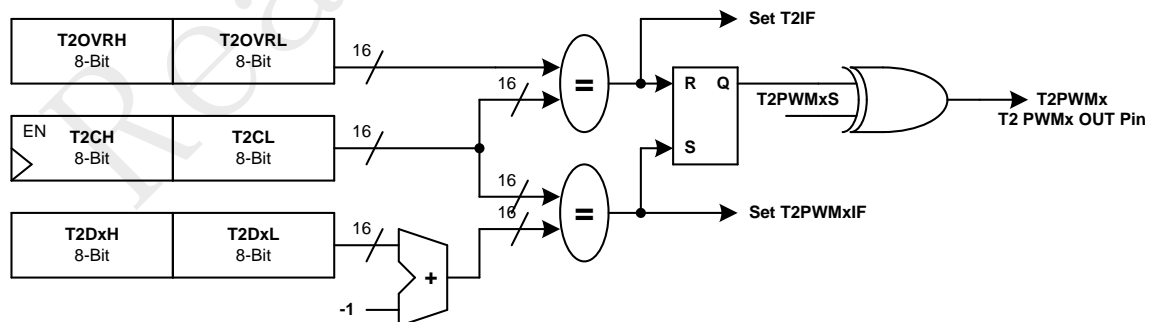
### 13.1 特性

定时器 2 模块的时钟来源于 CLKCPU 时钟模块。定时器模块前端集成了一个分频器，分频的时钟 T2CLK 作为 16 bits 计数器的输入时钟。当用户设置了定时器模块的使能标志，16 bits 计数器将启动，将会从 0 递增到 T2R。用户需要设置 T2R（定时器模块中断信号选择器）以选择定时超时中断信号。当定时超时发生时，中断标志位会自设置，程序计数器会跳转到 08h 以执行中断服务程序。内部定时计数器为 16 位宽。同样，T2 还集成了一些辅助模块，通过灵活的配置，除了作为普通的定时计数外，也可用作 T2PWM 功能和捕获功能。

### 13.2 工作原理



Timer2 模块框图



T2PWM 原理框图

## 13.3 寄存器

### 13.3.1 TIMER2 配置寄存器 0(T2M0)

寄存器	地址	R/W	描述	复位后的值
T2M0	0x2F	R/W	TIMER2 配置寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2CLR	保留	保留		T2PSC[2:0]			T2EN

位	描述	
[7]	T2CLR	计数寄存器清 0 使能位, 置 1, T2C 清 0, T2CLR 硬件自动清 0
[6:4]	保留	保留
[3:1]	T2PSC[2:0]	定时器 2 时钟分频 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128
[0]	T2EN	定时器 2 使能位 0: 禁止定时器 2 1: 使能定时器 2

### 13.3.2 TIMER2 配置寄存器 1(T2M1)

寄存器	地址	R/W	描述	复位后的值
T2M1	0x30	R/W	TIMER2 配置寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	T2PWM1S	保留	T2PWM0S	保留	T2PWM1EN	保留	T2PWM0EN

位	描述	
[7]	保留	保留
[6]	T2PWM1S	T2PWM1 输出有效电平选择位 0: T2PWM1 初始电平为 0, 占空比期间为高电平 1: T2PWM1 初始电平为 1, 占空比期间为低电平

[5]	保留	保留
[4]	<b>T2PWM0S</b>	<b>T2PWM1 输出有效电平选择位</b> 0: T2PWM0 初始电平为 0, 占空比期间为高电平 1: T2PWM0 初始电平为 1, 占空比期间为低电平
[3]	保留	保留
[2]	<b>T2PWM1EN</b>	<b>T2PWM1 使能位</b> 0: 关闭 T2PWM1 输出 1: 使能 T2PWM1 输出
[1]	保留	保留
[0]	<b>T2PWM0EN</b>	<b>T2PWM0 使能位</b> 0: 关闭 T2PWM0 输出 1: 使能 T2PWM0 输出

### 13.3.3 T2PWM0 占空比重载寄存器高 8 位(T2D0H)

寄存器	地址	R/W	描述	复位后的值
<b>T2D0H</b>	0x31	R/W	T2PWM0 占空比重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2D0[15:8]							

位	描述
[7:0]	<b>T2D0[15:8]</b> T2PWM0 占空比重载寄存器高 8 位

### 13.3.4 T2PWM0 占空比重载寄存器低 8 位(T2D0L)

寄存器	地址	R/W	描述	复位后的值
<b>T2D0L</b>	0x32	R/W	T2PWM0 占空比重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2D0[7:0]							

位	描述
[7:0]	<b>T2D0[7:0]</b> T2PWM0 占空比重载寄存器低 8 位

### 13.3.5 T1PWM2 占空比重载寄存器高 8 位(T2D1H)

寄存器	地址	R/W	描述	复位后的值
<b>T2D1H</b>	0x33	R/W	T2PWM1 占空比重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2D1[15:8]							

位	描述	
[7:0]	T2D1[15:8]	T2PWM1 占空比重载寄存器高 8 位

### 13.3.6 T2PWM1 占空比重载寄存器低 8 位(T2D1L)

寄存器	地址	R/W	描述	复位后的值
<b>T2D1L</b>	0x34	R/W	T2PWM1 占空比重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2D1[7:0]							

位	描述	
[7:0]	T2D1[7:0]	T2PWM1 占空比重载寄存器低 8 位

### 13.3.7 TIMER2 计数寄存器高 8 位(T2CH)

寄存器	地址	R/W	描述	复位后的值
<b>T2CH</b>	0x35	R/W	TIMER2 计数寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2C[15:8]							

位	描述	
[7:0]	T2C[15:8]	TIMER2 计数寄存器高 8 位

### 13.3.8 TIMER2 计数寄存器低 8 位(T2CL)

寄存器	地址	R/W	描述	复位后的值
-----	----	-----	----	-------

<b>T2CL</b>	0x36	R/W	TIMER2 计数寄存器低 8 位	00000000
-------------	------	-----	-------------------	----------

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2C[7:0]							

位	描述	
[7:0]	T2C[7:0]	TIMER2 计数寄存器低 8 位

### 13.3.9 TIMER2 重载寄存器高 8 位(T2RH)

寄存器	地址	R/W	描述	复位后的值
<b>T2RH</b>	0x37	R/W	TIMER2 重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2R[15:8]							

位	描述	
[7:0]	T2R[15:8]	TIMER2 重载寄存器高 8 位

### 13.3.10 TIMER2 重载寄存器低 8 位(T2RL)

寄存器	地址	R/W	描述	复位后的值
<b>T2RL</b>	0x38	R/W	TIMER2 重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2R[7:0]							

位	描述	
[7:0]	T2R[7:0]	TIMER2 重载寄存器低 8 位

### 13.3.11 TIMER2 捕获中断寄存器 (T2CAPINT)

寄存器	地址	R/W	描述	复位后的值
<b>T2CAPINT</b>	0x39	R/W	TIMER2 捕获中断寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留		T2C1F	T2C0F	保留		ET2C1	ET2C0

位	描述	
[7:6]	保留	保留
[5]	<b>T2C1F</b>	T2CAP1 中断标志位, 高有效, 需软件清零
[4]	<b>T2C0F</b>	T2CAP0 中断标志位, 高有效, 需软件清零
[3:2]	保留	保留
[1]	<b>ET2C1</b>	T2CAP1 中断使能位, 高有效 0: 关闭 T2CAP1 中断功能 1: 使能 T2CAP1 中断功能
[0]	<b>ET2C0</b>	T2CAP0 中断使能位, 高有效 0: 关闭 T2CAP0 中断功能 1: 使能 T2CAP0 中断功能

### 13.3.12 TIMER2 捕获控制寄存器 (T2CAPCON)

寄存器	地址	R/W	描述	复位后的值
<b>T2CAPCON</b>	0x3A	R/W	TIMER2 捕获控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留				T2C1MOD[1:0]		T2C0MOD[1:0]	

位	描述	
[7:4]	保留	保留
[3:2]	<b>T2C1MOD[1:0]</b>	<b>T2CAP1 模式选择</b> 00: 使能 PWM 模式, 不使能捕获模式 01: 使能捕获模式, 并且是上升沿捕获(关闭 PWM 功能) 10: 使能捕获模式, 并且是下降沿捕获(关闭 PWM 功能) 11: 使能捕获模式, 并且是双沿捕获(关闭 PWM 功能)
[1:0]	<b>T2C0MOD[1:0]</b>	<b>T2CAP0 模式选择</b> 00: 使能 PWM 模式, 不使能捕获模式 01: 使能捕获模式, 并且是上升沿捕获(关闭 PWM 功能) 10: 使能捕获模式, 并且是下降沿捕获(关闭 PWM 功能) 11: 使能捕获模式, 并且是双沿捕获(关闭 PWM 功能)

### 13.3.13 TIMER2 捕获个数配置寄存器 (T2CAPNUM)

寄存器	地址	R/W	描述	复位后的值
<b>T2CAPNUM</b>	0x3B	R/W	TIMER2 捕获个数配置寄存器	00000000



BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	T2C1NUM[2:0]			保留	T2C0NUM[2:0]		

位	描述	
[7]	保留	保留
[6:4]	T2C1NUM[2:0]	T2CAP1 的沿（下降沿/上升沿）的捕获个数软件配置 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128
[3]	保留	保留
[2:0]	T2C0NUM[2:0]	T2CAP0 的沿（下降沿/上升沿）的捕获个数软件配置 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128

### 13.3.14 TIMER2 捕获输入滤波时间配置寄存器 (T2CAPNFCNT)

寄存器	地址	R/W	描述	复位后的值
T2CAPNFCNT	0x3C	R/W	TIMER2 捕获输入滤波时间配置寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T2CAPNFCNT[7:0]							

位	描述	
[7:0]	T2CAPNFCNT	噪声滤波寄存器，总的滤波时间为 T2CAPNFCNT * 8Tclkcpu。 T2CAPNFCNT 值为：0~ff

## 13.4 定时器

定时器 2 模块的输入为 CLKCPU，在定时器模块集成了一个分频器，分频后的时钟作为 16 bits 计数器的输入时钟。当用户设置了定时器模块的使能标志，16 bits 计数器将启动，将会从 0 递增到 T2R。用户需要设置 T2R 以选择定时超时中断信号。当定时超时发生时，如果有开中断使能的话，中断标志位会自动置 1，程序计数器会跳转到 08h 以执行中断服务程序。

操作步骤：

- 1) 设置 T2PSC，选择定时器 2 时钟。
- 2) 设置 T2R，选择定时器溢出值。
- 3) 设置寄存器标志位：ET2 与 EA，使能定时器中断。
- 4) 设置寄存器标志位：T2EN，使能定时器。
- 5) 当定时超时发生时，程序计数器会跳转 T2 中断入口地址 08H。

定时器 2 溢出时间计算方法：

定时器 2 溢出时间 = (T2R[15:0]+1) / T2CLK

## 13.5 PWM 输出

定时器 2 有 2 路 T2PWM 输出，即 T2PWM0 和 T2PWM1 引脚，2 路 T2PWM 的周期大小通过配置 T2PSC[2:0] 和 T2R[15:0] 来实现，T2PWMx 占空比由 T2Dx[15:0] 来调整。

### 13.5.1 T2PWM 输出配置步骤

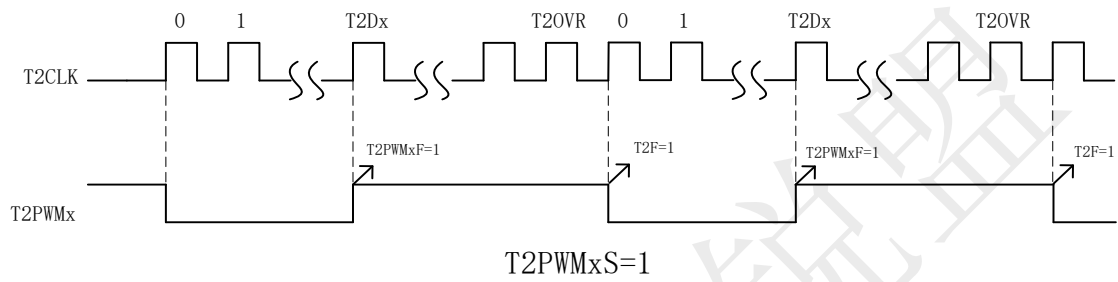
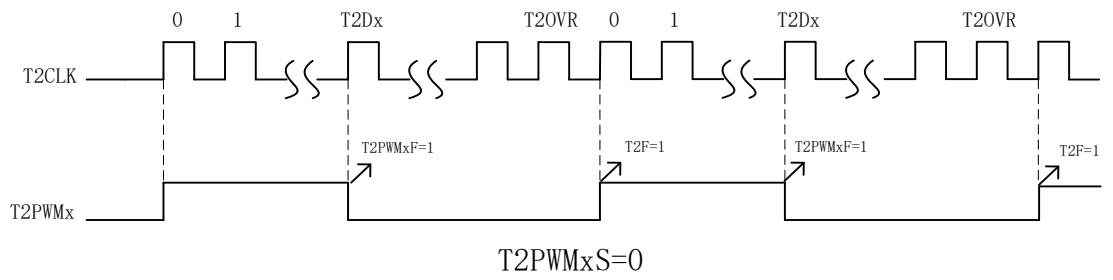
T2PWM 的周期 = (T2R[15:0]+1) / T2CLK.

T2PWMx 的有效电平宽度比 = T2Dx[15:0] / (T2R[15:0]+1)

操作步骤：

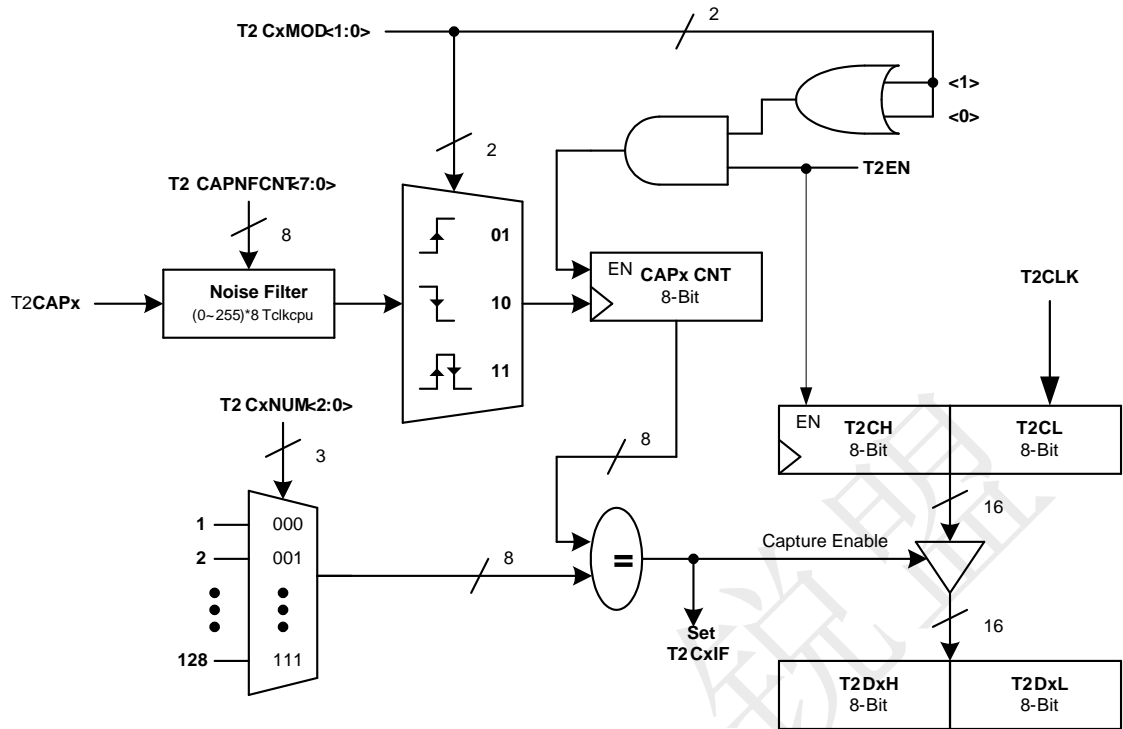
- 1) 设置 T2PSC[2:0]，配置 T2PWM 时钟。
- 2) 设置 T2R，设置 T2PWM 周期。
- 3) 设置 T2Dx，设置 T2PWMx 有效电平宽度。
- 4) 设置 T2PWMxS，设置 T2PWMx 输出有效电平。
- 5) 设置 T2PWMxEN=1，使能 T2PWMx 输出。
- 6) 设置 T2EN=1，使能定时器。
- 7) 当 T2PWMx 内部计数器计数等于 T2Dx 时，占空比中断标志会置位，如此时占空比中断使能打开，程序计数器会跳转到 08H。进入中断后，通过查询寄存器 T2PWMxF 来判断具体是哪个 T2PWM 引起的中断。
- 8) 当 T2C 计数等于 T2R 时，中断标志会置位。定时器 1 中断使能位如打开的话，程序计数器会跳转到 08H。

周期为 T2R+1，T2PWMx 高电平脉宽为 T2Dx。T2PWMx 波形输出如下：



## 13.6 捕获功能

捕获功能是指该芯片对事件持续的时间计时。定时器 2 共有 2 路捕获，分别从 T2CAP0(PA.2/PB.0/PC.0)、T2CAP1(PA.3/PB.2/PB.6)输入。该模式能够软件配置捕获计数时钟的快慢，软件配置需要捕获的沿的个数，并选择捕获下降沿或者上升沿。捕获完成后，相应的完成标志会置 1，若使能了中断，将产生中断请求。2 个捕获中断共用一个中断入口地址 08H。

