

E85F3365

1T 8051 内核微控制器

Part No.	Core	FLASH	SRAM	EE	I/O	Timer	PWM	ADC	TK	SPI/IIC	UART	LED	Package
E85F3365LJ	1T51	16 K	1 K	128	30	4 x 16	3 x 2Ch	12ch	24ch	1	2	8COM x 12SEG	LPGF32
E85F3365NJ					26			10ch	21ch				QFN32
E85F3365PH					26			10ch	21ch				SOP28
E85F3365TH					26			10ch	21ch				TSSOP28
E85F3365SG					22			8ch	21ch			4COM x 12SEG	SSOP24

注：请访问 www.esseic.com 关注规格更新

➤ 内核

- 1T51 内核，兼容标准 8051 指令集
- 内核最高工作频率 14MHz

➤ 存储

- 16K 字节 FLASH 程序存储器
- 128 字节 EEPROM 数据存储器
- 256 字节 IRAM，768 字节 XRAM

➤ 复位与启动

- 内置上电复位 POR
- 内置掉电复位 BOR，支持 8 档掉电复位 1.6V~4.4V，步进 0.4V

➤ 时钟

- 外部 32K-20MHz 晶体振荡器 XOSC
- 内部 32MHz 或 28MHz 或 16MHz 振荡器 HRC（出厂校准精度 $\leq \pm 1\%$ ，全温工作精度 $\leq \pm 2\%$ ）
- 内部 16KHz 低功耗振荡器 LRC，支持 HRC 辅助修正，修正精度 $\leq \pm 3\%$

➤ 调试和编程

- 单线调试，单线编程

➤ 工作条件

- VDD=4.5V~5.5V@14M
- VDD=2.7V~5.5V@8M
- VDD=1.8V~5.5V@2M
- 工作温度范围 -40~85°C

➤ 低功耗

- 待机睡眠功耗低至 5uA
- 8MHz@3V 运行功耗典型 5mA

➤ 端口

- 最多支持 30 个 I/O 端口，8 个 COM 复用端口灌电流 80mA，12 个 SEG 复用口支持 2/4/8/15mA 拉电流配置
- 所有端口支持独立弱上拉和弱下拉控制，同时开启可实现 1/2VDD 用于 LCD 驱动
- 所有端口支持外部中断唤醒

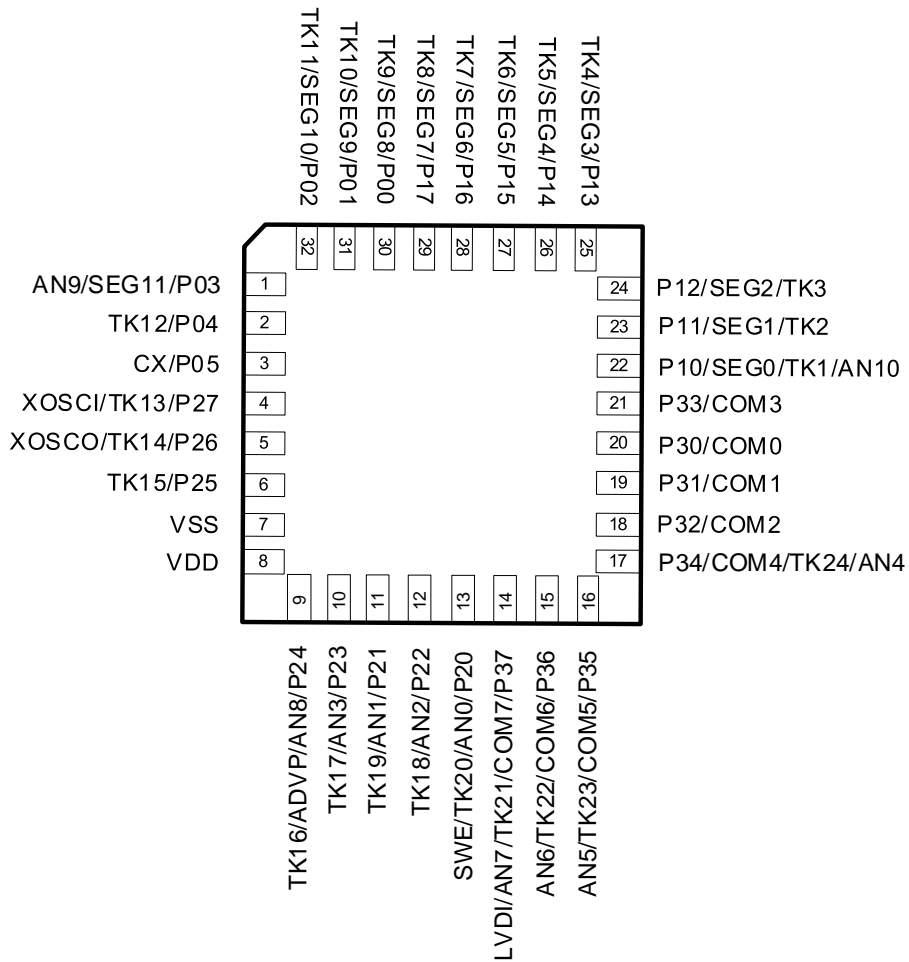
➤ 外设

- 4 路 16 位定时/计数器 TMR，支持比较捕捉功能
- 3 组独立 16+3 位 PWM，每组支持 2 路互补或同相输出
- 24 通道 TK 触摸按键扫描通道
- 12 通道 12 位 SAR ADC，支持 PWM 延时触发
- 内置多档参考电压，校准精度 $\leq \pm 1.5\%$
- 8COM x 12SEG LED 驱动
- LVD 低电压检测，支持对 VDD 8 档低电压检测，步进电压为 0.4V；支持外部管脚输入 LVD 检测，比较电压 0.5V
- 1 路 SPI，1 路 IIC，2 路 UART

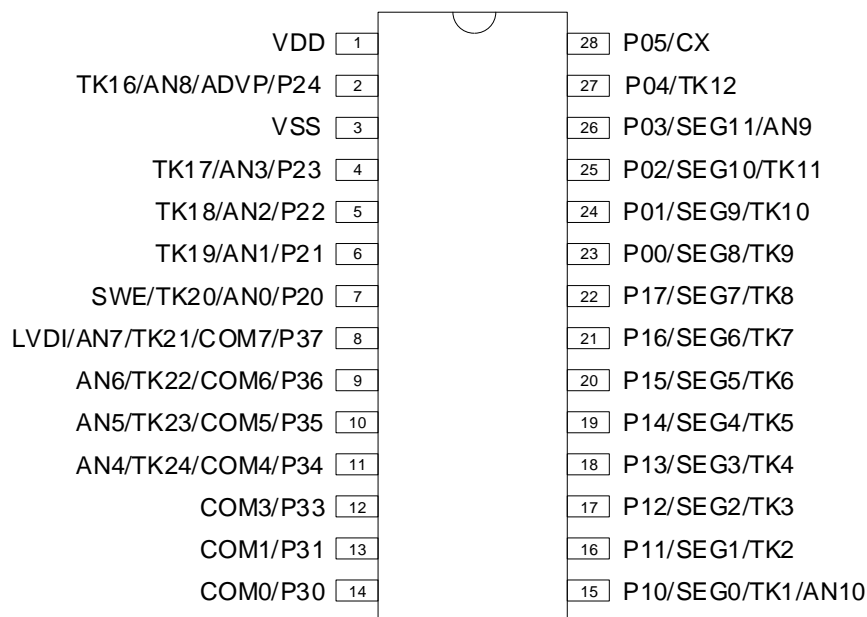
➤ 封装类型

- LQFP32/QFN32/ TSSOP28/SSOP28

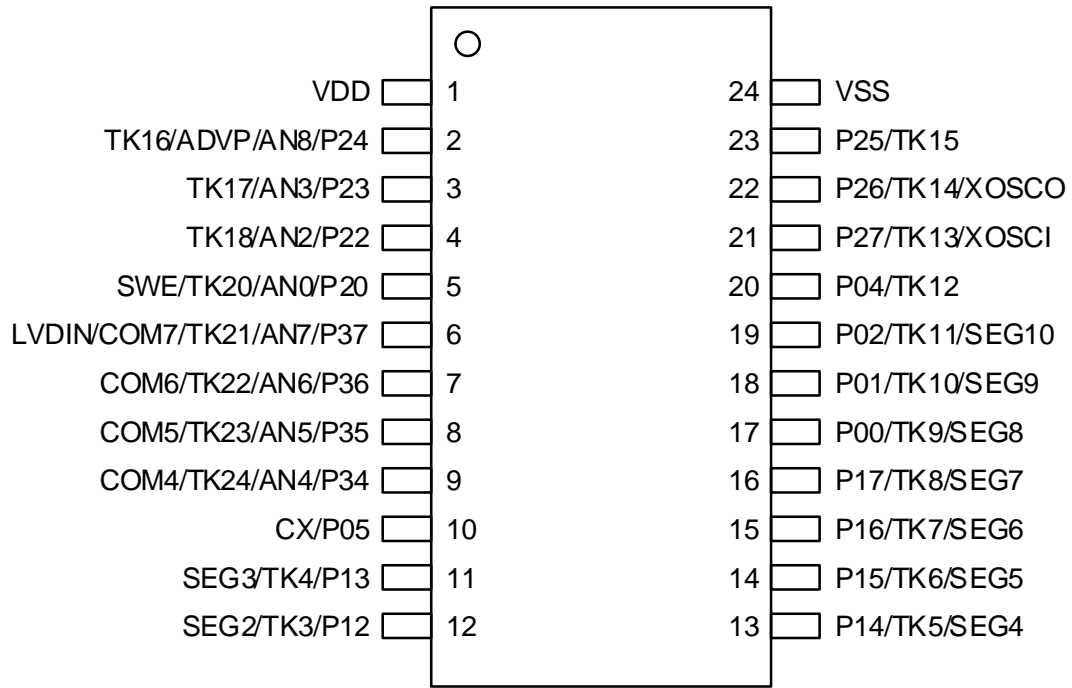
1 管脚配置



图表 1 32 脚封装顶视图



图表 2 28 脚封装顶视图



图表 3 24 脚封装顶视图

1.1 管脚复用表(FUN12~FUN14 未用)

Pin	FUN0	FUN1	FUN2	FUN3	FUN4	FUN5	FUN6	FUN7	FUN8	FUN9	FUN10	FUN11	FUN15	EINT	TK	ADC	其他
P00	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN0/TO0	CAP0	SS	SEG9	EINT00	TK9	—	—
P01	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN1/TO1	CAP1	SCK	SEG10	EINT01	TK10	—	—
P02	IO	PWM00	PWM10	PWM20	TX1	RX1	SCL	SDA	T1G	TIN2/TO2	CAP2	MISO	SEG11	EINT02	TK11	—	—
P03	IO	PWM01	PWM11	PWM21	TX1	RX1	SCL	SDA	T0G	TIN3/TO3	CAP0	MOSI	SEG12	EINT03	—	AN9	—
P04	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN0/TO0	CAP1A	—	—	EINT04	TK12	—	—
P05	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN1/TO1	CAP2	—	—	EINT05	CX	—	—
P10	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN2/TO2	CAP2	SS	SEG1	EINT10	TK1	AN10	—
P11	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN3/TO3	CAP0	SCK	SEG2	EINT11	TK2	—	—
P12	IO	PWM00	PWM10	PWM20	TX1	RX1	SCL	SDA	T1G	TIN0/TO0	CAP1	MISO	SEG3	EINT12	TK3	—	—
P13	IO	PWM01	PWM11	PWM21	TX1	RX1	SCL	SDA	T0G	TIN1/TO1	CAP2	MOSI	SEG4	EINT13	TK4	—	—
P14	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN2/TO2	CAP0	—	SEG5	EINT14	TK5	—	—
P15	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN3/TO3	CAP1	—	SEG6	EINT15	TK6	—	—
P16	IO	PWM00	PWM10	PWM20	TX1	RX1	SCL	SDA	T1G	TIN3/TO3	CAP2	—	SEG7	EINT16	TK7	—	—
P17	IO	PWM01	PWM11	PWM21	TX1	RX1	SCL	SDA	T0G	TIN2/TO2	CAP0	—	SEG8	EINT17	TK8	—	—
P20	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN0/TO0	CAP1	SS	—	EINT00	TK20	AN0	SWE
P21	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN1/TO1	CAP2	SCK	—	EINT01	TK19	AN1	—
P22	IO	PWM00	PWM10	PWM20	TX1	RX1	SCL	SDA	T1G	TIN2/TO2	CAP0	MISO	—	EINT02	TK18	AN2	—
P23	IO	PWM01	PWM11	PWM21	TX1	RX1	SCL	SDA	T0G	TIN3/TO3	CAP1	MOSI	—	EINT03	TK17	AN3	—
P24	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN0/TO0	CAP2	—	—	EINT04	TK16	AN8/ ADVP	—
P25	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN1/TO1	CAP0	—	—	EINT05	TK15	—	—

P26	IO	PWM00	PWM10	PWM20	TX1	RX1	SCL	SDA	T1G	TIN2/TO2	CAP1	—	—	EINT06	TK14	—	XOSCO
P27	IO	PWM01	PWM11	PWM21	TX1	RX1	SCL	SDA	T0G	TIN3/TO3	CAP2A	—	—	EINT07	TK13	—	XOSCI
P30	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN0/TO0	CAP0	SS	COM1	EINT10	—	—	—
P31	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN1/TO1	CAP1	SCK	COM2	EINT11	—	—	—
P32	IO	PWM00	PWM10	PWM20	TX1	RX1	SCL	SDA	T1G	TIN2/TO2	CAP2	MISO	COM3	EINT12	—	—	—
P33	IO	PWM01	PWM11	PWM21	TX1	RX1	SCL	SDA	T0G	TIN3/TO3	CAP0	MOSI	COM4	EINT13	—	—	—
P34	IO	PWM00	PWM10	PWM20	TX0	RX0	SCL	SDA	T3G	TIN0/TO0	CAP1	—	COM5	EINT14	TK24	AN4	—
P35	IO	PWM01	PWM11	PWM21	TX0	RX0	SCL	SDA	T2G	TIN1/TO1	CAP2	—	COM6	EINT15	TK23	AN5	—
P36	IO	PWM00	PWM10	PWM20	TX1	RX1	SCL	SDA	T1G	TIN0/TO0	CAP0	—	COM7	EINT16	TK22	AN6	—
P37	IO	PWM01	PWM11	PWM21	TX1	RX1	SCL	SDA	T0G	TIN1/TO1	CAP1	—	COM8	EINT17	TK21	AN7	LVDI

图表 4 端口逻辑复用表

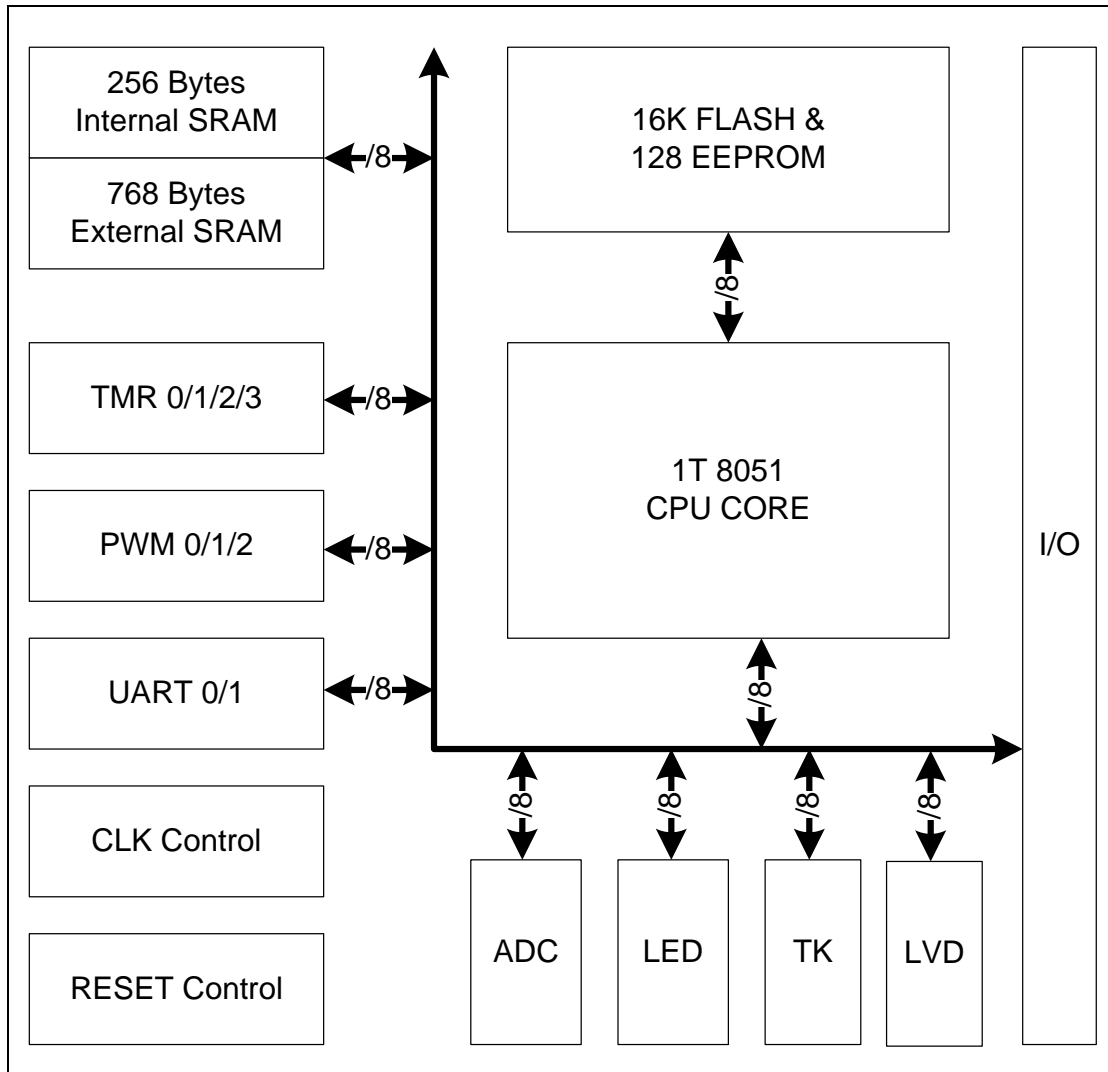
注：COM 和 SEG 的输出电流可调整，详情请查看 I/O 端口描述，P3_IOL_CL，P3_IOL_CH 设置 COM 灌电流，P0_IOH_CL，P1_IOH_CL，P0_IOH_CH 设置 SEG 的拉电流。

1.2 管脚复用功能说明

符号	类型	描述	备注
Pn0~Pn7	IO	8 位双向 IO 端口	支持上下拉电阻，支持中断唤醒
PWMn0、PWMn1	O	PWM 输出端口	支持同相或互补输出
TON(TO0~TO3)	O	TIMER 比较输出	比较输出
TINn(TIN0~TIN3)	I	TIMER 外部时钟输入	
CAPn(CAP0~CAP2)	I	捕捉模式下为捕捉输入 PWM 模式下为 PWM 暂停输入	
CAPnA	I	捕捉模式下为捕捉输入 PWM 模式下为 PWM STOP 输入	P04 和 P27 两管脚支持 PWM 全暂停
TXn	O	UART 发送端口	
RXn	I	UART 接收端口	
SCL	I	IIC 时钟线	
SDA	IO	IIC 数据线	
SS	IO	SPI 片选信号	
SCK	IO	SPI 时钟信号	
MOSI	IO	SPI 主输出从输入	
MISO	IO	SPI 主输入从输出	
COM1~8	O	LED COM 驱动端口	
SEG1~12	O	LED SEG 驱动端口	
EINT0n	I	外部中断输入端口	支持睡眠唤醒
EINT1n	I	外部中断输入端口	支持睡眠唤醒
SWE	IO	单线仿真烧录端口	
RSTN	I	外部复位输入端口	输入低电平复位
AN0~AN10	A	ADC 模拟检测端口	
ADVP	A	ADC 外部正端参考电压输入	
LVDI	A	LVD 模拟检测端口	检测 0.5V 电压
XOSCI	A	外部振荡器输入	
XOSCO	A	外部振荡器输出	

图表 5 端口描述表

2 功能框图



图表 6 功能框图

3 电气特性

3.1 极限参数

存储温度 T_{STG}	-55°C ~ 125°C
供电极限电压 $V_{DD}-V_{SS}$	-0.3V ~ 6.5V
输入极限电压 V_{IN}	$V_{SS}-0.3V \sim V_{DD}+0.3V$
VDD 最大承载电流 I_{VDD}	100mA
VSS 最大承载电流 I_{VSS}	100mA

3.2 工作条件

符号	描述	最小值	最大值	单位
F_{SYS_CLK}	4.5~5.5V 系统工作频率	—	14M	Hz
	2.7~5.5V 系统工作频率	—	8M	Hz
	1.8~5.5V 系统工作频率	—	2M	Hz
VDD	工作电压	1.8	5.5	V
T_A	工作温度	-40	85	°C
t_{VDD}	VDD 上电斜率			us/V

3.3 DC 特性

典型值测试基本条件： $T_A=25^\circ\text{C}$ ， $V_{DD}=3\text{V}$
电流测试时 I/O 输出无负载，I/O 输入不浮空

* 以下参数均为设计值

符号	描述	最小值	典型值	最大值	单位	条件
VDD	供电电压	1.8	—	5.5	V	—
I_{DDH}	高速工作电流	—	10	—	mA	$F_{SYS_CLK}=14\text{MHz @}5\text{V}$
I_{DDM}	中频工作电流	—	5	—	mA	$F_{SYS_CLK}=8\text{MHz @}5\text{V}$
I_{DDS}	待机电流	—	1	10	uA	进入 PD 模式，所有功能关闭
V_{IL}	输入低电压	0	—	1.0	V	I/O 均为 SCHMITT 输入特性
V_{IH}	输入高电压	1.8	—	VDD	V	I/O 均为 SCHMITT 输入特性
I_{LK}	输入漏电流	—	—	± 1	uA	内部上/下拉电阻关闭
V_{OL}	输出低电压	$V_{SS}+0.7$	—	—	V	$I_{OL}=15\text{mA}$
V_{OL}^*	大电流口输出低电压	—	—	$V_{SS}+1.0$	V	$I_{OL}=80\text{mA}$
V_{OH}	输出高电压	—	—	$V_{DD}-0.7$	V	$I_{OH}=15\text{mA}$
R_{PU}	内部上拉电阻	—	30K	—	Ω	—
R_{PD}	内部下拉电阻	—	30K	—	Ω	—

3.4 存储器特性

存储	操作	最小值	最大值	单位	条件
FLASH	编程次数	1000	—	—	—
	数据保持时间	10	—	year	85°C
	编程时间	—	0.7	ms	4.5~5.5V
EEPROM	编程次数	8000	—	—	0-70°C
		5000	—	—	-20-85°C
	数据保持时间	10	—	year	85°C
	编程时间	—	1.5	ms	2.5~5.5V
SRAM	最低数据保持电压	0.6	—	V	—

目录

1	产品特性	- 1 -
2	管脚配置	- 2 -
2.1	管脚复用表(FUN12~FUN14 未用)	- 4 -
2.2	管脚复用功能说明	- 6 -
3	功能框图	- 7 -
4	电气特性	- 8 -
4.1	极限参数	- 8 -
4.2	工作条件	- 8 -
4.3	DC 特性.....	- 8 -
4.4	存储器特性	- 9 -
5	内核	- 15 -
5.1	描述	- 15 -
5.2	内核寄存器	- 15 -
6	存储	- 17 -
6.1	程序寻址空间映射	- 17 -
6.2	FLASH+EEPROM 存储器.....	- 17 -
6.3	用户配置信息	- 18 -
6.4	IAP 操作.....	- 21 -
6.5	数据寻址空间	- 24 -
7	复位	- 32 -
7.1	描述	- 32 -
7.2	复位寄存器	- 33 -
8	时钟	- 34 -
8.1	描述	- 34 -
8.2	最大工作频率说明	- 35 -
8.3	时钟寄存器	- 35 -
9	低功耗	- 39 -
9.1	描述	- 39 -
9.2	低功耗寄存器	- 39 -
10	看门狗定时器 WDT	- 40 -
10.1	描述	- 40 -
10.2	WDT 寄存器	- 40 -
11	中断	- 41 -

11.1 中断向量	- 41 -
11.2 中断优先级	- 41 -
11.3 中断寄存器	- 42 -
11.1 外部端口中断 EINT	- 43 -
11.2 定时器 TMR 中断	- 45 -
11.3 边沿捕捉器 CAP 中断	- 45 -
11.4 脉宽调制器 PWM 中断	- 46 -
11.5 异步收发器 UART 中断	- 47 -
11.6 IIC 总线控制器中断	- 48 -
11.1 SPI 总线控制器中断	- 49 -
11.2 模拟模块中断	- 50 -
12 I/O 端口	- 52 -
12.1 I/O 功能框图	- 52 -
12.2 I/O 寄存器	- 52 -
12.1 I/O 功能复用功能寄存器	- 57 -
13 定时/计数器 TMR	- 59 -
13.1 描述	- 59 -
13.2 TMR 寄存器	- 61 -
14 边沿捕捉器 CAP	- 63 -
14.1 CAP 寄存器	- 65 -
15 脉宽调制器 PWM	- 67 -
15.1 描述	- 67 -
15.2 PWM 寄存器	- 69 -
16 SPI 通讯端口	- 72 -
16.1 描述	- 72 -
16.2 SPI 寄存器	- 74 -
17 IIC 总线控制器	- 76 -
17.1 描述	- 76 -
17.2 IIC 寄存器	- 79 -
18 UART 异步收发器	- 83 -
18.1 描述	- 83 -
18.2 UART 操作流程图	- 84 -
18.3 UART 寄存器	- 86 -
19 模数转换器 ADC	- 89 -
19.1 描述	- 89 -

19.2 ADC 操作流程	- 90 -
19.3 ADC 寄存器	- 91 -
20 触控按键控制器 TK	- 94 -
20.1 描述	- 94 -
20.2 TK 功能框图	- 94 -
20.3 TK 操作流程	- 96 -
20.4 TK 寄存器	- 96 -
21 内部参考电压 VREF	- 100 -
21.1 描述	- 100 -
21.2 VREF 寄存器	- 100 -
22 LED 驱动	- 101 -
22.1 描述	- 101 -
22.2 LED 寄存器	- 102 -
22.1 SRAM 映射	- 105 -
23 低电压检测 LVD	- 106 -
23.1 描述	- 106 -
23.2 LVD 寄存器	- 106 -
24 指令集	- 107 -
24.1 算术运算指令	- 107 -
24.2 逻辑操作指令	- 108 -
24.3 数据传送指令	- 109 -
24.4 位操作指令	- 110 -
24.5 程序转移指令	- 110 -
25 封装信息	- 111 -

图表目录

图表 1 32 脚封装顶视图	- 2 -
图表 2 28 脚封装顶视图	- 2 -
图表 3 24 脚封装顶视图	- 3 -
图表 4 端口逻辑复用表	- 5 -
图表 5 端口描述表	- 6 -
图表 6 功能框图	- 7 -
图表 7 程序存储空间映射图	- 17 -
图表 8 INFO FLASH 映射图	- 18 -
图表 9 IAP 操作流程图中	- 21 -
图表 10 数据寻址空间映射图	- 24 -
图表 11 系统时钟源功能框图	- 34 -
图表 12 外部振荡器 XOSC 连接示意图	- 34 -
图表 12 中断向量表	- 41 -
图表 14 I/O 功能框图	- 52 -
图表 15 TMR 功能框图	- 59 -
图表 16 TMR 操作流程图中	- 60 -
图表 16 捕捉清零模式示例波形图	- 63 -
图表 17 捕捉累加模式示例波形图	- 63 -
图表 18 CAP 操作流程图中	- 64 -
图表 19 PWM 边沿对齐工作示例波形图	- 67 -
图表 20 PWM 中心对齐工作示例波形图	- 67 -
图表 21 PWM 操作流程图中	- 68 -
图表 22 SPI 主控模式流程图	- 72 -

图表 23 SPI 从动模式流程图.....	- 73 -
图表 24 IIC 通讯中断机制图	- 76 -
图表 25 IIC 通信等待波形示意图.....	- 76 -
图表 26 IIC 主控通讯流程图	- 77 -
图表 27 IIC 从机通讯流程图	- 78 -
图表 28 UART 通讯中断机制 1	- 83 -
图表 29 UART 通讯中断机制 2	- 83 -
图表 30 UART 发送操作流程图	- 84 -
图表 31 UART 接收操作流程图	- 85 -
图表 32 ADC 转换时序图	- 89 -
图表 33 ADC 操作流程图	- 90 -
图表 34 TK 功能框图	- 94 -
图表 35 TK 运行时序图.....	- 95 -
图表 36 TK 运行流程图.....	- 96 -
图表 37 LED 驱动波形示意图	- 101 -

