

SinoMCU 8 位单片机

MC51F8114

用户手册

1T 8051 内核触控按键微控制器

V1.1



上海晟矽微电子股份有限公司

Shanghai SinoMCU Microelectronics Co., Ltd.

本公司保留对产品可靠性、功能和设计方面的改进作进一步说明的权利

本文档的更改，恕不另行通知

目录

1	产品特性	5
2	管脚配置	6
3	功能框图	10
4	电气特性	11
4.1	极限参数	11
4.2	工作条件	11
4.3	DC 特性.....	11
4.4	存储器特性	12
4.5	AC 特性.....	12
5	内核	13
5.1	描述	13
5.2	内核寄存器	13
6	存储	15
6.1	程序寻址空间映射	15
6.2	FLASH 程序存储器.....	15
6.3	用户配置信息	16
6.4	ISP 应用考虑.....	19
6.5	IAP 操作.....	20
6.6	数据寻址空间	23
7	复位	31
7.1	描述	31
7.2	复位寄存器	32
8	时钟	33
8.1	描述	33
8.2	时钟寄存器	34
9	低功耗	38
9.1	描述	38
9.2	低功耗寄存器	38
10	看门狗定时器 WDT	39
10.1	描述	39
10.2	WDT 寄存器	39
11	中断	40
11.1	中断向量	40

11.2 中断优先级	40
11.3 中断寄存器	41
11.4 外部端口中断 EINT	42
11.5 定时器 TMR 中断	43
11.6 边沿捕捉器 CAP 中断	43
11.7 脉宽调制器 PWM 中断	44
11.8 异步收发器 UART 中断	45
11.9 IIC 总线控制器中断	46
11.10 SPI 模块中断	47
11.11 模拟模块中断	48
12 I/O 端口	50
12.1 I/O 功能框图	50
12.2 I/O 寄存器	50
12.3 描述	57
12.4 TMR 寄存器	59
13 边沿捕捉器 CAP	61
13.1 描述	61
13.2 CAP 寄存器	63
14 脉宽调制器 PWM	64
14.1 描述	64
14.2 PWM 寄存器	66
15 UART 异步收发器	68
15.1 描述	68
15.2 UART 寄存器	71
16 IIC 总线控制器	74
16.1 描述	74
16.2 IIC 寄存器	77
17 SPI 通讯端口	80
17.1 描述	80
17.1 SPI 寄存器	82
18 模数转换器 ADC	84
18.1 描述	84
18.2 ADC 寄存器	85
19 触控按键控制器 TK	88
19.1 描述	88

19.2 TK 功能框图	88
19.3 TK 寄存器	89
20 内部参考电压 VREF	91
20.1 描述	91
20.2 VREF 寄存器	91
21 低电压检 LVD	92
21.1 描述	92
21.2 LVD 寄存器	92
22 指令集	93
22.1 算术运算指令	93
22.2 逻辑操作指令	94
22.3 数据传送指令	95
22.4 位操作指令	96
22.5 程序转移指令	96
23 封装信息	97
24 修订记录	98

1 产品特性

➤ 内核

- 1T 8051 内核，兼容标准 8051 指令集
- 内核最高工作频率 16MHz

➤ 存储

- 16K 字节 MAIN FLASH 程序存储器，支持 IAP（应用中自编程）功能（IAP 操作时 CPU 暂停运行）
- 512 字节 DATA FLASH 数据存储器，支持 IAP（应用中自编程）功能（IAP 操作时 CPU 暂停运行）
- 核内 256 字节 SRAM，核外 768 字节 SRAM

➤ 复位

- 内置上电复位 POR
- 内置掉电复位 BOR，支持 4 档掉电检测电压 2.1V, 2.5V, 3.7V, 4.2V
- 支持外部复位端口 RSTN，低电平复位

➤ 时钟

- 外部 2-20MHz 晶体振荡器，支持振荡器停振检测功能
- 内部 16MHz 高精度振荡器（出厂校准精度 $<\pm 1\%$ ，全温工作精度 $<\pm 2\%$ ）
- 内部 16KHz 低功耗振荡器

➤ 工作条件

- VDD 工作电压范围 2.0~5.5V
- VREF 工作电压范围 2.6~5.5V
- 工作温度范围 -40~85°C

➤ 低功耗

- 支持 IDLE 和 SLEEP 两种低功耗模式
- 待机睡眠功耗典型值 3uA
- 16KHz 运行功耗典型值 25uA
- 16MHz 运行功耗典型值 3mA

➤ 端口

- 最多支持 18 个 I/O 端口
- 所有端口支持独立弱上拉和弱下拉控制
- 所有端口支持外部中断功能复用
- 最多支持 10 个大电流驱动输出口，最大灌电流 80mA

➤ 外设

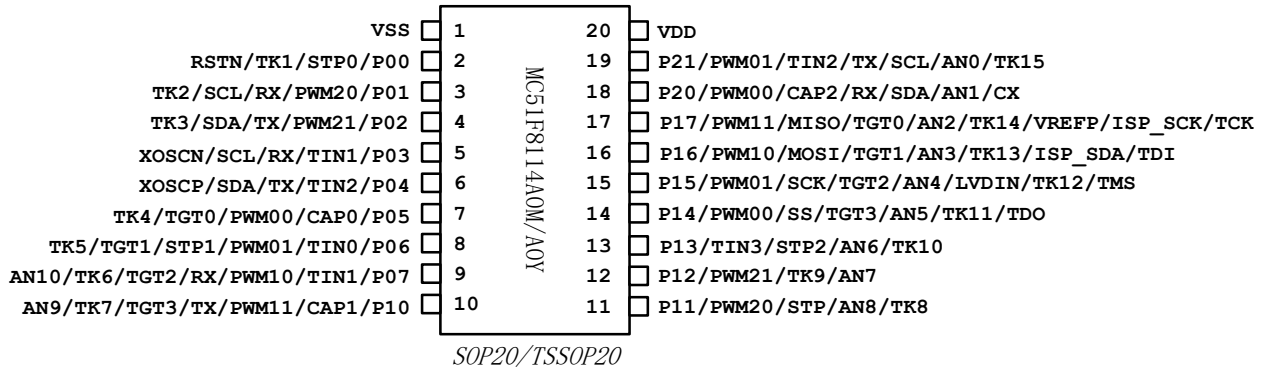
- 15 路触控按键检测器 TK
- 4 路 16 位定时/计数器 TMR
- 3 路边沿捕捉器 CAP
- 3 组独立 16 位脉宽调制器 PWM，每组支持 2 个 PWM 输出通道
- 1 路 IIC 总线控制器，支持 3 组分时通信端口
- 1 路 SPI 总线控制器
- 1 路 UART 收发器，支持 4 组分时通信端口
- 内置低电压检测模块 LVD
- 12 通道 12 位 SAR 结构 ADC，最高采样率 500KHz
- 内置多档可选高精度参考电压 VREF（出厂校准精度 $<\pm 10\text{mV}$ ）

➤ 封装

- TSSOP20/SOP20

2 管脚配置

MC51F8114A0M/A0Y



*ISP_SCK, ISP_SDA 为编程管脚；同时建议在布板时将 P00 一并接出；P00 作为复位管脚使用时将参与编程时序的运行

*TMS, TCK, TDI, TDO 为仿真管脚

*P01、P02、P05、P06、P07、P10、P11、P12、P16、P17 为大电流输出口

下表格中“其他”为模拟管脚功能或配置字控制的功能，作为模拟管脚使用时管脚输出功能需关闭，建议也将输入功能关闭。

其他	FUN3	FUN2	FUN1	FUN0	PIN		FUN0	FUN1	FUN2	FUN3	其他
RSTN/TK1	—	STP0	—	P00	P00	VSS	VSS				
TK2	SCL	RX	PWM20	P01*	P01	VDD	VDD				
TK3	SDA	TX	PWM21	P02*	P02	P21	P21	PWM01/TIN2	TX	SCL	AN0/TK15
XOSCN	SCL	RX	TIN1	P03	P03	P20	P20	PWM00/CAP2	RX	SDA	AN1/ CX
XOSCP	SDA	TX	TIN2	P04	P04	P17	P17*	PWM11	MISO	TGT0	AN2/TK14/VREFP
TK4	TGT0	—	PWM00/CAP0	P05*	P05	P16	P16*	PWM10	MOSI	TGT1	AN3/TK13
TK5	TGT1	STP1	PWM01/TIN0	P06*	P06	P15	P15	PWM01	SCK	TGT2	AN4/LVDIN/TK12
AN10/TK6	TGT2	RX	PWM10/TIN1	P07*	P07	P14	P14	PWM00	SS	TGT3	AN5/TK11
AN9/TK7	TGT3	TX	PWM11/CAP1	P10*	P10	P13	P13	TIN3	STP2	—	AN6/TK10
AN7/TK9	—	—	PWM21	P12*	P12	P11	P11*	PWM20	—	STP	AN8/TK8

图表 1 MC51F8114 管脚复用表

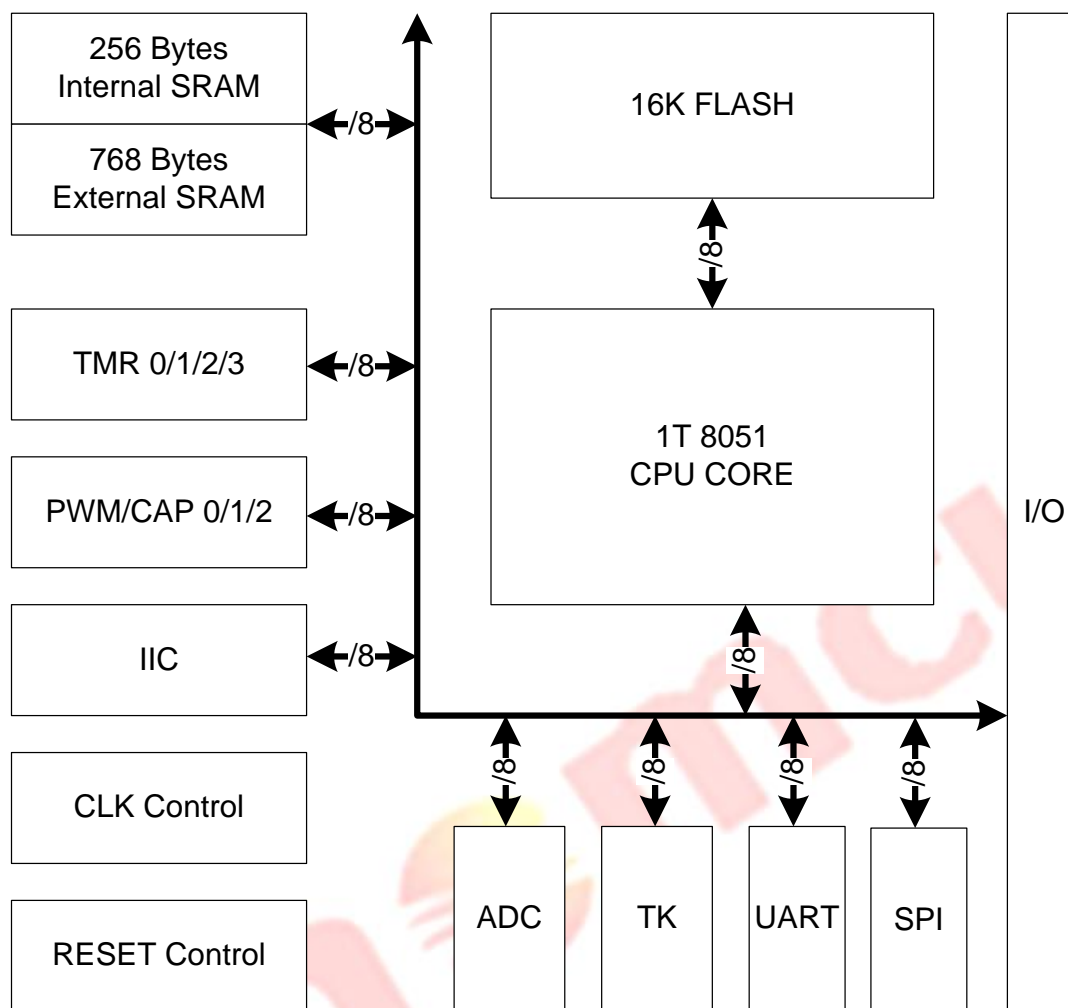
PIN	符号	描述	Px_FUN	Px_IE	Px_OE	Px_ODE	其他
P00	P00	通用 I/O	00	x	x	x	RSTN_EN=0
	STP0	PWM0 刹车输入	10	1	0	x	RSTN_EN=0
	RSTN	芯片外部复位输入	xx	x	x	x	RSTN_EN=1
	TK1	触控扫描通道 1	xx	0	0	x	RSTN_EN=0
P01	P01	通用 I/O (大电流输出)	00	x	x	x	—
	PWM20	PWM2 通道 0 输出	01	x	1	0	—
	RX	UART 接收端	10	1	0	0	—
	SCL	IIC 时钟线	11	1	1	1	—
	TK2	触控扫描通道 2	xx	0	0	x	—
P02	P02	通用 I/O (大电流输出)	00	x	x	x	—
	PWM21	PWM2 通道 1 输出	01	x	1	0	—
	TX	UART 发送端	10	x	1	0	—
	SDA	IIC 数据线	11	1	1	1	—
	TK3	触控扫描通道 3	xx	0	0	x	—
P03	P03	通用 I/O	00	x	x	x	—
	TIN1	TMR1 外部时钟输入	01	1	0	x	—
	RX	UART 接收端	10	1	0	0	—
	SCL	IIC 时钟线	11	1	1	1	—
	XOSCN	外部振荡器端口 (输出)	xx	x	x	x	XOSC 使能
P04	P04	通用 I/O	00	x	x	x	—
	TIN2	TMR2 外部时钟输入	01	1	0	x	—
	TX	UART 发送端	10	x	1	0	—
	SDA	IIC 数据线	11	1	1	1	—
	XOSCP	外部振荡器端口 (输入)	xx	x	x	x	XOSC 使能
P05	P05	通用 I/O (大电流输出)	00	x	x	x	—
	PWM00	PWM0 通道 0 输出	01	x	1	0	—
	CAPO	CAPO 捕捉输入	01	1	0	x	—
	TGT0	TMR0 门控输入	11	1	0	x	—
	TK4	触控扫描通道 4	xx	0	0	x	—
P06	P06	通用 I/O (大电流输出)	00	x	x	x	—
	PWM01	PWM0 通道 1 输出	01	x	1	0	—
	TIN0	TMR0 外部时钟输入	01	1	0	x	—
	STP1	PWM1 刹车输入	10	1	0	x	—
	TGT1	TMR1 门控输入	11	1	0	x	—
	TK5	触控扫描通道 5	xx	0	0	x	—
P07	P07	通用 I/O (大电流输出)	00	x	x	x	—
	PWM10	PWM1 通道 0 输出	01	x	1	0	—
	TIN1	TMR1 外部时钟输入	01	1	0	x	—
	RX	UART 接收端	10	1	0	0	—
	TGT2	TMR2 门控输入	11	1	0	x	—
	AN10	ADC 通道 10	xx	0	0	x	—
	TK6	触控扫描通道 6	xx	0	0	x	—
P10	P10	通用 I/O (大电流输出)	00	x	x	x	—

PIN	符号	描述	Px_FUN	Px_IE	Px_OE	Px_ODE	其他
	PWM11	PWM1 通道 1 输出	01	x	1	0	—
	CAP1	CAP1 捕捉输入	01	1	0	x	—
	TX	UART 发送端	10	x	1	0	—
	TGT3	TMR3 门控输入	11	1	0	x	—
	AN9	ADC 通道 9	xx	0	0	x	—
	TK7	触控扫描通道 7	xx	0	0	x	—
P11	P11	通用 I/O (大电流输出)	00	x	x	x	—
	PWM20	PWM2 通道 0 输出	01	x	1	0	—
	STP	所有 PWM 刹车输入	11	1	0	0	—
	AN8	ADC 通道 8	xx	0	0	x	—
	TK8	触控扫描通道 8	xx	0	0	x	—
P12	P12	通用 I/O (大电流输出)	00	x	x	x	—
	PWM21	PWM2 通道 1 输出	01	x	1	0	—
	AN7	ADC 通道 7	xx	0	0	x	—
	TK9	触控扫描通道 9	xx	0	0	x	—
P13	P13	通用 I/O	00	x	x	x	—
	TIN3	TMR3 外部时钟输入	01	1	0	x	—
	STP2	PWM2 刹车输入	10	1	0	x	—
	AN6	ADC 通道 6	xx	0	0	x	—
	TK10	触控扫描通道 10	xx	0	0	x	—
P14	P14	通用 I/O	00	x	x	x	—
	PWM00	PWM0 通道 0 输出	01	x	1	0	—
	SS	SPI 片选信号	10	x	x	0	—
	TGT3	TMR3 门控输入	11	1	0	x	—
	AN5	ADC 通道 5	xx	0	0	x	—
	TK11	触控扫描通道 11	xx	0	0	x	—
P15	P15	通用 I/O	00	x	x	x	—
	PWM01	PWM0 通道 1 输出	01	x	1	0	—
	SCK	SPI 时钟信号	10	x	x	0	—
	TGT2	TMR2 门控输入	11	1	0	x	—
	AN4	ADC 通道 4	xx	0	0	x	—
	TK12	触控扫描通道 12	xx	0	0	x	—
	LVDIN	LVD 外部输入	xx	0	0	x	—
P16	P16	通用 I/O (大电流输出)	00	x	x	x	—
	PWM10	PWM1 通道 0 输出	01	x	1	0	—
	MOSI	SPI 主输出从输入	10	x	x	x	—
	TGT1	TMR1 门控输入	11	1	0	x	—