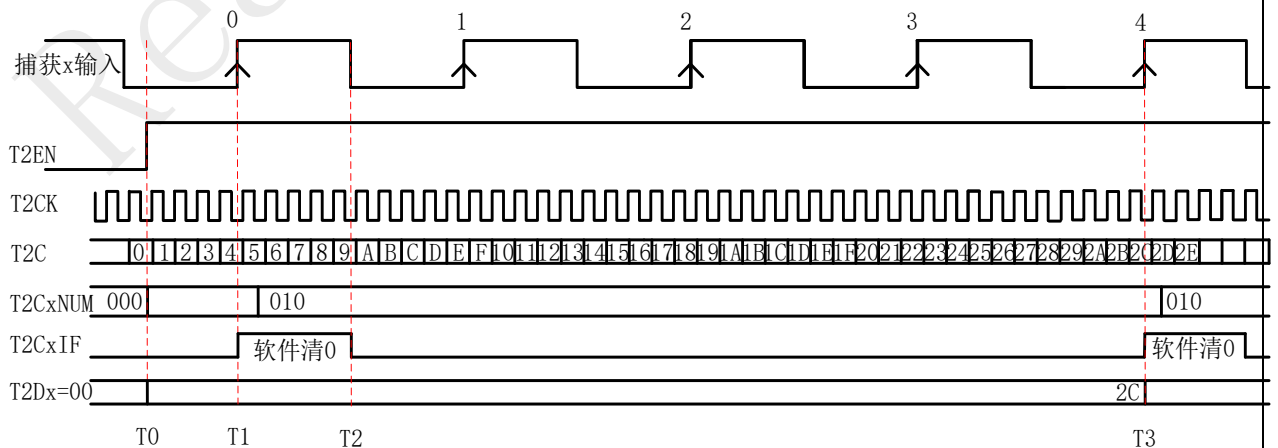


捕获结构框图

### 13.6.1 软件配置步骤

- 1) 配置捕捉管脚（同 T2PWMx 管脚，见代码选项选择位）；
- 2) 设置 T2PSC[2:0]确定 T2 时钟分频；
- 3) 设置 T2CAPNFCNT 噪音滤波时间；
- 4) 设置 T2CxMOD[1:0]触发的方式；
- 5) 设置 T2CxNUM[2:0]一次捕捉触发信号的个数；
- 6) 打开中断允许标志 ET2Cx=1（如果总中断没有打开，请首先打开 EA=1）；
- 7) 打开开始计时 T2EN=1。

图 12-4 捕获 x (x=1~2) 的示意图



### 13.6.2 脉冲周期测量

示意图中，软件配置捕获的边沿的个数  $A=4$ ，即  $T2CxNUM<2:0>=010$ 。为了表示方便，只显示了 16 位计数值的低 8 位。

由于被捕获波形输入边沿的时间的不确定性，使能  $T1EN$  可能大大早于被捕获的边沿，第一次产生计数是完全不准确的。为了解决这个问题，可以采用下述方法：

软件配置捕获的边沿的个数为一个，即  $T2CxNUM<2:0>=000$ 。当  $t=T0$  时，使能  $T2EN$ ，定时器即开始计数，当  $t=T1$  时，第 1 个边沿到来，即第一次捕获完成后，读取  $T2Dx[16:0]=04h$  的值作为初始值并清除  $T2CxF$ 。此时不关闭捕获，软件配置  $T2CxNUM[2:0]=010$ 。其它沿用之前软件配置，当  $t=T3$  时，第 4 个沿到来，捕获再次完成，读取此时  $T2Dx[16:0]=2Ch$  的值作为完成值。将捕获完成值减去捕获初始值，即得到被捕获波形的  $A$  个周期内， $T2CK$  的个数。如图示脉冲宽度为  $2Ch-04h=28h$ 。若依次循环测量，即可跟踪被捕获波形的周期变化。

注意：

1. 为了防止定时器溢出，用户需要注意被捕获的波形周期与计数时钟  $T2CK$  周期的配合；
2. 在被捕获的波形输入之前，不要过早的使能  $T2EN$ ；
3. 为了让捕获计数更精确，推荐用数倍于被捕获波形频率的  $T2CK$  采样；
4. 增加捕获的边沿的个数（ $A$  的整数倍），再进行平均计算。

### 13.6.3 电平宽度测量

软件配置捕获的边沿的个数为一个，即  $T2CxNUM[2:0]=000$ 。当  $t=T0$  时，使能  $T2EN$ ，定时器即开始计数，当  $t=T1$  时，第 1 个边沿到来，即第一次捕获完成后，读取  $T2Dx[16:0]=04h$  的值作为初始值并清除  $T2CxF$ 。此时不关闭捕获，改变触发方式，即软件配置  $T2CxMOD[1:0]=10$ ，其它沿用之前软件配置，当  $t=T2$  时，捕获再次完成，读取此时  $T2Dx[15:0]=09h$  的值作为完成值。将捕获完成值减去捕获初始值，即得到被捕获高电平宽度时间，高电平宽度为  $09h-04h=05h$ 。若依次循环测量，即可跟踪被捕获波形的周期变化。

## 14 Timer3

### 14.1 特性

TIMER3 为 16 位 Basic Timer。可以工作于定时器模式。  
 定时器模块的输入为 CLKCPU。

### 14.2 寄存器

#### 14.2.1 TIMER3 配置寄存器 (T3M)

寄存器	地址	R/W	描述	复位后的值
T3M	0x19	R/W	TIMER3 配置寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	保留			T3PSC[2:0]		T3EN	

位	描述	
7	保留	保留
[6:4]	保留	保留
[3:1]	T3PSC[2:0]	定时器 3 时钟分频 000: 1 001: 2 010: 4 011: 8 100: 16 101: 32 110: 64 111: 128
[0]	T3EN	定时器 3 使能位 0: 禁止定时器 3 1: 使能定时器 3

### 14.2.2 TIMER3 重载寄存器高 8 位(T3RH)

寄存器	地址	R/W	描述	复位后的值
<b>T3RH</b>	0x1A	R/W	TIMER3 重载寄存器高 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T3R[15:8]							

位	描述	
[7:0]	T3R[15:8]	TIMER3 重载寄存器高 8 位

### 14.2.3 TIMER3 重载寄存器低 8 位(T3RL)

寄存器	地址	R/W	描述	复位后的值
<b>T3RL</b>	0x1B	R/W	TIMER3 重载寄存器低 8 位	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
T3R[7:0]							

位	描述	
[7:0]	T3R[7:0]	TIMER3 重载寄存器低 8 位

## 14.3 定时器

定时器模块的输入为 CLKCPU。在定时器模块集成了一个分频器，分频的时钟 T3CLK 作为 16 bits 计数器的输入时钟。当用户设置了定时器模块的使能标志，16 bits 计数器将启动，将会从 0 递增至 T3R。当用户开启定时器 T3 中断，当定时超时发生时，中断标志位会置 1，程序计数器会跳转到 08h 以执行中断服务程序。

操作：

- 1) 设置 T3CLK，选择定时器时钟。
- 2) 设置 T3R，选择定时器溢出值。
- 3) 使能定时器中断：ET3 与 EA。
- 4) 使能定时器：T3EN。
- 5) 当定时超时发生时，程序计数器会跳转到 008H。

定时器 3 溢出时间= (T3R[15:0]+1) / T3CLK.

## 15 ADC

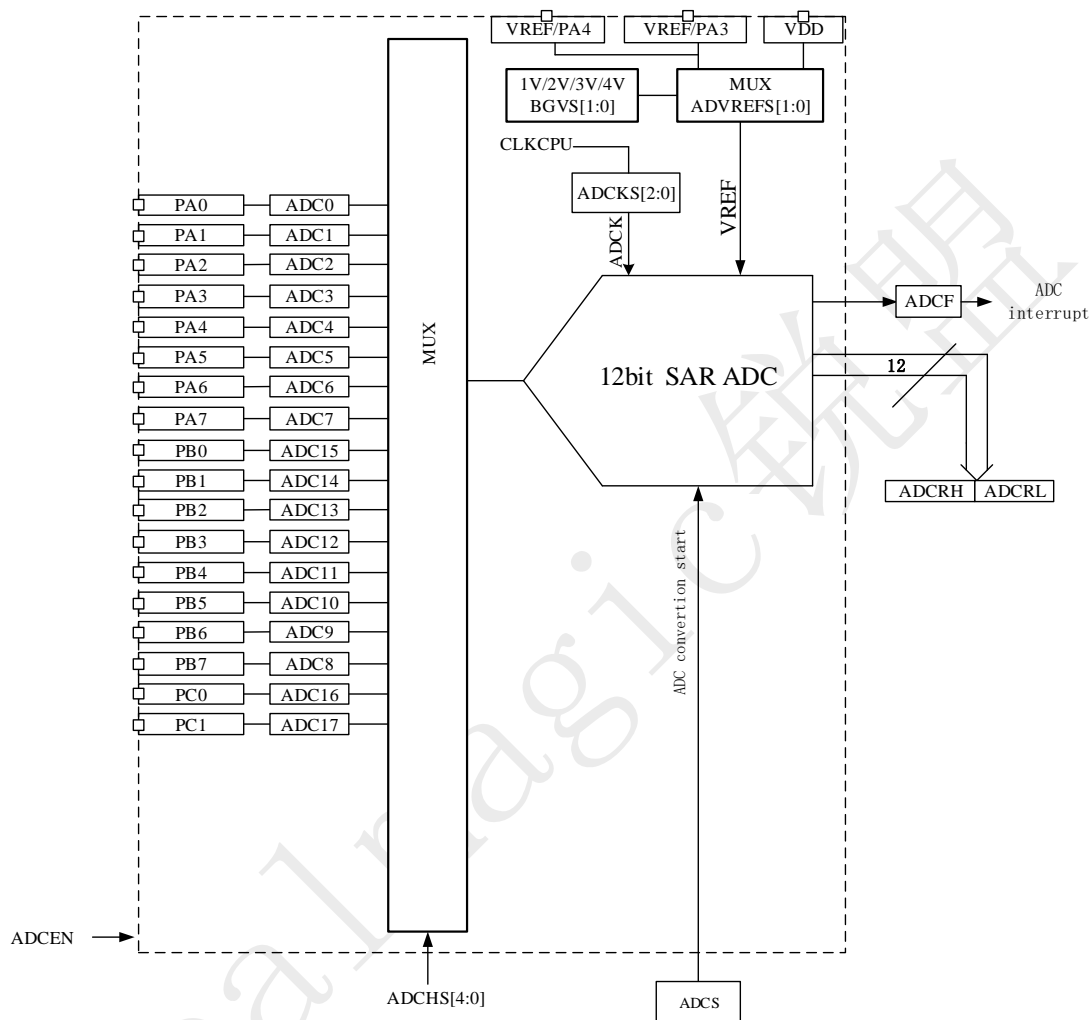
### 15.1 概述

RM1211A 内嵌 12 位逐次逼近寄存器型(SAR)的模拟数字转换器(ADC)。模数转换模块负责将管脚上的模拟信号转换为 12 位二进制数据。RM1211A 支持 12 信道单端输入模式。内部带隙电压(band-gap voltage)为  $4/3/2/1V$ ，同时也可用作内部 ADC 输入端。所有模拟电路复用同一组采样电路和同一组采样保持电容。该组采样保持电容为转换电路的输入端。然后转换器通过逐次逼近的方式得到有效结果并存放在 ADC 结果寄存器中。



## 15.2 功能描述

### 15.2.1 ADC 工作方式



12-位 ADC 模块结构图

在开始 ADC 转换前,通过设置 ADCEN(ADCCON1.0)位使能 ADC 电路,从而激活 ADC 电路,由于 ADC 模块需要额外功耗,一旦 ADC 转换模块不再使用,建议清零 ADCEN 位关闭 ADC 模块电路以节省功耗。

ADC 转换输入管脚需要特别配置,通过 ADCHS[4:0]来选择采样所需要的 ADC 通道连接到采样电路上。同时,用户需要通过配置 PACHS/PBCHS/PCCHS 寄存器相应位来将 ADC 输入脚变成纯模拟输入电路。

通过置 ADCS 位(ADCCON0.6)开启 AD 转换。当转换完成后,硬件会自动清除该位,同时置 ADCF(ADCCON0.7)位,如果之前 ADC 中断已使能,则会产生 ADC 中断。转换结果存放在 ADCRH (高 8 位)及 ADCRL (低 4 位)中。12 位转换结果值为  $4095 \times \frac{V_{AIN}}{V_{REF}}$

内部及外部数字电路,可能影响采样结果的准确度。所以如果需要高精度的转换结果,

请参考如下应用，以降低噪声电平干扰。

- ◆ 模拟输入脚尽量离芯片越近越好，避免管脚附近有高速数字电路经过，并离高速数字电路越远越好。
- ◆ 在转换过程中，将芯片进入空闲模式。
- ◆ 如果模拟输入脚 AIN 在系统中同时需要切换做数字管脚，请确保在转换过程中不要做数字/模拟切换动作。

## 15.2.2 ADC 参考带隙电压应用

RM1211A 内嵌带隙电压(band-gap voltage)为 4/3/2/1V 可选。可选择内部 VDD、内部带隙电压、外部 VREFP 从 PA.3 引脚输入作为 ADC 的正参考电压。

## 15.3 寄存器

### 15.3.1 ADC 控制寄存器 0 (ADCCON0)

寄存器	地址	R/W	描述	复位后的值
ADCCON0	0x3F	R/W	ADC 控制寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ADCF	ADCS	保留		保留			

位	描述	
[7]	ADCF	<b>ADC 标志</b> 当 AD 转换完成，该位置 1。可读取到当前 AD 转换结果。需要软件清零。
[6]	ADCS	<b>A/D 转换软件启动位</b> 该位置 1 启动 AD 转换。在 AD 转换过程中该位保持为 1，当转换结束硬件自动清 0。这意味着写入 ADCS 的值和读出的不一定相符 写： 0: 无动作 1: 开始 AD 转换 读： 0: ADC 模块空闲状态 1: ADC 模块工作中
[5:0]	保留	保留

### 15.3.2 ADC 控制寄存器 1 (ADCCON1)

寄存器	地址	R/W	描述	复位后的值
ADCCON1	0x40	R/W	ADC 控制寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
CHOP	ADVREFS[2:0]			保留		保留	ADCEN

位	描述	
[7]	<b>CHOP</b>	<b>OFFSET 交换</b> 0: 两端信号不交换 (正端为信号, 负端为参考电压) 1: 器两端信号交换  在 AD 转换过程中通过不断变换 ADCCON1 寄存器中的 CHOP 的值。如第一次 AD 转换 CHOP 置 0, 第二次 AD 转换 CHOP 置 1, 然后将第一次和第二次测试的 AD 值求平均值。两次转换得到的平均值就是去掉失调电压的正确结果。
[6:4]	<b>ADVREFS[2:0]</b>	<b>ADC 参考电压 VREF 选择</b> 000: 选择 VDD 为参考电压正端, VSS 为参考电压负端(默认) 001: 选择 VDD 为参考电压正端, VREFN (PA7) 为参考电压负端 010: 选择内部带隙电压为参考电压正端, VSS 为参考电压负端 011: 选择内部带隙电压为参考电压正端, VREFN (PA7) 为参考电压负端 100: 选择内部带隙电压为参考电压正端可输出接滤波电容 PA1 或 PA4, VSS 为参考电压负端 101: 选择内部带隙电压为参考电压正端可输出接滤波电容 PA1 或 PA4, VREFN (PA7) 为参考电压负端 110: 选择外部 VREFP 端口 PA1 或 PA4 输入作为参考电压正端, VSS 为参考电压负端 111: 选择外部 VREFP 端口 PA1 或 PA4 输入作为参考电压正端, VREFN (PA7) 为参考电压负端
[3:2]	保留	保留
[1]	保留	保留
[0]	<b>ADCEN</b>	<b>ADC 使能位</b> 0: ADC 转换电路关闭 1: ADC 转换电路打开

### 15.3.3 ADC 控制寄存器 2 (ADCCON2)

寄存器	地址	R/W	描述	复位后的值
ADCCON2	0x41	R/W	ADC 控制寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	保留	保留	保留	保留	ADCKS[2:0]		

位	描述	
[7]	保留	保留
[6]	保留	保留
[5]	保留	保留
[4]	保留	保留
[3]	保留	保留
[2:0]	ADCKS[2:0]	<b>ADC 转换时钟选择位</b> 000: Fcpuclk/1 001: Fcpuclk/2 010: Fcpuclk/4 (默认) 011: Fcpuclk/8 100: Fcpuclk/16 101: Fcpuclk/32 110: Fcpuclk/64 111: Fcpuclk/128

### 15.3.4 ADC 控制寄存器 3 (ADCCON3)

寄存器	地址	R/W	描述	复位后的值
ADCCON3	0x42	R/W	ADC 控制寄存器 3	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ADTS[1:0]		VCMS	ADT[4:0]				

位	描述	
[7:6]	ADTS[1:0]	<b>ADC 引入 OFFSET</b> 00: 不引入 OFFSET 01: ADC 引入 8 个码的正 OFFSET 10: ADC 引入 8 个码的负 OFFSET 11: 不引入 OFFSET
[5]	VCMS	<b>vcm 参考电压选择</b> 0: VCC 分压 1: REFP 分压
[4:0]	ADT[4:0]	<b>ADC 的 OFFSET 校准控制位 (默认为 10000)</b> 00000 为负的最大 OFFSET 11111 为正的最大 OFFSET

### 15.3.5 ADC 控制寄存器 4 (ADCCON4)

寄存器	地址	R/W	描述	复位后的值
ADCCON4	0x43	R/W	ADC 控制寄存器 4	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
DFILEN	DFILNUM[1:0]		ADCHS[4:0]				