4 内核

4.1 描述

芯片采用 1T51 架构 8 位 CPU 内核，兼容标准 8051 指令集。

4.2 内核寄存器

ACC 累加器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
A<7:0>							

Bit7-0 **A<7:0>** : 累加器

B B 寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
B<7:0>							

Bit7-0 **B<7:0>** : B 寄存器

SP 堆栈指针

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
SP<7:0>							

Bit7-0 **SP<7:0>** : 堆栈指针

DPL 数据指针低 8 位寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
DPTR<7:0>							

Bit7-0 **DPTR<7:0>** : 数据指针低 8 位

DPH 数据指针高 8 位寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
DPTR<15:8>							

Bit7-0 **DPTR<15:8>** : 数据指针高 8 位

PSW 状态寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-0	RW-00		RW-0	—	RW-0
CY	AC	F0	RS<1:0>		OV	—	P

Bit7 **CY** : 进位标志位

1 : 算数或逻辑运算无进/借位

0 : 算数或逻辑运算有进/借位

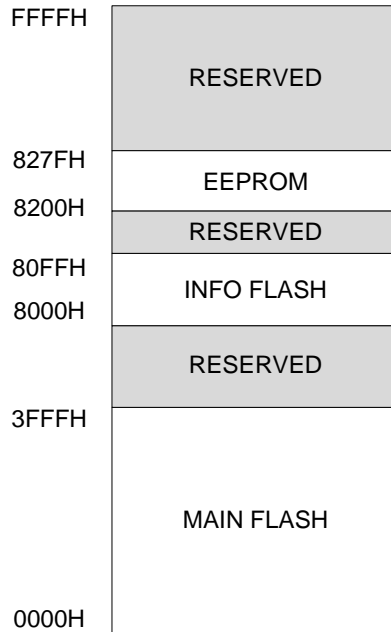
Bit6 **AC** : 辅助进位标志位 (用于 BCD 操作)

	1 : 算数或逻辑运算无辅助进/借位
	0 : 算数或逻辑运算有辅助进/借位
Bit5	F0 : 用户自定义标志位 0
Bit4-3	RS <1:0> : 工作寄存器 R0-R7 bank 选择位
	11 : bank3 (18H~1FH)
	10 : bank2 (10H~17H)
	01 : bank1 (08H~0FH)
	00 : bank0 (00H~07H)
Bit2	OV : 溢出标志位
	1 : 有符号数运算无溢出
	0 : 有符号数运算有溢出
Bit1	保留
Bit0	P : 奇偶标志位
	1 : 累加器中“1”的个数为奇数
	0 : 累加器中“1”的个数为偶数

5 存储

芯片存储采用 Harvard 架构，即程序寻址空间与数据寻址空间独立。

5.1 程序寻址空间映射



图表 7 程序存储空间映射图

5.2 FLASH+EEPROM 存储器

数据区 EEPROM

芯片内置 128 字节 EEPROM 存储器用于存储数据。

EEPROM 支持以下操作：

- MOV C 指令读取
- 应用中自编程 IAP 操作
- 烧录器编程 ISP 操作

信息区 INFO FLASH

芯片内置 INFO FLASH 存储器用于存储用户 ID 和用户配置选项。

INFO FLASH 支持以下操作：

- MOV C 指令读取
- 烧录器编程 ISP 操作

程序区 MAIN FLASH

芯片内置 16K 字节 MAIN FLASH 存储器用于存储程序代码。

MAIN FLASH 支持以下操作：

- 程序取指
- MOVC 指令读取（操作权限受 **CFG_WD3.IAP_PRn** 限制）
- 应用中自编程 IAP 操作（操作权限受 **CFG_WD3.IAP_PRn** 限制）
- 烧录器编程 ISP 操作（支持硬件代码加密）

5.3 用户配置信息

802FH	序列号(ID)
802CH	0X802F~0X802C
	reserved
801DH	用户识别码(USER_ID)
801AH	0X801D~0X801A
8019H	CHECKSUM
8016H	0x8016~0X8019
8015H	0XED
	reserved
8011H	CFG_WORD4
	reserved
800DH	CFG_WORD3
	reserved
8009H	CFG_WORD2
	reserved
8005H	CFG_WORD1
	reserved
8001H	CFG_WORD0
	0XAC

图表 8 INFO FLASH 映射图

用户 ID 和用户配置选项存储于 INFO FLASH。INFO FLASH 通过用户在烧录器界面配置烧录。

用户 ID

序列号 ID 共 4 字节，存储于程序程序存储空间 802FH~802CH。用户识别码 USER_ID 存储于程序存储空间 801DH~801AH,具体的信息由用户在量产烧录器中自行定义。

用户配置选项 CFG_WDn

用户配置选项共 4 字节 **CFG_WD0~CFG_WD3**。

CFG_WORD0 配置字 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OSCS<1:0>		PWRC<1:0>		WDT_EN	BOR_EN	BORS<1:0>	

- Bit7-6 **OSCS<1:0>** : 系统时钟选择位
 10 : 固定选择 HRC 11 : 上电默认选择 HRC , 软件可配置
 00 : 保留 01 : 固定选择 XOSC
- Bit5-4 **PWRC<1:0>** : 上电延时选择位
 10 : 约 64ms 11 : 约 128ms
 00 : 无上电延时 01 : 约 16ms
- Bit3 **WDT_EN** : WDT 使能位
 1 : 使能
 0 : 关闭
- Bit2 **BOR_EN** : BOR 使能位
 1 : 使能
 0 : 关闭
- Bit1-0 **BORS<1:0>** : BOR 复位电压选择位
 11 : 4.2V 10 : 3.8V
 01 : 2.5V 00 : 1.6V (选择该档位 BOR 可通过 SOFT_BOR 软件配置)

CFG_WORD1 配置字 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
XOSC_MD<7:0>							

- Bit7-0 **XOSC_MD<7:0>** : XOSC 工作模式选择位
 HS 模式 : 1011_1111 (外接 16MHz 晶振推荐值)
 LP 模式 : 0010_0000 (外接 32K 晶振推荐值)

CFG_WORD2 配置字 2

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DBG_N	—	—	—	POR_FLTS<1:0>		BOR_FLTS<1:0>	

- Bit7 **DBG_N** : 调试模式使能位
 1 : 正常工作模式
 0 : 使能调试模式
- Bit6-4 保留
- Bit3-2 **POR_FLTS<1:0>** : POR 滤波时间选择位
 11 : 约 200us 10 : 约 150us
 01 : 约 100us 00 : 无滤波
- Bit1-0 **BOR_FLTS<1:0>** : BOR 滤波时间选择位
 11 : 约 200us 10 : 约 150us
 01 : 约 100us 00 : 无滤波

CFG_WORD3 配置字 3

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
IAP_PR3<1:0>		IAP_PR2<1:0>		IAP_PR1<1:0>		IAP_PR0<1:0>	

- Bit7-6 **IAP_PR3<1:0>** : MAIN FLASH 地址 3000H~3FFFH 操作权限配置位

- 11 : IAP 擦写允许, MOVC 指令读取允许
 10 : IAP 擦写禁止, MOVC 指令读取允许
 0x : IAP 擦写禁止, MOVC 指令读取禁止
- Bit5-4 **IAP_PR2<1:0>** : MAIN FLASH 地址 2000H~2FFFH 操作权限配置位
 11 : IAP 擦写允许, MOVC 指令读取允许
 10 : IAP 擦写禁止, MOVC 指令读取允许
 0x : IAP 擦写禁止, MOVC 指令读取禁止
- Bit3-2 **IAP_PR1<1:0>** : MAIN FLASH 地址 1000H~1FFFH 操作权限配置位
 11 : IAP 擦写允许, MOVC 指令读取允许
 10 : IAP 擦写禁止, MOVC 指令读取允许
 0x : IAP 擦写禁止, MOVC 指令读取禁止
- Bit1-0 **IAP_PR0<1:0>** : MAIN FLASH 地址 0000H~0FFFH 操作权限配置位
 11 : IAP 擦写允许, MOVC 指令读取允许
 10 : IAP 擦写禁止, MOVC 指令读取允许
 0x : IAP 擦写禁止, MOVC 指令读取禁止

CFG_WORD4 配置字 4

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	CLKFLT<2:0>			—	—	—	—

- Bit7 配置为 1
- Bit6-4 **CLKFLT<2:0>** : 主时钟滤波
 000 : 推荐使用 8ns 滤波
 其他 : 保留
- Bit3-0 配置为 1

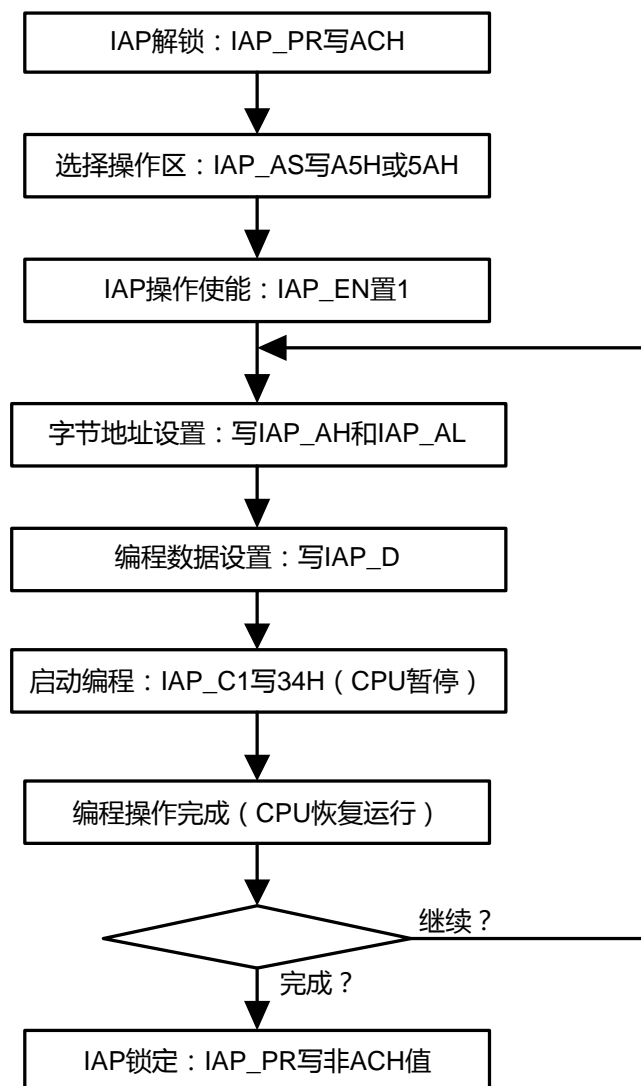
5.4 IAP 操作

MAIN FLASH 和 EEPROM 都支持应用中自编程 IAP 操作。地址由 IAP_AH.IAP_AL 指定,支持字节改写,无需擦除操作。

IAP 操作包括字节编程,读取校验通过 MOVC 指令完成。字节地址由 IAP_AH 和 IAP_AL 指定,编程操作启动时,CPU 自动进入暂停运行状态,直到编程操作完成,CPU 才恢复运行。在此过程中,外设保持当前运行状态,产生的中断请求会置位相应中断标志,但不响应中断服务程序。编程操作完成后,恢复对中断的正常响应。

需要注意的是 MAIN FLASH 编程时间为 1ms,EEPROM 编程时间为 1.5ms。

IAP 操作流程



图表 9 IAP 操作流程图

IAP 寄存器

为保护 MAIN FLASH 存储器不被异常的程序执行误改动，用户可通过 **CFG_WOOD3** 分区设置 IAP 操作权限，同时所有 IAP 寄存器默认是锁定状态。如果要对 IAP 寄存器进行写操作，必须通过 **IAP_PR** 寄存器进行解锁。

IAP_PR IAP 解锁寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
IAP_PR<7:0>							

Bit7-0 **IAP_PR<7:0>** : IAP 操作解锁字
 W : 写入 ACH 解锁，写入其他值锁定
 R : 锁定状态读数为 00H，解锁状态读数为 01H

IAP_AS IAP 区域选择寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
IAP_AS<7:0>							

Bit7-0 **IAP_AS<7:0>** : IAP 区域选择字
 W : 写入 A5H 选择程序区 MAIN FLASH，写入 5AH 选择数据区 EEPROM
 R : 选择程序区读数为 00H，选择数据区读数为 01H

IAP_AL IAP 地址寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
IAP_A<7:0>							

Bit7-0 **IAP_A<7:0>** : IAP 操作地址低 8 位

IAP_AH IAP 地址寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
IAP_A<15:8>							

Bit7-0 **IAP_A<15:8>** : IAP 操作地址高 8 位

IAP_D IAP 数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
IAP_D<7:0>							

Bit7-0 **IAP_D<7:0>** : IAP 操作数据

IAP_C0 IAP 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	—	RW-0
—	—	—	—	—	—	—	IAP_EN

Bit7-1 保留

Bit0 **IAP_EN** : IAP 使能位
1 : 使能
0 : 关闭

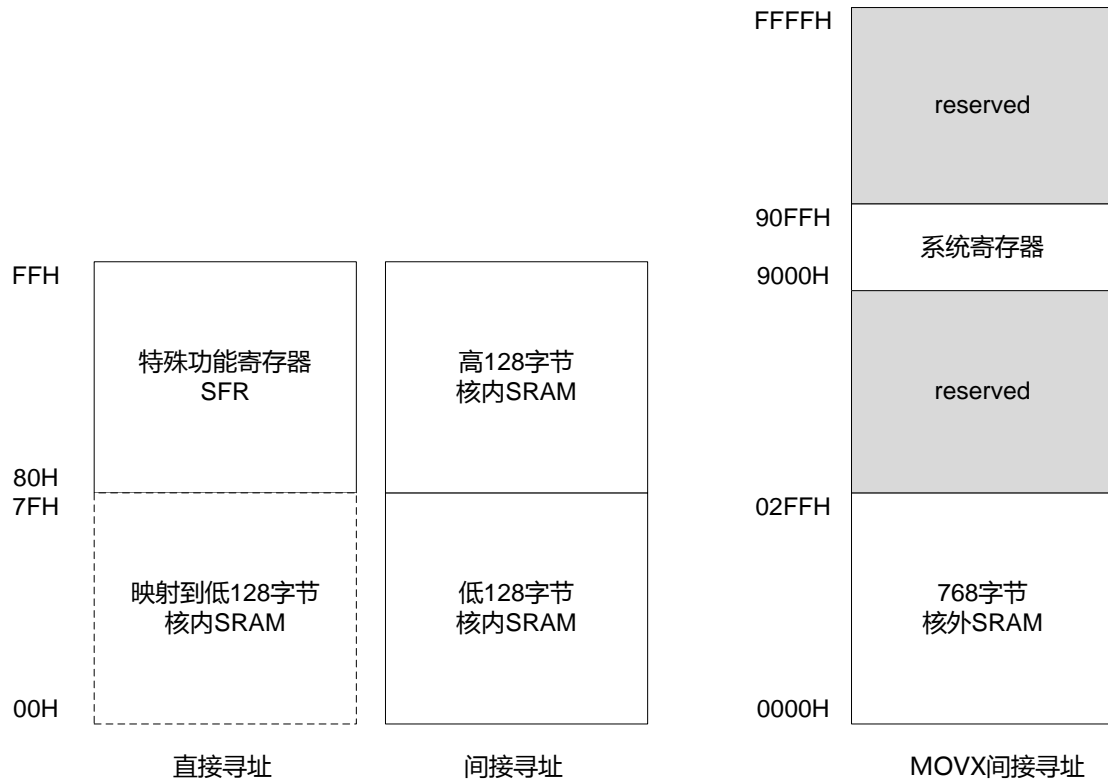
IAP_C1 IAP 控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
W-0000_0000							
IAP_OPS<7:0>							

Bit7-0 **IAP_OPS<7:0>** : IAP 操作触发字
写 34H : 触发 IAP 编程
写其他值 : 无操作

5.5 数据寻址空间

数据寻址空间映射



图表 10 数据寻址空间映射图

SFR 映射

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	
ACC	E0H	A<7:0>								
B	F0H	B<7:0>								
SP	81H	SP<7:0>								
DPL	82H	DPTR<7:0>								
DPH	83H	DPTR<15:8>								
PSW	D0H	CY	AC	F0	RS<1:0>		OV	—	P	
PCON	87H	PCON_PR<3:0>				—	—	IDLE	SLEEP	
RST_FLAG	B7H	—	—	—	ROM_OV	RSTN_F	SRST_F	WDTR_F	POR_F	
INTn_IE	E8H	GIE	INT6_IE	INT5_IE	INT4_IE	INT3_IE	INT2_IE	INT1_IE	INT0_IE	
INTn_IF	88H	—	INT6_IF	INT5_IF	INT4_IF	INT3_IF	INT2_IF	INT1_IF	INT0_IF	
INTn_IP	F8H	—	INT6_IP	INT5_IP	INT4_IP	INT3_IP	INT2_IP	INT1_IP	INT0_IP	
WDT_C	84H	—	—	WDT_PD	WDT_CKS	WDT_CY<3:0>				
WDT_OP	85H	WDT_OP								
HRC_TRML	89H	HRC_TRM<7:0>								
HRC_TRMH	8AH	HRC_PR<3:0>				—	—	HRC_TRM<9:8>		
PORT_C0	86H					PT_RDS<1:0>				
P0_OE	8BH	P0_OE<7:0>								
P1_OE	8CH	P1_OE<7:0>								
P2_OE	8DH	P2_OE<7:0>								
P3_OE	8EH	P3_OE<7:0>								
P0_DAT	80H	P0_DAT <7:0>								
P1_DAT	90H	P1_DAT <7:0>								
P2_DAT	A0H	P2_DAT <7:0>								
P3_DAT	B0H	P3_DAT <7:0>								
P0_IE	9100H	P0_IE<7:0>								
P1_IE	9101H	P1_IE<7:0>								
P2_IE	9102H	P2_IE<7:0>								
P3_IE	9103H	P3_IE<7:0>								
P0_PUE	9108H	P0_PUE<7:0>								
P1_PUE	9109H	P1_PUE<7:0>								
P2_PUE	910AH	P2_PUE<7:0>								

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
P3_PUE	910BH				P3_PUE<7:0>				
P0_PDE	9110H				P0_PDE<7:0>				
P1_PDE	9111H				P1_PDE<7:0>				
P2_PDE	9112H				P2_PDE<7:0>				
P3_PDE	9113H				P3_PDE<7:0>				
P0_ODE	9118H				P0_ODE<7:0>				
P1_ODE	9119H				P1_ODE<7:0>				
P2_ODE	911AH				P2_ODE<7:0>				
P3_ODE	911BH				P3_ODE<7:0>				
P0_SME	9120H				P0_SME<7:0>				
P1_SME	9121H				P1_SME<7:0>				
P2_SME	9122H				P2_SME<7:0>				
P3_SME	9123H				P3_SME<7:0>				
P0_FUN0	9130H	P01_FUN					P00_FUN		
P0_FUN1	9131H	P03_FUN					P02_FUN		
P0_FUN2	9132H	P05_FUN					P04_FUN		
P0_FUN3	9133H	P07_FUN					P06_FUN		
P1_FUN0	9134H	P11_FUN					P10_FUN		
P1_FUN1	9135H	P13_FUN					P12_FUN		
P1_FUN2	9136H	P15_FUN					P14_FUN		
P1_FUN3	9137H	P17_FUN					P16_FUN		
P2_FUN0	9138H	P21_FUN					P20_FUN		
P2_FUN1	9139H	P23_FUN					P22_FUN		
P2_FUN2	913AH	P25_FUN					P24_FUN		
P2_FUN3	913BH	P27_FUN					P26_FUN		
P3_FUN0	913CH	P31_FUN					P30_FUN		
P3_FUN1	913DH	P33_FUN					P31_FUN		
P3_FUN2	913EH	P35_FUN					P34_FUN		
P3_FUN3	913FH	P37_FUN					P36_FUN		
P3_IOL_CL	9156H	P33_IOL		P32_IOL		P31_IOL		P30_IOL	
P3_IOL_CH	9157H	P37_IOL		P36_IOL		P35_IOL		P34_IOL	
P0_IOH_CL	9160H	P03_IOH		P02_IOH		P01_IOH		P00_IOH	
P1_IOH_CL	9162H	P13_IOH		P12_IOH		P11_IOH		P10_IOH	

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	
P1_IOH_CH	9163H	P17_IOH		P16_IOH		P15_IOH		P14_IOH		
EINT0_IF	98H	EINT0_IF<7:0>								
EINT1_IF	D8H	EINT1_IF<7:0>								
EINT0_IE	9180H	EINT0_IE<7:0>								
EINT1_IE	9181H	EINT1_IE<7:0>								
EINT0_MD	9182H	EINT0_MD <7:0>								
EINT1_MD	9183H	EINT1_MD <7:0>								
EINT0_IOS	9184H	EINT0_IOS <7:0>								
EINT1_IOS	9185H	EINT1_IOS <7:0>								
LED_C0	8FH	LED_EN	—	—	—	—	—	LED_SEGM	LED_MODE	
LED_C1	91H	—	LED_GS<2:0>			LED_CKS<1:0>		LED_DS<1:0>		
COM_D	92H	COM_D<7:0>								
COM_EN	93H	C8_EN	C7_EN	C6_EN	C5_EN	C4_EN	C3_EN	C2_EN	C1_EN	
SEGDL	94H	SEGDL								
SEGDH	95H	—				SEGDH				
TMRn_IE	9DH	—	—	—	—	TMR3_IE	TMR2_IE	TMR1_IE	TMR0_IE	
TMRn_IF	A8H	—	—	—	—	TMR3_IF	TMR2_IF	TMR1_IF	TMR0_IF	
PWMn_IE	9EH	—	—	—	—	—	PWM2_IE	PWM1_IE	PWM0_IE	
PWMn_IF	C8H	—	—	—	—	—	PWM2_IF	PWM1_IF	PWM0_IF	
CAPn_IE	9EH	—	—	—	—	—	CAP2_IE	CAP1_IE	CAP0_IE	
CAPn_IF	C8H	—	—	—	—	—	CAP2_IF	CAP1_IF	CAP0_IF	
UART_IE	9FH	—	—	—	—	TX1_IE	RX1_IE	TX0_IE	RX0_IE	
IIC_IE		—	—	—	IIC_IE	—	—	—	—	
SPI_IE		—	SPI_IE	—	—	—	—	—	—	
UART_IF	D8H	—	—	—	—	TX1_IF	RX1_IF	TX0_IF	RX0_IF	
IIC_IF		—	—	—	IIC_IF	—	—	—	—	
SPI_IF		—	SPI_IF	—	—	—	—	—	—	
AN_IE	A1H	—	—	COM_IE	LED_IE	—	LVD_IE	TK_IE	ADC_IE	
AN_IF	C0H	—	—	COM_IF	LED_IF	—	LVD_IF	TK_IF	ADC_IF	
TMR0_C0	A2H	TMR0_EN	TMR0_PRE	TMR0_CKS<1:0>		TMR0_PST<3:0>				
TMR0_C1	A3H	—	—	—	—	—	—	TMR0_GS	TMR0_GE	
TMR0L	A4H	TMR0<7:0>								
TMR0H	A5H	TMR0<15:8>								

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
TMR0_CYL	A6H	TMR0_CY<7:0>							
TMR0_CYH	A7H	TMR0_CY<15:8>							
TMR1_C0	A9H	TMR1_EN	TMR1_PRE	TMR1_CKS<1:0>		TMR1_PST<3:0>			
TMR1_C1	A3H	—	—	—	—	TMR1_GS	TMR1_GE	—	—
TMR1L	AAH	TMR1<7:0>							
TMR1H	ABH	TMR1<15:8>							
TMR1_CYL	ACH	TMR1_CY<7:0>							
TMR1_CYH	ADH	TMR1_CY<15:8>							
TMR2_C0	AEH	TMR2_EN	TMR2_PRE	TMR2_CKS<1:0>		TMR2_PST<3:0>			
TMR2_C1	A3H	—	—	TMR2_GS	TMR2_GE	—	—	—	—
TMR2L	B1H	TMR2<7:0>							
TMR2H	B2H	TMR2<15:8>							
TMR2_CYL	B3H	TMR2_CY<7:0>							
TMR2_CYH	B4H	TMR2_CY<15:8>							
TMR3_C0	AFH	TMR3_EN	TMR3_PRE	TMR3_CKS<1:0>		TMR3_PST<3:0>			
TMR3_C1	A3H	TMR3_GS	TMR3_GE	—	—	—	—	—	—
TMR3L	B5H	TMR3<7:0>							
TMR3H	B6H	TMR3<15:8>							
TMR3_CYL	B9H	TMR3_CY<7:0>							
TMR3_CYH	BAH	TMR3_CY<15:8>							
PWM0_C	BBH	PWM0_SPS<1:0>		—	—	PWM0_TBS<1:0>		PWM0_MOD<1:0>	
PWM0_OC	BCH	—	PWMn_REX			PWMn1_T	PWMn0_T	PWMn1_P	PWMn0_P
PWM0_R0L	BDH	PWM0_R0<7:0>							
PWM0_R0H	BEH	PWM0_R0<15:8>							
PWM0_R1L	C1H	PWM0_R1<7:0>							
PWM0_R1H	C2H	PWM0_R1<15:8>							
PWM0_DL	BFH	PWM0_DL<7:0>							
PWM1_C	C3H	PWM1_SPS<1:0>		—	—	PWM1_TBS<1:0>		PWM1_MOD<1:0>	
PWM1_OC	C4H	—	PWMn_REX			PWMn1_T	PWMn0_T	PWMn1_P	PWMn0_P
PWM1_R0L	C5H	PWM1_R0<7:0>							
PWM1_R0H	C6H	PWM1_R0<15:8>							
PWM1_R1L	C9H	PWM1_R1<7:0>							
PWM1_R1H	CAH	PWM1_R1<15:8>							

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PWM1_DL	C7H	PWM1_DL<7:0>							
PWM2_C	CBH	PWM2_SPS<1:0>		—		PWM2_TBS<1:0>		PWM2_MOD<1:0>	
PWM2_OC	CCH	—		PWMn_REX		PWMn1_T	PWMn0_T	PWMn1_P	PWMn0_P
PWM2_R0L	CDH	PWM2_R0<7:0>							
PWM2_R0H	CEH	PWM2_R0<15:8>							
PWM2_R1L	D1H	PWM1_R1<7:0>							
PWM2_R1H	D2H	PWM2_R1<15:8>							
PWM2_DL	CFH	PWM2_DL<7:0>							
UART0_BRL	D3H	UART_BRR<7:0>							
UART0_BRH	D4H	UART_BRR<15:8>							
UART0_RXC	D5H	FERR	OERR	PERR	RX9D	PARS	BRFX	RXEN<1:0>	
UART0_RXB	D6H	RXB<7:0>							
UART0_TXC	D7H	TXST	TXBF	TX9S<1:0>		STPS	TXDM	TXEN<1:0>	
UART0_TXB	D9H	TXB<7:0>							
UART1_BRL	FAH	UART_BRR<7:0>							
UART1_BRH	FBH	UART_BRR<15:8>							
UART1_RXC	FCH	FERR	OERR	PERR	RX9D	PARS	BRFX	RXEN<1:0>	
UART1_RXB	FDH	RXB<7:0>							
UART1_TXC	FEH	TXST	TXBF	TX9S<1:0>		STPS	TXDM	TXEN<1:0>	
UART1_TXB	FFH	TXB<7:0>							
IIC_C0	DAH	IIC_SMPF<3:0>				IIC_IM	WTEN	IIC_MD	IIC_EN
IIC_C1	DBH	—		—		MTAI_MK	STOP	WAIT	ACK
IIC_STA	DCH	SLV_ADF	SLV_RWF	STOP_F	START_F	ACK_F	BUF_ST	OVT_F	BFOV_F
IIC_ADDR	DDH	IIC_ADDR<6:0>							
IIC_BRR		IIC_BRR<6:0>							
IIC_DATA	DEH	IIC_DATA							
IIC_TOC	DFH	TOC_EN	—		IIC_OVT				
SPI_C0	F6H	SPI_SSC	SPI_SSD	SPI_CPOL	SPI_CPHA	SPI_DIR	SPI_CLK		
SPI_C1	F7H	SPI_EN	SPI_MS	—		SPI_TXSP	SPI_RXSP	SPI_WOL	SPI_RXSP
SPI_DATA	F9H	SPI_DATA							
ADC_C0	E1H	ADC_EN	S_TRG	TRG_S<1:0>		CH_SEL<3:0>			
ADC_C1	E2H	VREFP_S<3:0>				—		ADC_DM	ADC_CKS<1:0>
ADC_C2	E3H	—		—		ADC_CNT<1:0>		SMP_T<2:0>	