PIN	符号	描述	Px_FUN	Px_IE	Px_OE	Px_ODE	其他
	AN3	ADC 通道 3	xx	0	0	x	—
	TK13	触控扫描通道 13	xx	0	0	x	—
P17	P17	通用 I/O (大电流输出)	00	x	x	x	—
	PWM11	PWM1 通道 1 输出	01	x	1	0	—
	MISO	SPI 主输入从输出	10	x	x	0	—
	TGT0	TMR0 门控输入	11	1	0	x	—
	AN2	ADC 通道 2	xx	0	0	x	—
	TK14	触控扫描通道 14	xx	0	0	x	—
	VREFP	AD 外部参考输入	xx	0	0	x	—
P20	P20	通用 I/O	00	x	x	x	—
	PWM00	PWM0 通道 0 输出	01	x	1	0	—
	CAP2	CAP2 捕捉输入	01	1	0	x	—
	RX	UART 接收端	10	1	0	0	—
	SDA	IIC 数据线	11	1	1	1	—
	AN1	ADC 通道 1	xx	0	0	x	—
	CX	触控外接 CX 电容	xx	0	0	x	—
P21	P21	通用 I/O	00	x	x	x	—
	PWM01	PWM0 通道 1 输出	01	x	1	0	—
	TIN2	TMR2 外部时钟输入	01	1	0	x	—
	TX	UART 发送端	10	x	1	0	—
	SCL	IIC 时钟线	11	1	1	1	—
	AN0	ADC 通道 0	xx	0	0	x	—
	TK15	触控扫描通道 15	xx	0	0	x	—

图表 2 MC51F8114 管脚复用配置表

3 功能框图



图表 3 MC51F8114 功能框图

4 电气特性

4.1 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

4.2 工作条件

符号	描述	最小值	最大值	单位
F_{SYS_CLK}	系统工作频率	—	16M	Hz
VDD	工作电压	2.0	5.5	V
T_A	工作温度	-40	85	°C
t_{VDD}	VDD 上电斜率			us/V

4.3 DC 特性

* 以下参数均为设计值

典型值测试基本条件: $T_A=25^{\circ}C$, $V_{DD}=3V$

电流测试时 I/O 输出无负载, I/O 输入不浮空。

符号	描述	最小值	典型值	最大值	单位	条件
VDD	供电电压	2.0	—	5.5	V	—
I_{DDH}	高速工作电流	—	2	—	mA	$F_{SYS_CLK}=16MHz$
I_{DDL}	低速工作电流	—	20	25	uA	$F_{SYS_CLK}=16KHz$
I_{DDS}	待机电流	—	3	10	uA	进入 PD 模式, 所有功能关闭
V_{IL}	输入低电压	0	—	0.3 VDD	V	I/O 均为 SCHMITT 输入特性
V_{IH}	输入高电压	0.7 VDD	—	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	$V_{SS}+0.6$	—	—	V	大电流端口 $I_{OL}=60mA$ 非大电流端口 $I_{OL}=10mA$
V_{OH}	输出高电压	—	—	$V_{DD}-0.7$	V	$I_{OH}=10mA$
R_{PU}	内部上拉电阻	—	10K	—	Ω	—
R_{PD}	内部下拉电阻	—	10K	—	Ω	—

4.4 存储器特性

存储	操作	最小值	最大值	单位	条件
FLASH	擦写次数	100,000	—	—	最小擦写时间
	数据保持时间	100	—	year	室温
	擦除时间	20	40	ms	—
	编程时间	20	40	us	—
SRAM	最低数据保持电压	1.5	—	V	—

4.5 AC 特性

ADC 特性

符号	描述	最小值	典型值	最大值	单位	备注
V_{ADC}	ADC 工作电压	2.6	5.0	5.5	V	使用 V_{REF} 作为参考时最低工作电压为 2.6V
N_R	分辨率	—	—	12	bit	—
E_{IL}	积分非线性误差	—	± 1	± 2	LSB	—
E_{DL}	微分非线性误差	—	± 1	± 1	LSB	—
E_{OF}	偏置误差	—	± 1	± 2	LSB	—
E_{GN}	增益误差	—	± 1	± 2	LSB	—
V_{AIN}	输入电压	V_{SS}	—	V_{REF}	V	—
Z_{AIN}	输入阻抗	—	—	10K	Ω	—
T_{AD}	ADC 时钟周期	125	—	—	nS	—
T_{CNV}	ADC 转换周期数	—	12	—	T_{AD}	不包含采样时间

5 内核

5.1 描述

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

5.2 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: 算数或逻辑运算有进/借位
- 0: 算数或逻辑运算无进/借位

Bit6 AC: 辅助进位标志位 (用于 BCD 操作)

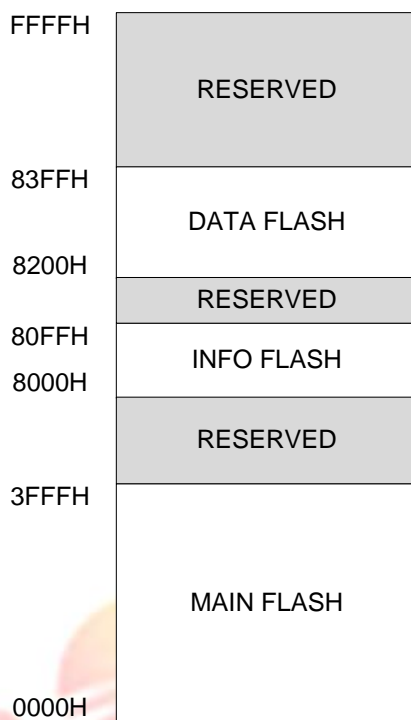
- 1: 算数或逻辑运算有辅助进/借位
- 0: 算数或逻辑运算无辅助进/借位

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: 有符号数运算有溢出 0: 有符号数运算无溢出
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

6 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。

6.1 程序寻址空间映射



图表 4 程序存储空间映射图

6.2 FLASH 程序存储器

数据区 DATA FLASH

芯片内置 512 字节 DATA FLASH 存储器用于存储数据。

DATA FLASH 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 存储器用于存储用户代码。

MAIN FLASH 支持以下操作：

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

6.3 用户配置信息

80FFH	reserved
803FH	UID_15

8030H	UID_00
	reserved
800DH	CFG_WD3
	reserved
8009H	CFG_WD2
	reserved
8005H	CFG_WD1
	reserved
8001H	CFG_WD0
	reserved

图表 5 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

用户 ID 共 16 字节 **UID_00~UID15**，映射于程序程序存储空间 8030H~803FH。用户 ID 存储的信息由用户自行定义。

用户配置选项 CFG_WD_n

用户配置选项共 4 字节 **CFG_WD0~CFG_WD3**。

CFG_WD0 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
 11: 上电默认选择 HRC, 软件可配置
 10: 固定选择 HRC
 01: 固定选择 XOSC
 00: 未用

Bit5-4 **PWRC<1:0>**: 上电延时选择位
 11: 约 128ms
 10: 约 64ms
 01: 约 16ms
 00: 无上电延时

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: 3.7V
 01: 2.5V
 00: 2.1V

CFG_WD1 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)

CFG_WD2 配置字 2

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
DBG_N	CCM_E	RSTN_EN	—	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 **CCM_E**: XOSC 停振检测使能位
 1: 使能
 0: 关闭

- Bit5 **RSTN_EN**: RSTN 复用功能使能位
 1: P00 用作 RSTN 复用功能
 0: P00 用作 I/O 功能
- Bit4 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_WD3 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_PRO<1:0>	

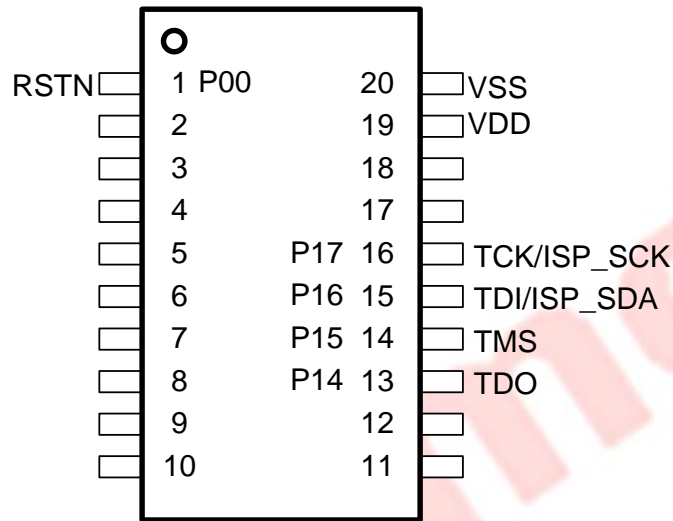
- Bit7-6 **IAP_PR3<1:0>**: MAIN FLASH 地址 3000H~3FFFH 操作权限配置位
 11: IAP 擦写允许, MOVC 指令读取允许
 10: IAP 擦写禁止, MOVC 指令读取允许
 0x: IAP 擦写禁止, MOVC 指令读取禁止
- Bit5-4 **IAP_PR2<1:0>**: MAIN FLASH 地址 2000H~2FFFH 操作权限配置位
 11: IAP 擦写允许, MOVC 指令读取允许
 10: IAP 擦写禁止, MOVC 指令读取允许
 0x: IAP 擦写禁止, MOVC 指令读取禁止
- Bit3-2 **IAP_PR1<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_PRO<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

6.4 ISP 应用考虑

芯片开发工具支持调试器和量产烧录器。调试器支持烧录和代码在线调试功能。量产烧录器支持量产批量烧录。调试器和烧录器均通过 ISP 复用端口进行操作。用户如需要在应用系统板上进行程序下载、更新或调试，需在 PCB 布板时预先对 ISP 复用端口予以考虑。

烧录器涉及管脚：ISP_SCK，ISP_SDA 为编程管脚，同时建议在布板时考虑将 P00 也一并接出；P00 作为复位管脚使用时将参与编程时序的运行。

调试器涉及管脚：TCK，TDI，TMS，TDO，为调试时使用管脚。



图表 61 ISP 复用端口定义图

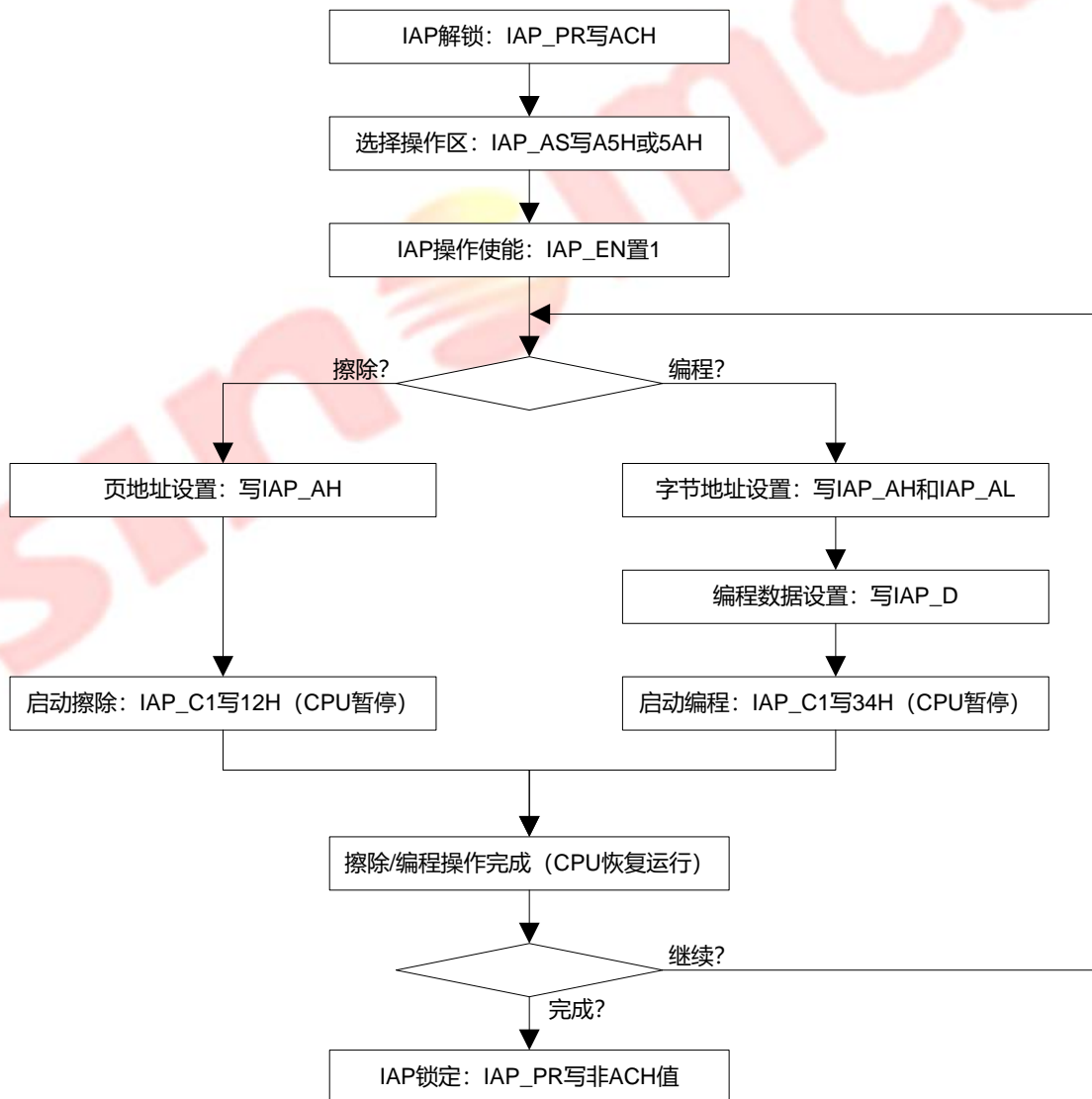
6.5 IAP 操作

MAIN FLASH 和 DATA FLASH 都支持应用中自编程 IAP 操作。MAIN FLASH 分为 32 页 (Page) , 每页包含 512 字节 (Byte) FLASH 存储单元, 页地址由 IAP_AH.IAP_A<15:9>指定。DATA FLASH 分为 2 页 (Page) , 每页包含 256 字节 (Byte) FLASH 存储单元, 页地址由 IAP_AH.IAP_A<15:8>指定(8200h 为 DATA FLASH 第一页起始地址, 8300h 为 DATA FLASH 第二页起始地址)。

IAP 操作包括页擦除和字节编程, 读取校验通过 MOVC 指令完成。IAP 页擦除操作的操作单位为页, IAP 编程操作的操作单位为字节, 字节地址由 IAP_AH 和 IAP_AL 指定。IAP 页擦除或编程操作启动时, CPU 自动进入暂停运行状态, 直到擦除或编程操作完成, 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。擦除或编程操作完成后, 恢复对中断的正常响应。

需注意的是页擦除时间为 2ms, 编程时间为 20us。

IAP 操作流程



图表 72 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_WD3** 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, 写入 5AH 选择数据区
 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