位	描述																																																																				
[7]	<b>DFILEN</b> <b>SAR ADC 数字滤波器使能信号</b> 0: 关闭 1: 开启																																																																				
[6:5]	<b>DFILNUM[1:0]</b> <b>SAR ADC 数字滤波器滤波次数</b> 00: 2 次, 01: 4 次 10: 8 次 11: 16 次 此时 ADR 的结果为 2 次或 4 次 ADC 转换的平均值, 开启数字滤波后 ADC 的转换速率会下降到原来的 1/2 或 1/4																																																																				
[4:0]	<b>ADCHS[4:0]</b> <b>A/D 转换通道选择</b> 该位用于选择 ADC 转换通道。当 ADCEN 为 0 所有输入无效 <table border="1" style="width: 100%; border-collapse: collapse; margin-top: 5px;"> <thead> <tr> <th>ADCHS[4:0]</th> <th>ADC 转换通道</th> <th>ADCHS[4:0]</th> <th>ADC 转换通道</th> </tr> </thead> <tbody> <tr> <td>00000</td> <td>AIN0</td> <td>10000</td> <td>AIN16</td> </tr> <tr> <td>00001</td> <td>AIN1</td> <td>10001</td> <td>AIN17</td> </tr> <tr> <td>00010</td> <td>AIN2</td> <td>10010</td> <td>1/4 VDD</td> </tr> <tr> <td>00011</td> <td>AIN3</td> <td>10011</td> <td>内部带隙电压 VREFP(1V/2V/3V/4V 根据 BGV[S][1:0]选择</td> </tr> <tr> <td>00100</td> <td>AIN4</td> <td>10100</td> <td>*运放 0 内部输出</td> </tr> <tr> <td>00101</td> <td>AIN5</td> <td>10101</td> <td>*运放 1 内部输出</td> </tr> <tr> <td>00110</td> <td>AIN6</td> <td>10110</td> <td>*运放 2 内部输出</td> </tr> <tr> <td>00111</td> <td>AIN7</td> <td>10111</td> <td>*运放 3 内部输出</td> </tr> <tr> <td>01000</td> <td>AIN8</td> <td>11000</td> <td>1/4 PA5</td> </tr> <tr> <td>01001</td> <td>AIN9</td> <td>11001</td> <td>1/4 PB6</td> </tr> <tr> <td>01010</td> <td>AIN10</td> <td>11010</td> <td>VREFN</td> </tr> <tr> <td>01011</td> <td>AIN11</td> <td>11011</td> <td>温度传感器</td> </tr> <tr> <td>01100</td> <td>AIN12</td> <td>11100</td> <td>保留</td> </tr> <tr> <td>01101</td> <td>AIN13</td> <td>11101</td> <td>保留</td> </tr> <tr> <td>01110</td> <td>AIN14</td> <td>11110</td> <td>保留</td> </tr> <tr> <td>01111</td> <td>AIN15</td> <td>11111</td> <td>保留</td> </tr> </tbody> </table> *接运放 0~3 内部输出, 包括内部 PGA 放大的输出	ADCHS[4:0]	ADC 转换通道	ADCHS[4:0]	ADC 转换通道	00000	AIN0	10000	AIN16	00001	AIN1	10001	AIN17	00010	AIN2	10010	1/4 VDD	00011	AIN3	10011	内部带隙电压 VREFP(1V/2V/3V/4V 根据 BGV[S][1:0]选择	00100	AIN4	10100	*运放 0 内部输出	00101	AIN5	10101	*运放 1 内部输出	00110	AIN6	10110	*运放 2 内部输出	00111	AIN7	10111	*运放 3 内部输出	01000	AIN8	11000	1/4 PA5	01001	AIN9	11001	1/4 PB6	01010	AIN10	11010	VREFN	01011	AIN11	11011	温度传感器	01100	AIN12	11100	保留	01101	AIN13	11101	保留	01110	AIN14	11110	保留	01111	AIN15	11111	保留
ADCHS[4:0]	ADC 转换通道	ADCHS[4:0]	ADC 转换通道																																																																		
00000	AIN0	10000	AIN16																																																																		
00001	AIN1	10001	AIN17																																																																		
00010	AIN2	10010	1/4 VDD																																																																		
00011	AIN3	10011	内部带隙电压 VREFP(1V/2V/3V/4V 根据 BGV[S][1:0]选择																																																																		
00100	AIN4	10100	*运放 0 内部输出																																																																		
00101	AIN5	10101	*运放 1 内部输出																																																																		
00110	AIN6	10110	*运放 2 内部输出																																																																		
00111	AIN7	10111	*运放 3 内部输出																																																																		
01000	AIN8	11000	1/4 PA5																																																																		
01001	AIN9	11001	1/4 PB6																																																																		
01010	AIN10	11010	VREFN																																																																		
01011	AIN11	11011	温度传感器																																																																		
01100	AIN12	11100	保留																																																																		
01101	AIN13	11101	保留																																																																		
01110	AIN14	11110	保留																																																																		
01111	AIN15	11111	保留																																																																		

### 15.3.6 ADC 转换结果低位寄存器 (ADCRL)

寄存器	地址	R/W	描述	复位后的值
ADCRL	0x46	R/W	ADC 转换结果低位寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
BGVS[1:0]		TSAMPS[1:0]		ADCR[3:0] (只读)			

位	描述	
[7:6]	BGVS[1:0]	ADC 内部带隙基准电压选择 00: 1V 01: 2V (默认) 10: 3V 11: 4V
[5:4]	TSAMPS[1:0]	ADC 输入采样时间选择 00: 2*T_ADCK 01: 4*T_ADCK (默认) 10: 8*T_ADCK 11: 16*T_ADCK
[3:0]	ADCR[3:0]	ADC 转换结果低位 ADC 转换结果低 4 位

### 15.3.7 ADC 转换结果高位寄存器 (ADCRH)

寄存器	地址	R/W	描述	复位后的值
ADCRH	0x47	R	ADC 转换结果高位寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ADCR[11:4]							

位	描述	
[7:0]	ADCR[11:4]	ADC 转换结果高位 ADC 转换结果高 8 位

### 15.3.8 ADC 模拟输入通道 (PA) 功能寄存器 (PACHS)

寄存器	地址	R/W	描述	复位后的值
PACHS	0x49	R/W	ADC 模拟输入通道 (PA) 功能寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0

<b>PACHS[7:0]</b>
-------------------

位	描述	
[7:0]	<b>PACHS[7:0]</b>	<b>ADC 模拟输入通道使能位</b> 0: PAx 为数字通道 (x=0~7) 1: PAx 为 ADC 模拟输入通道 (x=0~7)

### 15.3.9 ADC 模拟输入通道 (PB) 功能寄存器 (PBCHS)

寄存器	地址	R/W	描述	复位后的值
<b>PBCHS</b>	0x4A	R/W	ADC 模拟输入通道 (PB) 功能寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留		<b>PBCHS[6:0]</b>					

位	描述	
[7]	保留	保留
[6:0]	<b>PBCHS[6:0]</b>	<b>ADC 模拟输入通道使能位</b> 0: PBx 为数字通道 (x=0~7) 1: PBx 为 ADC 模拟输入通道 (x=0~7)

### 15.3.10 ADC 模拟输入通道 (PC) 功能寄存器 (PCCHS)

寄存器	地址	R/W	描述	复位后的值
<b>PCCHS</b>	0x4B	R/W	ADC 模拟输入通道 (PC) 功能寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留						<b>PCCHS[1:0]</b>	

位	描述	
[7:2]	保留	保留
[1:0]	<b>PCCHS[7:0]</b>	<b>ADC 模拟输入通道使能位</b> 0: PCx 为数字通道 (x=0~1) 1: PCx 为 ADC 模拟输入通道 (x=0~1)

## 15.4 ADC 应用注意事项

在 ADC 使用过程中需要注意以下事项：

- 1、ADEN 使能后需要等待至少 5 $\mu$ S，等待 ADC 内部基准建立后方可进行转换；
- 2、ADC 最快速度可以接近 1MSPS，此时需要改变 ADC BUFFER 和 ADC 比较器电流档位，这时候功耗会增加，性能会稍微下降，最优的性能速度为 250KSPS 及以下。
- 4、为了小信号性能提升，实际 ADC 示范 Demo 程序可以打开 SAR ADC 数字滤波器功能，开启此功能后 ADC 转换速率下降。
- 5、对 AD 转换数据进行处理，可以采用以下几种滤波算法：限幅滤波、中位值滤波、算数平均滤波、递推平均滤波、中位值平均滤波、限幅平均滤波、一阶滞后滤波、加权递推平均滤波、消抖滤波、限幅消抖滤波。

ADC 转换步骤如下：

- 1、配置 ADC 时钟和采样建立时间；
- 2、配置引脚功能和选择 ADC 通道；
- 3、配置 ADC 参考电压，内部参考或外部参考；
- 4、使能 ADC，等待至少 5 $\mu$ S 等待 ADC 内部基准稳定；
- 5、启动 ADC 转换，判断 ADCF 是否置 1，根据应用是否进入中断；
- 6、清除 ADCF，完成一次 ADC 转换。



## 16 比较器/运算放大器

### 16.1 特性

RM1211A 集成了 4 路运放/比较器电路。

4 个比较器的正/负输入端和输出端均可以灵活配置是否引出到 IO 口，若输出端不引出到 IO 口，用户仍然可以通过寄存器读到比较结果。4 个比较器还可配置成 OP 使用。

四个通道均支持滤波功能，通过软件配置 CMPxDBS 来选择滤波的宽度和滤波时钟。

### 16.2 寄存器

#### 16.2.1 比较器 0 控制寄存器 0 (CMP0CON0)

寄存器	地址	R/W	描述	复位后的值
CMP0CON0	0x30	R/W	比较器 0 控制寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
CMP0O	CMP0HYS	SF0EN	OPA0CRS	OPA0COFM	CMP0OEN	CMP0MODE	OPA0EN

位	描述
[7]	<b>CMP0O</b> 比较器 0 的输出结果，只读。
[6]	<b>CMP0HYS</b> 比较器 0 迟滞效应使能位 0: 无迟滞效果 (默认) 1: 带 40mv 的迟滞
[5]	<b>SF0EN</b> 运放 source follower 级使能开关 0: 关闭 1: 使能
[4]	<b>OPA0CRS</b> OPA0 调节模式输入端选择位 0: OPA0 调节模式负端输入 1: OPA0 调节模式正端输入
[3]	<b>OPA0COFM</b> OPA0 工作模式选择位 0: OPA0 工作在正常模式 1: OPA0 工作在调节模式 (运放正端、负端短接)
[2]	<b>CMP0OEN</b> 比较器模式下，比较器输出到 GPIO 使能位 0: 不输出 1: 比较器 0 输出到 PA5
[1]	<b>CMP0MODE</b> 比较器 0 模式选择位 0: 运放模式

		1: 比较器模式
[0]	<b>OPA0EN</b>	<b>OPA0 使能位</b> 0: 禁止 OPA0 1: 使能 OPA0

### 16.2.2 比较器 0 控制寄存器 1 (CMP0CON1)

寄存器	地址	R/W	描述	复位后的值
<b>CMP0CON1</b>	0x31	R/W	比较器 0 控制寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>A0P[2:0]</b>			<b>OPA0ADJ[4:0]</b>				

位	描述	
[7:5]	<b>A0P[2:0]</b>	<b>运放正端输入选择</b> 000: 无效 001: PA3 010: VSS 011: DAC00 100: A1O 101: A2O 110: A3O 111: DAC00, 也就是接到 PA3, 即 PA3 输出了 DAC0 的输出
[4:0]	<b>OPA0ADJ[4:0]</b>	<b>运放 0 失调电压调节位, 默认为 10000,00000 为负失调最大, 11111 为正失调最大</b>

### 16.2.3 比较器 0 控制寄存器 2 (CMP0CON2)

寄存器	地址	R/W	描述	复位后的值
<b>CMP0CON2</b>	0x32	R/W	比较器 0 控制寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>A0N[2:0]</b>			<b>A0SW4</b>	<b>A0SW3</b>	<b>A0SW2</b>	<b>A0SW1</b>	<b>A0SW0</b>

位	描述	
[7:5]	<b>A0N[2:0]</b>	<b>运放负端输入选择</b> 000: 无效 001: PA4 010: VSS 011: DAC00 100: A1O

		101: A2O 110: A3O 111: 无效
[4]	<b>A0SW4</b>	运放 0 通路开关 4 选择位 0: 关闭 1: 开启
[3]	<b>A0SW3</b>	运放 0 通路开关 3 选择位 0: 关闭 1: 开启
[2]	<b>A0SW2</b>	运放 0 通路开关 2 选择位 0: 关闭 1: 开启
[1]	<b>A0SW1</b>	运放 0 通路开关 1 选择位 0: 关闭 1: 开启
[0]	<b>A0SW0</b>	运放 0 通路开关 0 选择位 0: 关闭 1: 开启

### 16.2.4 比较器 0 控制寄存器 3 (CMP0CON3)

寄存器	地址	R/W	描述	复位后的值
<b>CMP0CON3</b>	0x33	R/W	比较器 0 控制寄存器 3	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A0GS[2:0]			CMP0INV	CMP0DBS[1:0]		INT0SEL[1:0]	

位	描述
[7:5]	<b>A0GS[2:0]</b> 运放 PGA 放大倍数选择 000: 内部 PGA 放大倍数为 4 倍 001: 内部 PGA 放大倍数为 8 倍 010: 内部 PGA 放大倍数为 16 倍 011: 内部 PGA 放大倍数为 32 倍 100: 内部 PGA 放大倍数为 64 倍 101~111: 无效
[4]	<b>CMP0INV</b> 比较器 0 反相输出结果 0: 禁止 1: 使能
[3:2]	<b>CMP0DBS[1:0]</b> 比较器 0 输出结果滤波时间 00: 无 01: 4*Fsys/Fcpu

		10: 8*Fsys/Fcpu 11: 16*Fsys/Fcpu
[1:0]	<b>INT0SEL[1:0]</b>	<b>比较器 0 输出结果中断触发方式</b> 00: 上升沿触发 01: 下降沿触发 1x: 上升下降沿均触发

### 16.2.5 比较器 1 控制寄存器 0 (CMP1CON0)

寄存器	地址	R/W	描述	复位后的值
<b>CMP1CON0</b>	0x34	R/W	比较器 1 控制寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>CMP1O</b>	<b>CMP1HYS</b>	<b>SF1EN</b>	<b>OPA1CRS</b>	<b>OPA1COFM</b>	<b>CMP1OEN</b>	<b>CMP1MODE</b>	<b>OPA1EN</b>

位	描述	
[7]	<b>CMP1O</b>	比较器 1 的输出结果，只读。
[6]	<b>CMP1HYS</b>	比较器 1 迟滞效应使能位 0: 无迟滞效果（默认） 1: 带 40mv 的迟滞
[5]	<b>SF1EN</b>	运放 source follower 级使能开关 0: 关闭 1: 使能
[4]	<b>OPA1CRS</b>	OPA1 调节模式输入端选择位 0: OPA1 调节模式负端输入 1: OPA1 调节模式正端输入
[3]	<b>OPA1COFM</b>	OPA1 工作模式选择位 0: OPA1 工作在正常模式 1: OPA1 工作在调节模式（运放正端、负端短接）
[2]	<b>CMP1OEN</b>	比较器模式下，比较器输出到 GPIO 使能位 0: 不输出 1: 比较器 1 输出到 PA6
[1]	<b>CMP1MODE</b>	比较器 1 模式选择位 0: 运放模式 1: 比较器模式
[0]	<b>OPA1EN</b>	OPA1 使能位 0: 禁止 OPA1 1: 使能 OPA1

### 16.2.6 比较器 1 控制寄存器 1 (CMP1CON1)

寄存器	地址	R/W	描述	复位后的值
CMP1CON1	0x35	R/W	比较器 1 控制寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A1P[2:0]			OPA1ADJ[4:0]				

位	描述	
[7:5]	A1P[2:0]	运放正端输入选择 000: 无效 001: PC1 010: VSS 011: DAC00 100: A1O 101: A2O 110: A3O 111: DAC00, 也就是接到 PC1, 即 PC1 输出了 DAC0 的输出
[4:0]	OPA1ADJ[4:0]	运放 1 失调电压调节位, 默认为 10000,00000 为负失调最大, 11111 为正失调最大

### 16.2.7 比较器 1 控制寄存器 2 (CMP1CON2)

寄存器	地址	R/W	描述	复位后的值
CMP1CON2	0x36	R/W	比较器 1 控制寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A1N[2:0]			A1SW4	A1SW3	A1SW2	A1SW1	A1SW0

位	描述	
[7:5]	A1N[2:0]	运放负端输入选择 000: 无效 001: PA7 010: VSS 011: DAC00 100: A1O 101: A2O 110: A3O 111: 无效
[4]	A1SW4	运放 1 通路开关 4 选择位 0: 关闭 1: 开启

[3]	<b>A1SW3</b>	运放 1 通路开关 3 选择位 0: 关闭 1: 开启
[2]	<b>A1SW2</b>	运放 1 通路开关 2 选择位 0: 关闭 1: 开启
[1]	<b>A1SW1</b>	运放 1 通路开关 1 选择位 0: 关闭 1: 开启
[0]	<b>A1SW0</b>	运放 1 通路开关 0 选择位 0: 关闭 1: 开启

### 16.2.8 比较器 1 控制寄存器 3 (CMP1CON3)

寄存器	地址	R/W	描述	复位后的值
<b>CMP1CON3</b>	0x37	R/W	比较器 1 控制寄存器 3	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A1GS[2:0]			CMP1INV	CMP1DBS[1:0]		INT1SEL[1:0]	

