

**8 位 MCU**  
**ES7P2031FGSA**

# 数 据 手 册

- 产品简介
- 数据手册
- 产品规格

上海东软载波微电子有限公司

2019 年 12 月 6 日

## MCU 芯片使用注意事项

### 关于芯片的上/下电

东软微电子 MCU 芯片具有独立电源管脚。当 MCU 芯片应用在多电源供电系统时，应先对 MCU 芯片上电，再对系统其它部件上电；反之，下电时，先对系统其它部件下电，再对 MCU 芯片下电。若操作顺序相反则可能导致芯片内部元件过压或过流，从而导致芯片故障或元件退化。具体可参照芯片的数据手册说明。

### 关于芯片的复位

东软微电子 MCU 芯片具有内部上电复位。对于不同的快速上/下电或慢速上/下电系统，内部上电复位电路可能失效，建议用户使用外部复位、下电复位、看门狗复位等，确保复位电路正常工作。在系统设计时，若使用外部复位电路，建议采用三极管复位电路、RC 复位电路。若不使用外部复位电路，建议采用复位管脚接电阻到电源，或采取必要的电源抖动处理电路或其它保护电路。具体可参照芯片的数据手册说明。

### 关于芯片的时钟

东软微电子 MCU 芯片具有内部和外部时钟源。内部时钟源会随着温度、电压变化而偏移，可能会影响时钟源精度；外部时钟源采用陶瓷、晶体振荡器电路时，建议使能起振延时；使用 RC 振荡电路时，需考虑电容、电阻匹配；采用外部有源晶振或时钟输入时，需考虑输入高/低电平电压。具体可参照芯片的数据手册说明。

### 关于芯片的初始化

东软微电子 MCU 芯片具有各种内部和外部复位。对于不同的应用系统，有必要对芯片寄存器、内存、功能模块等进行初始化，尤其是 I/O 管脚复用功能进行初始化，避免由于芯片上电以后，I/O 管脚状态的不确定情况发生。

### 关于芯片的管脚

东软微电子 MCU 芯片具有宽范围的输入管脚电平，建议用户输入高电平应在  $V_{IHMIN}$  之上，低电平应在  $V_{ILMAX}$  之下。避免输入电压介于  $V_{IHMIN}$  和  $V_{ILMAX}$  之间，以免波动噪声进入芯片。对于未使用的输入/输出管脚，建议用户设为输入状态，并通过电阻上拉至电源或下拉至地，或设置为输出管脚，输出固定电平并浮空。对未使用的管脚处理因应用系统而异，具体遵循应用系统的相关规定和说明。

### 关于芯片的 ESD 防护措施

东软微电子 MCU 芯片具有满足工业级 ESD 标准保护电路。建议用户根据芯片存储/应用的环境采取适当静电防护措施。应注意应用环境的湿度；建议避免使用容易产生静电的绝缘体；存放和运输应在抗静电容器、抗静电屏蔽袋或导电材料容器中；包括工作台在内的所有测试和测量工具必须保证接地；操作者应该佩戴静电消除手腕环手套，不能用手直接接触芯片等。

### 关于芯片的 EFT 防护措施

东软微电子 MCU 芯片具有满足工业级 EFT 标准的保护电路。当 MCU 芯片应用在 PCB 系统时，需要遵守 PCB 相关设计要求，包括电源、地走线（包括数字/模拟电源分离，单/多点接地等）、复位管脚保护电路、电源和地之间的去耦电容、高低频电路单独分别处理以及单/多层板选择等。

### 关于芯片的开发环境

东软微电子 MCU 芯片具有完整的软/硬件开发环境，并受知识产权保护。选择上海东软载波微电子有限公司或其指定的第三方公司的汇编器、编译器、编程器、硬件仿真器开发环境，必须遵循与芯片相关的规定和说明。

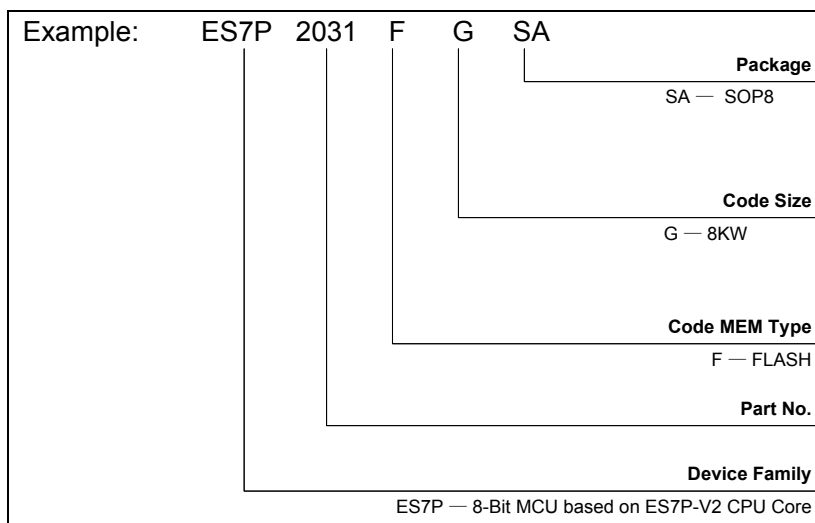
注：在产品开发时，如遇到不清楚的地方，请通过销售或其它方式与上海东软载波微电子有限公司联系。

## 产品订购信息

型号	FLASH	RAM	I/O	Timer	UART	I2C	ADC	LVD	TK	封装类型
ES7P2031FGSA	8K Word	512 Byte	6	8-bit×1 16-bit×2	1	1	12-bit× (2+1)	1	3	SOP8

注 1: 512 Byte RAM 中内部含可配置的最大级数为 32 级的堆栈。

注 2: 本数据手册所述资源为本产品所属产品系列资源最大化描述。文中所述（包括管脚、寄存器、芯片配置字等）不为本产品所有的资源，请将其保持在默认设置值状态。如有其它需求，请与我司 FAE 部门联系！



地 址：中国上海市龙漕路 299 号天华信息科技园 2A 楼 5 层

邮 编：200235

E-mail: support@essemi.com

电 话：+86-21-60910333

传 真：+86-21-60914991

网 址：<http://www.essemi.com>

版权所有©

### 上海东软载波微电子有限公司

本资料内容为上海东软载波微电子有限公司在现有数据资料基础上慎重且力求准确无误编制而成，本资料中所记载的实例以正确的使用方法和标准操作为前提，使用方在应用该等实例时请充分考虑外部诸条件，上海东软载波微电子有限公司不承担或确认该等实例在使用方的适用性、适当性或完整性，上海东软载波微电子有限公司亦不对使用方因使用本资料所有内容而可能或已经带来的风险或后果承担任何法律责任。基于使本资料的内容更加完善等原因，上海东软载波微电子有限公司保留未经预告的修改权。使用方如需获得最新的产品信息，请随时用上述联系方式与上海东软载波微电子有限公司联系。

## 修订历史

版本	修改日期	更改概要
V1.0	2019-12-6	初版

## 目 录

### 内容目录

<b>第 1 章</b>	<b>芯片简介</b> .....	<b>12</b>
1.1	概要 .....	12
1.1.1	特性.....	12
1.1.2	应用领域 .....	14
1.2	结构框图 .....	15
1.3	管脚分配图 .....	16
1.3.1	SOP8 封装图.....	16
1.4	管脚说明 .....	17
<b>第 2 章</b>	<b>内核特性</b> .....	<b>18</b>
2.1	CPU 内核概述 .....	18
2.2	硬件乘法器和硬件除法器.....	18
2.3	特殊功能寄存器 .....	18
<b>第 3 章</b>	<b>存储资源</b> .....	<b>24</b>
3.1	概述 .....	24
3.2	程序寻址空间映射 .....	24
3.3	FLASH 程序存储器.....	25
3.3.1	概述.....	25
3.3.2	程序计数器 (PC) .....	25
3.3.3	硬件堆栈 .....	25
3.3.4	FLASH 存储器的查表读和 IAP 操作 .....	26
3.3.4.1	概述 .....	26
3.3.4.2	FLASH 存储器的查表读.....	26
3.3.4.3	FLASH 存储器的 IAP 擦除.....	27
3.3.4.4	FLASH 存储器的 IAP 编程.....	29
3.3.4.5	特殊寄存器.....	32
3.4	在线编程 ISP 和在线调试 ICD.....	35
3.5	数据寻址空间.....	36
3.5.1	概述.....	36
3.5.2	数据寻址空间映射 .....	36
3.5.3	通用数据存储器 SRAM .....	37
3.5.4	特殊功能寄存器 SFR .....	38
3.5.5	寻址方式 .....	41
3.5.5.1	直接寻址 .....	41
3.5.5.2	GPR 特殊寻址 .....	41
3.5.5.3	间接寻址 .....	42
3.6	特殊功能寄存器 .....	42
<b>第 4 章</b>	<b>输入/输出端口</b> .....	<b>46</b>
4.1	概述 .....	46
4.2	I/O 结构框图 .....	46
4.3	I/O 端口功能 .....	47
4.3.1	I/O 端口输入/输出控制 .....	47

4.3.2	I/O 端口弱上拉、弱下拉功能 .....	47
4.3.3	I/O 端口模拟/数字类型选择功能.....	47
4.3.4	增强驱动能力 I/O .....	47
4.3.5	I/O 端口复用功能 .....	47
4.4	I/O 端口操作注意事项.....	47
4.5	特殊功能寄存器 .....	48
<b>第 5 章</b>	<b>特殊功能及操作特性.....</b>	<b>55</b>
5.1	系统时钟和振荡器 .....	55
5.1.1	概述.....	55
5.1.2	结构框图 .....	55
5.1.3	时钟源 .....	55
5.1.3.1	内部高速 16MHz RC 振荡器 HRC.....	55
5.1.3.2	内部低速 32KHz RC 振荡器 LRC.....	55
5.1.4	系统时钟源切换.....	56
5.1.4.1	系统上电时序.....	56
5.1.5	系统时钟分频 .....	56
5.1.6	时钟切换等待 .....	57
5.1.7	特殊功能寄存器.....	57
5.2	看门狗定时器.....	59
5.2.1	概述.....	59
5.2.2	WDT 操作.....	59
5.2.3	特殊功能寄存器.....	60
5.3	复位模块.....	62
5.3.1	概述.....	62
5.3.2	上电复位 POR.....	62
5.3.3	掉电复位 BOR.....	63
5.3.4	看门狗定时器溢出复位.....	63
5.3.5	RST 指令复位 .....	63
5.3.6	特殊功能寄存器.....	64
5.4	低功耗操作 .....	66
5.4.1	概述.....	66
5.4.2	IDLE 状态 .....	66
5.4.3	唤醒方式配置 .....	66
5.4.4	低功耗下的功能模块 .....	66
5.4.1	时钟源的关闭和唤醒 .....	66
5.4.2	特殊功能寄存器.....	67
<b>第 6 章</b>	<b>外设 .....</b>	<b>68</b>
6.1	定时器/计数器 (Timer/Counter) 模块.....	68
6.1.1	8 位定时器/计数器 (T10) .....	68
6.1.1.1	概述 .....	68
6.1.1.2	内部结构图.....	68
6.1.1.3	预分频器 .....	68
6.1.1.4	工作模式 .....	69
6.1.1.5	定时器模式.....	69

6.1.1.6	计数器模式.....	70
6.1.1.7	特殊功能寄存器.....	70
6.1.2	16位多功能定时器 (T20/T21) .....	72
6.1.2.1	概述.....	72
6.1.2.2	预分频器.....	72
6.1.2.3	工作模式.....	72
6.1.2.4	定时器模式.....	72
6.1.2.5	特殊功能寄存器.....	73
6.2	异步接收发送器 (UART0) .....	75
6.2.1	概述.....	75
6.2.2	内部结构图.....	75
6.2.3	波特率配置.....	76
6.2.4	传输数据格式.....	76
6.2.5	异步发送器.....	76
6.2.6	异步接收器.....	77
6.2.7	UARTn 使用注意事项.....	78
6.2.8	特殊功能寄存器.....	79
6.3	I2C 总线从动器 (I2CS) .....	81
6.3.1	概述.....	81
6.3.2	I2CS 端口配置.....	81
6.3.3	通讯协议.....	82
6.3.4	数据传输格式参考.....	82
6.3.5	中断和暂停.....	83
6.3.6	特殊功能寄存器.....	83
6.4	触摸按键控制 (TK) .....	87
6.4.1	概述.....	87
6.4.2	内部结构图.....	87
6.4.3	触摸按键扫描工作原理.....	87
6.4.4	触摸按键端口控制.....	89
6.4.5	数据管理.....	89
6.4.6	触摸按键扫描参考流程.....	91
6.4.7	特殊功能寄存器.....	92
6.5	模/数转换器模块 (ADC) .....	98
6.5.1	概述.....	98
6.5.2	ADC 内部结构图.....	98
6.5.3	ADC 配置.....	99
6.5.4	ADC 转换步骤.....	99
6.5.5	ADC 时序特征示意图.....	100
6.5.6	特殊功能寄存器.....	101
6.6	低电压检测模块 (LVD) .....	103
6.6.1	概述.....	103
6.6.2	LVD 操作.....	103
6.6.3	特殊功能寄存器.....	104
<b>第 7 章</b>	<b>中断处理.....</b>	<b>105</b>

7.1	概述 .....	105
7.2	内部结构 .....	105
7.3	中断模式选择 .....	106
7.3.1	默认中断模式 .....	106
7.3.2	向量中断模式 .....	107
7.3.2.1	向量表配置 .....	107
7.3.2.2	中断向量分组 .....	107
7.4	中断使能配置 .....	108
7.5	中断现场保护 .....	108
7.6	中断操作 .....	108
7.6.1	T10 溢出中断 .....	109
7.6.2	DIV 中断 .....	109
7.6.3	T2n 溢出中断 .....	109
7.6.4	UART 发送/接收中断 .....	109
7.6.5	I2CS 中断 .....	109
7.6.6	ADC 中断 .....	110
7.6.7	IAP 中断 .....	110
7.6.8	LVD 中断 .....	110
7.6.9	触摸按键中断 .....	110
7.7	中断操作注意事项 .....	111
7.8	特殊功能寄存器 .....	111
<b>第 8 章</b>	<b>芯片配置字 .....</b>	<b>115</b>
<b>第 9 章</b>	<b>芯片封装图 .....</b>	<b>117</b>
9.1	8-pin 封装图 .....	117
<b>附录 1</b>	<b>指令集 .....</b>	<b>118</b>
附录 1.1	概述 .....	118
附录 1.2	寄存器操作指令 .....	118
附录 1.3	程序控制指令 .....	118
附录 1.4	算术/逻辑运算指令 .....	120
<b>附录 2</b>	<b>特殊功能寄存器总表 .....</b>	<b>122</b>
<b>附录 3</b>	<b>电气特性 .....</b>	<b>131</b>
附录 3.1	参数特性表 .....	131
附录 3.2	参数特性图 .....	136



## 图目录

图 1-1 结构框图.....	15
图 1-2 SOP8 封装图.....	16
图 3-1 程序寻址空间映射图 .....	24
图 3-2 堆栈示意图.....	26
图 3-3 页擦流程图.....	28
图 3-4 编程流程图.....	30
图 3-5 数据寻址空间映射图 .....	36
图 3-6 SRAM 地址映射示意图 .....	37
图 3-7 快速访问区特殊功能寄存器 .....	38
图 3-8 特殊功能寄存器 (Section 0) .....	39
图 3-9 特殊功能寄存器 (Section 1) .....	40
图 3-10 直接寻址示意图.....	41
图 3-11 GPR 特殊寻址示意图 .....	41
图 3-12 间接寻址示意图.....	42
图 4-1 I/O 端口结构图 .....	46
图 5-1 系统时钟内部结构图 .....	55
图 5-2 系统上电时序图.....	56
图 5-3 看门狗定时器内部结构图 .....	60
图 5-4 系统复位内部结构图 .....	62
图 5-5 上电复位时序示意图 .....	62
图 5-6 低电压复位时序示意图.....	63
图 5-7 看门狗溢出复位.....	63
图 5-8 RST 指令复位 .....	64
图 6-1 T10 内部结构图.....	68
图 6-2 定时器模式时序图 .....	70
图 6-3 计数器模式时序图 (T10EG=0, T10CKI 上升沿计数) .....	70
图 6-4 T2n 定时器模式时序图.....	73
图 6-5 UART 发送端原理图 .....	75
图 6-6 UART 接收端原理图 .....	76
图 6-7 UARTn 数据格式示意图.....	76
图 6-8 UARTn 发送器操作流程 .....	77
图 6-9 UARTn 发送器发送数据时序图 (9 位数据格式, 第 9 位数据为“0”) .....	77
图 6-10 UARTn 接收器操作流程 .....	78
图 6-11 UARTn 接收器接收数据时序图 (9 位数据格式) .....	78
图 6-12 I2C 总线通讯协议示意图.....	82
图 6-13 主控器写入从动器数据示意图.....	82
图 6-14 主控器读取从动器数据示意图.....	83
图 6-15 触摸按键触摸结构示意图.....	87
图 6-16 无触摸状态示意图 .....	88
图 6-17 有触摸状态示意图 .....	88
图 6-18 多通道轮询扫描工作示意图 .....	89
图 6-19 触摸按键操作流程 .....	91

---

图 6-20	ADC 内部结构图 .....	98
图 6-21	ADC 时序特征示意图 .....	100
图 6-22	LVD 工作时序图 .....	103
图 7-1	默认中断模式中中断控制逻辑 .....	105
图 7-2	向量中断模式中中断控制逻辑 .....	106

## 表目录

表 1-1	管脚说明.....	17
表 3-1	在线编程/调试管脚说明.....	35
表 5-1	功能模块低功耗配置分类表 .....	66
表 6-1	T10 预分频器配置表 .....	69
表 6-2	UARTn 波特率配置表 .....	76
表 7-1	中断向量表.....	107
表 7-2	向量中断模式中断分组配置表.....	107
表 7-3	中断使能配置表.....	108

## 第 1 章 芯片简介

### 1.1 概要

#### 1.1.1 特性

- ◆ 工作条件
  - ◇ 工作电压范围：2.7V ~ 5.5V
  - ◇ 工作温度范围：-40 ~ 85°C
- ◆ 工艺、封装
  - ◇ 低功耗、高速 FLASH CMOS 工艺
  - ◇ 采用 SOP8 封装
- ◆ 内核
  - ◇ 采用 ES7P-V2 RISC CPU 内核，79 条精简指令集
  - ◇ 采用 2T 架构，系统时钟最高支持 16MHz，最小指令周期 125ns
  - ◇ 支持中断优先级和中断向量表
  - ◇ 程序指针硬件堆栈共享 SRAM 空间，堆栈级数用户可配置，最大 32 级
- ◆ 复位
  - ◇ 内嵌上电复位 POR(释放电压点 2.1V)
  - ◇ 内嵌下电复位 BOR，支持 2.3V、2.5V、3.1V 和 3.7V 四个电压点选择
  - ◇ 支持看门狗定时器 WDT 计数溢出复位
  - ◇ 支持指令 RST 复位
- ◆ 时钟源
  - ◇ 内部 16MHz 高频振荡器 HRC
    - 用于主系统时钟源，内置系统时钟分频器
    - 出厂校准精度为±1%，校准后全温度全电压范围内精度±2%
  - ◇ 内部 32KHz 低频振荡器 LRC
    - 用于 WDT 时钟源，可选择用于主系统和部分外设时钟源
    - 出厂校准精度为±2%，校准后全温度全电压范围内精度±15%
- ◆ 低功耗
  - ◇ 支持低功耗睡眠模式及硬件唤醒
  - ◇ 待机电流：I<sub>vdd</sub> ≤ 5uA（常温，典型值）
  - ◇ 工作电流：I<sub>vdd</sub> ≤ 3mA（@内部 HRC 16MHz，典型值）
- ◆ 硬件看门狗电路
  - ◇ 采用内部 32KHz LRC 时钟源，最大计数溢出周期约 4.096 秒
  - ◇ 可选择睡眠模式下运行或暂停，睡眠模式下运行计数溢出可唤醒 CPU
- ◆ 存储资源

- ◇ 8K Words FLASH 程序存储器
  - 程序存储器空间中的 512 Words 可配置为 DataFLASH
  - 支持 IAP(In Application Programming)操作,可以擦写 CodeFlash 与 DataFlash
  - IAP 操作时 CPU 暂停运行, 外设可设置为运行或停止
- ◇ 512 字节 SRAM 数据存储器 (部分空间与程序指针堆栈共享)
- ◆ 编程和调试
  - ◇ 支持 5 线制编程接口 (ISP)
  - ◇ 支持 5 线制在线调试接口 (ICD)
  - ◇ 支持两组编程/调试接口可选择
  - ◇ 支持整体编程代码加密保护(不支持分区加密)
- ◆ I/O 端口
  - ◇ 支持最多 6 个 I/O 端口
  - ◇ 支持源电流可调 (3 级: 4mA/7mA/11mA, 4 个 IO 口为一组, 总共设置 3 组)
- ◆ 外设
  - ◇ 1 路 8 位定时/计数器 T10
    - 内置 8 位预分频器
    - 支持定时器/计数器模式
    - 支持计数器溢出中断
  - ◇ 2 路 16 位定时/计数器 T20/T21
    - 内置 8 位预分频器
    - 支持定时器模式
    - 支持后分频溢出中断
  - ◇ 1 路异步串行收发器 UART0
    - 内置波特率发生器
    - 支持异步全双工收发
    - 8 位/9 位数据格式可选
    - 约定从最低位接收/发送
    - 支持接收和发送中断
    - UART0 可软件配置所需使用的通讯端口
  - ◇ 1 路 I2C 总线
    - 只支持从动模式
    - 支持标准 I2C 总线协议, 最高传输速率 400K bit/s
    - 支持 7 位寻址方式
    - 约定数据从最高位开始接收/发送
    - 支持中断产生

- ◇ 触摸按键功能模块 TK
  - 支持最多 3 个触摸按键
  - 支持工作频率可配置：8MHz、4MHz、2MHz、1MHz、500KHz
  - 支持工作频率的抖频，抖频范围为（+100%，-50%）
  - 支持触摸按键扫描中断
  - 支持多次扫描结果的累计或平均运算
  - 可配置计时扫描或计次扫描
  - 支持工作错误（启动错误、扫描溢出、运算出错）监测
- ◇ 模拟数字转换器 ADC
  - 12 位数字转换
  - 2+1 通道模拟输入端
  - 支持 ADC 转换完成中断
  - 仅支持 VDD 作为正参考源、VSS 作为负参考源
- ◇ 低电压监测模块
  - 支持 VDD 监测电压点可配置（2.4V ~ 4.6V, 7 档）
  - 支持被监测电压的掉电和上电中断
  - 可用于 FLASH 存储器 IAP 操作时的 VDD 电压异常保护机制

### 1.1.2 应用领域

本芯片可广泛用于台灯、墙壁开关、小家电等触控领域。

## 1.2 结构框图

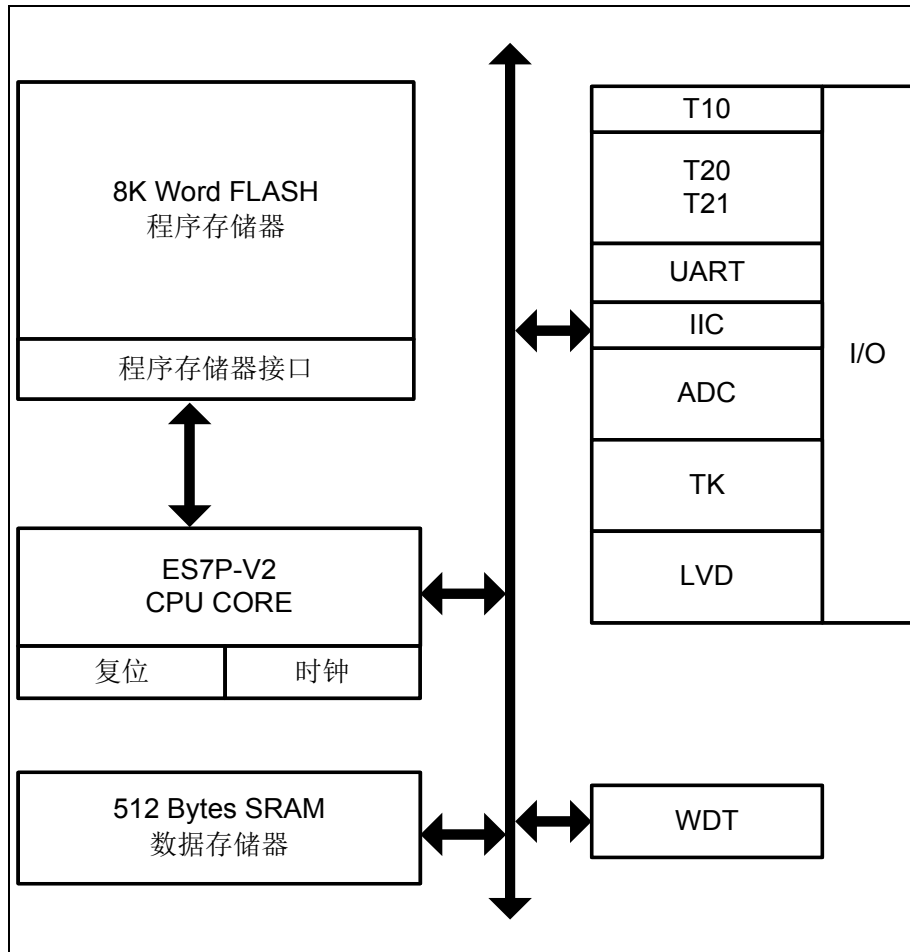


图 1-1 结构框图

### 1.3 管脚分配图

#### 1.3.1 SOP8 封装图

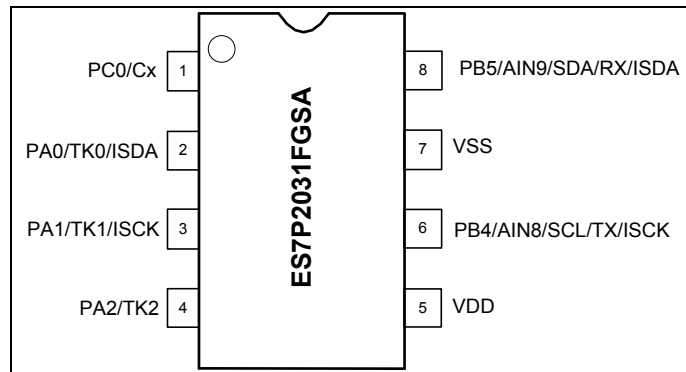


图 1-2 SOP8 封装图



## 1.4 管脚说明

管脚名称	功能	A/D	端口说明
PA0/TK0/ISDA	PA0	D	通用 I/O 端口
	TK0	A	触摸按键输入 0 通道
	ISDA	D	ISP/ICD 串行数据
PA1/TK1/ISCK	PA1	D	通用 I/O 端口
	TK1	A	触摸按键输入 1 通道
	ISCK	D	ISP/ICD 串行时钟
PA2/TK2	PA2	D	通用 I/O 端口
	TK2	A	触摸按键输入 2 通道
PB4/AIN8/SCL/TX/ISCK	PB4	D	通用 I/O 端口
	AIN8	A	ADC 模拟通道 8
	SCL	D	I2C 时钟端口
	TX	D	UART 发送端
	ISCK	D	ISP/ICD 串行时钟
PB5/AIN9/SDA/RX/ISDA	PB5	D	通用 I/O 端口
	AIN9	A	ADC 模拟通道 9
	SDA	D	I2C 数据端口
	RX	D	UART 接收端口
	ISDA	D	ISP/ICD 串行数据
PC0/Cx	PC0	D	通用 I/O 端口
	Cx	A	触控按键参考电容
VDD	VDD	P	芯片电源
VSS	VSS	P	芯片地

表 1-1 管脚说明

注 1: A = 模拟端口, D = 数字端口, P = 电源端口;

## 第 2 章 内核特性

### 2.1 CPU内核概述

- ◇ 采用 ES7P-V2 RISC CPU 内核，79 条精简指令集
- ◇ 采用 2T 架构，系统时钟最高支持 16MHz，最小指令周期 125ns
- ◇ 支持中断优先级和中断向量表
- ◇ 程序指针硬件堆栈共享 SRAM 空间，堆栈级数用户可配置，最大 32 级

### 2.2 硬件乘法器和硬件除法器

芯片指令集不包含乘/除法指令，内部集成独立的硬件乘法器和硬件除法器，通过读写相应寄存器进行操作。

硬件乘法器完成 8 位被乘数 MULA 与 8 位乘数 MULB 的相乘操作，在 MULA 和 MULB 写入完成的 1 个机器周期内将 16 位结果高、低 8 位分别存储于 2 个 8 位结果寄存器 MULH 和 MULL。

硬件除法器完成 32 位被除数 DIVE（由 4 个 8 位寄存器 DIVEU, DIVEH, DIVEM, DIVEL 组成）除以 16 位除数 DIVS（由 2 个 8 位寄存器 DIVSH, DIVSL 组成）的操作，所得结果 32 位商存储于 32 位寄存器 DIVQ（由 4 个 8 位寄存器 DIVQU, DIVQH, DIVQM, DIVQL 组成），16 位余数存储于 16 位寄存器 DIVR（由 2 个 8 位寄存器 DIVRH, DIVRL 组成）。

硬件除法器为多节拍除法器，完成 1 次除法运算最多需要 36 个机器周期（VGO 等于 1 的时间）。写入除数和被除数后，通过控制寄存器启动除法运算，除法运算完成后结果自动载入相应结果寄存器，同时触发除法完成中断 DIVIF。用户可通过相应状态位判断运算过程是否出错。

### 2.3 特殊功能寄存器

PCRL: 程序计数器低 8 位								
Bit	7	6	5	4	3	2	1	0
Name	PCR<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PCR<7:0>: 程序计数器低 8 位

PCRH: 程序计数器高 8 位								
Bit	7	6	5	4	3	2	1	0
Name	PCR<15:8>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PCR<15:8>: 程序计数器高 8 位

PSW: 程序状态字寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	UF	OF	N	OV	Z	DC	C
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	x	x	x	x	x

- Bit 7            保留
- Bit 6            UF: 程序出栈溢出标志位  
                  0: 程序出栈未溢出  
                  1: 程序出栈溢出
- Bit 5            OF: 程序压栈溢出标志位  
                  0: 程序压栈未溢出  
                  1: 程序压栈溢出
- Bit 4            N: 负数标志位  
                  0: 正数  
                  1: 负数
- Bit 3            OV: 有符号运算溢出标志位  
                  0: 无溢出  
                  1: 溢出
- Bit 2            Z: 零标志位  
                  0: 算术或逻辑运算的结果不为零  
                  1: 算术或逻辑运算的结果为零
- Bit 1            DC: 半进位或半借位标志位  
                  0: 低四位无进位或低四位有借位  
                  1: 低四位有进位或低四位无借位
- Bit 0            C: 全进位或全借位标志位  
                  0: 无进位或有借位  
                  1: 有进位或无借位

注 1: 仅部分指令可对 PSW 寄存器进行写操作, 包括 JDEC、JINC、SWAP、BCC、BSS、BTT、MOVA 和 SETR。其它指令对 PSW 寄存器的写操作, 只根据运行结果影响相应状态标志位。

注 2: OF 和 UF 位为只读标志位, 仅上电复位、复位指令和 N\_MRST 复位会将其清零, 其它复位不影响该两位标志位。

AREG: 累加器 A 寄存器								
Bit	7	6	5	4	3	2	1	0
Name	AREG<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0            AREG<7:0>: 累加器的值

MULA: 乘数 A 寄存器								
Bit	7	6	5	4	3	2	1	0
Name	MULA<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 MULA<7:0>: 乘数 A

MULB: 乘数 B 寄存器								
Bit	7	6	5	4	3	2	1	0
Name	MULB<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 MULB<7:0>: 乘数 B

MULL: 乘积低 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	MULL<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0 MULL<7:0>: 乘积低 8 位

MULH: 乘积高 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	MULH<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0 MULH<7:0>: 乘积高 8 位

DIVEL: 被除数低 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVE<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVE<7:0>: 被除数低 8 位

DIVEM: 被除数次低 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVE<15:8>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVE<15:8>: 被除数次低 8 位

DIVEH: 被除数次高 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVE<23:16>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVE<23:16>: 被除数次高 8 位

DIVEU: 被除数高 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVE<31:24>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVE<31:24>: 被除数高 8 位

DIVSL: 除数低 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVSL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVSL<7:0>: 除数低 8 位

DIVSH: 除数高 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVSH<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVSH<7:0>: 除数高 8 位

DIVQL: 商低 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVQ<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVQ<7:0>: 商低 8 位

DIVQM: 商次低 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVQ<15:8>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0 DIVQ<15:8>: 商次低 8 位

DIVQH: 商次高 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVQ <23:16>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0      DIVQ <23:16>: 商次高 8 位

DIVQU: 商高 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVQ<31:24>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0      DIVQ<31:24>: 商高 8 位

DIVRL: 余数低 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVRL<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0      DIVRL<7:0>: 余数低 8 位

DIVRH: 余数高 8 位寄存器								
Bit	7	6	5	4	3	2	1	0
Name	DIVRH<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0      DIVRH<7:0>: 余数高 8 位

DIVC: 除法控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	SIGNED	—	—	VERR	VGO
R/W	—	—	—	R/W	—	—	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~5      保留

Bit4      SIGNED: 有符号除法控制位  
           0: 进行无符号运算 (默认为无符号运算)  
           1: 进行有符号运算

Bit3~2      保留

Bit 1      VERR: 除法运算错误标志位  
           0: 无运算错误  
           1: 运算错误 (除数为 0, 或运算异常终止)

Bit 0      VGO: 除法运算触发位

- 0: 除法运算已完成
- 1: 写 1 启动除法运算，完成后硬件自动清 0

注：当除数为 0 或异常终止时，VERR 标志位置 1，此时读到的商值为 0xFFFF\_FFFF，余数寄存器值为 0xFFFF。  
必须通过软件将 VERR 标志清零才可以进行下次除法运算。

### 第 3 章 存储资源

#### 3.1 概述

本芯片采用哈佛总线架构，程序寻址空间和数据寻址空间相互独立。

片内存储器资源包括：

- ◇ 8K 字 FLASH 程序存储器
- ◇ 程序存储器中可配置 1K 字作为 FLASH 数据存储区
- ◇ 512 字节 SRAM 数据存储区。

其中 FLASH 程序存储器和 FLASH 数据存储区被映射到程序寻址空间，SRAM 数据存储区被映射到数据寻址空间。

#### 3.2 程序寻址空间映射

程序寻址空间包括 2 个部分：

8K 字 FLASH 程序存储器：存储用户程序，支持 PC、查表和 IAP 访问。由配置字 DPAGES<0>配置 FLASH 数据存储区，根据不同的 FLASH 数据存储区，程序寻址空间映射分为以下 2 种：

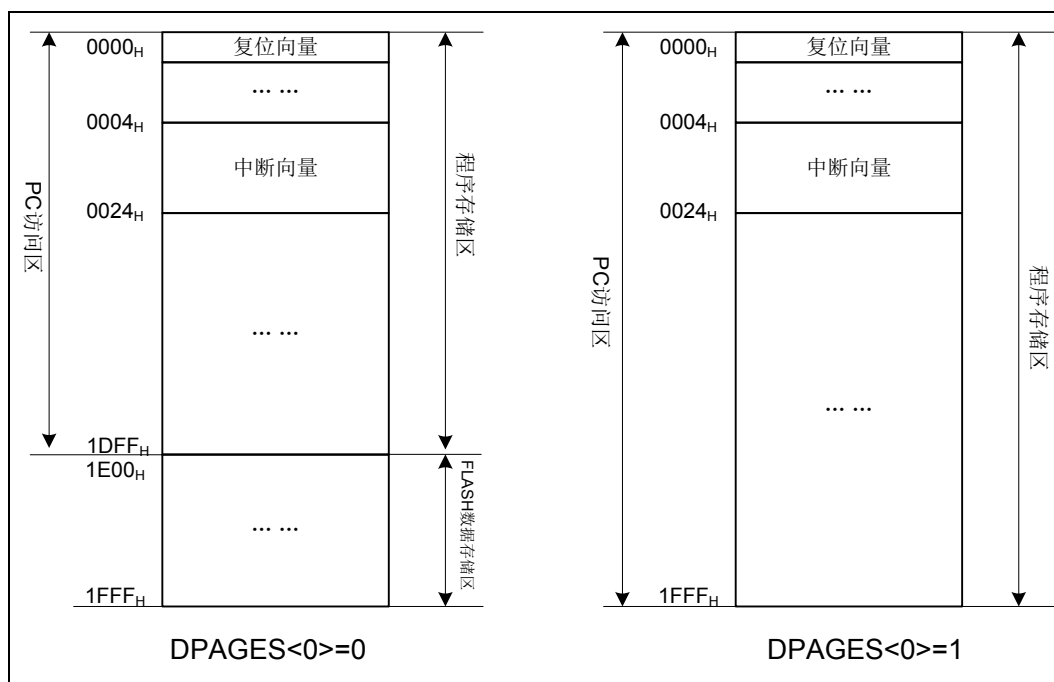


图 3-1 程序寻址空间映射图

注：当配置了 FLASH 数据存储区后，应用程序禁止超出程序区，以免引起误操作。



### 3.3 FLASH程序存储器

#### 3.3.1 概述

FLASH 程序存储器用于存储用户程序。由于芯片指令位宽为 16 位（2 个字节），因此 8K 字 FLASH 程序存储器被映射到程序寻址空间的 8K 地址空间 0000H~1FFFH，每个访问地址对应 16 位宽（2 个字节）的存储单元。通过 13 位程序计数器 PC 进行程序寻址访问。

FLASH 程序存储器可通过 ISP 接口或 IAP 操作进行擦除和编程，擦除以页为单位（256 个地址单元，即 256 字），编程以字为单位（1 个地址单元）。

#### 3.3.2 程序计数器（PC）

程序计数器中存放的是要执行的下一条指令的地址。CPU 运行时，PC 在每个指令周期后都会自动加 1，除非 PC 的值被指令或中断异常改写。程序计数器 PC 寻址超出地址范围会导致 PC 循环（又从 0000H 开始访问）。程序计数器 PC 的低 8 位 PC<7:0>可通过 PCRL 直接读写，而 PC 高 5 位不能直接读写，只能通过 PCRH 寄存器来间接赋值（如 RCALL、CALL、GOTO 等指令）。复位时，PCRL、PCRH 和 PC 都会被清零。PC 硬件堆栈操作不会影响 PCRH 的值。