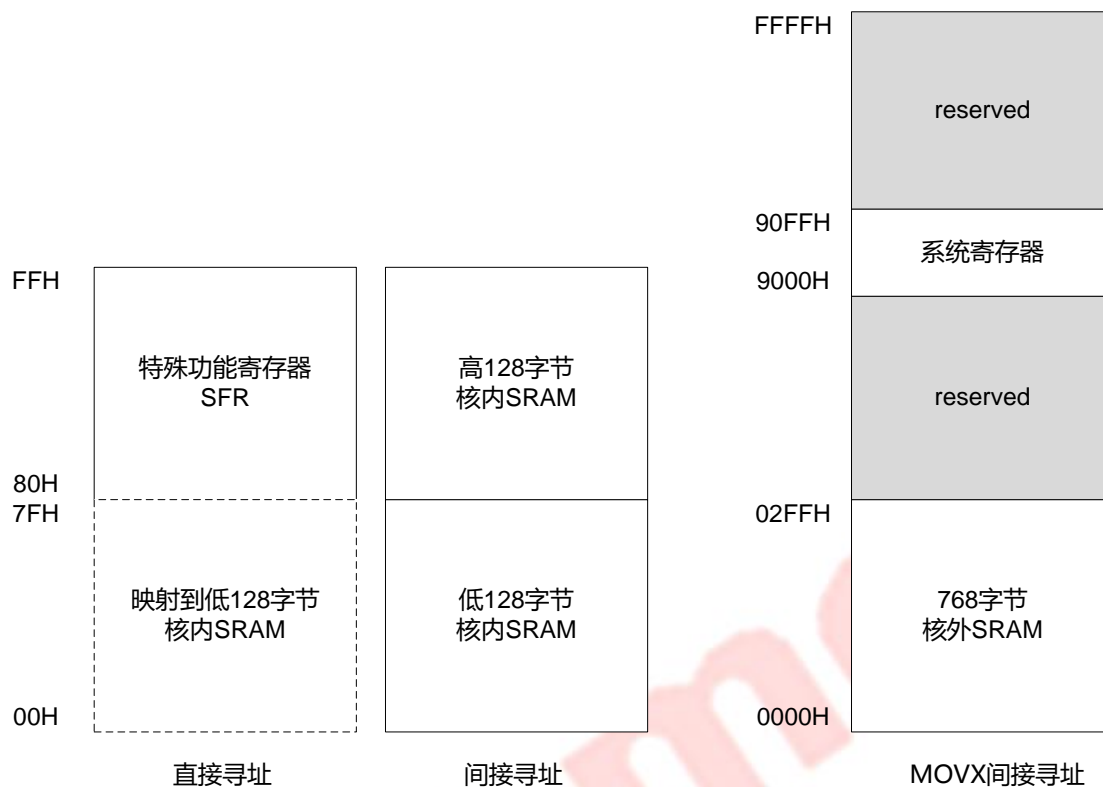
写 12H : 触发 IAP 擦除

写 34H : 触发 IAP 编程

写其他值: 无操作

6.6 数据寻址空间

数据寻址空间映射



图表 8 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	C8H	—	—	—	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_CKS	WDT_CY<2:0>		
WDT_OP	85H	WDT_OP<7:0>							
PORT_C0	8EH	—	—	PT_ODS<1:0>		PT_RDS<1:0>		PT_PLS	PT_IOS
P0_IE	8FH	P0_IE<7:0>							
P0_OE		P0_OE<7:0>							
P1_IE	91H	P1_IE<7:0>							
P1_OE		P1_OE<7:0>							
P2_IE	92H	—	—	—	—	—	—	P2_IE<1:0>	
P2_OE		—	—	—	—	—	—	P2_OE<1:0>	
P0_DAT	80H	P0_n<7:0>							
P1_DAT	90H	P1_n<7:0>							
P2_DAT	A0H	—	—	—	—	—	—	P2_n<1:0>	
P0_PUE	94H	P0_PUE<7:0>							
P0_PDE		P0_PDE<7:0>							
P1_PUE	95H	P1_PUE<7:0>							

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
P1_PDE		P1_PDE<7:0>							
P2_PUE	96H	—	—	—	—	—	—	P2_PUE<1:0>	
P2_PDE		—	—	—	—	—	—	P2_PDE<1:0>	
P0_ODE		P0_ODE<7:0>							
P1_ODE	97H	P1_ODE<7:0>							
P2_ODE		—	—	—	—	—	—	P2_ODE<1:0>	
P0_FUNL	9AH	P03_FUN<1:0>		P02_FUN<1:0>		P01_FUN<1:0>		P00_FUN<1:0>	
P0_FUNH	9BH	P07_FUN<1:0>		P06_FUN<1:0>		P05_FUN<1:0>		P04_FUN<1:0>	
P1_FUNL	9CH	P13_FUN<1:0>		P12_FUN<1:0>		P11_FUN<1:0>		P10_FUN<1:0>	
P1_FUNH	9DH	P17_FUN<1:0>		P16_FUN<1:0>		P15_FUN<1:0>		P14_FUN<1:0>	
P2_FUNL	9EH	—	—	—	—	P21_FUN<1:0>		P20_FUN<1:0>	
EINTn_IE	9FH	EINT7_IE	EINT6_IE	EINT5_IE	EINT4_IE	EINT3_IE	EINT2_IE	EINT1_IE	EINT0_IE
EINTn_IF	98H	EINT7_IF	EINT6_IF	EINT5_IF	EINT4_IF	EINT3_IF	EINT2_IF	EINT1_IF	EINT0_IF
EINT_MSL	A1H	EINT3_MS<1:0>		EINT2_MS<1:0>		EINT1_MS<1:0>		EINT0_MS<1:0>	
EINT_MSH	A2H	EINT7_MS<1:0>		EINT6_MS<1:0>		EINT5_MS<1:0>		EINT4_MS<1:0>	
EINT_IOS	A3H	EINT_IOS<7:0>							
TMRn_IE	BFH	—	—	—	—	TMR3_IE	TMR2_IE	TMR1_IE	TMR0_IE
TMRn_IF	A8H	—	—	—	—	TMR3_IF	TMR2_IF	TMR1_IF	TMR0_IF
PWMn_IE	D9H	—	STP2_IE	STP1_IE	STP0_IE	—	PWM2_IE	PWM1_IE	PWM0_IE
CAPn_IE		—	—	—	—	—	CAP2_IE	CAP1_IE	CAP0_IE
PWMn_IF	B0H	—	STP2_IF	STP1_IF	STP0_IF	—	PWM2_IF	PWM1_IF	PWM0_IF
CAPn_IF		—	—	—	—	—	CAP2_IF	CAP1_IF	CAP0_IF
UARTn_IE	E7H	—	—	—	—	—	—	TX_IE	RX_IE
IICn_IE		—	—	—	IIC_IE	—	—	—	—
SPIn_IE		—	SPI_IE	—	—	—	—	—	—
UARTn_IF	B8H	—	—	—	—	—	—	TX_IF	RX_IF
IICn_IF		—	—	—	IIC_IF	—	—	—	—

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0	
SPI _n _IF		—	SPI_IF	—	—	—	—	—	—	
AN_IE	FFH	—	CCM_IE	—	LVD_IE	—	ADC_IE	—	TK_IE	
AN_IF	C0H	—	CCM_IF	—	LVD_IF	—	ADC_IF	—	TK_IF	
HRC_TRML	89H	HRC_TRM<7:0>								
HRC_TRMH	8AH	HRC_PR<3:0>				—	—	HRC_TRM<9:8>		
TMR0_C0	A4H	TMR0_EN	TMR0_PRE	TMR0_CKS<1:0>		TMR0_PST<3:0>				
TMR0_C1	A5H	—	—	—	—	—	—	TMR0_GS	TMR0_GE	
TMR0L	A6H	TMR0<7:0>								
TMR0H	A7H	TMR0<15:8>								
TMR0_CYL	A9H	TMR0_CY<7:0>								
TMR0_CYH	AAH	TMR0_CY<15:8>								
TMR1_C0	ABH	TMR1_EN	TMR1_PRE	TMR1_CKS<1:0>		TMR1_PST<3:0>				
TMR1_C1	ACH	—	—	—	—	—	—	TMR1_GS	TMR1_GE	
TMR1L	ADH	TMR1<7:0>								
TMR1H	AEH	TMR1<15:8>								
TMR1_CYL	AFH	TMR1_CY<7:0>								
TMR1_CYH	B1H	TMR1_CY<15:8>								
TMR2_C0	B2H	TMR2_EN	TMR2_PRE	TMR2_CKS<1:0>		TMR2_PST<3:0>				
TMR2_C1	B3H	—	—	—	—	—	—	TMR2_GS	TMR2_GE	
TMR2L	B4H	TMR2<7:0>								
TMR2H	B5H	TMR2<15:8>								
TMR2_CYL	B6H	TMR2_CY<7:0>								
TMR2_CYH	B7H	TMR2_CY<15:8>								
TMR3_C0	B9H	TMR3_EN	TMR3_PRE	TMR3_CKS<1:0>		TMR3_PST<3:0>				
TMR3_C1	BAH	—	—	—	—	—	—	TMR3_GS	TMR3_GE	
TMR3L	BBH	TMR3<7:0>								
TMR3H	BCH	TMR3<15:8>								

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
TMR3_CYL	BDH	TMR3_CY<7:0>							
TMR3_CYH	BEH	TMR3_CY<15:8>							
PWM0_C	C1H	PWM0_SPS<1:0>		—	—	PWM0_TBS<1:0>		PWM0_MOD<1:0>	
CAPO_C		CAPO_EGS<1:0>		CAPO_CNT<1:0>		CAPO_TBS<1:0>		CAPO_MOD<1:0>	
PWM0_OC	C2H	PWM01_D	PWM00_D	PWM01_T	PWM00_T	PWM01_N	PWM01_P	PWM00_N	PWM00_P
PWM0_R0L	C3H	PWM0_R0<7:0>							
CAPO_DL		CAPO_D<7:0>							
PWM0_R0H	C4H	PWM0_R0<15:8>							
CAPO_DH		CAPO_D<15:8>							
PWM0_R1L	C5H	PWM0_R1<7:0>							
PWM0_R1H	C6H	PWM0_R1<15:8>							
PWM0_DT	C7H	PWM0_DT<7:0>							
PWM1_C	C9H	PWM1_SPS<1:0>		—	—	PWM1_TBS<1:0>		PWM1_MOD<1:0>	
CAP1_C		CAPO_EGS<1:0>		CAPO_CNT<1:0>		CAPO_TBS<1:0>		CAPO_MOD<1:0>	
PWM1_OC	CAH	PWM11_D	PWM10_D	PWM11_T	PWM10_T	PWM11_N	PWM11_P	PWM10_N	PWM10_P
PWM1_R0L	CBH	PWM1_R0<7:0>							
CAP1_DL		CAPO_D<7:0>							
PWM1_R0H	CCH	PWM1_R0<15:8>							
CAP1_DH		CAPO_D<15:8>							
PWM1_R1L	CDH	PWM1_R1<7:0>							
PWM1_R1H	CEH	PWM1_R1<15:8>							
PWM1_DT	CFH	PWM1_DT<7:0>							
PWM2_C	D1H	PWM2_SPS<1:0>		—	—	PWM2_TBS<1:0>		PWM2_MOD<1:0>	
CAP2_C		CAPO_EGS<1:0>		CAPO_CNT<1:0>		CAPO_TBS<1:0>		CAPO_MOD<1:0>	
PWM2_OC	D2H	PWM21_D	PWM20_D	PWM21_T	PWM20_T	PWM21_N	PWM21_P	PWM20_N	PWM20_P
PWM2_R0L	D3H	PWM2_R0<7:0>							
CAP2_DL		CAPO_D<7:0>							

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
PWM2_R0H	D4H	PWM2_R0<15:8>							
CAP2_DH		CAP0_D<15:8>							
PWM2_R1L	D5H	PWM2_R1<7:0>							
PWM2_R1H	D6H	PWM2_R1<15:8>							
PWM2_DT	D7H	PWM2_DT<7:0>							
UART_BRL	DAH	UART_BRR<7:0>							
UART_BRH	DBH	UART_BRR<15:8>							
UART_RXC	DCH	FERR	OERR	PERR	RX9D	PARS	BRFX	RXEN<1:0>	
UART_RXB	DDH	RXB<7:0>							
UART_TXC	DEH	TXST	TXBF	TX9S<1:0>		STPS	TXDM	TXEN<1:0>	
UART_TXB	DFH	TXB<7:0>							
UART_TXM	8EH	TXM<1:0>		—	—	—	—	—	—
IIC_C0	E1H	IIC_SMPF<3:0>				IIC_IM	WTEN	IIC_MD	IIC_EN
IIC_C1	E2H	—	—	—	MTAI_MK	STOP	START	WAIT	ACK
IIC_STA	E3H	SLV_ADF	SLV_RWF	STOP_F	START_F	ACK_F	BUF_ST	OVT_F	BFOV_F
IIC_ADDR	E4H	—	IIC_ADDR<6:0>						
IIC_DATA	E5H	IIC_DATA<7:0>							
IIC_BRR	E4H	—	IIC_BRR<6:0>						
SPI_C0	86H	SPI_SSC	SPI_SSD	SPI_CPOL	SPI_CPHA	SPI_DIR	SPI_CLK		
SPI_C1	8BH	SPI_EN	SPI_MS	—		SPI_TXSP	SPI_RXSP	SPI_WOL	SPI_RXSP
SPI_DATA	8CH	SPI_DATA							
LVD_C0	ECH	LVD_EN	LVD_IM	LVD_FLT<1:0>		LVD_CKS	LVD_INS	LVD_VS<1:0>	
ADC_C0	FAH	ADC_EN	S_TRG	TRG_S<1:0>		CH_SEL<3:0>			
ADC_C1	FBH	VREFP_EN	VREFP_S<2:0>			AVDD_S	ADC_DM	ADC_CKS<1:0>	
ADC_C2	FCH	—	—	ADC_CNT<1:0>		—	SMP_T<2:0>		
ADC_DL	FDH	ADC_DL<7:0>							
ADC_DH	FEH	ADC_DH<7:0>							

SFR	ADDR	BIT7	BIT6	BIT5	BIT4	BIT3	BIT2	BIT1	BIT0
ADC_CMP_C	F7H	ADC_CP_E	MAX_E	MIN_E	INT_MD	MAX_F	MIN_F	ADC_DMD	
VREF_C	F6H	VREF_C<7:0>							
TK_VREF_C	EDH	TK_VREF_C<7:0>							
TK_C0	EFH	TK_EN	TK_VTKS	—	—	TK_OV	TK_ERR	—	TK_GO
TK_C1	EEH	TK_FLT<3:0>				—	TK_VRFS<2:0>		
TK_C2	F1H	TK_DIV<5:0>						TK_CKS<1:0>	
TK_CHEL	F2H	TK_CHE<7:0>							
TK_CHEH	F3H	TK_CHE<15:8>							
TK_CNTL	F4H	TK_CNT<7:0>							
TK_CNTH	F5H	TK_CNT<15:8>							

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_AS	9001H	IAP_AS<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	IAP_OPS<7:0>							
CLK_PR	9020H	CLK_PR<7:0>							
CLK_C0	9021H	—	—	CLK_S<1:0>		—	CLK_DIV<2:0>		
CLK_C1	9022H	OSC_F<1:0>		XOSC_ST	HRC_ST	—	—	XOSC_EN	HRC_EN
PCK_GTC0	9023H	—	CP2_G	CP1_G	CP0_G	TMR3_G	TMR2_G	TMR1_G	TMR0_G
PCK_GTC1	9024H	—	TK_G	—	ADC_G	IIC_G	—	—	UART_G
CCM_C	9025H	CCM_EN	—	—	—	—	CCM_CKS<2:0>		
CLK_LP	9026H	—	—	—	CLK_LEN	—	—	—	FLT_BPS
WKUP_T	9027H	WKUP_T<7:0>							

位操作映射

寄存器		F/7	E/6	D/5	C/4	B/3	A/2	9/1	8/0
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>							
EINTn_IF	98H	EINT7_IF	EINT6_IF	EINT5_IF	EINT4_IF	EINT3_IF	EINT2_IF	EINT1_IF	EINT0_IF
P2_DAT	A0H	—	—	—	—	—	—	P2_n<1:0>	
TMRn_IF	A8H	—	—	—	—	TMR3_IF	TMR2_IF	TMR1_IF	TMR0_IF
PWMn_IF	B0H	—	STP2_IF	STP1_IF	STP0_IF	—	PWM2_IF	PWM1_IF	PWM0_IF
CAPn_IF		—	—	—	—	—	CAP2_IF	CAP1_IF	CAP0_IF
UARTn_IF	B8H	—	—	—	—	—	—	TX_IF	RX_IF
IICn_IF		—	—	—	IIC_IF	—	—	—	—
SPIn_IF		—	SPI_IF	—	—	—	—	—	—
AN_IF	C0H	—	CCM_IF	—	LVD_IF	—	ADC_IF	—	TK_IF
RST_FLAG	C8H	—	—	—	ROM_OV	RSTN_F	SRST_F	WDTR_F	POR_F
PSW	D0H	CY	AC	F0	RS<1:0>		OV	—	P
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

7 复位

7.1 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_WD2.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 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。

7.2 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

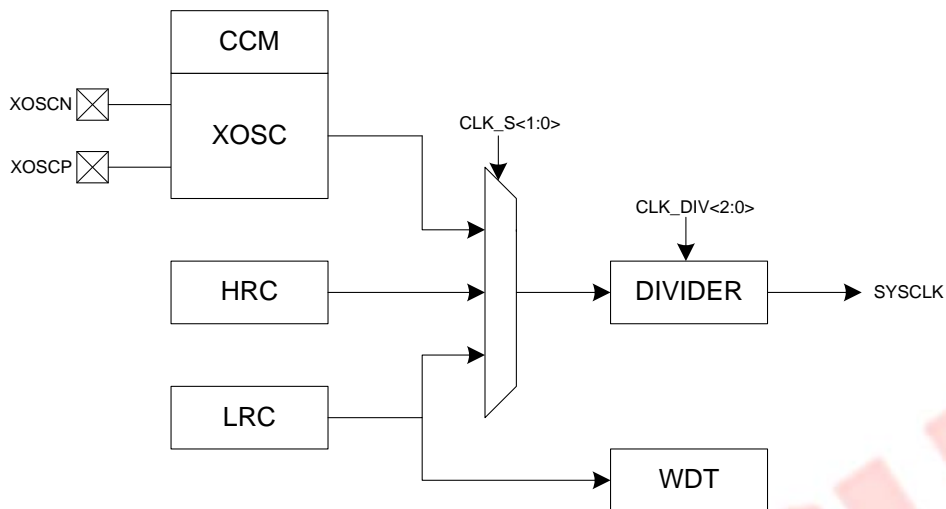
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
W-0000_0000							
SOFT_RST<7:0>							