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ADC_DL	E4H	ADC_DL<7:0>							
ADC_DH	E5H	ADC_DH<7:0>							
ADC_CMP_C	E6H	ADC_CP_E	MAX_E	MIN_E	INT_MD	MAX_F	MIN_F	ADC_DMD<1:0>	
TK_C0	EAH	TK_EN	TK_VTKS	—	—	TK_OV	TK_ERR	—	TK_GO
TK_C1	E9H	TK_FLT<3:0>				—	TK_VRFS<2:0>		
TK_C2	EBH	SOFT_CK	TK_DIV<4:0>				TK_CKS<1:0>		
TK_CHEL	ECH	TK_CHE<7:0>							
TK_CHEM	EDH	TK_CHE<15:8>							
TK_CNTL	F1H	TK_CNT<7:0>							
TK_CNTH	F2H	TK_CNT<15:8>							
VREF_C0	EFH	VREF_EN	—	—	—	—	—	—	—
LVD_C0	F3H	LVD_EN	LVD_IM	LVD_FLT	LVD_CKS	LVD_INS	LVD_VS<2:0>		

系统寄存器映射

SYS_SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
IAP_PR	9000H	IAP_PR<7:0>							
IAP_AL	9002H	IAP_A<7:0>							
IAP_AH	9003H	IAP_A<15:8>							
IAP_D	9004H	IAP_D<7:0>							
IAP_C0	9005H	—	—	—	—	—	—	—	IAP_EN
IAP_C1	9006H	EE_OPS<7:0>							
CLK_PR	9020H	CLK_PR<7:0>							
CLK_C0	9021H	MEM_MD<1:0>		CLK_S<1:0>		—	CLK_DIV<2:0>		
CLK_C1	9022H	OSC_F<1:0>		XOSC_ST	HRC_ST	SRAM_SEL<1:0>		XOSC_EN	HRC_EN
PCK_GTC0	9023H	—	PWM2_G	PWM1_G	PWM0_G	TMR3_G	TMR2_G	TMR1_G	TMR0_G
PCK_GTC1	9024H	LED_G	TK_G	—	ADC_G	IIC_G	SPI_G	UART1_G	UART0_G
CLK_LP	9026H	—	—	—	CLK_LEN	—	—	—	FLT_BPS
WKUP_T	9027H	—	—	—	—	WKUP_T<3:0>			
SOFT_RST	90FFH	SOFT_RST							

位操作映射

寄存器		BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
P0_DAT	80H	P0_n<7:0>							
INTn_IF	88H	—	INT6_IF	INT5_IF	INT4_IF	INT3_IF	INT2_IF	INT1_IF	INT0_IF
P1_DAT	90H	P1_n<7:0>							
EINT0n_IF	98H	EINT07_IF	EINT06_IF	EINT05_IF	EINT04_IF	EINT03_IF	EINT02_IF	EINT01_IF	EINT00_IF
P2_DAT	A0H	P2_n<7:0>							
TMRn_IF	A8H	—	—	—	—	TMR3_IF	TMR2_IF	TMR1_IF	TMR0_IF
P3_DAT	B0H	P3_n<7:0>							
AN_IF	C0H	—	—	COM_IF	LED_IF	—	LVD_IF	TK_IF	ADC_IF
PWMn_IF	C8H	—	—	—	—	—	PWM2_IF	PWM1_IF	PWM0_IF
PSW	D0H	CY	AC	F0	RS<1:0>		OV	—	P
EINT1n_IF	D8H	EINT17_IF	EINT16_IF	EINT15_IF	EINT14_IF	EINT13_IF	EINT12_IF	EINT11_IF	EINT10_IF
ACC	E0H	A<7:0>							
INTn_IE	E8H	GIE	INT6_IE	INT5_IE	INT4_IE	INT3_IE	INT2_IE	INT1_IE	INT0_IE
B	F0H	B<7:0>							
INTn_IP	F8H	—	INT6_IP	INT5_IP	INT4_IP	INT3_IP	INT2_IP	INT1_IP	INT0_IP

6 复位

6.1 描述

程序溢出复位

由于程序执行异常，程序计数器 PC 指向合法程序空间之外取指时，产生程序溢出复位。程序溢出复位同时，将 **RST_FLAG.ROM_OV** 标志位置 1，该标志可通过软件清 0。

RSTN 外部端口复位

当用户配置选项 **CFG_WOOD2.RSTN_EN** 位使能时，RSTN 复用端口复用为外部复位功能。当 RSTN 端口输入有效宽度的低电平时，使芯片复位。RSTN 复位同时，将 **RST_FLAG.RSTN_F** 标志位置 1，该标志可通过软件清 0。

SRST 软件复位

对软件复位寄存器 **SOFT_RST** 写入 5AH，在写入 A5H 产生软件复位。软件复位同时，将 **RST_FLAG.SRST_F** 标志位置 1，该标志可通过软件清 0。

WDT 复位

芯片内置硬件看门狗电路 WDT。在 WDT 溢出标志存在的情况下（WDT_OP 读出不为 00H），再次产生 WDT 溢出，会产生 WDT 复位。WDT 复位同时，将 **RST_FLAG.WDTR_F** 标志置 1，该标志可通过软件清 0。

POR 和 BOR 复位

POR 复位是芯片的内部复位，复位自动发生于芯片初始上电或芯片电源发生极端异常波动后的上电恢复。POR 硬件强制使能。

BOR 复位是芯片的内部复位，主要用于芯片电源 VDD 跌落至用户设定的 BOR 复位门限电压以下时产生芯片复位。用户可根据系统需要，使能或关闭 BOR 功能，并可选择不同的 BOR 的复位门限电压。POR 或者 BOR 复位发生时，将 **RST_FLAG.POR_F** 标志置 1，并将 **RST_FLAG** 寄存器的其他标志位复位清 0，POR_F 标志位可通过软件清 0。

6.2 复位寄存器

RST_FLAG 复位标志寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	RW-0	RW-0	RW-0	RW-0
—	—	—	ROM_OV	RSTN_F	SRST_F	WDTR_F	POR_F

Bit7-5 保留

Bit4 **ROM_OV** : 程序溢出复位标志位

1 : 程序执行溢出产生芯片复位

0 : 未发生程序执行溢出

Bit3 **RSTN_F** : RSTN 复位标志位

1 : 通过 RSTN 复用端口产生芯片复位

0 : 未发生 RSTN 复用端口复位

Bit2 **SRST_F** : 软件复位标志位

1 : 通过写 SOFT_RST 寄存器产生软件复位

0 : 未发生软件复位

Bit1 **WDTR_F** : WDT 复位标志位

1 : WDT 溢出产生复位

0 : 未发生 WDT 溢出复位

Bit0 **POR_F** : POR 上电复位标志位

1 : 发生 POR 上电复位 (系统初次上电或系统电源的异常跌落恢复产生的重新上电)

0 : 未发生 POR 上电复位

SOFT_RST 软件复位寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
W-0000_0000							
SOFT_RST<7:0>							

Bit7-0 **SOFT_RST<7:0>** : 软件复位字

W : 先写 5AH,在写入 A5H 产生软件复位

SOFT_BOR 软件复位寄存器(90F1H)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	RW-000		
—	—	—	—	—	SOFT_BOR<2:0>		

Bit7-3 保留

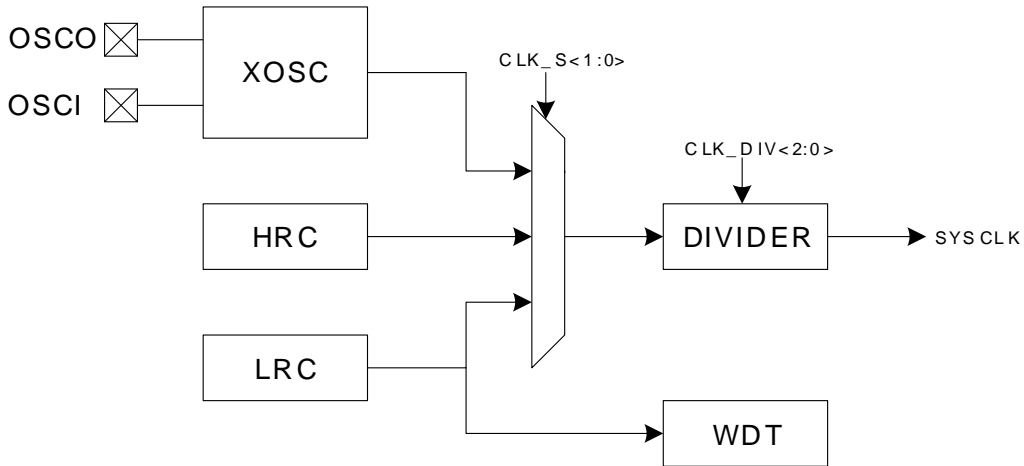
Bit2-0 **SOFT_BOR<2:0>** : 软件 BOR(当配置字 BOR 设置为最低档位时)

111 : 4.4V 110 : 4.0V 101 : 3.6V 100 : 3.2V

011 : 2.8V 010 : 2.4V 001 : 2.0V 000 : 1.6V

7 时钟

7.1 描述



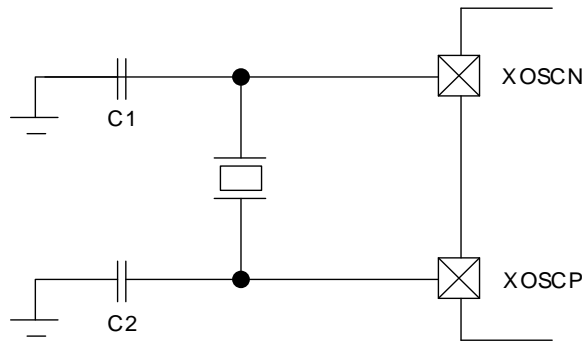
图表 11 系统时钟源功能框图

外部振荡器 XOSC

外部振荡器 XOSC 支持 2 种振荡模式，LP 模式和 HS 模式。晶振连接请靠近芯片管脚端。通过用户配置选项 `CFG_WD1.XOSC_MD<7:0>` 配置。

LP 模式适用于外接 32KHz 低频晶振(C1=33PF, C2=33PF)。

HS 模式适用于外接 1~25MHz 高频晶振(C1=15PF, C2=15PF)。。



图表 12 外部振荡器 XOSC 连接示意图

内部高频 RC 时钟 HRC

芯片内置一个高频 RC 时钟，有 32M 或者 28M 或者 16M 三个频点可以设置，读取校准区为 0x8027 和 0x8026 写入 HRC_TRMH 和 HRC_TRML 时钟 HRC 为 16M, 读取校准区为 0x8023 和 0x8022 写入 HRC_TRMH 和 HRC_TRML 时钟 HRC 为 32M, 读取校准区为 0x8025 和 0x8024 写入 HRC_TRMH 和 HRC_TRML 时钟 HRC 为 28M (有部分早期产品不支持 28M 这个频点)，本时钟用于系统和外设时钟源。HRC 出厂校准精度 $\pm 1\%$ (T=25°C)。

内部低频 RC 时钟 LRC

芯片内置 16KHz 低频 RC 时钟，用于系统和外设时钟源，同时用于 WDT 时钟源。

7.2 最大工作频率说明

工作条件	VDD	最大工作频率 (MHz)
芯片运行时钟不分频	4.5~5.5V	14M
	2.7~5.5V	8M
	1.8~5.5V	2M

7.3 时钟寄存器

为保护时钟相关寄存器不被异常的程序执行误改动，所有时钟寄存器默认是锁定状态。如果要对时钟寄存器进行改写，必须通过 **CLK_PR** 寄存器进行解锁。

CLK_PR 时钟解锁寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
CLK_PR<7:0>							

Bit7-0 **CLK_PR<7:0>** : 时钟寄存器解锁字
 W : 写入 A5H 解锁，写入其他值锁定
 R : 锁定状态读数为 00H，解锁状态读数为 01H

CLK_C0 时钟控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		—	RW-011		
MEM_MD<1:0>		CLK_S<1:0>		—	CLK_DIV<2:0>		

Bit7-6 **MEM_MD<1:0>** (不关注低速运行功耗，请保持 MEM_MD 为 00)
 11 : 低速 500K
 10 : 中速 (2V 可运行 1M，2.5V 可运行 4M，4.5V 以上可运行 14M)
 0X : 高速

Bit5-4 **CLK_S<1:0>** : 系统时钟源选择位
 11 : 保留
 10 : 选择 LRC
 01 : 选择 XOSC
 00 : 选择 HRC

Bit3 保留

Bit2-0 **CLK_DIV<2:0>** : 系统时钟分频位
 系统时钟频率 $F_{SYSCLK} = \frac{F_{SOURCE}}{2^{CLK_DIV<2:0>}}$

CLK_C1 时钟控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R-00		R-0	R-0	—	—	RW-0	RW-0
OSC_F<1:0>		XOSC_ST	HRC_ST	—	—	XOSC_EN	HRC_EN

- Bit7-6 **OSC_F<1:0>** : 系统时钟源状态位
 11 : 保留
 10 : 当前系统时钟源为 LRC
 01 : 当前系统时钟源为 XOSC
 00 : 当前系统时钟源为 HRC
- Bit5 **XOSC_ST** : 外部振荡器 XOSC 工作状态位
 1 : XOSC 已进入稳定工作状态
 0 : XOSC 启动中
- Bit4 **HRC_ST** : 内部高速振荡器 HRC 工作状态位
 1 : HRC 已进入稳定工作状态
 0 : HRC 启动中
- Bit3-2 保留
- Bit1 **XOSC_EN** : 外部振荡器 XOSC 使能位
 1 : 使能
 0 : 关闭
- Bit0 **HRC_EN** : 内部高速振荡器 HRC 使能位
 1 : 使能
 0 : 关闭

PCK_GTC0 外设时钟控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-1	RW-1	RW-1	RW-1	RW-1	RW-1	RW-1
—	CP2_G	CP1_G	CP0_G	TMR3_G	TMR2_G	TMR1_G	TMR0_G

- Bit7 保留
- Bit6-4 **CPn_G** : CAP0~CAP2 和 PWM0~PWM2 外设时钟使能位
 1 : 使能
 0 : 关闭
- Bit3-0 **TMRn_G** : TMR0~ TMR3 外设时钟使能位
 1 : 使能
 0 : 关闭

PCK_GTC1 外设时钟控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-1	RW-1	—	RW-1	RW-1	RW-1	RW-1	RW-1
LED_G	TK_G	—	ADC_G	IIC_G	SPI_G	UART1_G	UART0_G

- Bit7 **LED_G** : LED 外设时钟使能位
 1 : 使能
 0 : 关闭
- Bit6 **TK_G** : TK 外设时钟使能位
 1 : 使能
 0 : 关闭
- Bit5 保留
- Bit4 **ADC_G** : ADC 外设时钟使能位
 1 : 使能
 0 : 关闭
- Bit3 **IIC_G** : IIC 外设时钟使能位
 1 : 使能
 0 : 关闭
- Bit2 **SPI_G** : SPI 外设时钟使能位
 1 : 使能
 0 : 关闭
- Bit1-0 **UARTn_G** : UART0~UART1 外设时钟使能位
 1 : 使能
 0 : 关闭

CLK_LP 时钟低功耗控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	—	—	—	RW-0
—	—	—	CLK_LEN	—	—	—	FLT_BPS

- Bit7-5 保留
- Bit4 **CLK_LEN** : SLEEP 模式下系统时钟源使能位
 1 : 系统时钟源在 SLEEP 模式下保持工作
 0 : 系统时钟源在 SLEEP 模式下关闭
- Bit3-1 保留
- Bit0 **FLT_BPS** : 系统时钟滤波器旁路控制位
 1 : 系统时钟滤波器关闭 (旁路)
 0 : 系统时钟滤波器使能

WKUP_T SLEEP 唤醒等待时间寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—				RW-1111			
—				WKUP_T<3:0>			

Bit3-0 **WKUP_T<3:0>** : SLEEP 唤醒等待时间控制位
唤醒等待时间 $T_{WKUP} = (WKUP_T<3:0> + 1) \times 16 \times T_{SYS_CLK}$

HRC_TRML HRC 调校寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-1000_0000							
HRC_TRM<7:0>							

Bit7-0 **HRC_TRM<7:0>** : HRC 频率调校位低 8 位。校准数据低位写完成后生效。

HRC_TRMH HRC 调校寄存器高 2 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				—	—	RW-00	
HRC_PR<3:0>				—	—	HRC_TRM<9:8>	

Bit7-4 **HRC_PR<3:0>** : HRC 调校位写保护
W : 写入 5H 锁定, 写入 AH 解锁
R : 锁定状态读为 0H, 解锁状态读为 1H

Bit3-2 保留

Bit1-0 **HRC_TRM<9:8>** : HRC 频率调校位高 2 位

注 : HRC_TRML 和 HRC_TRMH 寄存器不受 CLK_PR 控制。

8 低功耗

8.1 描述

芯片支持 2 种低功耗模式，IDLE 模式和 SLEEP 模式。

IDLE 模式

IDLE 模式下，CPU 暂停执行指令，系统时钟和外设均保持当前的工作状态。

用户可关闭不需要运行模块的使能位，并通过 PCK_GTC0 寄存器关闭相应模块的时钟，以节省不必要的功率消耗。

保持运行的外设产生的中断，如果相应的中断使能位为 1（GIE 不需使能），可将芯片从 IDLE 状态唤醒。如果 GIE 使能，唤醒芯片后，会进行中断服务程序。IDLE 唤醒无等待时间，唤醒后 CPU 继续运行。

SLEEP 模式

SLEEP 模式下，系统时钟自动关闭，CPU 和所有采用系统时钟工作的外设模块均暂停工作。

采用非系统时钟工作的外设可在 SLEEP 模式下保持工作，并且产生的中断，如果相应的中断使能位为 1（GIE 不需使能），可将芯片从 SLEEP 状态唤醒。如果 GIE 使能，唤醒芯片后，会进行中断服务程序。SLEEP 唤醒需要一定的等待时间，用户可通过 WKUP_T 寄存器设定，等待时间用以确保芯片恢复运行前内部的部分模块已达到稳定工作状态，唤醒后系统时钟恢复运行，CPU 和经系统时钟同步的外设继续运行。

8.2 低功耗寄存器

PCON 低功耗控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				—	—	RW-0	RW-0
PCON_PR<3:0>				—	—	IDLE	SLEEP

Bit7-4 **PCON_PR<3:0>**：PCON 解锁字

W：写入 5H 解锁，写入其他值锁定

R：锁定状态读数为 0H，解锁状态读数为 1H

Bit3-2 保留

Bit1 **IDLE**：IDLE 模式使能位

1：进入 IDLE 模式（仅在 PCON_PR 解锁状态下，对 PCON 寄存器写 02H 可置 1）

0：退出 IDLE 模式（写 0 无效，唤醒后硬件自动清 0）

Bit0 **SLEEP**：SLEEP 模式使能位

1：进入 SLEEP 模式（仅在 PCON_PR 解锁状态下，对 PCON 寄存器写 01H 可置 1）

0：退出 SLEEP 模式（写 0 无效，唤醒后硬件自动清 0）

9 看门狗定时器 WDT

9.1 描述

芯片内置 16 位硬件看门狗定时器 WDT。支持 2 种时钟源可选，支持溢出周期可配置。

WDT 溢出唤醒

当 16 位 WDT 计数器累加到与 **WDT_CYC** 位所选择的值相等时，WDT 计数器溢出。溢出后 WDT 计数器从 0 开始累加。在 SLEEP 或 IDLE 下，WDT 两次溢出可将 CPU 从低功耗模式下唤醒。读 **WDT_OP** 寄存器可判断溢出标志。

WDT 溢出复位

当 16 位 WDT 计数器累加到与 **WDT_CYC** 位所选择的值相等时，WDT 计数器溢出。溢出后 WDT 计数器从 0 开始累加。在 MCU 运行状态选，WDT 两次溢出芯片将产生复位。WDT 复位同时将 WDT 复位标志 **RST_FLAG.WDTF** 置 1。

喂狗操作

对 WDT_OP 寄存器写 5AH 即可进行喂狗操作，喂狗操作将 WDT 计数器清 0，同时清除 WDT 溢出标志。另外操作芯片进入 SLEEP 或者 IDLE 时，同时产生喂狗动作。

9.2 WDT 寄存器

WDT_C WDT 控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	RW-0	RW-0	RW-0			
—	—	WDT_PD	WDT_CKS	WDT_CYC			

Bit7-6 保留

Bit5 **WDT_PD**: 软件 WDT 关断位 (仅在配置字关闭后生效)

- 1 : 关闭 WDT
- 0 : 使能 WDT

Bit4 **WDT_CKS**: WDT 时钟源选择位

- 1 : 内部低频时钟 LRC
- 0 : 系统时钟 SYS_CLK

Bit3-0 **WDT_CYC** <3:0>: WDT 溢出周期寄存器

0000 : 2560ms	0001 : 2000ms	0010 : 1500ms	0011 : 1000ms
0100 : 900ms	0101 : 800ms	0110 : 700ms	0111 : 600ms
1000 : 500ms	1001 : 400ms	1010 : 300ms	1011 : 200ms
1100 : 100ms	1101 : 60ms	1110 : 30ms	1111 : 20ms

WDT_OP WDT 操作寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
WDT_OP <7:0>							

Bit7-0 **WDT_OP**<7:0>: WDT 操作字

W : 写入 5AH 将 WDT 计数器清 0，且清除 WDT 溢出标志

R : WDT 未溢出读为 00H，WDT 溢出读为 01H，WDT 溢出唤醒读为 03H

10 中断

10.1 中断向量

芯片共支持 7 个中断向量，每个中断向量对应单独的入口地址。如下表所示，芯片的所有中断源都有各自的中断标志和中断使能位，这些中断源被分组对应到 7 个中断向量。同时，每个中断向量也有一个向量使能位 $INTn_IE$ ，并且所有的中断向量还共用 1 个总的使能位 $INTn_IE.GIE$ 。GIE 禁止时所有中断不响应，但向量和中断源使能的中断仍支持低功耗模式唤醒功能。

向量编号	对应 Interrupt	入口地址	向量使能	向量标志	中断源使能	中断源标志
INT0	0	0003H	INT0_IE	INT0_IF	EINT0_IE	EINT0_IF
INT1	2	0013H	INT1_IE	INT1_IF	EINT1_IE	EINT1_IF
INT2	7	003BH	INT2_IE	INT2_IF	TMRn_IE	TMRn_IF
					CAPn_IE	CAPn_IF
					PWMn_IE	PWMn_IF
					STPn_IE	STPn_IF
INT3	8	0043H	INT3_IE	INT3_IF	TXn_IE	TXn_IF
					RXn_IE	RXn_IF
					IIC_IE	IIC_IF
					SPI_IE	SPI_IF
INT4	9	004BH	INT4_IE	INT4_IF	TK_IE	TK_IF
INT5	10	0053H	INT5_IE	INT5_IF	ADC_IE	ADC_IF
INT6	11	005BH	INT6_IE	INT6_IF	LVD_IE	LVD_IF
					LED_IE	LED_IF
					COM_IE	COM_IF

图表 13 中断向量表

10.2 中断优先级

中断系统分为 2 个优先级阶，即高阶优先级和低阶优先级，每个中断向量可通过相应的 $INTn_IP$ 位单独设置优先级阶。在同阶优先级中，中断向量号越小的中断向量优先级越高。同一中断向量对应的多个中断源不分优先级，用户需在对应于该向量入口地址的中断服务程序中进行软件判别。

高优先级的中断可嵌套低优先级中断。反之，低优先级中断只能等待高优先级或同级中断完成中断处理并退出中断服务程序后才可得到响应。

10.3 中断寄存器

INTn_IE 中断使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0
GIE	INT6_IE	INT5_IE	INT4_IE	INT3_IE	INT2_IE	INT1_IE	INT0_IE

Bit7 **GIE** : 中断总使能位
1 : 使能
0 : 禁止 (仅禁止中断响应, 不禁止中断唤醒)

Bit6-0 **INTn_IE** : 中断向量 INT0~6 使能位
1 : 使能
0 : 禁止 (禁止中断响应和中断唤醒)

INTn_IF 中断标志寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0
—	INT6_IF	INT5_IF	INT4_IF	INT3_IF	INT2_IF	INT1_IF	INT0_IF

Bit7 保留

Bit6-0 **INTn_IF** : 中断向量 INT0~6 标志位
1 : 有中断请求
0 : 无中断请求

INTn_IP 中断向量优先级寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0
—	INT6_IP	INT5_IP	INT4_IP	INT3_IP	INT2_IP	INT1_IP	INT0_IP

Bit7 保留

Bit6-0 **INTn_IP** : 中断向量 INT0~6 优先级选择位
1 : 高阶优先级
0 : 低阶优先级

10.1 外部端口中断 EINT

芯片支持 16 个外部端口中断源 EINT0<7:0>和 EINT1<7:0>，每个 EINT 中断源又支持 2 个端口可选。每个 EINT 中断源可独立设置中断触发方式。

EINT0_IE EINT0 使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT0_IE<7:0>							

Bit7-0 **EINT0_IE<7:0>** : 外部端口中断 EINT0 使能位
 1 : 使能
 0 : 禁止

EINT0_IF EINT0 标志寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT0_IF<7:0>							

Bit7-0 **EINT0_IF<7:0>** : 外部端口中断 EINT0 标志位
 1 : 有中断请求
 0 : 无中断请求

EINT1_IE EINT1 使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT1_IE<7:0>							

Bit7-0 **EINT1_IE<7:0>** : 外部端口中断 EINT1 使能位
 1 : 使能
 0 : 禁止

EINT1_IF EINT1 标志寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT1_IF<7:0>							

Bit7-0 **EINT1_IF<7:0>** : 外部端口中断 EINT1 标志位
 1 : 有中断请求
 0 : 无中断请求

EINT0_MD EINT0 触发选择寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT0_M<7:0>							

Bit7-0 **EINT0_MD<7:0>** : EINT0 触发方式选择位
 1 : 下降沿触发
 0 : 上升沿触发

EINT1_MD EINT1 触发选择寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT1_M<7:0>							

Bit7-0 **EINT1_MD<7:0>** : EINT1 触发方式选择位

1 : 下降沿触发

0 : 上升沿触发

EINT0_IOS EINT0 端口选择寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT_IOS<7:0>							

Bit7-0 **EINT0_IOS<7:0>** : EINT0~7 端口选择位

EINT0_IOS	EINT7	EINT6	EINT5	EINT4	EINT3	EINT2	EINT1	EINT0
1	P27	P26	P25	P24	P23	P22	P21	P20
0	—	—	P05	P04	P03	P02	P01	P00

EINT1_IOS EINT1 端口选择寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
EINT1_IOS<7:0>							

Bit7-0 **EINT1_IOS<7:0>** : EINT0~7 端口选择位

EINT1_IOS	EINT7	EINT6	EINT5	EINT4	EINT3	EINT2	EINT1	EINT0
1	P37	P36	P35	P34	P33	P32	P31	P30
0	P17	P16	P15	P14	P13	P12	P11	P10

10.2 定时器 TMR 中断

芯片的 4 个定时器 TMR0~3 都支持中断。当 TMRn 的 16 位计数器累加到与 16 位周期寄存器 TMRn_CY 相等时，TMRn 产生 1 次溢出。当溢出次数达到后分频位 TMRn_C0.TMRn_PST<3:0> 所设定的次数时，即产生 TMRn 中断。

TMRn_IE TMR 中断使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TMR3_IE	TMR2_IE	TMR1_IE	TMR0_IE

Bit7-4 保留

Bit3-0 **TMRn_IE** : TMRn 中断使能位

1 : 使能

0 : 禁止

TMRn_IF TMR 中断标志寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TMR3_IF	TMR2_IF	TMR1_IF	TMR0_IF