各种指令对 PC 的影响：

1. 通过指令直接修改 PC 值时，对 PCRL 为目标寄存器的操作可直接修改 PC<7:0>，即 PC<7:0>=PCRL<7:0>；而操作 PC<7:0>的同时也会执行 PC<12:8>=PCRH<4:0>，因此，修改 PC 时，应先修改 PCRH<4:0>，再修改 PCRL<7:0>。
2. 执行 RCALL 指令时，PC<7:0>为寄存器 R 中的值；而 PC<12:8>=PCRH<4:0>。
3. 执行 CALL，GOTO 指令时，PC<12:0>低 11 位为指令中 11 位立即数，而 PC<12>=PCRH<4>。
4. 执行 LCALL 指令时，该指令为双字指令共有 16 位立即数（操作数）。PC<12:0>被修改为该 16 位立即数的值的低 13 位；同时 PCRH<4:0>被修改为 I<12:8>的值。
5. 执行 AJMP 指令时，该指令为双字指令共有 16 位立即数（操作数）。PC<12:0> 被修改为该 16 位立即数的值的低 13 位，同时 PCRH<4:0>修改为 I<12:8>的值。
6. 执行 PAGE 指令时，PCRH<4:3>的值将被该指令的立即数 I<1:0>替换。
7. 执行其他指令时，PC 值自动加 1。

#### 3.3.3 硬件堆栈

PC 硬件堆栈用于子程序调用或中断处理时存储程序返回地址。SRAM 数据存储器的一部分被用于实现 PC 硬件堆栈，PC 硬件堆栈的级数可通过芯片配置字 STKLS 进行配置。

执行 CALL 或 LCALL 指令或中断被响应后，PC 自动压栈保护；当执行 RET、RETIA 或 RETIE 指令时，堆栈会将最近一次压栈的值恢复至 PC。超过堆栈级数的压栈被视为压栈溢出，超过已压栈级数的退栈被视为退栈溢出。压栈溢出和退栈溢出均会导致程序失控。芯片复位后，堆栈指针将重新指向堆栈的初始栈顶。

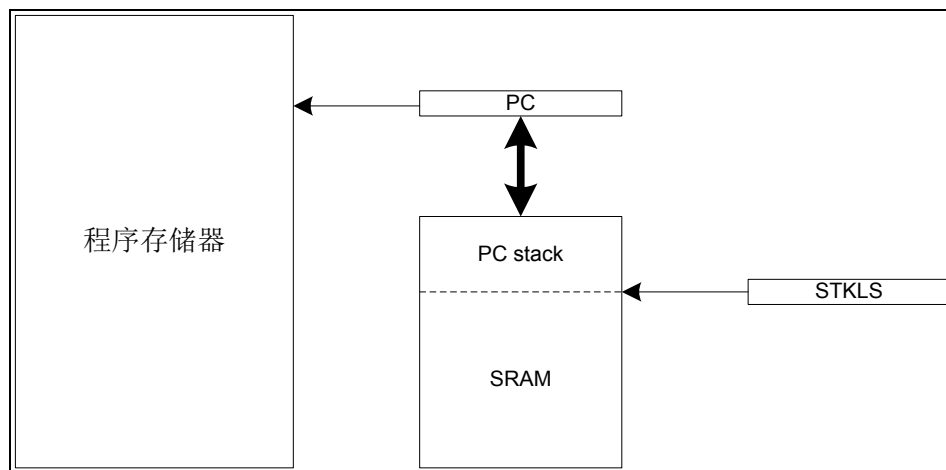


图 3-2 堆栈示意图

### 3.3.4 FLASH存储器的查表读和IAP操作

#### 3.3.4.1 概述

FLASH 存储器支持查表读和 IAP 页擦除和编程操作。

FLASH 存储器的 IAP 擦除操作是以页（Page）为单位，一页为 256 个地址单元，页地址对应到 FRAH<7:1>, 如 FRAH<7:1>=00H, 为第 0 页; FRAH<7:1>=01H, 为第 1 页; ...; FRAH<7:1>=1FH, 为第 31 页。FLASH 存储器的读出和 IAP 写入操作以 1 个地址单元为单位，通过 FRA（FRAH, FRAL）寻址。

必须对 IAPUL 寄存器写入 0xA5 来解锁 IAP 操作，即使能 IAPERSS, IAPPRGS 以及 IAPC 寄存器的写访问。IAP 编程或擦除操作后不会自动上锁，需要用户在其所需要的编程或擦除结束后，向 IAPUL 写入非 0xA5 值来上锁，防止误操作改写 FLASH 内容。

当 FLASH 存储器进行 IAP 擦除或 IAP 写入操作时，CPU 内核暂停执行，外设可按预设状态继续运行，外设的中断请求将置位相应的中断标志。当 IAP 擦除或 IAP 写入操作完成时，CPU 内核恢复执行。

#### 3.3.4.2 FLASH存储器的查表读

本芯片的程序存储器查表只支持查表读操作，1 次查表读操作访问 1 个地址单元的 16 位数据。当芯片配置字中 IAP 操作使能位 IAPEN（CFG\_WORD0 <10>）使能时，通过查表读指令将查表地址寄存器 FRA（FRAH, FRAL）所指向的程序存储器中的 16 位数据读入 ROMD（ROMDH, ROMDL）中。

设置了 FLASH 全加密，仍可通过 TBR 指令进行访问。

本芯片查表写指令保留未用（执行时仅影响 FRA 寄存器）。

#### 应用例程：程序存储器查表读。

```

MOVI    0x05           ; 读取程序存储器 0105H
MOVA    FRAL
MOVI    0X01
MOVA    FRAH
    
```

```
TBR
MOV    ROMDH, 0
... ..
MOV    ROMDL, 0
... ..
```

### 3.3.4.3 FLASH存储器的IAP擦除

通过 IAP 进行 FLASH 擦除的最小操作单位为页，一页为 256 个地址单元。页通过寄存器 FRAH<7:1>寻址。对 IAPERSS 寄存器写入 0xAE 来启动擦除。

若芯片配置字的 LVDIAPTEN 设置为 1，在启动 IAP 擦除时，若 LVD 有效则无法启动擦除；在擦除过程中，若 LVD 有效则硬件自动终止当前的页擦操作。为了防止电压低于 LVD 设定门限时，发生误擦除，在 IAP 擦除时，建议设置寄存器位 LVDFLTEN=0，禁止 LVD 滤波。

为保证可靠地访问 FLASH，可设定终止页擦后到 FLASH 可访问的延时等待时间，由 IAPC 寄存器的 ERSTWT 位进行设置。还可启动 T10 作为页擦超时计数器，通过 IAPC 寄存器的 TOEN 位来使能超时发生时，终止 IAP 擦除操作，使用 T10 之前必需设定合适的超时时间，对页擦操作需大于 2ms，而且必须清除 T10 中断标志。

IAP 擦除流程图如下图所示，该流程图假设配置字的 LVDIAPTEN=1。

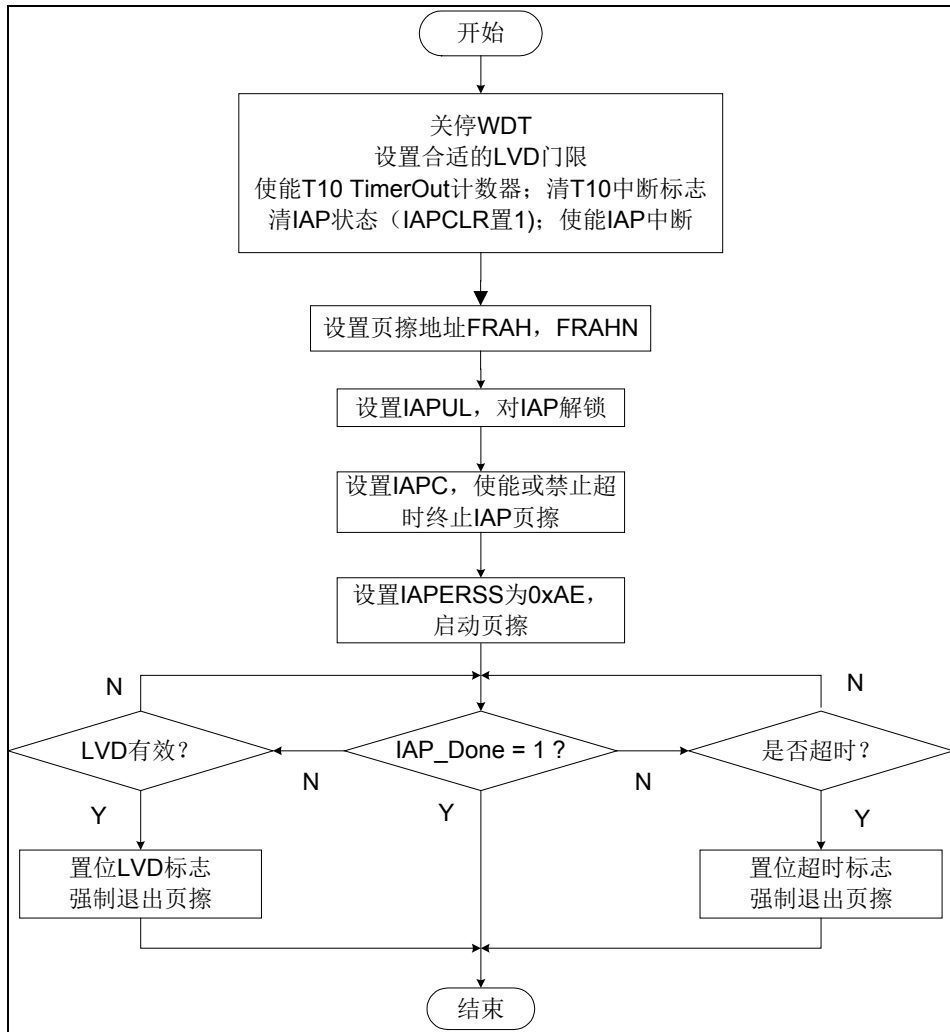


图 3-3 页擦流程图

**应用例程：FLASH 页擦除。**

```

MOVI    0xA5          ; 解锁 WDT 寄存器
MOVA    WDTUL
BCC     WDT, WDTEN   ; 关 WDT, 使 WDT 停止计数, 以免溢出复位 IAP 操作

MOVI    0X0A          ; 擦除 FLASH 第 6 页 (0A00H~0BFFH)
MOVA    FRAH
MOVI    0XF5
MOVA    FRAHN
BSS     INTE4, IAPIE  ; 使能 IAP 中断

BSS     IAPC, 1       ; 清除 IAP 状态 (必须在执行下一个 IAP 操作
                      ; 之前清除 IAP 状态)

MOVI    0xA5          ; 解锁 IAP
MOVA    IAPUL
MOVI    0xAE
  
```

MOVA IAPERSS ; 启动 FLASH IAP 擦除操作  
; CPU 进入暂停状态, 直到擦除完成后 CPU 恢复运行  
; 一次页擦除时间约 2ms, 具体时间随操作电压和操作温度有所不同

ERS\_WAIT

JBS IAPS, 1 ; 等待擦除结束  
GOTO ERS\_WAIT

MOVI 0x00 ; 锁定 IAP  
MOVA IAPUL

MOVI 0xA5 ; 解锁 WDTC 寄存器  
MOVA WDTUL

BSS WDTC, WDTEN ; WDT 恢复计数 (如果配置字 WDTEN=1)

BCC INTE4, IAPIE ; 开中断

.....

注 1: 在向 IAPERSS 寄存器写入 0xAE 启动 IAP FLASH 页擦之后, 由于 CPU 会继续执行下一条指令, 建议启动页擦后, 紧接的下一条指令为 NOP 指令或如上例所示, 为 IAP 状态查询指令。

注 2: 在 IAP 中断服务程序中, 可通过查询 IAPS 状态寄存器来判断执行其它子程序。

- 如果 FRAH\_ERR, FRAL\_ERR 或 ROMD\_ERR 错误, 则重新写 FRA 或 ROMD 的正反寄存器, 直至成功。
- 如果 ERSS\_ERR 错误, 则: 程序调试阶段, 查阅代码中的启动代码是否写错, 若是则修改错误; 否则再次写 IAPERSS 寄存器, 直至成功。
- 如果 IAP\_LVDIF 为 1, 则停止 IAP 操作, 直至该标志为 0。

### 3.3.4.4 FLASH存储器的IAP编程

FLASH 存储器中已擦除的地址单元支持 IAP 编程操作。IAP 编程操作通过 IAPPRGS 控制寄存器, 将 ROMD (ROMDH, ROMDL) 中的内容写入 FRA (FRAH, FRAL) 指向的 FLASH 地址单元。

对 IAPUL 写入 0xA5 解锁 IAP 后, 可对 IAPPRGS 寄存器写入 0xD5 启动编程。编程结束后需对 IAPC 寄存器的 IAPCLR 置 1 来清编程状态。

数据和地址更新后需重新对 IAPPRGS 写入 0xD5 以启动编程。每次编程结束之后可以对 IAPUL 写入非 0xA5 上锁, 亦可保持解锁状态, 直至所有编程已结束再对 IAP 上锁。

在启动 IAP 编程时, 若 LVD 有效则无法启动编程。在编程操作过程中, 若 LVD 有效, 且当芯片配置字的 LVDIAPTEN 设置为 1 时, 则终止当前的编程操作。为了防止电压低于 LVD 设定门限时, 发生误编程, 在 IAP 编程时, 建议设置寄存器位 LVDFLTEN=0, 禁止 LVD 滤波。

为保证可靠地访问 FLASH, 可设定终止编程后到 FLASH 可访问的延时等待时间, 由 IAPC 寄存器的 PRGTWT 位进行设置。还可启动 T10 作为编程超时计数器, 通过 IAPC 寄存器的 TOEN 位来使能超时发生时, 终止当前的 IAP 编程操作。使用 T10 之前必需设定合适的超时时间, 对编程操作需大于 20us, 而且必须清除 T10 中断标志。

IAP 编程流程图 3-4 所示。

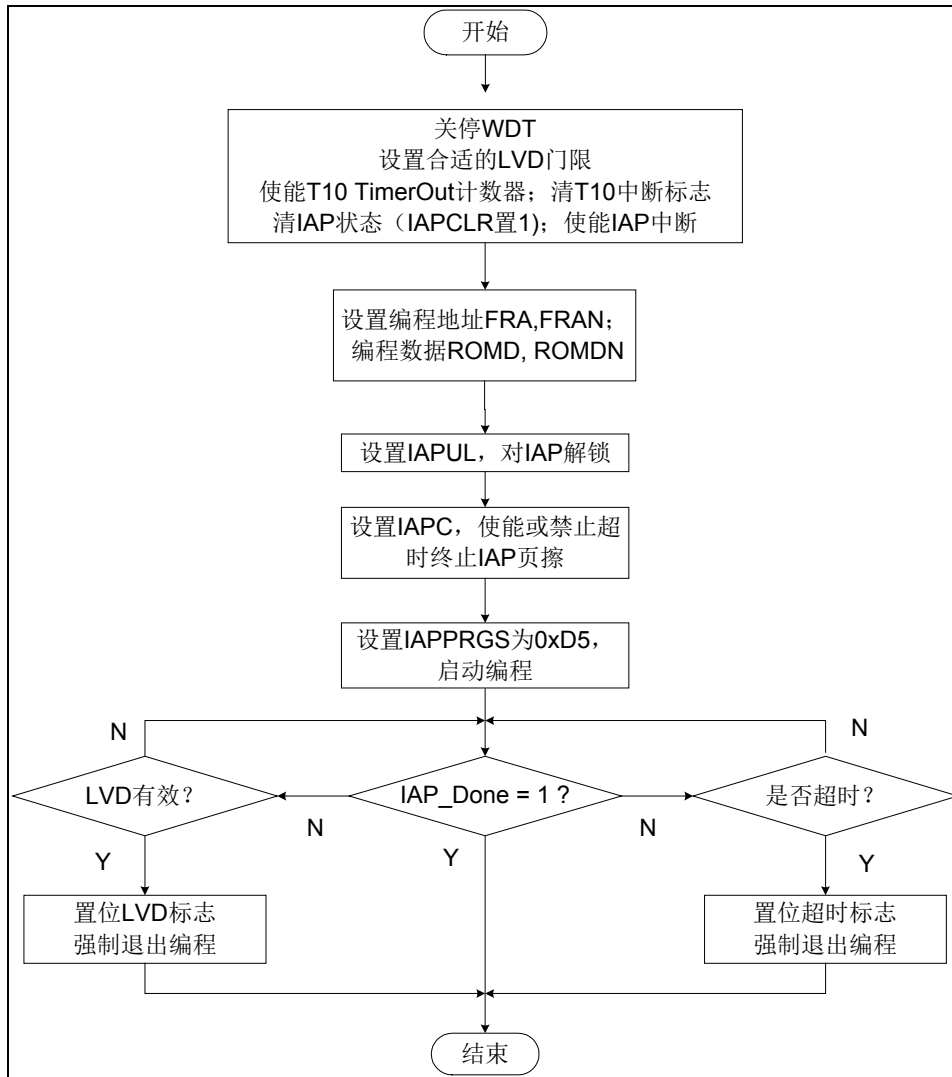


图 3-4 编程流程图

**应用例程：FLASH IAP 编程**

```

MOVl    0xA5          ; 解锁 WDTC 寄存器
MOVA    WDTUL

BCC     WDTC, WDTEN  ; 关 WDT, 使 WDT 停止计数, 以免溢出复位 IAP 操作

MOVl    0x05          ; 将 55AAH 写入 FLASH 的 0500H 地址单元
MOVA    FRAH
MOVl    0x00
MOVA    FRAL
MOVl    0xFA          ; 将地址的反 FAFH 写入 FRAHN 和 FRALN
MOVA    FRAHN
MOVl    0xFF
MOVA    FRALN
  
```

```

MOVI    0x55
MOVA    ROMDH
MOVI    0xAA
MOVA    ROMDL
MOVI    0xAA           ; 将数据的反 AA55H 写入 ROMDHN 和 ROMDLN
MOVA    ROMDHN
MOVI    0x55
MOVA    ROMDLN
    
```

```

BSS     INTE4, IAPIE   ; 使能 IAP 中断
    
```

```

BSS     IAPC,1         ; 清 IAP 状态以启动下一个 IAP 操作
MOVI    0xA5           ; 解锁 IAP
MOVA    IAPUL
    
```

```

MOVI    0xD5
MOVA    IAPPRGS       ; 启动 FLASH IAP 编程操作
NOP
    
```

; CPU 进入暂停状态，直到编程完成后 CPU 恢复运行

; 一次页编程时间约 20us，具体时间随操作电压和操作温度有所不同

PRG\_WAIT

```

JBS     IAPS,1         ; 等待 IAP 编程操作结束
GOTO    PRG_WAIT
    
```

```

MOVI    0x00           ; 锁定 IAP
MOVA    IAPUL
    
```

```

MOVI    0xA5           ; 解锁 WDTIC 寄存器
MOVA    WDTUL
    
```

```

BSS     WDTIC, WDTEN   ; WDT 恢复计数（如果配置字 WDTEN=1）
    
```

```

BCC     INTE4, IAPIE   ; 关 IAP 中断
    
```

.....

注 1: 对未进行擦除操作的 FLASH 存储器单元进行编程操作，会导致该 FLASH 存储器单元的存储数据被破坏。

注 2: 在向 IAPPRGS 寄存器写入 0xD5 启动 IAP FLASH 编程之后，由于 CPU 会继续执行下一条指令，该指令不能为 TBR 指令，以避免 ROMD 数据被 TBR 指令所读取的数据覆盖，而导致数据写入失败。建议启动编程后，紧接的下一条指令为 NOP 指令或如上例所示，为 IAP 状态查询指令。

注 3: 在 IAP 中断服务程序中，可通过查询 IAPS 状态寄存器来判断执行其它子程序：

- 如果 FRAH\_ERR, FRAL\_ERR 或 ROMD\_ERR 错误，则重新写 FRA 或 ROMD 的正反寄存器，直至成功。
- 如果 PRGS\_ERR 错误，则：程序调试阶段，查阅代码中的启动代码是否写错，若是则修改错误；否则再次写 IAPPRGS 寄存器，直至成功。
- 如果 IAP\_LVDIF 为 1，则停止 IAP 操作，直至该标志为 0。

### 3.3.4.5 特殊寄存器

IAPUL: IAP 解锁寄存器								
Bit	7	6	5	4	3	2	1	0
Name	IAPUL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7-0 IAPUL<7:0>: IAP 解锁寄存器  
写入 0xA5 解锁 IAP。IAP 解锁后, 才可对 IAPERSS, IAPPRGS 寄存器进行写操作。  
若解锁成功, 读出值为 0x01, 否则为 0x00。

注: 只需向 IAPUL 写入非 0xA5 值, 即可完成上锁。

IAPERSS: 擦除启动控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	IAPERSS<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7-0 IAPERSS<7:0>: IAP 擦除启动控制寄存器  
写入 0xAE 启动页擦; 若页擦启动成功, 则读出值为 0x01, 否则为 0x00。  
IAPERSS 可有两种方式清零: IAP 页擦成功后自动清零; 若页擦失败, 可将 IAPC 控制寄存器的 IAPCLR 置 1 来清零。  
该寄存器不允许写入除 0xAE 外的任何值, 否则 IAPS 状态寄存器的 ERSS\_ERR 标志位将被置 1, 在 IAP 中断使能时将触发中断。

注 1: 写入 0xAE 后, 可读取 IAPERSS 寄存器, 若值为 0x01 则表示成功启动; 否则启动失败。

注 2: 启动页擦失败的情况: 工作电压小于设定的 LVD 阈值电压 (配置位 LVDIAPTEN=1); FRA 和 FRAN 寄存器值非反码, 或向 IAPERSS 寄存器写入了非 0xAE 值。页擦启动失败可触发 IAP 中断, 可查询 IAPS 状态寄存器来判断中断源: 地址错, 或因产生了有效的 LVD 信号; 若非这两者, 则为写入了非 0xAE 值或待擦除页已被分区加密。

IAPPRGS: 编程启动控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	IAPPRGS<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7-0 IAPPRGS<7:0>: IAP 编程启动控制寄存器  
写入 0xD5 启动编程; 若编程启动成功, 则读出值为 0x01, 否则为 0x00  
IAPPRGS 可有两种方式清零: IAP 编程成功后自动清零; 若编程失败, 可将 IAPC 控制寄存器的 IAPCLR 置 1 来清零。  
该寄存器不允许写入除 0xD5 外的任何值, 否则 IAPS 状态寄存器的 PRGS\_ERR 标志位将被置 1, 在 IAP 中断使能时将触发中断。

注 1: 写入 0xD5 后, 可读取 IAPPRGS 寄存器, 若值为 0x01 则表示成功启动; 否则启动失败。

注 2: 启动编程失败的情况: 工作电压小于设定的 LVD 阈值电压 (配置位 LVDIAPTEN=1); FRA 和 FRAN 寄存器



值非反码，或 ROMD 和 ROMDN 寄存器的值非反；写入值非 0xD5。编程启动失败可触发 IAP 中断，可查询 IAPS 寄存器标志位判断中断源：地址错，数据错或产生了有效的 LVD 信号；若非这三者，则为写入了非 0xD5 值或待编程地址已被分区加密。

IAPC: IAP 控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	ERSTWT<3:0>				PRGTWT<1:0>		IAPCLR	TOEN
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	1	1	0	0	0	0	0

Bit7~4 ERSTWT<3:0>: 擦除终止等待时间 (erase terminated wait time)

- 0000~0101: 218.75us
- 0110: 218.75us (7 个 32KHz 时钟周期)
- 0111: 250us (8 个 32KHz 时钟周期)
- 1000: 281.25us (9 个 32KHz 时钟周期)
- 1001: 312.5us (10 个 32KHz 时钟周期)
- 1010: 343.75us (11 个 32KHz 时钟周期)
- 1011: 375us (12 个 32KHz 时钟周期)
- 1100: 406.25us (13 个 32KHz 时钟周期)
- 1101: 437.5us (14 个 32KHz 时钟周期)
- 1110: 468.75us (15 个 32KHz 时钟周期)
- 1111: 500us (16 个 32KHz 时钟周期)

Bit3~2 PRGTWT<1:0>: 编程终止等待时间 (program terminated wait time)

- 00: 31.25us (1 个 32KHz 时钟周期)
- 01: 62.5us (2 个 32KHz 时钟周期)
- 10: 93.75us (3 个 32KHz 时钟周期)
- 11: 125us (4 个 32KHz 时钟周期)

Bit1 IAPCLR: IAP 状态清除

- 0: 禁止
- 1: 清除 IAP 状态 (该位置 1 后自动清零)

Bit 0 TOEN: Timeout 超时使能寄存器 (timer-out enable)

- 0: 禁止
- 1: 使能

注 1: 进入 IAP 模式之前，可设置 T10 为 IAP 擦除或编程的 timeout 计数器。

例如: 页擦除操作，正常完成擦除约需 2ms，因此可设置 t10 prescaler 为 128 分频，16MHz 时钟二分频，timeout 时间长度约 4.096ms。如果超出该时间仍未完成擦除操作，则 IAP 操作将被强制退出。

注 2: 每次完成 IAP 操作之后必须将 IAPCLR 置 1 来清除当前状态，才允许下一次 IAP 页擦除或编程操作。该位置 1 后自动清零。该位可清除擦除启动失败、编程启动失败、擦除/编程超时以及 IAP 编程/擦除终止标志等。

注 3: 进入 IAP 模式之前，可设置 T10 为 IAP 擦除或编程的 timeout 计数器。

IAPS: IAP 状态寄存器								
Bit	7	6	5	4	3	2	1	0
Name	FRAH_ERR	FRAL_ERR	ROMD_ERR	ERSS_ERR	PRGS_ERR	IAP_TIMEOUT	IAP_DONE	IAP_LVDIF
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	1	0	0	0	0	0

- Bit 7      FRAH\_ERR: FRA 高位地址出错标志。写入正确的 FRAH 正反值清零  
             0: 写入的 FRAH 与 FRAHN 值为反码  
             1: 写入的 FRAH 与 FRAHN 值不为反码
- Bit6      FRAL\_ERR: FRA 低位地址出错标志。写入正确的 FRAL 正反值清零  
             0: 写入的 FRAL 与 FRALN 值为反码  
             1: 写入的 FRAL 与 FRALN 值不为反码
- Bit5      ROMD\_ERR: ROMD 出错标志。写入正确的 ROMD 正反值清零  
             0: 写入的 ROMD 和 ROMDN 值为反码  
             1: 写入的 ROMD 和 ROMDN 值不为反码
- Bit4      ERSS\_ERR: 页擦启动出错标志。该位需要 IAPCLR 位置 1 来清零或向 IAPERSS 寄存器写入正确的 0xAE 数据启动页擦来清零。  
             0: 页擦未启动, 或已启动成功  
             1: 页擦启动失败。因 LVD 中断、FRAH 和 FRAHN 地址非反、以及 IAPERSS 寄存器写入数据非 0xAE 等而启动失败
- Bit3      PRGS\_ERR: 编程启动出错标志。该位需要 IAPCLR 位置 1 来清零或向 IAPPRGS 寄存器写入正确的 0xD5 数据启动编程来清零。  
             0: 编程未启动, 或已启动成功  
             1: 编程启动失败。因 LVD 中断、FRA 和 FRAN 地址非反、ROMD 和 ROMDH 数据非反, 以及 IAPPRGS 寄存器写入数据非 0xD5 等而启动失败
- Bit 2      IAP\_TIMEOUT: IAP 编程/擦除超时标志。该位需要 IAPCLR 位置 1 来清零。  
             0: 编程/擦除操作正常  
             1: 编程/擦除操作超时 (软件清零, 或启动 IAP 编程/擦除操作时自动清零)
- Bit 1      IAP\_DONE: IAP 编程/擦除状态位。该位需要 IAPCLR 位置 1 来清零。  
             0: 编程/擦除操作未完成  
             1: 编程/擦除操作已完成
- Bit 0      IAP\_LVDIF: IAP 因电压低于 LVD 阈值设定而终止标志位。  
             该位在工作电压大于 LVD 阈值电压后自动清零。  
             0: IAP 操作正常执行  
             1: IAP 操作因电压低于 LVD 阈值电压而终止

注 1: 如果编程/擦除超时, 自动退出 IAP 操作。可通过查询 IAP\_TIMEOUT 位来判断是否发生了 timeout 而导致 IAP 退出。

注 2: 可通过查询 IAPS 的其它标志位来判断 IAP 操作失败的原因。  
 注 3: FRAH\_ERR, FRAL\_ERR, ROMD\_ERR 等标志位上电后因 FRA 与 ROMD 的正反寄存器不满足正反关系, 初值为 1, 需通过写入正确的 FRA 和 ROMD 正反值清零。

### 3.4 在线编程ISP和在线调试ICD

FLASH 存储器具有可重复烧写的功能，便于客户代码和数据的更新升级。为了实现开发人员在开发过程中能够更轻松地进行代码的调试、更新、升级，本芯片还支持在线编程 ISP 和在线调试 ICD，用户只需在电路系统板上引出五根连接编程和调试接口线，即可实现程序的重烧录和调试，更加方便高效。

MRSTN 配置为 GPIO 使用时，用户只需在电路系统板上引出四根连接编程和调试接口线，但芯片必须有上电的过程，才能进入 ISP 模式。

芯片管脚	编程器管脚	管脚说明
ISDA	ISDA	编程/调试串行数据端口
ISCK	ISCK	编程/调试串行时钟端口
VSS	VSS	地
VDD	VDD	电源
MRSTN	MRSTN	复位脚

表 3-1 在线编程/调试管脚说明

注 1: 对编程/调试接口中的 ISCK 和 ISDA 管脚，芯片支持两组管脚可选择，其中 PB4 和 PB5 作为一组编程/调试接口，PA0 和 PA1 作为另一组编程/调试接口；

注 2: 上述两组接口，用作编程使用时，无需进行选择设置，芯片自动识别有效使用的编程接口；用作调试使用时，通过配置字 ICDEB 使能，并由配置字 ICDSEL 进行选择设置。

### 3.5 数据寻址空间

#### 3.5.1 概述

- ◇ 数据寻址空间由 2 部分组成
  - 分页访问存储区
  - 快速访问存储区
- ◇ 物理存储包括
  - 512 字节 SRAM 数据存储器（部分用于 PC 硬件堆栈）
  - 特殊功能寄存器 SFR
- ◇ 支持 3 种寻址方式
  - 直接寻址
  - GPR 特殊寻址
  - 间接寻址

#### 3.5.2 数据寻址空间映射

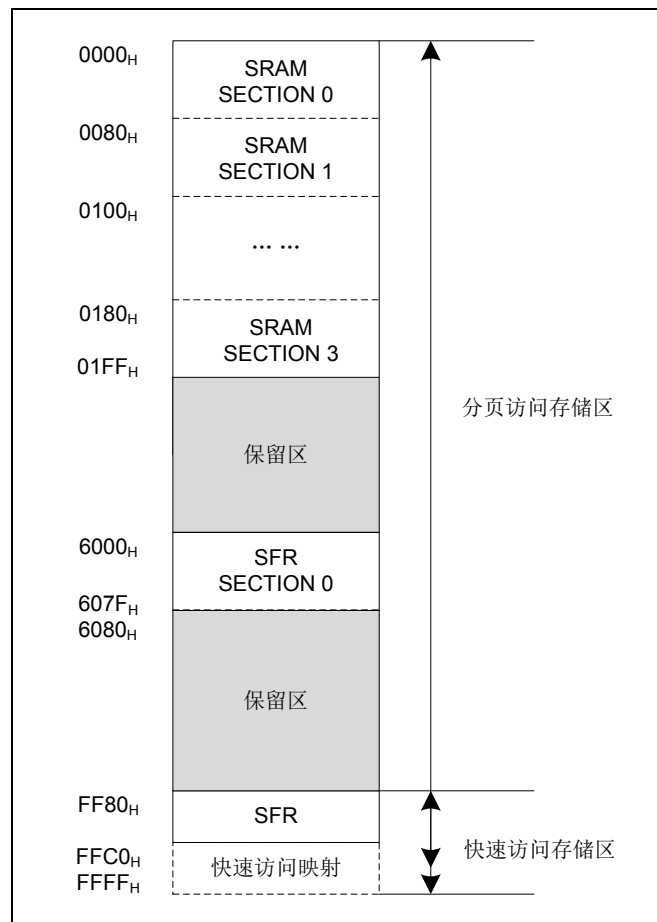


图 3-5 数据寻址空间映射图

### 3.5.3 通用数据存储器SRAM

通用数据存储器被用于临时存放数据和控制信息，可以通过指令进行读写操作。本芯片通用数据存储器空间为 512 字节，分为 4 个存储体组 (SRAM SECTION 0~3)，每个 SRAM 存储体组为 128 字节，地址范围为 0000<sub>H</sub>~01FF<sub>H</sub>。程序指令对 SRAM 进行直接寻址访问时，需通过寄存器 BKSR 选择 SRAM 存储体组，实现在不同存储体组间的切换。通用数据存储器内容在上电复位后是不确定的，未掉电的其它复位后，将保存复位前的内容。

SRAM 的最后 1 个存储体组 SECTION 3，从高地址 (01FF<sub>H</sub>) 开始的部分或全部地址空间可分配用于 PC 硬件堆栈，通过用户配置字 STKLS 选择 PC 硬件堆栈的级数。当 STKLS 选定后，PC 硬件堆栈占用的 SRAM 地址空间是堆栈级数的 2 倍。例如，设定堆栈级数为 16 级，SRAM 存储空间的 01E0<sub>H</sub>~01FF<sub>H</sub> 的 32 个地址被 PC 硬件堆栈占用，数据访问无法访问到该地址空间。两部分空间相互硬件隔离，PC 硬件堆栈的溢出也不会影响堆栈区之外的 SRAM 地址空间。

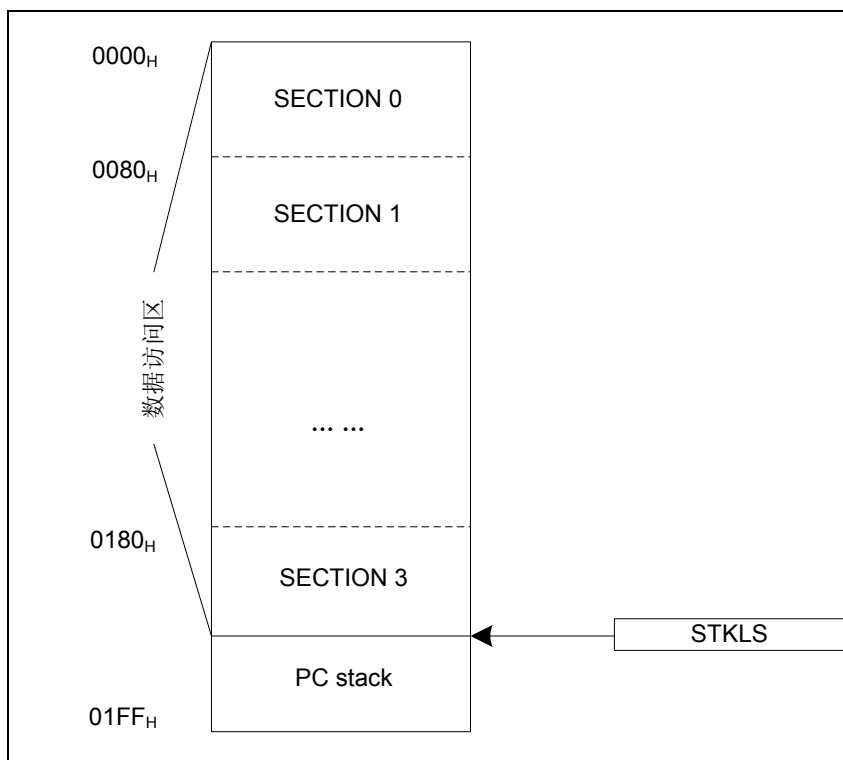


图 3-6 SRAM 地址映射示意图

### 3.5.4 特殊功能寄存器SFR

特殊功能寄存器 SFR 用于片内各功能模块的设置和控制。SFR 被存储于两部分寻址空间。一部分存储于快速访问区的低地址空间 (FF80<sub>H</sub>~FFBF<sub>H</sub>)，共 64 字节。另一部分存储于分页访问区的 SFR SECTION 0 (6000<sub>H</sub>~607F<sub>H</sub>)，共 128 字节，通过设置 FAMR 寄存器，将 SFR SECTION 0 中被选中的区域 (共 64 字节)，映射到快速访问区的高地址空间 (FFC0<sub>H</sub>~FFFF<sub>H</sub>)，然后进行访问。各 SFR 寄存器的功能描述分别在各个章节中描述。