位	描述	
[7:5]	<b>A1GS[2:0]</b>	运放 PGA 放大倍数选择 000: 内部 PGA 放大倍数为 4 倍 001: 内部 PGA 放大倍数为 8 倍 010: 内部 PGA 放大倍数为 16 倍 011: 内部 PGA 放大倍数为 32 倍 100: 内部 PGA 放大倍数为 64 倍 101~111: 无效
[4]	<b>CMP1INV</b>	比较器 1 反相输出结果 0: 禁止 1: 使能
[3:2]	<b>CMP1DBS[1:0]</b>	比较器 1 输出结果滤波时间 00: 无 01: 4*Fsys/Fcpu 10: 8*Fsys/Fcpu 11: 16*Fsys/Fcpu
[1:0]	<b>INT1SEL[1:0]</b>	比较器 1 输出结果中断触发方式 00: 上升沿触发 01: 下降沿触发 1x: 上升下降沿均触发

### 16.2.9 比较器 2 控制寄存器 0 (CMP2CON0)

寄存器	地址	R/W	描述	复位后的值
CMP2CON0	0x38	R/W	比较器 2 控制寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
CMP2O	CMP2HYS	SF2EN	OPA2CRS	OPA2COFM	CMP2OEN	CMP2MODE	OPA2EN

位	描述
[7]	<b>CMP2O</b> 比较器 2 的输出结果, 只读。
[6]	<b>CMP2HYS</b> 比较器 2 迟滞效应使能位 0: 无迟滞效果 (默认) 1: 带 40mv 的迟滞
[5]	<b>SF2EN</b> 运放 source follower 级使能开关 0: 关闭 1: 使能
[4]	<b>OPA2CRS</b> OPA2 调节模式输入端选择位 0: OPA2 调节模式负端输入 1: OPA2 调节模式正端输入
[3]	<b>OPA2COFM</b> OPA2 工作模式选择位 0: OPA2 工作在正常模式 1: OPA2 工作在调节模式 (运放正端、负端短接)
[2]	<b>CMP2OEN</b> 比较器模式下, 比较器输出到 GPIO 使能位 0: 不输出 1: 比较器 2 输出到 PB6
[1]	<b>CMP2MODE</b> 比较器 2 模式选择位 0: 运放模式 1: 比较器模式
[0]	<b>OPA2EN</b> OPA2 使能位 0: 禁止 OPA2 1: 使能 OPA2

### 16.2.10 比较器 2 控制寄存器 1 (CMP2CON1)

寄存器	地址	R/W	描述	复位后的值
CMP2CON1	0x39	R/W	比较器 2 控制寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A2P[2:0]				OPA2ADJ[4:0]			

位	描述	
[7:5]	A2P[2:0]	运放正端输入选择 000: 无效 001: PC0 010: VSS 011: DAC10 100: A1O 101: A2O 110: A3O 111: DAC10, 也就是接到 PC0, 即 PC0 输出了 DAC1 的输出
[4:0]	OPA2ADJ[4:0]	运放 2 失调电压调节位, 默认为 10000,00000 为负失调最大, 11111 为正失调最大

### 16.2.11 比较器 2 控制寄存器 2 (CMP2CON2)

寄存器	地址	R/W	描述	复位后的值
CMP2CON2	0x3A	R/W	比较器 2 控制寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A2N[2:0]			A2SW4	A2SW3	A2SW2	A2SW1	A2SW0

位	描述	
[7:5]	A2N[2:0]	运放负端输入选择 000: 无效 001: PB7 010: VSS 011: DAC00 100: A1O 101: A2O 110: A3O 111: 无效
[4]	A2SW4	运放 2 通路开关 4 选择位 0: 关闭 1: 开启
[3]	A2SW3	运放 2 通路开关 3 选择位 0: 关闭 1: 开启
[2]	A2SW2	运放 2 通路开关 2 选择位 0: 关闭 1: 开启
[1]	A2SW1	运放 2 通路开关 1 选择位 0: 关闭



		1: 开启
[0]	A2SW0	运放 2 通路开关 0 选择位 0: 关闭 1: 开启

### 16.2.12 比较器 2 控制寄存器 3 (CMP2CON3)

寄存器	地址	R/W	描述	复位后的值
CMP2CON3	0x3B	R/W	比较器 2 控制寄存器 3	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A2GS[2:0]			CMP2INV	CMP2DBS[1:0]		INT2SEL[1:0]	

位	描述	
[7:5]	A2GS[2:0]	运放 PGA 放大倍数选择 000: 内部 PGA 放大倍数为 4 倍 001: 内部 PGA 放大倍数为 8 倍 010: 内部 PGA 放大倍数为 16 倍 011: 内部 PGA 放大倍数为 32 倍 100: 内部 PGA 放大倍数为 64 倍 101~111: 无效
[4]	CMP2INV	比较器 2 反相输出结果 0: 禁止 1: 使能
[3:2]	CMP2DBS[1:0]	比较器 2 输出结果滤波时间 00: 无 01: 4*Fsys/Fcpu 10: 8*Fsys/Fcpu 11: 16*Fsys/Fcpu
[1:0]	INT2SEL[1:0]	比较器 2 输出结果中断触发方式 00: 上升沿触发 01: 下降沿触发 1x: 上升下降沿均触发

### 16.2.13 比较器 3 控制寄存器 0 (CMP3CON0)

寄存器	地址	R/W	描述	复位后的值
CMP3CON0	0x3C	R/W	比较器 3 控制寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
------	------	------	------	------	------	------	------

CMP3O	CMP3HYS	SF3EN	OPA3CRS	OPA3COFM	CMP3OEN	CMP3MODE	OPA3EN
-------	---------	-------	---------	----------	---------	----------	--------

位	描述	
[7]	<b>CMP3O</b>	比较器 3 的输出结果，只读。
[6]	<b>CMP3HYS</b>	比较器 3 迟滞效应使能位 0: 无迟滞效果（默认） 1: 带 40mv 的迟滞
[5]	<b>SF3EN</b>	运放 source follower 级使能开关 0: 关闭 1: 使能
[4]	<b>OPA3CRS</b>	OPA3 调节模式输入端选择位 0: OPA3 调节模式负端输入 1: OPA3 调节模式正端输入
[3]	<b>OPA3COFM</b>	OPA3 工作模式选择位 0: OPA3 工作在正常模式 1: OPA3 工作在调节模式（运放正端、负端短接）
[2]	<b>CMP3OEN</b>	比较器模式下，比较器输出到 GPIO 使能位 0: 不输出 1: 比较器 3 输出到 PB5
[1]	<b>CMP3MODE</b>	比较器 3 模式选择位 0: 运放模式 1: 比较器模式
[0]	<b>OPA3EN</b>	OPA3 使能位 0: 禁止 OPA3 1: 使能 OPA3

### 16.2.14 比较器 3 控制寄存器 1 (CMP3CON1)

寄存器	地址	R/W	描述	复位后的值
<b>CMP3CON1</b>	0x3D	R/W	比较器 3 控制寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A3P[2:0]			OPA3ADJ[4:0]				

位	描述	
[7:5]	<b>A3P[2:0]</b>	运放正端输入选择 000: 无效 001: PB3 010: VSS 011: DAC10 100: A10

		101: A2O 110: A3O 111: DAC1O, 也就是接到 PB3, 即 PB3 输出了 DAC1 的输出
[4:0]	OPA3ADJ[4:0]	运放 3 失调电压调节位, 默认为 10000,00000 为负失调最大, 11111 为正失调最大

## 16.2.15 比较器 3 控制寄存器 2 (CMP3CON2)

寄存器	地址	R/W	描述	复位后的值
CMP3CON2	0x3E	R/W	比较器 3 控制寄存器 2	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A3N[2:0]			A3SW4	A3SW3	A3SW2	A3SW1	A3SW0

位	描述	
[7:5]	A3N[2:0]	运放负端输入选择 000: 无效 001: PB4 010: VSS 011: DAC0O 100: A1O 101: A2O 110: A3O 111: 无效
[4]	A3SW4	运放 3 通路开关 4 选择位 0: 关闭 1: 开启
[3]	A3SW3	运放 3 通路开关 3 选择位 0: 关闭 1: 开启
[2]	A3SW2	运放 3 通路开关 2 选择位 0: 关闭 1: 开启
[1]	A3SW1	运放 3 通路开关 1 选择位 0: 关闭 1: 开启
[0]	A3SW0	运放 3 通路开关 0 选择位 0: 关闭 1: 开启

### 16.2.16 比较器 3 控制寄存器 3 (CMP3CON3)

寄存器	地址	R/W	描述	复位后的值
CMP3CON3	0x3F	R/W	比较器 3 控制寄存器 3	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
A3GS[2:0]			CMP3INV	CMP3DBS[1:0]		INT3SEL[1:0]	

位	描述	
[7:5]	A3GS[2:0]	<b>运放 PGA 放大倍数选择</b> 000: 内部 PGA 放大倍数为 4 倍 001: 内部 PGA 放大倍数为 8 倍 010: 内部 PGA 放大倍数为 16 倍 011: 内部 PGA 放大倍数为 32 倍 100: 内部 PGA 放大倍数为 64 倍 101~111: 无效
[4]	CMP3INV	<b>比较器 0 反相输出结果</b> 0: 禁止 1: 使能
[3:2]	CMP3DBS[1:0]	<b>比较器 3 输出结果滤波时间</b> 00: 无 01: 4*Fsys/Fcpu 10: 8*Fsys/Fcpu 11: 16*Fsys/Fcpu
[1:0]	INT3SEL[1:0]	<b>比较器 3 输出结果中断触发方式</b> 00: 上升沿触发 01: 下降沿触发 1x: 上升下降沿均触发

### 16.2.17 防水功能控制寄存器 (SHDCON)

寄存器	地址	R/W	描述	复位后的值
SHDCON	0x4A	R/W	防水功能控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留		OPAIS[1:0]		SHDEN[3:0]			

位	描述	
[7:6]	保留	保留
[5:4]	OPAIS[1:0]	四个运放共用相同配置

		00: x1 01: x2 10: x3 11: x4
[3:0]	SHDEN[3:0]	SHDEN 置 1 代表第几个运放的正端接到了 TI 总线上, 可以配置相关寄存器将 OPA 变为防水 BUFFER

## 16.3 运算放大器的使用

### 16.3.1 OPA 使能

将寄存器 CMPxCON0 的第 0 位置 1, 使能运算放大器。将 OPAxEN 置 0, 禁止运算放大器。

### 16.3.2 OPA 端口选择

#### 16.3.2.1 OPA 正端输入

通过设置 CMPxCON1 寄存器的第 7~5 位 AxP[2:0]。

#### 16.3.2.2 OPA 负端输入

通过设置 CMPxCON2 寄存器的第 7~5 位 AxN[2:0]。

#### 16.3.2.3 OPA 输出

运放的输出可以从 AxO 引脚输出。

#### 16.3.2.4 OPA 使用时端口方向设置

OPA 使用相关的 I/O 口时, 将会自动将相应 I/O 变为模拟口。

### 16.3.3 OPA 工作模式

OPA 正常工作时, 用户预判 OPA 正端、负端信号范围, 如果在 0~300mV 之间, 可以

打开运放小信号性能提升使能开关，它位于 CMPxCON0 寄存器的第 5 位 SFxEN。

调节模式工作流程：

1. 使能运放功能；
2. 设置运放进入调节模式；
3. 设置运放调节模式从正端输入或者负端输入，输入端不能悬空；
4. 将调节位 OPAxADJ<4:0>设置成初始值，最大(1FH)或最小(00H)；
5. 延时一段时间，该时间和外部电容参数有关。
6. 读取运放输出；
7. 将调节位自减 1（初始值设置成最大 1FH）或者自加 1（初始值设置成 00H）；
8. 延时；
9. 读取运放输出，是否发生改变，如果没有改变，则继续执行步骤 7；
10. 读取值发生改变，调零结束，将 OPA0COFM 清零，进入正常工作模式。

## 16.4 比较器的使用

通过配置 CMPxCON0~ CMPxCON3 来设置比较器。比较器的输出可通过查询 CMPxCON0 寄存器的第 7 位 CMPxO 来读到。可通过设置 CMPxCON3 寄存器的第 1~0 位 INTxSEL[1:0]来选择比较器的结果进中断的类型。当使能了比较器的中断使能 ECMPx 后，达到设定的中断类型后，就会进去中断服务程序。

## 17 UART

### 17.1 概述

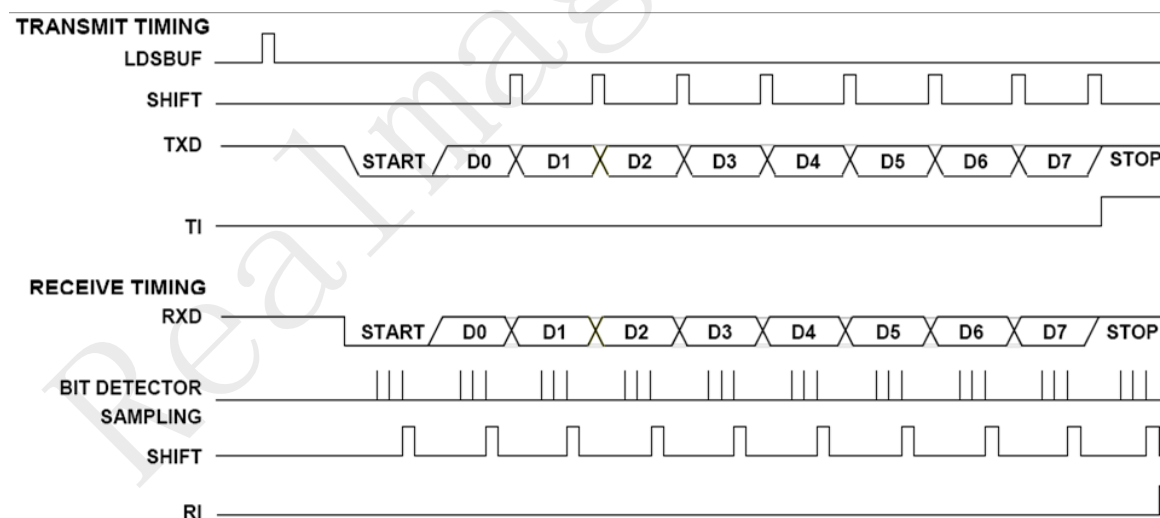
RM1211A 提供一个可编程全双工串行通信接口,该接口能同时进行数据的发送和接收。

串口接收带有接收缓存,意味着在接收的前一个数据在被读取之前,串口就能接收第二个数据。接收和发送都是对 SBUF 进行操作访问,写入 SBUF 数据将直接传到发送寄存器,而读取 SBUF 是访问一个具有独立物理地址的接收寄存器。

### 17.2 功能描述

#### 17.2.1 模式 1

模式 1 (8 位 UART) 为异步全双工的工作方式。异步通讯模式通常用于 PC 间,调制解调器和其它类似接口间通讯。模式 1 下,10 位数据通过 TXD 发送,通过 RXD 接收。10 位数据组成如下:起始位(逻辑 0),8 位数据(最低位在前),停止位(逻辑 1)。波特率由 SOVR[10:0]决定。下图为串口模式 1 发送和接收的时序图。



串口模式1的时序图

任何将 SBUF 作为目标寄存器的写操作都会启动发送,数据位时间与波特率移位时钟是同步的,与对 SBUF 的写操作不同步。起始位首先在 TXD 引脚上移出,然后是 8 位数据位。在发送移位寄存器中的所有 8 位数据都发送完后,停止位在 TXD 引脚上移出,在停止位发出的同时 TI (SCON.1) 标志置 1。TI (SCON.1) 可以软件清掉。

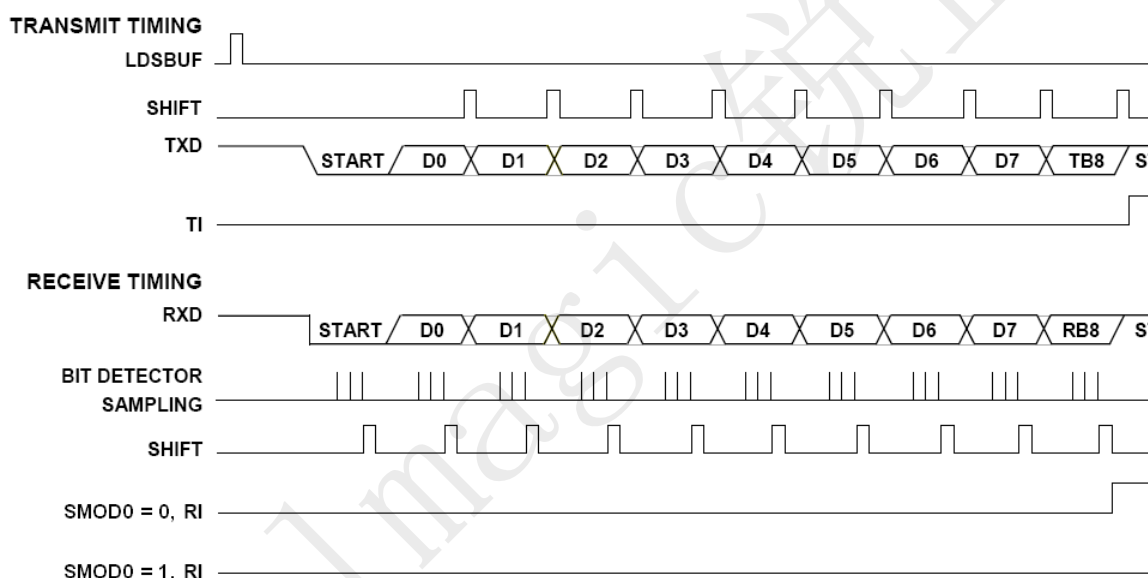
当 RXD 引脚检测到下降沿时串行口开始接收串行数据。CPU 在 RXD 数据位的正中进行采样。如果所接收的第一位不是 0,说明这位不是一帧数据的起始位,该位被忽略,接