Bit7-4 保留

Bit3-0 **TMRn_IF** : TMRn 中断标志位

1 : 有 TMRn 中断请求

0 : 无 TMRn 中断请求

10.3 边沿捕捉器 CAP 中断

芯片的 3 个边沿捕捉器 CAP0~2 都支持中断。当满足用户设定的捕捉条件的捕捉事件发生时，即产生捕捉中断。

CAPn_IE CAP 中断使能寄存器 (与 PWMn_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	RW-0	RW-0	RW-0
—	—	—	—	—	CAP2_IE	CAP1_IE	CAP0_IE

Bit7-3 保留

Bit2-0 **CAPn_IE** : CAPn 中断使能位

1 : 使能

0 : 禁止

CAPn_IF CAP 中断标志寄存器 (与 PWMn_IF 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	RW-0	RW-0	RW-0
—	—	—	—	—	CAP2_IF	CAP1_IF	CAP0_IF

Bit7-3 保留

Bit2-0 **CAPn_IF** : CAPn 中断使能位

1 : 有 CAPn 中断请求

0 : 无 CAPn 中断请求

10.4 脉宽调制器 PWM 中断

芯片的 3 个脉宽调制器 PWM0~2 都支持 2 种中断，即 PWM 周期中断和 PWM 刹车中断。

PWM 周期中断

当 PWMn 所选时基的计数器值累加到该时基对应的周期值时，即产生 PWM 周期中断。

PWM 刹车中断

当 PWMn 发生刹车事件时，会产生 PWM 刹车中断。

PWMn_IE PWM 中断使能寄存器 (与 CAPn_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	RW-0	RW-0	—	RW-0	RW-0	RW-0
—	STP2_IE	STP1_IE	STP0_IE	—	PWM2_IE	PWM1_IE	PWM0_IE

Bit7 保留

Bit6-4 **STPn_IE** : PWMn 刹车中断使能位
1 : 使能
0 : 禁止

Bit3 保留

Bit2-0 **PWMn_IE** : PWMn 周期中断使能位
1 : 使能
0 : 禁止

PWMn_IF PWM 中断标志寄存器 (与 CAPn_IF 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	RW-0	RW-0	—	RW-0	RW-0	RW-0
—	STP2_IF	STP1_IF	STP0_IF	—	PWM2_IF	PWM1_IF	PWM0_IF

Bit7 保留

Bit6-4 **STPn_IF** : PWMn 刹车中断标志位
1 : 有 PWMn 刹车中断请求
0 : 无 PWMn 刹车中断请求

Bit3 保留

Bit2-0 **PWMn_IF** : PWMn 周期中断标志位
1 : 有 PWMn 周期中断请求
0 : 无 PWMn 周期中断请求

10.5 异步收发器 UART 中断

UARTn 支持 2 个中断，即 TXn 发送中断和 RXn 接收中断。

TX 发送中断

根据 UARTn_TXC.TXEN<3:0>位的配置，TXn 发送中断可以是 UARTn 发送空闲产生中断，或发送寄存器空产生中断，或两者都产生中断。

RX 接收中断

UARTn 接收寄存器接收到 1 帧数据，即产生 RXn 接收中断。

UART_IE **UART 中断使能寄存器 (与 IIC_IE 寄存器复用)**

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TX1_IE	RX1_IE	TX0_IE	RX0_IE

- Bit7-4 保留
- Bit3 **TX1_IE** : UART1 发送中断使能位
1 : 使能
0 : 禁止
- Bit2 **RX1_IE** : UART1 接收中断使能位
1 : 使能
0 : 禁止
- Bit1 **TX0_IE** : UART0 发送中断使能位
1 : 使能
0 : 禁止
- Bit0 **RX0_IE** : UART0 接收中断使能位
1 : 使能
0 : 禁止

UART_IF **UART 中断标志寄存器 (与 IIC_IF 寄存器复用)**

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TX1_IF	RX1_IF	TX0_IF	RX0_IF

- Bit7-2 保留
- Bit1 **TX1_IF** : UART1 发送中断标志位
1 : 有 UART1 发送中断请求
0 : 无 UART1 发送中断请求
- Bit0 **RX1_IF** : UART1 接收中断标志位
1 : 有 UART1 接收中断请求
0 : 无 UART1 接收中断请求
- Bit1 **TX0_IF** : UART0 发送中断标志位
1 : 有 UART0 发送中断请求
0 : 无 UART0 发送中断请求
- Bit0 **RX0_IF** : UART0 接收中断标志位
1 : 有 UART0 接收中断请求
0 : 无 UART0 接收中断请求

10.6 IIC 总线控制器中断

IIC 总线控制器支持 1 个中断。通过 IIC_C0.IIC_IM 位可配置如下事件产生 IIC 中断。

START/STOP 位中断

检测到总线上有 START 或 STOP 位，产生中断。

发送/接收中断

完成 1 个字节的发送或接收，产生中断。

IIC_IE IIC 中断使能寄存器 (与 UART_IE SPI_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	—	—	—	—
—	—	—	IIC_IE	—	—	—	—

Bit7-5 保留

Bit4 IIC_IE : IIC 中断使能位

1 : 使能

0 : 禁止

Bit3-0 保留

IIC_IF IIC 中断标志寄存器 (与 UART_IF SPI_IF 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	—	—	—	—
—	—	—	IIC_IF	—	—	—	—

Bit7-5 保留

Bit4 IIC_IF : IIC 中断标志位

1 : 有中断请求

0 : 无中断请求

Bit3-0 保留

10.1 SPI 总线控制器中断

SPI 支持 1 个中断。一个数据传输完成产生一个中断。

SPI_IE SPI 中断使能寄存器 (与 UART_IE IIC_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	—	—	—	—	—	—
—	SPI_IE	—	—	—	—	—	—

Bit7 保留

Bit6 **SPI_IE** : SPI 中断使能位

1 : 使能

0 : 禁止

Bit5-0 保留

SPI_IF SPI 中断标志寄存器 (与 UART_IF IIC_IF 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	—	—	—	—	—	—
—	SPI_IF	—	—	—	—	—	—

Bit7 保留

Bit6 **SPI_IF** : SPI 中断标志位

1 : 有中断请求

0 : 无中断请求

Bit5-0 保留

10.2 模拟模块中断

模拟模块包括 LED、LVD、ADC、TK，每个模块都有独立的中断使能位和中断标志位。

LED 中断

以下 2 种情况会产生 LED 中断。

- LED_EN 使能或每个 COM 周期扫描完成
- 整个显示帧扫描完成

低电压检测 LVD 中断

当满足 LVD_C0.LVD_IM 所设置的条件时，产生 LVD 中断。

模数转换器 ADC 中断

当 ADC 转换完成时，并达到 ADC_C2.ADC_CNT<1:0>所设定的转换次数时，产生 ADC 中断。

触控按键扫描器 TK 中断

以下 3 种情况会产生 TK 中断。

- TK 扫描完成
- 发生 TK 扫描溢出
- 发生 TK 扫描启动错误

AN_IE 模拟中断使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	RW-0	RW-0	—	RW-0	RW-0	RW-0
—	—	COM_IE	LED_IE	—	LVD_IE	TK_IE	ADC_IE

Bit7-6 保留

Bit5 **COM_IE** : COM 中断使能位

1 : 使能

0 : 禁止

Bit4 **LED_IE** : LED 中断使能位

1 : 使能

0 : 禁止

Bit3 保留

Bit2 **LVD_IE** : LVD 中断使能位

1 : 使能

0 : 禁止

Bit1 **TK_IE** : TK 中断使能位

1 : 使能

0 : 禁止

Bit0 **ADC_IE** : ADC 中断使能位

1 : 使能

0 : 禁止

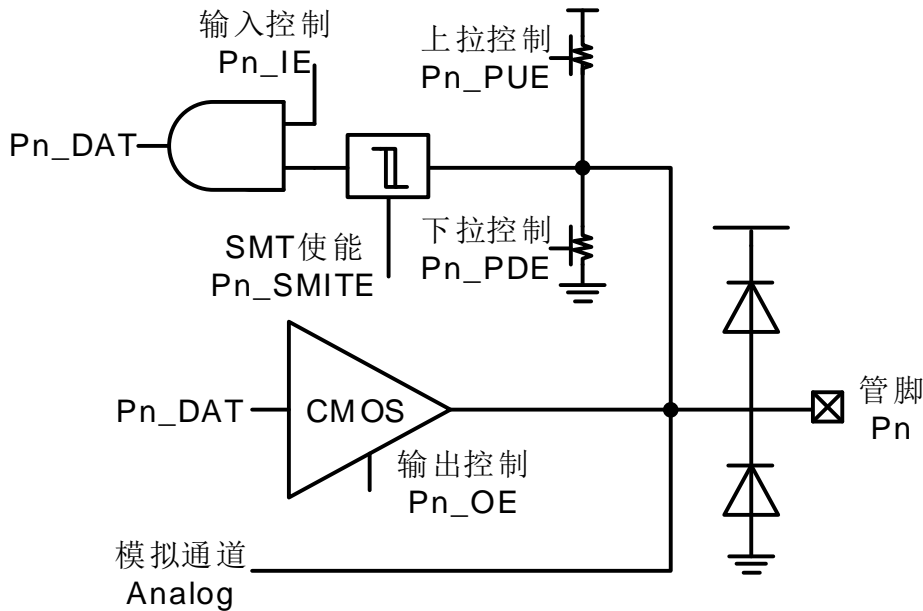
AN_IF 模拟中断标志寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	RW-0	RW-0	—	RW-0	RW-0	RW-0
—	—	COM_IF	LED_IF	—	LVD_IF	TK_IF	ADC_IF

- Bit7-6 保留
- Bit5 **COM_IF** : COM 中断标志位
 1 : 有 COM 中断请求
 0 : 无 COM 中断请求
- Bit4 **LED_IF** : LED 中断标志位
 1 : 有 LED 中断请求
 0 : 无 LED 中断请求
- Bit3 保留
- Bit2 **LVD_IF** : LVD 中断标志位
 1 : 有 LVD 中断请求
 0 : 无 LVD 中断请求
- Bit1 **TK_IF** : TK 中断标志位
 1 : 有 TK 中断请求
 0 : 无 TK 中断请求
- Bit0 **ADC_IF** : ADC 中断标志位
 1 : 有 ADC 中断请求
 0 : 无 ADC 中断请求

11 I/O 端口

11.1 I/O 功能框图



图表 14 I/O 功能框图

11.2 I/O 寄存器

PORT_C0 端口控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-00		—	—
—	—	—	—	PT_RDS<1:0>		—	—

Bit7-4 保留

Bit3-2 **PT_RDS<1:0>** : 读端口模式选择位

11 : 输出模式时读端口寄存器, 输入模式下读端口电平

10 : 始终读端口寄存器

01 : 始终读端口电平

00 : 输出模式时读端口寄存器, 输入模式下读端口电平

Bit1-0 保留

Pn_OE Pn 端口输出使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
Pn_OE<7:0>							

Bit7-0 **Pn_OE<7:0>** : Pn 端口输出使能位

1 : 输出使能

0 : 输出禁止

Pn_IE Pn 端口输入使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
Pn_IE<7:0>							

Bit7-0 **Pn_IE<7:0>** : Pn 端口输入使能位
 1 : 输入使能
 0 : 输入禁止

Pn_DAT Pn 端口数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
Pn_DAT<7:0>							

Bit7-0 **Pn_DAT<7:0>** : Pn 端口数据位
 1 : 驱动高电平
 0 : 驱动低电平

Pn_PUE Pn 端口弱上拉使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
Pn_PUE<7:0>							

Bit7-0 **Pn_PUE<7:0>** : Pn 端口弱上拉使能位
 1 : 弱上拉使能
 0 : 弱上拉关闭

Pn_PDE Pn 端口弱下拉使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
Pn_PDE<7:0>							

Bit7-0 **Pn_PDE<7:0>** : Pn 端口弱下拉使能位
 1 : 弱下拉使能
 0 : 弱下拉关闭

Pn_ODE Pn 端口开漏使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
Pn_ODE<7:0>							

Bit7-0 **Pn_ODE<7:0>** : Pn 端口输出模式选择位
 1 : 开漏输出模式 (要求开漏端口电平不高于 VDD 电平)
 0 : 推挽输出模式

Pn_SME Pn 端口施密特使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-1111_1111							
Pn_SME<7:0>							

Bit7-0 **Pn_SME<7:0>** : Pn 端口施密特使能位
 1 : 使能
 0 : 禁止

P0_IOH_CL P00~P03 高电平驱动力设置

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
P03_IOH<1:0>		P02_IOH<1:0>		P01_IOH<1:0>		P00_IOH<1:0>	

Bit7-6 **P03_IOH <1:0>** : P03 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

Bit5-4 **P02_IOH <1:0>** : P02 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

Bit3-2 **P01_IOH <1:0>** : P01 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

Bit1-0 **P00_IOH <1:0>** : P00 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

P1_IOH_CL P10~P13 高电平驱动力设置

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
P13_IOH		P12_IOH		P11_IOH		P10_IOH	

Bit7-6 **P13_IOH <1:0>** : P13 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

Bit5-4 **P12_IOH <1:0>** : P12 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

Bit3-2 **P11_IOH <1:0>** : P11 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

Bit1-0 **P10_IOH <1:0>** : P10 端口驱动电流设置

11 : 8mA 10 : 4mA

01 : 2mA 00 : 15mA

P1_IOH_CH P14~P17 高电平驱动力设置

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
P17_IOH		P16_IOH		P15_IOH		P14_IOH	

Bit7-6 **P17_IOH** <1:0> : P17 端口驱动电流设置

11 : 8mA 10 : 4mA
 01 : 2mA 00 : 15mA

Bit5-4 **P16_IOH** <1:0> : P16 端口驱动电流设置

11 : 8mA 10 : 4mA
 01 : 2mA 00 : 15mA

Bit3-2 **P15_IOH** <1:0> : P15 端口驱动电流设置

11 : 8mA 10 : 4mA
 01 : 2mA 00 : 15mA

Bit1-0 **P14_IOH** <1:0> : P14 端口驱动电流设置

11 : 8mA 10 : 4mA
 01 : 2mA 00 : 15mA

P3_IOL_CL P30~P33 低电平驱动力设置

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	—	RW-0	—	RW-0	—	RW-0
—	P33_IOL	—	P32_IOL	—	P31_IOL	—	P30_IOL

Bit7 保留

Bit6 **P33_IOL** : P33 端口驱动电流设置

0 : 15mA 1 : 80mA

Bit5 保留

Bit4 **P32_IOL** : P32 端口驱动电流设置

0 : 15mA 1 : 80mA

Bit3 保留

Bit2 **P31_IOL** : P31 端口驱动电流设置

0 : 15mA 1 : 80mA

Bit1 保留

Bit0 **P30_IOL** : P30 端口驱动电流设置

0 : 15mA 1 : 80mA

P3_IOL_CH P34~P37 低电平驱动力设置

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	—	RW-0	—	RW-0	—	RW-0
—	P37_IOL	—	P36_IOL	—	P35_IOL	—	P34_IOL

Bit7 保留

Bit6 **P37_IOL** : P37 端口驱动电流设置
0 : 15mA 1 : 80mA

Bit5 保留

Bit4 **P36_IOL** : P36 端口驱动电流设置
0 : 15mA 1 : 80mA

Bit3 保留

Bit2 **P35_IOL** : P35 端口驱动电流设置
0 : 15mA 1 : 80mA

Bit1 保留

Bit0 **P34_IOL** : P34 端口驱动电流设置
0 : 15mA 1 : 80mA

11.1 I/O 功能复用功能寄存器

符号	类型	描述	备注
Pn0~Pn7	IO	8 位双向 IO 端口	支持上下拉电阻，支持中断唤醒
PWMn0、PWMn1	O	PWM 输出端口	支持同相或互补输出
TON(TO0~TO3)	O	TIMER 比较输出	比较输出
TINn(TIN0~TIN3)	I	TIMER 外部时钟输入	
CAPn(CAP0~CAP2)	I	捕捉模式下为捕捉输入 PWM 模式下为 PWM 暂停输入	
CAPnA	I	捕捉模式下为捕捉输入 PWM 模式下为 PWM STOP 输入	P04 和 P27 两管脚支持 PWM 全暂停
TXn	O	UART 发送端口	
RXn	I	UART 接收端口	
SCL	I	IIC 时钟线	
SDA	IO	IIC 数据线	
SS	IO	SPI 片选信号	
SCK	IO	SPI 时钟信号	
MOSI	IO	SPI 主输出从输入	
MISO	IO	SPI 主输入从输出	
COM1~8	O	LED COM 驱动端口	
SEG1~12	O	LED SEG 驱动端口	
EINT0n	I	外部中断输入端口	支持睡眠唤醒
EINT1n	I	外部中断输入端口	支持睡眠唤醒
SWE	IO	单线仿真烧录端口	
RSTN	I	外部复位输入端口	输入低电平复位
AN0~AN10	A	ADC 模拟检测端口	
ADVP	A	ADC 外部正端参考电压输入	
LVDI	A	LVD 模拟检测端口	检测 0.5V 电压
XOSCI	A	外部振荡器输入	
XOSCO	A	外部振荡器输出	

Pn_FUN0 Pn 端口复用选择寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				RW-0000			
Pn1_FUN<3:0>				Pn0_FUN<3:0>			

0000~1111 : 分别代表 FUN0~FUN15

Pn_FUN1 Pn 端口复用选择寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				RW-0000			
Pn3_FUN<3:0>				Pn2_FUN<3:0>			

0000~1111 : 分别代表 FUN0~FUN15

Pn_FUN2 Pn 端口复用选择寄存器 2

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				RW-0000			
Pn5_FUN<3:0>				Pn4_FUN<3:0>			

0000~1111 : 分别代表 FUN0~FUN15

Pn_FUN3 Pn 端口复用选择寄存器 3

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				RW-0000			
Pn7_FUN<3:0>				Pn6_FUN<3:0>			

0000~1111 : 分别代表 FUN0~FUN15

参见 2.1 和 2.2 管脚复用表

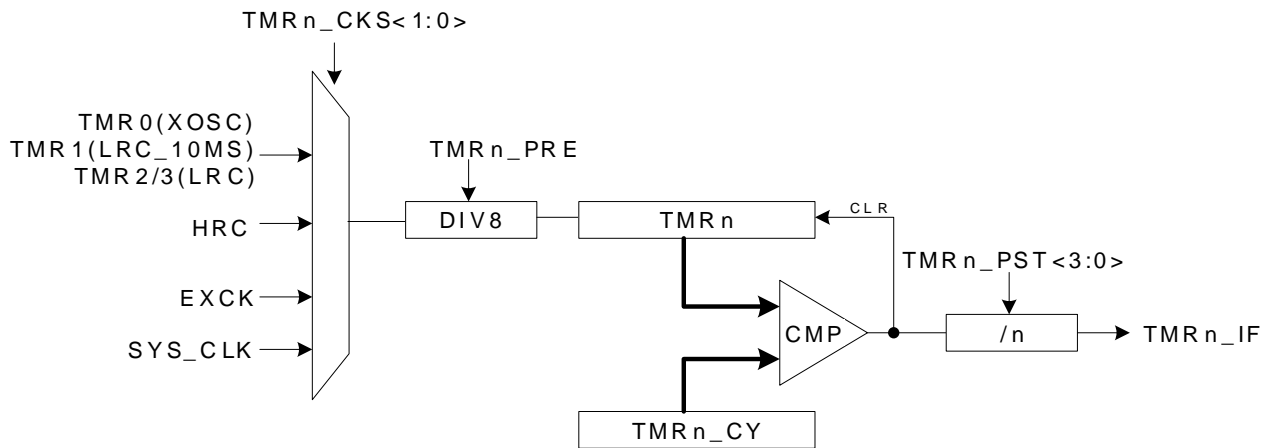
12 定时/计数器 TMR

12.1 描述

芯片内置 4 组 16 位定时/计数器 TMR0~TMR3

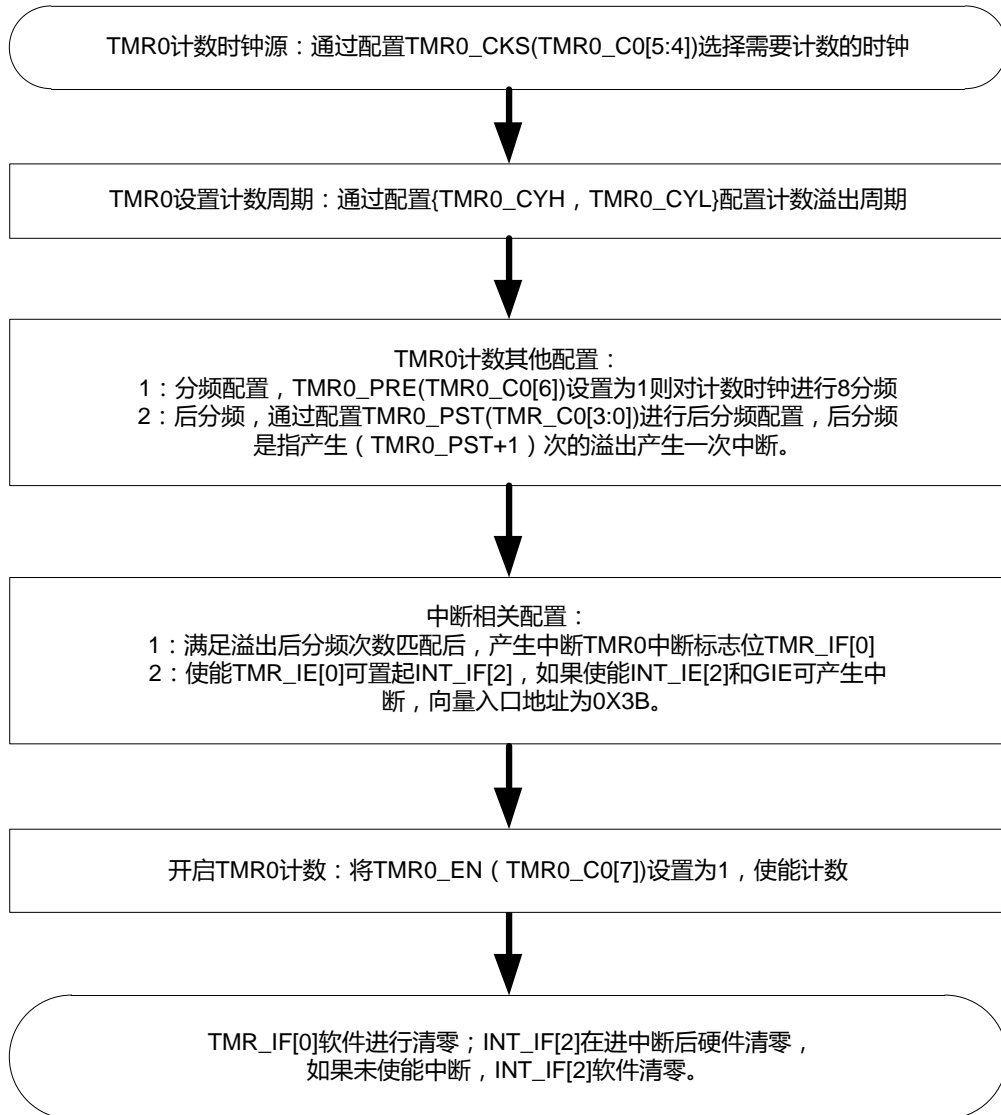
计数源可以有 4 种选择

16 位 TMRn 使能后进行累加计数，当计数器 TMRn 值与周期寄存器 TMRn_CY 值相等时，产生 1 次计数溢出，TMRn 被自动清零后继续累加计数。



图表 15 TMR 功能框图

TMR 操作流程 (以 TMR0 为例)



图表 16 TMR 操作流程

12.2 TMR 寄存器

TMRn_C0 TMRn 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-00		RW-0000			
TMRn_EN	TMRn_PRE	TMRn_CKS<1:0>		TMRn_PST<3:0>			

Bit7 **TMRn_EN** : TMRn 使能位

1 : 使能

0 : 关闭

Bit6 **TMRn_PRE** : TMRn 计数时钟 8 分频使能位

1 : 使能

0 : 关闭

Bit5-4 **TMRn_CKS<3:0>** : TMRn 计数时钟选择位

11 : TMR0 为 XOSC , TMR1 为 LRC 10ms 脉冲计数 , TMR2 , TMR3 为 LRC 计数

10 : HRC

01 : EXCK 复用端口输入

00 : SYS_CLK 系统时钟源

Bit3-0 **TMRn_PST<3:0>** : TMRn 后分频位

TMRn 的 n 次溢出产生中断 (n = TMRn_POS<3:0> + 1)

TMRnL TMRn 计数寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TMRn<7:0>							

Bit7-0 **TMRn<7:0>** : TMRn 计数值低 8 位 , 写时应先低位后高位 ; 读出时先高位后低位

TMRnH TMRn 计数寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TMRn<15:8>							

Bit7-0 **TMRn<15:8>** : TMRn 计数值高 8 位 , 写时应先低位后高位 ; 读出时先高位后低位

TMRn_CYL TMRn 周期寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TMRn_CY<7:0>							

Bit7-0 **TMRn_CY<7:0>** : TMRn 周期值低 8 位

TMRn_CYH TMRn 周期寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TMRn_CY<15:8>							

Bit7-0 **TMRn_CY<15:8>** : TMRn 周期值高 8 位

TMRn_IE TMR 中断使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
------	------	------	------	------	------	------	------

—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TMR3_IE	TMR2_IE	TMR1_IE	TMR0_IE

Bit7-4 保留

Bit3-0 **TMRn_IE**<3:0> : TMRn 中断使能位

1 : 使能

0 : 禁止

TMRn_IF **TMR 中断标志寄存器 (支持位操作)**

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TMR3_IF	TMR2_IF	TMR1_IF	TMR0_IF

Bit7-4 保留

Bit3-0 **TMRn_IF** : TMRn 中断标志位

1 : 有 TMRn 中断请求

0 : 无 TMRn 中断请求

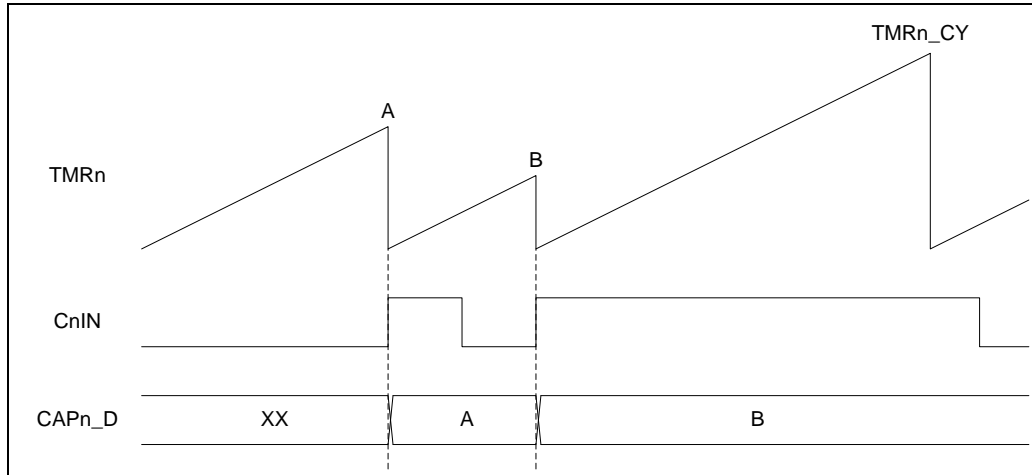
13 边沿捕捉器 CAP

芯片内置 3 路边沿捕捉器 CAP0~CAP2。

边沿捕捉器 CAP 为 TMR 的附加功能，工作时需选定一个 TMR 作为其工作时基。

捕捉清零模式

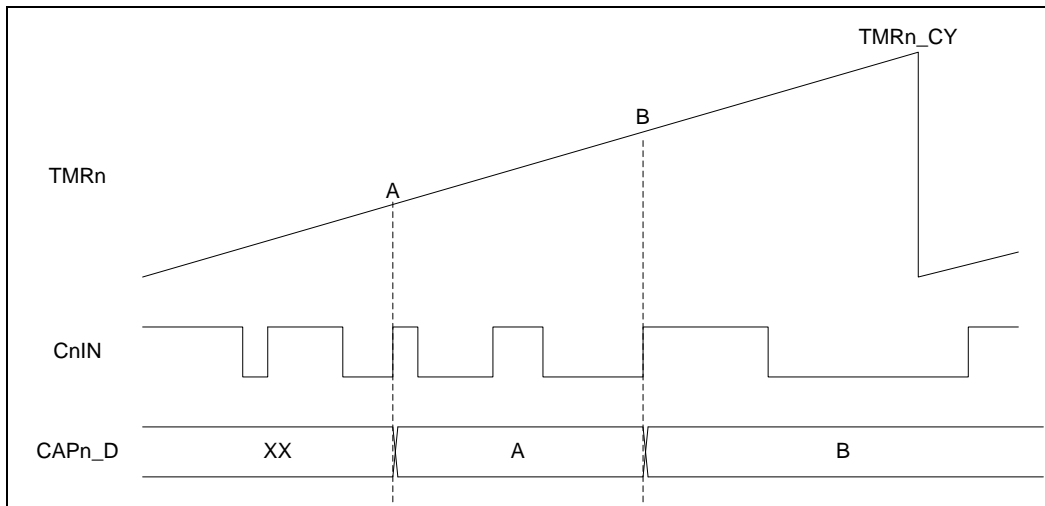
示例： $CAPn_MOD<1:0>=11$ ， $CAPn_EGS<1:0>=01$ ， $CAPn_CNT<1:0>=00$