FF80 <sub>H</sub>	IAD	FFA0 <sub>H</sub>	PAPU	FFC0 <sub>H</sub>	Mapped	FFE0 <sub>H</sub>	Mapped
FF81 <sub>H</sub>	IAAL	FFA1 <sub>H</sub>	PAPD	FFC1 <sub>H</sub>	Mapped	FFE1 <sub>H</sub>	Mapped
FF82 <sub>H</sub>	IAAH	FFA2 <sub>H</sub>	PALC	FFC2 <sub>H</sub>	Mapped	FFE2 <sub>H</sub>	Mapped
FF83 <sub>H</sub>	BKSR	FFA3 <sub>H</sub>	PAOD	FFC3 <sub>H</sub>	Mapped	FFE3 <sub>H</sub>	Mapped
FF84 <sub>H</sub>	FAMR	FFA4 <sub>H</sub>	PB	FFC4 <sub>H</sub>	Mapped	FFE4 <sub>H</sub>	Mapped
FF85 <sub>H</sub>	AREG	FFA5 <sub>H</sub>	PBT	FFC5 <sub>H</sub>	Mapped	FFE5 <sub>H</sub>	Mapped
FF86 <sub>H</sub>	PSW	FFA6 <sub>H</sub>	PBS	FFC6 <sub>H</sub>	Mapped	FFE6 <sub>H</sub>	Mapped
FF87 <sub>H</sub>	PWRCWP	FFA7 <sub>H</sub>	PBPU	FFC7 <sub>H</sub>	Mapped	FFE7 <sub>H</sub>	Mapped
FF88 <sub>H</sub>	PWRC	FFA8 <sub>H</sub>	PBPD	FFC8 <sub>H</sub>	Mapped	FFE8 <sub>H</sub>	Mapped
FF89 <sub>H</sub>	PCRL	FFA9 <sub>H</sub>	-	FFC9 <sub>H</sub>	Mapped	FFE9 <sub>H</sub>	Mapped
FF8A <sub>H</sub>	PCRH	FFAA <sub>H</sub>	PBLC	FFCA <sub>H</sub>	Mapped	FFEA <sub>H</sub>	Mapped
FF8B <sub>H</sub>	-	FFAB <sub>H</sub>	PBOD	FFCB <sub>H</sub>	Mapped	FFEB <sub>H</sub>	Mapped
FF8C <sub>H</sub>	-	FFAC <sub>H</sub>	-	FFCC <sub>H</sub>	Mapped	FFEC <sub>H</sub>	Mapped
FF8D <sub>H</sub>	-	FFAD <sub>H</sub>	PC	FFCD <sub>H</sub>	Mapped	FFED <sub>H</sub>	Mapped
FF8E <sub>H</sub>	-	FFAE <sub>H</sub>	PCT	FFCE <sub>H</sub>	Mapped	FFEE <sub>H</sub>	Mapped
FF8F <sub>H</sub>	-	FFAF <sub>H</sub>	PCS	FFCF <sub>H</sub>	Mapped	FFEF <sub>H</sub>	Mapped
FF90 <sub>H</sub>	-	FFB0 <sub>H</sub>	PCPU	FFD0 <sub>H</sub>	Mapped	FFF0 <sub>H</sub>	Mapped
FF91 <sub>H</sub>	-	FFB1 <sub>H</sub>	PCPD	FFD1 <sub>H</sub>	Mapped	FFF1 <sub>H</sub>	Mapped
FF92 <sub>H</sub>	-	FFB2 <sub>H</sub>	PCLC	FFD2 <sub>H</sub>	Mapped	FFF2 <sub>H</sub>	Mapped
FF93 <sub>H</sub>	-	FFB3 <sub>H</sub>	PCOD	FFD3 <sub>H</sub>	Mapped	FFF3 <sub>H</sub>	Mapped
FF94 <sub>H</sub>	INTG	FFB4 <sub>H</sub>	-	FFD4 <sub>H</sub>	Mapped	FFF4 <sub>H</sub>	Mapped
FF95 <sub>H</sub>	INTP	FFB5 <sub>H</sub>	-	FFD5 <sub>H</sub>	Mapped	FFF5 <sub>H</sub>	Mapped
FF96 <sub>H</sub>	INTE0	FFB6 <sub>H</sub>	-	FFD6 <sub>H</sub>	Mapped	FFF6 <sub>H</sub>	Mapped
FF97 <sub>H</sub>	INTF0	FFB7 <sub>H</sub>	-	FFD7 <sub>H</sub>	Mapped	FFF7 <sub>H</sub>	Mapped
FF98 <sub>H</sub>	-	FFB8 <sub>H</sub>	-	FFD8 <sub>H</sub>	Mapped	FFF8 <sub>H</sub>	Mapped
FF99 <sub>H</sub>	-	FFB9 <sub>H</sub>	-	FFD9 <sub>H</sub>	Mapped	FFF9 <sub>H</sub>	Mapped
FF9A <sub>H</sub>	INTE2	FFBA <sub>H</sub>	-	FFDA <sub>H</sub>	Mapped	FFFA <sub>H</sub>	Mapped
FF9B <sub>H</sub>	INTF2	FFBB <sub>H</sub>	-	FFDB <sub>H</sub>	Mapped	FFFB <sub>H</sub>	Mapped
FF9C <sub>H</sub>	INTC	FFBC <sub>H</sub>	-	FFDC <sub>H</sub>	Mapped	FFFC <sub>H</sub>	Mapped
FF9D <sub>H</sub>	PA	FFBD <sub>H</sub>	SCC	FFDD <sub>H</sub>	Mapped	FFFD <sub>H</sub>	Mapped
FF9E <sub>H</sub>	PAT	FFBE <sub>H</sub>	SCSELO	FFDE <sub>H</sub>	Mapped	FFFE <sub>H</sub>	Mapped
FF9F <sub>H</sub>	PAS	FFBF <sub>H</sub>	SCSEL1	FFDF <sub>H</sub>	Mapped	FFFF <sub>H</sub>	Mapped

图 3-7 快速访问区特殊功能寄存器

FAMR[1:0] = 00				FAMR[1:0] = 01			
6000 <sub>H</sub>	FRAL	6020 <sub>H</sub>	-	6040 <sub>H</sub>	TKSELL	6060 <sub>H</sub>	TK8DAH
6001 <sub>H</sub>	FRALN	6021 <sub>H</sub>	-	6041 <sub>H</sub>	TKSELH	6061 <sub>H</sub>	TK9DAL
6002 <sub>H</sub>	FRAH	6022 <sub>H</sub>	-	6042 <sub>H</sub>	TKTUN	6062 <sub>H</sub>	TK9DAH
6003 <sub>H</sub>	FRAHN	6023 <sub>H</sub>	T21CH	6043 <sub>H</sub>	TKCTL0	6063 <sub>H</sub>	TK10DAL
6004 <sub>H</sub>	ROMDL	6024 <sub>H</sub>	T21CM	6044 <sub>H</sub>	TKCTL1	6064 <sub>H</sub>	TK10DAH
6005 <sub>H</sub>	ROMDLN	6025 <sub>H</sub>	-	6045 <sub>H</sub>	TKFCTL	6065 <sub>H</sub>	TK11DAL
6006 <sub>H</sub>	ROMDH	6026 <sub>H</sub>	-	6046 <sub>H</sub>	VRC	6066 <sub>H</sub>	TK11DAH
6007 <sub>H</sub>	ROMDHN	6027 <sub>H</sub>	-	6047 <sub>H</sub>	TKMODL	6067 <sub>H</sub>	-
6008 <sub>H</sub>	IAPUL	6028 <sub>H</sub>	MULA	6048 <sub>H</sub>	TKMODM	6068 <sub>H</sub>	BR0R
6009 <sub>H</sub>	IAPC	6029 <sub>H</sub>	MULB	6049 <sub>H</sub>	TKMODH	6069 <sub>H</sub>	RX0B
600A <sub>H</sub>	IAPERSS	602A <sub>H</sub>	MULL	604A <sub>H</sub>	TKMODU	606A <sub>H</sub>	RX0C
600B <sub>H</sub>	IAPPRGS	602B <sub>H</sub>	MULH	604B <sub>H</sub>	TKSDAL	606B <sub>H</sub>	TX0B
600C <sub>H</sub>	IAPS	602C <sub>H</sub>	DIVEL	604C <sub>H</sub>	TKSDAH	606C <sub>H</sub>	TX0C
600D <sub>H</sub>	T10	602D <sub>H</sub>	DIVEM	604D <sub>H</sub>	-	606D <sub>H</sub>	I2CX16
600E <sub>H</sub>	T10C	602E <sub>H</sub>	DIVEH	604E <sub>H</sub>	-	606E <sub>H</sub>	I2CC
600F <sub>H</sub>	T20L	602F <sub>H</sub>	DIVEU	604F <sub>H</sub>	TK0DAL	606F <sub>H</sub>	I2CSA
6010 <sub>H</sub>	T20H	6030 <sub>H</sub>	DIVSL	6050 <sub>H</sub>	TK0DAH	6070 <sub>H</sub>	I2CTB
6011 <sub>H</sub>	T20PL	6031 <sub>H</sub>	DIVSH	6051 <sub>H</sub>	TK1DAL	6071 <sub>H</sub>	I2CRB
6012 <sub>H</sub>	T20PH	6032 <sub>H</sub>	DIVQL	6052 <sub>H</sub>	TK1DAH	6072 <sub>H</sub>	I2CIEC
6013 <sub>H</sub>	-	6033 <sub>H</sub>	DIVQM	6053 <sub>H</sub>	TK2DAL	6073 <sub>H</sub>	I2CIFC
6014 <sub>H</sub>	-	6034 <sub>H</sub>	DIVQH	6054 <sub>H</sub>	TK2DAH	6074 <sub>H</sub>	ADRL
6015 <sub>H</sub>	-	6035 <sub>H</sub>	DIVQU	6055 <sub>H</sub>	TK3DAL	6075 <sub>H</sub>	ADRH
6016 <sub>H</sub>	-	6036 <sub>H</sub>	DIVRL	6056 <sub>H</sub>	TK3DAH	6076 <sub>H</sub>	ADCCL
6017 <sub>H</sub>	-	6037 <sub>H</sub>	DIVRH	6057 <sub>H</sub>	TK4DAL	6077 <sub>H</sub>	ADCCH
6018 <sub>H</sub>	T20CH	6038 <sub>H</sub>	DIVC	6058 <sub>H</sub>	TK4DAH	6078 <sub>H</sub>	WDTUL
6019 <sub>H</sub>	T20CM	6039 <sub>H</sub>	-	6059 <sub>H</sub>	TK5DAL	6079 <sub>H</sub>	WDTC
601A <sub>H</sub>	T21L	603A <sub>H</sub>	-	605A <sub>H</sub>	TK5DAH	607A <sub>H</sub>	OSCC1
601B <sub>H</sub>	T21H	603B <sub>H</sub>	-	605B <sub>H</sub>	TK6DAL	607B <sub>H</sub>	OSCC2
601C <sub>H</sub>	T21PL	603C <sub>H</sub>	-	605C <sub>H</sub>	TK6DAH	607C <sub>H</sub>	OSCWP
601D <sub>H</sub>	T21PH	603D <sub>H</sub>	-	605D <sub>H</sub>	TK7DAL	607D <sub>H</sub>	CLKG
601E <sub>H</sub>		603E <sub>H</sub>	-	605E <sub>H</sub>	TK7DAH	607E <sub>H</sub>	LVDCL
601F <sub>H</sub>		603F <sub>H</sub>	-	605F <sub>H</sub>	TK8DAL	607F <sub>H</sub>	LVDCH

图 3-8 特殊功能寄存器 (Section 0)

注: FAMR[1:0] = 00, 访问 6000<sub>H</sub> ~ 603F<sub>H</sub> 区间的 SFR;  
FAMR[1:0] = 01, 访问 6040<sub>H</sub> ~ 607F<sub>H</sub> 区间的 SFR。

FAMR[1:0] = 10		FAMR[1:0] = 11	
6080 <sub>H</sub>	-	60A0 <sub>H</sub>	-
6081 <sub>H</sub>	-	60A1 <sub>H</sub>	-
6082 <sub>H</sub>	-	60A2 <sub>H</sub>	-
6083 <sub>H</sub>	-	60A3 <sub>H</sub>	-
6084 <sub>H</sub>	-	60A4 <sub>H</sub>	-
6085 <sub>H</sub>	-	60A5 <sub>H</sub>	-
6086 <sub>H</sub>	-	60A6 <sub>H</sub>	-
6087 <sub>H</sub>	-	60A7 <sub>H</sub>	-
6088 <sub>H</sub>	-	60A8 <sub>H</sub>	-
6089 <sub>H</sub>	-	60A9 <sub>H</sub>	-
608A <sub>H</sub>	-	60AA <sub>H</sub>	-
608B <sub>H</sub>	-	60AB <sub>H</sub>	-
608C <sub>H</sub>	-	60AC <sub>H</sub>	-
608D <sub>H</sub>	-	60AD <sub>H</sub>	-
608E <sub>H</sub>	-	60AE <sub>H</sub>	-
608F <sub>H</sub>	-	60AF <sub>H</sub>	-
6090 <sub>H</sub>	-	60B0 <sub>H</sub>	-
6091 <sub>H</sub>	-	60B1 <sub>H</sub>	-
6092 <sub>H</sub>	-	60B2 <sub>H</sub>	-
6093 <sub>H</sub>	-	60B3 <sub>H</sub>	-
6094 <sub>H</sub>	-	60B4 <sub>H</sub>	-
6095 <sub>H</sub>	-	60B5 <sub>H</sub>	-
6096 <sub>H</sub>	-	60B6 <sub>H</sub>	-
6097 <sub>H</sub>	-	60B7 <sub>H</sub>	-
6098 <sub>H</sub>	-	60B8 <sub>H</sub>	-
6099 <sub>H</sub>	-	60B9 <sub>H</sub>	-
609A <sub>H</sub>	-	60BA <sub>H</sub>	-
609B <sub>H</sub>	-	60BB <sub>H</sub>	-
609C <sub>H</sub>	-	60BC <sub>H</sub>	-
609D <sub>H</sub>	-	60BD <sub>H</sub>	-
609E <sub>H</sub>	-	60BE <sub>H</sub>	-
609F <sub>H</sub>	-	60BF <sub>H</sub>	-
60C0 <sub>H</sub>	-	60E0 <sub>H</sub>	-
60C1 <sub>H</sub>	-	60E1 <sub>H</sub>	-
60C2 <sub>H</sub>	-	60E2 <sub>H</sub>	-
60C3 <sub>H</sub>	-	60E3 <sub>H</sub>	-
60C4 <sub>H</sub>	-	60E4 <sub>H</sub>	-
60C5 <sub>H</sub>	-	60E5 <sub>H</sub>	-
60C6 <sub>H</sub>	-	60E6 <sub>H</sub>	-
60C7 <sub>H</sub>	-	60E7 <sub>H</sub>	-
60C8 <sub>H</sub>	-	60E8 <sub>H</sub>	-
60C9 <sub>H</sub>	-	60E9 <sub>H</sub>	-
60CA <sub>H</sub>	-	60EA <sub>H</sub>	-
60CB <sub>H</sub>	-	60EB <sub>H</sub>	-
60CC <sub>H</sub>	-	60EC <sub>H</sub>	-
60CD <sub>H</sub>	-	60ED <sub>H</sub>	-
60CE <sub>H</sub>	-	60EE <sub>H</sub>	-
60CF <sub>H</sub>	-	60EF <sub>H</sub>	-
60D0 <sub>H</sub>	-	60F0 <sub>H</sub>	-
60D1 <sub>H</sub>	-	60F1 <sub>H</sub>	-
60D2 <sub>H</sub>	-	60F2 <sub>H</sub>	-
60D3 <sub>H</sub>	-	60F3 <sub>H</sub>	-
60D4 <sub>H</sub>	-	60F4 <sub>H</sub>	-
60D5 <sub>H</sub>	-	60F5 <sub>H</sub>	-
60D6 <sub>H</sub>	-	60F6 <sub>H</sub>	-
60D7 <sub>H</sub>	-	60F7 <sub>H</sub>	-
60D8 <sub>H</sub>	-	60F8 <sub>H</sub>	-
60D9 <sub>H</sub>	-	60F9 <sub>H</sub>	-
60DA <sub>H</sub>	-	60FA <sub>H</sub>	-
60DB <sub>H</sub>	-	60FB <sub>H</sub>	-
60DC <sub>H</sub>	-	60FC <sub>H</sub>	-
60DD <sub>H</sub>	-	60FD <sub>H</sub>	-
60DE <sub>H</sub>	-	60FE <sub>H</sub>	-
60DF <sub>H</sub>	-	60FF <sub>H</sub>	-

图 3-9 特殊功能寄存器 (Section 1)

注: FAMR[1: 0] = 10, 访问 6080<sub>H</sub> ~ 60BF<sub>H</sub> 区间的 SFR;  
 FAMR[1: 0] = 11, 访问 60C0<sub>H</sub> ~ 60FF<sub>H</sub> 区间的 SFR。



### 3.5.5 寻址方式

#### 3.5.5.1 直接寻址

当指令中的 8 位地址信息小于 80<sub>H</sub> 时，寻址 SRAM 地址空间。BKSR 用于选择 SRAM 存储体组，指令中的低 7 位地址信息用于在 BKSR 所选的存储体组中寻址。

当指令中的 8 位地址信息大于或等于 80<sub>H</sub> 时，寻址指向快速访问区。指令字中的低 7 位地址信息用于在快速访问区寻址。快速访问映射寄存器 FAMR 用于将 SFR 某个 SECTION 的高或低地址段映射到快速访问区的高地址段（FFC0<sub>H</sub>~FFFF<sub>H</sub>），FAMR 的 BIT0 用于选择映射高或低地址段。

示意图如下：

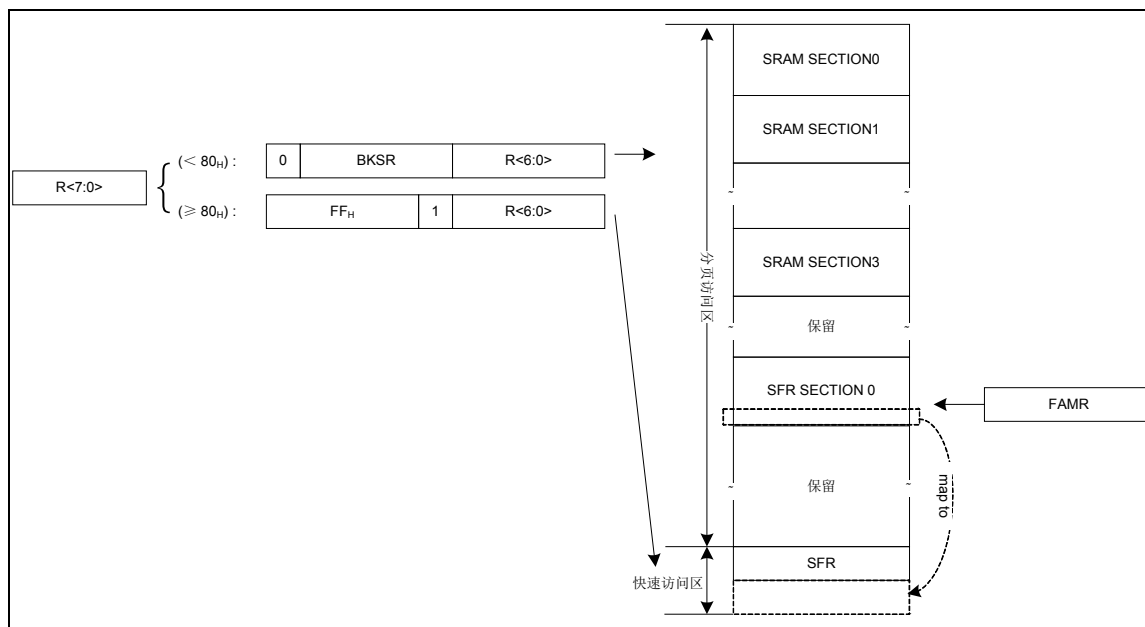


图 3-10 直接寻址示意图

#### 3.5.5.2 GPR特殊寻址

为方便较大的数据段（例如数组）在 GPR 中的移动，指令 MOVAR 和 MOVRA 用于对 GPR 进行特殊寻址读写操作，本芯片指令中支持 9 位地址信息（R<8:0>），可寻址 512 字节地址空间，无需进行 SECTION 间切换。MOVAR 和 MOVRA 指令无法访问 SFR。

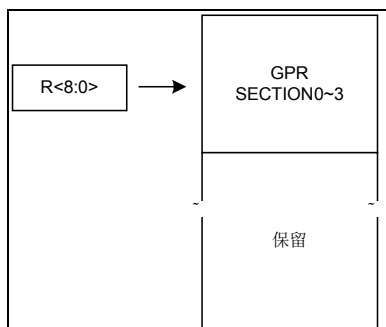


图 3-11 GPR 特殊寻址示意图

### 3.5.5.3 间接寻址

间接寻址是通过 16 位间接地址寄存器 IAA（由 2 个 8 位寄存器 IAAH 和 IAAL 组成）和 8 位虚拟数据寄存器 IAD 间接访问数据寻址空间中的存储单元。先将访问目的地址存放于间接地址寄存器 IAA, 再通过指令对 IAD 进行读/写操作, 实际的读/写操作对象则是 IAA 指向的数据寻址空间中的目的地址单元。

由于 IAD 寄存器本身也映射到数据寻址空间的 FF80H 地址。因此, 当 IAA 存放的地址值为 FF80H 时, 读/写 IAD 相当于用间接寻址方式访问虚拟寄存器 IAD 本身, 此时读操作将始终读取出 00H, 写操作则是一个空操作 (可能影响状态位)。

ISTEP 指令, 用来对 16 位间接地址寄存器 IAA 进行偏移操作。执行该指令时, 先将指令字中的 8 位有符号立即数进行符号位扩展为 16 位数, 再将 IAA 的值加上这个数的结果存回 IAA 寄存器。ISTEP 可实现的偏移范围为-128~127。

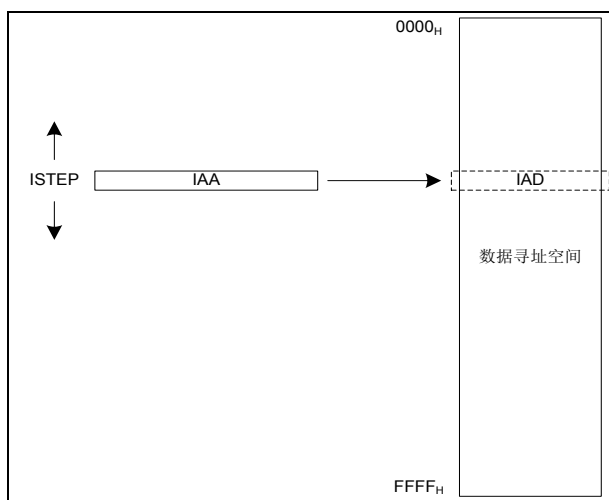


图 3-12 间接寻址示意图

## 3.6 特殊功能寄存器

### IAI: 间接寻址数据寄存器

Bit	7	6	5	4	3	2	1	0
Name	IAI<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 IAI<7:0>: 间接寻址数据

### IAAL: 间接寻址索引寄存器低 8 位

Bit	7	6	5	4	3	2	1	0
Name	IAAL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 IAAL<7:0>: 间接寻址索引低 8 位

IAAH: 间接寻址索引寄存器高 8 位								
Bit	7	6	5	4	3	2	1	0
Name	IAA<15:8>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 IAA<15:8>: 间接寻址索引高 8 位

BKSR: 存储体组选择寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	BKSR<1:0>	
R/W	—	—	—	—	—	—	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~2 保留

Bit 1~0 BKSR<1:0>: SRAM 存储体组选择

00: SRAM Section 0

01: SRAM Section 1

10: SRAM Section 2

11: SRAM Section 3

FAMR: 快速访问映射寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	FAMS	FMHL
R/W	—	—	—	—	—	—	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~2 保留

Bit 1 FAMS: 快速访问映射 SFR 存储体组选择

0: SFR Section 0

1: SFR Section 1

Bit 0 FMHL: 快速访问映射高/低地址段选择位

0: 选择低地址段映射

1: 选择高地址段映射

FRAL: 查表地址寄存器低 8 位								
Bit	7	6	5	4	3	2	1	0
Name	FRAL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 FRAL<7:0>: 查表地址低 8 位

FRAH: 查表地址寄存器高 8 位								
Bit	7	6	5	4	3	2	1	0
Name	FRAH<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 FRAH<7:0>: 查表地址高 8 位

FRALN: 查表地址寄存器低 8 位反								
Bit	7	6	5	4	3	2	1	0
Name	FRALN<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 FRALN<7:0>: 查表地址低 8 位的反

FRAHN: 查表地址寄存器高 8 位反								
Bit	7	6	5	4	3	2	1	0
Name	FRAHN<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 FRAHN<7:0>: 查表地址高 8 位的反

ROMDL: 查表数据寄存器低 8 位								
Bit	7	6	5	4	3	2	1	0
Name	ROMDL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 ROMDL<7:0>: 查表数据低 8 位

ROMDH: 查表数据寄存器高 8 位								
Bit	7	6	5	4	3	2	1	0
Name	ROMDH<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 ROMDH<7:0>: 查表数据高 8 位

ROMDLN: 查表数据寄存器低 8 位反								
Bit	7	6	5	4	3	2	1	0
Name	ROMDLN<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 ROMDLN<7:0>: 查表数据低 8 位的反

ROMDHN: 查表数据寄存器高 8 位反								
Bit	7	6	5	4	3	2	1	0
Name	ROMDHN<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0      ROMDHN<7:0>: 查表数据高 8 位的反

## 第 4 章 输入/输出端口

### 4.1 概述

本芯片最多支持 6 个 I/O 端口，共分为 PA, PB, PC 共 3 组。所有 I/O 端口都是 TTL/SMT 输入和 CMOS 输出驱动。端口驱动能力可配置。

每个端口都有相应的控制寄存器 PxT，来进行输入/输出控制。若 PxT 置 1，则 I/O 端口为输入状态；若 PxT 清 0，则 I/O 端口为输出状态。

IO 端口都有独立的内部弱上、下拉控制寄存器。若控制寄存器位置 1，则 I/O 端口弱上或者下拉使能；若控制寄存器位清 0，则 I/O 端口弱上或者下拉禁止。当端口设置为输出、模拟输入端口时，内部弱上、下拉自动禁止。

IO 端口都有独立的开漏输出控制寄存器；当端口设置为模拟输入端口时，开漏输出控制自动禁止。

端口的拉电流和灌电流驱动能力可以通过 PALC, PBLC, PCLC, SCC, SCSEL0 和 SCSEL1 寄存器来选择。

当 IO 复用设置为模拟端口时，相应的端口方向寄存器 PxT 应软件设置为输入端口。

### 4.2 I/O结构框图

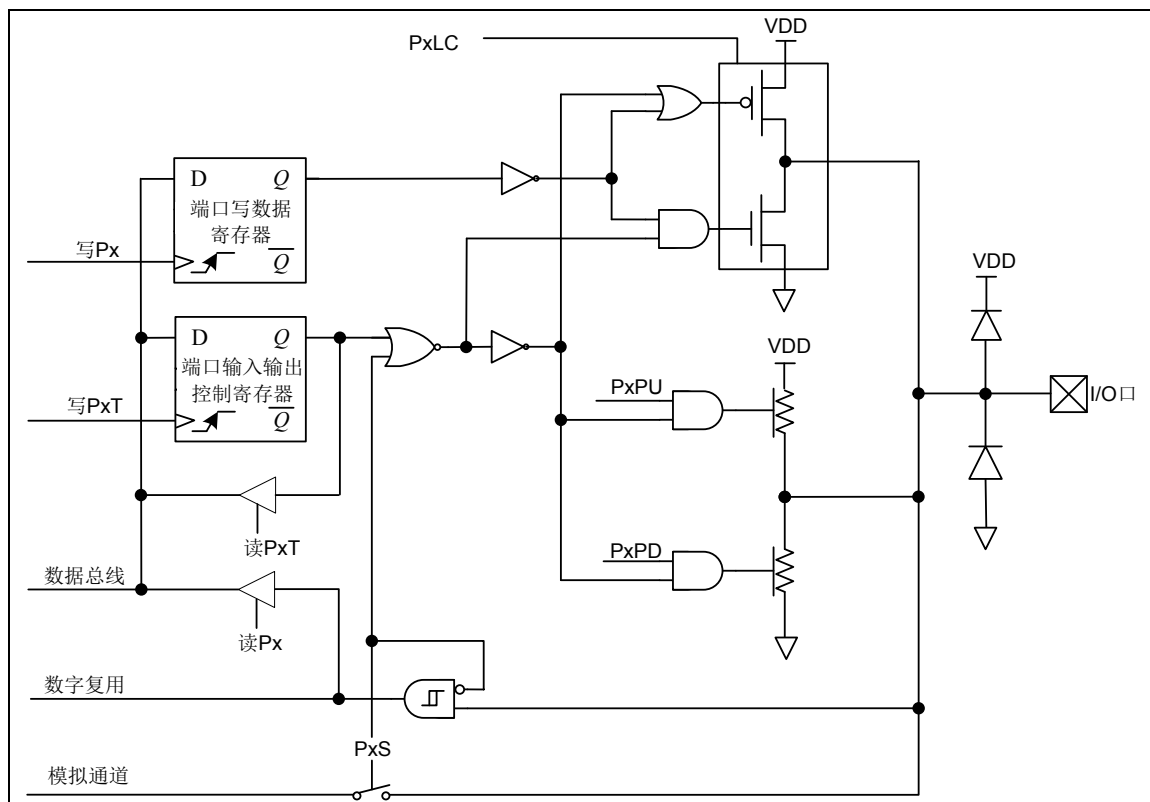


图 4-1 I/O 端口结构图

## 4.3 I/O端口功能

### 4.3.1 I/O端口输入/输出控制

芯片中的所有 I/O 端口都具有输入/输出的能力，端口控制寄存器 PAT/PBT/PCT 用于相应端口的输入或输出功能选择。当 I/O 端口设置为数字输出状态时，I/O 端口输出 PA/PB/PC 寄存器内容，即写相应 I/O 端口电平状态，读取 PA/PB/PC 寄存器的操作实际为读取相应 I/O 端口电平状态。当 I/O 端口设置为数字输入状态时，读取 PA/PB/PC 寄存器的操作实际为读取相应 I/O 端口电平状态。

### 4.3.2 I/O端口弱上拉、弱下拉功能

很多产品的应用中需要端口连接上拉或下拉电阻，使端口固定在一个稳定的电平状态，防止外界干扰以及其它影响。本芯片中所有端口支持独立的弱上、下拉功能。弱上、下拉功能仅在相应端口设为数字输入或数字复用输入模式下有效。

### 4.3.3 I/O端口模拟/数字类型选择功能

当数字信号和模拟信号共用管脚时，在使用对应端口的数字信号或模拟信号功能前，须正确设置端口的类型，否则可能不会达到预期的结果。本芯片部分复用端口具有独立的模拟/数字信号选择功能，由相应 PAS/PBS/PCS 寄存器控制。当端口被配置为模拟端口时，读取相应的 PA/PB/PC 寄存器始终读到“0”。

### 4.3.4 增强驱动能力I/O

端口驱动能力可通过 PALC, PBLC, PCLC, SCSEL0 和 SCSEL1 寄存器来配置。端口驱动能力参数请参考附录“电气特性”。

### 4.3.5 I/O端口复用功能

为了使资源合理利用最优化，本芯片大部分的 I/O 端口都具有复用功能。当端口用于复用功能时，管脚电平由复用功能决定。

## 4.4 I/O端口操作注意事项

当执行以端口寄存器为目标的指令（除位操作指令）时，芯片实际执行读-修改-写过程，即先读取该组全部 I/O 端口的电平，修改后再写回端口寄存器。位操作指令对 I/O 的修改操作只影响选定的位，对同组其它 I/O 不造成影响。因此建议用户对单个 I/O 的修改采用位操作指令。此外在复用功能使能和关闭时，应充分考虑当前 I/O 端口的输出寄存器值，并判断是否需要重新对这些 I/O 端口进行初始化赋值。

## 4.5 特殊功能寄存器

**PA: PA 端口电平状态寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PA<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	X	x	x	x	x

Bit 7~0 PA<7:0>: PA 端口电平状态

- 0: 低电平
- 1: 高电平

**PAT: PA 端口输入输出控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PAT<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	1	1	1	1	1	1

Bit 7~0 PAT<7:0>: PA 端口输入输出状态控制位

- 0: 输出状态
- 1: 输入状态

**PAS: PA 端口类型选择寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PAS<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	1	1	1	1	0	0

Bit 7~0 PAS<7:0>: PA 端口数字/模拟类型选择位

- 0: 数字端口
- 1: 模拟端口

注 1: PA0 和 PA1 分别复用为 ISP 的 ISDA 和 ISCK 端口, 默认为数字输入。

注 2: PA<1:0>之外的 PAx 端口均默认为模拟端口, 用户将这类端口用作数字端口时, 需先配置为数字端口。

**PAPU: PA 端口弱上拉控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PAPU<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PAPU<7:0>: PA 端口内部弱上拉控制位

- 0: 禁止
- 1: 使能



**PAPD: PA 端口弱下拉控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PAPD<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PAPD<7:0>: PA 端口内部弱下拉控制位

- 0: 禁止
- 1: 使能

**PALC: PA 端口灌电流驱动能力控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PALC<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PALC<7:0>: PA 端口灌电流驱动能力控制位

- 0: 弱驱动能力
- 1: 强驱动能力

**PAOD: PA 端口开漏控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PAOD<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PAOD<7:0>: PA 端口开漏控制位

- 0: 禁止
- 1: 使能

**PB: PB 端口电平状态寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PB<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	x	x	x	x	x	x	x	x

Bit 7~0 PB<7:0>: PB 端口电平状态

- 0: 低电平
- 1: 高电平

**PBT: PB 端口输入输出控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PBT<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	1	1	1	1	1	1

Bit 7~0 PBT<7:0>: PB 端口输入输出状态控制位

- 0: 输出状态
- 1: 输入状态

**PBS: PB 端口类型选择寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PBS<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	0	0	1	1	1	1

Bit 7~0      PBS<7:0>: PB 端口数字/模拟类型选择位

- 0: 数字端口
- 1: 模拟端口

注 1: PB4 和 PB5 分别复用为 ISP 的 ISDA 和 ISCK 端口，默认为数字输入。  
注 2: PB 除 PB<5:4>之外的端口默认为模拟端口，用户将这类端口用作数字端口时，需提前配置为数字端口。

**PBPU: PB 端口弱上拉控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PBPU<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0      PBPU<7:0>: PB 端口内部弱上拉控制位

- 0: 禁止
- 1: 使能

**PBPD: PB 端口弱下拉控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PBPD<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0      PBPD<7:0>: PB 端口内部弱下拉控制位

- 0: 禁止
- 1: 使能

**PBLC: PB 端口灌电流驱动能力控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PBLC<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0      PBLC<7:0>: PB 端口灌电流驱动能力控制位

- 0: 弱驱动能力
- 1: 强驱动能力

**PBOD: PB 端口开漏控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	PBOD<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PBOD<7:0>: PB 端口开漏控制位

0: 禁止

1: 使能

**PC: PC 端口电平状态寄存器**

Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	—	PC<0>
R/W	—	—	—	—	—	—	—	R/W
POR	0	0	0	0	0	0	x	x

Bit 7~1 保留

Bit 0 PC<0>: PC0 端口电平状态

0: 低电平

1: 高电平

**PCT: PC 端口输入输出控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	—	PCT<0>
R/W	—	—	—	—	—	—	—	R/W
POR	0	0	0	0	0	0	1	1

Bit 7~1 保留

Bit 0 PCT<0>: PC0 端口输入输出状态控制位

0: 输出状态

1: 输入状态

**PCS: PC 端口类型选择寄存器**

Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	—	PCS<0>
R/W	—	—	—	—	—	—	—	R/W
POR	0	0	0	0	0	0	0	1

Bit 7~1 保留

Bit 0 PCS<0>: PC0 端口数字/模拟类型选择位

0: 数字端口

1: 模拟端口

PCPU: PC 端口弱上拉控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	—	PCPU<0>
R/W	—	—	—	—	—	—	—	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~1 保留

Bit 0 PCPU<0>: PC0 端口内部弱上拉控制位

0: 禁止

1: 使能

PCPD: PC 端口弱下拉控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	—	PCPD<0>
R/W	—	—	—	—	—	—	—	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~1 保留

Bit 0 PCPD<0>: PC0 端口内部弱下拉控制位

0: 禁止

1: 使能

PCLC: PC 端口灌电流驱动能力控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	—	PCLC<0>
R/W	—	—	—	—	—	—	—	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~1 保留

Bit 0 PCLC<0>: PC0 端口灌电流驱动能力控制位

0: 弱驱动能力

1: 强驱动能力

PCOD: PC 端口开漏控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	—	—	—	COD<0>
R/W	—	—	—	—	—	—	—	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~1 保留

Bit 0 PCOD<0>: PC0 端口开漏控制位

0: 禁止

1: 使能

SCC: 端口源电流可调功能控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	SCC<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7            保留
- Bit 6            PA2~PA0 端口源电流可调使能位  
0: 禁止 (源电流大小参考直流电气特性)  
1: 使能
- Bit 5            PB5~PB4 端口源电流可调使能位  
0: 禁止 (源电流大小参考直流电气特性)  
1: 使能
- Bit 4~3         保留
- Bit 2            PC0 端口源电流可调使能位  
0: 禁止 (源电流大小参考直流电气特性)  
1: 使能
- Bit 1~0         保留

SCSEL0: 端口源电流选择寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—		SCSEL0<5:2>				—	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7~6         保留
- Bit 5~4         SCSEL0<5:4>: PA2~PA0 端口源电流选择位  
00: 4mA (典型值)  
01: 7mA (典型值)  
10: 11mA (典型值)  
11: 高阻
- Bit 3~2         SCSEL0<3:2>: PB5~PB4 端口源电流选择位  
00: 4mA (典型值)  
01: 7mA (典型值)  
10: 11mA (典型值)  
11: 高阻
- Bit 1~0         保留

SCSEL1: 端口源电流选择寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—		SCSEL1<5:4>		—			
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7~6         保留
- Bit 5~4         SCSEL1<5:4>: PC0 端口源电流选择位

00: 4mA (典型值)  
01: 7mA (典型值)  
10: 11mA (典型值)  
11: 高阻  
Bit 3~0 保留

## 第 5 章 特殊功能及操作特性

### 5.1 系统时钟和振荡器

#### 5.1.1 概述

芯片运行所需要的时钟源由振荡器提供，不同的振荡器选择可以让使用者在不同的应用需求中实现更大范围的功能。本款芯片所提供的振荡器有 2 种：内部高速 RC 振荡器（16MHz）和内部低速 RC 振荡器（32KHz）。灵活选择振荡器，使得产品在速度和功耗方面可以达到最优化。振荡器除了作为系统时钟源外，还可以作为看门狗定时器、ADC 电路等提供所需要的时钟源。

##### ◇ HRC

- 内部 16MHz RC 振荡器
- 出厂前，振荡器频率已经在常温下已经校准，校准精度在±1%以内

##### ◇ LRC

- 内部 32KHz RC 振荡器
- 出厂前，振荡器频率已经在常温下已经校准，校准精度在±2%以内

#### 5.1.2 结构框图

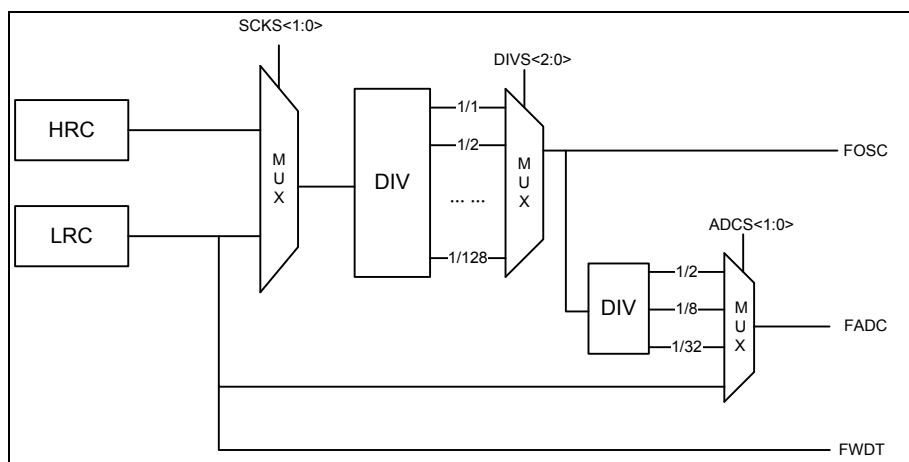


图 5-1 系统时钟内部结构图

#### 5.1.3 时钟源

##### 5.1.3.1 内部高速 16MHz RC 振荡器 HRC

芯片内置 16MHz RC 时钟振荡器，不需要外接其它外部器件。出厂前，内置 16MHz RC 时钟振荡器已在常温下校准。

##### 5.1.3.2 内部低速 32KHz RC 振荡器 LRC

芯片内置 32KHz RC 时钟振荡器（固定使能），不需要外接其它外部器件，可用作 WDT 计数时钟源，也可配置为主系统时钟源。出厂前，内置 32KHz RC 时钟振荡器已在常温下校准。