收电路被复位，等待 RXD 引脚上另一个下降沿的到来。若起始位有效，则移入移位寄存器，并接着移入其它位到移位寄存器。8 个数据位和 1 个停止位移入之后，移位寄存器的内容被装入 SBUF，RI (SCON.0) 置 1。

在接收停止位后，接收器将重新去探测 RXD 端的另一个下降沿。用户必须用软件清 RI (SCON.0)，然后才能再次接收。

## 17.2.2 模式 2

模式 2 (9 位 UART) 为全双工异步通信，与模式 1 不同的是，模式 2 是 11 位收发。数据由起始位 (逻辑 0)，8 位数据 (最低位在前)，第 9 位数据 (TB8 或 RB8) 和停止位 (逻辑 1) 组成。第 9 位做奇偶校验位或多机通信时用来区分数据和地址。波特率由 SOVR[10:0] 决定。下图指示串口模式 2 的传输时序。



串口模式2的传输时序

在数据传送时，第 9 数据位 (寄存器 SCON 中的 TB8) 可以写 0 或 1，例如，可写入奇偶位 P，或用作多机通讯中的数据/地址标志位。当接收到数据时，第 9 数据位移入 RB8 而停止位不保存。

任何将 SBUF 作为目标寄存器的写操作都会启动发送，同时也将 TB8 载入到发送移位寄存器的第 9 位中。数据位时间与波特率移位时钟是同步的，与对 SBUF 的写操作不同步。起始位首先在 TXD 引脚上移出，然后是 9 位数据。在发送转换寄存器中的所有 9 位数据都发送完后，停止位在 TXD 引脚上移出，在停止位发出的同时 TI (SCON.1) 标志置 1。TI (SCON.1) 可以软件清掉。

当 RXD 引脚检测到下降沿时串行口开始接收串行数据。CPU 在 RXD 数据位的正中进行采样接收。如果所接收的第一位不是 0，说明这位不是一帧数据的起始位，该位被忽略，接收电路被复位，等待 RXD 引脚上另一个下降沿的到来。若起始位有效，则移入移位寄存器，并接着移入其它位到移位寄存器。9 个数据位和 1 个停止位移入之后，移位寄存器的内容被分别装入 SBUF 和 RB8 中，RI (SCON.0) 置 1。



在接收停止位后，接收器将重新去探测 RXD 端的另一个下降沿。用户必须用软件清 RI (SCON.0)，然后才能再次接收。

### 17.2.3 波特率

UART 实质上是一个 16 位递增计数器，其计数时钟为 CLKCPU。用户软件配置寄存器 SOVRH 和 SOVRL 来软件配置计数器的溢出值，从而实现需要的波特率。

$$\text{波特率计算公式: 波特率} = \frac{FSYS}{1+SOVR[10:0]}。$$

## 17.3 寄存器描述

### 17.3.1 串口控制寄存器 (SCON)

寄存器	地址	R/W	描述	复位后的值
SCON	0x10	R/W	串口 0 控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
-	-	B8EN	REN	TB8	RB8	TI	RI

位	描述	
[7:6]	保留	保留
[5]	B8EN	串口第 9 位使能位 0: 关闭串口第 9 位 1: 使能串口第 9 位
[4]	REN	串口使能位 0: 关闭串口 1: 使能串口
[3]	TB8	串口第 9 位发送位 串口要被发送的第九位数据
[2]	RB8	串口第 9 位接收位 串口接收到的第九位数据
[1]	TI	串口发送中断标志位 发送中断标志: 该标志由硬件在发送完 8 位数据后置 1。当该位中断使能, 发生中断后会转至中断子程序。该位必须由软件来清除。
[0]	RI	串口接收中断标志 该标志由硬件置位。当串口中断使能, 该位置位会转跳到中断服务子程序运行。该位必须由软件来清除。

### 17.3.2 串口数据缓存寄存器 (SBUF)

寄存器	地址	R/W	描述	复位后的值
<b>SBUF</b>	0x11	R/W	串口数据缓存寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>SBUF[7:0]</b>							

位	描述	
[7:0]	<b>SBUF[7:0]</b>	<b>串口数据缓存</b> 串口接收或发送的数据都放在这个寄存器中。实际上该地址上有 2 个独立的 8 位寄存器。一个用于接收数据，一个用于发送数据。对它进行读操作将会接收串行数据，对它进行写操作则发送串行数据。 每次向 SBUF 写入一字节数据，启动一次发送。

### 17.3.3 串口波特率低字节 (SOVRL)

寄存器	地址	R/W	描述	复位后的值
<b>SOVRL</b>	0x12	R/W	串口波特率低字节	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>SOVRL[7:0]</b>							

位	描述	
[7:0]	<b>SOVRL[7:0]</b>	串口波特率低字节

### 17.3.4 串口波特率高字节 (SOVRH)

寄存器	地址	R/W	描述	复位后的值
<b>SOVRH</b>	0x13	R/W	串口波特率高字节	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
-					<b>SOVR[10:8]</b>		

位	描述	
[7:3]	保留	保留
[2:0]	<b>SOVR[10:8]</b>	串口波特率高字节

## 18 触摸按键(TOUCH KEY)

### 18.1 概述

触摸按键提供了方便简单而可靠的方法来实现手指触摸的检测。在触摸应用中，并不需要一个外接电容组件。RM1211A 集成了硬件数字滤波器，可以实现任意 4 键自动检测，最多支持 18 通道触摸按键检测。RM1211A 也可以实现水位检测功能。

RealMagiC 锐盟

## 19 MTP 和 EEPROM 的自编程

### 19.1 特性

RM1211A 可以对 MTP 自编程操作，写数据时，通过 IAPADR[11:0]来选择地址，可以自编程的 MTP 地址空间从 0H 到 FFFH，IAPADR[11:0]=0 对应 MTP 的地址 0H。IAPDAT[15:0]为 16 位数据。

**注意：**

- 1、MTP 改写寿命为 1000 次，注意控制改写次数，防止 MTP ROM 失效。
- 2、为了保证正常读写指令，要求在读写 MTP 存储器或者 EEPROM 存储器时把总中断使能关闭。

RM1211A 芯片内置有 128x8 bit 的 EEPROM 数据存储器，该存储器可以提供 20000 次的烧写。EEPROM 数据存储器的读写操作以 1 个字节为单位，每次读写 1 个字节。

EEPROM 数据存储器的读写操作与 MTP 自编程的操作相似。因为 EEPROM 数据存储器的规格为 128x8 bit，所以对应的读写操作地址只需要写入至 IAPADRL 寄存器中（IAPADRH 不起作用），EEPROM 数据存储器的读写数据对应的寄存器为 IAPDATL 寄存器（不需要用到 IAPDATH 寄存器）。

注：当读取 EEPROM 数据时，读出的数据更新至 IAPDATL 寄存器，同时 IAPDATH 寄存器的值会被置为 0x00。

**注意：**

- 1、EEPROM 改写寿命为 20000 次，注意控制改写次数，防止 EEPROM 失效。
- 2、为了保证正常读写指令，要求在读写 MTP 存储器或者 EEPROM 存储器时把总中断使能关闭。

### 19.2 寄存器

#### 19.2.1 IAP 执行寄存器（IAPTRIG）

寄存器	地址	R/W	描述	复位后的值
IAPTRIG	0x10	R/W	IAP 执行寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
IAPTRIG[7:0]							

位	描述
[7:0]	IAPTRIG[7:0] IAPTIR 连着写入 0xAA, 0xBB, 0xCC, 进入到 IAP MODE 注意：如果往 IAPTIR 写入其他任何值，则进入 IAPLOCK 状态

### 19.2.2 IAP 控制寄存器 (IAPCON)

寄存器	地址	R/W	描述	复位后的值
IAPCON	0x11	R/W	IAP 控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留	IAPLOCK	WAITEN	STATICEN	WAITSEL[1:0]		IAPRD	IAPWR

位	描述													
[7]	保留	保留												
[6]	IAPLOCK	此位为 1 时, EEPROM/MTP 编程处于锁定状态, 需要软件对该位进行清零操作或者重新上电才可以继续进行编程写入 EEPROM/MTP 操作。只有发生不按文档规定的顺序值写入 IAPTIR 时, 此位才会变为 1。正常正确的操作都不会使得改位为 1。												
[5]	WAITEN	<p><b>使能写等待超时功能, 与 WAITSEL 配合使用。</b></p> <p>此控制位与 MTP 自编程写入/EEPROM 写入操作相关。当 MTP 自编程/EEPROM 写入操作的过程中, CPU 处于等待状态直至到 MTP 自编程写入/EEPROM 写入操作完成后, CPU 再会继续执行下一条指令。当使能 MTP 自编程/EEPROM 写入操作超时退出时(WAITEN=1), 并且 CPU 等待超时, 芯片将会放弃写入操作, 然后 CPU 再会继续执行下一条指令。WAITEN 控制位需要与 WAITSEL 一起配合使用。</p> <p>0: 关闭 MTP 自编程/EEPROM 写入操作超时退出 1: 使能 MTP 自编程/EEPROM 写入操作超时退出</p>												
[4]	STATICEN	<p><b>MTP 的 STATICEN 脚</b></p> <p>此 bit 与代码选项的 ISAVB_EN 配合使用, 可以节省 MTP 低速下的功耗。此 bit 默认为 1, 即不开低速省功耗。500KHz/128KHz/32KHz 下建议配置为 STATICEN=0, 开启低速省功耗模式。</p> <table border="1" data-bbox="502 1294 1353 1594"> <thead> <tr> <th>ISAVB</th> <th>STATICEN</th> <th>功能</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>省电模式: (500KHz &amp; 32KHz)</td> </tr> <tr> <td>0</td> <td>1</td> <td>工作电压: 8MHz @4.5V&lt;VDD&lt;5.5V 4MHz @2.5V&lt;VDD&lt;5.5V 1MHz @2.0V&lt;VDD&lt;5.5V</td> </tr> <tr> <td>1</td> <td>X</td> <td>高速模式 (16MHz)</td> </tr> </tbody> </table>	ISAVB	STATICEN	功能	0	0	省电模式: (500KHz & 32KHz)	0	1	工作电压: 8MHz @4.5V<VDD<5.5V 4MHz @2.5V<VDD<5.5V 1MHz @2.0V<VDD<5.5V	1	X	高速模式 (16MHz)
ISAVB	STATICEN	功能												
0	0	省电模式: (500KHz & 32KHz)												
0	1	工作电压: 8MHz @4.5V<VDD<5.5V 4MHz @2.5V<VDD<5.5V 1MHz @2.0V<VDD<5.5V												
1	X	高速模式 (16MHz)												
[3:2]	WAITSEL[1:0]	<p><b>编程数据等待时间选择位</b></p> <p>00: 等待 16ms 01: 等待 32ms 10: 等待 64ms 11: 等待 128ms (默认)</p>												
[1]	IAPRD	MTP 编程读出操作使能位, 高有效, 读取完成, 硬件自动清 0												
[0]	IAPWR	MTP 编程写入操作使能位, 高有效, 写操作完成, 硬件自动清 0												

### 19.2.3 IAP 地址高字节 (IAPADRH)

寄存器	地址	R/W	描述	复位后的值
IAPADRH	0x12	R/W	IAP 地址高字节	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留				IAPADR[11:8]			

位	描述	
[7:4]	保留	保留
[3:0]	IAPADR[11:8]	IAP 地址高字节

### 19.2.4 IAP 地址低字节 (IAPADRL)

寄存器	地址	R/W	描述	复位后的值
IAPADRL	0x13	R/W	IAP 地址低字节	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
IAPADR[7:0]							

位	描述	
[7:0]	IAPADR[7:0]	IAP 地址低字节

### 19.2.5 IAP 数据高字节 (IAPDATH)

寄存器	地址	R/W	描述	复位后的值
IAPDATH	0x14	R/W	IAP 数据高字节	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
IAPDAT[15:8]							

位	描述	
[7:0]	IAPDAT[15:8]	IAP 数据高字节

### 19.2.6 IAP 数据低字节 (IAPDATL)

寄存器	地址	R/W	描述	复位后的值
-----	----	-----	----	-------

<b>IAPDATL</b>	0x15	R/W	IAP 数据低字节	00000000
----------------	------	-----	-----------	----------

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>IAPDAT[7:0]</b>							

位	描述	
[7:0]	<b>IAPDAT[7:0]</b>	<b>IAP 数据低字节</b>

### 19.2.7 E2PROM 控制寄存器 (E2CON)

寄存器	地址	R/W	描述	复位后的值
<b>E2CON</b>	0x16	R/W	E2PROM 控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留						<b>E2RD</b>	<b>E2WR</b>

位	描述	
[7:2]	保留	保留
[1]	<b>E2RD</b>	<b>E2PROM 编程读出操作使能位，高有效，读取完成，硬件自动清 0</b>
[0]	<b>E2WR</b>	<b>E2PROM 编程写入操作使能位，高有效，写操作完成，硬件自动清 0</b>

MTP 写入操作：

- (1) 打开 WCK 时钟，也就是设置寄存器 OSCM.2 为 1。
- (2) 对 IAPTRIG 寄存器依次写入 0xAA, 0xBB, 0xCC，进入到 IAP 模式。如果写入顺序或写入值不对，则 IAP 就会进入 LOCK 状态，必须重新上电，才可以进行 IAP 操作。
- (3) 设置 IAPADR[11:0]，配置写入 ROM 地址，IAPADR[11:0]=0，对应 ROM 地址 0H。
- (4) 设置 IAPDAT[15:0]，配置写入 ROM 数据。
- (5) 把 IAPWR 置 1，CPU 将 IAPDAT[15:0]数据写入 IAPADR[11:0]对应的 ROM 地址。
- (6) 插入 4 个 NOP 指令。
- (7) 如果要继续写操作，则回到步骤 (3)；如果要退出写操作，则对 IAPTRIG 写入 0x00,即可退出 IAP 模式。

MTP 读出操作：

- (1) 设置 IAPADR[11:0]，配置读出 ROM 地址，IAPADR[11:0]=0，对应 ROM 地址 0H。
- (2) 把 IAPRD 置 1。
- (3) 插入 4 个 NOP 指令。
- (4) 从 IAPDAT[15:0]中读取数据。CPU 将从 IAPADR[11:0]对应的 ROM 地址的数据读入到 IAPDAT[15:0]。
- (5) 如果要继续读操作，则回到步骤 (1)。否则，对 IAPTRIG 写入 0，便可以退出读操作。

## EEPROM 写入操作:

- (1) 打开 WCK 时钟使能, 也就是设置寄存器 OSCM.2 为 1。
- (2) 对 IAPTRIG 连着写入 0xAA, 0xBB, 0xCC, 进入到 IAP 模式。
- (3) 设置 EEPROM 写入的操作地址 (IAPADRL [6:0])。
- (4) 设置 EEPROM 写入的操作数据 (IAPDATL [7:0])。
- (5) 把 E2WR 置 1。
- (6) 插入 4 个 NOP 指令。
- (7) 如果要进行写操作, 则回到步骤(3)开始。如果要退出写操作, 则对 IAPTRIG 写入 0x00,即可退出 IAP 模式。

## EEPROM 读出操作:

- (1) 设置 IAP 读出的操作地址 (IAPADRL [6:0])。
- (2) 把 E2RD 置 1。
- (3) 插入 4 个 NOP 指令。
- (4) 取出对应的 EEPROM 读出的数据 (比如, 将 IAPDATL 缓存)。
- (5) 如果要继续进行读操作, 则回到步骤 (1) 开始。否则, 对 IAPTRIG 写入 0, 便可以退出读操作。

**为了保证 CPU 能正确读取程序, 对 MTP/EEPROM 读写时需把总中断使能关闭。**



## 20 时钟 Trimming

### 20.1 ICK

芯片内部有一个高速的 ICK 时钟，频率是 16MHz。在芯片出厂后，我们会将此时钟 *trimming* 后，使得 ICK 在 16MHz。后面可通过调整 ICK\_TRIM 这个寄存器的值来改变芯片的 ICK 频率。当 ICK\_TRIM 这个寄存器的值变大时，ICK 频率会变小；反之，当 ICK\_TRIM 这个寄存器的值变小时，ICK 频率会变大。

#### 20.1.1 ICK 配置寄存器(ICK\_TRIM)

寄存器	地址	R/W	描述	复位后的值
ICK_TRIM	0x28	R/W	ICK 配置寄存器	10000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ICK_TRIM [7:0]							

位	描述	
[7:0]	ICK_TRIM [7:0]	系统时钟修调控制寄存器，也受代码选项控制

### 20.2 WCK

芯片内部有一个低速的 WCK 时钟，频率是 128KHz。在芯片出厂后，我们会将此时钟 *trimming* 后，使得 WCK 在 128KHz。后面可通过调整 WCK\_TRIM 这个寄存器的值来改变芯片的 WCK 频率。当 WCK\_TRIM 这个寄存器的值变大时，WCK 频率会变小；反之，当 WCK\_TRIM 这个寄存器的值变小时，WCK 频率会变大。