图表 17 捕捉清零模式示例波形图

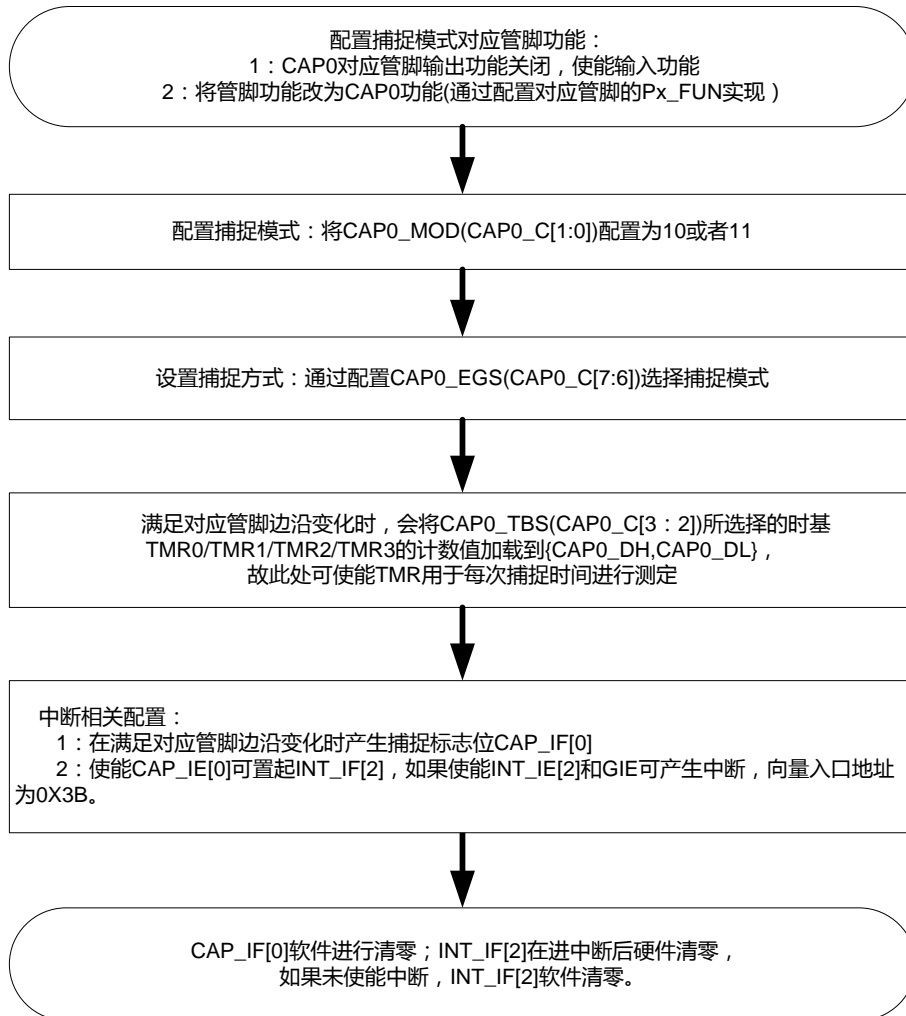
捕捉累加模式

示例： $CAPn_MOD<1:0>=10$ ， $CAPn_EGS<1:0>=11$ ， $CAPn_CNT<1:0>=01$



图表 18 捕捉累加模式示例波形图

CAP 操作流程 (以 CAP0 为例)



图表 19 CAP 操作流程图

13.1 CAP 寄存器

CAPn_C CAPn 捕捉控制寄存器 (与 PWMn_C 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
CAPn_EGS<1:0>		CAPn_CNT<1:0>		CAPn_TBS<1:0>		CAPn_MOD<1:0>	

- Bit7-6 **CAPn_EGS<1:0>** : CAPn 捕捉边沿选择位
 10 : 下降沿捕捉 11 : 上升和下降沿都捕捉
 00 : 保留 01 : 上升沿捕捉
- Bit5-4 **CAPn_CNT<1:0>** : CAPn 捕捉边沿个数选择位
 11 : 16 个触发边沿产生 1 次捕捉事件
 10 : 8 个触发边沿产生 1 次捕捉事件
 01 : 4 个触发边沿产生 1 次捕捉事件
 00 : 1 个触发边沿产生 1 次捕捉事件
- Bit3-2 **CAPn_TBS<1:0>** : CAPn 工作时基选择位
 10 : 选择 TMR2 11 : 选择 TMR3
 00 : 选择 TMR0 01 : 选择 TMR1
- Bit1-0 **CAPn_MOD<1:0>** : CAPn 捕捉模式选择位
 11 : 捕捉清零模式 (捕捉事件发生时, 将时基 TMR 清零)
 10 : 捕捉累加模式 (捕捉事件发生时, 时基 TMR 继续累加)
 01 : 保留 (用于 PWM 模式)
 00 : 关闭

CAPn_DL CAPn 捕捉寄存器低 8 位 (与 PWMn_R0L 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
CAPn_D<7:0>							

- Bit7-0 **CAPn_D<7:0>** : CAPn 捕捉值低 8 位

CAPn_DH CAPn 捕捉寄存器高 8 位 (与 PWMn_R0H 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
CAPn_D<15:8>							

- Bit7-0 **CAPn_D<15:8>** : CAPn 捕捉值高 8 位

CAPn_IE CAP 中断使能寄存器 (与 PWMn_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	RW-0	RW-0	RW-0
—	—	—	—	—	CAP2_IE	CAP1_IE	CAPO_IE

Bit7-3 保留

Bit2-0 **CAPn_IE** : CAPn 中断使能位

1 : 使能

0 : 禁止

CAPn_IF CAP 中断标志寄存器 (与 PWMn_IF 寄存器复用, 支持位操作)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	RW-0	RW-0	RW-0
—	—	—	—	—	CAP2_IF	CAP1_IF	CAPO_IF

Bit7-3 保留

Bit2-0 **CAPn_IF** : CAPn 中断使能位

1 : 有 CAPn 中断请求

0 : 无 CAPn 中断请求

14 脉宽调制器 PWM

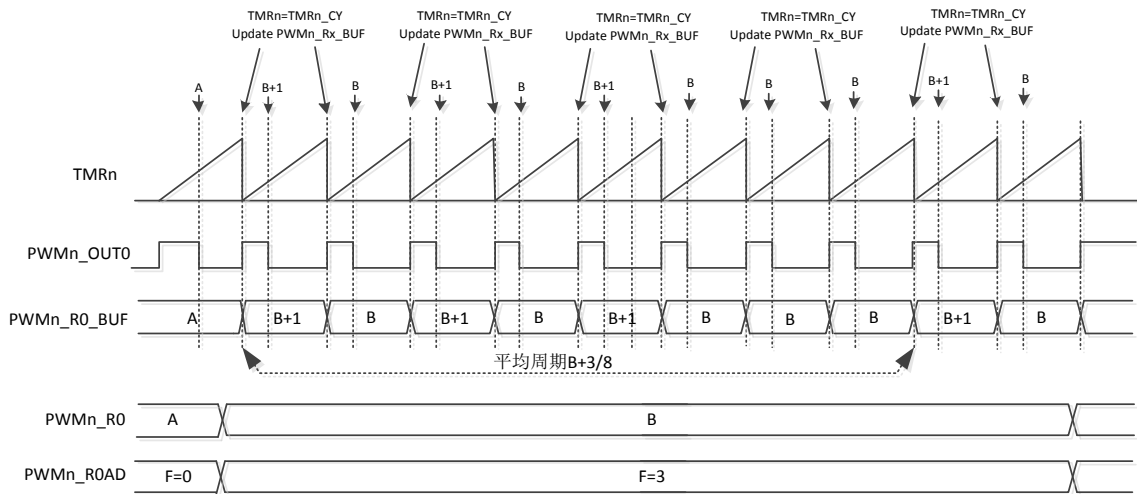
14.1 描述

芯片内置 3 路 16+3 位脉宽调制器 PWM0~PWM2。PWM 支持边沿对齐和中心对齐。

脉宽调制器 PWM 为 TMR 的附加功能，工作时需选定一个 TMR 作为工作时基。在选择时基时，根据应用系统的需求，可多组 PWM 选择同一个 TMR 时基，也可分别选择不同的 TMR 时基。当选择同一 TMR 时基时，采用同一时基工作的多组 PWM 的输出频率是相同的。

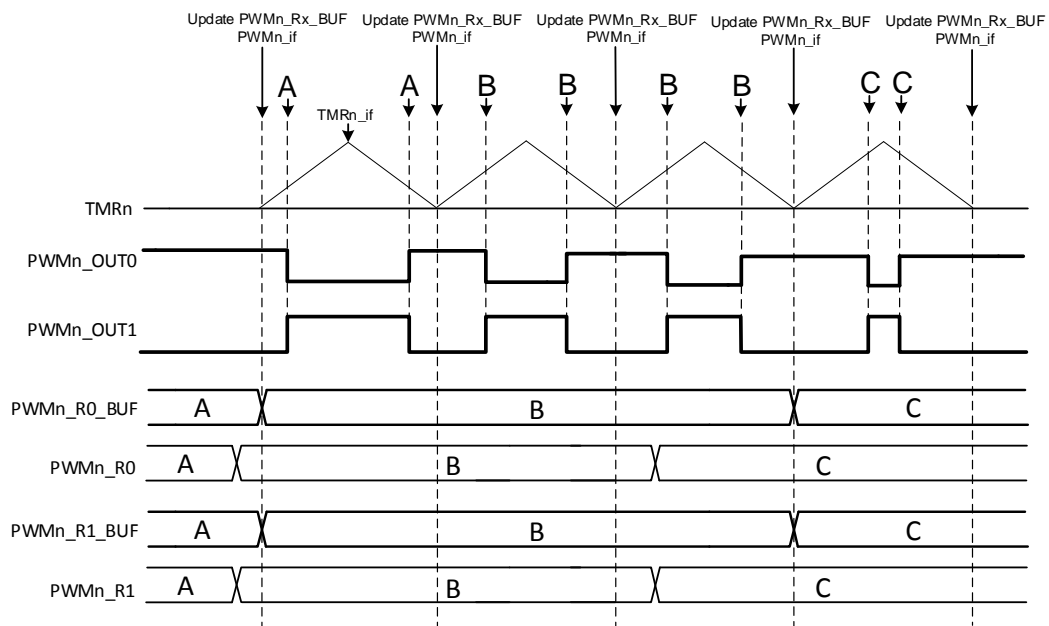
示例 1：边沿对齐，无死区，扩展设置为 3，设置如下：

PWMn_DT<7:0>=00H，PWMn0_T=0，PWMn0_P=0，PWMn_R0EX=3



图表 20 PWM 边沿对齐工作示例波形图

示例 2：中心对齐，设置如下：



图表 21 PWM 中心对齐工作示例波形图

PWM 周期

PWM 以所选定 TMR 作为时基进行工作，PWM 的周期即为所选定 TMR 的 TMRn_CY 寄存器所设定的计数周期。

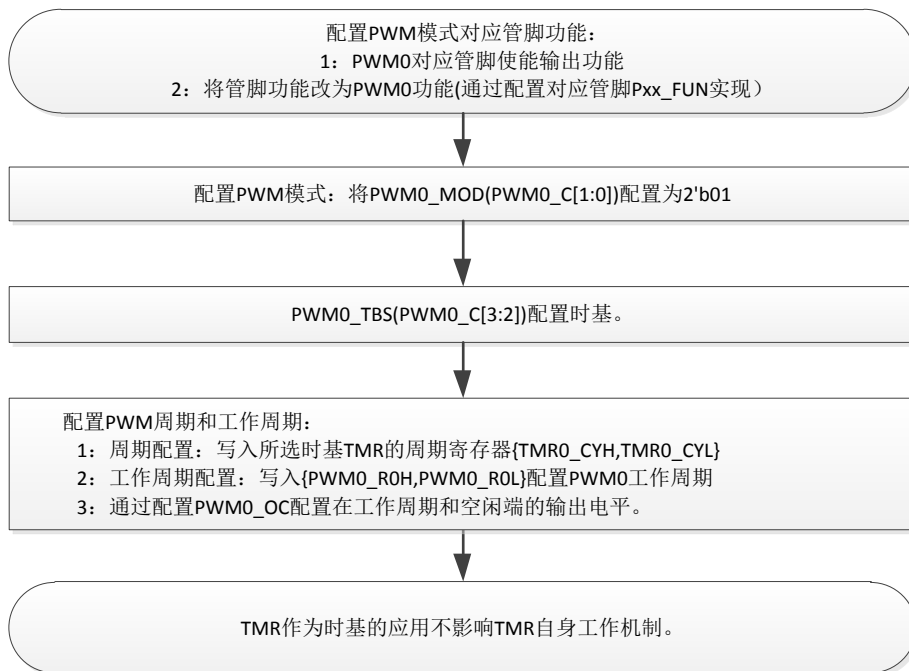
PWM 匹配点

PWM 模块包含 1 组 16+3 位的匹配寄存器 PWMn_R0，且匹配寄存器各有 1 级缓冲器 PWMn_R0_BUF 对应于 PWMn 的 1 路输出通道。当 PWM 关闭时，写匹配寄存器会同时将写入值更新到匹配缓冲器中；当 PWM 运行时，写匹配寄存器不会立即更新匹配缓冲器，而是在一个完整的 PWM 周期结束时才将匹配寄存器的内容更新到匹配缓冲器中。PWM 工作时，时基的计数值会实时和匹配缓冲器进行比较，当 PWM 时基计数值与匹配缓冲器值相等时，即为匹配点。用户可分别设定 PWM 周期内匹配点前和匹配点后的 PWM 输出电平。

PWM 输出初始态

当 PWM 模块不使能 PWM 波形输出初始态，用户可设定初始态的输出电平。当然，这个初始态的输出电平是否通过 PWM 复用端口输出，要取决于相应复用端口的设置。

PWM 操作流程（以 PWM0 为例）



图表 22 PWM 操作流程图

14.2 PWM 寄存器

PWMn_C PWMn 控制寄存器 (与 CAPn_C 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
PWMn_SPS<1:0>		PWM_MD<1:0>		PWMn_TBS<1:0>		PWMn_MOD<1:0>	

Bit7-6 **PWMn_SPS<1:0>** : PWMn 刹车模式选择位 (刹车后 PWM 模式关闭, 需要软件启动)

11 : 软件刹车 (写 “11” 立即进入刹车状态)

10 : 比较器输出低刹车

01 : STP 复用端口输入低电平刹车

00 : 关闭刹车功能

Bit5-4 **PWM_MD<1:0>**

11 : 保留

10 : 保留

01 : 中心对齐模式

00 : 边沿对齐模式

Bit3-2 **PWMn_TBS<1:0>** : PWMn 工作时基选择位

11 : 选择 TMR3

10 : 选择 TMR2

01 : 选择 TMR1

00 : 选择 TMR0

Bit1-0 **PWMn_MOD<1:0>** : PWMn 模式选择位

1x : 保留

01 : PWM 模式

00 : 关闭

PWMn_R0L PWMn 匹配点寄存器低 8 位 (与 CAPn_DL 寄存器复用) 写入先高后低

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
PWMn_R0<7:0>							

Bit7-0 **PWMn_R0<7:0>** : PWMn0 占空比低 8 位

PWMn_R0H PWMn 匹配点寄存器高 8 位 (与 CAPn_DH 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
PWMn_R0<15:8>							

Bit7-0 **PWMn_R0<15:8>** : PWMn0 占空比高 8 位

PWMn_R1L PWMn 匹配点寄存器低 8 位, 写入先高后低

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
PWMn_R1<7:0>							

Bit7-0 **PWMn_R1<7:0>** : PWMn1 占空比低 8 位

PWMn_R1H PWMn 匹配点寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
PWMn_R1<15:8>							

Bit3-0 **PWMn_R1<15:8>** : PWMn1 占空比高 8 位

PWMn_OC PWMn 输出控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-000			RW-0	RW-0	RW-0	RW-1
—	PWMn_REX<2:0>			PWMn1_T	PWMn0_T	PWMn1_P	PWMn0_P

Bit7 保留

Bit6-4 **PWMn_REX** : PWMn 平均占空比扩展位

Bit3 **PWMn1_T** : PWMn 通道 1 输出初始态选择位

1 : 输出高电平

0 : 输出低电平

Bit2 **PWMn0_T** : PWMn 通道 0 输出初始态选择位

1 : 输出高电平

0 : 输出低电平

Bit1 **PWMn1_P** : PWMn 通道 1 输出配置

1 : 反向输出 (匹配点前输出值 0 , 匹配点后输出 1)

0 : 正向输出 (匹配点前输出值 1 , 匹配点后输出 0)

Bit0 **PWMn0_P** : PWMn 通道 0 输出配置

1 : 反向输出 (匹配点前输出值 0 , 匹配点后输出 1)

0 : 正向输出 (匹配点前输出值 1 , 匹配点后输出 0)

PWMn_DT PWMn 输出控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
PWMn_DT<7:0>							

Bit7-0 **PWMn_D<7:0>** : 死区延时时间

PWMn_IE PWM 中断使能寄存器 (与 CAPn_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	RW-0	RW-0	—	RW-0	RW-0	RW-0
—	STP2_IE	STP1_IE	STP0_IE	—	PWM2_IE	PWM1_IE	PWM0_IE

Bit7 保留

Bit6-4 **STPn_IE** : PWMn 刹车中断使能位
1 : 使能
0 : 禁止

Bit3 保留

Bit2-0 **PWMn_IE** : PWMn 周期中断使能位
1 : 使能
0 : 禁止

PWMn_IF PWM 中断标志寄存器 (与 CAPn_IF 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	RW-0	RW-0	—	RW-0	RW-0	RW-0
—	STP2_IF	STP1_IF	STP0_IF	—	PWM2_IF	PWM1_IF	PWM0_IF

Bit7 保留

Bit6-4 **STPn_IF** : PWMn 刹车中断标志位
1 : 有 PWMn 刹车中断请求
0 : 无 PWMn 刹车中断请求

Bit3 保留

Bit2-0 **PWMn_IF** : PWMn 周期中断标志位
1 : 有 PWMn 周期中断请求
0 : 无 PWMn 周期中断请求

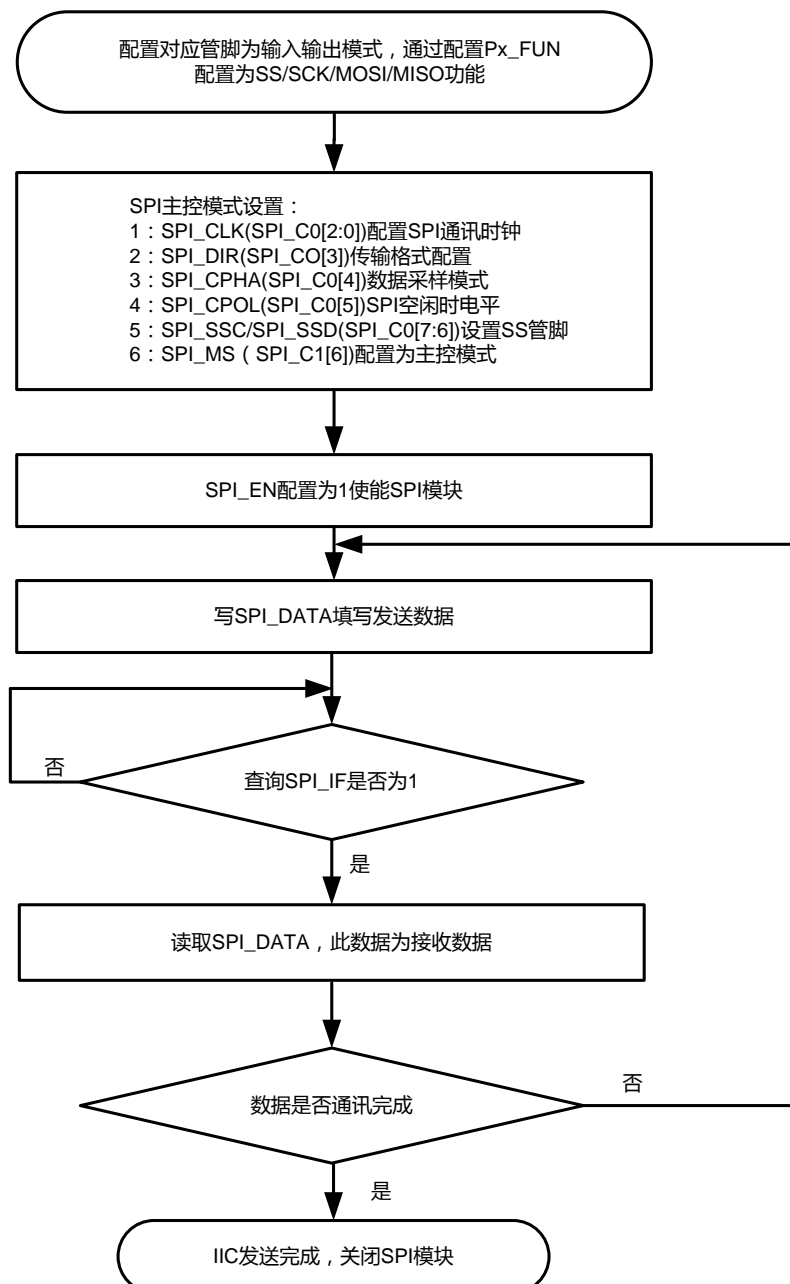
15 SPI 通讯端口

15.1 描述

芯片具备 1 组 SPI 通讯端口。

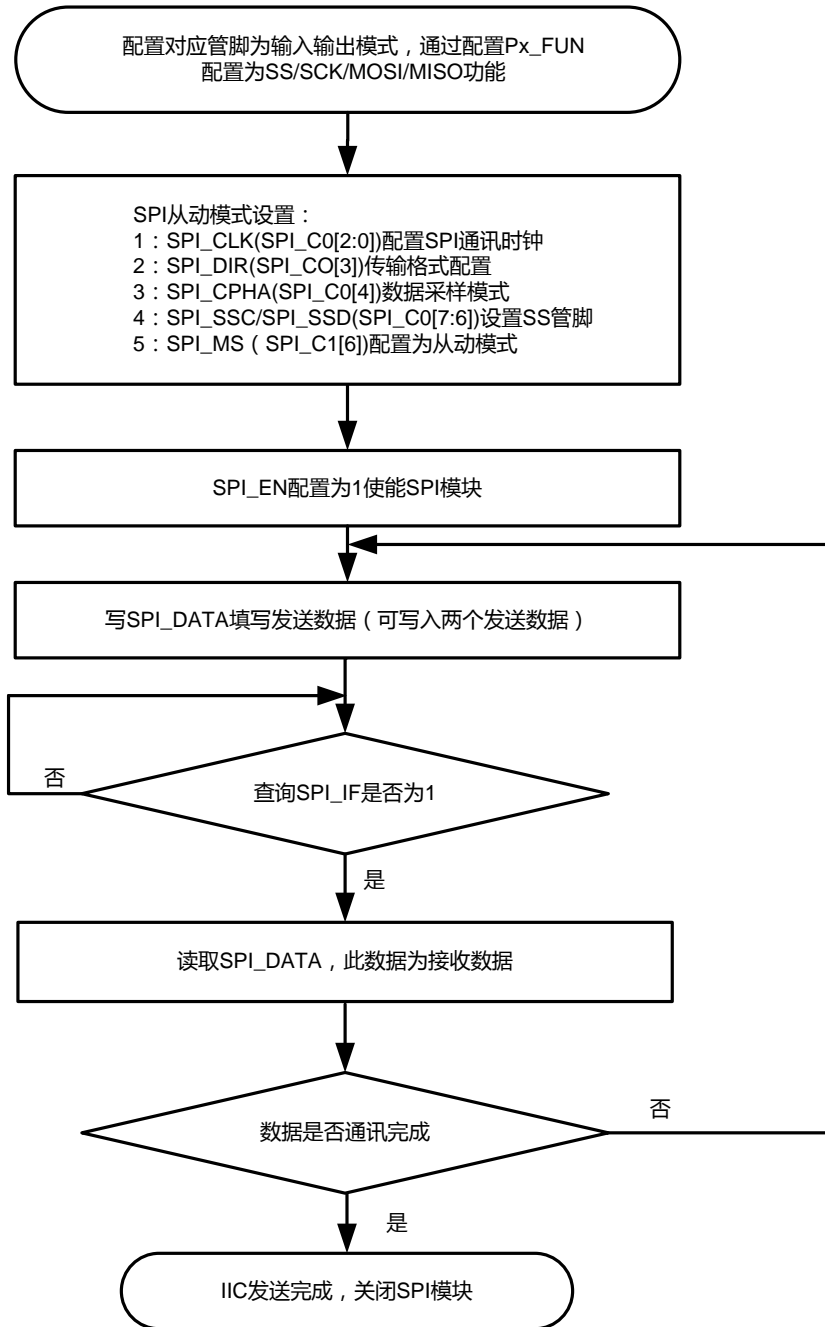
- 支持全双工工作模式
- 支持主从模式
- 极性可配置
- 支持传输格式高低位可配置

SPI 主控通讯流程



图表 23 SPI 主控模式流程图

SPI 从动通讯流程



图表 24 SPI 从动模式流程图

15.2 SPI 寄存器

SPI_C0 SPI 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-0	RW-0	RW-0	RW-000		
SPI_SSC	SPI_SSD	SPI_CPOL	SPI_CPHA	SPI_DIR	SPI_CLK		

- Bit7 **SPI_SSC** : SPI 的 SS 使能控制位
1 : 片选信号软件控制,受 SPI_SSD 控制
0 : SS 管脚有效
- Bit6 **SPI_SSD** : SPI 的 SS 软件控制位
1 : 关闭片选
0 : 片选使能
- Bit5 **SPI_CPOL** : SPI 管脚空闲状态
1 : 空闲状态是 SCK 保持 1
0 : 空闲状态是 SCK 保持 0
- Bit4 **SPI_CPHA** : SPI 数据采样格式
1 : SCK 周期的第二个沿采集数据
0 : SCK 周期的第一个沿采集数据
- Bit3 **SPI_DIR** : SPI 传输格式
1 : LSB 先发送
0 : MSB 先发送
- Bit2-0 **SPI_CLK** : SPI_CLK 分频器
111 : SYSCLK/2 110 : SYSCLK/128 101 : SYSCLK/64 100 : SYSCLK/32
011 : SYSCLK/16 010 : SYSCLK/8 001 : SYSCLK/4 000 : SYSCLK/2

SPI_C1 SPI 控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	—	—	R-0	R-0	RW-0	RW-0
SPI_EN	SPI_MS	—	—	SPI_TXSP	SPI_RXSP	SPI_WOL	SPI_ROV

- Bit7 **SPI_EN** : SPI 使能控制位
1 : 使能
0 : 禁止
- Bit6 **SPI_MS** : SPI 主从模式
1 : 从机模式
0 : 主控模式
- Bit5-4 保留
- Bit3 **SPI_TXSP** : 发送缓存区空满标志
1 : 发送缓存区满
0 : 发送缓存区空
- Bit2 **SPI_RXSP** : 接收缓存区空满标志
1 : 接收缓存区满

- 0 : 接收缓存区空
- Bit1 **SPI_WOL** : SPI 写冲突标志位
1 : 写冲突, 硬件置 1
0 : 无写冲突, 软件清 0
- Bit0 **SPI_RXSP** : SPI 接收冲突标志位
1 : 有接收溢出, 硬件置 1
0 : 无接收溢出, 软件清 0

SPI_DATA SPI 数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
SPI_DATA<7:0>							

- Bit7-0 **SPI_DATA<7:0>** : SPI 发送/接收数据
R : 读取 SPI_DATA 时将获得接收移位换寄存器的数据
W : 写入 SPI_DATA 的数据被放置到发送移位寄存器中