### 5.1.4 系统时钟源切换

高频时钟能提供系统更高的性能，低频时钟能够提供更低的功耗。因此，根据用户需要，灵活地安排高低频时钟切换，可以使系统在执行速度和功耗方面最优化。

芯片上电复位后，内部 16MHz RC 振荡器(HRC)默认作为系统时钟源。可通过配置 OSSCL 寄存器中的 SCKS 位切换系统时钟源。当芯片从睡眠模式唤醒时，默认系统时钟可由芯片配置字来确定使用 HRC 时钟或 LRC 时钟。

系统支持 2 种时钟之间相互切换，共 2 种情况：

- ◇ 内部高速 HRC 时钟切换到内部低速 LRC 时钟
- ◇ 检测 OSCCC2 寄存器中的 LRCON 位，检测到 LRCON=1；
- ◇ 设置 OSCCC1 寄存器中的 SCKS<0>=1；
- ◇ 检测 OSCCC1 寄存器的 CHG 位，直到检测到 CHG=0；
- ◇ 用户可选择关闭 HRC 时钟，即设置 OSCCC2 寄存器中的 HRCEN=0。
- ◇ 内部低速 LRC 时钟切换到内部高速 HRC 时钟
- ◇ 设置 OSCCC2 寄存器中的 HRCEN=1；
- ◇ 检测 OSCCC2 寄存器中的 HRCON 位，直到检测到 HRCON =1；
- ◇ 设置 OSCCC1 寄存器中的 SCKS<0>=0；
- ◇ 检测 OSCCC1 寄存器的 CHG 位，直到检测到 CHG=0；

#### 5.1.4.1 系统上电时序

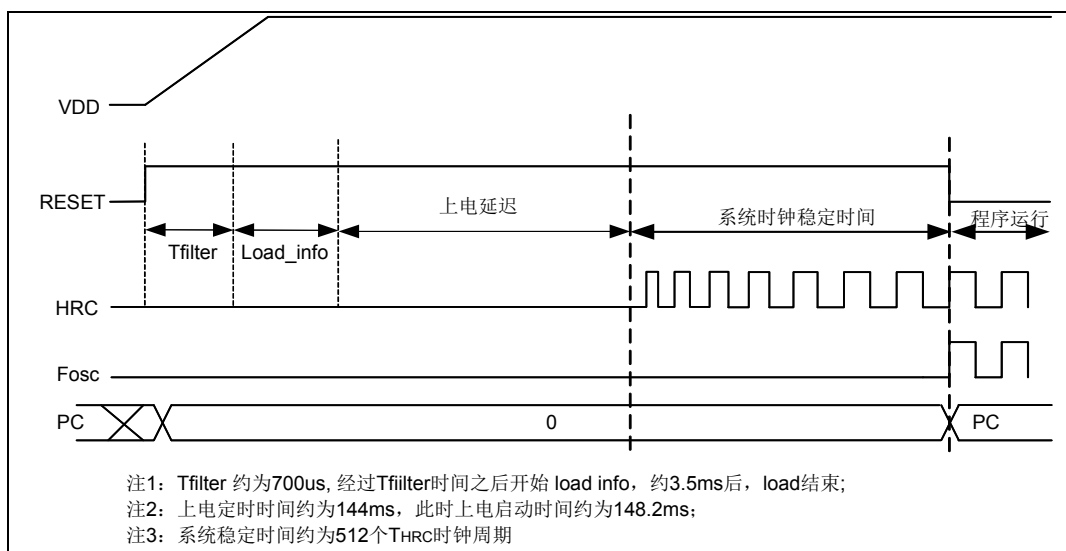


图 5-2 系统上电时序图

### 5.1.5 系统时钟分频

系统时钟支持 1 个最大分频比为 1:256 的后分频器，可通过 OSSCL 寄存器中的 DIVS<2:0> 位进行选择分频比。后分频器本身不可读写，配置系统时钟切换选择位 SCKS 后，后分频器计数自动清零，但不影响分频比设置。



改变系统时钟的分频比也同样视为系统时钟源的切换。

### 5.1.6 时钟切换等待

改变系统时钟的时钟源和改变系统时钟分频比都被视为系统时钟切换操作。为确保时钟切换时的系统稳定，在执行时钟切换操作时系统时钟会暂停运行，直到时钟切换结束再恢复运行。切换等待时间的长短视切换中的 2 个时钟源的频率而定。

### 5.1.7 特殊功能寄存器

OSCWP: 时钟控制写保护寄存器								
Bit	7	6	5	4	3	2	1	0
Name	OSCWP<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 为保证系统的稳定运行 OSCC2 和 OSCC1 寄存器默认处于写保护状态。解锁需对 OSCWP 寄存器写入 55H，写入其它值将恢复写保护状态。

OSCC1: 时钟控制寄存器 1								
Bit	7	6	5	4	3	2	1	0
Name	CHG	DIVS<2:0>			—	SST	—	SCKS
R/W	R	R/W	R/W	R/W	—	R	—	R/W
POR	0	0	0	0	0	0	0	0

Bit 7 CHG: 系统时钟切换标志位  
0: 切换完成  
1: 切换进行中

Bit 6~4 DIVS<2:0>: 系统时钟后分频比选择位  
000 = 1:1  
001 = 1:2  
010 = 1:4  
011 = 1:8  
100 = 1:16  
101 = 1:32  
110 = 1:64  
111 = 1:128

Bit 3 保留

Bit 2 SST: 系统时钟源标志位  
0: HRC 时钟源  
1: LRC 时钟源

Bit 1 保留

Bit 0 SCKS: 系统时钟源选择位  
0: HRC 时钟源  
1: LRC 时钟源

OSCC2: 时钟控制寄存器 2								
Bit	7	6	5	4	3	2	1	0
Name	CLKOEN	—	HRCON	LRCON	—	—	HRCEN	—
R/W	R/W	—	R	R	—	—	R/W	—
POR	0	0	0	0	0	0	0	0

Bit 7           CLKOEN: 系统时钟 128 分频输出使能位

0: 关闭

1: 使能, 输出到 CLK0 (PB0 端口)

Bit 6           保留

Bit 5           HRCON: HRC 时钟状态位

0: 关闭状态

1: 开启状态

Bit 4           LRCON: LRC 时钟状态位

0: 关闭状态

1: 开启状态

Bit 3~2        保留

Bit 1           HRCEN: HRC 时钟使能位

0: 关闭 (无其它硬件强制使能时)

1: 使能

Bit 0           保留

## 5.2 看门狗定时器

### 5.2.1 概述

看门狗定时器是芯片的一个组成部分，它的功能在于在发生软件故障时，通过器件复位将芯片复位。若系统进入了错误的工作状态，看门狗可以在合理的时间范围内使芯片复位。使能看门狗时，若用户程序清除看门狗定时器失败，则在预定的时间范围内，看门狗会使系统复位。

#### ◇ WDT 定时器

- 8 位 WDT 定时计数器（无实际物理地址，不可读写）
- 定时器时钟源为 LRC 时钟
- 8 位预分频器（无实际物理地址，不可读写）
- WDTUL 解锁寄存器
- WDT 控制寄存器（WDTC）
- 唤醒功能
- 复位功能

### 5.2.2 WDT操作

芯片提供 8 位 WDT 定时计数器，通过芯片配置字 WDTCEN 可使能硬件看门狗 WDT。当芯片配置字 WDTCEN 使能时，且 WDTC 寄存器的 SWDTCEN 控制位使能时，WDT 定时器计数使能。当 WDTCEN 关闭时，WDT 定时器计数禁止。用户通过编程界面选择。用户也可通过配置 WDTC 寄存器的 SWDTCEN 位来使能或禁止 WDT 计数，设置该位之前必须先对 WDT 进行解锁，即向 WDTUL 寄存器写入 0xA5。每一次对 WDTC 的写操作都必须重新解锁。

在 IDLE 模式下，WDT 计数溢出会唤醒 CPU；在正常运行模式下，WDT 计数溢出会复位芯片。为了避免不必要的复位，可使用 CWDT 指令适时清零 WDT 计数器。

使用 CWDT 指令将 WDT 计数器清零。WDT 支持一个预分频器，可通过 WDTC 寄存器中的 WDTPRS<2:0>位设置 WDT 时钟源的预分频比，再将分频后的时钟信号作为 WDT 定时器的计数时钟。WDT 时钟源来自内部 32KHz RC 振荡器 LRC 时钟。在预分频器分频比为 1:1 时，常温下（25℃）WDT 计数溢出时间约为 8ms。其它工作条件下，WDT 的计数溢出时间，可参考《附录 参数特性图》章节的相关图示。

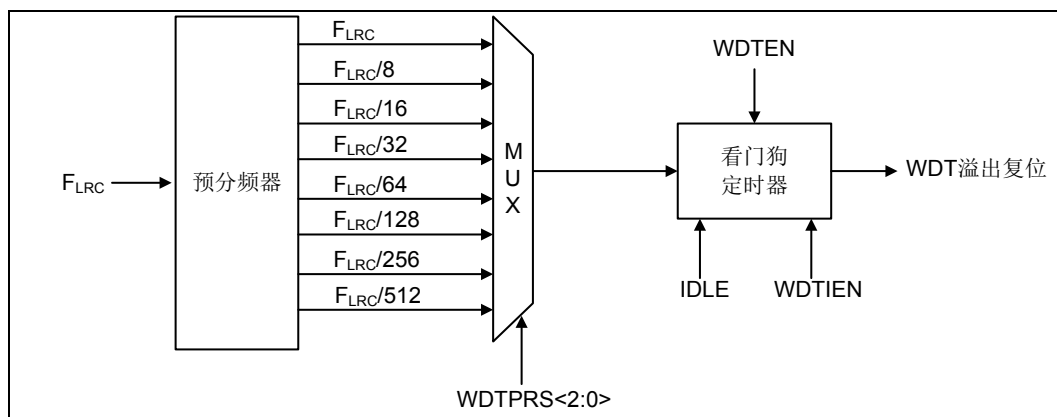


图 5-3 看门狗定时器内部结构图

### 5.2.3 特殊功能寄存器

WDT 的功能控制由 WDTC 寄存器和芯片配置字共同完成。WDT 功能的使能控制由芯片配置字的 WDTEN 位和 WDTC 控制寄存器的 SWDTEN 位控制。当 WDTEN 为 0 时，可以通过设置 SWDTEN 位来关闭 WDT；当 WDTEN 为 1 时，SWDTEN 位无效。WDT 在 IDLE 模式下的使能控制，以及 WDT 预分频器的分频比选择，由 WDTC 寄存器设置。

每次对 WDTC 寄存器进行写操作前都必须对 WDT 解锁，即向 WDTUL 寄存器写入 0xA5。

WDTUL: WDT 解锁控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	WDTUL<7:0>							
R/W	W	W	W	W	W	W	W	W
POR	0	0	0	0	0	0	0	0

Bit 7~0 WDTUL<7:0>: WDT 解锁寄存器  
写入 0xA5 解锁 WDTC 寄存器的写操作。WDTUL 寄存器读出值总为 0x00

WDTC: WDT 控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	SWDTEN	WDTIEN	WDTPRS<2:0>		
R/W	—	—	—	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	1	1	1	1	1

Bit 7~5 保留

Bit 4 SWDTEN: WDT 软件使能位 (仅当配置字 WDTEN=0 时有效)

0: 关闭

1: 使能 (缺省值)

Bit 3 WDTIEN: WDT 在 IDLE 模式下使能位 (仅当配置字 WDTEN=1 或 SWDTEN=1 时有效)

0: 关闭

1: 使能 (缺省值)

Bit 2~0 WDTPRS <2:0>: WDT 预分频器分频比选择位

000: 1:1

001: 1:8

010: 1:16  
011: 1:32  
100: 1:64  
101: 1:128  
110: 1:256  
111: 1:512(缺省值)

### 5.3 复位模块

#### 5.3.1 概述

- ◇ 复位功能是所有芯片中基本的部分，该芯片支持五种复位方式：
- ◇ 上电复位 POR
- ◇ 掉电复位 BOR，复位电压点可配置
- ◇ 看门狗定时器 WDT 计数溢出复位
- ◇ RST 指令复位

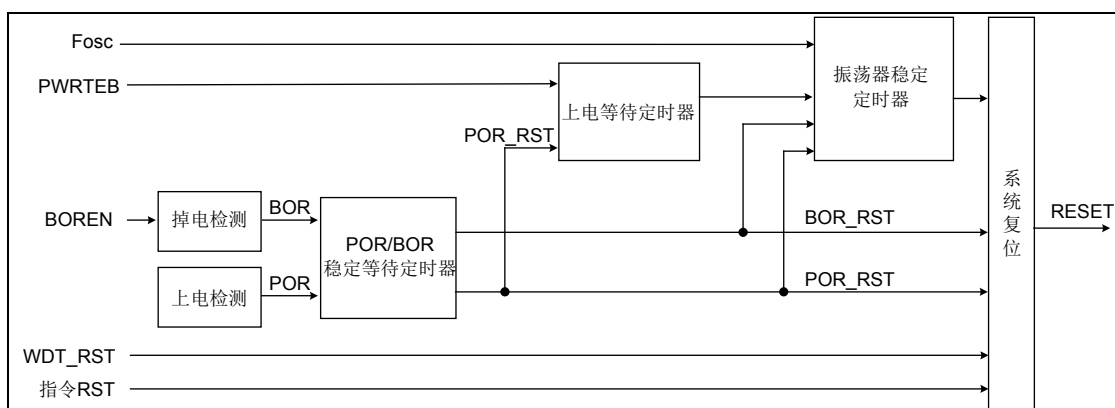


图 5-4 系统复位内部结构图

#### 5.3.2 上电复位POR

芯片上电过程中会产生 POR 复位，并且该复位信号将会一直保持到电源电压升高到芯片能够正常工作的电压为止。系统上电过程呈逐渐上升的曲线形式，需要一定时间才能达到正常电平值。上电复位的时序如下。

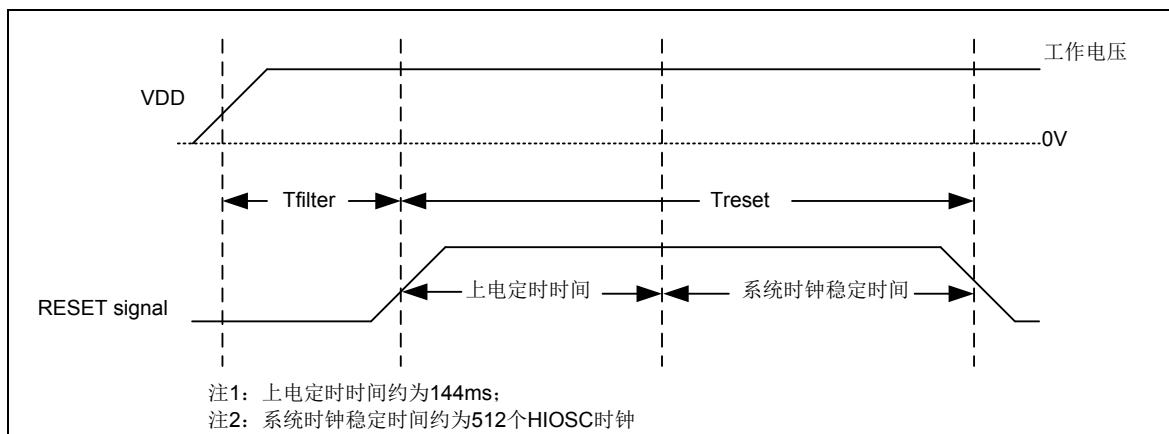


图 5-5 上电复位时序示意图

注：144ms 上电定时时间可以通过芯片配置字中的 PWRTEB 位屏蔽，但如果配置位 MRSTEN=0，MRSTN/PC1 管脚复用为 GPIO 端口，则该 144ms 上电定时时间无法被屏蔽。

### 5.3.3 掉电复位BOR

掉电复位针对外部因素引起的系统电压跌落情形（例如：更换电池），掉电复位可能会引起系统工作状态不正常或程序执行错误。

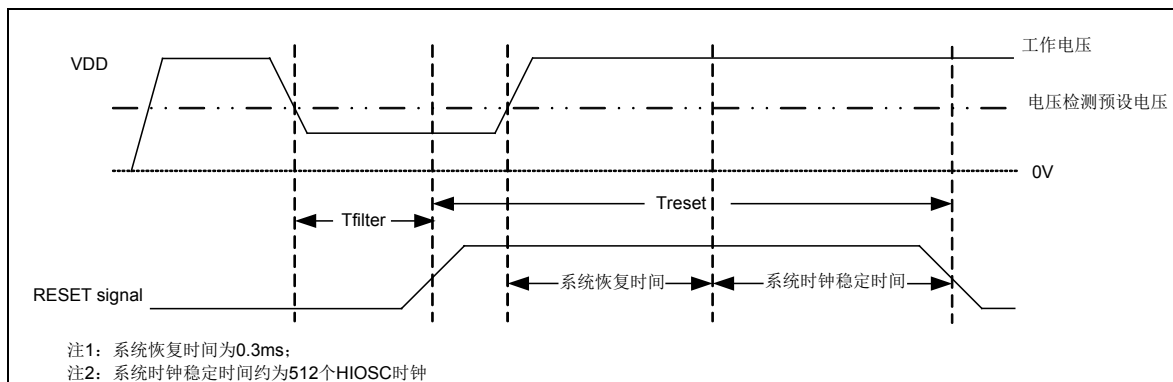


图 5-6 低电压复位时序示意图

注：144ms 上电定时时间可以通过芯片配置字中的 PWRTEB 位屏蔽，但如果配置位 MRSTEN=0，MRSTN/P4 管脚复用为 GPIO 端口，则该 144ms 上电定时时间无法被屏蔽。

### 5.3.4 看门狗定时器溢出复位

看门狗复位是系统的一种保护设置。在正常状态下，由程序将看门狗定时器清零。若出错，系统处于未知状态，程序无法清除看门狗，导致看门狗定时器溢出，此时系统复位。看门狗溢出复位后，系统重启进入正常状态。

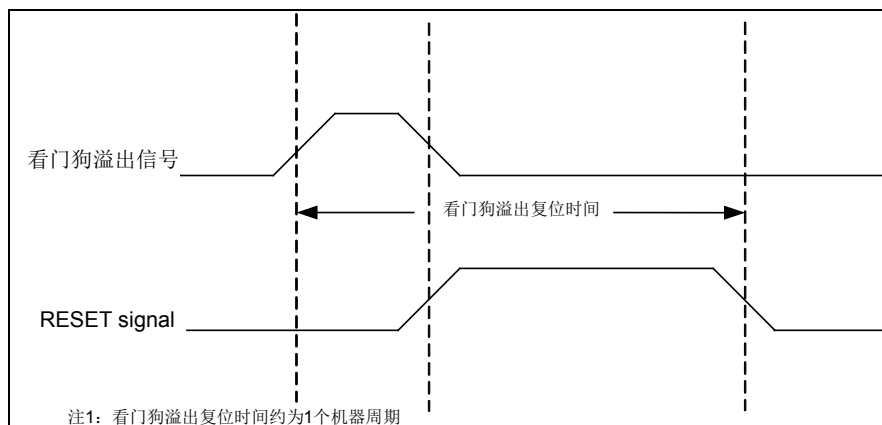


图 5-7 看门狗溢出复位

### 5.3.5 RST指令复位

整个芯片可通过执行 RST 指令复位，复位后，全部寄存器状态位都将被影响。

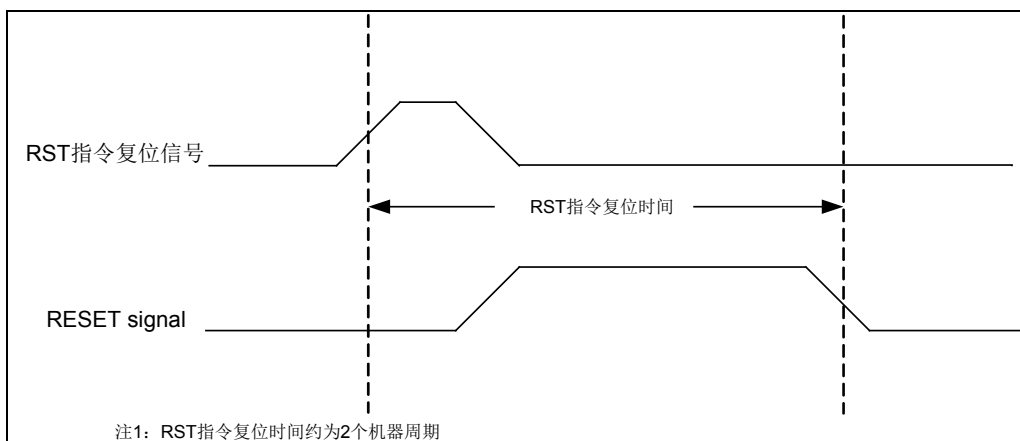


图 5-8 RST 指令复位

### 5.3.6 特殊功能寄存器

PWRCWP: 源控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	PWRCWP<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 PWRCWP<7:0>: PWRC 寄存器写保护寄存器

当 PWRCWP 写入 0xA5 时, PWRC 的 SMRSTEN 控制位的写保护解除。

PWRCWP 写入其它值无效, 保持写保护状态。当 SMRSTEN 控制位被写入后, 自动重新进入写保护状态。下一次写操作前必须重新解除写保护。

PWRC 寄存器的其它位不受 PWRCWP 写保护寄存器的影响。

PWRC: 电源控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	SMRSTEN	—	IRSTB	TOB	PDB	PORB	BORB
R/W	—	R/W	—	R/W	R/W	R/W	R/W	R/W
POR	1	1	0	1	1	1	0	x

Bit 7 保留

Bit 6 SMRSTEN: MRSTN/PC1 管脚复用软件配置位 (仅当配置字 MRSTEN =1 时有效)

0: 管脚配置为 GPIO PC1 功能

1: 管脚配置为 MRSTN 功能 (缺省值)

Bit 5 保留

Bit 4 IRSTB: 指令复位标志位

0: 执行复位指令 (必须用软件置位)

1: 未执行复位指令

Bit 3 TOB: WDT 溢出标志位

0: WDT 计数溢出时被清零

1: 上电复位或执行 CWDT、IDLE 指令后被置 1

Bit 2 PDB: 低功耗标志位



- 0: 执行 IDLE 指令后清零
  - 1: 上电复位或执行 CWDT 指令后置 1
- Bit 1      PORB: 上电复位状态位
- 0: 上电复位发生（上电复位后，必须软件置位）
  - 1: 无上电复位发生
- Bit 0      BORB: 低电压复位状态位
- 0: 低电压复位发生（低电压复位后，必须软件置位）
  - 1: 无低电压复位发生

注：如果用户需要将 MRSTN/PC1 管脚复用为通用 I/O 端口，但又希望屏蔽 144ms 上电定时时间，则可以设置配置位 MRSTEN=1，配置位 PWRTEB=1，通过软件设置寄存器位 SMRSTEN=0，使 MRSTN/PC1 管脚在芯片上电复位期间复用为外部复位端口 MRSTN 且屏蔽了 144ms 上电定时时间，当复位结束后由软件设置复用为 PC1 端口。

## 5.4 低功耗操作

### 5.4.1 概述

用户可通过 IDLE 指令使 CPU 暂停执行，进入 IDLE 状态以降低芯片功耗。用户还可以在执行 IDLE 指令前，关闭部分或全部芯片模块，以进入更深程度的睡眠状态，最大限度的降低芯片功耗。芯片支持多种 IDLE 唤醒源，用于 IDLE 模式下的芯片唤醒。

### 5.4.2 IDLE 状态

IDLE 指令执行后，系统时钟 FOSC 暂停，CPU 停止运行，PC 保持当前值，采用系统时钟运行的同步功能模块均保持当前状态暂停执行，其它异步功能模块可根据 IDLE 前的设置继续运行或关闭。所有 I/O 端口将保持进入 IDLE 前的状态，若使能 WDT，则 WDT 将被清零并保持运行。PDB 标志位被清零，TOB 标志位被置 1。IDLE 状态下保持异步运行的外设可产生中断，并置相应的中断标志。

### 5.4.3 唤醒方式配置

序号	唤醒方式	唤醒使能	中断模式	备注
1	WDT	WDTIEN	—	—
2	LVD	LVDIE	默认/向量	—
3	ADC	ADIE	默认/向量	需选择 LRC 作为转换时钟

### 5.4.4 低功耗下的功能模块

执行 IDLE 指令前，通过关闭各功能模块使能位，可使芯片在执行 IDLE 指令后进入更深程度的低功耗状态。同时应避免数字输入的 I/O 管脚处于浮空状态，需将这些管脚接固定电平，或在芯片外部进行上拉或下拉处理，否则会引起 I/O 端口漏电。

功能	类型	使能/关闭	唤醒使能	备注
WDT	异步	WDTEN	WDTIEN	—
LVD	异步	LV DEN	LVDIE	—
ADC	异步	ADEN	ADIE	需选择 LRC 作为转换时钟

表 5-1 功能模块低功耗配置分类表

### 5.4.1 时钟源的关闭和唤醒

芯片进入 IDLE 状态后，HRC 时钟源自动关闭，LRC 时钟源始终保持运行。

当 IDLE 被唤醒时，时钟源首先被唤醒，每个时钟源从唤醒到进入正常工作状态都有 WARMUP 时间，WARMUP 时间根据时钟源的时钟频率不同，时间长短也有不同。

在运行或 IDLE 状态下各个外设模块的时钟源也可以设置关停或打开，由寄存器 CLKG 来控制。

ADC 模块在 IDLE 状态下可运行。

### 5.4.2 特殊功能寄存器

CLKG: 模块时钟关停寄存器								
Bit	7	6	5	4	3	2	1	0
Name	ADCCE	T21CE	T20CE	—	TKCE	IICCE	UART0CE	T10CE
R/W	R/W	R/W	R/W	—	R/W	R/W	R/W	R/W
POR	1	1	1	0	1	1	1	1

- Bit7      ADCCE: ADC 时钟使能位  
           0: ADC 关停  
           1: ADC 工作
- Bit6      T21CE: T21 模块时钟使能位  
           0: T21 关停  
           1: T21 工作
- Bit5      T20CE: T20 模块时钟使能位  
           0: T20 关停  
           1: T20 工作
- Bit4      保留
- Bit3      TKCE: TK 模块时钟使能位  
           0: TK 关停  
           1: TK 工作
- Bit2      IICCE: IIC 模块时钟使能位  
           0: IIC 关停  
           1: IIC 工作
- Bit1      UART0CE: UART0 模块时钟使能位  
           0: UART0 关停  
           1: UART0 工作
- Bit0      T10CE: T10 模块时钟使能位  
           0: T10 关停  
           1: T10 工作

## 第 6 章 外设

### 6.1 定时器/计数器 (Timer/Counter) 模块

定时/计数器模块包括：

- ◇ 1 路 8 位定时器/计数器 T10；
- ◇ 2 路 16 位多功能定时器 T20/T21；

#### 6.1.1 8 位定时器/计数器 (T10)

##### 6.1.1.1 概述

8 位定时器/计数器包括定时器和计数器两种工作模式。定时器模式根据寄存器制定的定时时间进行定时，可以使定时器有选择地产生中断请求或完成其它操作。计数器模式用于对外部时钟信号 (T10CKI) 进行计数。

- ◇ T10 支持两种工作模式
  - 定时器模式 (时钟源为系统时钟二分频 (Fosc/2))
  - 计数器模式 (时钟源为外部输入时钟 T10CKI)
- ◇ T10 支持以下功能组件
  - 8 位预分频器 (无实际物理地址, 不可读写)
  - 8 位计数器 (T10)
  - 8 位控制寄存器 (T10C)
- ◇ 中断和暂停
  - 支持溢出中断标志 (T10IF)
  - 支持中断处理
  - 在 IDLE 模式下, T10 暂停工作

##### 6.1.1.2 内部结构图

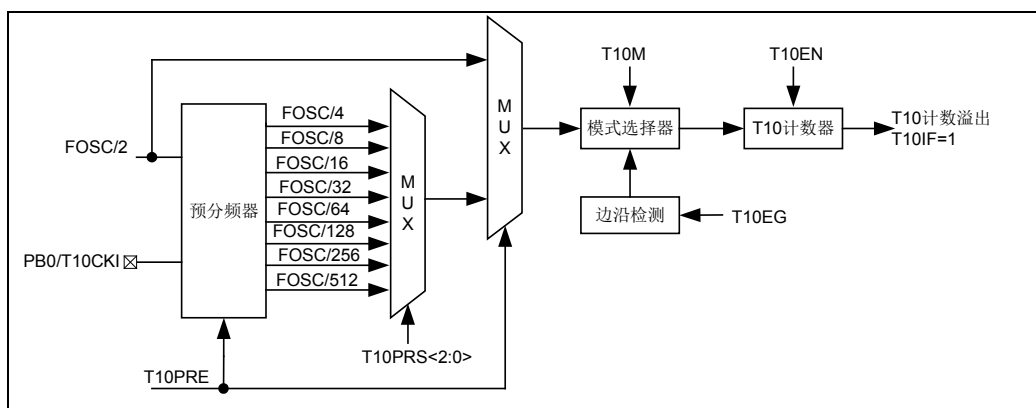


图 6-1 T10 内部结构图

##### 6.1.1.3 预分频器

预分频器可以提供定时器/计数器一个更长的溢出周期。当 T10C 寄存器中的 T10PRE 为

“1”时，使能 T10 预分频器。任何对 T10 计数器的写操作都会清零预分频器，但不影响预分频器的分频比设置，预分频器的计数值无法读写。预分频器的分频比可通过 T10C 寄存器中的 T10PRS <2:0>位进行设置，预分频比范围为 1: 2~1: 256。

工作模式	T10PRE	T10PRS<2:0>	T10 计数时钟
定时器模式	0	-	Fosc/2
	1	000	Fosc/4
	1	001	Fosc/8
	1	010	Fosc/16
	1	011	Fosc/32
	1	100	Fosc/64
	1	101	Fosc/128
	1	110	Fosc/256
	1	111	Fosc/512
计数器模式	0	-	T10CKI
	1	000	T10CKI/2
	1	001	T10CKI/4
	1	010	T10CKI/8
	1	011	T10CKI/16
	1	100	T10CKI/32
	1	101	T10CKI/64
	1	110	T10CKI/128
	1	111	T10CKI/256

表 6-1 T10 预分频器配置表

#### 6.1.1.4 工作模式

T10 有定时器和计数器两种工作模式，定时器模式和计数器模式，通过 T10M 进行选择。定时器和计数器计数模式均支持预分频器。配置为定时器模式时，T10 计数器的时钟源为系统时钟 2 分频 (Fosc/2)；配置为计数器模式时，T10 计数器的时钟源为经系统机器周期同步的外部输入时钟 T10CKI。因此，T10CKI 输入的高电平和低电平时间都至少为一个机器周期。通过 T10C 寄存器中的 T10EG 位选择外部时钟的计数边沿为上升沿或下降沿。T10CKI 所在 IO 端口必须配置为数字输入状态。

#### 6.1.1.5 定时器模式

T10 计数器为递增计数，计数值由 FFH 变为 00H 时，T10 计数器发生溢出并重新开始计数。T10 计数器发生溢出，中断标志 T10IF 位被置“1”。如果中断使能位 T10IE 和全局中断使能位 GIE 都为“1”，则向 CPU 发出 T10 溢出中断响应请求。中断标志 T10IF 位需通过软件清零。在 CPU 进入休眠模式后，T10 模块不工作，因此不产生中断。

当 T10 配置为定时器模式时，若禁止预分频器，T10 计数器的时钟为系统时钟二分频 (Fosc/2)；若使能预分频器，分频器对 Fosc/2 进行分频，此时，T10 计数器的计数时钟为分频后的时钟。

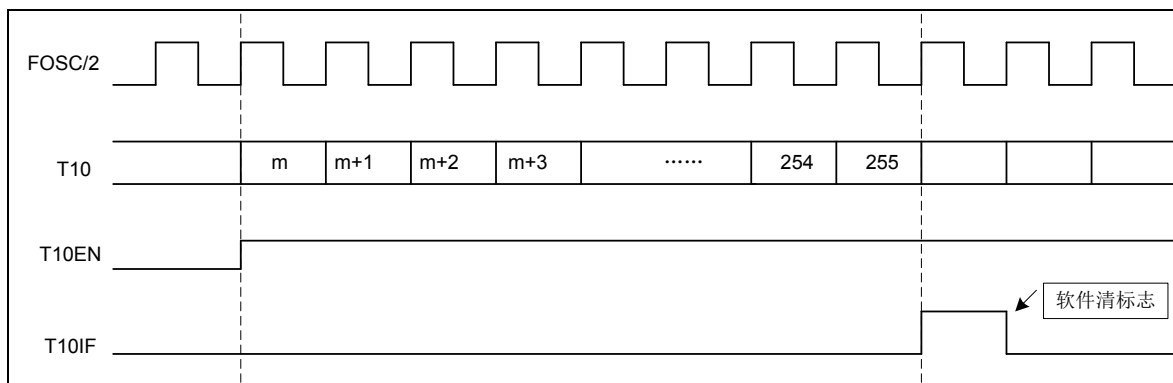


图 6-2 定时器模式时序图

### 6.1.1.6 计数器模式

当 T10 配置为计数器模式时，若禁止预分频器，则 T10 计数器的时钟为外部输入时钟 T10CKI，并由二分频后的系统时钟 Fosc/2 进行同步，所以 T10CKI 保持高电平或者低电平的时间至少为一个机器周期。通过设置 T10EG (T10C<4>)选择外部时钟的计数边沿为上升沿或下降沿。

同样，计数器模式也支持预分频器对外部时钟 T10CKI 进行分频。并且，T10CKI 复用的 IO 端口必须配置为数字输入状态。

当 T10 计数器递增计数由 FFH 变为 00H 时，T10 计数器发生溢出，中断标志 T10IF 位被置“1”。如果中断使能位 T10IE 和全局中断使能位 GIE 都为“1”，则向 CPU 发出 T10 溢出中断响应请求。中断标志 T10IF 位需通过软件清零。在 CPU 进入休眠模式后，T10 模块不工作，因此不产生中断。

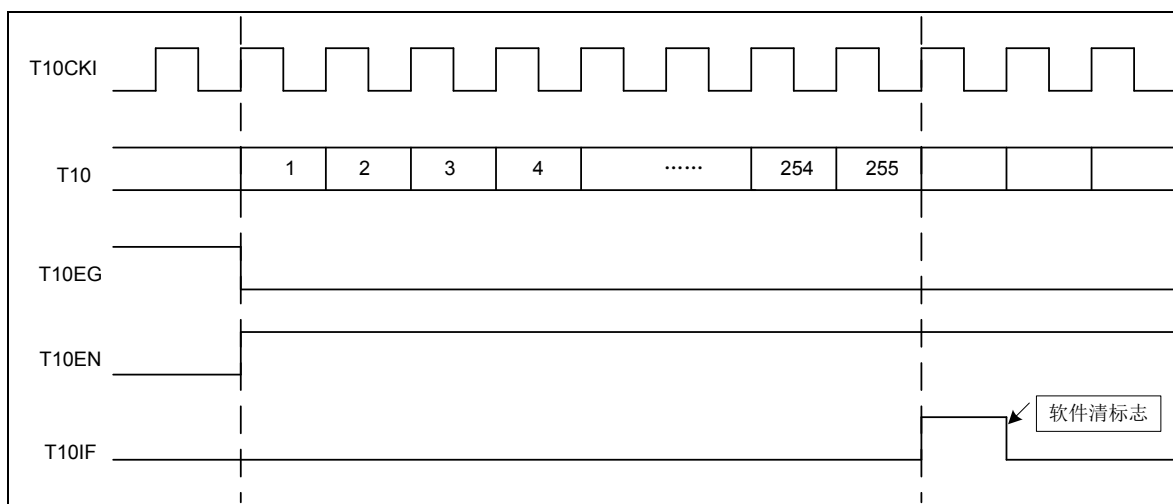


图 6-3 计数器模式时序图 (T10EG=0, T10CKI 上升沿计数)

### 6.1.1.7 特殊功能寄存器

8 位定时器/计数器 T10 由两个寄存器控制，一个 8 位计数器 T10 和一个控制寄存器 T10C。T10 寄存器用于存放计数值，T10C 控制寄存器用于控制 T10 的使能、T10 的模式选择、T10CKI 计数边沿选择、预分频器使能位以及预分频器分频比选择。

T10: T10 计数器								
Bit	7	6	5	4	3	2	1	0
Name	T10<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0      T10<7:0>: 8 位 T10 计数值

T10C: T10 控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	T10EN	—	T10M	T10EG	T10PRE	T10PRS<2:0>		
R/W	R/W	—	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7      T10EN: T10 使能位

0: 关闭

1: 使能

Bit 6      保留

Bit 5      T10M: T10 模式选择位

0: 定时器模式 (时钟源为系统时钟二分频  $F_{osc}/2$ )

1: 计数器模式 (时钟源为 T10CKI)

Bit 4      T10EG: T10CKI 计数边沿选择位

0: T10CKI 上升沿计数

1: T10CKI 下降沿计数

Bit 3      T10PRE: 预分频器使能位

0: 禁止

1: 使能

Bit 2~0    T10PRS<2:0>: 预分频器分频比选择位

000: 1:2

001: 1:4

010: 1:8

011: 1:16

100: 1:32

101: 1:64

110: 1:128

111: 1:256

## 6.1.2 16 位多功能定时器 (T20/T21)

### 6.1.2.1 概述

16 位多功能定时器 T2n 支持 1 种工作模式:定时器模式。

- ◇ T2n 支持 1 种工作模式
  - 定时器模式 (时钟源为 Fosc)
- ◇ T2n 支持以下功能组件
  - 4 位预分频器 (无实际物理地址, 不可读写)
  - 16 位计数器 T2n (计数器初始值可写)
- ◇ 中断和暂停
  - 支持溢出中断 T2nVIF
  - 在 IDLE 模式下, T2n 停止工作