Bit7-0 **SOFT_RST<7:0>**: 软件复位字

W: 写 A5H 产生软件复位

8 时钟

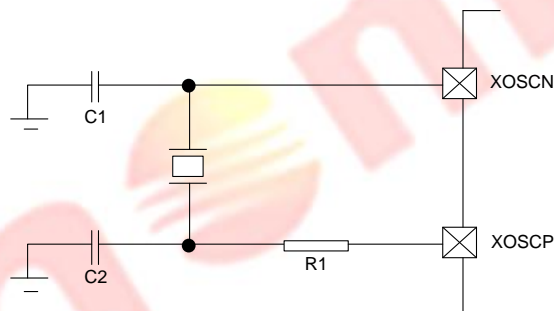
8.1 描述



图表 9 系统时钟源功能框图

外部振荡器 XOSC

外部振荡器 XOSC 适用于外接 2~20MHz 高频晶振。



图表 10 外部振荡器 XOSC 连接示意图

外部振荡器停振检测 CCM

芯片内置外部振荡器停振检测模块 CCM，用于监控外部振荡器的工作状态。当系统时钟源采用 XOSC 时，CCM 如果使能，CCM 会实时监控 XOSC 工作状态，当 XOSC 发生异常停振时，CCM 会产生停振异常中断标志 `AN_IF.CCM_IF`，同时自动将系统时钟切换为内部低频时钟 LRC。

内部高频 RC 时钟 HRC

芯片内置 16MHz 高频 RC 时钟，用于系统和外设时钟源。

HRC 出厂校准精度 $\pm 1\%$ ($T=25^{\circ}\text{C}$)。

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。LRC 时钟硬件强制使能，用户无法将其关闭。

8.2 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-000		
—	—	CLK_S<1:0>		—	CLK_DIV<2:0>		

Bit7-6 保留
 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 **CP2_G**: CAP2 和 PWM2 外设时钟使能位
 - 1: 使能
 - 0: 关闭
- Bit5 **CP1_G**: CAP1 和 PWM1 外设时钟使能位
 - 1: 使能
 - 0: 关闭
- Bit4 **CP0_G**: CAP0 和 PWM0 外设时钟使能位
 - 1: 使能
 - 0: 关闭
- Bit3 **TMR3_G**: TMR3 外设时钟使能位
 - 1: 使能
 - 0: 关闭
- Bit2 **TMR2_G**: TMR2 外设时钟使能位
 - 1: 使能
 - 0: 关闭
- Bit1 **TMR1_G**: TMR1 外设时钟使能位
 - 1: 使能
 - 0: 关闭
- Bit0 **TMR0_G**: TMR0 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
—	TK_G	—	ADC_G	IIC_G	—	—	UART_G

- Bit7 保留
- 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
- Bit2-1 保留
- Bit0 **UART_G**: UART 外设时钟使能位
1: 使能
0: 关闭

CCM_C 停振检测控制寄存器

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

- Bit7 **CCM_EN**: XOSC 停振检测使能位
1: 使能
0: 关闭
- Bit6-3 保留
- Bit2-0 **CCM_CKS**<2:0>: 检测器时钟频率选择位
111~100: 32KHz
011: 1MHz
010: 4MHz
001: 8MHz
000: 16MHz

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_1111							
WKUP_T <7:0>							

Bit7-0 **WKUP_T**<7:0>: SLEEP 唤醒等待时间控制位

唤醒等待时间 $T_{WKUP} = (WKUP_T<7:0> + 1) \times 4 \times T_{SYS_CLK}$

HRC_TRML HRC 调校寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-xxxx_xxxx							
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-xx	
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: 锁定状态读数为 0H, 解锁状态读数为 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。

9 低功耗

9.1 描述

芯片支持 2 种低功耗模式，IDLE 模式和 SLEEP 模式。

IDLE 模式

在 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_GTC0/1 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。IDLE 唤醒无等待时间，唤醒后 CPU 继续运行。

SLEEP 模式

在 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。用户还可通过 CLK_LP.CLK_LEN 关闭系统时钟源，以进一步节省功耗。

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。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。

9.2 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）

10 看门狗定时器 WDT

10.1 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

当 16 位 WDT 计数器累加到与 **WDT_C.WDT_CY<2:0>**位所选择的溢出值相等时，WDT 计数器溢出。溢出后 WDT 重新从 0 开始累加。WDT 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。

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。

WDT 复位

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 WDT 复位标志 **RST_FG.WDTF** 置 1。

10.2 WDT 寄存器

WDT_C WDT 控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	RW-0	RW-0100			
—	—	—	WDT_CKS	WDT_CY<3:0>			

Bit7-5 保留

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_CY<3:0>**: WDT 计数溢出值选择位

1100: 000FH 1011: 001FH 1010: 003FH 1001: 007FH

1000: 00FFH 0111: 01FFH 0110: 03FFH 0101: 07FFH

0100: 0FFFH 0011: 1FFFH 0010: 3FFFH 0001: 7FFFH

0000: FFFFH 其他: 0FFFH

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

11 中断

11.1 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	EINTn_IE(n=3:0)	EINTn_IF(n=3:0)
INT1	2	0013H	INT1_IE	INT1_IF	EINTn_IE(n=7:4)	EINTn_IF(n=7:4)
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	TX_IE	TX_IF
					RX_IE	RX_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
					CCM_IE	CCM_IF

图表 31 中断向量表

11.2 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。

11.3 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

11.4 外部端口中断 EINT

芯片支持 8 个外部端口中断源 EINT0~7，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。

EINTn_IE EINT 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
EINT7_IE	EINT6_IE	EINT5_IE	EINT4_IE	EINT3_IE	EINT2_IE	EINT1_IE	EINT0_IE

Bit7-0 **EINTn_IE**: 外部端口中断 EINT0~7 使能位

- 1: 使能
- 0: 禁止

EINTn_IF EINT 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	RW-0
EINT7_IF	EINT6_IF	EINT5_IF	EINT4_IF	EINT3_IF	EINT2_IF	EINT1_IF	EINT0_IF

Bit7-0 **EINTn_IF**: 外部端口中断 EINT0~7 标志位

- 1: 有 EINTn 中断请求
- 0: 无 EINTn 中断请求

EINT_MSL EINT 触发选择寄存器低 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
EINT3_MS<1:0>		EINT2_MS<1:0>		EINT1_MS<1:0>		EINT0_MS<1:0>	

EINT_MSH EINT 触发选择寄存器高 8 位

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
EINT7_MS<1:0>		EINT6_MS<1:0>		EINT5_MS<1:0>		EINT4_MS<1:0>	

EINTn_MS: EINTn 触发方式选择位

- 11: 低电平触发
- 10: 双边沿触发 (上升和下降沿都触发)
- 01: 下降沿触发
- 00: 上升沿触发

EINT_IOS EINT 端口选择寄存器

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

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

EINT_IOS	EINT7	EINT6	EINT5	EINT4	EINT3	EINT2	EINT1	EINT0
1	P21	P20	P17	P16	P15	P14	P13	P12
0	P11	P10	P07	P06	P05	P04	P02	P01

11.5 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**<3:0>: 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

11.6 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

11.7 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

11.8 异步收发器 UART 中断

UART 支持 2 个中断，即 TX 发送中断和 RX 接收中断。

TX 发送中断

根据 **UART_TXC.TXEN**<1:0>位的配置，TX 发送中断可以是 UART 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

UART 接收寄存器接收到 1 帧数据，即产生 RX 接收中断。

UARTn_IE **UART 中断使能寄存器** (与 **IICn_IE** 寄存器复用)

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

Bit7-2 保留

Bit1 **TX_IE**: UART 发送中断使能位
 1: 使能
 0: 禁止

Bit0 **RX_IE**: UART 接收中断使能位
 1: 使能
 0: 禁止

UARTn_IF **UART 中断标志寄存器** (与 **IICn_IF** 寄存器复用)

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

Bit7-2 保留

Bit1 **TX_IF**: UART 发送中断标志位
 1: 有 UART 发送中断请求
 0: 无 UART 发送中断请求

Bit0 **RX_IF**: UART 接收中断标志位
 1: 有 UART 接收中断请求
 0: 无 UART 接收中断请求

11.9 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。

IICn_IE IIC 中断使能寄存器 (与 UARTn_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

IICn_IF IIC 中断标志寄存器 (与 UARTn_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 IIC 中断请求

0: 无 IIC 中断请求

Bit3-0 保留

11.10 SPI 模块中断

SPI 支持 1 个中断。一个数据传输完成产生一个中断。

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

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

Bit7 保留

Bit6 **SPI_BIE**: SPI 中断使能位

1: 使能

0: 禁止

Bit5-0 保留

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

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

Bit7 保留

Bit6 **SPI_BIF**: SPI 中断标志位

1: 有 SPI 中断请求

0: 无 SPI 中断请求

Bit5-0 保留

11.11 模拟模块中断

模拟模块包括 CCM、LVD、ADC、TK，每个模块都有 1 组独立的中断使能位和中断标志位。

XOSC 停振检测 CCM 中断

检测到 XOSC 异常停振时，产生 CCM 中断。

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
—	CCM_IE	—	LVD_IE	—	ADC_IE	—	TK_IE

Bit7 保留

Bit6 **CCM_IE**: CCM 中断使能位

1: 使能

0: 禁止

Bit5 保留

Bit4 **LVD_IE**: LVD 中断使能位

1: 使能

0: 禁止

Bit3 保留

Bit2 **ADC_IE**: ADC 中断使能位

1: 使能

0: 禁止

Bit1 保留

Bit0 **TK_IE**: TK 中断使能位

1: 使能

0: 禁止

AN_IF 模拟中断标志寄存器

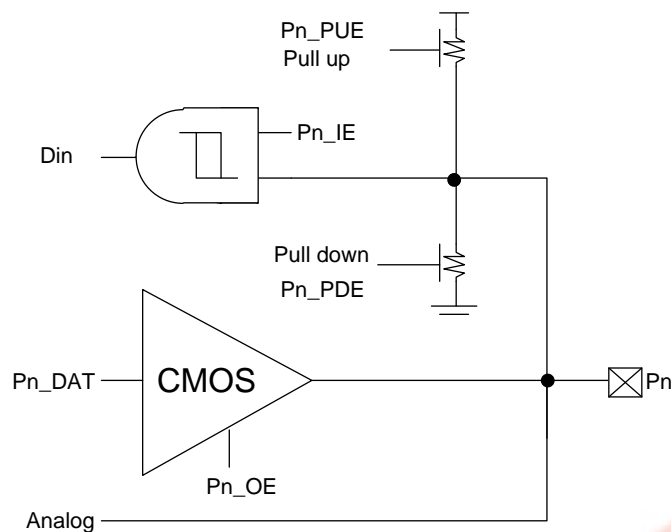
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	RW-0	—	RW-0	—	RW-0	—	RW-0
—	CCM_IF	—	LVD_IF	—	ADC_IF	—	TK_IF

- Bit7 保留
- Bit6 **CCM_IF**: CCM 中断标志位
 - 1: 有 CCM 中断请求
 - 0: 无 CCM 中断请求
- Bit5 保留
- Bit4 **LVD_IF**: LVD 中断标志位
 - 1: 有 LVD 中断请求
 - 0: 无 LVD 中断请求
- Bit3 保留
- Bit2 **ADC_IF**: ADC 中断标志位
 - 1: 有 ADC 中断请求
 - 0: 无 ADC 中断请求
- Bit1 保留
- Bit0 **TK_IF**: TK 中断标志位
 - 1: 有 TK 中断请求
 - 0: 无 TK 中断请求



12 I/O 端口

12.1 I/O 功能框图



图表 42 I/O 功能框图

12.2 I/O 寄存器

PORT_C0 端口控制寄存器 (与 UART_TXM 寄存器复用)

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

Bit7-6 保留

Bit5-4 **PT_ODS<1:0>**: 开漏控制选择位

11: 选择 P0 开漏控制

10: 选择 P2 开漏控制

01: 选择 P1 开漏控制

00: 选择 P0 开漏控制

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, 输入模式读端口电平

Bit1 **PT_PLS**: 端口上/下拉控制选择位

1: 选择端口下拉控制

0: 选择端口上拉控制

Bit0 **PT_IOS**: 端口输入/输出控制选择位

1: 选择 Px_IE 寄存器

0: 选择 Px_OE 寄存器

P0_IE **P0 端口输入使能寄存器** (与 P0_OE 寄存器复用, 通过 PORT_C0.PT_IOS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-1111_1111							
P0_IE<7:0>							

Bit7-0 **P0_IE<7:0>**: P0 端口输入使能位

- 1: 输入使能
- 0: 输入禁止

P1_IE **P1 端口输入使能寄存器** (与 P1_OE 寄存器复用, 通过 PORT_C0.PT_IOS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-1111_1111							
P1_IE<7:0>							

Bit7-0 **P1_IE<7:0>**: P1 端口输入使能位

- 1: 输入使能
- 0: 输入禁止

P2_IE **P2 端口输入使能寄存器** (与 P2_OE 寄存器复用, 通过 PORT_C0.PT_IOS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-11	
—	—	—	—	—	—	P2_IE<1:0>	

Bit7-2 保留

Bit1-0 **P2_IE<1:0>**: P2 端口输入使能位

- 1: 输入使能
- 0: 输入禁止

P0_OE **P0 端口输出使能寄存器** (与 P0_IE 寄存器复用, 通过 PORT_C0.PT_IOS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P0_OE<7:0>							

Bit7-0 **P0_OE<7:0>**: P0 端口输出使能位

- 1: 输出使能
- 0: 输出禁止

P1_OE **P1 端口输出使能寄存器** (与 P1_IE 寄存器复用, 通过 PORT_C0.PT_IOS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P1_OE<7:0>							

Bit7-0 **P1_OE<7:0>**: P1 端口输出使能位

- 1: 输出使能
- 0: 输出禁止

P2_OE **P2 端口输出使能寄存器** (与 P2_IE 寄存器复用, 通过 PORT_C0.PT_IOS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-00	
—	—	—	—	—	—	P2_OE<1:0>	

Bit7-2 保留

Bit1-0 **P2_OE<1:0>**: P2 端口输出使能位

- 1: 输出使能
- 0: 输出禁止

P0_DAT P0 端口数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P0_n<7:0>							

Bit7-0 **P0_n<7:0>**: P0 端口输出位
 1: 端口输出高电平
 0: 端口输出低电平

P1_DAT P1 端口数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P1_n<7:0>							

Bit7-0 **P1_n<7:0>**: P1 端口输出位
 1: 端口输出高电平
 0: 端口输出低电平

P2_DAT P2 端口数据寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-00	
—	—	—	—	—	—	P2_n<1:0>	

Bit7-2 保留
 Bit1-0 **P2_n<1:0>**: P2 端口输出位
 1: 端口输出高电平
 0: 端口输出低电平

P0_PUE P0 端口弱上拉使能寄存器 (与 P0_PDE 寄存器复用, 通过 PORT_C0.PT_PLS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P0_PUE<7:0>							

Bit7-0 **P0_PUE<7:0>**: P0 端口弱上拉使能位
 1: 弱上拉使能
 0: 弱上拉关闭

P1_PUE P1 端口弱上拉使能寄存器 (与 P1_PDE 寄存器复用, 通过 PORT_C0.PT_PLS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P1_PUE<7:0>							

Bit7-0 **P1_PUE<7:0>**: P1 端口弱上拉使能位
 1: 弱上拉使能
 0: 弱上拉关闭

P2_PUE P2 端口弱上拉使能寄存器 (与 P2_PDE 寄存器复用, 通过 PORT_C0.PT_PLS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-00	
							P2_PUE<1:0>

Bit7-2 保留

Bit1-0 **P2_PUE<1:0>**: P2 端口弱上拉使能位

1: 弱上拉使能

0: 弱上拉关闭

P0_PDE P0 端口弱下拉使能寄存器 (与 P0_PUE 寄存器复用, 通过 PORT_C0.PT_PLS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P0_PDE<7:0>							

Bit7-0 **P0_PDE<7:0>**: P0 端口弱下拉使能位

1: 弱下拉使能

0: 弱下拉关闭

P1_PDE P1 端口弱下拉使能寄存器 (与 P1_PUE 寄存器复用, 通过 PORT_C0.PT_PLS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000_0000							
P1_PDE<7:0>							

Bit7-0 **P1_PDE<7:0>**: P1 端口弱上拉使能位

1: 弱下拉使能

0: 弱下拉关闭

P2_PDE P2 端口弱下拉使能寄存器 (与 P2_PUE 寄存器复用, 通过 PORT_C0.PT_PLS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-00	
							P2_PDE<1:0>

Bit7-2 保留

Bit1-0 **P2_PDE<1:0>**: P2 端口弱上拉使能位

1: 弱下拉使能

0: 弱下拉关闭

P0_ODE P0 端口开漏使能寄存器 (与 P1_ODE、P2_ODE 寄存器复用, 通过 PORT_C0.PT_ODS 位选择)

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

Bit7-0 **P0_ODE <7:0>**: P0 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

P1_ODE **P1 端口开漏使能寄存器** (与 P0_ODE、P2_ODE 寄存器复用, 通过 PORT_C0.PT_ODS 位选择)

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

Bit7-0 **P1_ODE <7:0>**: P1 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

P2_ODE **P2 端口开漏使能寄存器** (与 P0_ODE、P1_ODE 寄存器复用, 通过 PORT_C0.PT_ODS 位选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	—	—	RW-00	
							P2_ODE<1:0>