SPI_n_IE SPI_n 中断使能寄存器 (与 UART_n_IE IIC_n_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	—	—	—	—	—	—
—	SPI0_IE	—	—	—	—	—	—

- Bit7 保留
- Bit6 **SPI0_IE** : SPI0 中断使能位
1 : 使能
0 : 禁止
- Bit5-0 保留

SPI_n_IF SPI_n 中断标志寄存器 (与 UART_n_IF IIC_n_IF 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	—	—	—	—	—	—
—	SPI0_IF	—	—	—	—	—	—

- Bit7 保留
- Bit6 **SPI0_IF** : SPI0 中断标志位
1 : 有中断请求
0 : 无中断请求
- Bit5-0 保留

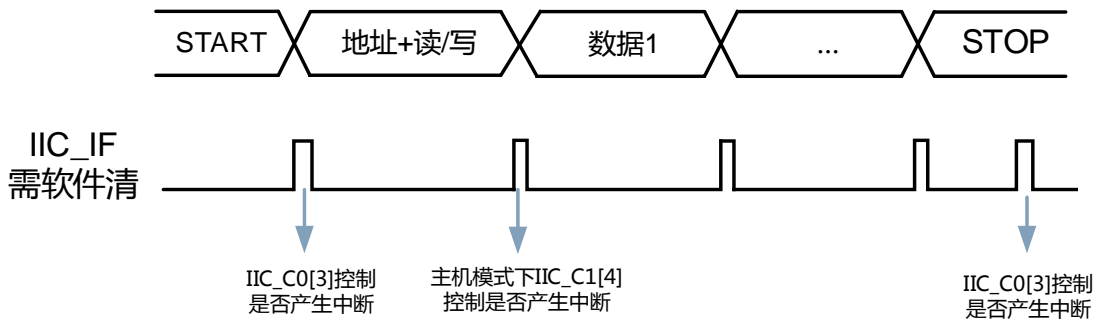
16 IIC 总线控制器

16.1 描述

芯片内置 1 路 IIC 总线控制器，3 组端口分时复用，功能特性如下：

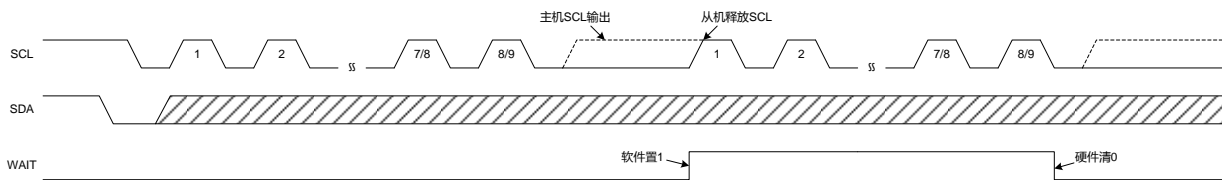
- 支持主控模式（仅支持单主机，不支持多主仲裁）
- 支持从动模式（仅支持 7 位从机地址，从机地址可设置）
- 支持通信等待功能
- 支持通信超时检测

IIC 通讯中断机制



图表 25 IIC 通讯中断机制图

IIC 通信等待



图表 26 IIC 通信等待波形示意图

IIC 总线控制器支持通信等待功能，通过 **IIC_C0.WTEN** 位使能。

在主机模式中使能通信等待功能，当 IIC 总线上每完成一个字节的发送或接收后，主机将时钟线 SCL 释放为高电平，并实时检测 SCL 线上的电平状态。如果 SCL 为高电平，则继续进行后续时钟发送；如果 SCL 被从机拉为低电平，说明从机没有作好通信准备，强制进入通信等待状态，则主机一直等到从机重新将 SCL 线释放为高电平后，才继续进行后续时钟发送。

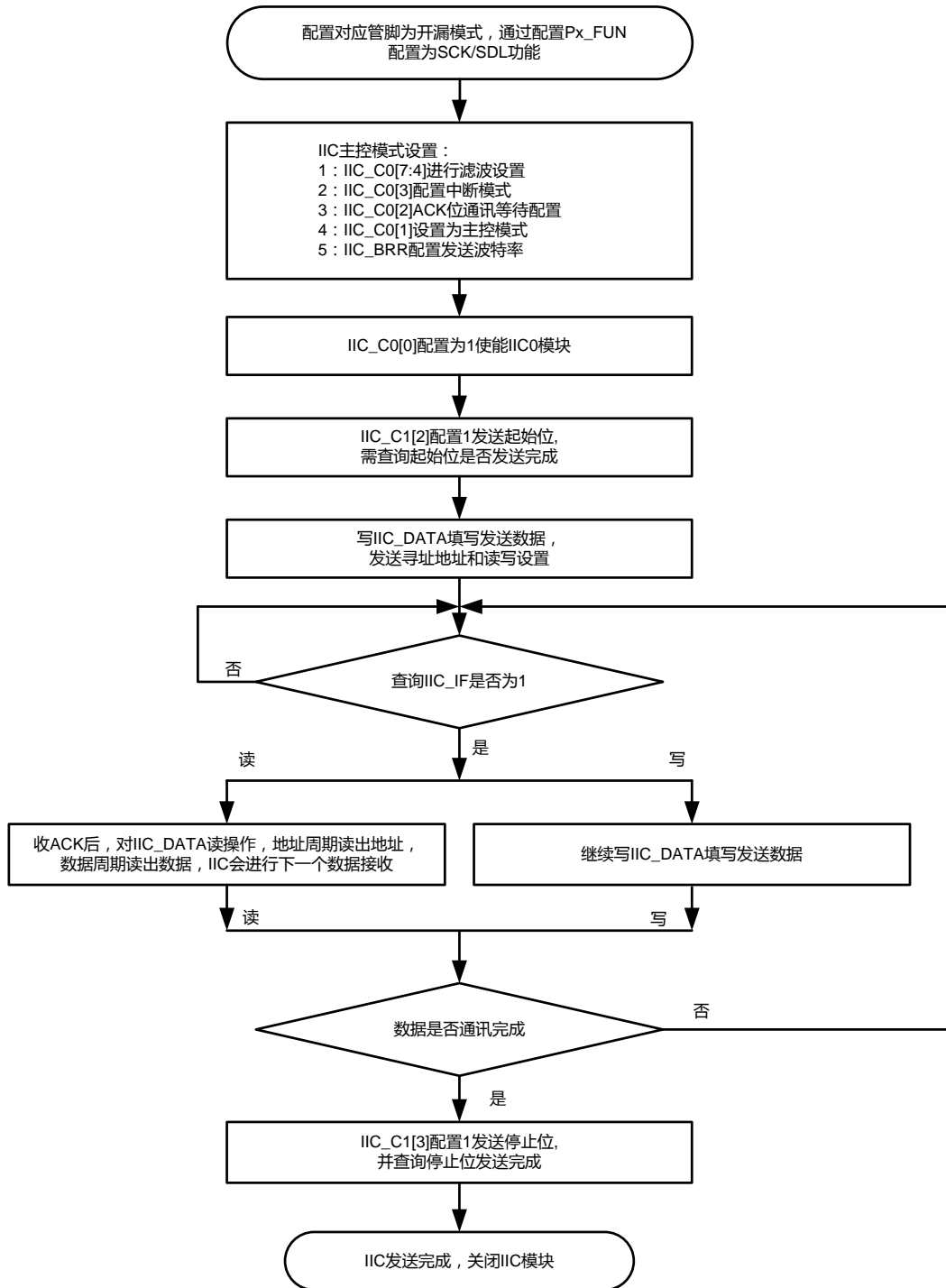
在从动模式中使能通信等待功能，当 IIC 总线上每完成一个字节的发送或接收后，从机自动将时钟线 SCL 拉为低电平，强制进入通信等待状态。待从机作好通信准备后，通过将 **IIC_C1.WAIT** 位置 1 将 SCL 线释放为高电平，主机检测到这个高电平后会继续进行后续传输。

IIC 通信超时检测

IIC 通信中，由于存在通信等待等主、从机之间的握手机制，因此有可能在异常情况下导致通信死锁。使能通信超时检测功能，可以通过超时中断发现通信中的异常死锁，以便进行软件处理。

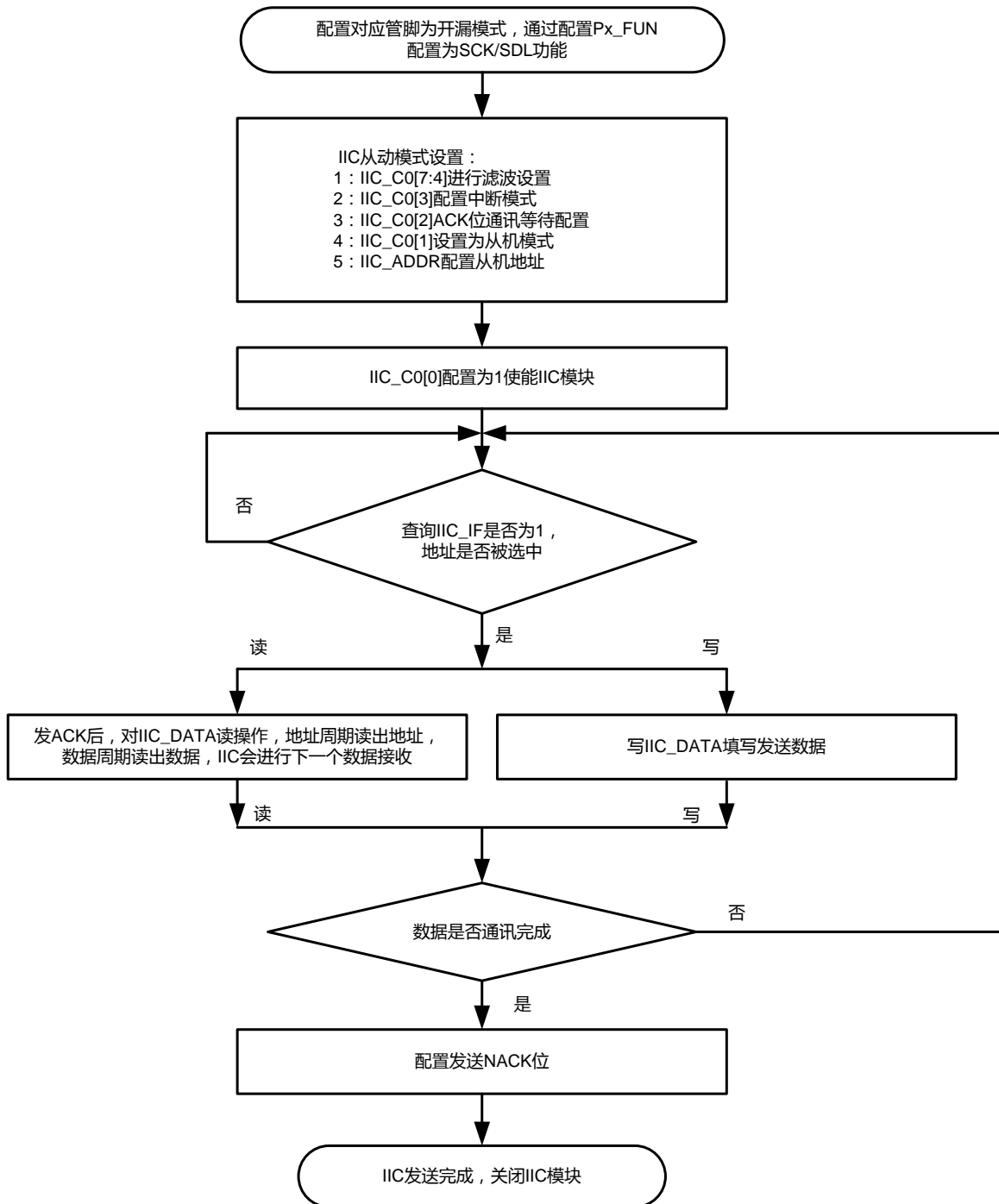
超时检测功能检测 IIC 总线上 SCL 线低电平的持续时间，如果 SCL 线低电平的持续时间超过 **IIC_TOC.IIC_OVT** 所设定的超时时间，则会产生 IIC 超时中断。

IIC 主控通讯流程



图表 27 IIC 主控通讯流程图

IIC 从机通讯流程



图表 28 IIC 从机通讯流程图

16.2 IIC 寄存器

IIC_C0 IIC 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				RW-0	RW-0	RW-0	RW-0
IIC_SMPF<3:0>				IIC_IM	WTEN	IIC_MD	IIC_EN

Bit7-4 **IIC_SMPF<3:0>** : SCL 和 SDA 采样滤波时间 (要求远小于波特率设定的时钟周期)

$$\text{滤波时间 } T_{\text{FLT}} = \frac{\text{IIC_SMPF<3:0>} + 1}{F_{\text{SYSCLK}}}$$

Bit3 **IIC_IM** : IIC 位中断模式选择位

1 : START 和 STOP 位不产生中断, 完成 1 个字节的接收或发送产生中断

0 : START 和 STOP 位产生中断, 完成 1 个字节的接收或发送产生中断

Bit2 **WTEN** : IIC 等待功能使能位

1 : 使能

0 : 关闭

Bit1 **IIC_MD** : IIC 主/从模式选择位

1 : 从动模式

0 : 主控模式

Bit0 **IIC_EN** : IIC 使能位

1 : 使能

0 : 关闭

IIC_C1 IIC 控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	RW-0	RW-0	RW-0	RW-0
—	—	—	MTAI_MK	STOP	START	WAIT	ACK

Bit7-5 保留

Bit4 **MTAI_MK** : 地址传输中断屏蔽位

1 : 地址传输产生中断

0 : 地址传输不产生中断

Bit3 **STOP** : STOP 位发送位

1 : 置 1 发送 STOP 位 (检测到 **STOP_F(IIC_STA<5>**位后需软件清零)

0 : STOP 位发送完成

Bit2 **START** : START 位发送位

1 : 置 1 发送 START 位 (发送完成后自动清零)

0 : START 位发送完成

Bit1 **WAIT** : 从机等待释放位 (仅在 **IIC_MD=1** 并且 **WTEN=1** 时有意义)

1 : 置 1 释放从机等待状态 (完成下一字节传送时, 硬件自动清零)

0 : 从机等待状态

Bit0 **ACK** : ACK 位发送选择位

1 : 发送 NACK

0 : 发送 ACK

IIC_STA IIC 状态寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R-0	R-0	R-0	R-0	R-0	R-0	RW-0	RW-0
SLV_ADF	SLV_RWF	STOP_F	START_F	ACK_F	BUF_ST	WERR	OERR

- Bit7 SLV_ADF** : 从机已接收字节类型标志位
 1 : 当前接收完成的字节为数据
 0 : 当前接收完成的字节为地址
- Bit6 SLV_RWF** : 从机读写标志位
 1 : 主机读从机
 0 : 主机写从机
- Bit5 STOP_F** : STOP 位检测标志位
 1 : 检测到 IIC 总线上有 STOP 位 (清 IIC_IF.IIC_IF 位时同步被清 0)
 0 : 未检测到 IIC 总线上有 STOP 位
- Bit4 START_F** : START 位检测标志位
 1 : 检测到 IIC 总线上有 START 位 (清 IIC_IF.IIC_IF 位时同步被清 0)
 0 : 未检测到 IIC 总线上有 START 位
- Bit3 ACK_F** : ACK 位检测标志位
 1 : 检测到 NACK
 0 : 检测到 ACK
- Bit2 BUF_ST** : 缓冲器状态位
 接收模式
 1 : 接收缓冲器满 (读 IIC_DATA 寄存器时同步被清 0)
 0 : 接收缓冲器未满
 发送模式
 1 : 发送缓冲器空 (写 IIC_DATA 寄存器时同步被清 0)
 0 : 发送缓冲器未空
- Bit1 OVT_F** : IIC 超时标志位
 1 : 超时 (清 IIC_IF.IIC_IF 位时同步清 0 , 也可软件清 0)
 0 : 未超时
- Bit0 BFOV_F** : 缓冲器溢出标志位
 接收模式
 1 : 接收缓冲器溢出 (软件清 0)
 0 : 接收缓冲器未溢出
 发送模式
 1 : 发送写缓冲器溢出 (软件清 0)
 0 : 发送写缓冲器未溢出

IIC_BRR IIC 波特率寄存器 (用于主控模式, 从动模式下复用为 IIC_ADDR 寄存器)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-000_0000						
—	IIC_BRR<6:0>						

Bit7 保留

Bit6-0 **IIC_BRR<6:0>** : IIC 波特率

$$\text{波特率 IIC_BR} = \frac{F_{\text{SYSCLK}}}{(\text{IIC_BRR}<6:0> + 1) \times 4}$$

IIC_ADDR IIC 从地址寄存器 (用于从动模式, 主控模式下复用为 IIC_BRR 寄存器)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-000_0000						
—	IIC_ADDR<6:0>						

Bit7 保留

Bit6-0 **IIC_ADDR<6:0>** : IIC 从机地址

IIC_DATA IIC 数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
IIC_DATA<7:0>							

Bit7-0 **IIC_DATA<7:0>** : IIC 发送/接收数据

IIC_TOC IIC 超时控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	—	RW-00_0000					
TOC_EN	—	IIC_OVT<5:0>					

Bit7 **TOC_EN** : IIC 超时检测使能位

1 : 使能

0 : 关闭

Bit6 保留

Bit5-0 **IIC_OVT<5:0>** : IIC 超时时间

$$\text{超时时间 } T_{\text{OV}} = \frac{(\text{IIC_OVT}<5:0> + 1) \times 64}{F_{\text{SYSCLK}}}$$

IICn_IE IICn 中断使能寄存器 (与 UARTn_IE SPIn_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	—	—	—	—
—	—	—	IIC0_IE	—	—	—	—

Bit7-5 保留

Bit4 **IIC0_IE** : IIC0 中断使能位
1 : 使能
0 : 禁止

Bit3-0 保留

IICn_IF IICn 中断标志寄存器 (与 UARTn_IF SPIn_IF 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	—	—	—	—
—	—	—	IIC0_IF	—	—	—	—

Bit7-5 保留

Bit4 **IIC0_IF** : IIC0 中断标志位
1 : 有中断请求
0 : 无中断请求

Bit3-0 保留

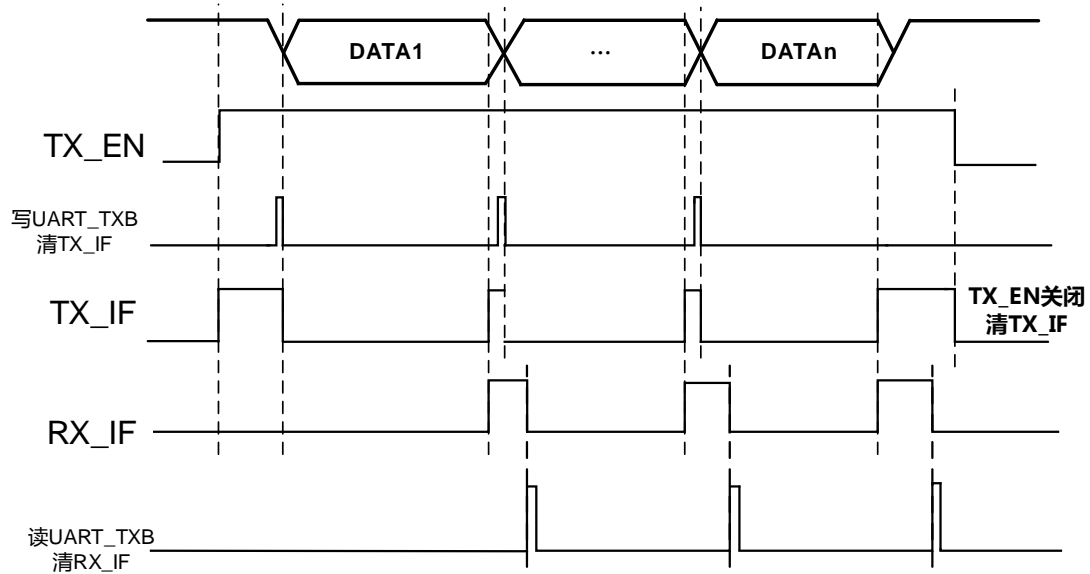
17 UART 异步收发器

17.1 描述

芯片内置 2 路 UART 异步收发器，功能特性如下：

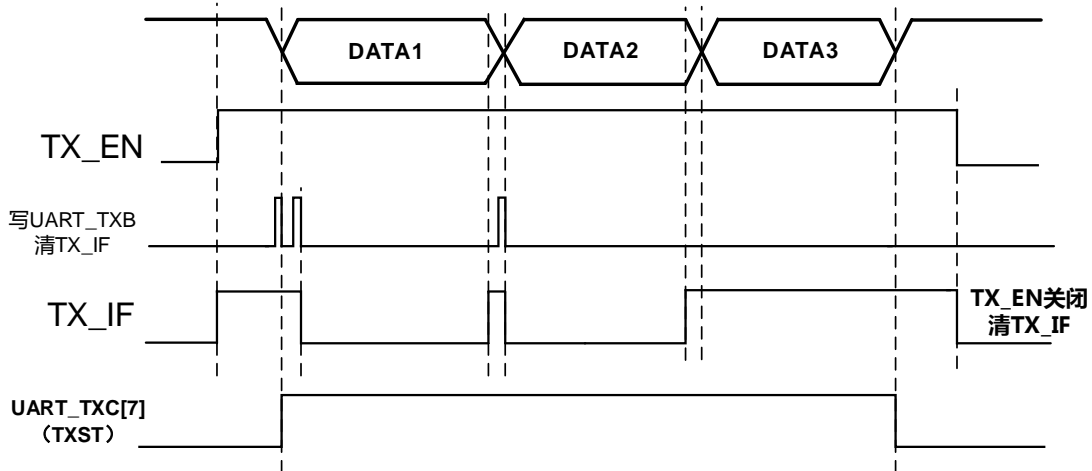
- 支持 8/9 数据传输，支持 1/2STOP 位
- 内部使用 16 次采样，特殊设计，波特率特殊计算方式等同于支持小数分频
- 波特率=系统时钟/ (UART_BRR+1)
- 支持通信溢出错误检测，支持奇偶自动校验
- UART 发送标志位通过写发送寄存器清除，软件无法直接清除，UART 接收寄存器通过读取接收寄存器清除，软件无法直接清除

UART 通讯中断机制（发送使用空闲中断）



图表 29 UART 通讯中断机制 1

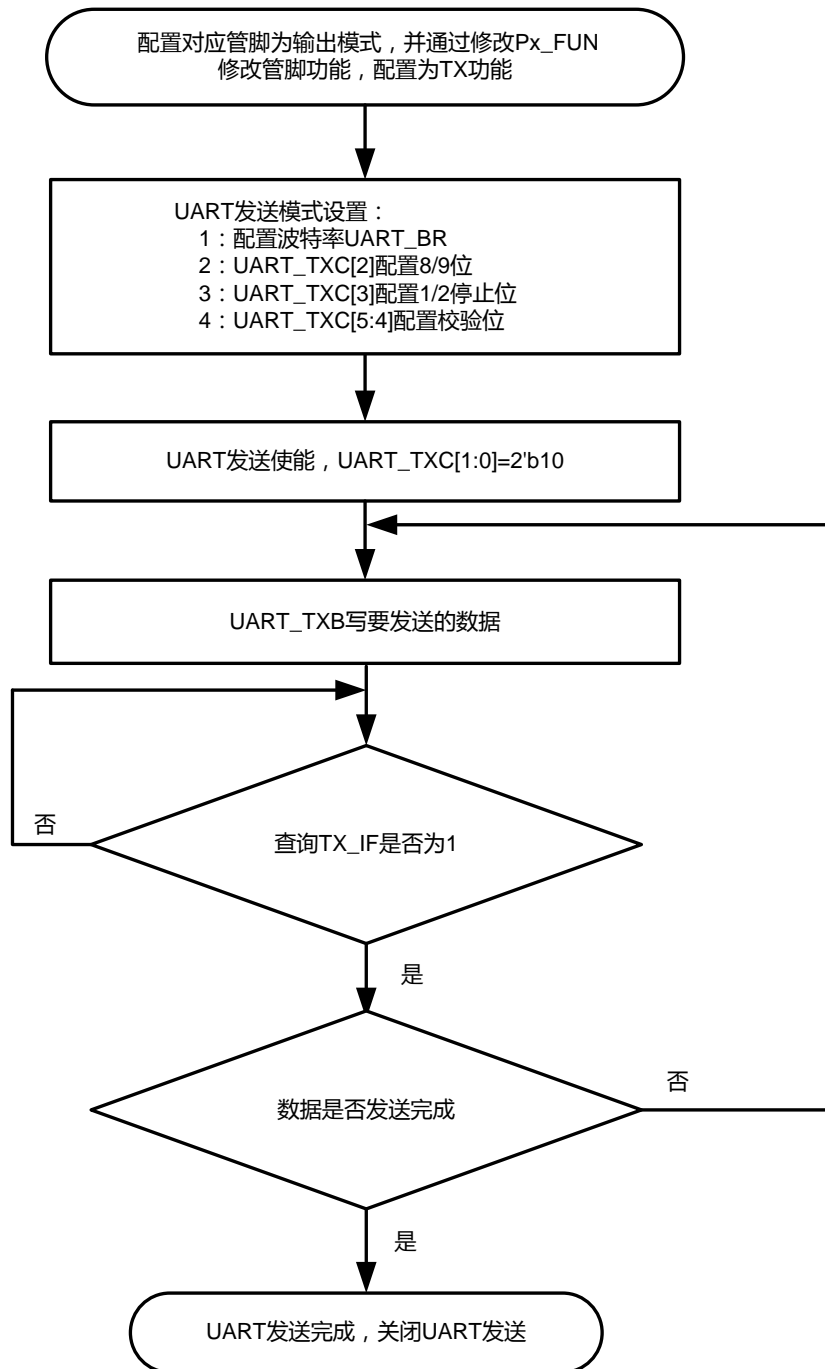
UART 通讯中断机制（发送使用寄存器空中断）



图表 30 UART 通讯中断机制 2

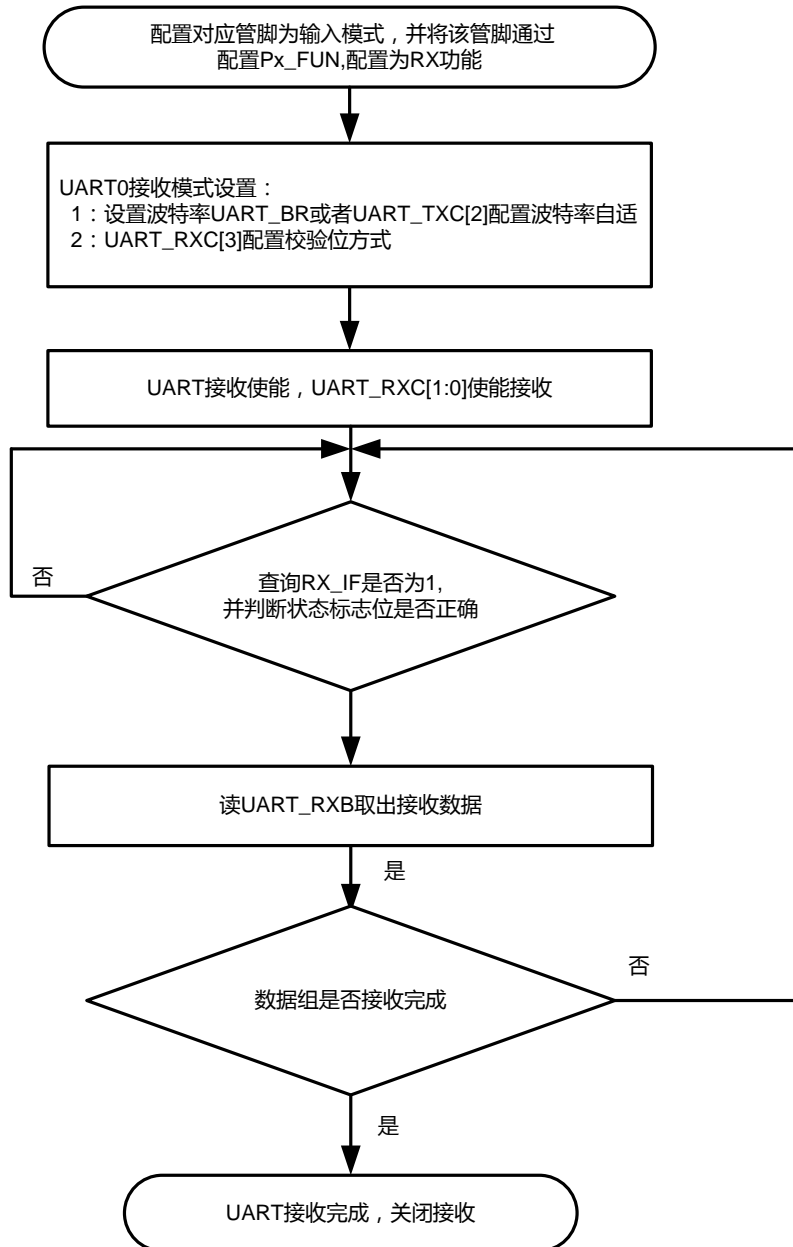
17.2 UART 操作流程

UART 发送操作流程



图表 31 UART 发送操作流程

UART 接收操作流程



图表 32 UART 接收操作流程

17.3 UART 寄存器

UARTn_BRL UARTn 波特率寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
UART_BRR<7:0>							