注: 本节中 T2n 代表 T20 或 T21。

### 6.1.2.2 预分频器

预分频器可以提供更长的溢出周期。T2n 支持可配置的预分频器。通过 T2nCL 寄存器中的 T2nPRS 位配置预分频器的分频比, 预分频比的范围为 1:1 ~ 1:16。任何对计数器或控制寄存器的写操作都会使预分频器清零, 但不改变配置的分频比。预分频器的计数值不可读写。

### 6.1.2.3 工作模式

T2nEN 置 1 使能之前, 需先配置好预分频器、后分频器、周期等参数。保证计数时钟在使能时已稳定。

### 6.1.2.4 定时器模式

当 T2nM 为 0000 时, Tn 工作在定时器模式。

T2n 计数器的时钟源为系统时钟 Fosc, 并支持预分频器和后分频器。

T2n 计数器为可读写寄存器, 支持计数初值的设置和计数过程中的计数值改变。

T2nEN 使能时, 16 位定时器 T2n 对计数时钟进行递增计数, 当 T2n 的计数值与周期寄存器 T2nP 相等时, 后分频计数器加 1, 同时 T2n 计数器被自动清零并重新开始计数。

当后分频器的计数值与后分频器分频比相同时, 复位后分频器, 并将定时溢出中断标志 T2nVIF 置“1”, 该中断标志需要软件清零。



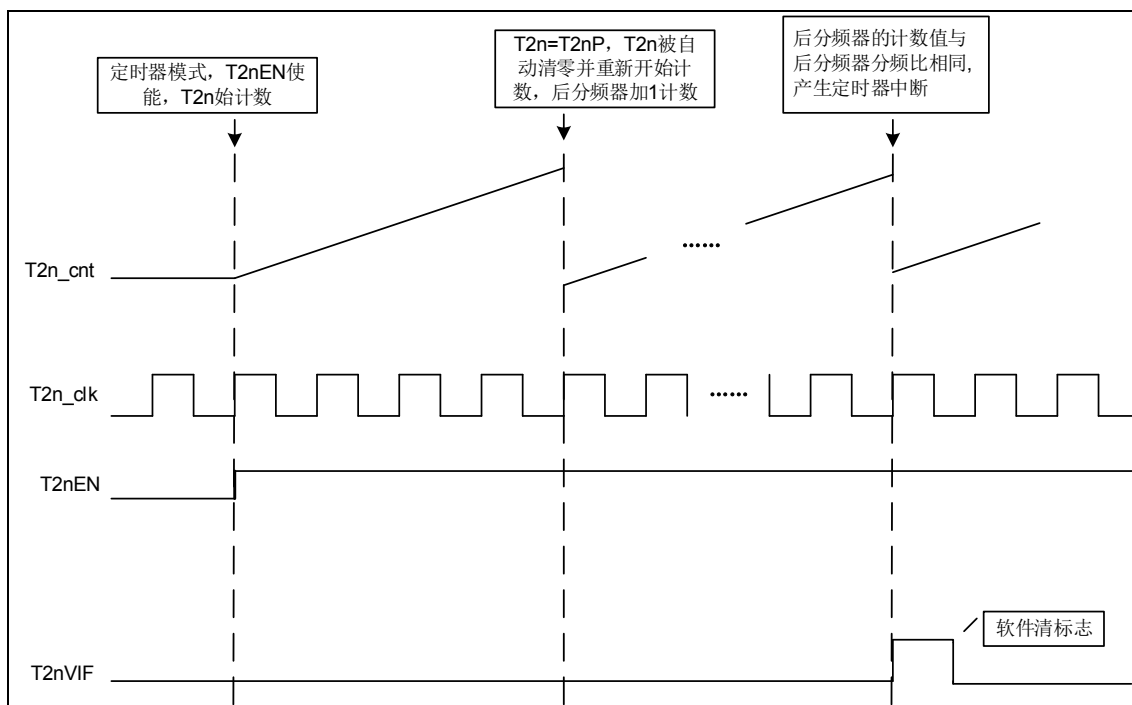


图 6-4 T2n 定时器模式时序图

### 6.1.2.5 特殊功能寄存器

T2nL: T2n 计数器低 8 位								
Bit	7	6	5	4	3	2	1	0
Name	T2nL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 T2nL<7:0>: T2n 计数器低 8 位

T2nH: T2n 计数器高 8 位								
Bit	7	6	5	4	3	2	1	0
Name	T2nH<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 T2nH<7:0>: T2n 计数器高 8 位

T2nPL: T2n 周期寄存器低 8 位								
Bit	7	6	5	4	3	2	1	0
Name	T2nPL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	1	1	1	1	1	1

Bit 7~0 T2nPL<7:0>: 周期值低 8 位

**T2nPH: T2n 周期寄存器高 8 位**

Bit	7	6	5	4	3	2	1	0
Name	T2nPH<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	1	1	1	1	1	1	1	1

Bit 7~0 T2nPH<7:0>: 周期值高 8 位

**T2nCM: T2n 控制寄存器中 8 位**

Bit	7	6	5	4	3	2	1	0
Name	—	—	—		T2nPRS<3:0>			
R/W	—	—	—	—	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~4 保留

Bit 3~0 T2nPRS<3:0>: T2n 预分频器分频比选择位

- 0000: 分频比为 1:1
- 0001: 分频比为 1:2
- 0010: 分频比为 1:3
- 0011: 分频比为 1:4
- 0100: 分频比为 1:5
- 0101: 分频比为 1:6
- 0110: 分频比为 1:7
- 0111: 分频比为 1:8
- 1000: 分频比为 1:9
- 1001: 分频比为 1:10
- 1010: 分频比为 1:11
- 1011: 分频比为 1:12
- 1100: 分频比为 1:13
- 1101: 分频比为 1:14
- 1110: 分频比为 1:15
- 1111: 分频比为 1:16

**T2nCH: T2n 控制寄存器高 8 位**

Bit	7	6	5	4	3	2	1	0
Name	T2nEN	T2nPOS<6:0>						
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7 T2nEN : T2n 使能位

- 0: 关闭
- 1: 使能

Bit 6~0 T2nPOS<6:0>: T2n 后分频器分频值

后分频次数 = T2nPOS<6:0> + 1

## 6.2 异步接收发送器 (UART0)

### 6.2.1 概述

本芯片支持 1 组全双工的通用异步接收器发送器 UART0，是与外部设备进行通讯的串行接口，可以很方便的与其它具有串行接口的外部设备通讯。

- ◇ 支持两种工作模式
  - 异步接收器
  - 异步发送器
- ◇ 传输波特率配置
  - 高速模式
  - 低速模式
  - 支持 8/9 位传输数据格式，约定数据从最低位开始接收/发送
- ◇ 支持全双工模式
- ◇ UARTn 支持以下功能组件
  - 接收数据寄存器 RXnB
  - 接收控制寄存器 RXnC
  - 发送数据寄存器 TXnB
  - 发送控制寄存器 TXnC
  - 发送移位寄存器 TXnR (无实际物理地址，不可读写)
  - 波特率寄存器 BRnR
- ◇ 中断和暂停
  - 支持接收中断标志 (RXnIF, 只读)
  - 支持发送中断标志 (TXnIF, 只读)
  - 支持中断处理
  - 在 IDLE 模式下，支持接收/发送中断唤醒
- ◇ 兼容 RS-232/RS-442/RS-485 的通讯接口

### 6.2.2 内部结构图

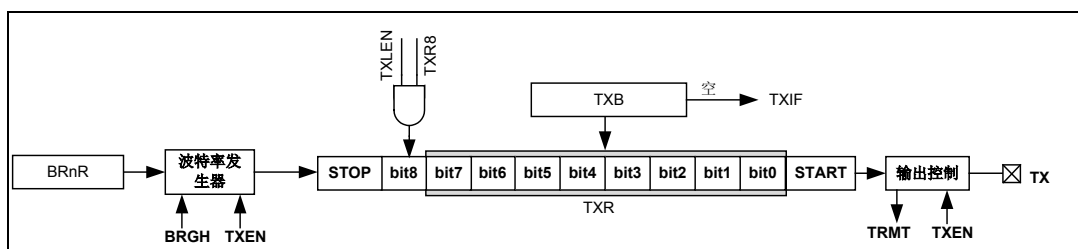


图 6-5 UART 发送端原理图

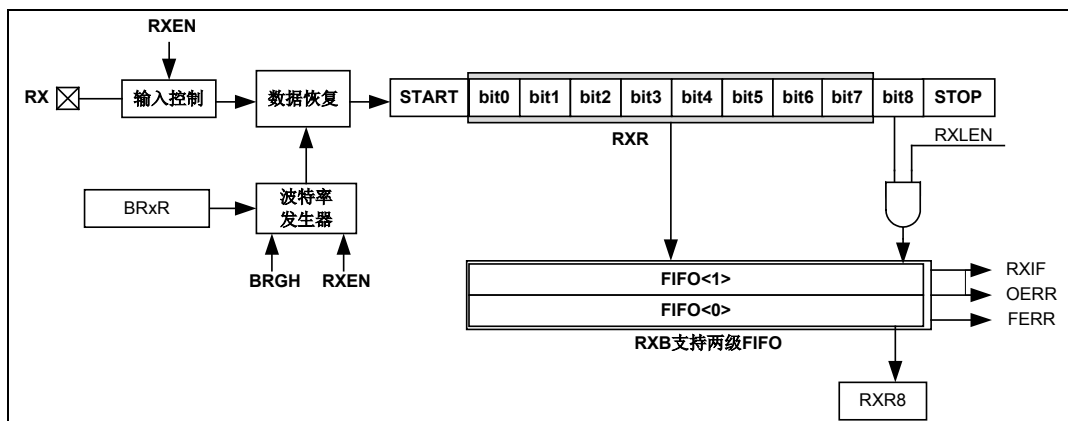


图 6-6 UART 接收端原理图

### 6.2.3 波特率配置

UARTn 自身具有一个波特率发生器，通过它可以设定数据传输速率。波特率是由一个独立的内部 8 位计数器产生，它由 BRnR 寄存器和 TXnC 寄存器的 BRGHn 来控制。BRGHn 是决定波特率发生器处于高速模式还是低速模式，从而决定计算公式的选用。

波特率	计算公式	BRGHn
低速模式	$F_{osc}/(64 \times (BRnR\langle 7:0 \rangle + 1))$	0
高速模式	$F_{osc}/(16 \times (BRnR\langle 7:0 \rangle + 1))$	1

表 6-2 UARTn 波特率配置表

### 6.2.4 传输数据格式

UARTn 的传输数据格式有两种选择，8 位或 9 位。接收 9 位数据时，读取 RXnC 寄存器中的 RXnR8 位可以得到接收的第 9 位数据。发送 9 位数据时，可通过 TXnC 寄存器中的 TXnR8 位设置将要发送的第 9 位数据。



图 6-7 UARTn 数据格式示意图

### 6.2.5 异步发送器

异步发送器发送数据时，起始位 (START) 和结束位 (STOP) 由芯片内部产生，用户只需要使能异步发送器，并将所要发送的数据写入 TXnB 和 TXnR8 内，就能实现异步发送，异步发送器还可以实现数据连续发送。注意，在发送数据时，用户使能异步发送器后，需先查询标志位 TRMTn，检查发送移位寄存器 (TXnR) 是否为空，只有在发送移位寄存器为空时，才能发送数据。由于 UART 发送器发送端口 TXn 与 I/O 端口复用，在使用 UART 发送端口前，需首先设置所复用的 I/O 端口处于输出状态。

操作流程图如下：

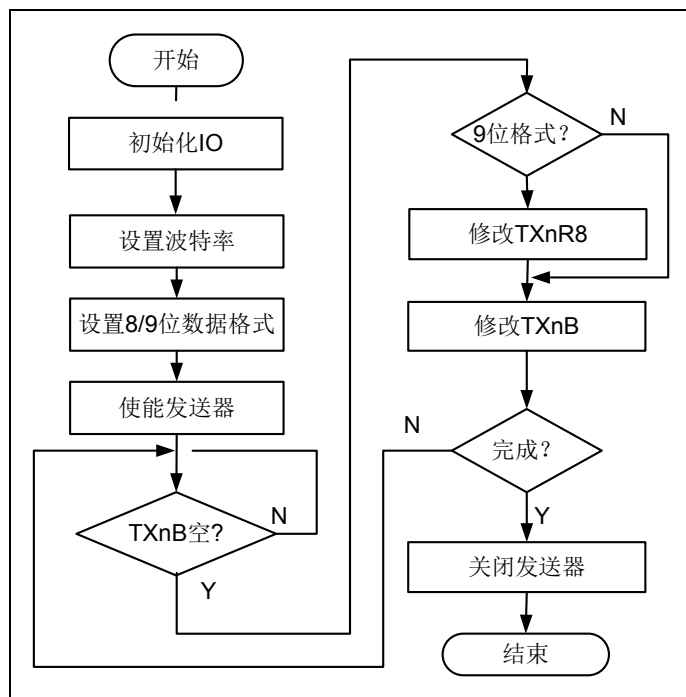


图 6-8 UARTn 发送器操作流程图

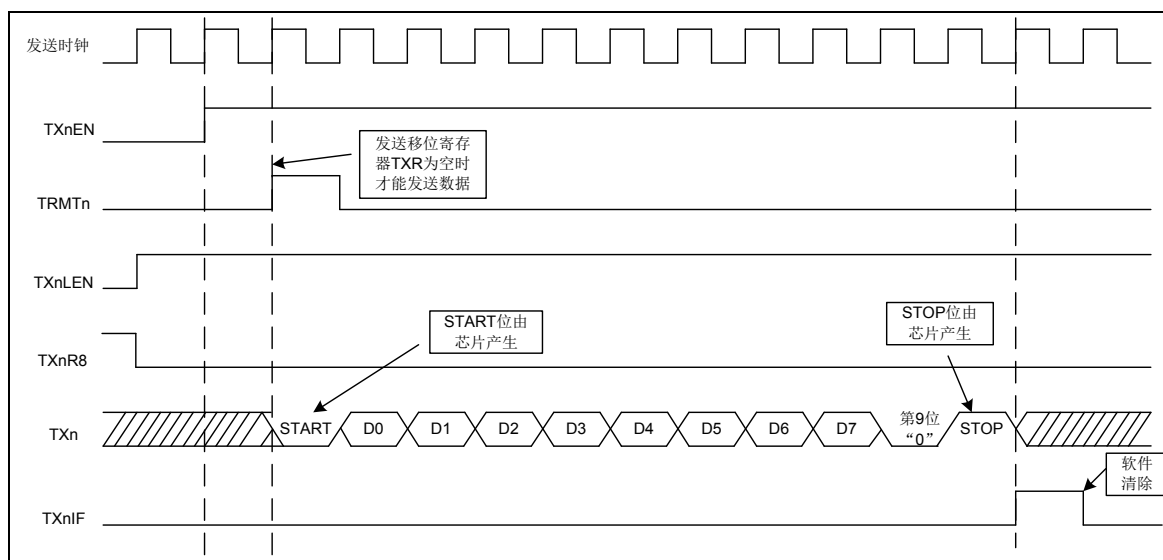


图 6-9 UARTn 发送器发送数据时序图（9 位数据格式，第 9 位数据为“0”）

### 6.2.6 异步接收器

异步接收器接收数据时，用户可以查询  $RXnIF$  中断标志位，来判断是否收到完整的一帧数据，并通过读取  $RXnB$  和  $RXnR8$  获得数据。芯片内部提供 2 级 9 位 FIFO 作为接收数据缓冲区，若用户在第 3 个数据接收完毕前，未读取  $RXnB$ ，则溢出标志位  $OERRn$  将置 1。 $FERRn$  在用户未接收到结束位 STOP 时置 1。注意，由于 UART 接收器接收端口  $RXn$  与 I/O 端口复用，在使用 UART 接收端口前，需首先设置所复用的 I/O 端口处于输入状态。

操作流程图如下:

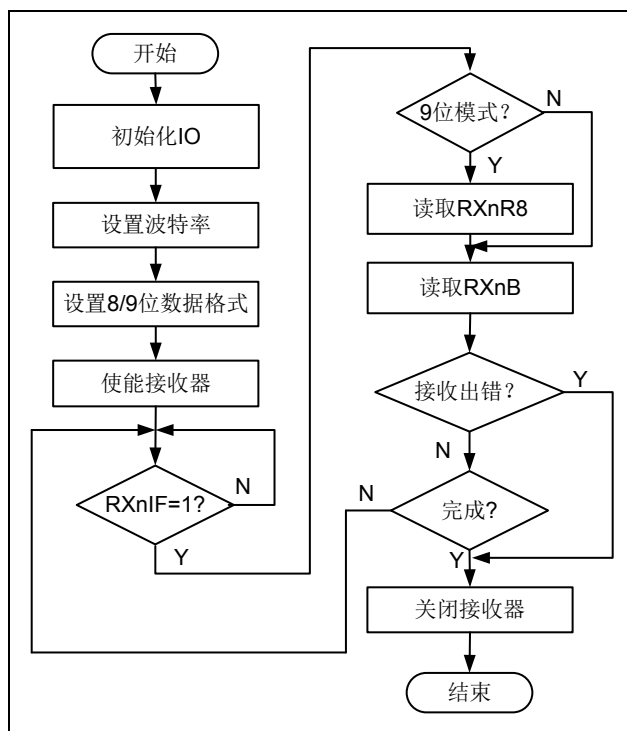


图 6-10 UARTn 接收器操作流程图

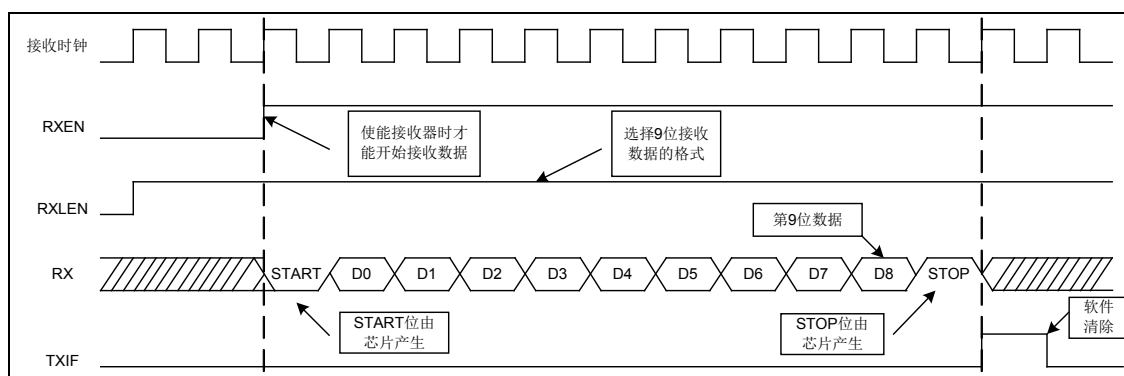


图 6-11 UARTn 接收器接收数据时序图 (9 位数据格式)

### 6.2.7 UARTn使用注意事项

在 UART 电路使能前, 首先要设置复用的 I/O 端口为数字端口, 并将 TXn 管脚需设置为输出口, RXn 管脚设置为输入口, 才能保证在 UART 模块使能后, 成功进行数据的发送或接收。另外, 程序中不建议用户频繁切换 TXn 和 RXn 管脚的输入/输出类型。

### 6.2.8 特殊功能寄存器

BRnR: UARTn 波特率寄存器								
Bit	7	6	5	4	3	2	1	0
Name	BRnR<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 BRnR<7:0>: UARTn 波特率设置, 00<sub>H</sub>~FF<sub>H</sub>

RXnB: UARTn 接收数据寄存器								
Bit	7	6	5	4	3	2	1	0
Name	RXnB<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	x	x	x	x	x	x	x	x

Bit 7~0 RXnB<7:0>: UARTn 接收到的数据

RXnC: UARTn 接收控制/状态寄存器								
Bit	7	6	5	4	3	2	1	0
Name	RXnEN	RXnLEN	—	—	—	OERRn	FERRn	RXnR8
R/W	R/W	R/W	—	—	—	R	R	R
POR	0	0	0	0	0	0	0	0

- Bit 7 RXnEN: 接收器使能位
  - 0: 禁止
  - 1: 使能
- Bit 6 RXnLEN: 接收器数据格式选择位
  - 0: 8 位数据接收格式
  - 1: 9 位数据接收格式
- Bit 5~3 保留
- Bit 2 OERRn: 接收溢出标志位
  - 0: 无溢出错误
  - 1: 有溢出错误 (清 RXnEN 清零)
- Bit 1 FERRn: 帧格式错标志位
  - 0: 无帧格式错误
  - 1: 帧格式错 (读 RXnB 清零)
- Bit 0 RXnR8: 第 9 位接收数据位
  - 0: 第 9 位数据为 0
  - 1: 第 9 位数据为 1

TXnB: UARTn 发送数据寄存器								
Bit	7	6	5	4	3	2	1	0
Name	TXnB<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 TXnB<7:0>: UART0 发送的数据

TXnC: UARTn 发送控制/状态寄存器								
Bit	7	6	5	4	3	2	1	0
Name	TXnEN	TXnLEN	BRGHn	—	—	—	TRMTn	TXnR8
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W
POR	0	0	0	0	0	0	1	0

- Bit 7 TXnEN: 发送器使能位  
0: 禁止  
1: 使能
- Bit 6 TXnLEN: 发送器数据格式选择位  
0: 8 位数据格式  
1: 9 位数据格式
- Bit 5 BRGHn: 波特率模式选择位  
0: 低速模式  
1: 高速模式
- Bit 4~2 保留
- Bit 1 TRMTn: 发送移位寄存器 (TXnR) 空标志位  
0: TXnR 不空  
1: TXnR 空
- Bit 0 TXnR8: 第 9 位发送数据设置  
0: 第 9 位数据为 0  
1: 第 9 位数据为 1



## 6.3 I2C总线从动器 (I2CS)

### 6.3.1 概述

- ◇ 只支持从动模式
  - 支持 7 位从机地址
  - 支持标准 I2C 总线协议，最大传输速率 400Kbit/s
  - 支持 I2CS 端口 SCL/SDA 开漏或者推挽输出
  - 支持 2 级发送/接收缓冲
  - 支持自动发送“未应答”功能
  - 约定数据从最高位开始接收/发送
- ◇ I2CS 支持以下功能组件
  - 5 位 I2C 采样滤波寄存器 (I2CX16)
  - I2C 控制寄存器 (I2CC)
  - 从机地址寄存器 (I2CSA)
  - 发送数据缓冲器 (I2CTB)
  - 接收数据缓冲器 (I2CRB)
  - 中断使能寄存器 (I2CIEC)
  - 中断标志寄存器 (I2CIFC)
- ◇ 中断和暂停
  - 支持接收“起始位+从机地址匹配+发送应答位”中断标志 (I2CSRIF)
  - 支持接收结束位中断标志 (I2CSPIF)
  - 支持发送空中断标志 (I2CTBIF, 只可读)
  - 支持接收满中断标志 (I2CRBIF, 只可读)
  - 支持发送错误标志 (I2CTEIF)
  - 支持接收溢出中断标志 (I2CROIF)
  - 支持接收未应答标志 (I2CNAIF)
  - 在 IDLE 模式下，暂停接收和发送

### 6.3.2 I2CS端口配置

I2CTE	I2CS 时钟端口配置	I2CS 数据端口配置
1	SCL	SDA
0	PB4	PB5

### 6.3.3 通讯协议

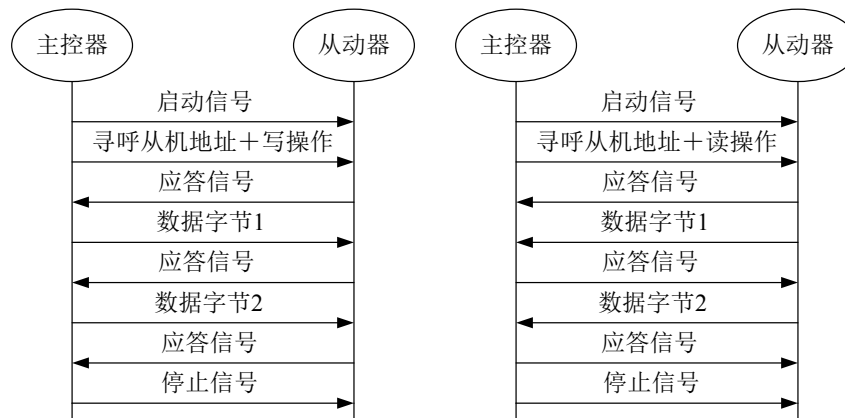


图 6-12 I2C 总线通讯协议示意图

- 注：I2C 通讯中，必须遵循以下协议
- 1: 通讯由主控器发起，发送启动信号 S（开始）控制总线，发送停止信号 P（结束）释放总线；
  - 2: 总线上可以同时有多个主控器（前提是每个主控器都支持多主机仲裁机制），但至少需要一个从动器，且每一个从动器都必须有一个独立且唯一的寻呼地址；
  - 3: 主控器在发送启动信号后，紧接着寻呼从动器地址以及发送读写方式位；
  - 4: 读写控制位 R/#W（称为方向位）用于通知从动器数据传送的方向，“0”表示这次通讯是由主控器向从动器“写”数据，“1”表示这次通讯是由主控器向从动器“读”数据；
  - 5: I2C 通讯协议支持应答机制，即发送方每传送一个字节的的数据（包括寻呼地址），接收方必须回答一个应答信号 (ACK 或者 NACK)，发送方再根据应答信号进行下一步的操作；
  - 6: 每个数据字节在传送时都是高位在前。

### 6.3.4 数据传输格式参考

I2C 存储器的数据传输参考格式如下：

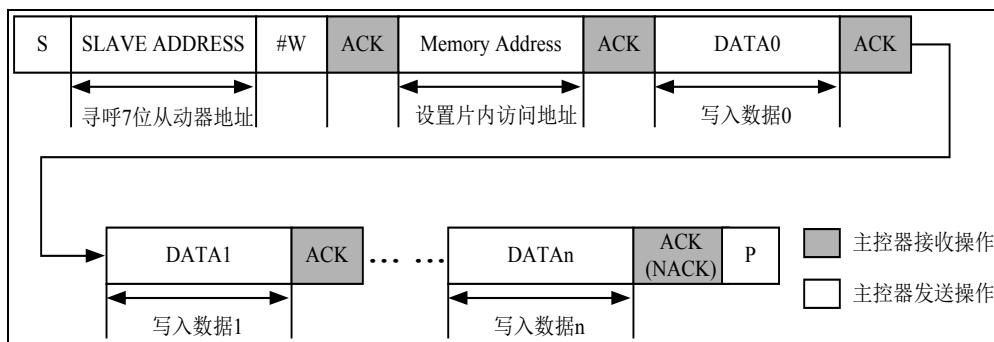


图 6-13 主控器写入从动器数据示意图

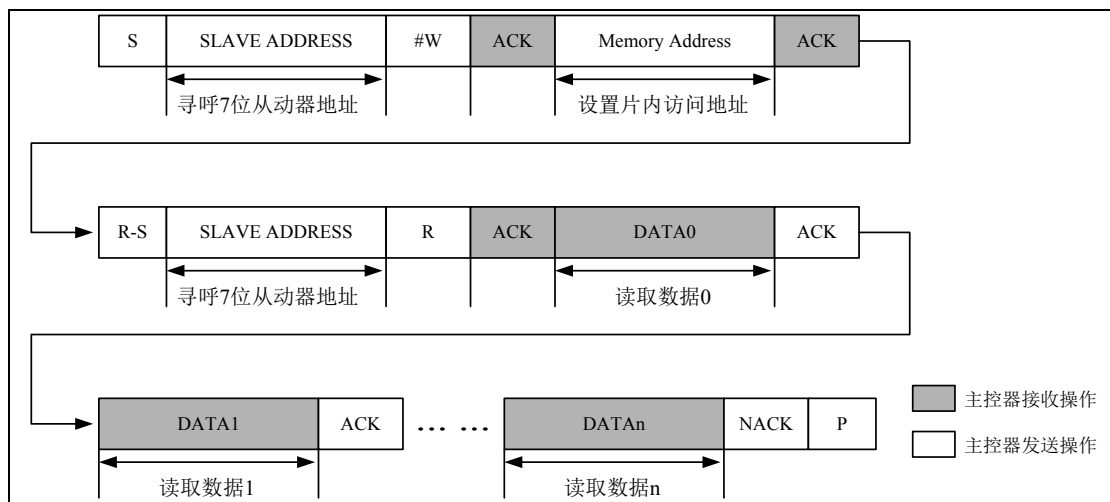


图 6-14 主控器读取从动器数据示意图

### 6.3.5 中断和暂停

当 I2CSRIF、I2CSPIF、I2CTBIF、I2CRBIF、I2CTEIF、I2CROIF 和 I2CNAIF 中任何一个中断标志位置 1 时，I2C 总中断标志位 I2CIF (INTF2<6>) 就会置 1，需要软件清零 I2CIF，在清零 I2CIF 总中断标志位之前，先清零 I2CSRIF、I2CSPIF、I2CTEIF、I2CROIF 和 I2CNAIF 等相关中断标志位。如果中断使能位 I2CIE (INTE2<6>) 和全局中断使能位 GIE (INTG<7>) 使能，则产生 I2C 中断，否则不响应中断。

在 IDLE 模式下，I2CS 模块通讯暂停。

注：GIE、I2CIE 和 I2CIF 位请参考《中断处理》章节中的中断使能寄存器 and 中断标志寄存器。

### 6.3.6 特殊功能寄存器

I2CX16: I2C 采样滤波寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	I2CX16<4:0>				
R/W	—	—	—	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~5 保留

Bit 4~0 I2CX16<4:0>: I2C 采样滤波器设置

01<sub>H</sub>~1F<sub>H</sub>: 通信时钟和数据采样滤波时间为  $T_{osc}(I2CX16+1) \times 3$

00<sub>H</sub> : 禁止采样滤波

I2CC: I2C 控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	I2CTE	I2COD<1:0>		I2CTAS	I2CANAE	—	I2CRST	I2CEN
R/W	R/W	R/W	R/W	R/W	R/W	—	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7            I2CTE: 通信端口使能  
                  0: 禁止  
                  1: 使能
- Bit 6~5        I2COD<1:0>: I2C 开漏输出使能位  
                  00: 禁止  
                  11: 使能  
                  其他: 保留
- Bit 4            I2CTAS: I2C 发送应答设置位  
                  0: 发送 ACK  
                  1: 发送 NACK
- Bit 3            I2CANAE: I2C 自动未应答使能位  
                  0: 禁止  
                  1: 使能
- Bit 2            保留
- Bit 1            I2CRST: 软件复位 I2C 模块  
                  0: 禁止  
                  1: 使能
- Bit 0            I2CEN: I2C 使能模块  
                  0: 禁止  
                  1: 使能

注 1: 当 I2C 自动未应答使能时:  
当片外主控制器寻呼本机地址+R 时, 若 2 级发送数据缓冲器全空, 则本机地址后的应答位为 “NACK”;  
当片外主控制器寻呼本机地址+#W 时, 若在数据接收之前, 2 级接收数据缓冲器全满时, 本机地址后的应答位为 “NACK”; 若在接收数据后, 2 级接收数据缓冲器全满时, 则接收数据后的应答位为 “NACK”。

注 2: 当 I2CTE=1 时, I2COD 控制 SCL/SDA 端口的开漏功能; 否则, 由 PBOD<5:4>控制 PB 端口的开漏功能。

I2CSA: I2C 地址寄存器								
Bit	7	6	5	4	3	2	1	0
Name	I2CSADR<6:0>							I2CRW
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R
POR	0	0	0	0	0	0	0	0

- Bit 7~1        I2CSADR<6:0>:从机地址
- Bit 0            I2CRW: 从机地址匹配后, 自动更新读/写位  
                  0: 读  
                  1: 写

**I2CTB: I2C 发送数据缓冲寄存器**

Bit	7	6	5	4	3	2	1	0
Name	I2CTB<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 I2CTB<7:0>: I2C 发送数据缓冲寄存器

注: 第一个需要发送的数据, 在发送使能前写入发送数据缓冲器。

**I2CRB: I2C 接收数据缓冲寄存器**

Bit	7	6	5	4	3	2	1	0
Name	I2CRB<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 I2CRB<7:0>: I2C 接收数据缓冲寄存器

**I2CIEC: I2C 中断使能寄存器**

Bit	7	6	5	4	3	2	1	0
Name	—	I2CNAIE	I2CROIE	I2CTEIE	I2CRBIE	I2CTBIE	I2CSPIE	I2CSRIE
R/W	—	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7 保留

Bit 6 I2CNAIE: I2C 接收未应答中断使能位

0: 禁止

1: 使能

Bit 5 I2CROIE: I2C 接收溢出中断使能

0: 禁止

1: 使能

Bit 4 I2CTEIE: I2C 发送错误中断使能

0: 禁止

1: 使能

Bit 3 I2CRBIE: I2C 接收满中断使能位

0: 禁止

1: 使能

Bit 2 I2CTBIE: I2C 发送缓冲器未满足中断使能位

0: 禁止

1: 使能

Bit 1 I2CSPIE: I2C 接收结束位中断使能位

0: 禁止

1: 使能

Bit 0 I2CSRIE: I2C 接收“起始位+从机地址匹配+发送应答位”中断使能位

0: 禁止

1: 使能

I2C1FC: I2C 中断使能寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	I2CNAIF	I2CROIF	I2CTEIF	I2CRBIF	I2CTBIF	I2CSPIF	I2CSRIF
R/W	—	R/W	R/W	R/W	R	R	R/W	R/W
POR	0	0	0	0	0	1	0	0

- Bit 7            保留
- Bit 6            I2CNAIF: I2C 接收未应答中断标志位  
                   0: 未接收或者未发送 NACK  
                   1: 接收或发送 NACK , 产生中断标志 (软件清零)
- Bit 5            I2CROIF: I2C 接收溢出中断标志  
                   0: 2 级接收数据缓冲器和 I2C 移位寄存器未全满  
                   1: 2 级接收数据缓冲器和 I2C 移位寄存器全满, 产生中断标志 (软件清零)
- Bit 4            I2CTEIF: I2C 发送错误中断标志  
                   0: 主机读从机数据操作正常  
                   1: 2 级发送数据缓冲器全空, 主机继续读从机数据, 产生中断标志 (软件清零)
- Bit 3            I2CRBIF: I2C 接收满中断使标志  
                   0: 2 级接收数据缓冲器未满  
                   1: 2 级接收数据缓冲器未空时, 产生中断标志
- Bit 2            I2CTBIF: I2C 发送缓冲器未空中断标志位  
                   0: 2 级发送数据缓冲器满  
                   1: 2 级发送数据缓冲器未空时, 产生中断标志
- Bit 1            I2CSPIF: I2C 接收结束位中断标志位  
                   0: 未接收到结束位  
                   1: 接收到结束位, 产生中断标志 (软件清零)
- Bit 0            I2CSRIF: I2C 接收 “起始位+从机地址匹配+发送应答位” 中断标志位  
                   0: 未接收到 “起始位+地址位且地址匹配+发送应答位”  
                   1: 接收到 “起始位+地址位且地址匹配+发送应答位”, 产生中断标志 (软件清零)

注 1: 清总中断标志位 I2CIF 前, 先清除 I2C1FC 寄存器的相关中断标志位;  
 注 2: 连续接收数据超过 2 个时, 发生接收溢出, 并且第 3 个接收数据会丢失;  
 注 3: I2C 模块在每帧数据发送完成后, 接收到结束位时, 硬件自动清零发送缓冲寄存器。

## 6.4 触摸按键控制 (TK)

### 6.4.1 概述

- ◇ 支持最多 3 个触摸按键；一路按键补偿通道
- ◇ 支持任意触摸按键选择，单选、多选或全选
- ◇ 支持 8MHz、4MHz、2MHz、1MHz、500KHz 共 5 种工作频率
- ◇ 支持扫描频率的抖频，抖频范围为 (+100%, -50%)
- ◇ 支持可选基准电压
- ◇ 支持 1 个模拟比较器 ACP；负端参考电压 VREF 可选
- ◇ 支持 Cx 充电补偿，提高按键一致性
- ◇ 支持多次扫描时间的累计或平均运算
- ◇ 支持触摸按键单次扫描中断，累计多次扫描中断

### 6.4.2 内部结构图

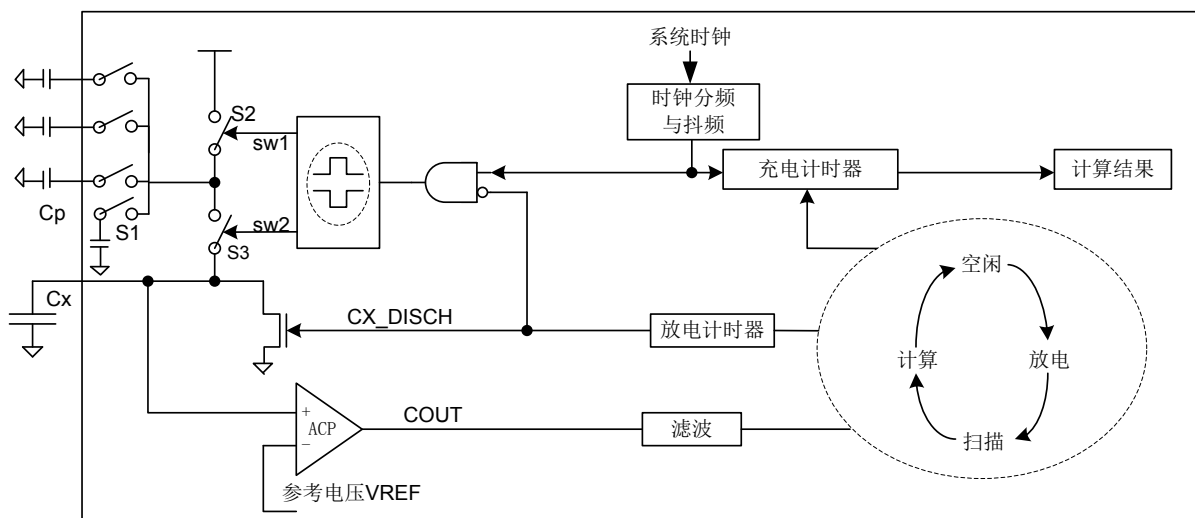


图 6-15 触摸按键触摸结构示意图

### 6.4.3 触摸按键扫描工作原理

本芯片实现一种电容式感应按键检测，电容式感应按键检测方法采用了“电容电荷转移”的工作原理。

在上图中，开关 S1 由按键选通寄存器控制；开关 S2 由脉宽调制后的系统时钟分频信号控制；开关 S3 由脉宽调制后的系统时钟分频信号的互补信号控制。当有手指触摸按键时，相当于增加了一个到地的电容 Cf，实际电容变成 Cp+Cf，相对于未触摸时，电容 Cx 充电的时间将会变短，如下面的触摸状态示意图所示。根据电容 Cx 充电时间的长短，来实现触摸按键的扫描。具体的工作原理如下：