Bit7-0 **P2_ODE <1:0>**: P0 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

P0_FUNL **P0 端口复用选择寄存器低 8 位** (P0<3: 0>功能选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
P03_FUN<1:0>		P02_FUN<1:0>		P01_FUN<1:0>		P00_FUN<1:0>	

Bit7-6 **P03_FUN<1:0>**: P03 端口复用选择位 (参见端口复用表)
 11: IIC 时钟线 SCL
 10: UART 接收端 RX
 01: TMR1 外部时钟输入 TIN1
 00: 通用 I/O P03

Bit5-4 **P02_FUN<1:0>**: P02 端口复用选择位 (参见端口复用表)
 11: IIC 数据线 SDA
 10: UART 发送端 TX
 01: PWM2 通道 1 输出 PWM21
 00: 通用 I/O (大电流输出) P02

Bit3-2 **P01_FUN<1:0>**: P01 端口复用选择位 (参见端口复用表)
 11: IIC 时钟线 SCL
 10: UART 接收端 RX
 01: PWM2 通道 0 输出 PWM20
 00: 通用 I/O (大电流输出) P01

Bit1-0 **P00_FUN<1:0>**: P00 端口复用选择位 (参见端口复用表)
 11: 通用 I/O P00
 10: PWM0 刹车输入 STP0
 01: 通用 I/O P00
 00: 通用 I/O P00

P0_FUNH P0 端口复用选择寄存器高 8 位 (P0<7: 4>功能选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
P07_FUN<1:0>		P06_FUN<1:0>		P05_FUN<1:0>		P04_FUN<1:0>	

- Bit7-6 **P07_FUN<1:0>**: P07 端口复用选择位 (参见端口复用表)
- 11: TMR2 门控输入 TGT2
 - 10: UART 接收端 RX
 - 01: TMR1 外部时钟输入 TIN1, PWM1 通道 0 输出 PWM00
 - 00: 通用 I/O (大电流输出) P07
- Bit5-4 **P06_FUN<1:0>**: P06 端口复用选择位 (参见端口复用表)
- 11: TMR1 门控输入 TGT1
 - 10: PWM1 刹车输入
 - 01: TMR0 外部时钟输入 TIN0, PWM0 通道 1 输出 PWM01
 - 00: 通用 I/O (大电流输出) P06
- Bit3-2 **P05_FUN<1:0>**: P05 端口复用选择位 (参见端口复用表)
- 11: TMR0 门控输入 TGT0
 - 10: 通用 I/O (大电流输出)
 - 01: CAP0 捕捉输入 CAP0, PWM0 通道 0 输出 PWM00
 - 00: 通用 I/O (大电流输出) P05
- Bit1-0 **P04_FUN<1:0>**: P04 端口复用选择位 (参见端口复用表)
- 11: IIC 数据线 SDA
 - 10: UART 发送端 TX
 - 01: TMR2 外部时钟输入 TIN2
 - 00: 通用 I/O P04

P1_FUNL P1 端口复用选择寄存器低 8 位 (P1<3: 0>功能选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
P13_FUN<1:0>		P12_FUN<1:0>		P11_FUN<1:0>		P10_FUN<1:0>	

- Bit7-6 **P13_FUN<1:0>**: P13 端口复用选择位 (参见端口复用表)
- 11: 通用 I/O P13
 - 10: PWM2 刹车输入 STP2
 - 01: TMR3 外部时钟输入 TIN3
 - 00: 通用 I/O P13
- Bit5-4 **P12_FUN<1:0>**: P12 端口复用选择位 (参见端口复用表)
- 11: 通用 I/O (大电流输出) P12
 - 10: 通用 I/O (大电流输出) P12
 - 01: PWM2 通道 1 输出 PWM21
 - 00: 通用 I/O (大电流输出) P12
- Bit3-2 **P11_FUN<1:0>**: P11 端口复用选择位 (参见端口复用表)
- 11: 所有 PWM 刹车输入 STP

10: 通用 I/O (大电流输出) P11

01: PWM2 通道 0 输出 PWM20

00: 通用 I/O (大电流输出) P11

Bit1-0 **P10_FUN<1:0>**: P10 端口复用选择位 (参见端口复用表)

11: TMR3 门控输入 TGT3

10: UART 发送端 TX

01: CAP1 捕捉输入 CAP1, PWM1 通道 1 输出 PWM11

00: 通用 I/O (大电流输出) P10

P1_FUNH P1 端口复用选择寄存器高 8 位 (P1<7: 4>功能选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00		RW-00		RW-00		RW-00	
P17_FUN<1:0>		P16_FUN<1:0>		P15_FUN<1:0>		P14_FUN<1:0>	

Bit7-6 **P17_FUN<1:0>**: P17 端口复用选择位 (参见端口复用表)

11: TMR0 门控输入 TGT0

10: SPI 主输入从输出 MISO

01: PWM1 通道 1 输出 PWM11

00: 通用 I/O (大电流输出) P17

Bit5-4 **P16_FUN<1:0>**: P16 端口复用选择位 (参见端口复用表)

11: TMR1 门控输入 TGT1

10: SPI 主输出从输入 MOSI

01: PWM1 通道 0 输出 PWM10

00: 通用 I/O (大电流输出) P16

Bit3-2 **P15_FUN<1:0>**: P15 端口复用选择位 (参见端口复用表)

11: TMR2 门控输入 TGT2

10: SPI 时钟信号 SCK

01: PWM0 通道 1 输出 PWM01

00: 通用 I/O P15

Bit1-0 **P14_FUN<1:0>**: P14 端口复用选择位 (参见端口复用表)

11: TMR3 门控输入 TGT3

10: SPI 片选信号 SS

01: PWM0 通道 0 输出 PWM00

00: 通用 I/O P14

P2_FUNL P2 端口复用选择寄存器 (P2<1: 0>功能选择)

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
—	—	—	—	RW-00		RW-00	
—	—	—	—	P21_FUN<1:0>		P20_FUN<1:0>	

Bit7-4 保留

Bit3-2 **P21_FUN<1:0>**: P21 端口复用选择位 (参见端口复用表)

11: IIC 时钟线 SCL

- 10: UART 发送端 TX
- 01: TMR2 外部时钟输入 TIN2, PWM0 通道 1 输出 PWM01
- 00: 通用 I/O P21

Bit1-0 **P20_FUN<1:0>**: P20 端口复用选择位 (参见端口复用表)

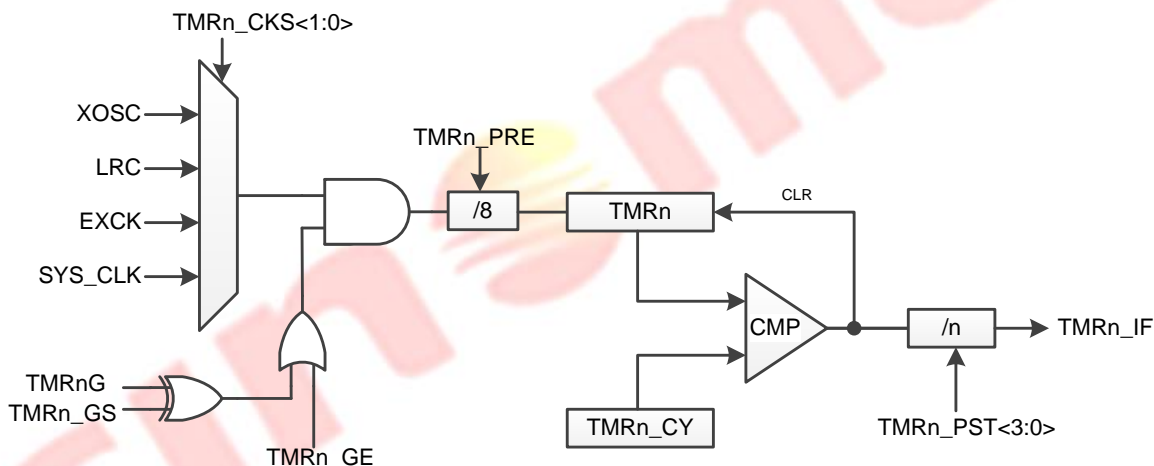
- 11: IIC 数据线 SDA
- 10: UART 接收端 RX
- 01: CAP2 捕捉输入 CAP2, PWM0 通道 0 输出 PWM00
- 00: 通用 I/O P20

定时/计数器 TMR

12.3 描述

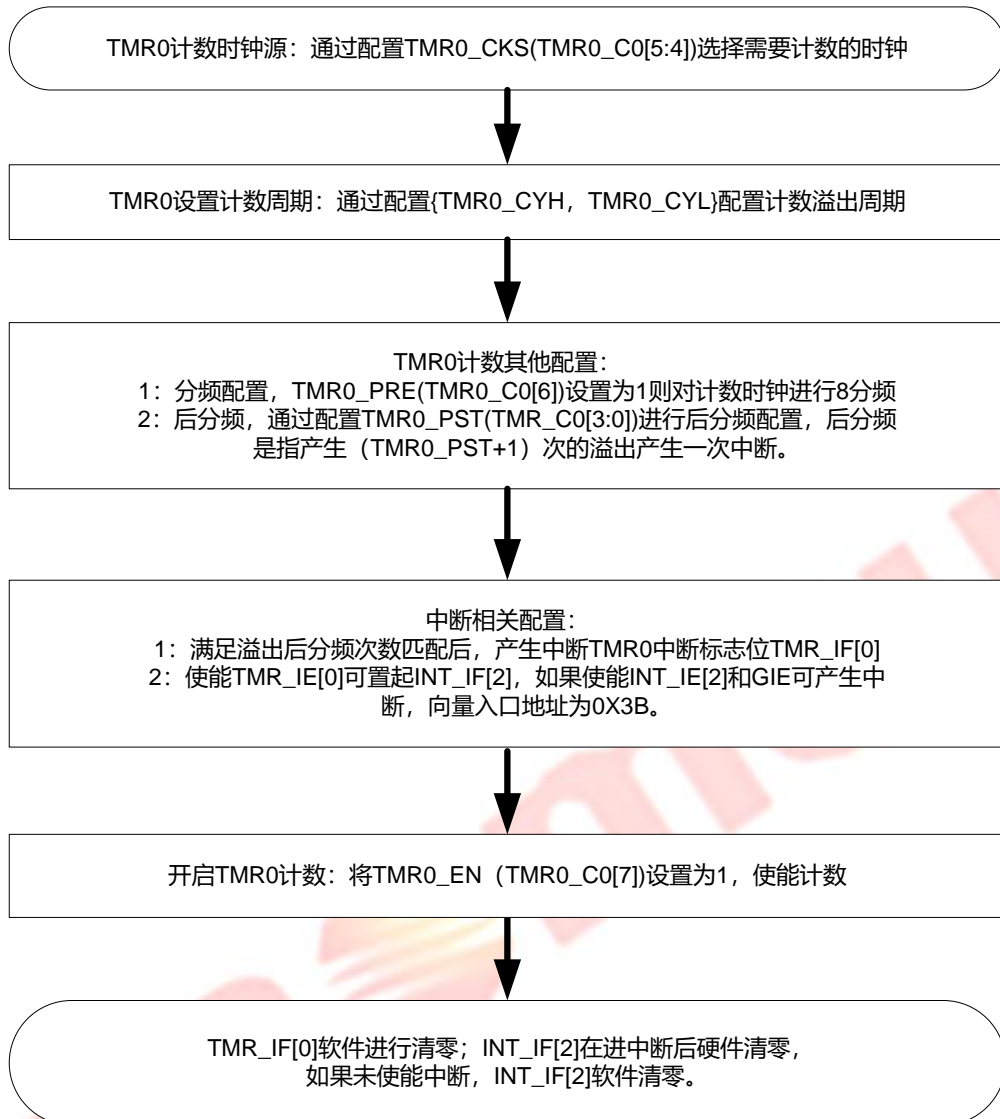
芯片内置 4 组 16 位定时/计数器 TMR0~TMR3。支持门控计数功能。

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 53 TMR 功能框图

TMR 操作流程 (以 TMR0 为例)



图表 64 TMR 操作流程图

12.4 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: XOSC 外部振荡器
10: LRC 内部低频时钟
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 ($n = \text{TMRn_POS}<3:0> + 1$)

TMRn_C1 TMRn 控制寄存器 1

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

- Bit7-2 保留
- Bit1 **TMRn_GS**: TMRn 门控极性选择位
1: TMRG 输入低电平计数
0: TMRG 输入高电平计数
- Bit0 **TMRn_GE**: TMRn 门控使能位
1: 使能
0: 关闭

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



13 边沿捕捉器 CAP

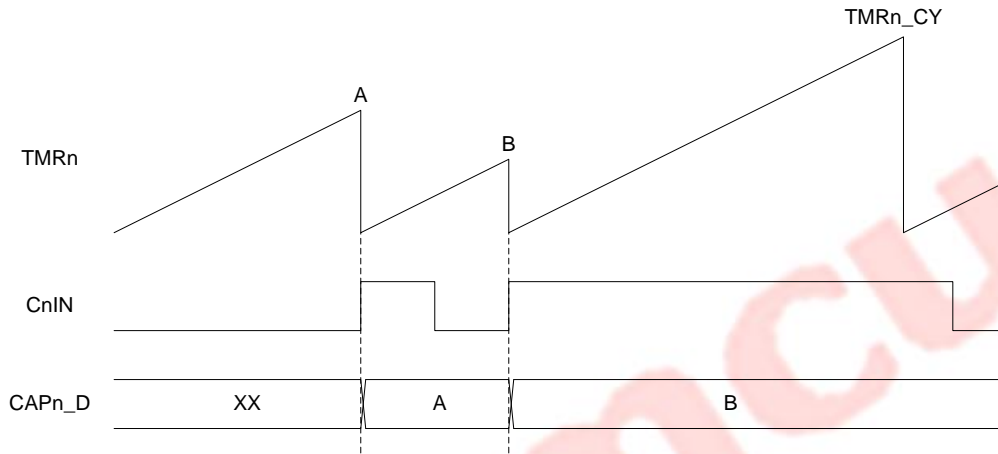
13.1 描述

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

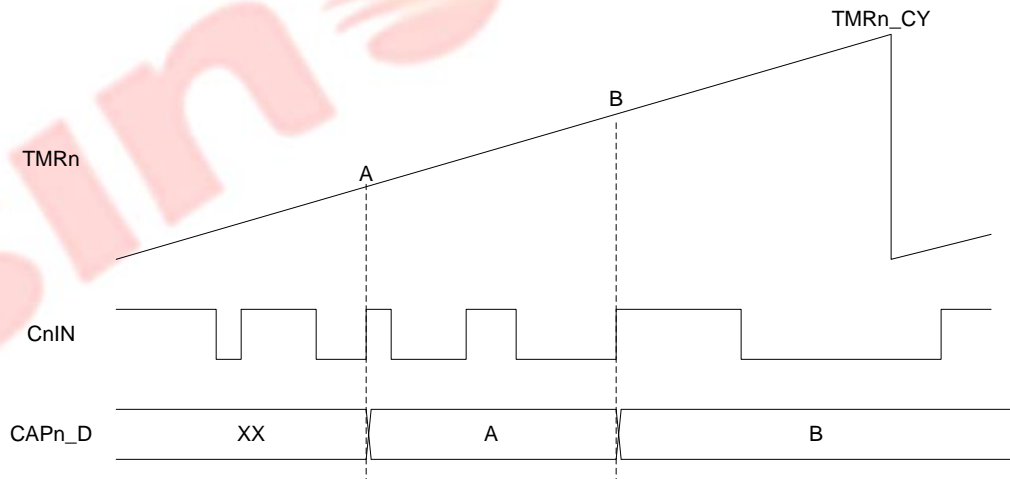
示例： $CAPn_MOD<1:0>=11$ ， $CAPn_EGS<1:0>=01$ ， $CAPn_CNT<1:0>=00$