#### 20.2.1 WCK 配置寄存器(WCK\_TRIM)

寄存器	地址	R/W	描述	复位后的值
WCK_TRIM	0x29	R/W	WCK 配置寄存器	10000000

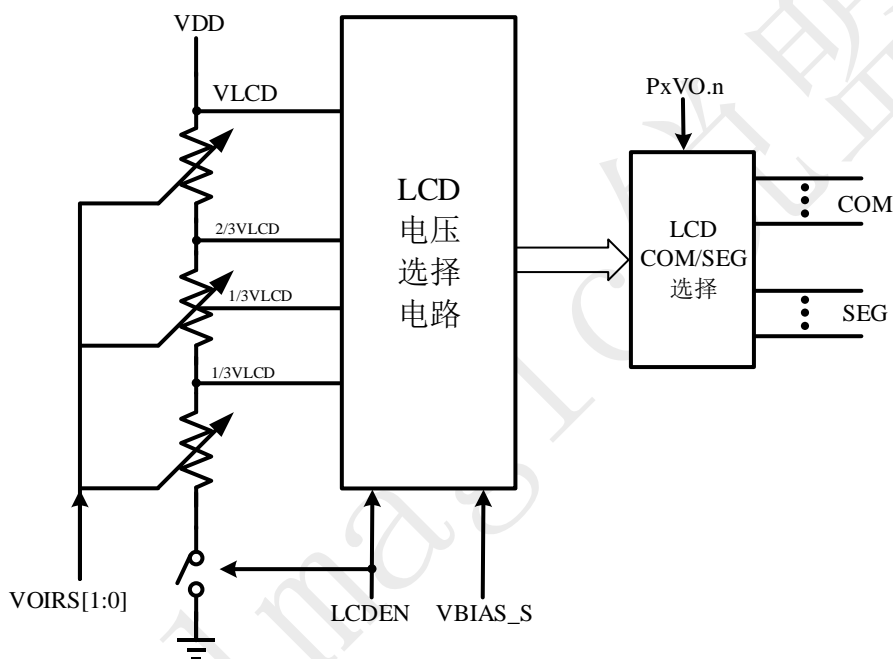
BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
WCK_TRIM [7:0]							

位	描述	
[7:0]	WCK_TRIM [7:0]	看门狗时钟修调控制寄存器，也受代码选项控制

## 21 LCD

### 21.1 概述

RM1211A 所有 IO 口通过配置寄存器都可以作为 LCD 的 COM 或 SEG，每个 IO 除了正常 IO 的功能外，还可以输出两组可选电压：1/3VDD，2/3VDD 和 1/2VDD。用户可根据使用情况，随意选择任意 IO 作为 LCD 驱动的 COM 或者 SEG，LCD 的驱动能力可配置，LCD 控制信号（COM 和 SEG）由软件程序实现。



### 21.2 LCD 使用说明

LCD 的 COM/SEG 的驱动信号为交流模拟信号，LCD 像素点的显示对比度取决于此显示点上的 COM 与 SEG 电压( $V_{com}-V_{seg}$  /  $V_{seg}-V_{com}$ )，当此电压差大于 LCD 的饱和电压就能打开此点的显示，小于 LCD 的饱和电压就能关闭此 LCD 点的显示。这需要每个 COM/SEG 在一个扫描周期内根据用户需要在选通和非选通电压之间切换。

一个完整的 LCD 波形周期包含两个 Frame，即 Frame0 和 Frame1，通过 Frame0 和 Frame1 实现交流驱动信号。每个 Frame 时间可由软件用定时器控制实现。

#### Frame0

在 Frame0 中，COM 信号输出可以是 VDD，或是  $VBIAS=1/3VDD$  ( $1/2VDD$ )；  
 在 Frame0 中，SEG 信号输出可以是 GND，或是  $VBIAS=2/3VDD$  ( $1/2VDD$ )。

#### Frame1

在 Frame1 中，COM 信号输出可以是 GND，或是  $VBIAS=2/3VDD$  ( $1/2VDD$ )；

在 Frame1 中，SEG 信号输出可以是 VDD，或是 VBIAS=1/3VDD (1/2VDD)。

通过软件在定时器定时到期后设定 FRAME 位及相应的 I/O 寄存器来决定 COM 口目前输出的是 VDD，GND 或 VBIAS。

通过软件在定时器定时到期后设定 FRAME 位及相应的 I/O 寄存器来决定 SEG 口目前输出是 VDD，GND 或 VBIAS (在 1/2bias 时，SEG 只输出 VDD 或 GND)。

下面的波形图显示了一个利用应用程序产生的典型 1/2Bias LCD 波形。COMn 和 SEGm 引脚上所产生的 COM 和 SEG 信号极性通过相应的 I/O 相关寄存器位来产生。

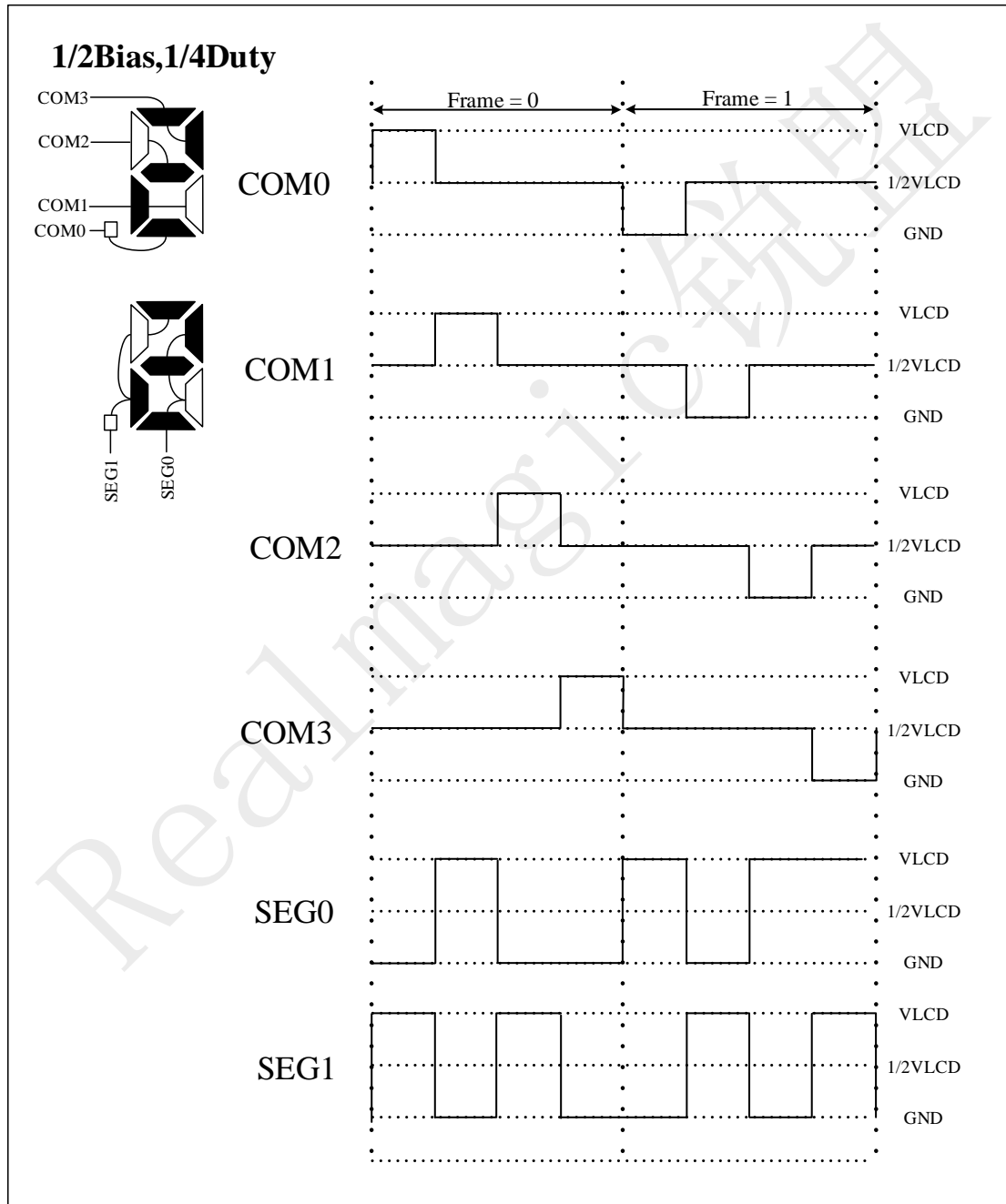


图 1/2LCD 波形图

1/2Bias, 1/4Duty LCD COM0~COM3 一轮扫描的 COM&SEG 电平和显示关系如下表:

Frame	Time	COM0	COM1	COM2	COM3	SEG0 点亮	SEG1 不点亮
-------	------	------	------	------	------	---------	----------

Frame0	t0	VLCD	1/2 VLCD	1/2 VLCD	1/2 VLCD	GND	VLCD
	t1	1/2VLCD	VLCD	1/2 VLCD	1/2 VLCD	GND	VLCD
	t2	1/2 VLCD	1/2 VLCD	V VLCD	1/2 VLCD	GND	VLCD
	t3	1/2 VLCD	1/2 VLCD	1/2 VLCD	VLCD	GND	VLCD
Frame1	t4	GND	1/2 VLCD	1/2 VLCD	1/2 VLCD	VLCD	GND
	t5	1/2 VLCD	GND	1/2 VLCD	1/2 VLCD	VLCD	GND
	t6	1/2 VLCD	1/2 VLCD	GND	1/2 VLCD	VLCD	GND
	t7	1/2 VLCD	1/2 VLCD	1/2 VLCD	GND	VLCD	GND

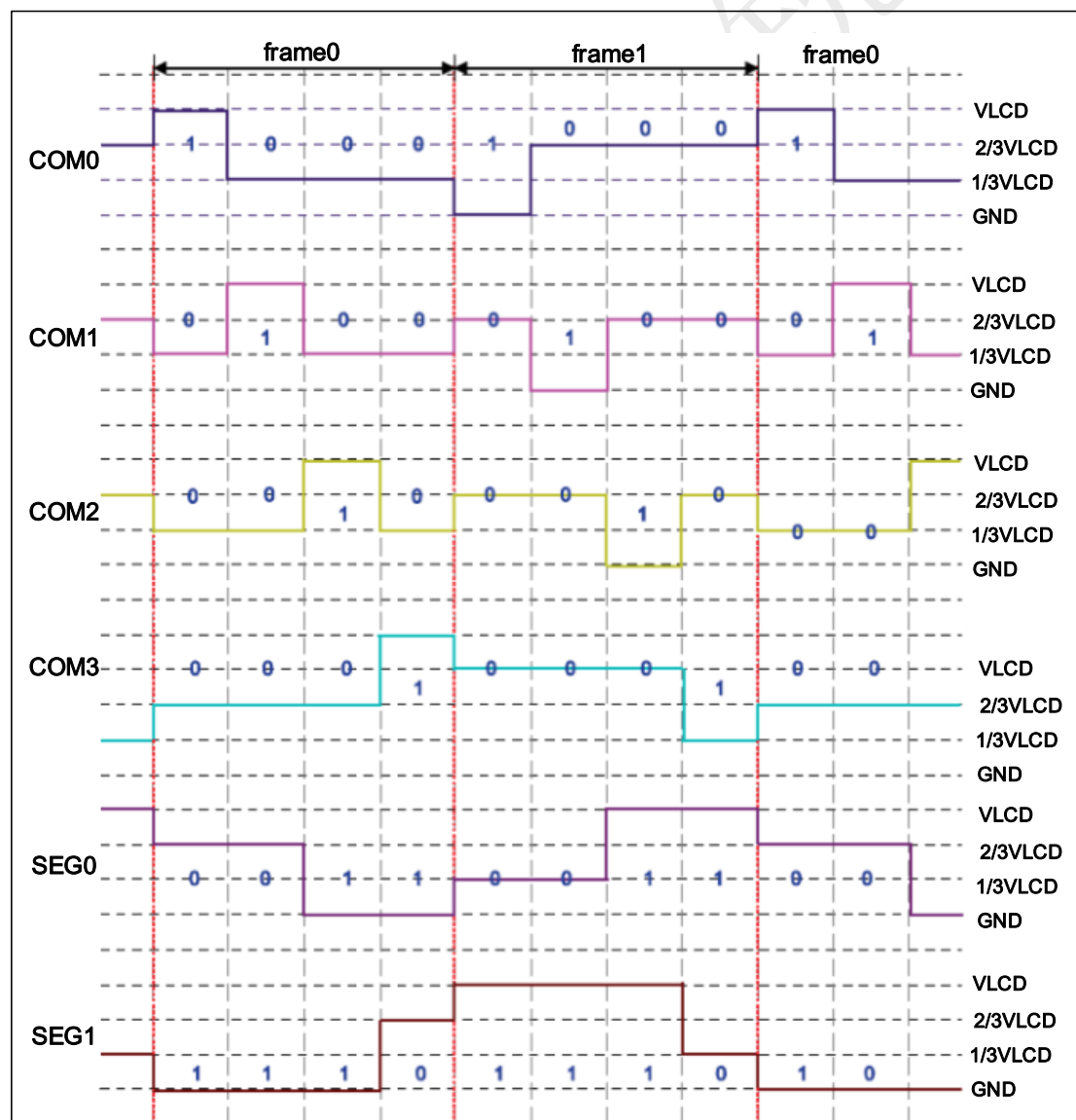
此时如果选择的 1/2BIAS LCD 电压输出口输出的具体电压跟寄存器 Px\_OUT、Px\_VO 具体 bit 的值有关，即

PORTx.y=0, PxVOy =0 时，设定 LCD 电压输出口输出电压为 GND；

PORTx.y=0, PxVOy =1 时，设定 LCD 电压输出口输出电压为 1/2VLCD；

PORTx.y=1, PxVOy =0 时，设定 LCD 电压输出口输出电压为 VLCD。

下面的波形图显示了一个利用应用程序产生的典型 1/3Bias LCD 波形。COMn 和 SEGm 引脚上所产生的 COM 和 SEG 信号极性通过相应的相应的 I/O 寄存器位来产生。



**图 1/3LCD 波形图**

此时如果选择的 1/3BIAS LCD 电压输出输出的具体电压跟寄存器 Px\_OUT、Px\_PU、PA\_PD 具体 bit 的值有关，即

- Px\_OUT.y=0, PxVOy =0 时，设定 LCD 电压输出输出电压为 GND;
- Px\_OUT.y=1, PxVOy =1 时，设定 LCD 电压输出输出电压为 1/3VLCD;
- Px\_OUT.y=0, PxVOy =1 时，设定 LCD 电压输出输出电压为 2/3VLCD;
- Px\_OUT.y=1, PxVOy =0 时，设定 LCD 电压输出输出电压为 VLCD。

## 21.3 寄存器

### 21.3.1 LCD 模块控制寄存器 (LCDCON)

寄存器	地址	R/W	描述	复位后的值
LCDCON	0x42	R/W	LCD 模块控制寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留		LCDEN	VBIASS	VOIRS[1:0]		PCVO [1:0]	

位	描述	
[7:6]	保留	保留
[5]	LCDEN	<b>LCD 使能位</b> 0: 关闭 LCD 1: 使能 LCD
[4]	VBIASS	<b>LCD 电压输出 BIAS 选择位</b> 0: PxVOy=1 时, LCD 电压输出输出 1/2VDD" 1: PxVOy=1 时, Px_OUTy=0, LCD 电压输出输出, 2/3VDD PxVOy=1 时, Px_OUTy=1, LCD 电压输出输出, 1/3VDD
[3:2]	VOIRS[1:0]	<b>LCD 电压输出分压电阻选择</b> 00: 关闭内部分压电阻 (省电) 01: 设定内部分压电阻为 12.5K 10: 设定内部分压电阻为 37.5K 11: 设定内部分压电阻为 87.5K
[1:0]	PCVO[1:0]	<b>PCx 口输出选择</b> 0: 普通 IO 口 1: PC 口输出 LCD 电压 (1/2VDD、1/3VDD、2/3VDD 中的一种)

### 21.3.2 PA 口输出选择寄存器 (PAVO)

寄存器	地址	R/W	描述	复位后的值
-----	----	-----	----	-------

<b>PAVO</b>	0x43	R/W	PA 口输出选择寄存器	00000000
-------------	------	-----	-------------	----------

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>PAVO[7:0]</b>							

D

位	描述	
[7:0]	<b>PAVO[7:0]</b>	<b>PA 口输出选择位</b> 0: PA.x 为普通 IO 1: PA.x 输出 LCD 电压 (1/2VLCD、1/3VLCD、2/3VLCD、VLCD、GND)

### 21.3.3 PB 口输出选择寄存器 (PBVO)

寄存器	地址	R/W	描述	复位后的值
<b>PBVO</b>	0x44	R/W	PB 口输出选择寄存器	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
<b>PBVO[6:0]</b>							

位	描述	
[7:0]	<b>PBVO[6:0]</b>	<b>PB 口输出选择位</b> 0: PB.x 为普通 IO 1: PB.x 输出 LCD 电压 (1/2VLCD、1/3VLCD、2/3VLCD、VLCD、GND)

## 22 1 DAC

### 22.1 概述

RM1211A 内嵌 2 个 10 位的 R-2R 型的数字模拟转换器 (DAC)。

### 22.2 寄存器

#### 22.2.1 DAC0 控制寄存器 0 (DAC0CON0)

寄存器	地址	R/W	描述	复位后的值
<b>DAC0CON0</b>	0x45	R/W	DAC0 控制寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
SLV0[7:0]							

位	描述	
[7:0]	SLV0[7:0]	DAC0 输出电压的低 8 位 选择 DAC0 输出电压, DAC 基准参考电压为 VDD 或 DACV $DAC0OUT=SVL0[9:0]/1024 * SVRF0$

### 22.2.2 DAC0 控制寄存器 1 (DAC0CON1)

寄存器	地址	R/W	描述	复位后的值
DAC0CON	0x46	R/W	DAC0 控制寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留		DAC0EN	SVRF0	保留		SLV0[9:8]	

位	描述	
[7:6]	保留	保留
[5]	DAC0EN	DAC0 使能位 0: 禁止 DAC0 1: 使能 DAC0
[4]	SVRF0	选择参考电压
[3:2]	保留	保留
[1:0]	SLV0[9:8]	DAC0 输出电压的高 2 位