1. S1, S2 闭合, S3 断开, 系统开始对电容 Cp 充电。
2. S1, S3 闭合, S2 断开, 电容 Cp 对电容 Cx 放电。

3. 循环往复 1、2 步骤给电容  $C_x$  充电，直到电容  $C_x$  上的电压大于比较器 ACP 负端的参考电压  $V_{REF}$  时，模拟比较器 COUT 输出高电平，滤波器开始采样处理。
4. 滤波器采样处理后，放电计时器输出高电平，并驱动 N 管导通，此时 S2, S3 断开，电容  $C_x$  开始放电。（为了保证电容  $C_x$  放电完全，可增加放电时间，即用户可根据电容  $C_x$  的容量大小，软件配置不同的放电时间。）
5. 滤波器输出为低电平并且电容  $C_x$  放电结束时，充电计时器将开始统计电容  $C_x$  充电的时间。每统计一次电容  $C_x$  充电的时间即为一次采样结果，软件可配置多次采样，硬件会自动计算多次采样的平均值。
6. 当滤波器输出高电平时，充电计时器把统计的数值送到运算放大器放大，以便增加当前状态值的噪声容限。同时放电计时器开始电容  $C_x$  放电的计时。如果没有错误启动和计算溢出，硬件会自动产生运算结束中断，并加载 24 位的采样值，从而完成一次按键扫描。

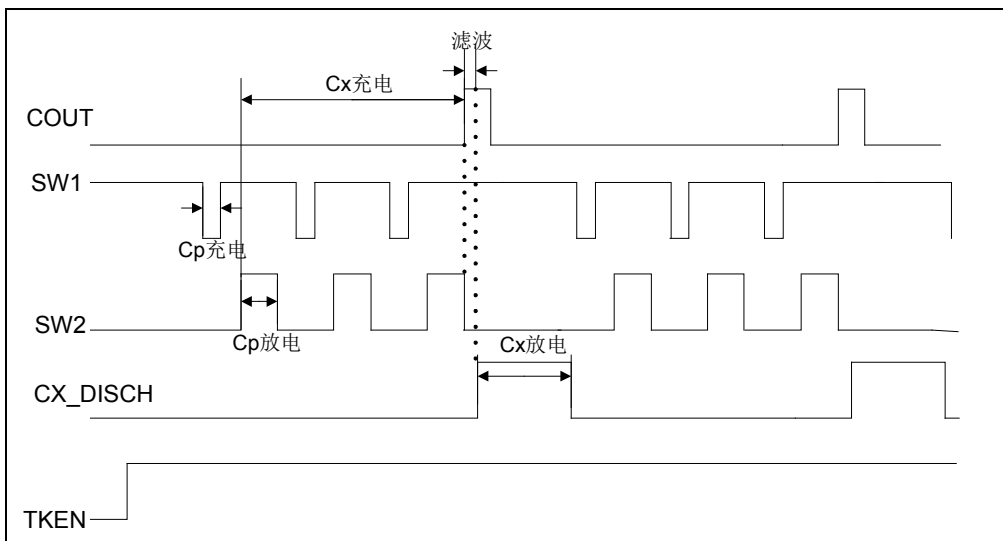


图 6-16 无触摸状态示意图

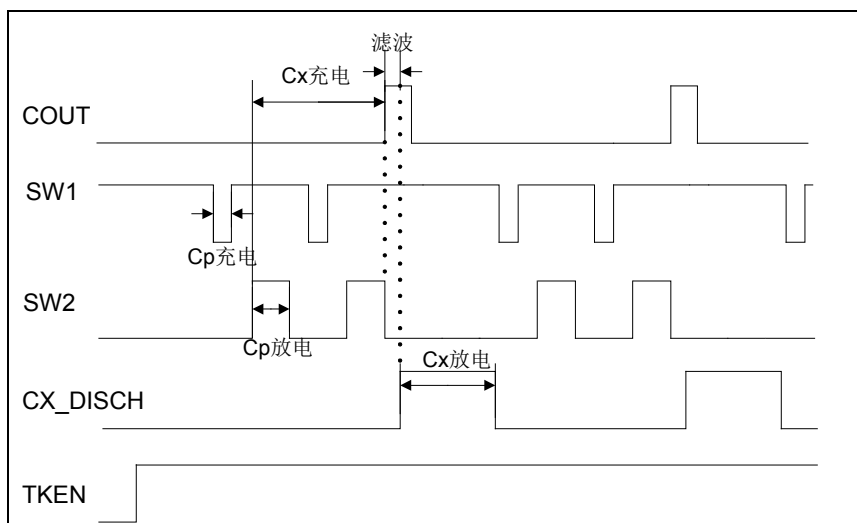


图 6-17 有触摸状态示意图



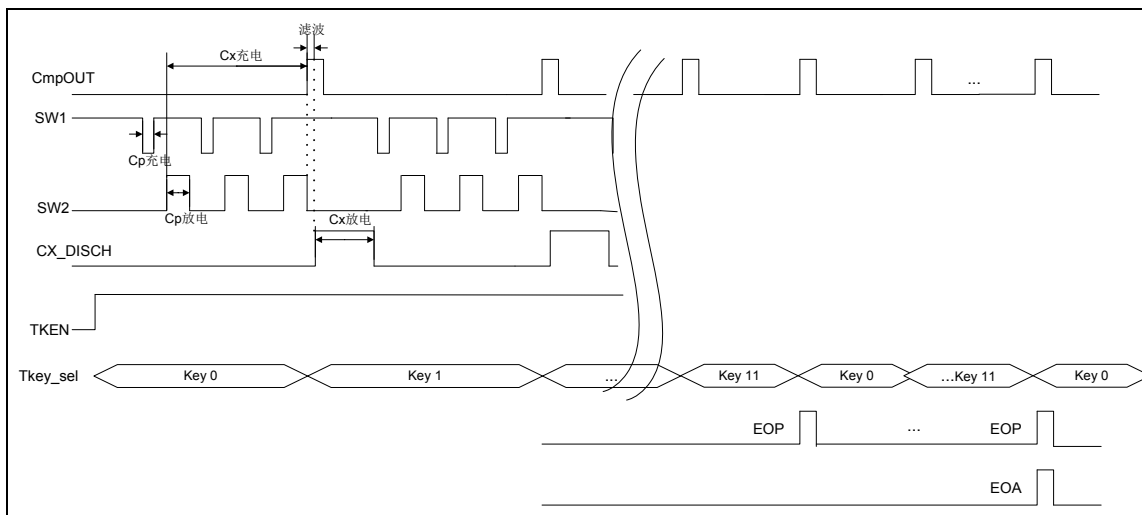


图 6-18 多通道轮询扫描工作示意图

#### 6.4.4 触摸按键端口控制

本芯片共有 3 个触摸按键，分别有一位独立通道寄存器选中相应按键，未被选中的按键端口可用作通用端口，选中且当前扫描到的触摸按键端口被禁止输出，选中而又未扫描的触摸按键端口有以下处理：

1. 当触摸按键模块使能，且寄存器 TKCTR 控制为 1 时，所有选中且未扫描的触摸按键的输出状态固定为 0。
2. 当 TKCTR 设置为 0 时，所有选中且未扫描的触摸按键端口状态完全取决于对应管脚上的其他复用功能，如无复用功能，则取决于 GPIO 的配置，若约束这些端口输出固定电平，则可配置 GPIO 对应寄存器，输出为相应固定电平。

#### 6.4.5 数据管理

本芯片可以有两种扫描模式：一种针对单一通道的 N 次扫描模式，另一种针对任意通道由低到高的顺序轮询扫描的模式；这两种模式可以通过通道选择寄存器和轮询采样次数配置组合如下几种工作方式：

轮询使能 (tkpe)	通道选择 (tkchs)	采样次数 (tktms)	中断产生	说明
0	Single	1	采样一次后	1.对单个选择通道进行单次采样； 2.tkave 设定值不影响结果值； 3.若 tkds=0, {TKxDAH,TKxDAL}存放的是采样值放大结果，否则存放原始采样结果； 4.{TKSDAH,TKSDAL}中存放的值与{TKxDAH,TKxDAL}相同
0	Single	N>1	采样多次后	1.对单个选择通道进行 N 次采样； 2.四种对数据的处理组合为 {tkds,tkave}: 2'b00:{TKxDAH,TKxDAL}中存放对每次采样结果放大后的累计值； 2'b01:{TKxDAH,TKxDAL}中存放对每次采样结果放大后的平均值； 2'b10:{TKxDAH,TKxDAL}中存放 N 次采样结果的累计值。

轮询使能 (tkpe)	通道选择 (tkchs)	采样次数 (tktms)	中断产生	说明
				<p>2'b11:{TKxDAH,TKxDAL}中存放 N 次采样结果的平均值。</p> <p>3.{TKSDAH, TKSDAL}中存放的值与{TKxDAH,TKxDAL}相同。</p>
0	Multi	1	采样一次后	<p>1.对所有选择通道同时打开进行单次采样;</p> <p>2.tkave 设定值不影响结果;</p> <p>3.若 tkds=0, {TKxDAH,TKxDAL}存放的是采样值放大结果, 否则存放原始采样结果, 所有选择通道对应的结果寄存器值相同;</p> <p>4.{TKSDAH, TKSDAL}中存放的值与{TKxDAH,TKxDAL}相同</p>
0	Multi	N>1	采样多次后	<p>1.对所有选择通道同时打开进行 N 次采样;</p> <p>2.四种对数据的处理组合为 {tkds,tkave):</p> <p>2'b00:{TKxDAH,TKxDAL}中存放对每次采样结果放大后的累计值;</p> <p>2'b01:{TKxDAH,TKxDAL}中存放对每次采样结果放大后的平均值;</p> <p>2'b10:{TKxDAH,TKxDAL}中存放 N 次采样结果的累计值。</p> <p>2'b11:{TKxDAH,TKxDAL}中存放 N 次采样结果的平均值。</p> <p>3.{TKSDAH, TKSDAL}中存放的值与{TKxDAH,TKxDAL}相同</p>
1	Single	1	采样一次后	<p>1.对单个选择通道进行单次采样;</p> <p>2.tkave 设定值不影响结果值;</p> <p>3.若 tkds=0, {TKxDAH,TKxDAL}存放的是采样值放大结果, 否则存放原始采样结果;</p> <p>4.{TKSDAH, TKSDAL}中存放的值与{TKxDAH,TKxDAL}相同</p>
1	Single	N>1	采样多次后	<p>1.对单个选择通道进行 N 次采样;</p> <p>2.四种对数据的处理组合为 {tkds,tkave):</p> <p>2'b00:{TKxDAH,TKxDAL}中存放对每次采样结果放大后的累计值;</p> <p>2'b01:{TKxDAH,TKxDAL}中存放对每次采样结果放大后的平均值;</p> <p>2'b10:{TKxDAH,TKxDAL}中存放 N 次采样结果的累计值。</p> <p>2'b11:{TKxDAH,TKxDAL}中存放 N 次采样结果的平均值。</p> <p>3.{TKSDAH, TKSDAL}中存放的值与{TKxDAH,TKxDAL}等同。</p>
1	Multi	1	轮询采样所有选择通道一次之后	<p>1.对所有选择通道中逐个按序轮询采样单次</p> <p>2.tkave 设定值不影响结果值;</p> <p>3.若 tkds=0, {TKxDAH,TKxDAL}存放的是对应选择通道的采样值得放大结果, 否则存放原始采样结果;;</p> <p>4.{TKSDAH, TKSDAL}中存放为无效值, 没有参考意义。</p>
1	Multi	N>1	轮询采样所有选择通道多次之后	<p>1.所有选择通道中逐个按序轮询采样 N 次</p> <p>2.四种对数据的处理组合为 {tkds,tkave):</p> <p>2'b00:{TKxDAH,TKxDAL}中存放对应选择通道每次采样结果放大后的累计值;</p> <p>2'b01:{TKxDAH,TKxDAL}中存放对应选择通道每次采样结果放大后的平均值;</p> <p>2'b10:{TKxDAH,TKxDAL}中存放对应选择通道 N 次采样结果的累计值。</p> <p>2'b11:{TKxDAH,TKxDAL}中存放对应选择通道 N 次采样结果的平均值。</p> <p>3.{TKSDAH, TKSDAL}中存放为无效值, 没有参考意义。</p>

6.4.6 触摸按键扫描参考流程

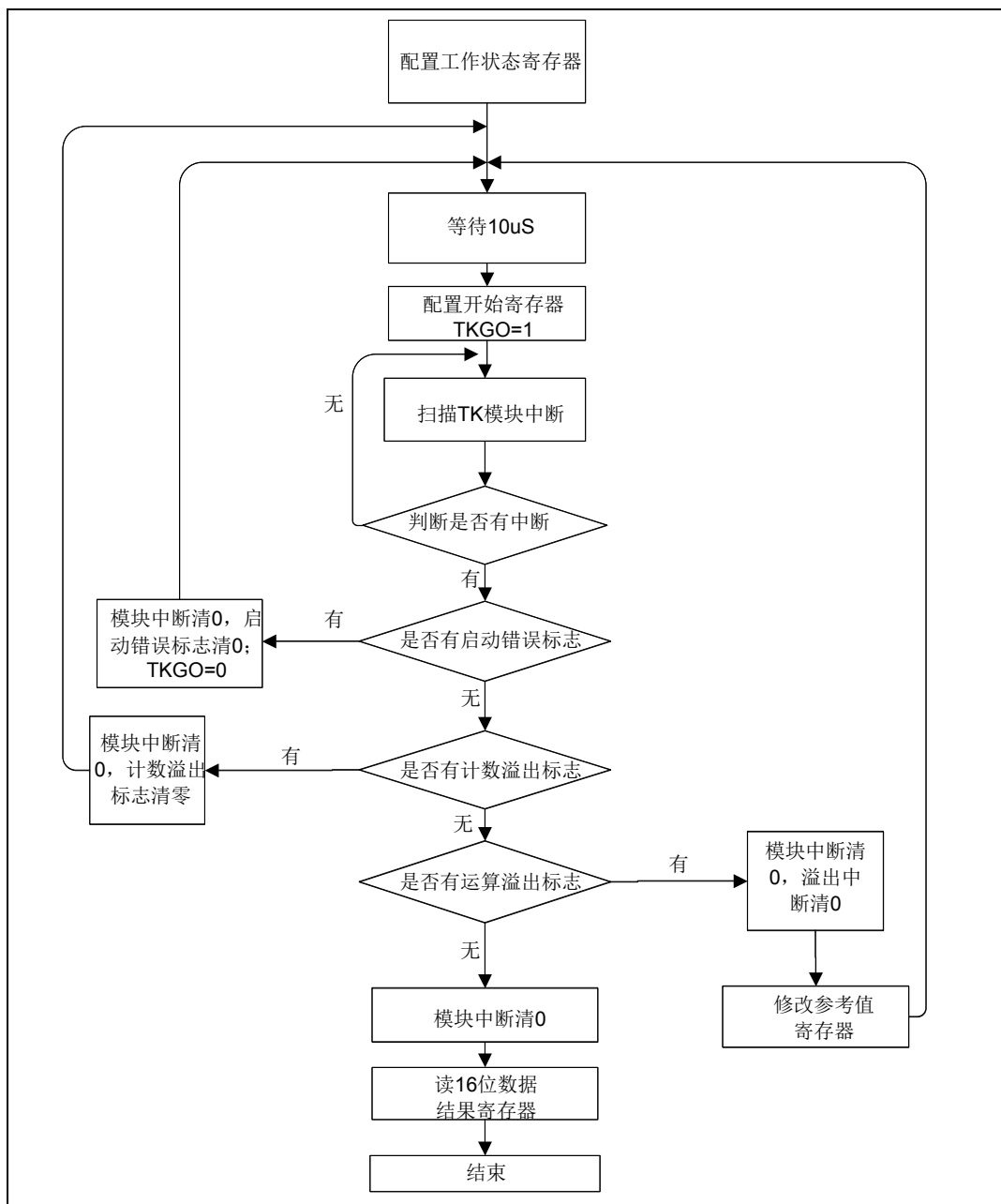


图 6-19 触摸按键操作流程

注 1: TK 启动错误标志 TKERR, 主要用于判断 Cx 放电时间设置和 Cx 电容大小的选择是否合适。

- a) 发生扫描启动错误时, 优先考虑增大 Cx 放电时间: TKGO 写 0 清 TKERR 标志位, 设置 TKDIST 寄存器将放电时间加大; 重新将 TKGO 置 1 启动扫描。
- b) 若即使将放电时间加至最大, 仍然发生扫描启动错误, 需考虑减小 Cx 电容值。
- c) 以上 a) 和 b) 方法在系统调试阶段使用。一旦系统调试稳定, 仍然发生 TK 启动错误问题, 如上图操作流程所示, 不修改放电时间, 通过多次启动, 完成对 Cx 电容的放电。用户可软件设定重复次数, 达到某一设定次数后, 若仍然产生 TKERR 错误, 则报系统错误, 提示硬件故障。

注 2: 充电计时器溢出标志 SCANOV 主要用于判断 TK 工作频率, 模拟比较器参考电压等是否合适。

- a) 设置 TKFS<1:0>适当降低工作频率, 保证 Cp 有足够时间可充电至与 Cx 上电压有足够的压差。
- b) 检查 Cp 充电电压与模拟比较器负端参考电压是否设置合理。如 Cp 充电电压为 2.6V, 则模拟比较器设置

为 2.5V 是不合适的，没有保证足够的压差。

注 3: 运算溢出标志 TKOV 主要用于判断放大系数设置是否合理。

a) 28 位放大系数为将触摸与非触摸的运算差值放大，如果运算结果的高位 25~28 位的数据不为零（此四位用户不可读），则会发生运算结果溢出，运算结果溢出标志 TKOV 会置 1。如果运算结果溢出，也就是 24 位运算结果大于 fffffH 时，则应当适当减小 28 位放大系数寄存器的值。

### 6.4.7 特殊功能寄存器

TKSELL: 触控按键功能选择寄存器								
Bit	7	6	5	4	3	2	1	0
Name	TKCHSL<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7~0 TKCHSL<7:0>: 按键通道选择位, 1-enable, 0-disable  
 TKCHSL<0>: Touch key0 selected enable  
 TKCHSL<1>: Touch key1 selected enable  
 TKCHSL<2>: Touch key2 selected enable  
 TKCHSL<3:7>: 保留

TKTUN: 触控调整寄存器								
Bit	7	6	5	4	3	2	1	0
Name	TKCHGS	TKDIST<2:0>			TKCFT<1:0>		TKDCE	TKDCS
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7 TKCHGS: TK 充电增强控制位  
 0: 充电电流不增强  
 1: 充电电流增强

Bit 6~4 TKDIST<2:0>: Cx 放电时间选择位  
 000: 32\*Tosc (2us @16MHz)  
 001: 64\*Tosc  
 010: 128\*Tosc  
 011: 256\*Tosc  
 100: 384\*Tosc  
 101: 512\*Tosc  
 110~111: 64\*Tosc

Bit 3~2 TKCFT<1:0>: 比较器输出滤波时间选择位  
 00: 无滤波  
 01: 2\*Tosc  
 10: 4\*Tosc  
 11: 8\*Tosc

Bit 1 TKDCE: 按键降耦使能位  
 0: 禁止按键降耦  
 1: 打开按键降耦

Bit 0 TKDCS: 按键降耦模式选择位

- 0: 降耦按键同频同相波形
- 1: 降耦按键同频反相波形（建议使用该模式）

TKCTL0: 触控控制寄存器 0								
Bit	7	6	5	4	3	2	1	0
Name	—	SCANOV	TKERR	TKOV	TKCCE	TKCTR	TKEN	TKGO
R/W	—	R	R	R	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7           保留
- Bit 6           SCANOV: 扫描计数器溢出标志位  
                  0: 未发生扫描计数器溢出  
                  1: 发生扫描计数器溢出
- Bit 5           TKERR: 扫描启动错误标志位  
                  0: 未发生扫描启动错误  
                  1: 发生扫描启动错误
- Bit 4           TKOV: 运算结果溢出标志位  
                  0: 本次运算结果未溢出  
                  1: 本次运算结果溢出
- Bit 3           TKCCE: 按键补偿电容使能位  
                  0: 禁止  
                  1: 使能
- Bit 2           TKCTR: 未扫描按键输出低电平使能位  
                  0: 禁止  
                  1: 使能
- Bit 1           TKEN: 触摸按键使能控制位  
                  0: 关闭触摸按键模块  
                  1: 使能触摸按键模块
- Bit 0           TKGO: 按键启动扫描控制位  
                  0: 未启动按键扫描  
                  1: 按键扫描进行中

注 1: 当 TK 被使能时, 若 TKCTR 为 1'b0 时, 在未运行按键扫描时的相应端口被设置为通用端口, 软件可使其数字输入和输出。通过比较输出和输入值的不同来判断该触控通道是否存在干扰。并采取相应抗干扰处理。亦可配置 TKCTR 为 1'b1, 将扫描按键的邻近未扫描键输出低电平, 提高抗干扰性。

注 2: 当 TKERR 有效, 发生扫描启动错误时, 可尝试将 TKGO 清零, 设置 TKDIST 寄存器将放电时间加大, 然后重新将 TKGO 置 1 启动扫描。若即使将放电时间加至最大, 仍然发生扫描启动错误, 则可能所使用 Cx 电容过大, 可考虑减小该电容值。

注 3: 若发现按键一致性差, 可尝试将充电补偿 TKCCE 设置为 1, 并联补偿电容, 提高按键扫描的一致性。

**TKCTL1: 触控控制寄存器 1**

Bit	7	6	5	4	3	2	1	0
Name	TKMS	TKDS	TKAVE	—	TKTMS<1:0>		—	TKPE
R/W	R/W	R/W	R/W	R	R	R	R	R/W
POR	0	0	0	0	0	0	0	0

Bit 7 TKMS: 触摸按键模式选择位

- 0: 选择充电次数来作为数据单位
- 1: 选择充电时间来作为数据单位

Bit 6 TKDS: TK 充放电结果数据选择位

- 0: TK 输出数据为放大系数寄存器/TK 充放电的计数值
- 1: TK 输出数据为 TK 充放电的计数值

Bit 5 TKAVE: TK 通道充电时间累加值取平均使能

- 0: 对充电时间累加和后不取平均, 保留累加值
- 1: 对充电时间累加和后取平均

Bit 4 保留

Bit 3~2 TKTMS<1:0>: 采样次数选择位

- 00: 采样 1 次, 选择 1 次采样
- 01: 采样 4 次, 选择 4 次采样
- 10: 采样 8 次, 选择 8 次采样
- 11: 采样 16 次, 选择 16 次采样

Bit 1 保留

Bit 0 TKPE: TK 通道的轮询扫描使能

- 0: 对 TK 通道选择的所有通道进行同时充电, 不轮询扫描
- 1: 对 TK 通道选择的所有通道按照由低到高的顺序进行循环轮询扫描

**TKFCTL: 触控频率控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	TKDUS	TKFS<2:0>			—	TKJFS<1:0>		TKJFE
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7 TKDUS: 按键充放电开关占空比控制位, TKJFE=0 时, 本位选择有效

- 0: 1/4 占空比
- 1: 1/2 占空比 (当系统时钟源小于等于 8MHz 时, 只能选占空比 1/2)

Bit 6~4 TKFS<2:0>: 按键充放电开关频率选择位

- 000: 8MHz
- 001: 4MHz
- 010: 2MHz
- 011: 1MHz
- 100: 0.5MHz
- other: 0.5MHz

Bit 3 保留

Bit 2~1 TKJFS<1:0>: 按键开关分频抖频时钟源选择位

- 00: 选择系统时钟作为分频抖频电路时钟源
- 01: 选择内部高频 32MHz 作为分频抖频电路时钟源 (Fosc 必须为 16MHz)

10: 选择内部高频 64MHz 作为分频抖频电路时钟源 (Fosc 必须为 16MHz)

11: 保留

(注: 不同开关频率, 对应抖频范围如下:

开关频率 8MHz: 8M~4M

开关频率 4MHz: 8M~2M

开关频率 2MHz: 4M~1M

开关频率 1MHz: 2M~0.5M

开关频率 0.5MHz: 1M~0.25M

)

Bit 0 TKJFE: 按键抖频使能位

0: 对按键抖频禁能

1: 对按键抖频使能

注 1: 在系统时钟不为 16MHz, 比如 2 分频等, 则 TKFS 只能选择 00, 其抖频范围也会降低对应比例。

注 2: 只能在系统时钟为 16MHz 时, TKFS 才可配置为 01 或 10, 其他情况**请勿选择**。

注 3: 按键抖频指的是对触摸按键充放电开关时钟的频率会在一定范围内变化, 用于提高系统抗干扰性。

注 4: 当系统时钟抖频使能 (TKJFE=1) 时, TKDUS 的选择被视为无效。

**TKSDAL: 合并扫描结果寄存器低 8 位**

Bit	7	6	5	4	3	2	1	0
Name	TKSDAL<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	0	0	0	0	0	0	0	0

Bit 7~0 TKSDAL<7:0>: 扫描结果寄存器低 8 位

**TKSDAH: 合并扫描结果寄存器高 8 位**

Bit	7	6	5	4	3	2	1	0
Name	TKSDAH<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	0	0	0	0	0	0	0	0

Bit 7~0 TKSDAH<7:0>: 扫描结果寄存器高 8 位

**TKMODL: 放大系数寄存器低 8 位**

Bit	7	6	5	4	3	2	1	0
Name	TKMODL<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	1	1	1	1	1	1	1	1

Bit 7~0 TKMODL<7:0>: 放大系数寄存器低 8 位

**TKMODM: 放大系数寄存器次低 8 位**

Bit	7	6	5	4	3	2	1	0
Name	TKMODM<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	1	1	1	1	1	1	1	1

Bit 7~0 TKMODM<7:0>: 放大系数寄存器次低 8 位

**TKMODH: 放大系数寄存器次高 8 位**

Bit	7	6	5	4	3	2	1	0
Name	TKMODH<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	1	1	1	1	1	1	1	1

Bit 7~0 TKMODH<7:0>: 放大系数寄存器次高 8 位

**TKMODU: 放大系数寄存器高 4 位**

Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	TKMODU<7:0>			
R/W	—	—	—	—	R	R	R	R
POR	—	—	—	—	1	1	1	1

Bit 7~0 TKMODU<7:0>: 放大系数寄存器高 4 位

注 1: 放大系数寄存器的作用是将触摸与非触摸的差值放大。

**VRC: 内部参考电压控制寄存器**

Bit	7	6	5	4	3	2	1	0
Name	—	—	—	—	VRCS<2:0>			HYSEN
R/W	—	—	—	—	R/W	R/W	R/W	R/W
POR	—	—	—	—	0	0	0	1

Bit 7~4 保留位

Bit 3~1 VRCS<2:0>: 内部参考电压选择位

- 000: 0.6V
- 001: 1.5V
- 010: 1.6V
- 011: 2.1V
- 100: 2.2V
- 101: 2.3V
- 110: 2.5V
- 111: 2.6V

Bit 0 HYSEN: 比较器迟滞使能

- 0: 禁止
- 1: 使能

注: 在出厂前, 芯片 2.6V 参考电压已经在常温 25°C 条件下校准, 校准精度在 ±2% 以内。



TKxDAL: TKx 结果寄存器低 8 位(x 取值为 0..11)								
Bit	7	6	5	4	3	2	1	0
Name	TKxDAL<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	0	0	0	0	0	0	0	0

Bit 7~0      TKxDAL<7:0>: 扫描结果寄存器低 8 位

TKxDAH: TKx 结果寄存器高 8 位(x 取值为 0..11)								
Bit	7	6	5	4	3	2	1	0
Name	TKxDAH<7:0>							
R/W	R	R	R	R	R	R	R	R
POR	0	0	0	0	0	0	0	0

Bit 7~0      TKxDAH<7:0>: 扫描结果寄存器高 8 位

## 6.5 模/数转换器模块 (ADC)

### 6.5.1 概述

模拟数字转换器用于将模拟信号转化成一组二进制代码组成的数字信号。模拟信号经由多路复用输入脚输入，通过一个采样-保持电路连接至转换器的输入端。

芯片支持 12-bit 2 通道的 A/D 转换器，经过 A/D 转换器转换的 12-bit 二进制数据存入 ADC 数据寄存器 ADCRH、ADCRL 中。

#### ◇ 模/数转换器特性

- 9 位 AD 采样精度
- 2 个模拟输入通道可选
- 12 位转换结果，支持高位对齐放置或低位对齐放置
- 可配置 AD 采样时间
- 支持高/低速转换选择
- 多种转换时钟频率可选
- 可配置多种参考源

#### ◇ 主要功能组件

- ADC 转换值寄存器 (ADCRL, ADCRH)
- ADC 控制寄存器 (ADCCL, ADCCH)
- 数模端口控制寄存器 PBS

#### ◇ 中断和暂停

- 支持 AD 转换中断
- 在 IDLE 模式下，当使用 LRC 时钟源时，ADC 保持工作，其中断可唤醒 CPU

### 6.5.2 ADC 内部结构图

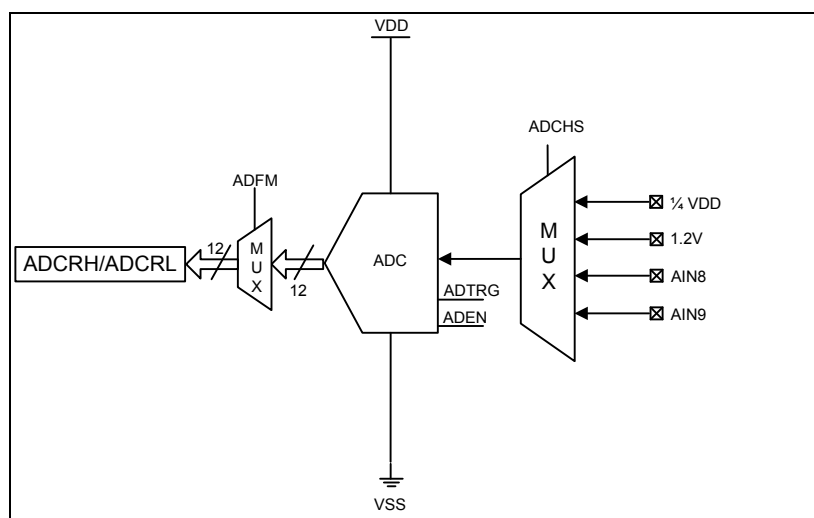


图 6-20 ADC 内部结构图

### 6.5.3 ADC配置

ADC 电路使用前，根据需要对以下几个方面进行正确的配置，才可得到正确转换结果。

#### 时钟选择

ADC 转换时钟频率可选  $F_{osc}$ 、 $F_{osc}/2$ 、 $F_{osc}/4$ 、 $F_{osc}/8$ 、 $F_{osc}/16$ 、 $F_{osc}/32$ 、 $F_{osc}/64$  或 LRC，可通过 ADCKS<2:0>寄存器选择所需要的时钟。

#### 参考电压

ADC 电路参考电压仅支持 VDD。

#### 采样时间选择

硬件采样，可通过 ADCCH 寄存器中的 ADST <1:0>位选择 2/4/8/16 个  $T_{adclk}$  四种采样时间。

软件采样，SMPS 位设置为 0，使能软件采样时，采样时间长度由软件等待时间决定。在将 SMPON 位置 1 使能 ADC 采样后，软件可等待任意长时间，再将 SMPON 清零来结束当前采样。

#### 复用端口类型选择

本芯片中 ADC 电路的所有模拟输入通道 AINx、参考电压外部输入脚均和 PB 端口复用，在使用 ADC 电路转换前，须先将所使用的管脚通过 PBS 寄存器设置为模拟类型。

#### 模拟信号输入通道选择

ADC 电路使能前，需先选择 A/D 模拟通道。本芯片 ADC 电路支持 2 个外部通道、一个  $1/4V_{DD}$  电压和一路内部 1.2V 基准电压可选，外部通道分别为 AIN8~AIN9。A/D 模拟通道选择哪个通道可通过 ADCCL 寄存器中的 ADCHS <3:0>位选择。

$1/4V_{DD}$  电压通道可用于电源电压监控等运用。

内部 1.2V 基准电压通道可用于参考电压 VDD 的校准等运用。

内部 1.2V 基准电压精度为  $1.2V \pm 1\%$  (@25°C)，具体参数特性可参考附录 3.2 内部 1.2V 基准电压精度特性图。

#### 对齐方式选择

本芯片 ADC 电路转换的结果支持两种对齐方式，低位对齐和高位对齐，可通过 ADCCH 寄存器中的 ADFM 位进行选择。

### 6.5.4 ADC转换步骤

下面概述实现 ADC 转换过程的各个步骤。

Step 1: 选择 ADC 转换时钟，通过 ADCCH 寄存器中的 ADCKS <2:0>选择 ADC 转换时钟。

Step 2: 选择 ADC 采样时间，通过 ADCCH 寄存器中的 A/D 采样时间选择位 ADST <1:0>设定。

Step 3: 设置复用端口设为模拟类型，即选择哪些管脚作为 ADC 转换输入管脚，由端口类型选择寄存器 PBS 控制选择。

Step 4: 选择模拟信号输入通道 AINx，通过 ADCCL 寄存器中的 ADCHS <3:0>选择 ADC 模拟通道。

Step 5: 设置转换结果对齐方式，通过 ADCCH 寄存器中的 ADFM 位，选择高位对齐放置还是低位对齐放置。

Step 6: 如果要使用中断，则中断控制寄存器需要正确地设置，以确保 ADC 中断功能被正确激活。在默认中断模式时，需将全局中断使能位/高优先级中断使能位 GIE 置“1”，将 ADC 中断使能位置“1”；在向量中断模式时，需将全局中断使能位/高优先级中断使能位 GIE 置“1”，根据 ADC 中断所在组的优先级决定是否要开启低优先级中断使能位 GIEL，将 ADC 中断使能位置“1”。

Step 7: 使能 ADC 电路，将 ADCCL 寄存器中的 ADC 使能位 ADEN 设置为“1”。

Step 8: 将 ADCCL 寄存器中的 ADC 转换启动位 ADTRG 位设置为“1”，开始 ADC 转换。

Step 9: 轮询 ADCCL 寄存器中的转换状态位 ADTRG 位，确定此次 ADC 转换是否完成。

Step 10: 读取 ADCRH 和 ADCRL 寄存器中的转换结果。

### 6.5.5 ADC时序特征示意图

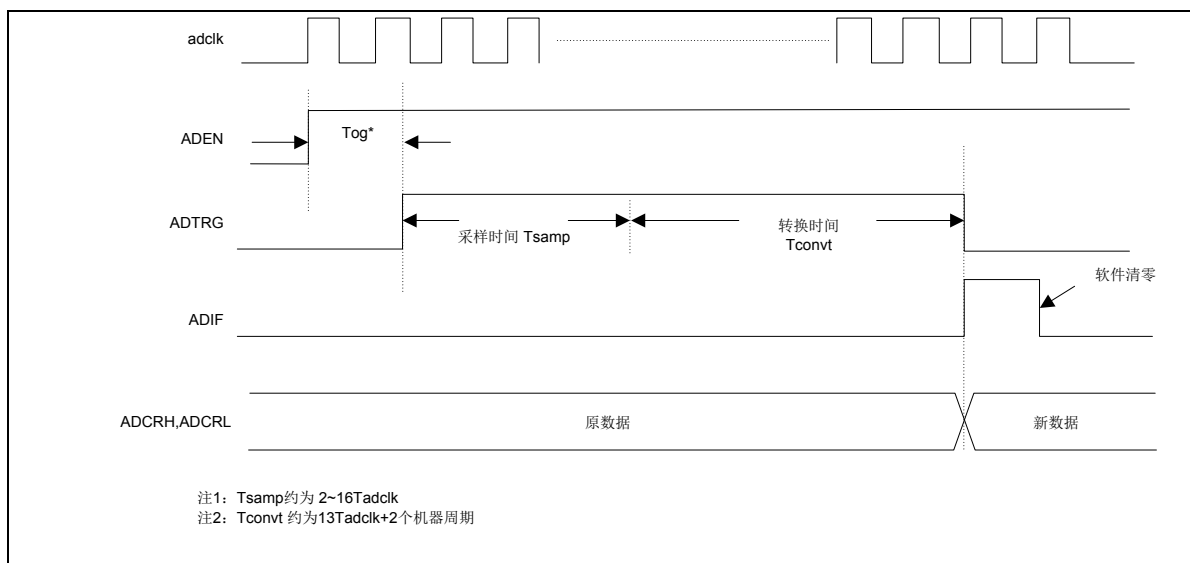


图 6-21 ADC 时序特征示意图

注 1:  $Tog > 0$ ;

注 2: AD 转换时钟周期  $T_{adclk}$ ，可通过  $ADCKS <2:0>$  寄存器配置不同的频率。

注 3: ADC 时钟周期  $T_{adclk} \geq 0.5\mu s$ ，ADC 转换时间表参考附录所示。

**6.5.6 特殊功能寄存器**

ADFM	ADCRH								ADCRL							
	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
1	—	—	—	—	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0
0	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	—	—	—	—

Bit 11~0 D11~D0: ADC 转换结果

ADCCL: ADC 控制寄存器低 8 位									
Bit	7	6	5	4	3	2	1	0	
Name	ADCHS<3:0>				SMPON	SMPS	ADTRG	ADEN	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	1	0	0	

Bit 7~4 ADCHS<3:0>: A/D 模拟通道选择位

- 0000~0111: 保留
- 1000: 通道 8 (AIN8)
- 1001: 通道 9 (AIN9)
- 1010: 保留
- 1011: 保留
- 1100: 内部 1.2V 基准电压
- 1101: 1/4 VDD
- 1110: 保留, 用作测试
- 1111: 保留, 用作测试

Bit 3 SMPON: A/D 采样软件控制位

- 0: 结束采样
- 1: 启动采样

Bit 2 SMPS: A/D 采样模式选择位

- 0: 使能软件采样, 硬件采样禁止
- 1: 禁止软件采样, 硬件采样使能

Bit 1 ADTRG: ADC 转换启动位

- 0: ADC 未进行转换, 或 A/D 转换已完成
- 1: ADC 转换正在进行, 该位置 1 启动 A/D 转换  
当 SMPS=1 时, 该位软件置 1 启动 A/D 采样转换  
当 SMPS=0 时, SMPON=1 启动采样, 该位硬件自动置 1