Bit7-0 **UART_BRR<7:0>** : UART 波特率低 8 位

波特率计算公式：波特率=系统时钟/ (UARTn_BRR+1)

UARTn_BRH UARTn 波特率寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
UART_BRR<15:8>							

Bit7-0 **UART_BRR<15:8>** : UART 波特率高 8 位

波特率计算公式：波特率=系统时钟/ (UARTn_BRR+1)

UARTn_RXC UARTn 接收控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R-0	R-0	R-0	R-0	RW-0	RW-0	RW-0	RW-0
FERR	OERR	PERR	RX9D	PARS	BRFX	RXEN<1:0>	

Bit7 **FERR** : UARTn 接收帧错误标志位

- 1 : 有帧错误
- 0 : 无帧错误

Bit6 **OERR** : UARTn 接收溢出错误标志位

- 1 : 有溢出错误
- 0 : 无溢出错误

Bit5 **PERR** : UARTn 接收校验错误标志位 (仅在 **RXEN<1:0>=11** 时有效)

- 1 : 有校验错误
- 0 : 无校验错误

Bit4 **RX9D** : UARTn 接收第 9 位数据

Bit3 **PARS** : 奇偶校验选择位

- 1 : 偶校验
- 0 : 奇校验

Bit2 **BRFX** : UARTn 波特率自适应使能位

- 1 : 使能
- 0 : 关闭

Bit1-0 **RXEN<1:0>** : UARTn 接收使能位

- 11 : 使能 9 位数据接收 (影响奇偶校验标志 **PERR**)
- 10 : 使能 9 位数据接收 (不影响奇偶校验标志 **PERR**)
- 01 : 使能 8 位数据接收 (不影响奇偶校验标志 **PERR**)
- 00 : 关闭 RX 接收功能

UARTn_RXB UARTn 接收数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
RXB<7:0>							

Bit7-0 **RXB<7:0>** : UARTn 接收数据

UARTn_TXC UARTn 发送控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
R-0	R-0	RW-00		RW-0	RW-0	RW-00	
TXST	TXBF	TX9S<1:0>		STPS	TXDM	TXEN<1:0>	

Bit7 **TXST** : UARTn 发送状态标志位

- 1 : UARTn 发送未完成
- 0 : UARTn 发送已完成

Bit6 **TXBF** : UARTn 发送缓冲区满标志位

- 1 : UARTn 发送缓冲区满
- 0 : UARTn 发送缓冲区空

Bit5-4 **TX9S<1:0>** : UARTn 发送第 9 位数据格式选择位

- 11 : 发送数据 1
- 10 : 发送数据 0
- 01 : 发送偶校验
- 00 : 发送奇校验

Bit3 **STPS** : 发送 STOP 位长度选择位

- 1 : 发送 2 位 STOP 位
- 0 : 发送 1 位 STOP 位

Bit2 **TXDM** : UARTn 发送数据格式选择位

- 1 : 9 位数据
- 0 : 8 位数据

Bit1-0 **TXEN<1:0>** : UARTn 发送使能位

- 11 : 使能, 发送空闲和发送寄存器空均产生中断
- 10 : 使能, 发送空闲产生中断
- 01 : 使能, 发送寄存器空产生中断
- 00 : 关闭

UARTn_IE UARTn 中断使能寄存器 (与 IICn_IE SPIn_IE 寄存器复用)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TX1_IE	RX1_IE	TX0_IE	RX0_IE

- Bit7-4 保留
- Bit3 **TX1_IE** : UART1 发送中断使能位
 1 : 使能
 0 : 禁止
- Bit2 **RX1_IE** : UART1 接收中断使能位
 1 : 使能
 0 : 禁止
- Bit1 **TX0_IE** : UART0 发送中断使能位
 1 : 使能
 0 : 禁止
- Bit0 **RX0_IE** : UART0 接收中断使能位
 1 : 使能
 0 : 禁止

UARTn_IF UARTn 中断标志寄存器 (与 IICn_IF SPIn_IF 寄存器复用, 支持位操作)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-0	RW-0	RW-0	RW-0
—	—	—	—	TX1_IF	RX1_IF	TX0_IF	RX0_IF

- Bit7-4 保留
- Bit1 **TX1_IF** : UART1 发送中断标志位
 1 : 有 UART1 发送中断请求
 0 : 无 UART1 发送中断请求
- Bit0 **RX1_IF** : UART1 接收中断标志位
 1 : 有 UART1 接收中断请求
 0 : 无 UART1 接收中断请求
- Bit1 **TX0_IF** : UART0 发送中断标志位
 1 : 有 UART0 发送中断请求
 0 : 无 UART0 发送中断请求
- Bit0 **RX0_IF** : UART0 接收中断标志位
 1 : 有 UART0 接收中断请求
 0 : 无 UART0 接收中断请求

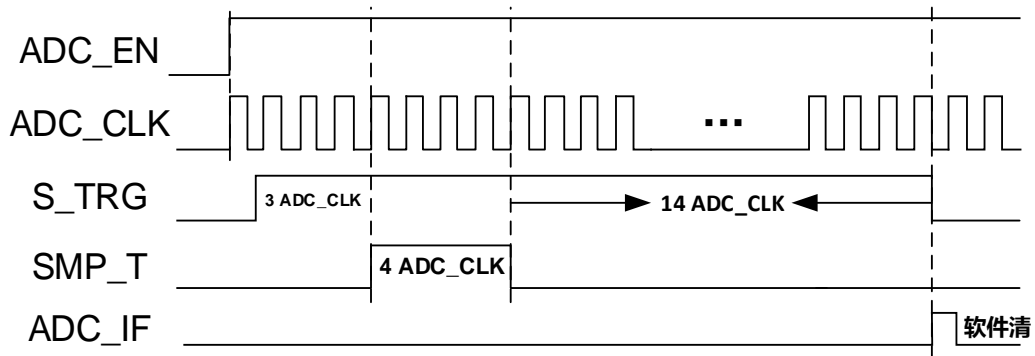
18 模数转换器 ADC

18.1 描述

芯片内置 12 位 SAR 结构 ADC，支持高速模式（最高采样率 100KHz）。

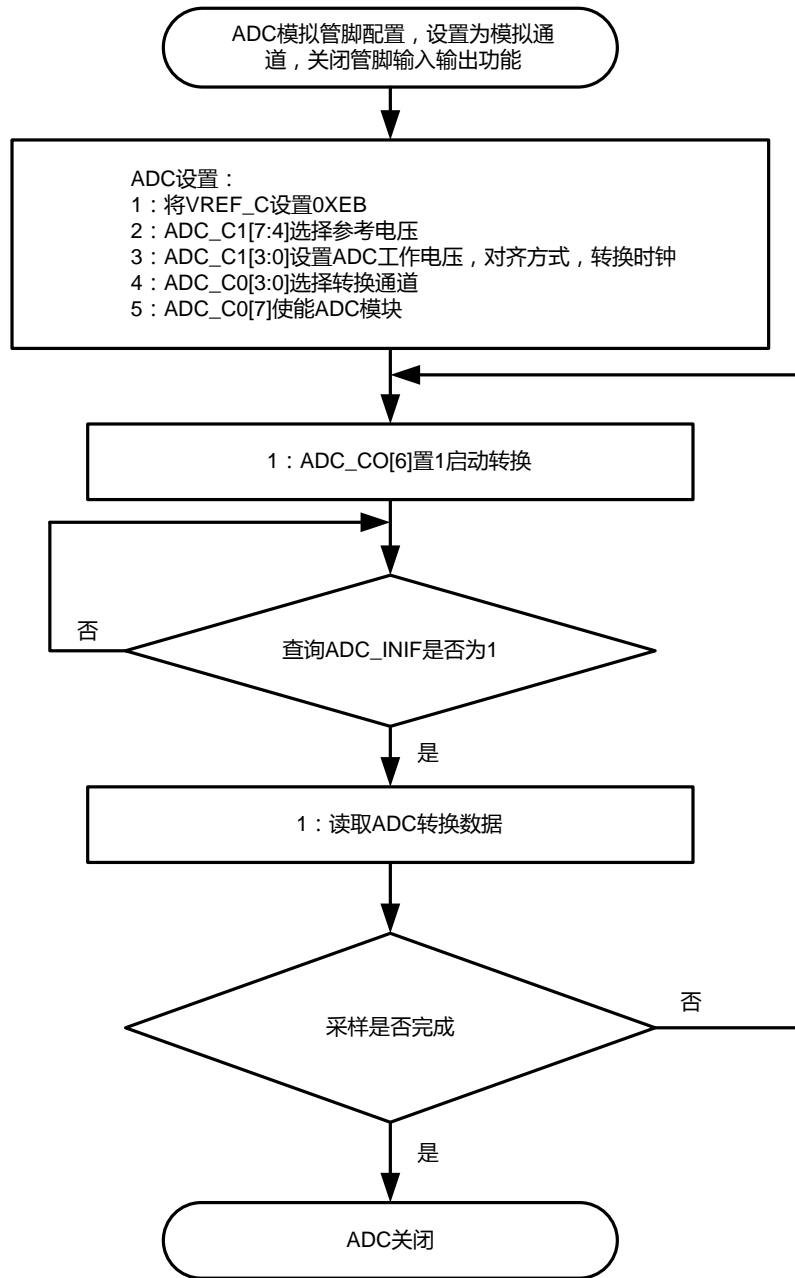
- 支持 11 个外部采样通道 AN0~AN10
- 支持内部 1/4 VDD 检测通道
- 转换时间为 17 个 ADC_CLK 加上 SMP_T(默认设置为 4 个 ADC_CLK),默认情况下转换为 21 个 ADC_CLK

ADC 转换时序图



图表 33 ADC 转换时序图

18.2 ADC 操作流程



图表 34 ADC 操作流程

18.3 ADC 寄存器

ADC_C0 ADC 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-00		RW-1111			
ADC_EN	S_TRG	TRG_S<1:0>		CH_SEL<3:0>			

- Bit7 **ADC_EN** : ADC 使能位
1 : 使能
0 : 关闭
- Bit6 **S_TRG** : ADC 软件触发位
1 : 触发 ADC 转换
0 : 转换完成 (硬件自动清 0)
- Bit5-4 **TRG_S<1:0>** : 触发方式选择位
1x : TMR2 计数值匹配 ADC_TR_R0 值后触发 ADC , 再 PWM1_R0+ADC_TR_R0 后再次触发 ADC,此处需注意是整体的周期限制不要溢出。
01 : TMR2 计数值匹配 ADC_TR_R0 值后触发 ADC
00 : 软件触发 (S_TRG 触发转换)
- Bit3-0 **CH_SEL<3:0>** : ADC 采样通道选择位
其他 : 所有通道关闭
1011 : 选择内部 1/4 VDD 通道
1010 : 选择通道 AN10
1001 : 选择通道 AN9
... ..
0001 : 选择通道 AN1
0000 : 选择通道 AN0

ADC_C1 ADC 控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-000				—	RW-0	RW-00	
VREF_S<3:0>				—	ADC_DM	ADC_CKS<1:0>	

- Bit7-4 **VREFP_S<3:0>** : **VREFP_S ADC 参考选择 (使用 VREF 需要先开启 VREF)**
1111 : 选择内部 VREF 0.2V 1110 : 选择内部 VREF 0.5V
1101 : 选择内部 VREF 1.2V 1100 : 选择内部 VREF 1.5V
1011 : 未用 1010 : 选择内部 VREF 1.024V
1001 : 选择内部 VREF 2.048V 1000 : 选择内部 VREF 2.5V
0010 : 选择内部 VDD 0001 : ADVP 复用端口输入参考
0000 : 关闭
- Bit3 保留
- Bit2 **ADC_DM** : ADC 转换数据格式选择位
1 : 左对齐 (结果放置{**ADC_DH<7:0>**, **ADC_DL<7:4>** })
0 : 右对齐 (结果放置{**ADC_DH<3:0>**, **ADC_DL<7:0>** })
- Bit1-0 **ADC_CKS<1:0>** : ADC 转换时钟 ADC_CLK 选择位
11 : 系统时钟 SYS_CLK 16 分频
10 : 系统时钟 SYS_CLK 8 分频
01 : 系统时钟 SYS_CLK 4 分频
00 : 系统时钟 SYS_CLK 2 分频

ADC_C2 ADC 控制寄存器 2

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	RW-00		—	RW-011		
—	—	ADC_CNT<1:0>		—	SMP_T<2:0>		

Bit7-6 保留 (用户固定写 11)

Bit5-4 **ADC_CNT<1:0>** : ADC 转换次数选择位

11 : 8 次转换取平均

10 : 4 次转换取平均

01 : 2 次转换取平均

00 : 1 次转换

Bit3 保留

Bit2-0 **SMP_T<2:0>** : ADC 采样时间控制位

采样时间 $T_{SAMPLE} = (SMP_T<2:0> \times 2 + 1) \times T_{ADC_CLK}$

ADC_DL ADC 结果寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
ADC_DL<7:0>							

Bit7-0 **ADC_DL<7:0>** : ADC 结果寄存器低 8 位

ADC_DH ADC 结果寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
ADC_DH<7:0>							

Bit7-0 **ADC_DH<7:0>** : ADC 结果寄存器高 8 位

ADC_TG_R0L TMR2 触发 ADC 延时值低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
ADC_TG_R0L<7:0>							

Bit7-0 **ADC_TG_R0L<7:0>** : TMR2 触发 ADC 延时值低位

ADC_TG_R0H TMR2 触发 ADC 延时值高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
ADC_TG_R0H<7:0>							

Bit7-0 **ADC_TG_R0H<7:0>** : TMR2 触发 ADC 延时值高位

ADC_CMP_C ADC 比较控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-00	
ADC_CP_E	MAX_E	MIN_E	INT_MD	MAX_F	MIN_F	ADC_DMD	

- Bit7 **ADC_CP_E** : ADC 比较功能使能位
 1 : 使能
 0 : 关闭
- Bit6 **MAX_E** : 上限比较使能位
 1 : 使能
 0 : 关闭
- Bit5 **MIN_E** : 下限比较使能位
 1 : 使能
 0 : 关闭
- Bit4 **INT_MD** : ADC 中断模式 (比较模式使能后有效)
 1 : 转换完成和比较超出上下限产生中断
 0 : 仅比较超出上下限产生中断
- Bit3 **MAX_F** : 上限比较使能位
 1 : 小于等于上限阈值
 0 : 大于上限阈值
- Bit2 **MIN_F** : 下限比较使能位
 1 : 大于等于下限阈值
 0 : 小于下限阈值
- Bit1-0 **ADC_DMD**<1:0> : ADC_DH 和 ADC_DL 操作对象控制
 00 : 仅可读出 ADC 转换值, 对齐方式受 AD_DM 控制
 01 : 对 ADC_DH 和 ADC_DL 写操作作为配置 ADC 上限阈值, 低对齐
 10 : 对 ADC_DH 和 ADC_DL 写操作作为配置 ADC 下限阈值, 低对齐
 11 : 仅可读出 ADC 转换值, 对齐方式受 AD_DM 控制

AN_IE 模拟中断使能寄存器 (与 TK LVD LED 共用地址)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	—	RW-0
—	—	—	—	—	—	—	ADC_IE

- Bit7-1 保留
- Bit0 **ADC_IE** : ADC 中断使能位
 1 : 使能
 0 : 禁止

AN_IF 模拟中断标志寄存器 (与 TK LVD LED 共用地址, 支持位操作)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	—	RW-0
—	—	—	—	—	—	—	ADC_IF

- Bit7-1 保留
- Bit0 **ADC_IF** : ADC 中断标志位
 1 : 有 ADC 中断请求
 0 : 无 ADC 中断请求

19 触控按键控制器 TK

19.1 描述

芯片内置电容式触控按键检测模块，可将连接于 TKn 复用引脚的触控按键电容值转换为 TK_CNT 的计数值，用户可通过调整外部基准电容 TK_CX 的容值大小（一般建议 10~33nf），以及设置 TK_C1.TK_VRFS<2:0>位调整 TK_CX 充电到的参考电压，以适用于各种不同的应用环境。芯片最多支持 16 路触控按键检测通道 TK1~TK24，需注意的，使能 TK 模块必须使能 VREF 模块

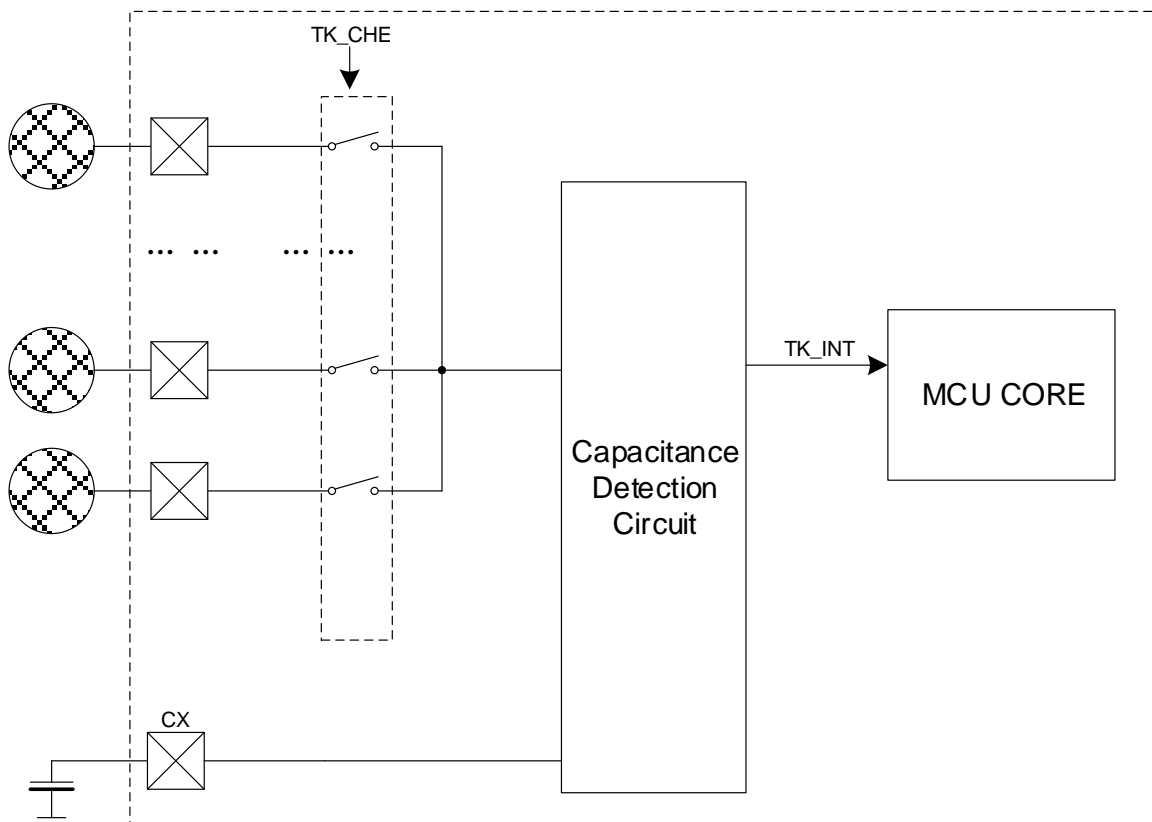
TK 扫描溢出

如果扫描放大倍数设置不合适，或者扫描过程中有异常干扰，都可能导致扫描过程中 TK_CNT 计数溢出。溢出时将 TK 扫描溢出标志位 TK_C0.TK_OV 置 1，同时触发 TK 中断。

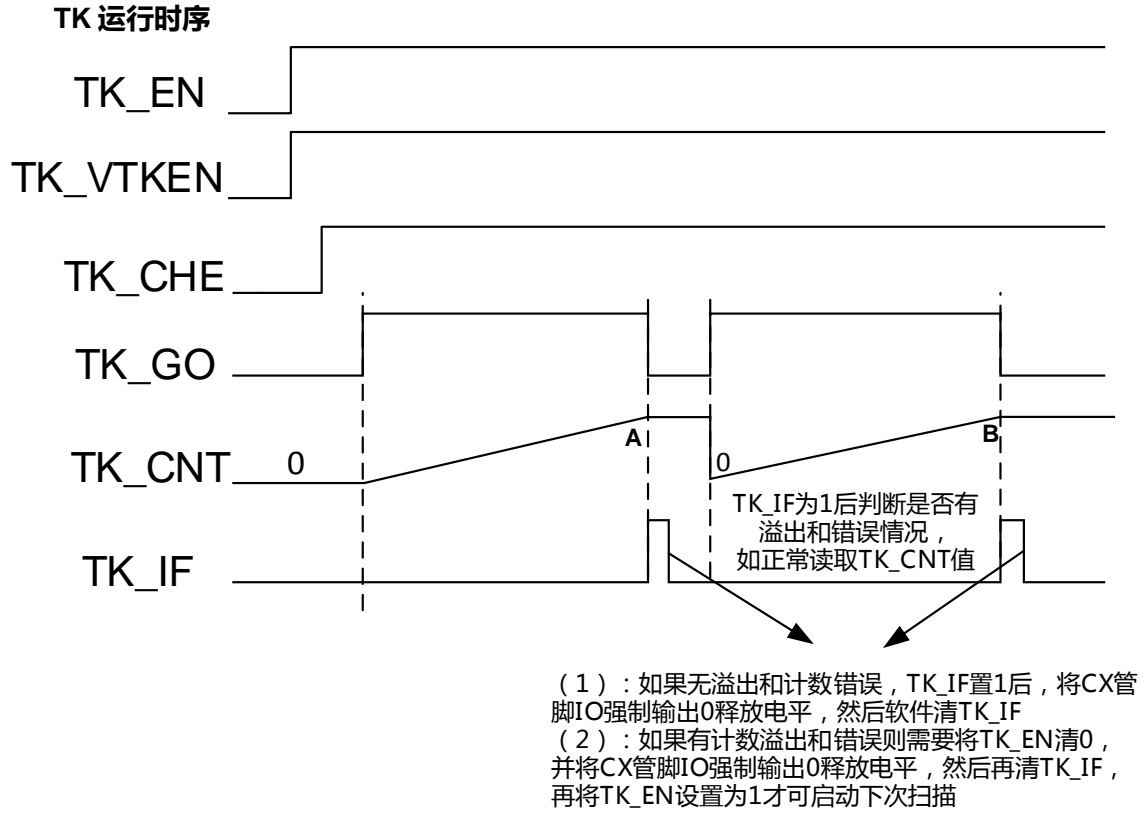
TK 扫描启动错误

启动扫描前，如果未对 CX 电容进行初始化，会产生启动错误。启动错误发生时将 TK 扫描启动错误标志位 TK_C0.TK_ERR 置 1，同时触发 TK 中断。

19.2 TK 功能框图

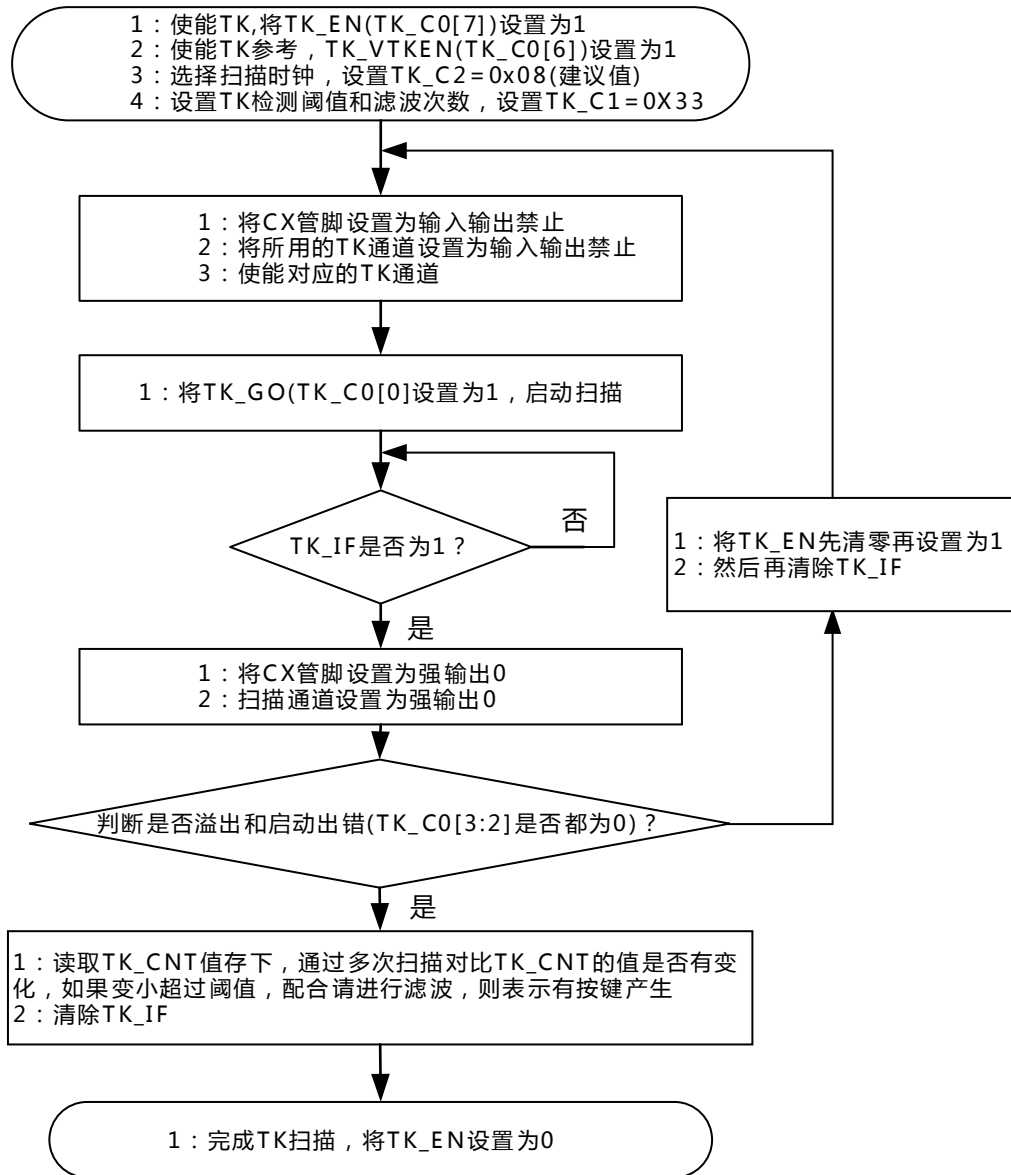


图表 35 TK 功能框图



图表 36 TK 运行时序图

19.1 TK 操作流程



图表 37 TK 运行流程图

TK_CHEL TK 扫描通道使能寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TK_CHE<7:0>							

Bit7-0 **TK_CHE<7:0>** : TK 扫描通道使能位 (对应 TK8~TK1)

1 : 使能 TKn 扫描通道

0 : 关闭 TKn 扫描通道

TK_CHEM TK 扫描通道使能寄存器中 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TK_CHE<15:8>							

Bit7-0 **TK_CHE <15:8>** : TK 扫描通道使能位 (对应 TK16~TK9)

1 : 使能 TKn 扫描通道

0 : 关闭 TKn 扫描通道

TK_CHEM TK 扫描通道使能寄存器中 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TK_CHE<23:16>							

Bit7-0 **TK_CHE <23:16>** : TK 扫描通道使能位 (对应 TK24~TK17)

1 : 使能 TKn 扫描通道

0 : 关闭 TKn 扫描通道

TK_CNTL TK 扫描结果寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TK_CNT<7:0>							

Bit7-0 **TK_CNT<7:0>** : TK 扫描结果低 8 位

TK_CNTH TK 扫描结果寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
TK_CNT<15:8>							