图表 75 捕捉清零模式示例波形图

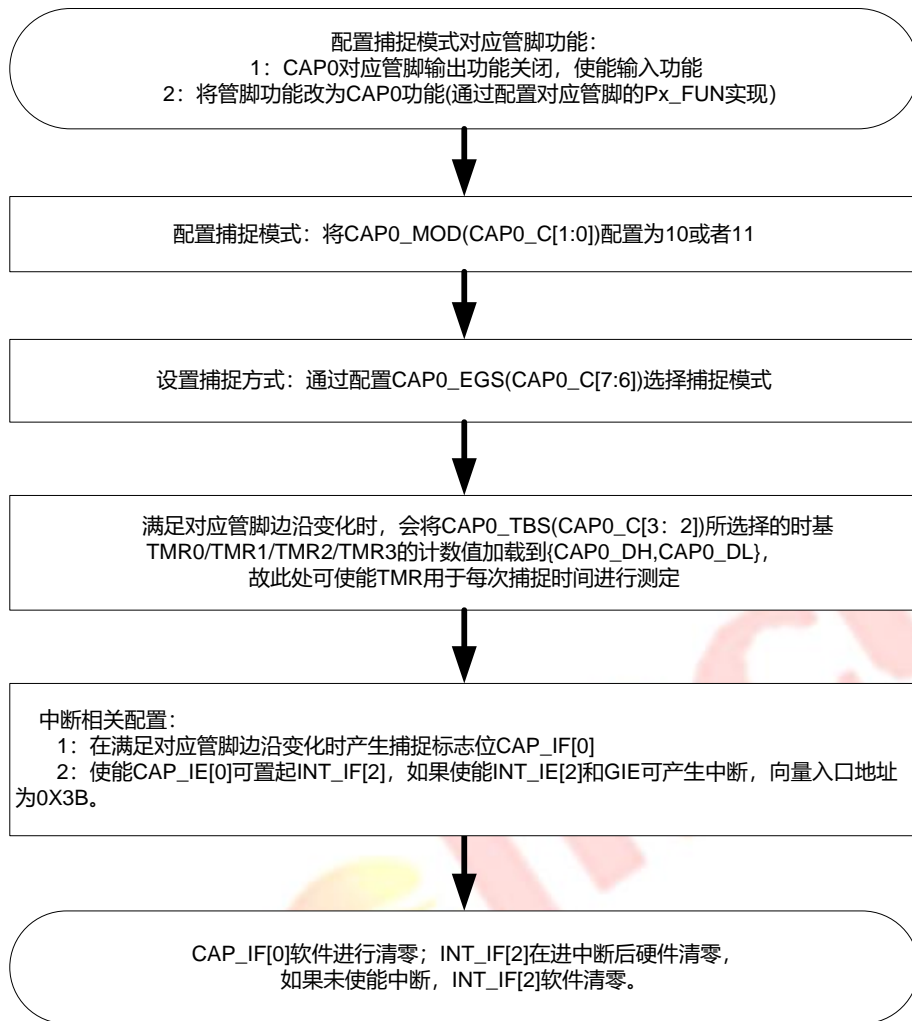
捕捉累加模式

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



图表 16 捕捉累加模式示例波形图

CAP 操作流程 (以 CAP0 为例)



图表 87 CAP 操作流程图

13.2 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

- 11: 上升和下降沿都捕捉
- 10: 下降沿捕捉
- 01: 上升沿捕捉
- 00: 保留

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

- 11: 选择 TMR3
- 10: 选择 TMR2
- 01: 选择 TMR1
- 00: 选择 TMR0

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

14 脉宽调制器 PWM

14.1 描述

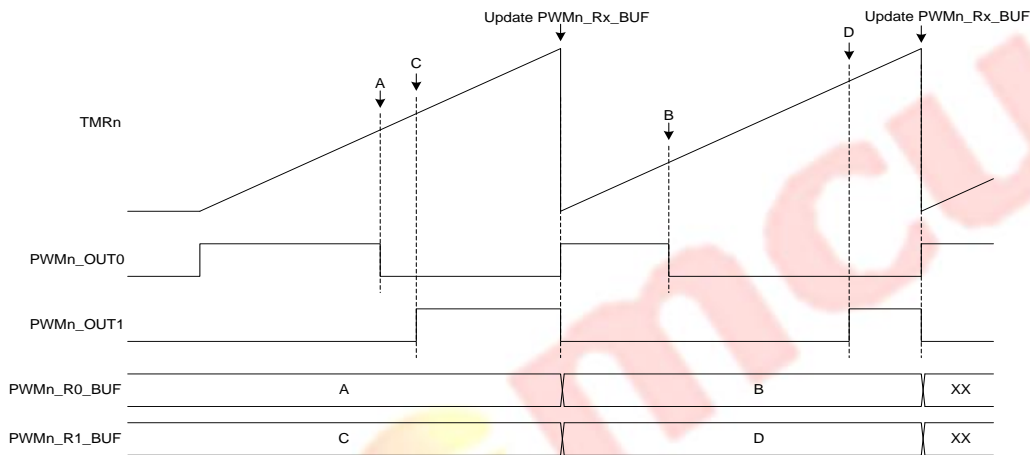
芯片内置 3 组脉宽调制器 PWM0~PWM2，每组支持 2 路 PWM 输出。

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： $PWMn_DT<7:0>=00H$

$PWMn0_T=0, PWMn0_P=1, PWMn0_N=0$

$PWMn1_T=0, PWMn0_P=0, PWMn0_N=1$



图表 18 PWM 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 2 组 16 位的匹配寄存器 PWMn_R0 和 PWMn_R1，且 2 组匹配寄存器各有 1 级缓冲器 PWMn_R0_BUF 和 PWMn_R1_BUF，分别对应于 PWMn 的 2 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，或 PWM 工作中进入刹车状态，PWM 波形为输出初始态，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 19 PWM 操作流程

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>		PWMn_MT		PWMn_TBS<1:0>		PWMn_MOD<1:0>	

- Bit7-6 **PWMn_SPS<1:0>**: PWMn 刹车模式选择位
 11: 软件刹车 (写“11”立即进入刹车状态)
 10: 保留
 01: STP 复用端口输入低电平刹车
 00: 关闭刹车功能
- Bit5-4 **PWMn_MT**
 11: 移相模式, 该模式下 PWM_OUT1 以 PWM_OUT0 为基础, 延时 PWMn_R1 个周期后输出
 其他: 正常 PWM 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 (用于 CAP 模式)
 01: PWM 模式
 00: 关闭

PWMn_R0L PWMn 通道 0 匹配点寄存器低 8 位 (与 CAPn_DL 寄存器复用)

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

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

PWMn_R0H PWMn 通道 0 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>**: PWMn 通道 0 占空比高 8 位

PWMn_R1L PWMn 通道 1 匹配点寄存器低 8 位

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

- Bit7-0 **PWMn_R1<7:0>**: PWMn 通道 1 占空比低 8 位

PWMn_R1H PWMn 通道 1 匹配点寄存器高 8 位

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

Bit7-0 **PWMn_R1<15:8>**: PWMn 通道 1 占空比高 8 位

PWMn_DT PWMn 死区时间寄存器

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

Bit7-0 **PWMn_DT<7:0>**: PWMn 死区时间值

PWMn_OC PWMn 输出控制寄存器

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	RW-0	RW-0	RW-0	RW-1	RW-0	RW-1
PWMn1_D	PWMn0_D	PWMn1_T	PWMn0_T	PWMn1_N	PWMn1_P	PWMn0_N	PWMn0_P

Bit7 **PWMn1_D**: PWMn 通道 1 死区延时边沿选择位

- 1: 延时上升沿
- 0: 延时下降沿

Bit6 **PWMn0_D**: PWMn 通道 0 死区延时边沿选择位

- 1: 延时上升沿
- 0: 延时下降沿

Bit5 **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

Bit4 **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

Bit3 **PWMn1_N**: PWMn 通道 1 匹配点后输出值

- 1: 输出高电平
- 0: 输出低电平

Bit2 **PWMn1_P**: PWMn 通道 1 匹配点前输出值

- 1: 输出高电平
- 0: 输出低电平

Bit1 **PWMn0_N**: PWMn 通道 0 匹配点后输出值

- 1: 输出高电平
- 0: 输出低电平

Bit0 **PWMn0_P**: PWMn 通道 0 匹配点前输出值

- 1: 输出高电平
- 0: 输出低电平

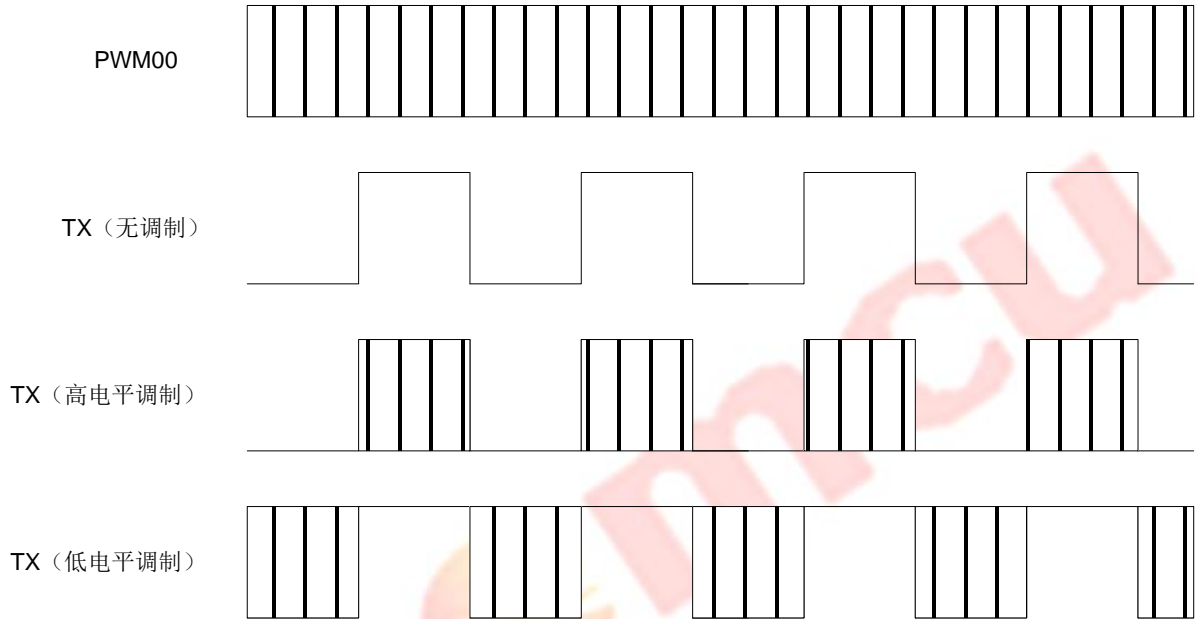
15 UART 异步收发器

15.1 描述

芯片内置 1 路 UART 异步收发器，支持 4 组 UART 通信端口，可实现分时复用。

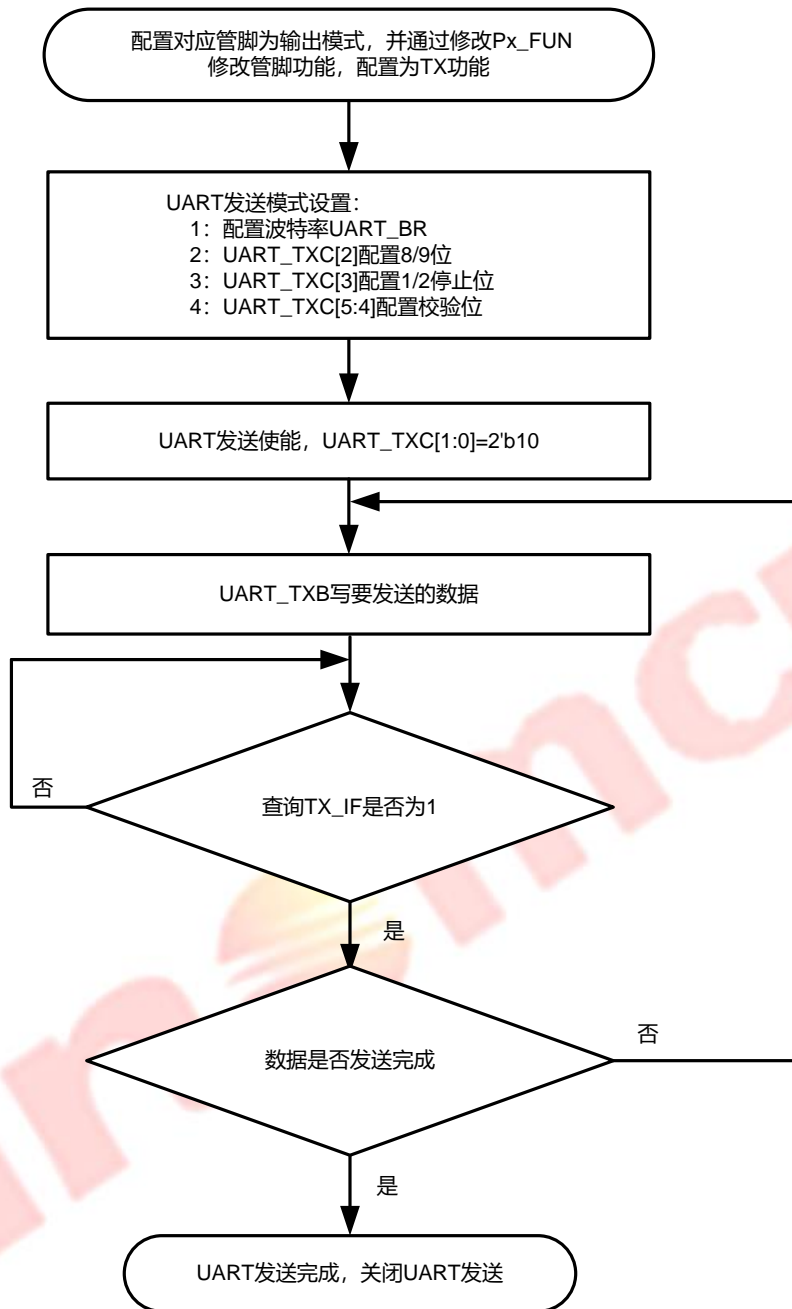
UART 发送 PWM 调制

UART 发送支持 PWM 调制输出，可选择输出高电平调制或输出低电平调制。UART 调制功能固定选择 PWM00 波形进行调制。



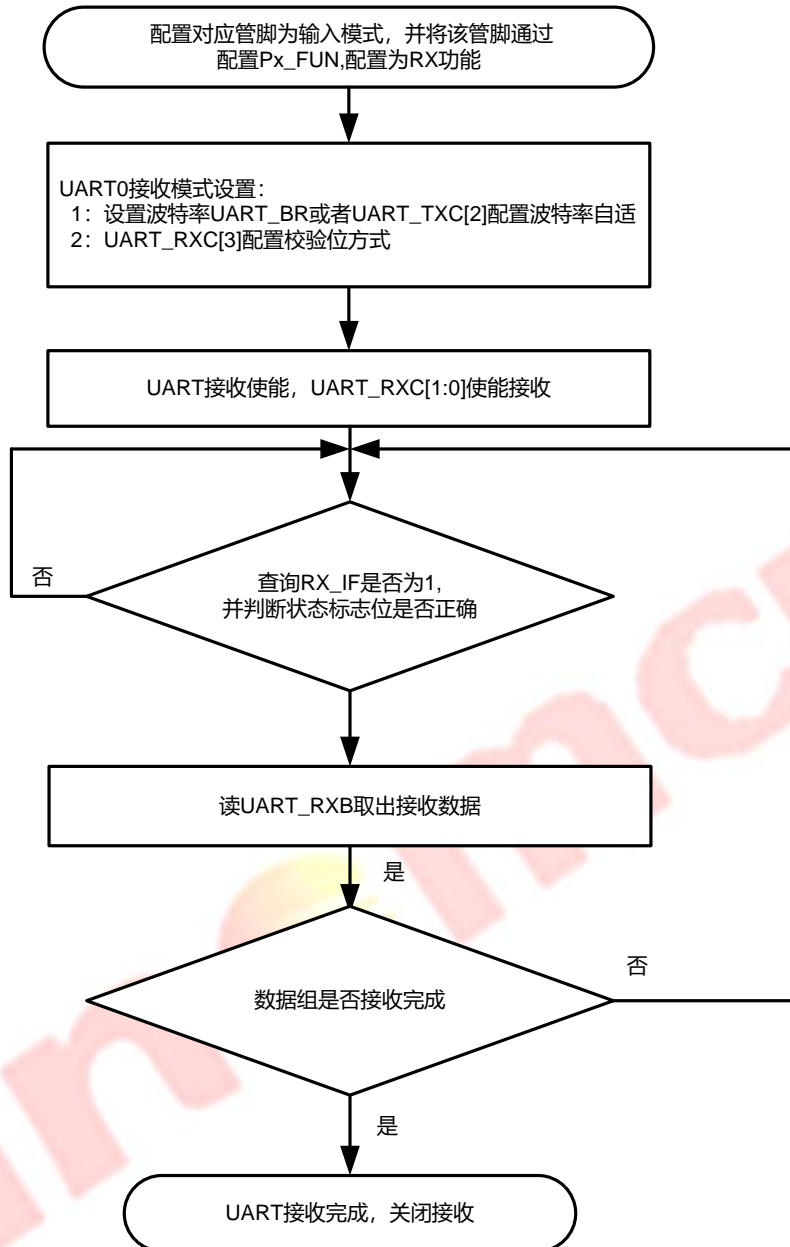
图表 90 UART 发送调制示意图

UART 发送操作流程



图表 21 UART 发送操作流程图

UART 接收操作流程



图表 102 UART 接收操作流程

15.2 UART 寄存器

UART_BRL UART 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

UART_BRH UART 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

UART_RXC UART 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**: UART 接收帧错误标志位

- 1: 有帧错误
- 0: 无帧错误

Bit6 **OERR**: UART 接收溢出错误标志位

- 1: 有溢出错误
- 0: 无溢出错误

Bit5 **PERR**: UART 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**: UART 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**: UART 波特率自适应使能位

- 1: 使能
- 0: 关闭

Bit1-0 **RXEN<1:0>**: UART 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**)
- 10: 关闭 RX 接收功能

UART_RXB UART 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>**: UART 接收数据

UART_TXC UART 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**: UART 发送状态标志位

- 1: UART 发送未完成
- 0: UART 发送已完成

Bit6 **TXBF**: UART 发送缓冲区满标志位

- 1: UART 发送缓冲区满
- 0: UART 发送缓冲区空

Bit5-4 **TX9S<1:0>**: UART 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**: UART 发送数据格式选择位