Bit 0 ADEN: ADC 使能位

- 0: 关闭
- 1: 使能

ADCCH: ADC 控制寄存器高 8 位									
Bit	7	6	5	4	3	2	1	0	
Name	ADFM	ADCKS<2:0>			ADST<1:0>		ADVCMHS	ADHSEN	
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	1	0	0	1	0	1	1	

Bit 7 ADFM: 结果对齐方式选择位

- 0: 高位对齐 (ADCRH<7:0>, ADCRL<7:4>)

- 1: 低位对齐 (ADCRH<3:0>, ADCRL<7:0>)
- Bit 6~4      ADCKS<2:0>: A/D 转换时钟频率 (Tadclk) 选择位
- 000: Fosc
  - 001: Fosc/2
  - 010: Fosc/4
  - 011: Fosc/8
  - 100: Fosc/16
  - 101: Fosc/32
  - 110: Fosc/64
  - 111: LRC 时钟
- Bit 3~2      ADST<1:0>: A/D 采样时间选择位
- 00: 2 个 Tadclk
  - 01: 4 个 Tadclk
  - 10: 8 个 Tadclk
  - 11: 16 个 Tadclk
- Bit 1      ADVCMHS: AD VCM 高速模式控制位 (必须软件设置为 1, 选择高速模式)
- 0: 禁止
  - 1: 使能
- Bit0      ADHSEN: AD 转换速度控制位 (必须设置为 1, 选择高速模式)
- 0: 低速
  - 1: 高速

## 6.6 低电压检测模块 (LVD)

### 6.6.1 概述

芯片支持低电压检测功能，即 LVD，用于监测电源电压 VDD。在供电电源不稳定的情况下，如外部电源噪声串扰或 EMC 测试条件下，会使电源剧烈波动。在目标电压未稳定时，可能会低于芯片工作电压。若检测到电压低于所设定的阈值，可提供一个警告信号。低电压检测也可产生中断信号。

### 6.6.2 LVD操作

LVD 功能的使能由 LVDC 寄存器中的 LVDEN 控制位设置。当 LVDEN 位清零时，LVD 功能禁能。当 LVDEN 位置高时，LVD 功能使能。LVD 模块将电源电压 VDD 与预先设定的阈值电压进行比较，比较结果通过 LVDC 寄存器的 LVDO 位进行查询。预置电压的阈值由 LVDC 寄存器中的 LVDVS 配置，当检测电源电压 VDD 时，预置电压阈值范围为 2.0V~4.7V。当目标电压低于预置电压阈值时，LVDO 位被置高，表明检测到低电压产生。当 LVDO 变化时，产生 LVD 中断标志，触发 LVD 中断标志的条件可选择为 LVDO 上升沿产生、LVDO 下降沿产生和 LVDO 上升沿或下降沿都产生。当 LVD 中断使能开启时产生 LVD 中断请求。在睡眠模式下 LVD 中断可唤醒芯片。

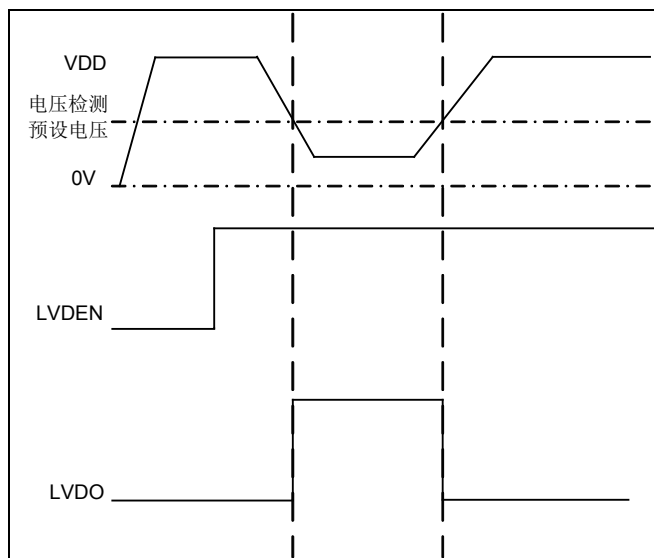


图 6-22 LVD 工作时序图

### 6.6.3 特殊功能寄存器

LVDCIL: LVD 控制寄存器低 8 位								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	LVDFLTEN	LVDIFS<1:0>		LVDO	LV DEN
R/W	—	—	—	R/W	R/W	R/W	R	R/W
POR	0	0	0	0	0	0	x	0

- Bit 7~5 保留
- Bit 4 LVDFLTEN: LVD 滤波器使能  
0: 禁止  
1: 使能
- Bit 3~2 LVDIFS <1:0>: LVD 中断标志产生模式选择位  
00: LVDO 上升沿产生中断  
01: LVDO 下降沿产生中断  
10: LVDO 上升或下降沿都产生中断  
11: 保留
- Bit 1 LVDO: LVD 输出状态位  
0: 被监测电压高于电压阈值  
1: 被监测电压低于电压阈值
- Bit 0 LV DEN: LVD 使能位  
0: 禁止  
1: 使能

注 1: 当 IAP 擦除或编程时, 为防止误擦或误编程, 建议使能 LVD, 在 LVD 有效时, 自动退出 IAP 操作;  
注 2: 当 LVD 滤波器使能时, 约 200us 宽度的 LVD 低压信号被滤除。为防止误编程、误擦除, 建议禁止 LVD 滤波。

LVDCIH: LVD 控制寄存器高 8 位								
Bit	7	6	5	4	3	2	1	0
Name	LVDVS<3:0>				—	—	—	—
R/W	R/W	R/W	R/W	R/W	—	—	—	—
POR	0	0	0	0	0	0	0	0

- Bit 7~4 LVDVS<3:0>: LVD 触发电压选择 (电压为设计值)  
0011: 2.4V  
0100: 2.6V  
0101: 2.8V  
0110: 3.0V  
0111: 3.6V  
1000: 4.0V  
1001: 4.6V  
其他: 保留
- Bit 3~0 保留



## 第 7 章 中断处理

### 7.1 概述

中断是芯片的一个重要功能。它能将芯片从睡眠模式中唤醒，也可以使系统在正常运行过程中响应突发事件，中止并保存当前运行程序的信息，跳转到请求中断服务程序的入口地址，执行相对应的中断服务程序，处理突发事件。本芯片支持的中断模式有两种：默认中断模式和向量中断模式，最多可支持 12 个中断源：1 个软件中断和 11 个硬件中断。

### 7.2 内部结构

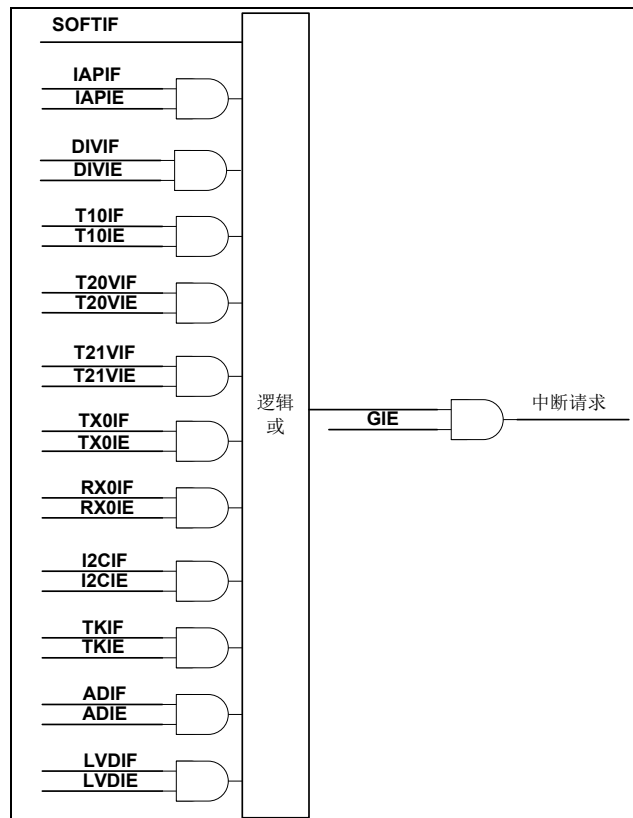


图 7-1 默认中断模式中中断控制逻辑

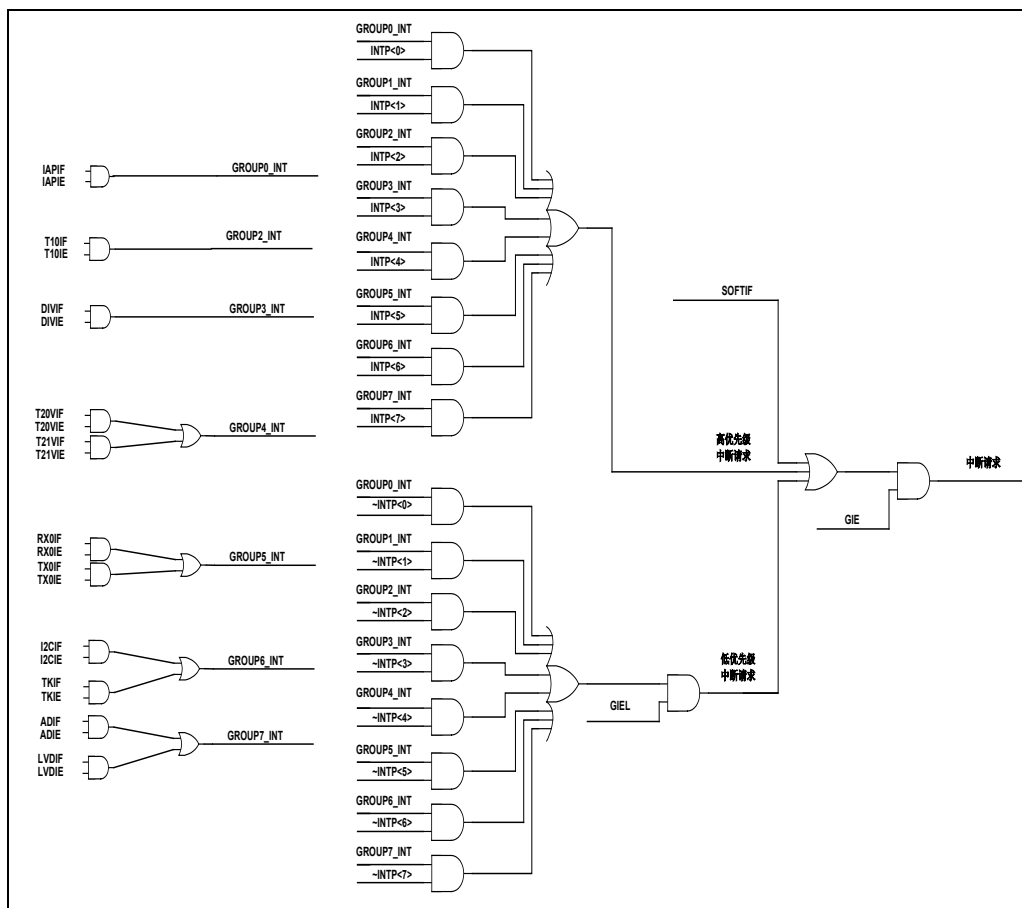


图 7-2 向量中断模式中中断控制逻辑

### 7.3 中断模式选择

芯片支持两种中断模式，默认中断模式和向量中断模式，可通过 INTG 寄存器中的 INTVEN0 位和芯片配置字中的 INTVEN1 位选择。值得注意的是，INTVEN0 和 INTVEN1 只有同时置“1”时，向量中断模式才有效。

INTVEN0 (INTG<2>)	INTVEN1 (芯片配置字 bit 13)	中断模式
0	0	默认中断模式
0	1	
1	0	向量中断模式
1	1	

表 7-1 中断模式选择表

默认中断模式只支持 1 个中断入口，即 0004H 入口地址，不支持中断优先级和中断嵌套。向量中断模式支持多个中断入口，且支持中断优先级和中断嵌套。

#### 7.3.1 默认中断模式

当配置为默认中断模式时，所有中断向量的入口地址均位于 0004H。用户需通过中断服务程序对各中断标志及中断使能位进行判断，确认引起中断操作的中断源，从而执行相应的中断服务子程序。该模式不支持中断优先级配置。

### 7.3.2 向量中断模式

#### 7.3.2.1 向量表配置

当配置为向量中断模式时，各中断源按组划分，每组中断对应一个中断向量入口地址。软中断入口地址为 0004<sub>H</sub>，优先级最高；其它硬件中断分 8 组(IG0~IG7)，通过中断全局寄存器 INTG 中的 INTV<1:0>配置支持不同的向量表优先级排序，并对应 8 个中断入口地址。每组硬件中断可以分别设置高低优先级，响应中断嵌套。通过中断优先级寄存器 INTP 中的 IGPx 配置，将所有硬件中断源分为高低两个优先级仲裁区。根据 INTV<1:0>的设置，对处于该仲裁区内的硬件中断组，进行优先级排序，并响应优先级最高的。高低两个优先级仲裁区分别由高优先级中断使能位 GIE 和低优先级中断使能位 GIEL 来使能。在执行低优先级中断服务程序时，可嵌套响应高优先级中断组。

优先级	0 (高)	1	2	3	4	5	6	7	8 (低)	
入口地址	0004 <sub>H</sub>	0008 <sub>H</sub>	000C <sub>H</sub>	0010 <sub>H</sub>	0014 <sub>H</sub>	0018 <sub>H</sub>	001C <sub>H</sub>	0020 <sub>H</sub>	0024 <sub>H</sub>	
INTV	00	软中断	IG0	IG1	IG2	IG3	IG4	IG5	IG6	IG7
	01		IG0	IG1	IG6	IG7	IG4	IG5	IG2	IG3
	10		IG4	IG5	IG2	IG3	IG0	IG1	IG6	IG7
	11		IG7	IG6	IG5	IG4	IG3	IG2	IG1	IG0

表 7-1 中断向量表

#### 7.3.2.2 中断向量分组

中断组号	优先级设置	中断名	对应中断寄存器位
IG0	IGP0	IAPINT	INTE2<1>
IG1	IGP1	保留	-
IG2	IGP2	T10INT	INTE0<2>
IG3	IGP3	DIVINT	INTE2<7>
IG4	IGP4	T20VINT	INTE0<0>
		T21VINT	INTE0<4>
IG5	IGP5	RX0INT	INTE2<5>
		TX0INT	INTE2<6>
IG6	IGP6	I2CINT	INTE2<2>
		TKINT	INTE2<0>
IG7	IGP7	ADINT	INTE2<3>
		LVDINT	INTE2<4>

表 7-2 向量中断模式中中断分组配置表

## 7.4 中断使能配置

序号	中断源	中断名	中断标志	中断使能	全局中断服务使能		
					默认模式	向量模式	
						高优先级	低优先级
1	软中断	软中断	SOFTIF	-	GIE	GIE	GIE & GIEL
2	T10 溢出中断	T10INT	T10IF	T10IE			
3	DIV 中断	DIVINT	DIVIF	DIVIE			
4	T20 溢出中断	T20VINT	T20VIF	T20VIE			
5	T21 溢出中断	T21VINT	T21VIF	T21VIE			
6	UART0 发送中断	TX0INT	TX0IF	TX0IE			
7	UART0 接收中断	RX0INT	RX0IF	RX0IE			
8	I2C 中断	I2CINT	I2CIF	I2CIE			
9	TK 触控扫描中断	TKINT	TKIF	TKIE			
10	ADC 中断	ADINT	ADIF	ADIE			
11	IAP 中断	IAPINT	IAPIF	IAPIE			
12	LVD 中断	LVDINT	LVDIF	LVDIE			

表 7-3 中断使能配置表

## 7.5 中断现场保护

中断现场保护是中断程序中一个很重要的组成部分。

指令系统中有 PUSH（压栈）和 POP（出栈）指令，可以方便的实现当前工作状态的保存和恢复。A、PSW、PCRH、BKSR 和 FAMR 寄存器，分别有各自的两级镜像寄存器 AS1、PSWS1、PCRHS1、BKSR1、FAMRS1 和 AS0、PSWS0、PCRHS0、BKSR0、FAMRS0，用于对相应寄存器的保存和恢复。镜像寄存器无物理地址，他们只能通过 PUSH 和 POP 指令自动完成相应的保存与恢复动作，两级镜像寄存器采用堆栈的操作方式。

## 7.6 中断操作

若中断事件条件产生，相关中断标志将被置为“1”。中断标志产生后程序要跳转至相应的服务程序地址执行，需满足以下两个条件：

- 1) 当对应中断使能位为“1”时，继续判断第二个条件是否满足；当对应中断使能位为“0”时，即使中断标志为“1”，中断也不会发生，程序也不会跳转至中断服务程序地址执行。
- 2) 在默认中断模式下，当全局中断使能位 GIE 为“0”时，将屏蔽所有中断请求；当全局中断使能位 GIE 为“1”时，程序将跳至中断服务程序地址执行。在向量中断模式下，当全局中断使能位 GIE 为“0”时，将屏蔽所有中断请求；当全局中断使能位 GIE 为“1”时，若对应中断所在组为高优先级，程序将跳至中断服务程序地址执行；若对应中断所在组为低优先级，当低优先级中断使能位 GIEL 为“1”时，在无高优先级中断请求时，程序将跳至中断服务程序地址执

行，当低优先级中断使能位 GIEL 为“0”时，将屏蔽所有低优先级中断请求。

### 7.6.1 T10 溢出中断

8 位定时器/计数器 T10 处于定时器模式或计数器模式，当 T10 计数器递增计数由 FFH 变为 00H 时，T10 计数器发生溢出，将中断标志 T10IF 位置“1”。当 T10 溢出中断使能位 T10IE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 T10 溢出中断请求。CPU 根据中断的优先级响应当前中断的请求，当 T10 溢出中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，T10 溢出中断标志位 T10IF 和中断使能位 T10IE 都需通过软件清除。

### 7.6.2 DIV 中断

硬件除法器 DIV 计算完成时，将中断标志 DIVIF 位置“1”。当 DIV 中断使能位 DIVIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 DIV 中断请求。CPU 根据中断的优先级响应当前中断的请求，当 DIV 中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，DIV 中断标志位 DIVIF 和中断使能位 DIVIE 都需通过软件清除。

### 7.6.3 T2n 溢出中断

16 位多功能定时器 T2n 处于定时器模式时，对计数时钟进行递增计数，当 T2n 后分频器的计数值与后分频器分频比相同时，产生溢出中断。

T2n 溢出中断产生时，将中断标志 T2nVIF 位置“1”。当 T2n 溢出中断使能位 T2nVIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 T2n 溢出中断请求。CPU 根据中断的优先级响应当前中断的请求，当 T2n 溢出中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，T2n 溢出中断标志位 T2nVIF 和中断使能位 T2nVIE 都需通过软件清除。

### 7.6.4 UART 发送/接收中断

UART 中断包括两种：发送中断和接收中断。当 UART 异步发送器的发送数据寄存器 TXnB 为空，或异步接收器完成一个数据接收时，产生 UART 发送/接收中断，发送/接收中断标志位 RXnIF/TXnIF 被置为“1”。如果发送/接收中断使能位 RXnIE/TXnIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 UART 发送/接收中断请求。CPU 根据中断的优先级响应当前中断的请求，当 UART 发送/接收中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，接收/发送中断标志位 RXnIF/TXnIF 为只读，不可软件清零，读接收数据寄存器 RXnB，可清除 RXnIF，写发送数据寄存器 TXnB，可清除 TXnIF；接收/发送中断使能位 RXnIE/TXnIE 需通过软件清除。

### 7.6.5 I2CS 中断

当 I2CSRIF、I2CSPIF、I2CTBIF、I2CRBIF、I2CTEIF、I2CROIF 和 I2CNAIF 中任何一个中断标志位置 1 时，I2C 总中断标志位 I2CIF 就会置 1。如果 I2C 中断使能位 I2CIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 I2C 中断请求。CPU 根据中断的优先级响应当前中断的请求，当 I2C 中断条

件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，需要软件清零 I2CIF，但在清零 I2CIF 总中断标志位之前，需先清零 I2CSRIF、I2CSPIF、I2CTEIF、I2CROIF 和 I2CNAIF 等相关中断标志位。

### 7.6.6 ADC中断

ADC 中断由 ADC 转换动作控制，当 ADC 转换完成时，将产生 ADC 中断，ADC 中断标志位 ADIF 被置“1”。当 ADC 中断使能位 ADIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 ADC 中断请求。CPU 根据中断的优先级响应当前中断的请求，当 ADC 中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，ADC 中断标志位 ADIF 和中断使能位 ADIE 都需通过软件清除。

### 7.6.7 IAP中断

IAP 有三种情况可以产生中断标志位，任何一种情况都可以产生中断请求：页擦启动出错、编程启动出错和 IAP 因电压低于 LVD 阈值电压而终止。

1. 页擦启动出错：当工作电压小于设定的 LVD 阈值电压（配置位 LVDIAPTEN=1），FRA 和 FRAN 寄存器值非反码，或向 IAPERSS 寄存器写入了非 0xAE 值时，页擦启动出错标志 ERSS\_ERR 位置“1”，IAP 总中断标志 IAPIF 也被置“1”。

2. 编程启动出错：当工作电压小于设定的 LVD 阈值电压（配置位 LVDIAPTEN=1），FRA 和 FRAN 寄存器值非反码，或 ROMD 和 ROMDN 寄存器的值非反，写入值非 0xD5 时，编程启动出错标志 PRGS\_ERR 位置“1”，IAP 总中断标志 IAPIF 也被置“1”。

3. IAP 因电压低于 LVD 阈值电压而终止：当 IAP 因电压低于 LVD 阈值电压而终止，标志位 IAP\_LVDIF 置“1”，IAP 总中断标志 IAPIF 也被置“1”。

如果 IAP 中断使能位 IAPIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 IAP 中断请求。CPU 根据中断的优先级响应当前中断的请求，当 IAP 中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，IAP 中断标志位 IAPIF 和中断使能位 IAPIE 都需通过软件清除。

### 7.6.8 LVD中断

当 VDD 电压小于 LVDCH 寄存器设置的阈值电压时，低电压产生，中断标志 LVDIF 位被置“1”。如果中断使能位 LVDIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出 LVD 中断请求。CPU 根据中断的优先级响应当前中断的请求，当 LVD 中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，LVD 中断标志位 LVDIF 和中断使能位 LVDIE 都需通过软件清除。

### 7.6.9 触摸按键中断

触摸按键有三种情况可以产生中断标志位，任何一种情况都可以产生中断请求：启动错误、计数溢出和触摸按键扫描结束。

1. 启动错误：按键启动扫描标志位 TKGO 置“1”，系统自动检测比较输出状态是否正常，如果异常会将扫描启动错误标志 TKERR 位置“1”，此时为按键控制器启动错误，触摸按键总中断标志 TKIF 位也被置“1”。
2. 计数溢出：扫描按键计数过程中，当计数寄存器溢出时，将扫描计数器溢出标志 SCANOV 位置“1”，触摸按键总中断标志 TKIF 位也被置“1”。
3. 触摸按键扫描结束：触摸按键扫描结束后，如未发现异常，触摸按键总中断标志 TKIF 位被置“1”。

如果触摸按键中断使能位 TKIE 置为“1”，且全局中断使能位 GIE 和低优先级中断使能位 GIEL 根据中断模式正确使能时，则向 CPU 发出触摸按键中断请求。CPU 根据中断的优先级响应当前中断的请求，当触摸按键中断条件允许时，系统将进入相应中断的服务程序入口地址，进行中断程序处理。需要注意的是，触摸按键中断标志位 TKIF 和中断使能位 TKIE 需通过软件清除。

### 7.7 中断操作注意事项

用户在使能中断前需先清除相应的中断标志，避免中断的误触发。

除只读的中断标志（由硬件清除）外，其余的中断标志必须通过软件清除。

为避免中断的发生与中断标志清除操作冲突时，清除中断标志不成功，建议用户在进行中断标志清除操作后，对中断标志清除成功与否进行软件判断。如果操作不成功则再次进行中断标志清除操作，直到中断标志清除成功为止。用户也可以连续执行两次中断标志清除操作达到相同目的。

### 7.8 特殊功能寄存器

中断功能由一系列的控制寄存器和芯片配置字共同控制。

INTG: 中断全局寄存器								
Bit	7	6	5	4	3	2	1	0
Name	GIE	GIEL	—	SGIE	SOFTIF	INTVEN0	INTV<1:0>	
R/W	R/W	R/W	—	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7      GIE: 全局中断使能位，或高优先级中断使能位  
0: 禁止所有的中断  
1: 使能所有未屏蔽的中断，或使能高优先级中断
- Bit 6      GIEL: 低优先级中断使能位（仅向量中断模式有效）  
0: 禁止低优先级中断  
1: 使能低优先级中断
- Bit 5      保留
- Bit 4      SGIE: 保留位，仅供测试使用，用户使用中需始终保持该位为 0
- Bit 3      SOFTIF: 软中断标志位  
0: 无软中断  
1: 有软中断
- Bit 2      INTVEN0: 中断模式选择位



- 0: 默认中断模式
  - 1: 向量中断模式 (芯片配置字 INTVEN1 必须为 1)
- Bit 1~0 INTV<1:0>: 中断向量表选择位, 参考向量中断配置表

INTP: 中断优先级寄存器								
Bit	7	6	5	4	3	2	1	0
Name	IGP<7:0>							
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7~0 IGP<7:0>: IG7-IG0 中断优先级设置
- 0: 低优先级
  - 1: 高优先级

INTC: 中断控制寄存器								
Bit	7	6	5	4	3	2	1	0
Name	—	—	KMSK1	KMSK0	—	—	—	—
R/W	—	—	R/W	R/W	—	—	—	—
POR	0	0	0	0	0	0	0	0

- Bit 7~6 保留
- Bit 5 KMSK1: KIN1 按键输入屏蔽位
- 0: 屏蔽
  - 1: 不屏蔽
- Bit 4 KMSK0: KIN0 按键输入屏蔽位
- 0: 屏蔽
  - 1: 不屏蔽
- Bit 3~0 保留

INTE0: 中断使能寄存器 0								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	T21VIE	—	T10IE	—	T20VIE
R/W	—	—	—	R/W	—	R/W	—	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7~5 保留
- Bit4 T21VIE: T21 溢出中断使能位
- 0: 禁止
  - 1: 使能
- Bit3 保留
- Bit2 T10IE: T10 溢出中断使能位
- 0: 禁止
  - 1: 使能
- Bit1 保留
- Bit 0 T20VIE: T20 溢出中断使能位
- 0: 禁止



1: 使能

INTF0: 中断标志寄存器 0								
Bit	7	6	5	4	3	2	1	0
Name	—	—	—	T21VIF	—	T10IF	—	T20VIF
R/W	—	—	—	R/W	—	R/W	—	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7~5      保留
- Bit 4      T21VIF: T21 溢出中断标志位  
0: 无中断请求  
1: T21 溢出中断请求
- Bit 3      保留
- Bit 2      T10IF: T10 溢出中断标志位  
0: 无中断请求  
1: T10 溢出中断请求
- Bit 1      保留
- Bit 0      T20VIF: T20 溢出中断标志位  
0: 无中断请求  
1: T20 溢出中断请求

INTE2: 中断使能寄存器 2								
Bit	7	6	5	4	3	2	1	0
Name	DIVIE	TX0IE	RX0IE	LVDIE	ADIE	I2CIE	IAPIE	TKIE
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

- Bit 7      DIVIE: DIV 中断使能位  
0: 禁止  
1: 使能
- Bit 6      TX0IE: UART0 发送中断使能位  
0: 禁止  
1: 使能
- Bit 5      RX0IE: UART0 接收中断使能位  
0: 禁止  
1: 使能
- Bit 4      LVDIE: LVD 中断使能位  
0: 禁止  
1: 使能
- Bit 3      ADIE: AD 转换中断使能位  
0: 禁止  
1: 使能
- Bit 2      I2CIE: I2C 中断使能位
- Bit 1      IAPIE: IAP 中断使能位  
0: 禁止  
1: 使能

Bit 0           TKIE: 触控中断使能位  
                  0: 禁止  
                  1: 使能

INTF2: 中断标志寄存器 2								
Bit	7	6	5	4	3	2	1	0
Name	DIVIF	TX0IF	RX0IF	LVDIF	ADIF	I2CIF	IAPIF	TKIF
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W
POR	0	0	0	0	0	0	0	0

Bit 7           DIVIF: DIV 中断标志位  
                  0: 无中断请求  
                  1: DIV 中断请求

Bit 6           TX0IF: UART0 发送中断标志位  
                  0: 无中断请求  
                  1: UART0 发送中断请求

Bit 5           RX0IF: UART0 接收中断标志位  
                  0: 无中断请求  
                  1: UART0 接收中断请求

Bit 4           LVDIF: LVD 中断标志位  
                  0: 无中断请求  
                  1: LVD 中断请求

Bit 3           ADIF: AD 转换中断标志位  
                  0: 无中断请求  
                  1: AD 中断请求

Bit 2           I2CIF: I2C 中断标志位  
                  0: 无中断请求  
                  1: I2C 中断请求

Bit 1           IAPIF: IAP 中断标志位  
                  0: 无中断请求  
                  1: IAP 中断请求

Bit 0           TKIF: 触控中断标志位  
                  0: 无中断请求  
                  1: TK 中断请求

## 第 8 章 芯片配置字

芯片的各种功能配置由芯片配置字和各功能相关寄存器共同设置完成。

名称	芯片配置字 (CFG_WORD0)	
—	bit1-0	保留
—	bit2	固定 0
<b>WDTEN</b>	bit3	<b>硬件看门狗使能位</b> 0: 禁止 1: 使能
—	bit4	固定 0
<b>ICDEB</b>	bit5	<b>调试模式使能位</b> 0: 使能 1: 禁止
<b>ICDSEL</b>	bit6	<b>调试端口选择位</b> 0: PB5 用作 ISDA, PB4 用作 ISCK 1: PA0 用作 ISDA, PA1 用作 ISCK
<b>BOREN</b>	bit7	<b>掉电复位使能位</b> 0: 禁止 1: 使能
<b>BORVS</b>	bit9-8	<b>BOR 电压选择位</b> 00: 3.7V    01: 3.1V    10: 2.5V    11: 2.3V
<b>IAPEN</b>	bit10	<b>IAP 操作使能位</b> 0: 使能 1: 禁止
<b>INTVEN1</b>	bit11	<b>中断模式选择位</b> 0: 默认中断模式 1: 向量中断模式 (控制寄存器位 INTVEN0 也必须为 1)
—	bit12	保留
<b>STKLS</b>	bit15-13	<b>PC 硬件堆栈级数配置位</b> 000: 4 级    001: 8 级    010: 12 级    011: 16 级 100: 20 级    101: 24 级    110: 28 级    111: 32 级

名称	芯片配置字 (CFG_WORD1)	
<b>PWRDWN</b>	bit0	<b>Flash 节电控制位</b> 0: 禁止在 idle 模式关闭 flash 电源 1: idle 模式关闭 flash 电源 (缺省值)
<b>DPAGES</b>	bit1	<b>数据 FLASH 存储区选择位</b> 0: 512 Words 1: 禁止
—	bit2	保留
<b>VTKVSEL</b>	bit3	<b>Touch key 充电电压选择位选择位</b> 0: VDD5V 电源供电 1: 内部基准供电 2.6v(default)

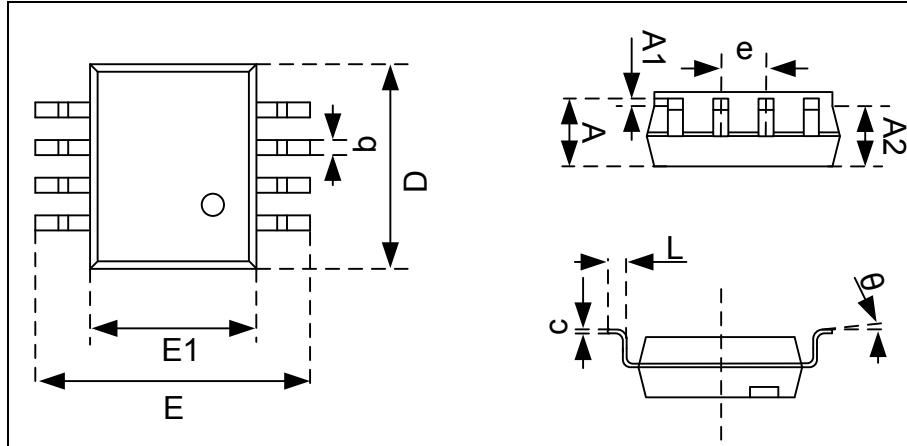
名称	芯片配置字 (CFG_WORD1)	
<b>BORFLTSEL</b>	bit6-4	<b>BOR 滤波时间长度选择位</b> (32KHz LRC 时钟个数) 3'b000: 7 (218.75us) 3'b001: 1 (31.25us) 3'b010: 2 (62.5us) ... 3'b110: 6 (187.5us) 3'b111: 7 (218.75us) (缺省值)
<b>LVDIAPTEN</b>	bit7	<b>LVD 终止 IAP 编程或擦除使能位</b> (enable of IAP termination by LVD) 0: 禁止 LVD 终止 IAP 编程或页擦 1: 使能 LVD 终止 IAP 编程或页擦 (缺省值)
—	Bit15-8	保留

名称	芯片配置字 (CFG_WORD2)	
—	Bit0	保留
<b>DFLT_CLK</b>	Bit1	<b>唤醒后时钟选择位</b> 0: 进入睡眠前系统时钟 1: LRC (32KHz) 时钟
<b>VR_LP2HP</b>	Bit2	<b>唤醒时 VR 从 LP 模式切换到 HP 模式时 flash 上电延时设定</b> 0: LP 切换至 HP 时, flash 电源打开等待时间为 3 个 WDT 时钟 1: LP 切换至 HP 时, flash 电源打开等待时间为 15 个 WDT 时钟
—	Bit15-3	保留

## 第 9 章 芯片封装图

### 9.1 8-pin 封装图

SOP8



标号	公制 (mm)			英制 (inch)		
	MIN	NOM	MAX	MIN	NOM	MAX
A	—	—	1.75	—	—	0.07
A1	0.10	—	0.225	0.004	—	0.009
A2	1.30	1.40	1.50	0.05	0.06	0.06
D	4.70	4.90	5.10	0.19	0.19	0.20
E	5.80	6.00	6.20	0.23	0.24	0.24
E1	3.70	3.90	4.10	0.15	0.15	0.16
b	0.39	—	0.48	0.02	—	0.02
c	0.21	—	0.26	0.01	—	0.01
e	—	1.27	—	—	0.05	—
L	0.50	—	0.80	0.02	—	0.03
$\theta$	0°	—	8°	0°	—	8°

## 附录1 指令集

### 附录1.1 概述

本芯片提供了 79 条精简指令。

汇编指令为了方便程序设计者使用，指令名称大多是由指令功能的英文缩写所组成的。这些指令所组成的程序经过编译器的编译与链接后，会被转换为相对应的指令码。转换后的指令码可以分为操作码（OP Code）与操作数（Operand）两个部分，操作码部分对应到指令本身。

芯片运行在 4MHz 主系统时钟时，一个机器周期的时间为 500ns。

按照指令执行的机器周期数可将指令分为双周期指令和单周期指令，其中 CALL、LCALL、RCALL、GOTO、JUMP、RET、RETIA、RETIE 为双周期指令；满足跳转条件时，JBC、JBS、JDEC、JINC 指令为双周期指令，否则为单周期指令；其它指令为单周期指令。

### 附录1.2 寄存器操作指令

序号	指令		影响 状态位	机器周期	操作
1	SECTION	I<7:0>	—	1	I<7:0>->BKSR<7:0>
2	PAGE	I<8:0>	—	1	I<4:0>->PCRH<7:3>
3	ISTEP	I<7:0>	—	1	IAA+i->IAA(-128≤i≤127)
4	MOVI	I<7:0>	—	1	I<7:0>->(A)
5	MOV	R<7:0>,F	Z,N	1	(R)->(目标)
6	MOVA	R<7:0>	—	1	(A)->(R)
7	MOVAR	R<10:0>	—	1	(A)->(R<10:0>)
8	MOVRA	R<10:0>	—	1	(R<10:0>)->(A)

附录表 9-1 寄存器操作指令表

### 附录1.3 程序控制指令

序号	指令		影响 状态位	机器周期	操作
9	JUMP	I<7:0>	—	2	PC+1+i<7:0>->PC (-128≤i≤127)
10	AJMP	I<19:0>	—	2	I<13:0>->PC<13:0> I<13:8>->PCRH<5:0>
11	GOTO	I<10:0>	—	2	I<10:0>->PC<10:0>, PCRH<5:3>->PC<13:11>
12	CALL	I<10:0>	—	2	PC+1->TOS,I<10:0>->PC<10:0>, PCRH<5:3>->PC<13:11>
13	LCALL	I<19:0>	—	2	PC+1->TOS,I<13:0>->PC<13:0>

序号	指令		影响状态位	机器周期	操作
					I<13:8>->PCRH<5:0>
14	RCALL	R<7:0>	—	2	PC+1->TOS, (R)->PC<7:0>, PCRH<5:0>->PC<13:8>
15	JBC	R<7:0>, B<2:0>	—	2	当 R<B> = 0 时跳过下一条指令
16	JBS	R<7:0>, B<2:0>	—	2	当 R<B> = 1 时跳过下一条指令
17	JCAIE	I<7:0>	—	2	当(A) = I 时跳过下一条指令
18	JCAIG	I<7:0>	—	2	当(A) > I 时跳过下一条指令
19	JCAIL	I<7:0>	—	2	当(A) < I 时跳过下一条指令
20	JCRAE	R<7:0>	—	2	当(R) = (A)时跳过下一条指令
21	JCRAG	R<7:0>	—	2	当(R) > (A)时跳过下一条指令
22	JCRAL	R<7:0>	—	2	当(R) < (A)时跳过下一条指令
23	JCCRE	R<7:0>, B<2:0>	—	2	当 C = R(B)时跳过下一条指令
24	JCCRG	R<7:0>, B<2:0>	—	2	当 C > R(B)时跳过下一条指令
25	JCCRL	R<7:0>, B<2:0>	—	2	当 C < R(B)时跳过下一条指令
26	JDEC	R<7:0>, F	—	2	(R-1)->(目标寄存器), 当目标寄存器的值为 0 时则跳过下一条指令
27	JINC	R<7:0>, F	—	2	(R+1)->(目标寄存器), 当目标寄存器的值为 0 时则跳过下一条指令
28	NOP	—	—	1	空操作
29	POP	—	—	1	AS->A, PSWS->PSW, BKSr->BKSRS, PCRHS->PCRH
30	PUSH	—	—	1	A->AS, PSW->PSWS, BKSr->BKSRS, PCRH->PCRHS
31	RET	—	—	2	TOS->PC
32	RETIA	I<7:0>	—	2	I->(A), TOS->PC
33	RETIE	—	—	2	TOS->PC, 1->GIE
34	RST	—	全部状态位均被影响	1	软件复位指令
35	CWDT	—	N_TO, N_PD	1	00H->WDT, 0->WDTPrescaler, 1-> N_TO, 1-> N_PD
36	IDLE	—	N_TO, N_PD	1	00H->WDT, 0->WDTPrescaler, 1-> N_TO, 0-> N_PD