Bit7-0 **TK_CNT<15:8>** : TK 扫描结果高 8 位

AN_IE 模拟中断使能寄存器 (与 ADC LVD LED 共用地址)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-0	—
—	—	—	—	—	—	TK_IE	—

Bit7-2 保留

Bit1 **TK_IE** : TK 中断使能位

1 : 使能

0 : 禁止

Bit0 保留

AN_IF 模拟中断标志寄存器 (与 ADC LVD LED 共用地址, 支持位操作)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-0	—
—	—	—	—	—	—	TK_IF	—

Bit7-2 保留

Bit1 **TK_IF** : TK 中断标志位

1 : 有 TK 中断请求

0 : 无 TK 中断请求

Bit0 保留

20 内部参考电压 VREF

20.1 描述

芯片内置高精度参考电压模块 VREF，出厂校准精度 $< \pm 1\%$ 。

VREF 模块用于 ADC 和 TK 的内部参考电压，当 ADC 和 TK 选择 VREF 作为参考时，需预先通过 VREF_C 寄存器使能 VREF 模块，不选用 VREF 时需将模块关闭，以节省电流消耗。

20.2 VREF 寄存器

VREF_C VREF 配置寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
VREF_C<7:0>							

Bit7-0 VREF_C<7:0> : VREF 配置位

EBH : 使能内部参考电压

00H : 关闭 VREF 模块

其他 : 保留

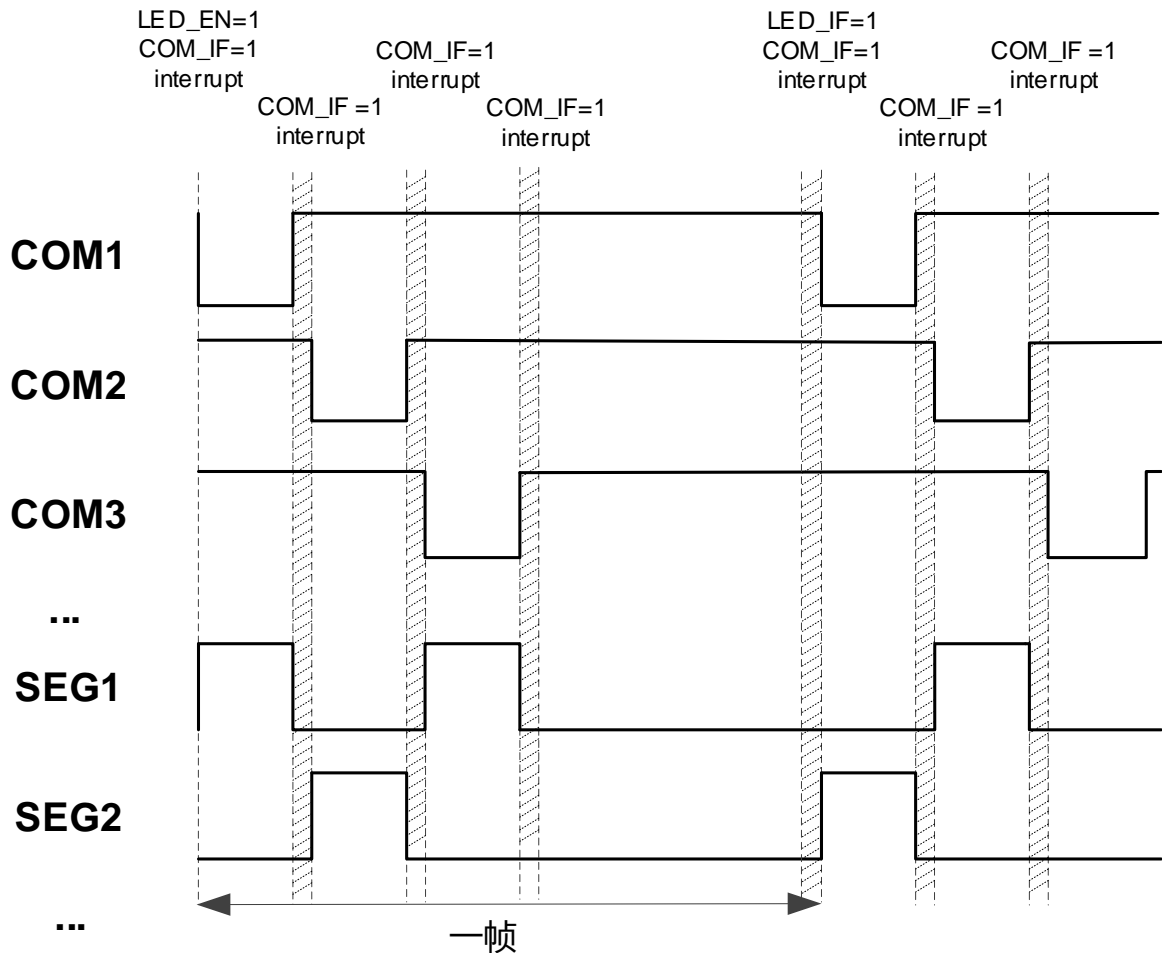
21 LED 驱动

21.1 描述

LED 驱动电路为共阴极驱动模式，最多支持 8 个 COM 端口和 12 个 SEG 端口。

- 采用 LRC 作为时钟源时，支持 CPU 睡眠显示
- COM 端口为大电流驱动口，灌电流能力 80mA/15mA 可选
- COM 扫描时长线性可设，可选切换死区时长
- SEG 端口支持 4 档恒流驱动模式 4mA、8mA、12mA、15mA
- SEG 支持 8 档占空比调节 1/8~8/8，用于调节显示亮度。

下图为 LED 驱动波形示例，COM 端口低电平驱动有效，SEG 端口高电平驱动有效。示例为 7/8 占空比显示亮度。



图表 38 LED 驱动波形示意图

21.2 LED 寄存器

LED_C0 LED 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	—	—	—	—	—	RW-0	RW-0
LED_EN	—	—	—	—	—	LED_SEGM	LED_MODE

- Bit7 **LED_EN** : LED 模块使能位
1 : 使能
0 : 关闭
- Bit6-2 保留
- Bit1 **LED_SEGM** : LED SEG 数据来源配置位
1 : SEGD 寄存器
0 : LED SRAM
- Bit0 **LED_MODE** : LED 中断运行模式选择位
1 : 触发中断 LEDIF 时, LED 停止扫描, 需重新使能 LED_EN 继续扫描
0 : 触发中断 LEDIF 时, LED 继续扫描

LED_C1 LED 控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-000			RW-01		RW-00	
—	LED_GS<2:0>			LED_CKS		LED_DS<1:0>	

- Bit7 保留
- Bit6-4 **LED_GS<2:0>** : LED 亮度调节位 (SEG 占空比)
111 : 8/8 占空比 110 : 7/8 占空比 101 : 6/8 占空比 100 : 5/8 占空比
011 : 4/8 占空比 010 : 3/8 占空比 001 : 2/8 占空比 000 : 1/8 占空比
- Bit3-2 **LED_CKS** : LED 时钟源选择位
11 : 保留
10 : XTAL
01 : HRC/256
00 : LRC
- Bit1-0 **LED_DS<1:0>** : COM 死区时长选择位
11 : $T_{LED_CK} \times 4$ 10 : $T_{LED_CK} \times 2$
01 : $T_{LED_CK} \times 1$ 00 : 无死区

COM_D COM 扫描时长寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
COM_D<7:0>							

- Bit7-0 **COM_D<7:0>** : COM 扫描时长设置位
LED_CKS=10 : $T_{COM} = T_{XTAL} \times COM_D$
LED_CKS=01 : $T_{COM} = T_{HRC} \times 256 \times COM_D$
LED_CKS=00 : $T_{COM} = T_{LRC} \times COM_D$

注：为配合 SRAM 数据预取，当 LED_SEG_M = 0 时 COM_D 设置值需 ≥ 4；当 LED_SEG_M = 1 时 COM_D 设置值最小值为 1，此时 LED_EN 使能中断和第一个 COM 扫描结束中断同一时间置起。

COM_EN COM 使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0	RW-0
C7_EN	C6_EN	C5_EN	C4_EN	C3_EN	C2_EN	C1_EN	C0_EN

Bit7 **Cn_EN** : COM7~COM0 使能位.
1 : 使能
0 : 关闭

SEGDL SEG 数据寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
SEGD<7:0>							

Bit7-0 **SEGD<7:0>** : seg 数据寄存器低 8 位

SEGDH SEG 数据寄存器高 4 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—				RW-0000			
—				SEGD<11:8>			

Bit7-4 保留

Bit3-0 **SEGD<11:8>** : seg 数据寄存器高 4 位

AN_IE 模拟中断使能寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	RW-0	RW-0	—	—	—	—
—	—	COM_IE	LED_IE	—	—	—	—

Bit7-6 保留

Bit5 **COM_IE** : COM 中断使能位

1 : 使能

0 : 禁止

Bit4 **LED_IE** : LED 中断使能位

1 : 使能

0 : 禁止

Bit3-0 保留

AN_IF 模拟中断标志寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	RW-0	RW-0	—	—	—	—
—	—	COM_IF	LED_IF	—	—	—	—

Bit7-6 保留

Bit5 **COM_IF** : COM 中断标志位

1 : 有 COM 中断请求

0 : 无 COM 中断请求

Bit4 **LED_IF** : LED 中断标志位

1 : 有 LED 中断请求

0 : 无 LED 中断请求

Bit3-0 保留

21.1 SRAM 映射

地址		7	6	5	4	3	2	1	0
02F0H	COM0	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02F1H	COM0	—	—	—	—	SEG12	SEG11	SEG10	SEG9
02F2H	COM1	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02F3H	COM1	—	—	—	—	SEG12	SEG11	SEG10	SEG9
02F4H	COM2	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02F5H	COM2	—	—	—	—	SEG12	SEG11	SEG10	SEG9
02F6H	COM3	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02F7H	COM3	—	—	—	—	SEG12	SEG11	SEG10	SEG9
02F8H	COM4	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02F9H	COM4	—	—	—	—	SEG12	SEG11	SEG10	SEG9
02FAH	COM5	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02FBH	COM5	—	—	—	—	SEG12	SEG11	SEG10	SEG9
02FCH	COM6	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02FDH	COM6	—	—	—	—	SEG12	SEG11	SEG10	SEG9
02FEH	COM7	SEG8	SEG7	SEG6	SEG5	SEG4	SEG3	SEG2	SEG1
02FFH	COM7	—	—	—	—	SEG12	SEG11	SEG10	SEG9

22 低电压检测 LVD

22.1 描述

芯片内置低电压检测模块 LVD。

该模块用于监测 VDD 电压，也可用于对 LVDIN 复用端口输入电压进行监测检测值为 0.5V。

22.2 LVD 寄存器

LVD_C0 LVD 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-0	—	—	RW-000		
LVD_EN	LVD_IM	LVD_FLT	—	—	LVD_VS<2:0>		

Bit7 **LVD_EN** : LVD 使能位

1 : 使能

0 : 关闭

Bit6 **LVD_IM** : LVD 中断模式选择位

1 : 被监测电压由高于比较阈值产生 LVD 中断

0 : 被监测电压由低于比较阈值产生 LVD 中断

Bit5-4 **保留**

Bit3 **LVD_INS** : 被监测电压选择位

1 : LVDI 复用管脚 (比较阈值电压固定为 0.5V)

0 : VDD

Bit2-0 **LVD_VS<2:0>** : 阈值电压选择位 (仅用于监测 VDD)

111 : 4.6V 110 : 4.2V 101 : 3.8V 100 : 3.4V

011 : 3.0V 010 : 2.6V 001 : 2.2V 000 : 1.8V

AN_IE 模拟中断使能寄存器 (与 TK ADC LED 共用地址)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	RW-0	—	—
—	—	—	—	—	LVD_IE	—	—

Bit7-3 **保留**

Bit2 **LVD_IE** : LVD 中断使能位

1 : 使能

0 : 禁止

Bit1-0 **保留**

AN_IF 模拟中断标志寄存器 (与 TK ADC LED 共用地址, 支持位操作)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	RW-0	—	—
—	—	—	—	—	LVD_IF	—	—

Bit7-3 **保留**

Bit2 **LVD_IF** : LVD 中断标志位

1 : 有 LVD 中断请求

0 : 无 LVD 中断请求

Bit1-0 **保留**

23 指令集

23.1 算术运算指令

指令		描述	字节	周期
ADD	A, Rn	$A = A + Rn$	1	1
ADD	A, direct	$A = A + [direct]$	2	2
ADD	A, @Ri	$A = A + [Ri]$	1	2
ADD	A, #data	$A = A + data$	2	2
ADDC	A, Rn	$A = A + Rn + C$	1	1
ADDC	A, direct	$A = A + [direct] + C$	2	2
ADDC	A, @Ri	$A = A + [Ri] + C$	1	2
ADDC	A, #data	$A = A + data + C$	2	2
SUBB	A, Rn	$A = A - Rn - C$	1	1
SUBB	A, direct	$A = A - [direct] - C$	2	2
SUBB	A, @Ri	$A = A - [Ri] - C$	1	2
SUBB	A, #data	$A = A - data - C$	2	2
INC	A	$A = A + 1$	1	1
INC	Rn	$Rn = Rn + 1$	1	2
INC	direct	$[direct] = [direct] + 1$	2	3
INC	@Ri	$[Ri] = [Ri] + 1$	1	3
DEC	A	$A = A - 1$	1	1
DEC	Rn	$Rn = Rn - 1$	1	2
DEC	direct	$[direct] = [direct] - 1$	2	3
DEC	@Ri	$[Ri] = [Ri] - 1$	1	3
INC	DPTR	$DPTR = DPTR + 1$	1	4
MUL	AB	$B:A = B \times A$	1	11
DIV	AB	$A = \text{Int} [A/B] ; B = \text{Mod} [A/B]$	1	11
DA	A	十进制调整	1	1

23.2 逻辑操作指令

指令	描述	字节	周期
ANL	A, Rn	$A = A \text{ and } Rn$	1
ANL	A, direct	$A = A \text{ and } [direct]$	2
ANL	A, @Ri	$A = A \text{ and } [Ri]$	1
ANL	A, #data	$A = A \text{ and } data$	2
ANL	direct, A	$[direct] = [direct] \text{ and } A$	2
ANL	direct, #data	$[direct] = [direct] \text{ and } data$	3
ORL	A, Rn	$A = A \text{ or } Rn$	1
ORL	A, direct	$A = A \text{ or } [direct]$	2
ORL	A, @Ri	$A = A \text{ or } [Ri]$	1
ORL	A, #data	$A = A \text{ or } data$	2
ORL	direct, A	$[direct] = [direct] \text{ or } A$	2
ORL	direct, #data	$[direct] = [direct] \text{ or } data$	3
XRL	A, Rn	$A = A \text{ xor } Rn$	1
XRL	A, direct	$A = A \text{ xor } [direct]$	2
XRL	A, @Ri	$A = A \text{ xor } [Ri]$	1
XRL	A, #data	$A = A \text{ xor } data$	2
XRL	direct, A	$[direct] = [direct] \text{ xor } A$	2
XRL	direct, #data	$[direct] = [direct] \text{ xor } data$	3
CLR	A	$A = 00H$	1
CPL	A	$A = \text{not } A$	1
RL	A	$A<7:0> = \{A<6:0>, A<7>\}$	1
RLC	A	$\{C, A<7:0>\} = \{A<7:0>, C\}$	1
RR	A	$A<7:0> = \{A<0>, A<7:1>\}$	1
RRC	A	$\{C, A<7:0>\} = \{A<0>, C, A<7:1>\}$	1
SWAP	A	$A<7:0> = \{A<3:0>, A<7:4>\}$	1

23.3 数据传送指令

指令		描述	字节	周期
MOV	A, Rn	A = Rn	1	1
MOV	A, direct	A = [direct]	2	2
MOV	A, @Ri	A = [Ri]	1	2
MOV	A, #data	A = data	2	2
MOV	Rn, A	Rn = A	1	2
MOV	Rn, direct	Rn = [direct]	2	3
MOV	Rn, #data	Rn = data	2	2
MOV	direct, A	[direct] = A	2	2
MOV	direct, Rn	[direct] = Rn	2	2
MOV	direct1, direct2	[direct1] = [direct2]	3	3
MOV	direct, @Ri	[direct] = [Ri]	2	3
MOV	direct, #data	[direct] = data	3	3
MOV	@Ri, A	[Ri] = A	1	2
MOV	@Ri, direct	[Ri] = [direct]	2	3
MOV	@Ri, #data	[Ri] = data	2	2
MOV	DPTR, #data 16	DPTR = data(16-bit)	3	3
MOVC	A, @A+DPTR	A = [A+DPTR](程序代码)	1	7
MOVC	A, @A+PC	A = [A+PC](程序代码)	1	8
MOVX	A, @Ri	A = [Ri](核外 RAM)	1	5
MOVX	A, @DPTR	A = [DPTR](核外 RAM)	1	6
MOVX	@Ri, A	[Ri](核外 RAM) = A	1	4
MOVX	@DPTR, A	[DPTR](核外 RAM) = A	1	5
PUSH	direct	SP=SP+1, [SP] = [direct]	2	5
POP	direct	[direct] = [SP], SP = SP-1	2	5
XCH	A, Rn	A ↔ Rn	1	3
XCH	A, direct	A ↔ [direct]	2	4
XCH	A, @Ri	A ↔ [Ri]	1	4
XCHD	A, @Ri	A<3:0> ↔ [Ri]<3:0>	1	4

23.4 位操作指令

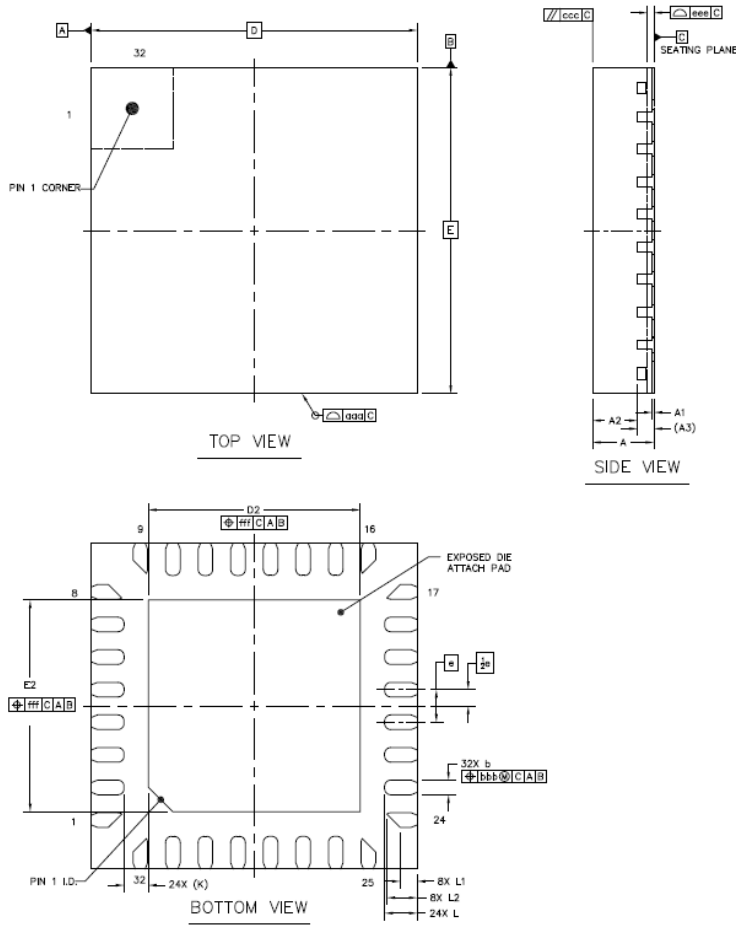
指令		描述	字节	周期
CLR	C	$C = 0$	1	1
CLR	bit	$bit = 0$	2	3
SETB	C	$C = 1$	1	1
SETB	bit	$bit = 1$	2	3
CPL	C	$C = \text{not } C$	1	1
CPL	bit	$bit = \text{not } bit$	2	3
ANL	C, bit	$C = C \text{ and } bit$	2	2
ANL	C, /bit	$bit = C \text{ and } (\text{not } bit)$	2	2
ORL	C, bit	$C = C \text{ or } bit$	2	2
ORL	C, /bit	$bit = C \text{ or } (\text{not } bit)$	2	2
MOV	C, bit	$C = bit$	2	2
MOV	bit, C	$bit = C$	2	3

23.5 程序转移指令

指令		描述	字节	周期
ACALL	addr11	2K 空间子程序调用	2	7
LCALL	addr16	64K 空间子程序调用	3	7
RET		调用程序返回	1	8
RETI		中断返回	1	8
AJMP	addr11	2K 空间程序跳转	2	4
LJMP	addr16	64K 空间程序跳转	3	5
SJMP	rel	相对短跳转	2	4
JMP	@A+DPTR	相对长跳转	1	6
JZ	rel	相对短跳转(如果 A=0)	2	3/5
JNZ	rel	相对短跳转(如果 A≠0)	2	3/5
JC	rel	相对短跳转(如果 C=1)	2	2/4
JNC	rel	相对短跳转(如果 C=0)	2	2/4
JB	bit, rel	相对短跳转(如果 bit=1)	3	4/6
JNB	bit, rel	相对短跳转(如果 bit=0)	3	4/6
JBC	bit, rel	相对短跳转(如果 bit=1), bit = 0	3	4/6
CJNE	A, direct, rel	相对短跳转(如果 A≠[direct])	3	4/6
CJNE	A, #data, rel	相对短跳转(如果 A≠data)	3	4/6
CJNE	Rn, #data, rel	相对短跳转(如果 Rn≠data)	3	4/6
CJNE	#Ri, #data, rel	相对短跳转(如果 [Ri]≠data)	3	4/6
DJNZ	Rn, rel	$Rn = Rn - 1$, 相对短跳转(如果 $Rn \neq 0$)	2	3/5
DJNZ	direct, rel	$[direct] = [direct] - 1$, 相对短跳转(如果 $[direct] \neq 0$)	2	3/5
NOP		空操作	1	1

24 封装信息

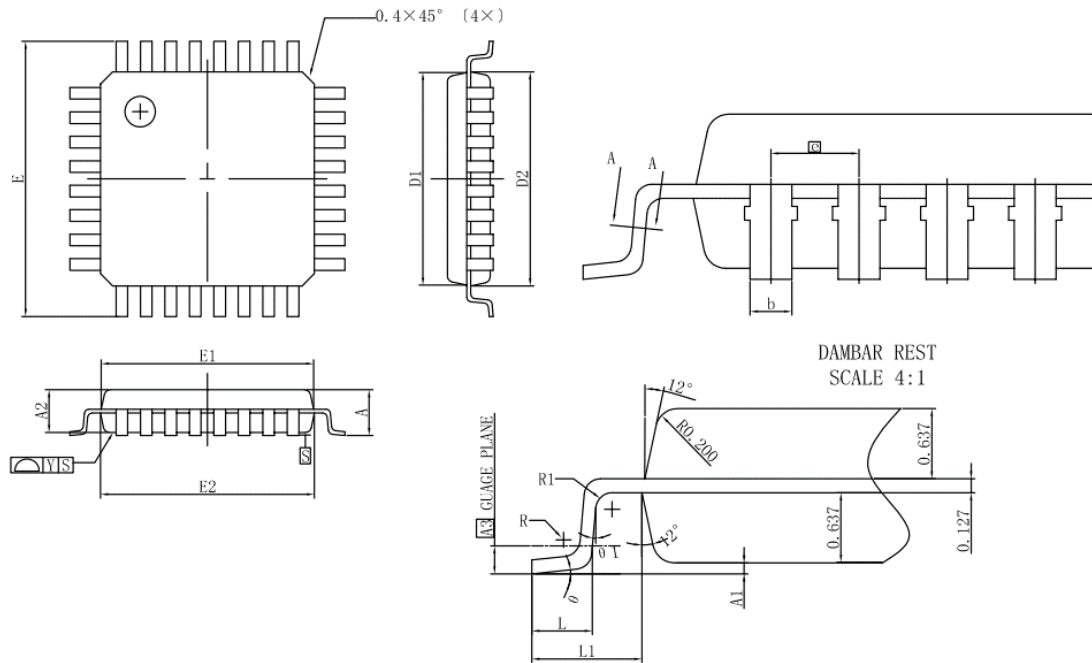
QFN32 封装尺寸



	SYMBOL	MIN	NOM	MAX	
TOTAL THICKNESS	A	0.7	0.75	0.8	
STAND OFF	A1	0	0.02	0.05	
MOLD THICKNESS	A2	---	0.55	---	
L/F THICKNESS	A3	0.203 REF			
LEAD WIDTH	b	0.15	0.2	0.25	
BODY SIZE	X	4 BSC			
	Y	4 BSC			
LEAD PITCH	e	0.4 BSC			
EP SIZE	X	D2	2.5	2.6	2.7
	Y	E2	2.5	2.6	2.7
LEAD LENGTH	L	0.3	0.4	0.5	
	L1	0.108	0.208	0.308	
	L2	0.272	0.372	0.472	
LEAD TIP TO EXPOSED PAD EDGE	K	0.3 REF			
PACKAGE EDGE TOLERANCE	aaa	0.1			
MOLD FLATNESS	ccc	0.1			
COPLANARITY	eee	0.08			
LEAD OFFSET	bbb	0.1			
EXPOSED PAD OFFSET	fff	0.1			

NOTES
 1. REFER TO JEDEC MO-220;
 2. COPLANARITY APPLIES TO LEADS, CORNER LEADS AND DIE ATTACH PAD;
 3. BAN TO USE THE LEVEL 1 ENVIRONMENT-RELATED SUBSTANCES OF JQET PRESCRIBING;
 4. FINISH: Cu/EP + Sn8~20s

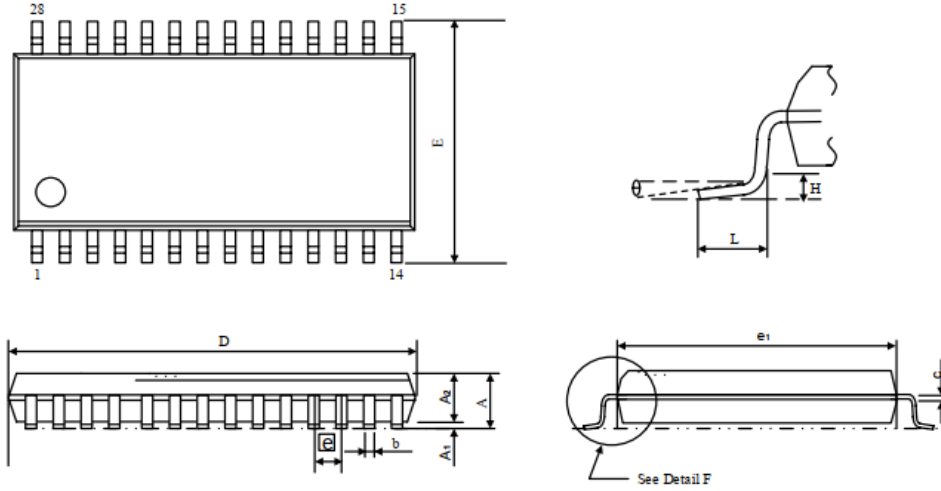
LQFP32 封装尺寸



标号	公制 (mm)			英制 (inch)		
	MIN	NOM	MAX	MIN	NOM	MAX
D	8.80	9.00	9.20	0.35	0.35	0.36
E	8.80	9.00	9.20	0.35	0.35	0.36
D1	6.85	6.95	7.05	0.27	0.27	0.28
E1	6.85	6.95	7.05	0.27	0.27	0.28
A	1.45	1.55	1.65	0.06	0.06	0.06
A1	0.01		1.60	0.00		0.06
A2	0.08		0.21	0.00		0.01
b	0.30	0.35	0.40	0.01	0.01	0.02
e		0.80			0.05	
L	0.43		0.71	0.02		0.03
L1	0.90	1.00	1.10	0.04	0.04	0.04
θ	0°		10°	0°		10°

TSSOP28 封装尺寸

TSSOP28 外形尺寸 单位：毫米



符号	mm(毫米)		
	最小	正常	最大
A	-	-	1.200
A1	0.050	-	0.150
A2	0.800	-	1.000
b	0.190	-	0.300
c	0.090	-	0.200
D	9.600	-	9.800
E	6.250	-	6.550
e1	4.300	-	4.500
e	0.65(BSC)		
L	0.500	-	0.700
θ	1°	-	7°
H	0.25(TYP)		

SSOP24