- 1: 9 位数据
- 0: 8 位数据

Bit1-0 **TXEN<1:0>**: UART 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

UART_TXB UART 接收数据寄存器

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

Bit7-0 **TXB<7:0>**: UART 发送数据

UART_TXM **UART 发送调制控制寄存器** (与 **PORT_C0** 寄存器复用)

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

Bit7-6 **TXM<1:0>**: UART 发送调制模式选择位

- 1x: TX 输出 0 调制
- 01: TX 输出 1 调制
- 00: 关闭调制功能

Bit5-0 保留



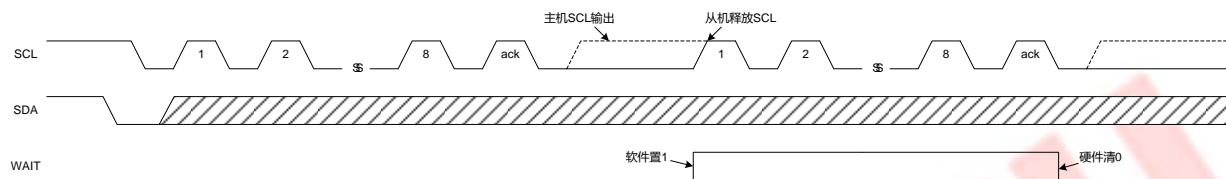
16 IIC 总线控制器

16.1 描述

芯片内置 1 路 IIC 总线控制器，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 通信等待



图表 23 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。

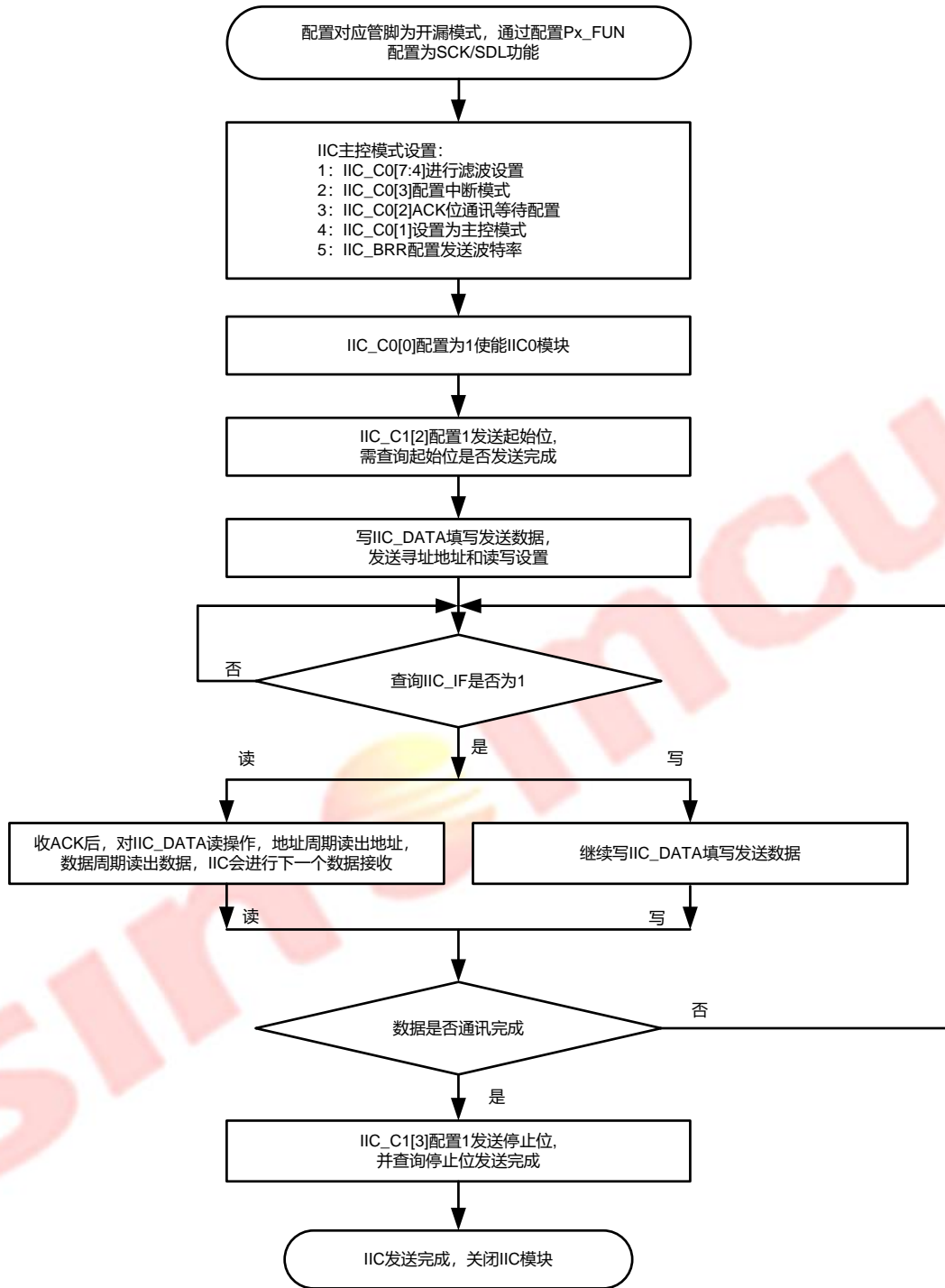
在从动模式中使能通信等待功能，当 IIC 总线上每完成一个字节的发送或接收后，从机自动将时钟线 SCL 拉为低电平，强制进入通信等待状态。待从机作好通信准备后，通过将 **IIC_C1.WAIT** 位置 1 将 SCL 线释放为高电平，主机检测到这个高电平后会继续进行后续传输。

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。

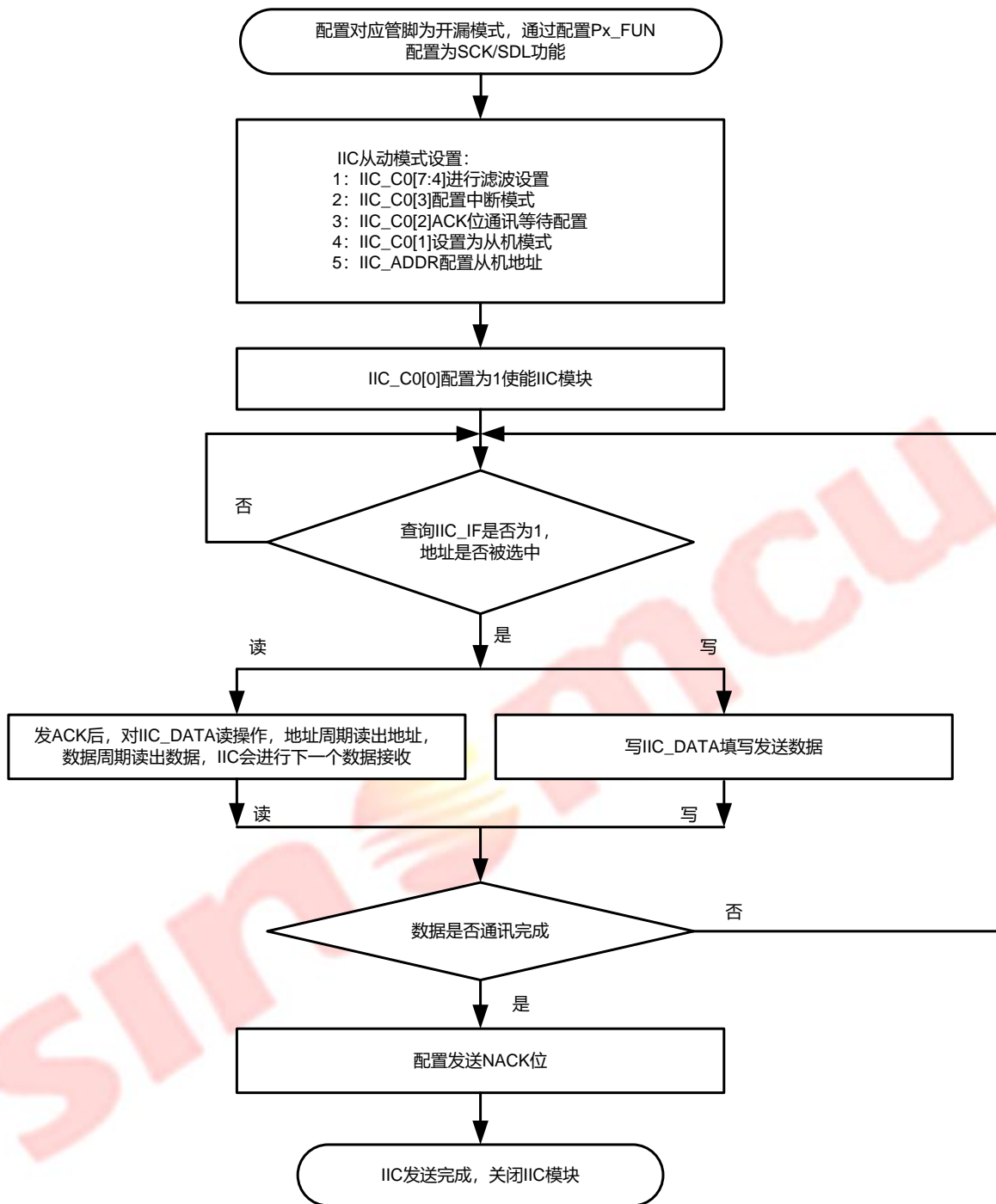
超时检测功能检测 IIC 总线上 SCL 线低电平的持续时间，如果 SCL 线低电平的持续时间超过 **IIC_TOC.IIC_OVT** 所设定的超时时间，则会产生 IIC 超时中断。

IIC 主控通讯流程



图表 24 IIC 主控通讯流程图

IIC 从机通讯流程



图表 2511 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**: 主机发送地址中断屏蔽位

- 1: 主机发送地址不产生中断
- 0: 主机发送地址产生中断

Bit3 **STOP**: STOP 位发送位

- 1: 置 1 发送 STOP 位 (发送完成硬件自动清 0)
- 0: STOP 位发送完成

Bit2 **START**: START 位发送位

- 1: 置 1 发送 START 位 (发送完成硬件自动清 0)
- 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

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

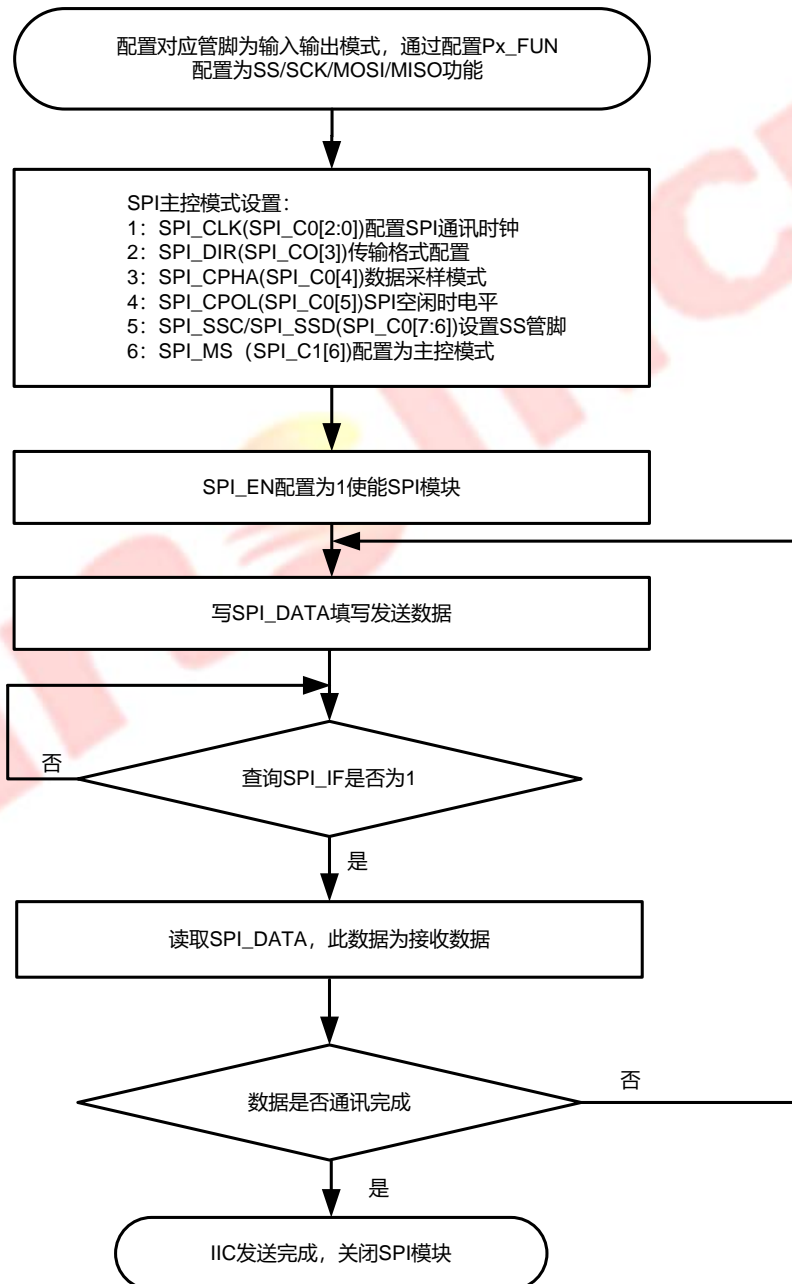
17 SPI 通讯端口

17.1 描述

芯片具备 1 组 SPI 通讯端口。

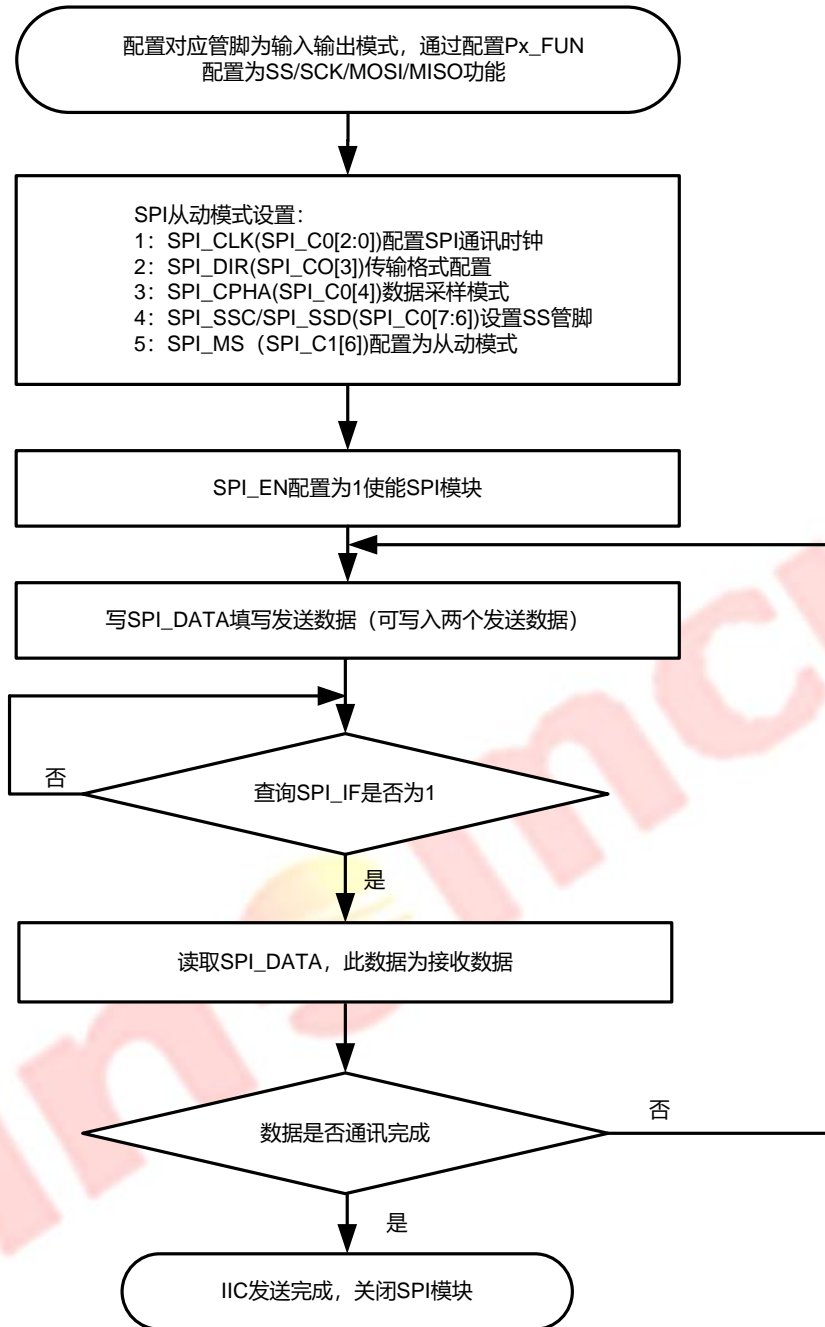
- 支持全双工工作模式
- 支持主从模式
- 极性可配置
- 支持传输格式高低位可配置