### 22.2.3 DAC1 控制寄存器 0 (DAC1CON0)

寄存器	地址	R/W	描述	复位后的值
DAC1CON0	0x47	R/W	DAC1 控制寄存器 0	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
SLV1[7:0]							

位	描述	
[7:0]	SLV11[7:0]	DAC1 输出电压的低 8 位 选择 DAC1 输出电压, DAC 基准参考电压为 VDD 或 DACV $DAC1OUT=SVL1[9:0]/1024 * SVRF1$

## 22.2.4 DAC1 控制寄存器 1 (DAC1CON1)

寄存器	地址	R/W	描述	复位后的值
DAC1CON	0x48	R/W	DAC1 控制寄存器 1	00000000

BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
保留		DAC1EN	SVRF1	保留		SLV1[9:8]	

位	描述	
[7:6]	保留	保留
[5]	DAC1EN	DAC1 使能位 0: 禁止 DAC1 1: 使能 DAC1
[4]	SVRF1	选择参考电压
[3:2]	保留	保留
[1:0]	SLV1[9:8]	DAC1 输出电压的高 2 位

## 22.3 DAC 的使用

### 22.3.1 DAC 使能

将寄存器 DAC0CON1 和 DAC1CON1 的第 5 位 DAC0EN、DAC1EN 置 1, 分别使能 DAC0 和 DAC1。将 DAC0EN、DAC1EN 置 0, 禁止数模转换器。

### 22.3.2 DAC 参考电压来源选择

通过设置寄存器 DAC0CON1 和 DAC1CON1 的第 4 位 SVRF0、SVRF1, 可以分别选择 DAC0 和 DAC1 的参考电压来源:

- 0: 参考电压来自 VDD
- 1: 参考电压来自内部参考 (为 ADC 中的内嵌带隙基准电压 1V/2V/3V/4V)

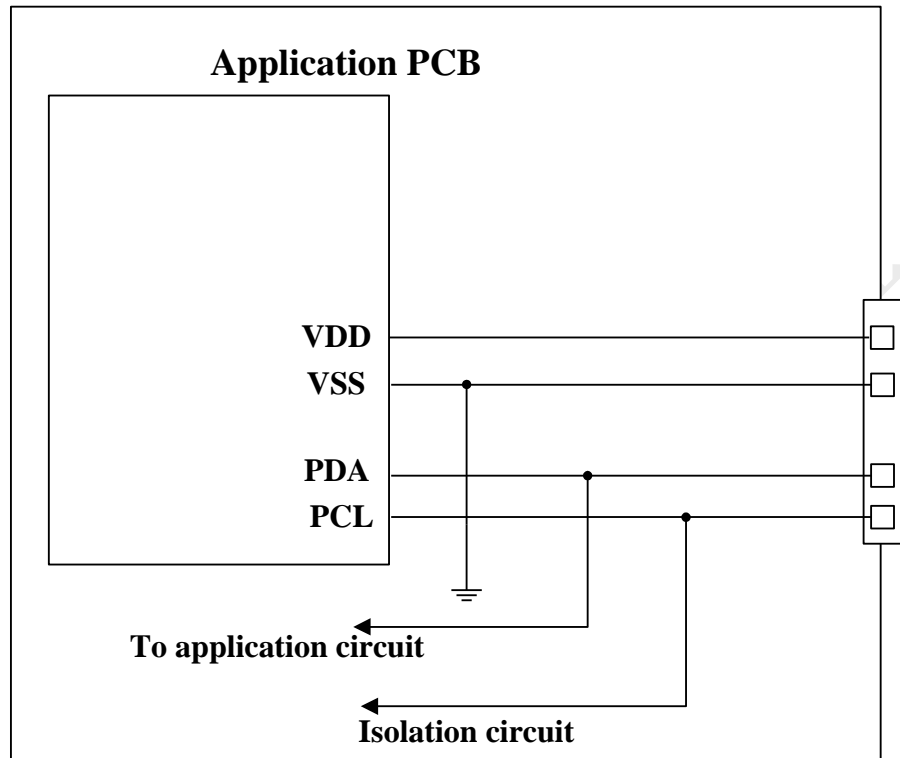
### 22.3.3 DAC 输出

DAC0 的输出可以通过设置寄存器 DAC0CON0(SVL0<7:0>)和 DAC0CON1 的第 0~1 位(SVL0<9:8>)来配置:  $DAC0OUT = \frac{SVL0<9:0>}{1024} \times VREF0$

同理, DAC1 的输出可以通过设置寄存器 DAC1CON0 (SVL1<7:0>) 和 DAC1CON1 的第 0~1 位 (SVL1<9:8>) 来配置:  $DAC1OUT = \frac{SVL1<9:0>}{1024} \times VREF1$



## 23 烧录



烧写器的接口：

图3 烧写器接口图

表 14 接口说明表

端口名称	型式	说明	备注
VDD	输入	电源正端	
VSS	输入	电源负端	
PDA	输入/输出	PA.3 端口，数据信号	
PCL	输入	PA.2 端口，时钟信号	

## 24 指令集

指令	指令格式	描述	C	AC	Z	周期
赋值	MOV A, M	$A \leftarrow M$	-	-	-	1
	MOV M, A	$M \leftarrow A$	-	-	-	1
	MOV A, I	$A \leftarrow I$	-	-	-	1
算术	ADD A, I	$A \leftarrow A + I$	√	√	√	1
	ADD A, M	$A \leftarrow A + M$	√	√	√	1
	ADD M, A	$M \leftarrow A + M$	√	√	√	1
	ADC A, M	$A \leftarrow A + M + C$	√	√	√	1
	ADC M, A	$M \leftarrow A + M + C$	√	√	√	1
	SUB A, I	$A \leftarrow A - I$	√	√	√	1
	SUB A, M	$A \leftarrow A - M$	√	√	√	1
	SUB M, A	$M \leftarrow A - M$	√	√	√	1
	SBC A, M	$A \leftarrow A - M - /C$	√	√	√	1
SBC M, A	$M \leftarrow A - M - /C$	√	√	√	1	
逻辑	AND A, I	$A \leftarrow A \& I$	-	-	√	1
	AND A, M	$A \leftarrow A \& M$	-	-	√	1
	AND M, A	$M \leftarrow A \& M$	-	-	√	1
	CPLM M	$M \leftarrow \sim(M)$	-	-	√	1
	OR A, I	$A \leftarrow A   I$	-	-	√	1
	OR A, M	$A \leftarrow A   M$	-	-	√	1
	OR M, A	$M \leftarrow A   M$	-	-	√	1
	XOR A, I	$A \leftarrow A \wedge I$	-	-	√	1
	XOR A, M	$A \leftarrow A \wedge M$	-	-	√	1
	XOR M, A	$M \leftarrow A \wedge M$	-	-	√	1
BCPL M.b	$\sim(M.b)$	-	-	-	1	
递增和递减	INC M	$A \leftarrow M + 1$	-	-	√	1
	INCM M	$M \leftarrow M + 1$	-	-	√	1
	INCS M	$A \leftarrow M + 1$ , 如果A=0, 则跳过下一条指令	-	-	-	1 or 2
	INCMS M	$M \leftarrow M + 1$ , 如果M=0, 则跳过下一条指令	-	-	-	1 or 2
	DEC M	$A \leftarrow M - 1$	-	-	√	1
	DECM M	$M \leftarrow M - 1$	-	-	√	1
	DECS M	$A \leftarrow M - 1$ , 如果A=0, 则跳过下一条指令	-	-	-	1 or 2
DECMS M	$M \leftarrow M - 1$ , 如果M=0, 则跳过下一条指令	-	-	-	1 or 2	

位操作	RLC M	$A \leftarrow M$ 带进位左移1位	√	-	-	1
	RLCM M	$M \leftarrow M$ 带进位左移1位	√	-	-	1
	RRC M	$A \leftarrow M$ 带进位右移1位	√	-	-	1
	RRCM M	$M \leftarrow M$ 带进位右移1位	√	-	-	1
	BCLR M.b	$M.b \leftarrow 0$	-	-	-	1
	BSET M.b	$M.b \leftarrow 1$	-	-	-	1
	SWAP M	$A(b3 \sim b0, b7 \sim b4) \leftarrow M(b7 \sim b4, b3 \sim b0)$	-	-	-	1
	SWAPM M	$M(b3 \sim b0, b7 \sim b4) \leftarrow M(b7 \sim b4, b3 \sim b0)$	-	-	-	1
跳转和返回	JMP addr	跳转指令, $PC \leftarrow \text{addr}$ , addr为13bit长度	-	-	-	2
	BTSZ M.b	如果 $M.b = 0$ , 则跳过下一条指令	-	-	-	1 or 2
	BTSNZ M.b	如果 $M.b = 1$ , 则跳过下一条指令	-	-	-	1 or 2
	CALL addr	子程序调用指令, $\text{stack} \leftarrow PC, PC \leftarrow \text{addr}$	-	-	-	2
	RET	子程序跳出指令, $PC \leftarrow \text{Stack}$	-	-	-	5
	RETI	中断处理程序跳出指令, $PC \leftarrow \text{Stack}$ , 使能全局中断控制位	-	-	-	5
	CMPRS A, M	如果 $A=M$ , 则跳过下一条指令	√	-	√	1 or 2
	CMPRS A, I	如果 $A=I$ , 则跳过下一条指令	√	-	√	1 or 2
其它	NOP	空指令不作任何操作	-	-	-	1
	IDLE	芯片进入IDLE模式	-	-	-	1
	STOP	芯片进入睡眠模式	-	-	-	1
	CLR M	SFR或RAM赋值0x00	-	-	-	1
	SET M	SFR或RAM赋值0xFF	-	-	-	1
查表	MOV C M	$[EDATH, M] \leftarrow \text{ROM}[EADRH, EADRL]$ 把ROM地址(EADRH, EADRL)中的值高8位赋给EDATH, 低8位赋值给M	-	-	-	3

参数说明:

M: 系统寄存器或RAM

A: 工作寄存器

I: 立即数

b: 位选择

C: 进位/借位标志

AC: 半进位/借位标志

Z: 零标志

PC: 程序计数器

## 25 电气特性

### 25.1 极限值

表 15 RM1211A 极限值

参数	标号	条件	范围	单位
供电电压	VDD	-	- 0.3 to + 5.5	V
输入电压	VI	所有 I/O 口	- 0.3 to VDD + 0.3	V
输出电压	VO	所有输出口	- 0.3 to VDD + 0.3	V
I/O 口输出电流	IOH	单个 I/O 口工作时	- 25	mA
		所有 I/O 口工作时	- 80	mA
I/O 口输入电流	IOL	单个 I/O 口工作时	+ 20	mA
		所有 I/O 口工作时	+ 80	mA
工作温度	TA	-	- 40 to + 85	°C
储藏温度	TSTG	-	- 40 to + 125	°C

### 25.2 直流特性

(VDD = 1.8~5.5V, TA = 25 °C, 如无其他说明则都是此条件)

表 16 RM1211A 直流特性

参数	标号	条件	最小值	典型值	最大值	单位
工作电压	VDD	振荡器频率 = 0.4 - 2MHz/2CLK	1.8	-	5.5	V
		振荡器频率 = 0.4 - 8MHz/2CLK	3.0	-	5.5	
		振荡器频率 = 0.4 - 16MHz/2CLK	4.0	-	5.5	
输入高电压	VIH1	VDD = 1.8 to 5.5V	0.7VDD	-	VDD	V
	VIH2		0.4VDD			
输入低电压	VIL1	VDD = 1.8 to 5.5V	-	-	0.3VDD	V
	VIL2				0.2VDD	
输出高电压	VOH	IOH = -10mA PA, PB	VDD = 3.0 to 5.5V	0.9VDD		V
输出低电压	VOL	IOL = 20mA PA, PB	VDD = 3.0 to 5.5V	-	0.1VDD	V
	VOL_H	IOL = 40mA PB.3, PA.3	VDD = 3.0 to 5.5V		0.1VDD	V

输入高漏电流	ILIH1	所有输入口	VIN = VDD	-	-	1	uA
输入低漏电流	ILIL1	所有输入口	VIN = 0V	-	-	- 1	uA
输出高漏电流	ILOH	所有输出管脚	VOUT = VDD	-	-	2	uA
输出低漏电流	ILOL	所有输出管脚	VOUT = 0V	-	-	- 2	uA
上拉电阻	RP1	VIN = 0V,	VDD =5V TA = 25°C	25	50	100	kΩ
下拉电阻	RP2	VIN = 0V,	VDD =5V TA = 25°C	25	50	100	
供电电流	IDD1 Run 模式	IRC 16MHz 频率, 2CLK	VDD=5VTA=25°C	-	3.2	6	mA
		晶振 4MHz 频率, 2CLK	VDD=5VTA=25°C	-	1.1	2	mA
		晶振 32768HZ 频率, 2CLK	VDD=5VTA=25°C	-	-	25	uA
		LIRC128KHz 频率, 2CLK	VDD=5VTA=25°C	-	-	50	uA
	IDD2 IDLE 模式	IRC 16MHz 频率, 2CLK	VDD=5VTA=25°C	-	-	400	uA
		晶振 4MHz 频率, 2CLK	VDD=5VTA=25°C	-	-	200	uA
		晶振 32768HZ 频率, 2CLK	VDD=5VTA=25°C	-	-	15	uA
		LIRC128KHz 频率, 2CLK	VDD=5VTA=25°C	-	-	25	uA
	IDD3 睡眠模式		VDD = 3.3V (禁止 LVR) TA = 25°C	-	-	0.5	uA
			VDD = 3.3V (使能 LVR) TA = 25°C	-	-	1.0	
			VDD = 5V (禁止 LVR) TA = 25°C	-	-	1.2	
			VDD = 5V (使能 LVR) TA = 25°C	-	-	1.8	

### 25.3 内部 16 MHz 高速时钟(HIRC)

符号	参数	条件	最小值	典型值	最大值	单位
F <sub>HIRC</sub>	中心频率	-	-	16	-	MHz
	校准后内部时钟频率	TA = 25 °C, VDD = 5V	-1	-	+1	%
		TA =-40~85°C, VDD=2.5V~5.5V	-3	-	+3	%

## 25.4 内部 128 KHz 低速时钟(LIRC)

符号	参数	条件	最小值	典型值	最大值	单位
F <sub>LIRC</sub>	中心频率	-	-	128	-	KHz
	校准后内部时钟频率	TA = 25 °C, VDD = 2.5V~5.5V	-10	-	+10	%
		TA = -40~85°C, VDD=2.5V~5.5V	-40	-	+40	%

## 25.5 低压复位规格

参数	符号	复位值档位	条件	最小值	典型值	最大值
低电压复位 电压值	V <sub>LVR</sub>	1.9V			1.9V	
		2.3V			2.3V	
		2.7V			2.7V	
		3.0V			3.0V	
		3.3V			3.3V	
		3.5V			3.5V	
		3.7V			3.7V	
		4.1V			4.1V	

## 25.6 欠压检测规格

参数	符号	复位值档位	条件	最小值	典型值	最大值
低电压检测 电压值	V <sub>LVD</sub>	1.9V			1.9V	
		2.0V			2.0V	
		2.1V			2.1V	
		2.2V			2.2V	
		2.3V			2.3V	
		2.4V			2.4V	
		2.5V			2.5V	
		2.7V			2.7V	
		3.0V			3.0V	
		3.3V			3.3V	
		3.5V			3.5V	
		3.7V			3.7V	
		3.9V			3.9V	
		4.1V			4.1V	

参数	符号	复位值档位	条件	最小值	典型值	最大值
		4.3V			4.3V	
		4.5V			4.5V	

## 25.7 ADC 电气特性

( $T_A=25^{\circ}\text{C}$ , 除非另有说明)

参数	符号	条件	最小值	典型值	最大值	单位
供电电压	$V_{DD}$					V
精度	$N_R$	$GND \leq V_{AIN} \leq V_{DD}$				bit
ADC 输入电压	$V_{AIN}$		GND		VDD	V
ADC 输入电阻	$R_{AIN}$	$V_{AIN} = 5V$				$\Omega$
ADC 转换电流 1	$I_{ADC1}$	ADC 模块打开 $V_{DD} = 5V$ , 电流 X1				mA
ADC 转换电流 2	$I_{ADC2}$	ADC 模块打开 $V_{DD} = 5V$ , 电流 X2				mA
微分非线性	DNL	$V_{DD} = 5V$ $V_{REF} = 5V$ ADC CLOCK = 2M ADC 采样周期=4				LSB
积分非线性	INL					LSB
偏移量误差	$E_Z$					LSB
满刻度误差	$E_F$					LSB
总绝对误差	$E_{AD}$					LSB
ADC 转换时间 1	$T_{ADC1}$	ADC CLOCK = 2M ADC 采样周期=4				$\mu\text{S}$
ADC 转换时间 2	$T_{ADC2}$	ADC CLOCK = 1M ADC 采样周期=4				$\mu\text{S}$

## 25.8 OPA 电气特性

( $T_A=25^{\circ}\text{C}$ , 除非另有说明)

参数	符号	条件	最小值	典型值	最大值	单位
供电电压	$V_{DD}$		3.0		5.5	V
工作电流	$I_{OPA}$	$V_{DD} = 5V$ , 无负载 $A0P = A0N = 1V$				$\mu\text{A}$
共模输入范围	$V_{IN}$	$V_{DD} = 5V$	GND		VDD	V
输出电压范围	$V_{OUT}$	$V_{DD} = 5V$ $I_{LOAD} = 2\text{mA}$				V
失调电压	$V_{OFFSET}$	校准后				mv

参数	符号	条件	最小值	典型值	最大值	单位
增益带宽	GBW	$C_{LOAD}=100pF$				MHz
转换速率	SR	$VDD=5V$ $C_{LOAD}=100pF$				V/uS
开环增益	AOL		60	90		dB
电源抑制比	PSRR		60	80		dB
共模抑制比	CMRR		60	90		dB

## 25.9 交流电气特性

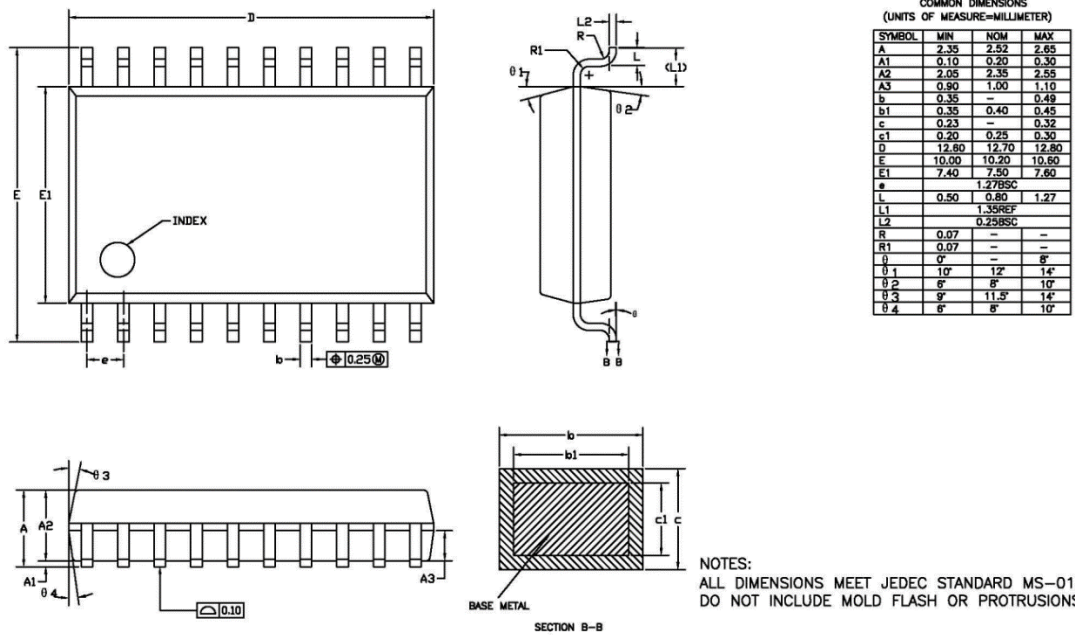
( $T_A=25^{\circ}C$ , 除非另有说明)

参数	符号	条件	最小值	典型值	最大值	单位
外置高频振荡器起振时间	$T_{OSC1}$	外置 16MHz 晶体				ms
外置高频振荡器起振时间	$T_{OSC1}$	外置 32KHz 晶体				ms
POWER ON RESET 时间	$T_{POR}$					ms
PD 模式唤醒时间	$T_{PDW}$					ms
复位脉冲宽度	$T_{RESET}$					us
内置 RC 振荡起振时间	$f_{HRC}$	$VDD=3.3V\sim 5.5V$ $T_A=-40\sim 85^{\circ}C$				uS
MTP 编程时间	$T_{MTP}$	$VDD=3V$				
		$VDD=5V$				
EEPROM 编程时间	$T_{EEPROM}$	$VDD=3V$				ms
		$VDD=5V$				ms

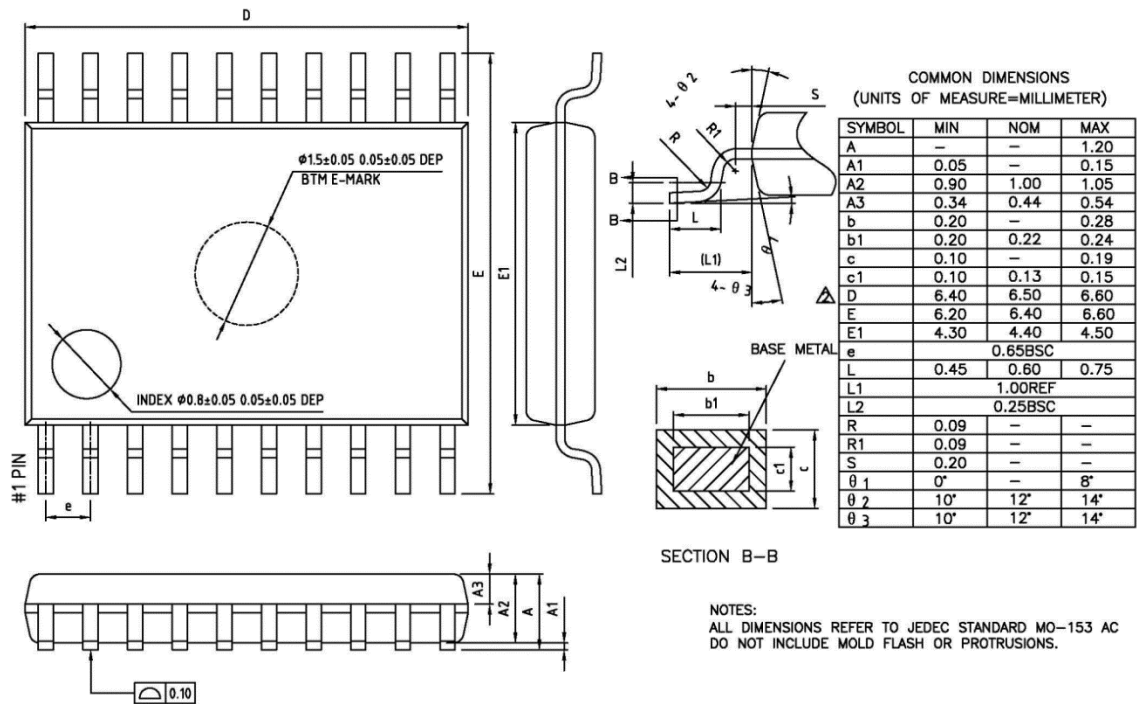


## 26 封装

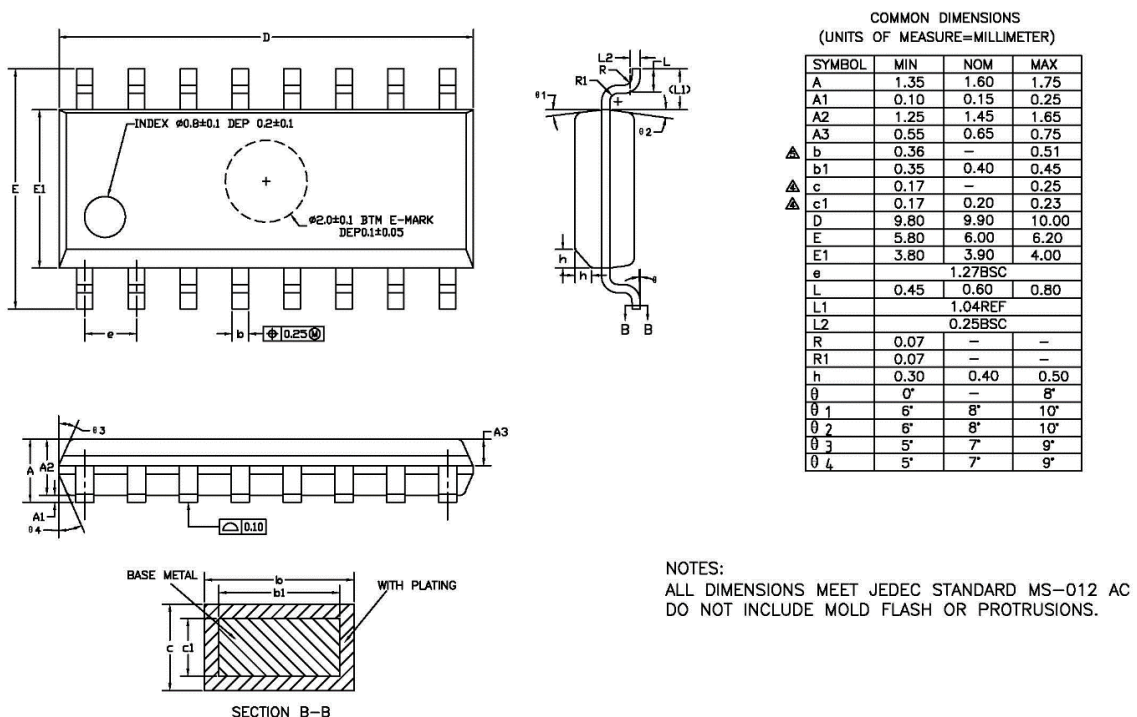
### 26.1 SOP20 封装尺寸与外形图



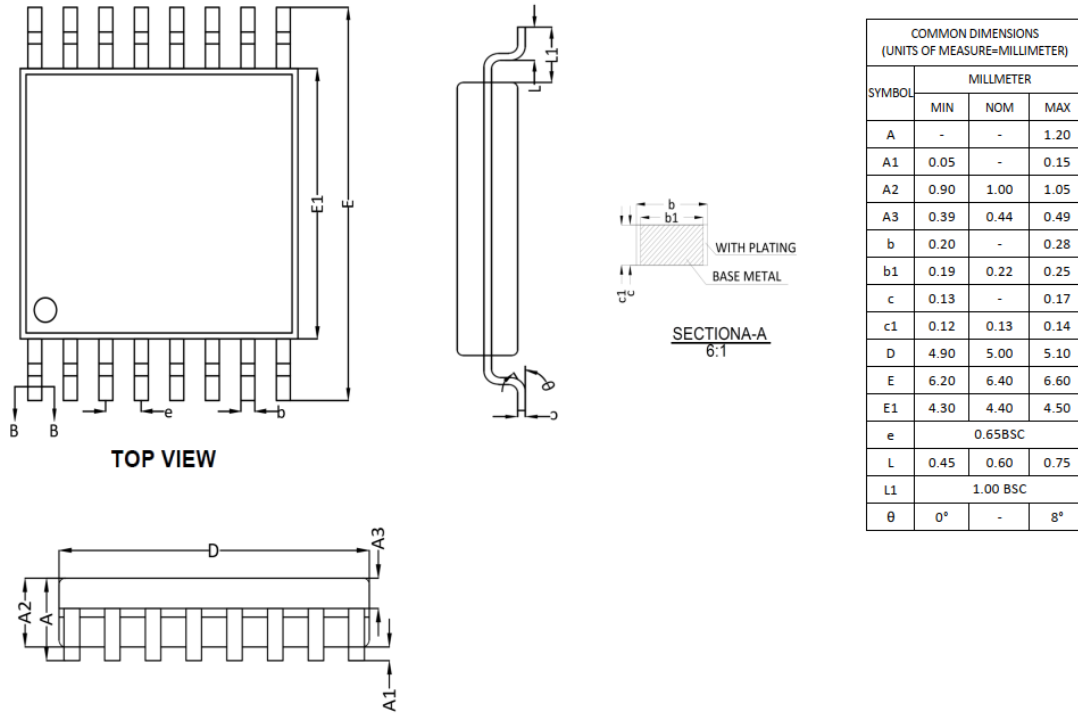
## 26.2 TSSOP20 封装尺寸与外形图



## 26.3 SOP16 封装尺寸与外形图

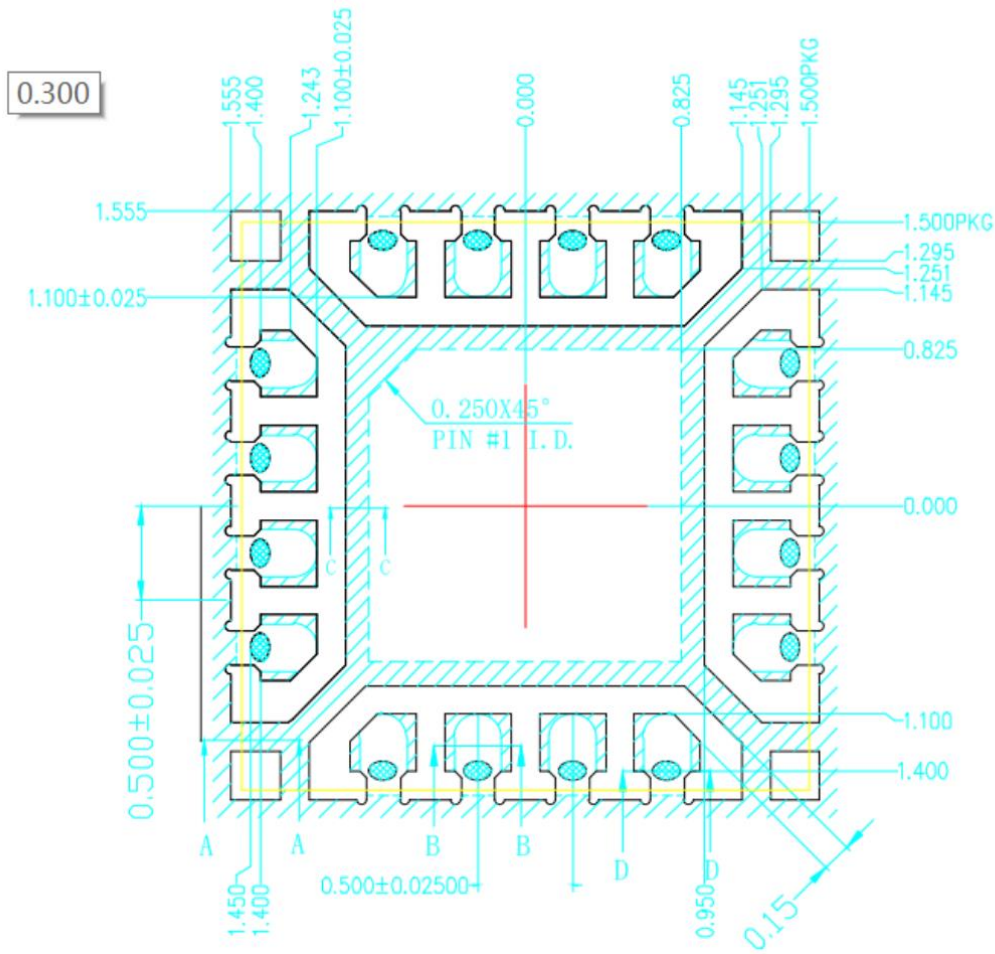


### 26.4 TSSOP16 封装尺寸图



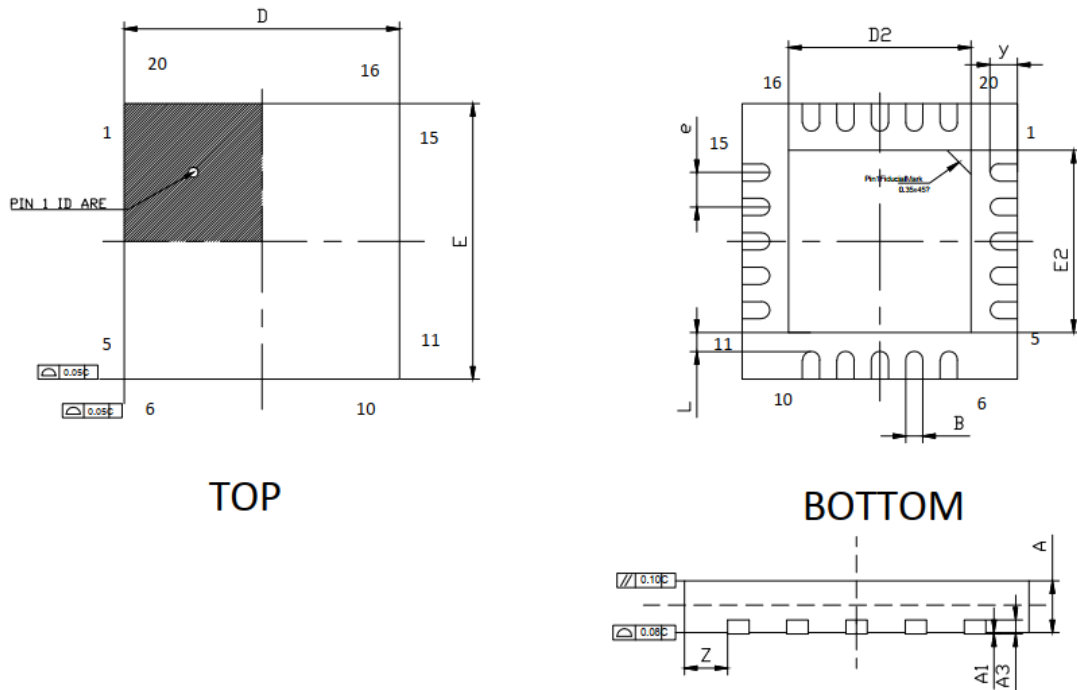
RealMagi

26.5 QFN16 3mm\*3mm 封装尺寸图



Real

## 26.6 QFN20 3mm\*3mm 封装尺寸图



### Dimensions

Unit	D	E	D2	E2	A	A1	A3	B	e	K	L	y	Z
mm	3.025 (3.00) 2.975	3.025 (3.00) 2.975	1.65 (1.6) 1.55	1.65 (1.6) 1.55	0.80 (0.75) 0.70	0.05 (0.02) 0.00	0.203 REF	0.30 (0.25) 0.20	0.40 BSC	-	0.33 (0.28) 0.23	0.40 REF	0.655 REF

## 26.7 订购信息

产品型号	封装形式
RM1211ASO20	SOP20
RM1211ATS20	TSSOP20
RM1211ASO16	SOP16
RM1211ATS16	TSSOP16
RM1211AQN16	QFN16_3*3
RM1211AQN20	QFN20_3*3