SPI 主控通讯流程



图表 2612 SPI 主控模式流程图

SPI 从动通讯流程



图表 13 SPI 从动模式流程图

17.1 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 时将获得接收移位寄存器中的数据

W: 写入 SPI_DATA 的数据被放置到发送移位寄存器中

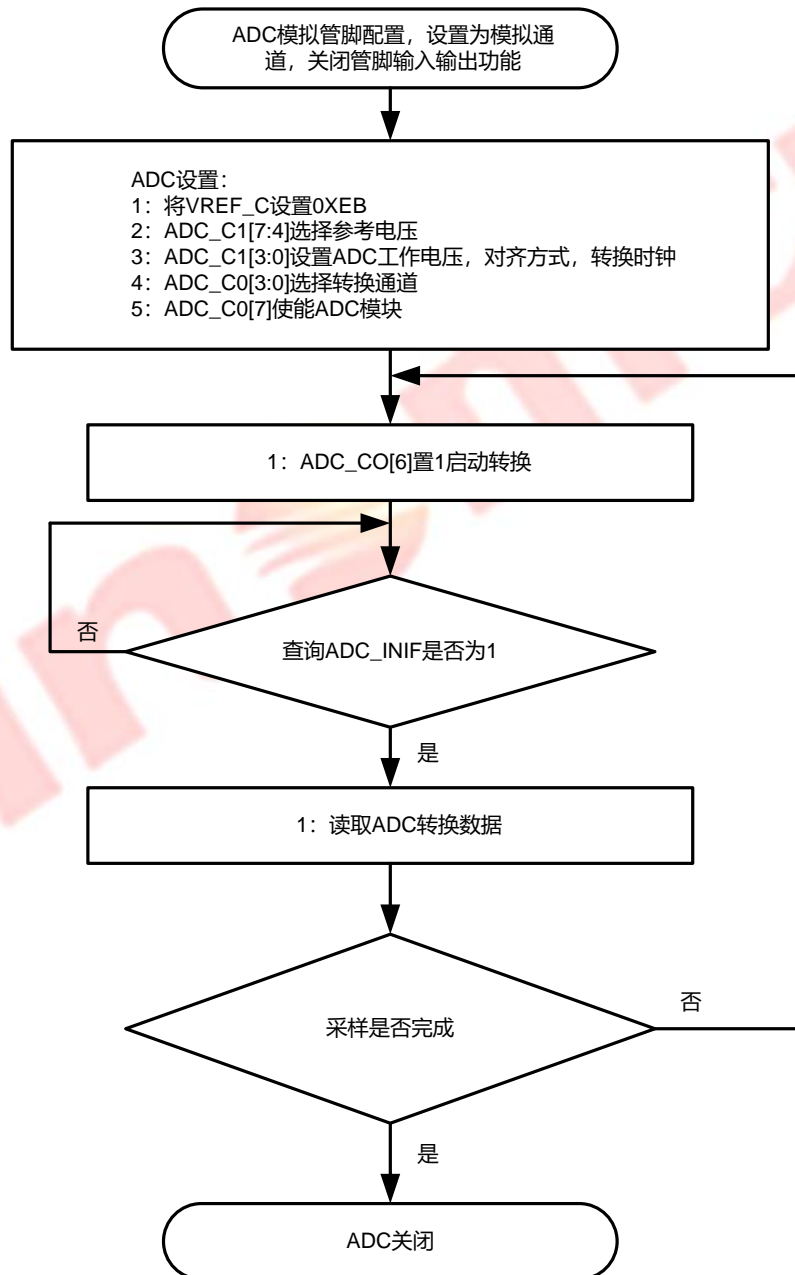
18 模数转换器 ADC

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 500KHz）。

- 支持 11 个外部采样通道 AN0~AN10
- 支持内部 1/2 VDD 和 1/4 VDD 检测通道
- 内部温度传感器

ADC 操作流程



图表 14 ADC 操作流程

18.2 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
 11: TMR3 中断触发
 10: TMR2 中断触发
 01: TMR1 或 TMR0 中断触发
 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
 1111: 所有通道关闭
 1110: 未用
 1101: 选择内部 1/4 VDD 通道
 1100: 选择内部 1/2 VDD 通道
 1011: 内部温度传感器 (需使能 VREF 模块)
 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-0	RW-000		RW-0		RW-0	RW-00	
VREFP_EN	VREFP_S <2:0>		AVDD_S		ADC_DM	ADC_CKS <1:0>	

- Bit7 **VREFP_EN**: VREFP 复用端口选通使能位
 1: VREFP 复用端口选通 (预先设置 P1_IE<7>=0, P1_OE<7>=0,
P1_PUE<7>=0, P1_PDE<7>=0)
 0: VREFP 复用端口关闭
- Bit6-4 **VREFP_S**<2:0>: 内部参考电压源选择位
 111: 选择内部 1/4 VDD
 110: 选择内部 1/2 VDD

101: 选择内部 VREF 2.048V (预先设置 **VREF_C** = EBH)

100: 选择内部 VREF 2.5V (预先设置 **VREF_C** = EBH)

001: 选择 VDD

其他: 关闭内部参考电压 VREF

Bit3 **AVDD_S**: ADC 工作系统电压选择位

1: 芯片工作在 5V 系统 (使用外部参考或者 VDD 参考时无需使能 VREF)

0: 芯片工作在 3.3V 系统 (该模式下必须使能 VREF)

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 (用户固定写 1)

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_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

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 CX 的容值大小，以及设置 TK_C1.TK_VRFS<2:0>位调整 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 15 路触控按键检测通道 TK1~TK15，以及 1 路内部参考通道 TK0。内部参考通道集成固定 10pF 电容，用于调整放大倍数的参考基准。

需要注意的是，使能 TK 模块必须使能 VREF 模块

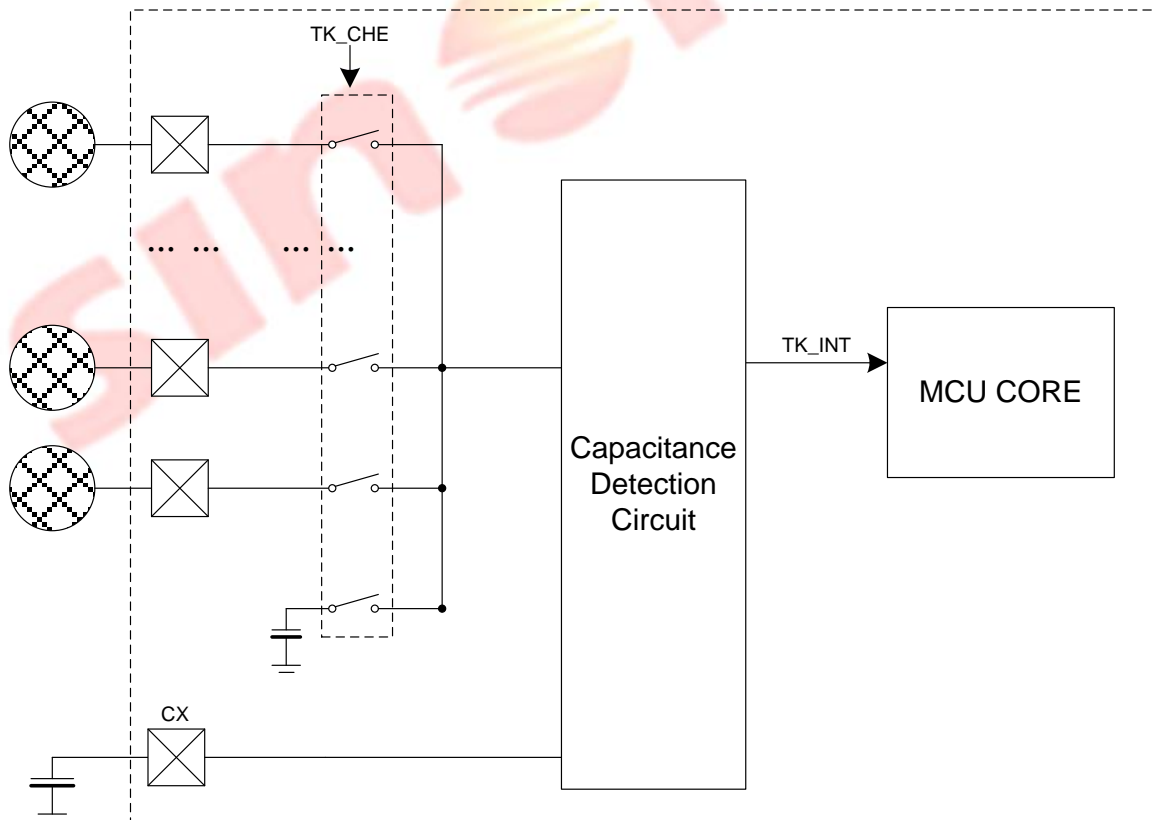
TK 扫描溢出

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 29 TK 功能框图

19.3 TK 寄存器

TK_C0 TK 控制寄存器 0

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0	RW-0	—	—	R-0	R-0	—	RW-0
TK_EN	TK_VTKS	—	—	TK_OV	TK_ERR	—	TK_GO

- Bit7 **TK_EN**: TK 模块使能位
1: 使能
0: 关闭
- Bit6 **TK_VTKS**: TK 模块基准电源选择位
1: VREF 2.5V
0: VDD
- Bit5-4 保留
- Bit3 **TK_OV**: TK 扫描溢出标志位
1: 上一次扫描结果溢出
0: 上一次扫描结果未溢出
- Bit2 **TK_ERR**: TK 扫描启动错误标志位
1: 上一次扫描发生启动错误
0: 上一次扫描未发生启动错误
- Bit1 保留
- Bit0 **TK_GO**: TK 扫描触发位
1: 启动 TK 扫描
0: TK 扫描已完成

TK_C1 TK 控制寄存器 1

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-0000				—	RW-000		
TK_FLT <3:0>				—	TK_VRFS <2:0>		

- Bit7-4 **TK_FLT**<3:0>: TK 滤波控制位
0000: 滤波关闭
其他: TK 滤波时间 $T_{FLT} = \frac{TK_FLT<3:0> + 1}{F_{TKCLK}}$
- Bit3 保留
- Bit2-0 **TK_VRFS**<2:0>: TK 检测阈值选择位, 需使能 VREF 模块, 请查看 21 节
111: 2.0V 110: 1.6V 101: 1.4V 100: 1.2V
011: 1.0V 010: 0.8V 001: 0.6V 000: 0.2V

TK_C2 TK 控制寄存器 2

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
RW-00_0000						RW-00	
TK_DIV <5:0>						TK_CKS <1:0>	

- Bit7-2 **TK_DIV**<4:0>: TK 扫描时钟分频位

$$F_{TKCLK} = \frac{F_{TK_SRC}}{(TK_DIV<5:0> + 1) \times 2}$$

- Bit1-0 **TK_CKS<1:0>**: TK 扫描时钟源 TK_SRC 选择位
- 11: 选择 PWM0 波形输出
 - 10: 选择 HRC
 - 01: 选择 LRC
 - 00: 选择系统时钟 SYSCLK

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 TK7~TK0, 其中 TK_CHE<0>对应内部通道)
- 1: 使能 TKn 扫描通道
 - 0: 关闭 TKn 扫描通道

TK_CHEH TK 扫描通道使能寄存器高 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 TK15~TK8)
- 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

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。

ADC 的温度采样通道也来自于 VREF 模块，进行 ADC 的温度采样前，需预先使能 VREF 模块。

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

TK_VREF_C TK 模式下参考配置寄存器

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

Bit7-0 **TK_VREF_C <7:0>**: TK_VREF_C 配置位
 08H: 使能 TK 参考电压
 00H: 关闭 TK 参考模块
 其他: 保留

21 低电压检 LVD

21.1 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，被监测电压由低于比较阈值产生 LVD 中断标志。

21.2 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-0	RW-0	RW-000		
LVD_EN	LVD_IM	LVD_FLT	LVD_CKS	LVD_INS	LVD_VS<2:0>		

Bit7 **LVD_EN**: LVD 使能位

- 1: 使能
- 0: 关闭

Bit6 **保留(用户固定写 0)**

Bit5 **LVD_FLT**: LVD 滤波选择位

- 1: 7 个 LVD 工作时钟
- 0: 关闭滤波器

Bit4 **LVD_CKS**: LVD 工作时钟选择位

- 1: 内部低频时钟 LRC
- 0: 系统时钟 SYS_CLK

Bit3 **LVD_INS**: 被监测电压选择位

- 1: LVDIN 复用管脚 (阈值电压固定为 1.25V)
- 0: VDD

Bit2-0 **LVD_VS<1:0>**: 阈值电压选择位 (仅用于监测 VDD)

- | | | | |
|-----------|-----------|-----------|-----------|
| 111: 4.3V | 110: 4.0V | 101: 3.7V | 100: 3.4V |
| 011: 3.1V | 010: 2.8V | 001: 2.5V | 000: 2.2V |

22 指令集

22.1 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

22.2 逻辑操作指令

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

22.3 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

22.4 位操作指令

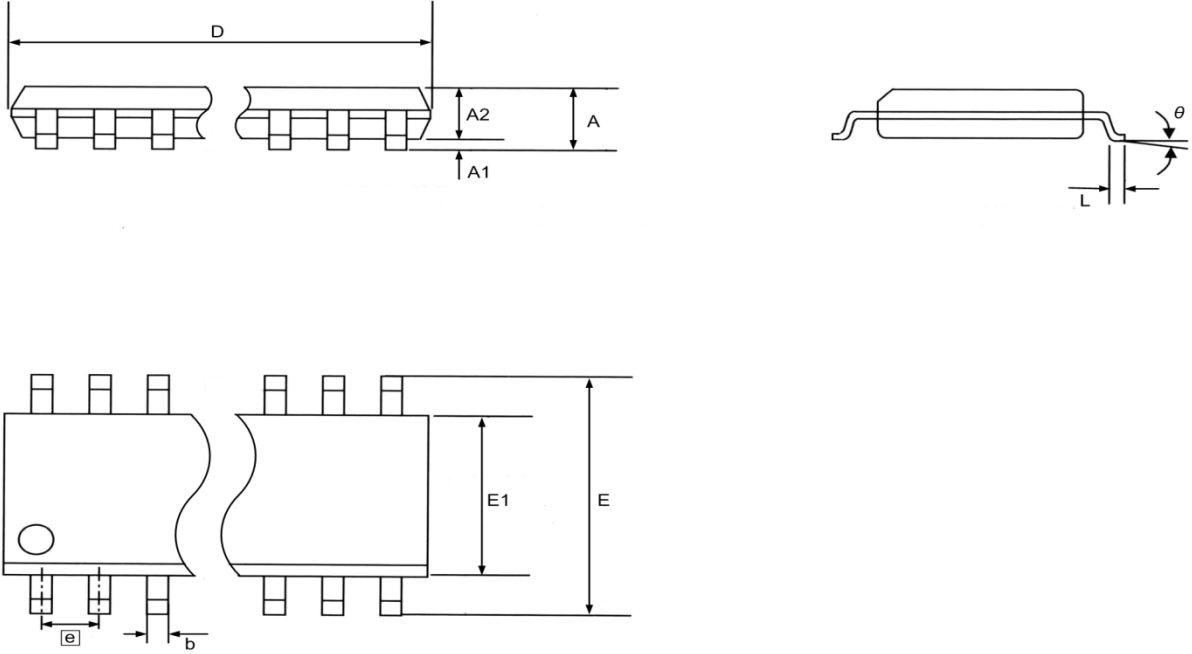
指令		描述	字节	周期
CLR	C	C = 0	1	1
CLR	bit	bit = 0	2	3
SETB	C	C = 1	1	1
SETB	bit	bit = 1	2	3
CPL	C	C = <i>not</i> C	1	1
CPL	bit	bit = <i>not</i> bit	2	3
ANL	C, bit	C = C <i>and</i> bit	2	2
ANL	C, /bit	bit = C <i>and</i> (<i>not</i> bit)	2	2
ORL	C, bit	C = C <i>or</i> bit	2	2
ORL	C, /bit	bit = C <i>or</i> (<i>not</i> bit)	2	2
MOV	C, bit	C = bit	2	2
MOV	bit, C	bit = C	2	3

22.5 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≠0)	2	3/5
DJNZ	direct, rel	[direct]=[direct]-1, 相对短跳转(如果 [direct]≠0)	2	3/5
NOP		空操作	1	1

23 封装信息

SOP20



标号	公制 (mm)			英制 (inch)		
	MIN	NOM	MAX	MIN	NOM	MAX
A	2.30	2.50	2.70	0.090	0.098	0.107
A1	0.10	0.20	0.30	0.003	0.007	0.012
A2	2.10	2.30	2.50	0.082	0.090	0.099
D	12.60	12.80	13.00	0.496	0.504	0.513
E	10.10	10.30	10.50	0.397	0.405	0.414
E1	7.30	7.50	7.70	0.287	0.295	0.303
b	-	0.40	-	-	0.015	-
e	-	1.27	-	-	0.050	-
L	0.75	0.85	0.95	0.029	0.033	0.038
θ	0°	-	8°	0°	-	8°

24 修订记录

版本	修订日期	修订内容
V1.0	2019-3-14	初版发布
V1.1	2019-5-20	修改封装脚位