附录表 9-2 程序控制指令表

附录1.4 算术/逻辑运算指令

序号	指令		影响状态位	机器周期	操作
37	ADD	R<7:0>,F	C,DC,Z,OV,N	1	(R)+(A)->(目标)
38	ADDC	R<7:0>,F	C,DC,Z,OV,N	1	(R)+(A)+C->(目标)
39	ADDCI	I<7:0>	C,DC,Z,OV,N	1	I+(A)+C->(A)
40	ADDI	I<7:0>	C,DC,Z,OV,N	1	I+(A)->(A)
41	AND	R<7:0>,F	Z,N	1	(A).AND.(R)->(目标)
42	ANDI	I<7:0>	Z,N	1	I.AND.(A)->(A)
43	BCC	R<7:0>,B<2:0>	—	1	0->R<B>
44	BSS	R<7:0>,B<2:0>	—	1	1->R<B>
45	BTT	R<7:0>,B<2:0>	—	1	(~R<B>)->R<B>
46	CLR	R<7:0>	Z	1	(R)=0
47	SETR	R<7:0>	—	1	FF <sub>H</sub> ->(R)
48	NEG	R<7:0>	C,DC,Z,OV,N	1	~(R)+1->(R)
49	COM	R<7:0>,F	Z,N	1	(~R)->(目标)
50	DAR	R<7:0>,F	C	1	对(R)十进制调整->(目标)
51	DAA	—	C	1	对(A)十进制调整->(A)
52	DEC	R<7:0>,F	C,DC,Z,OV,N	1	(R-1)->(目标)
53	INC	R<7:0>,F	C,DC,Z,OV,N	1	(R+1)->(目标)
54	IOR	R<7:0>,F	Z,N	1	(A).OR.(R)->(目标)
55	IORI	I<7:0>	Z,N	1	I.OR.(A)->(A)
56	RLB	R<7:0>,F,B<2:0>	C,Z,N	1	C<< R<7:0> <<C
57	RLBNC	R<7:0>,F,B<2:0>	Z,N	1	R<7:0> << R<7>
58	RRB	R<7:0>,F,B<2:0>	C,Z,N	1	C>> R<7:0> >>C
59	RRBNC	R<7:0>,F,B<2:0>	Z,N	1	R<0> >> R<7:0>
60	SUB	R<7:0>,F	C,DC,Z,OV,N	1	(R)-(A)->(目标)
61	SUBC	R<7:0>,F	C,DC,Z,OV,N	1	(R)-(A)- (~C)->(目标)
62	SUBCI	I<7:0>	C, DC, Z,OV,N	1	I-(A)- (~C)->(A)
63	SUBI	I<7:0>	C, DC, Z,OV,N	1	I-(A)->(A)
64	SSUB	R<7:0>,F	C, DC, Z,OV,N	1	(A)-(R)->(目标)
65	SSUBC	R<7:0>,F	C, DC, Z,OV,N	1	(A)-(R)- (~C)->(目标)
66	SSUBCI	I<7:0>	C, DC, Z,OV,N	1	(A)-I- (~C)->(A)
67	SSUBI	I<7:0>	C, DC, Z,OV,N	1	(A)-I->(A)



序号	指令		影响 状态位	机器 周期	操作
68	SWAP	R<7:0>,F	—	1	R<3:0>->(目标)<7:4>, R<7:4>->(目标)<3:0>
69	TBR	—	—	2	Pmem(FRA)->ROMD
70	TBR#1	—	—	2	Pmem(FRA)-> ROMD, FRA+1->FRA
71	TBR_1	—	—	2	Pmem(FRA)-> ROMD, FRA-1->FRA
72	TBR1#	—	—	2	FRA+1->FRA, Pmem(FRA)-> ROMD
73	TBW	—	—	2	ROMD->prog buffer
74	TBW#1	—	—	2	ROMD>prog buffer, FRA+1->FRA
75	TBW_1	—	—	2	ROMD->prog buffer, FRA-1->FRA
76	TBW1#	—	—	2	FRA+1->FRA, ROMD->prog buffer
77	XOR	R<7:0>, F	Z,N	1	(A).XOR.(R)->(目标)
78	XORI	I<7:0>	Z,N	1	I.XOR.(A)->(A)

附录表 9-3 算术/逻辑运算指令表

注：指令集说明

- 1: i—立即数， F—标志位， A—寄存器 A， R—寄存器 R， B—寄存器 R 的第 B 位。
- 2: C—进位/借位， DC—半进位/半借位， Z—零标志位， OV—溢出标志位， N—负标志位。
- 3: TOS—顶级堆栈。
- 4: 如果 F = 0，则目标寄存器为寄存器 A；如果 F = 1，则目标寄存器为寄存器 R。
- 5: 79 条指令中另有一条 NOP 指令未在上表中描述。
- 6: SECTION 指令中， N 的位数，视实际芯片而定。对本芯片，通用数据存储器 GPR 分为 12 个存储体组，所以 N 的位数是 4 位。
- 7: PAGE 指令中， N 的位数，视实际芯片而定。对本芯片，没有 PCRU 寄存器，N 的位数是 3 位。
- 8: PC 的位数以及 PCRU 寄存器，视实际芯片而定。对本芯片，PC 的位数是 14 位，没有 PCRU 寄存器。

## 附录2 特殊功能寄存器总表

### 快速访问区特殊功能寄存器

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电 复位值
FF80 <sub>H</sub>	IAD	IAD<7:0>								0000 0000
FF81 <sub>H</sub>	IAAL	IAA<7:0>								0000 0000
FF82 <sub>H</sub>	IAAH	IAA<15:8>								0000 0000
FF83 <sub>H</sub>	BKSR	—	—	—	—	—	—	BKSR<1:0>		0000 0000
FF84 <sub>H</sub>	FAMR	—	—	—	—	—	—	FAMS	FMHL	0000 0000
FF85 <sub>H</sub>	AREG	AREG<7:0>								xxxx xxxx
FF86 <sub>H</sub>	PSW	—	UF	OF	N	OV	Z	DC	C	000x xxxx
FF87 <sub>H</sub>	PWRCWP	PWRCWP<7:0>								0000 0000
FF88 <sub>H</sub>	PWRC	—	SMRSTEN	—	IRSTB	TOB	PDB	PORB	BORB	1101 110x
FF89 <sub>H</sub>	PCRL	PCR<7:0>								0000 0000
FF8A <sub>H</sub>	PCRH	PCR<15:8>								0000 0000
FF8B <sub>H</sub>	—	—								0000 0000
FF8C <sub>H</sub>	—	—								0000 0000
FF8D <sub>H</sub>	—	—								0000 0000
FF8E <sub>H</sub>	—	—								0000 0000
FF8F <sub>H</sub>	—	—								0000 0000
FF90 <sub>H</sub>	—	—								0000 0000
FF91 <sub>H</sub>	—	—								0000 0000
FF92 <sub>H</sub>	—	—								0000 0000
FF93 <sub>H</sub>	—	—								0000 0000
FF94 <sub>H</sub>	INTG	GIE	GIEL	—	SGIE	SOFTIF	INTVEN0	INTV<1:0>		0000 0000
FF95 <sub>H</sub>	INTP	IGP<7:0>								0000 0000

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电 复位值
FF96 <sub>H</sub>	INTE0	—	—	T21PIE	T21VIE	—	T10IE	T20PIE	T20VIE	0000 0000
FF97 <sub>H</sub>	INTF0	—	—	T21PIF	T21VIF	—	T10IF	T20PIF	T20VIF	0000 0000
FF98 <sub>H</sub>	INTE1	KIE	—	—	—	—	PIE<2:0>			0000 0000
FF99 <sub>H</sub>	INTF1	KIF	—	—	—	—	PIF<2:0>			0000 0000
FF9A <sub>H</sub>	INTE2	DIVIE	TX0IE	RX0IE	LVDIE	ADIE	I2CIE	IAPIE	TKIE	0000 0000
FF9B <sub>H</sub>	INTF2	DIVIF	TX0IF	RX0IF	LVDIF	ADIF	I2CIF	IAPIF	TKIF	0000 0000
FF9C <sub>H</sub>	INTC	—	—	KMSK1	KMSK0	—	PEG0<2:0>			0000 0000
FF9D <sub>H</sub>	PA	PA<7:0>								xxxx xxxx
FF9E <sub>H</sub>	PAT	PAT<7:0>								1111 1111
FF9F <sub>H</sub>	PAS	PAS<7:0>								1111 1100
FFA0 <sub>H</sub>	PAPU	PAPU<7:0>								0000 0000
FFA1 <sub>H</sub>	PAPD	PAPD<7:0>								0000 0000
FFA2 <sub>H</sub>	PALC	PALC<7:0>								0000 0000
FFA3 <sub>H</sub>	PAOD	PAOD <7:0>								0000 0000
FFA4 <sub>H</sub>	PB	PB<7:0>								xxxx xxxx
FFA5 <sub>H</sub>	PBT	PBT<7:0>								1111 1111
FFA6 <sub>H</sub>	PBS	PBS<7:0>								1100 1111
FFA7 <sub>H</sub>	PBPU	PBPU<7:0>								0000 0000
FFA8 <sub>H</sub>	PBPD	PBPD<7:0>								0000 0000
FFA9 <sub>H</sub>	—	—								—
FFAA <sub>H</sub>	PBLC	PBLC<7:0>								0000 0000
FFAB <sub>H</sub>	PBOD	PBOD<7:0>								0000 0000
FFAC <sub>H</sub>	PBSMIT	—	—	—	—	PBSMIT<3:0>				0000 0000
FFAD <sub>H</sub>	PC	—	—	—	—	—	—	PC<1:0>		0000 00xx
FFAE <sub>H</sub>	PCT	—	—	—	—	—	—	PCT<1:0>		0000 0011

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电 复位值	
FFAF <sub>H</sub>	PCS	—	—	—	—	—	—	PCS<1:0>		0000 0001	
FFB0 <sub>H</sub>	PCPU	—	—	—	—	—	—	PCPU<1:0>		0000 0000	
FFB1 <sub>H</sub>	PCPD	—	—	—	—	—	—	PCPD<1:0>		0000 0000	
FFB2 <sub>H</sub>	PCLC	—	—	—	—	—	—	PCLC<1:0>		0000 0000	
FFB3 <sub>H</sub>	PCOD	—	—	—	—	—	—	PCOD<1:0>		0000 0000	
FFB4 <sub>H</sub>	PE	PE<7:0>									xxxx xxxx
FFB5 <sub>H</sub>	PET	PET<7:0>									1111 1111
FFB6 <sub>H</sub>	PES	PES<7:0>									1111 1111
FFB7 <sub>H</sub>	PEPU	PEPU<7:0>									0000 0000
FFB8 <sub>H</sub>	PEPD	PEPD<7:0>									0000 0000
FFB9 <sub>H</sub>	—	—									—
FFBA <sub>H</sub>	PELC	PELC<7:0>									0000 0000
FFBB <sub>H</sub>	PEOD	PEOD<7:0>									0000 0000
FFBC <sub>H</sub>	PESMIT	PESMIT<7:6>	—	—	—	—	—	—	—	0000 0000	
FFBD <sub>H</sub>	SCC	SCC<7:0>									0000 0000
FFBE <sub>H</sub>	SCSEL0	SCSEL0<7:0>									0000 0000
FFBF <sub>H</sub>	SCSEL1	SCSEL1<7:0>									0000 0000

分页访问区 Section 0 特殊功能寄存器

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电 复位值
6000 <sub>H</sub>	FRAL	FRAL<7:0>								0000 0000
6001 <sub>H</sub>	FRALN	FRALN<7:0>								0000 0000
6002 <sub>H</sub>	FRAH	FRAH<7:0>								0000 0000
6003 <sub>H</sub>	FRAHN	FRAHN<7:0>								0000 0000
6004 <sub>H</sub>	ROMDL	ROMDL<7:0>								xxxx xxxx
6005 <sub>H</sub>	ROMDLN	ROMDLN<7:0>								xxxx xxxx
6006 <sub>H</sub>	ROMDH	ROMDH<7:0>								xxxx xxxx
6007 <sub>H</sub>	ROMDHN	ROMDHN<7:0>								xxxx xxxx
6008 <sub>H</sub>	IAPUL	IAPUL<7:0>								0000 0000
6009 <sub>H</sub>	IAPC	ERSTWT<3:0>				PRGTWT<1:0>		IAPCLR	TOEN	0110 0000
600A <sub>H</sub>	IAPERSS	IAPERSS<7:0>								0000 0000
600B <sub>H</sub>	IAPPRGS	IAPPRGS<7:0>								0000 0000
600C <sub>H</sub>	IAPS	FRAH_ERR	FRAL_ERR	ROMD_ERR	ERSS_ERR	PRGS_ERR	IAP_TIMEOUT	IAP_DONE	IAP_LVDIF	1110 0000
600D <sub>H</sub>	T10	T10<7:0>								0000 0000
600E <sub>H</sub>	T10C	T10EN	—	T10M	T10EG	T10PRE	T10PRS<2:0>			0000 0000
600F <sub>H</sub>	T20L	T20L<7:0>								0000 0000
6010 <sub>H</sub>	T20H	T20H<7:0>								0000 0000
6011 <sub>H</sub>	T20PL	T20PL<7:0>								1111 1111
6012 <sub>H</sub>	T20PH	T20PH<7:0>								1111 1111
6013 <sub>H</sub>	T20R0L	T20R0L<7:0>								0000 0000
6014 <sub>H</sub>	T20R0H	T20R0H<7:0>								0000 0000
6015 <sub>H</sub>	T20R1L	T20R1L<7:0>								0000 0000
6016 <sub>H</sub>	T20R1H	T20R1H<7:0>								0000 0000
6017 <sub>H</sub>	T20CL	T20M<3:0>				—	—	—	T20TR	0000 0000

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电 复位值	
6018 <sub>H</sub>	T20CH	T20EN	T20POS<6:0>							0000 0000	
6019 <sub>H</sub>	T20CM	—	—	T20OM<1:0>		T20PRS<3:0>				0000 0000	
601A <sub>H</sub>	T21L	T21L<7:0>									0000 0000
601B <sub>H</sub>	T21H	T21H<7:0>									0000 0000
601C <sub>H</sub>	T21PL	T21PL<7:0>									1111 1111
601D <sub>H</sub>	T21PH	T21PH<7:0>									1111 1111
601E <sub>H</sub>	T21R0L	T21R0L<7:0>									0000 0000
601F <sub>H</sub>	T21R0H	T21R0H<7:0>									0000 0000
6020 <sub>H</sub>	T21R1L	T21R1L<7:0>									0000 0000
6021 <sub>H</sub>	T21R1H	T21R1H<7:0>									0000 0000
6022 <sub>H</sub>	T21CL	T21M<3:0>				—	—	—	T21TR	0000 0000	
6023 <sub>H</sub>	T21CH	T21EN	T21POS<6:0>								0000 0000
6024 <sub>H</sub>	T21CM	—	—	T21OM<1:0>		T21PRS<3:0>				0000 0000	
6025 <sub>H</sub>	T2nOC	—	—	—	—	P211EN	P210EN	P201EN	P200EN	0000 0000	
6026 <sub>H</sub>	—	—									-
6027 <sub>H</sub>	—	—									-
6028 <sub>H</sub>	MULA	MULA<7:0>									XXXX XXXX
6029 <sub>H</sub>	MULB	MULB<7:0>									XXXX XXXX
602A <sub>H</sub>	MULL	MULL<7:0>									XXXX XXXX
602B <sub>H</sub>	MULH	MULH<7:0>									XXXX XXXX
602C <sub>H</sub>	DIVEL	DIVE<7:0>									XXXX XXXX
602D <sub>H</sub>	DIVEM	DIVE<15:8>									XXXX XXXX
602E <sub>H</sub>	DIVEH	DIVE<23:16>									XXXX XXXX
602F <sub>H</sub>	DIVEU	DIVE<31:24>									XXXX XXXX
6030 <sub>H</sub>	DIVSL	DIVSL<7:0>									XXXX XXXX

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电 复位值
6031 <sub>H</sub>	DIVSH	DIVSH<7:0>								XXXX XXXX
6032 <sub>H</sub>	DIVQL	DIVQ<7:0>								XXXX XXXX
6033 <sub>H</sub>	DIVQM	DIVQ<15:8>								XXXX XXXX
6034 <sub>H</sub>	DIVQH	DIVQ<23:16>								XXXX XXXX
6035 <sub>H</sub>	DIVQU	DIVQ<31:24>								XXXX XXXX
6036 <sub>H</sub>	DIVRL	DIVRL<7:0>								XXXX XXXX
6037 <sub>H</sub>	DIVRH	DIVRH<7:0>								XXXX XXXX
6038 <sub>H</sub>	DIVC	—	—	—	SIGNED	—	—	VERR	VGO	0000 0000
6039 <sub>H</sub> ~ 603F <sub>H</sub>	—	—								

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电复位值
6040 <sub>H</sub>	TKSELL	TKCHSL<7:0>								0000 0000
6041 <sub>H</sub>	TKSELH	—	—	—	—	TKCHSH<3:0>				0000 0000
6042 <sub>H</sub>	TKTUN	TKCHGS	TKDIST<2:0>			TKCFT<1:0>		TKDCE	TKDCS	0000 0000
6043 <sub>H</sub>	TKCTL0	—	SCANOV	TKERR	TKOV	TKCCE	TKCTR	TKEN	TKGO	0000 0000
6044 <sub>H</sub>	TKCTL1	TKMS	TKDS	TKAVE	—	TKTMS<1:0>		—	TKPE	0000 0000
6045 <sub>H</sub>	TKFCTL	TKDUS	TKFS<2:0>			—	TKJFS<1:0>		TKJFE	0000 0000
6046 <sub>H</sub>	VRC	—	—	—	—	VRCS<2:0>			HYSEN	0000 0001
6047 <sub>H</sub>	TKMODL	TKMODL<7:0>								1111 1111
6048 <sub>H</sub>	TKMODM	TKMODM<7:0>								1111 1111
6049 <sub>H</sub>	TKMODH	TKMODH<7:0>								1111 1111
604A <sub>H</sub>	TKMODU	TKMODU<7:0>								0000 1111
604B <sub>H</sub>	TKSDAL	TKSDAL<7:0>								0000 0000
604C <sub>H</sub>	TKSDAH	TKSDAH<7:0>								0000 0000
604D <sub>H</sub>	—	—								—
604E <sub>H</sub>	—	—								—
604F <sub>H</sub>	TK0DAL	TK0DAL<7:0>								0000 0000
6050 <sub>H</sub>	TK0DAH	TK0DAH<7:0>								0000 0000
6051 <sub>H</sub>	TK1DAL	TK1DAL<7:0>								0000 0000
6052 <sub>H</sub>	TK1DAH	TK1DAH<7:0>								0000 0000
6053 <sub>H</sub>	TK2DAL	TK2DAL<7:0>								0000 0000
6054 <sub>H</sub>	TK2DAH	TK2DAH<7:0>								0000 0000
6055 <sub>H</sub>	TK3DAL	TK3DAL<7:0>								0000 0000
6056 <sub>H</sub>	TK3DAH	TK3DAH<7:0>								0000 0000
6057 <sub>H</sub>	TK4DAL	TK4DAL<7:0>								0000 0000
6058 <sub>H</sub>	TK4DAH	TK4DAH<7:0>								0000 0000



地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电复位值
6059 <sub>H</sub>	TK5DAL	TK5DAL<7:0>								0000 0000
605A <sub>H</sub>	TK5DAH	TK5DAH<7:0>								0000 0000
605B <sub>H</sub>	TK6DAL	TK6DAL<7:0>								0000 0000
605C <sub>H</sub>	TK6DAH	TK6DAH<7:0>								0000 0000
605D <sub>H</sub>	TK7DAL	TK7DAL<7:0>								0000 0000
605E <sub>H</sub>	TK7DAH	TK7DAH<7:0>								0000 0000
605F <sub>H</sub>	TK8DAL	TK8DAL<7:0>								0000 0000
6060 <sub>H</sub>	TK8DAH	TK8DAH<7:0>								0000 0000
6061 <sub>H</sub>	TK9DAL	TK9DAL<7:0>								0000 0000
6062 <sub>H</sub>	TK9DAH	TK9DAH<7:0>								0000 0000
6063 <sub>H</sub>	TK10DAL	TK10DAL<7:0>								0000 0000
6064 <sub>H</sub>	TK10DAH	TK10DAH<7:0>								0000 0000
6065 <sub>H</sub>	TK11DAL	TK11DAL<7:0>								0000 0000
6066 <sub>H</sub>	TK11DAH	TK11DAH<7:0>								0000 0000
6067 <sub>H</sub>	—	—								—
6068 <sub>H</sub>	BR0R	BR0R<7:0>								0000 0000
6069 <sub>H</sub>	RX0B	RX0B<7:0>								xxxx xxxx
606A <sub>H</sub>	RX0C	RX0EN	RX0LEN	—	—	—	OERR0	FERR0	RX0R8	0000 0000
606B <sub>H</sub>	TX0B	TX0B<7:0>								0000 0000
606C <sub>H</sub>	TX0C	TX0EN	TX0LEN	BRGH0	—	—	—	TRMT0	TX0R8	0000 0010
606D <sub>H</sub>	I2CX16	—	—	—	I2CX16<4:0>				—	0000 0000
606E <sub>H</sub>	I2CC	I2CTE	I2COD<1:0>		I2CTAS	I2CANAE	I2CCSE	I2CRST	I2CEN	0000 0000
606F <sub>H</sub>	I2CSA	I2CSADR<6:0>							I2CRW	0000 0000
6070 <sub>H</sub>	I2CTB	I2CTB<7:0>								0000 0000
6071 <sub>H</sub>	I2CRB	I2CRB<7:0>								0000 0000

地址	名称	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0	上电复位值
6072 <sub>H</sub>	I2CIEC	—	I2CNAIE	I2CROIE	I2CTEIE	I2CRBIE	I2CTBIE	I2CSPIE	I2CSRIE	0000 0000
6073 <sub>H</sub>	I2CIFC	—	I2CNAIF	I2CROIF	I2CTEIF	I2CRBIF	I2CTBIF	I2CSPIF	I2CSRIF	0000 0100
6074 <sub>H</sub>	ADCRL	ADCRL<7:0>								0000 0000
6075 <sub>H</sub>	ADCRH	ADCRH<7:0>								0000 0000
6076 <sub>H</sub>	ADCCL	ADCHS<3:0>				SMPON	SMPS	ADTRG	ADEN	0000 0100
6077 <sub>H</sub>	ADCCH	ADFM	ADCKS<2:0>			ADST<1:0>		ADVCMHS	ADHSEN	0100 1011
6078 <sub>H</sub>	WDTUL	WDTUL<7:0>								0000 0000
6079 <sub>H</sub>	WDTC	—	—	—	SWDTEN	WDTIEN	WDTPRS<2:0>			0001 1111
607A <sub>H</sub>	OSCC1	CHG	DIVS<2:0>			—	SST	—	SCKS	0000 0000
607B <sub>H</sub>	OSCC2	CLKOEN	—	HRCON	LRCON	—	—	HRCEN	—	0000 0000
607C <sub>H</sub>	OSCWP	OSCWP<7:0>								0000 0000
607D <sub>H</sub>	CLKG	ADCCE	T21CE	T20CE	-	TKCE	IICCE	UART0CE	T10CE	1110 1111
607E <sub>H</sub>	LVDCL	—	—	—	LVDFLTEN	LVDIFS<1:0>		LVDO	LVDEN	0000 00x0
607F <sub>H</sub>	LVDCH	LVDVS<3:0>				—	—	—	—	0000 0000
6080 <sub>H</sub> ~ 60FF <sub>H</sub>	—	—								

### 附录3 电气特性

#### 附录3.1 参数特性表

◆ 最大标称值

参数	符号	条件	标称值	单位
电源电压	VDD	—	-0.3 ~ 5.8	V
输入电压	V <sub>IN</sub>	—	-0.3 ~ VDD + 0.3	V
输出电压	V <sub>OUT</sub>	—	-0.3 ~ VDD + 0.3	V
存储温度	T <sub>STG</sub>	—	-55 ~ 125	°C
操作温度	T <sub>OPR</sub>	VDD: 2.2 ~ 5.5V	-40 ~ 85	°C

◆ ESD 特性参数

参数	符号	等级	最大值	单位	测试条件
ESD 电压 (人体模型)	V <sub>ESDHBM</sub>	3A	4000	V	25°C, MIL-STD-883J
ESD 电压 (机器模型)	V <sub>ESDMM</sub>	3	400	V	25°C, JESD22-A115
ESD 电压 (充电器件模型)	V <sub>ESDCDM</sub>	C3	1000	V	25°C, JEDEC JS-002-2014
LatchUp 电流	I <sub>LAT</sub>	I	±350	mA	25°C, JESD78

注：上述 ESD 特性参数值是基于理论设计值和被测样品的测试值，不是批量产品测试值，仅供芯片应用时参考。

◆ 芯片功耗特性参数表

参数	符号	最小值	典型值	最大值	单位	工作条件
芯片供电电压	VDD	2.5	—	5.5	V	F <sub>osc</sub> ≤ 2MHz -40°C ~ 85°C
IDLE 休眠模式 下芯片电流	I <sub>PD1</sub>	—	5	—	μA	25°C, VDD = 5V, BOR 使能, WDT 不使能。
		—	4	—	μA	25°C, VDD = 5V, BOR 不使能, WDT 使能。
芯片静态电流	I <sub>DD</sub>	—	400	—	uA	25°C, VDD = 5V, 所有的 I/O 端口输入低电平, N_MRST = 0。
正常运行模式 芯片电流	I <sub>OP1</sub>	—	3	—	mA	25°C, VDD = 5V, 正常运 行模式, 内部 16MHz HRC 时钟, I/O 端口输出固定电 平, 无负载。
	I <sub>OP2</sub>	—	100	—	uA	25°C, VDD = 5V, 正常运 行模式, 内部 32KHZ LRC 时钟, I/O 端口输出固定电 平, 无负载, ADC、TK 关 闭。

参数	符号	最小值	典型值	最大值	单位	工作条件
VDD 管脚的最大输入电流	$I_{MAXVDD}$	—	—	80	mA	25°C, VDD = 5V
VSS 管脚的最大输出电流	$I_{MAXVSS}$	—	—	200	mA	25°C, VDD = 5V
I/O 端口灌电流	$I_{OL1}$	—	6.5	—	mA	25°C, VDD = 5V $V_{OL} = 0.6V$ , 弱驱动
	$I_{OL2}$	—	8	—	mA	25°C, VDD = 5V $V_{OL} = 0.6V$ , 强驱动
I/O 端口拉电流	$I_{OH}$	—	4	—	mA	25°C, VDD = 5V $V_{OH} = 4.4V$ , 禁止源电流
I/O 端口拉电流	$I_{OH1}$	—	4	—	mA	25°C, VDD = 5V $V_{OH} = 4.4V$ , 源电流 1
	$I_{OH1}$	—	7	—	mA	25°C, VDD = 5V $V_{OH} = 4.4V$ , 源电流 2
	$I_{OH1}$	—	11	—	mA	25°C, VDD = 5V $V_{OH} = 4.4V$ , 源电流 3

◆ 芯片输入端口特性表

芯片工作温度范围: -40°C ~ 85°C						
参数	符号	最小值	典型值	最大值	单位	测试条件
IO 端口输入高电平 (SMIT0)	$V_{IH}$	0.6VDD	—	VDD	V	2.7V ≤ VDD ≤ 5.5V
IO 端口输入高电平 (SMIT1)		0.4VDD	—	VDD	V	
主复位信号 N_MRST 输入高电平 (有施密特输入特性)		0.6VDD	—	VDD	V	
IO 端口输入低电平 (SMIT0)	$V_{IL}$	VSS	—	0.3VDD	V	
IO 端口输入低电平 (SMIT1)		VSS	—	0.16VDD	V	
主复位信号 N_MRST 输入低电平		VSS	—	0.3VDD	V	
IO 端口输入漏电流	$I_{IL}$	—	—	±1	μA	2.7V ≤ VDD ≤ 5.5V VSS ≤ Vpin ≤ VDD (端口处于高阻状态)

芯片工作温度范围: -40°C ~ 85°C						
参数	符号	最小值	典型值	最大值	单位	测试条件
主复位端口漏电流		—	—	±1	μA	VSS ≤ Vpin ≤ VDD
IO 端口输入弱上拉电流	I <sub>WPU</sub>	—	110	—	μA	2.7V ≤ VDD ≤ 5.5V Vpin = VSS
IO 端口输入弱下拉电流	I <sub>WPD</sub>	—	140	—	μA	2.7V ≤ VDD ≤ 5.5V Vpin = VDD

◆ 芯片输出端口特性表

芯片工作温度范围: -40°C ~ 85°C						
参数	符号	最小值	典型值	最大值	单位	测试条件
I/O 端口输出高电平	V <sub>OH</sub>	VDD-0.7	—	—	V	2.7V ≤ VDD ≤ 5.5V I <sub>OH</sub> = 2mA
I/O 端口输出低电平	V <sub>OL</sub>	—	—	0.6	V	2.7V ≤ VDD ≤ 5.5V I <sub>OL</sub> = 3mA

◆ 系统时钟要求表

参数	符号	最小值	典型值	最大值	单位	测试条件
系统时钟频率	F <sub>OSC</sub>	—	—	16M	Hz	2.7V ≤ VDD ≤ 5.5V
系统时钟周期	T <sub>OSC1</sub>	62.5	—	—	ns	2.7V ≤ VDD ≤ 5.5V
外部时钟高电平和低电平时间	T <sub>OSL</sub> , T <sub>OSH</sub>	15	—	—	ns	—
外部时钟上升和下降时间	T <sub>OSR</sub> , T <sub>OSF</sub>	—	—	15	ns	—
WDT 溢出时间	T <sub>WDT</sub>	6 (40KHz)	8 (32KHz)	10 (24KHz)	ms	2.7V ≤ VDD ≤ 5.5V -40°C ~ 85°C

◆ 内部 16MHz 时钟校准特性表

校准条件	工作条件	最小值	典型值	最大值	单位
5V, 25°C 将频率校准至 16MHz	25°C, VDD = 5V	15.84	16	16.16	MHz
	-40°C ~ 85°C, VDD = 2.7V ~ 5.5V	15.68	16	16.32	MHz

◆ 内部 32KHz 时钟校准特性表

校准条件	工作条件	最小值	典型值	最大值	单位
5V, 25°C 将频率校准至 32KHz	25°C, VDD = 5V	31.36	32	32.64	KHz
	-40°C ~ 85°C, VDD = 2.7V ~ 5.5V	27.2	32	36.8	KHz

◆ 低电压复位 BOR 电压特性

参数	符号	最小值	典型值	最大值	单位	测试条件
BOR 低电压设定电压 1	$V_{bor1}$	3.6	3.7	3.75	V	25°C, VDD=2.7V~5.5V
BOR 低电压设定电压 2	$V_{bor2}$	3.0	3.1	3.15	V	25°C, VDD=2.7V~5.5V
BOR 低电压设定电压 3	$V_{bor3}$	2.4	2.5	2.55	V	25°C, VDD=2.7V~5.5V
BOR 低电压设定电压 4	$V_{bor4}$	2.2	2.3	2.35	V	25°C, VDD=2.7V~5.5V
BOR 低电压复位脉宽	$T_{bor}$	—	220	—	us	设计理论值

◆ 低电压检测 LVD 电压特性

参数	符号	最小值	典型值	最大值	单位	测试条件
LVD 低电压设定电压 1	$V_{lvd1}$	4.5	4.6	4.65	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 2	$V_{lvd2}$	3.9	4.0	4.05	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 3	$V_{lvd3}$	3.5	3.6	3.65	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 4	$V_{lvd4}$	2.9	3.0	3.05	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 5	$V_{lvd5}$	2.7	2.8	2.85	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 6	$V_{lvd6}$	2.5	2.6	2.65	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 7	$V_{lvd7}$	2.3	2.4	2.45	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 8	$V_{lvd8}$	2.1	2.2	2.25	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 9	$V_{lvd9}$	2.0	2.1	2.15	V	25°C, VDD=2.7V~5.5V
LVD 低电压设定电压 10	$V_{lvd10}$	1.9	2.0	2.05	V	25°C, VDD=2.7V~5.5V
LVD 低电压复位脉宽	$T_{lvd}$	—	220	—	us	设计理论值

◆ ADC 交流特性表

参数	符号	最小值	典型值	最大值	单位	测试条件
电源电压	VDD	2.5	—	5.5	V	—
分辨率	R <sub>R</sub>	—	—	9	bit	—
差分线性度	DNL	—	—	±4	LSB	—
积分线性度	INL	—	—	±4	LSB	—
失调误差	E <sub>OFF</sub>	5	8	—	LSB	—
参考电压	V <sub>REF</sub>	—	—	VDD	V	—
模拟输入电压	V <sub>ADIN</sub>	0	—	VDD	V	—
输入电容	C <sub>ADIN</sub>	—	—	10	Pf	—
输入电阻	R <sub>ADIN</sub>	—	—	1	KΩ	—
转换时钟频率	F <sub>ADCLK</sub>	32KHz	—	16	MHz	AD 转换选择 VDD 作为参考电压； 高速模式 (ADHSEN=1, ADVCMHS=1)
转换时间 (不包括采样时间)	T <sub>ADC</sub>	—	13	—	Tadclk	—
采样时间	T <sub>ADS</sub>	2	—	—	Tadclk	—

注 1: 建议 ADC 转换在高速模式下进行

◆ ADC 转换时间对照表

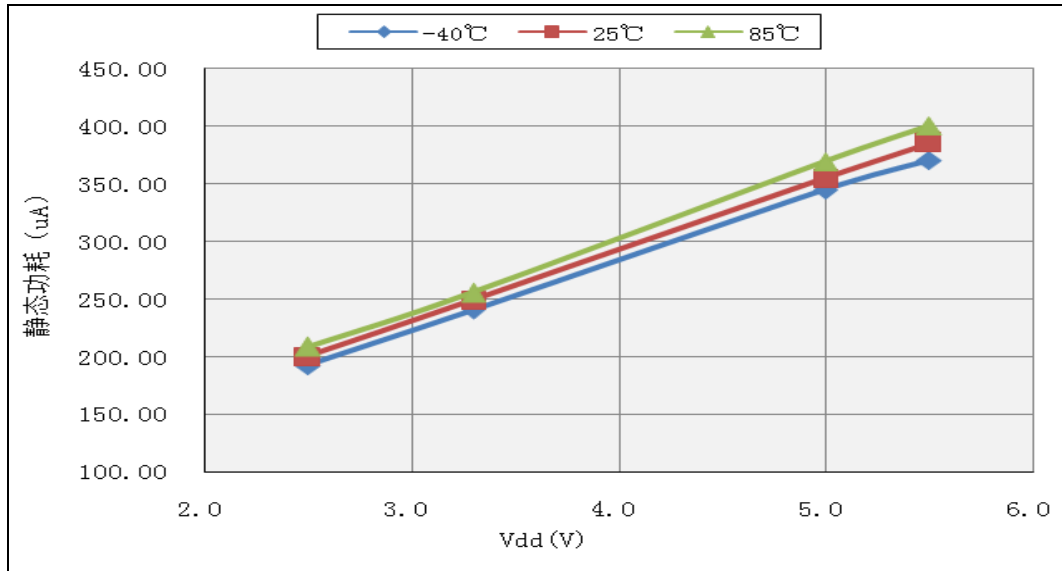
A/D 时钟源选择	工作频率
Fosc	不推荐使用*2
Fosc/2	T <sub>ADCCLK</sub> = 0.125us
Fosc/4	T <sub>ADCCLK</sub> = 0.25us
Fosc/8	T <sub>ADCCLK</sub> = 0.5us
Fosc/16	T <sub>ADCCLK</sub> = 1us
Fosc/32	T <sub>ADCCLK</sub> = 2us
Fosc/64	T <sub>ADCCLK</sub> = 4us
LRC	T <sub>ADCCLK</sub> = 31.25us

注\*2: Tad 值不满足设计要求不推荐使用;

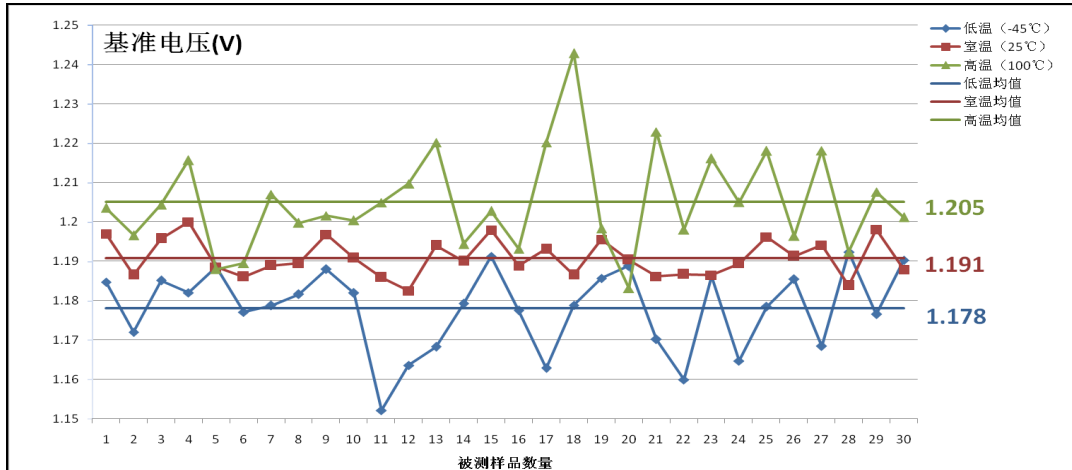
### 附录3.2 参数特性图

本节中所列图示均为抽样测试，仅作为设计参考之用。其中部分图示中所列的数据已超出指定的操作范围，此类信息也仅供参考，芯片只保证在指定的范围内正常工作。

◆ 芯片静态电流随芯片电压变化特性图

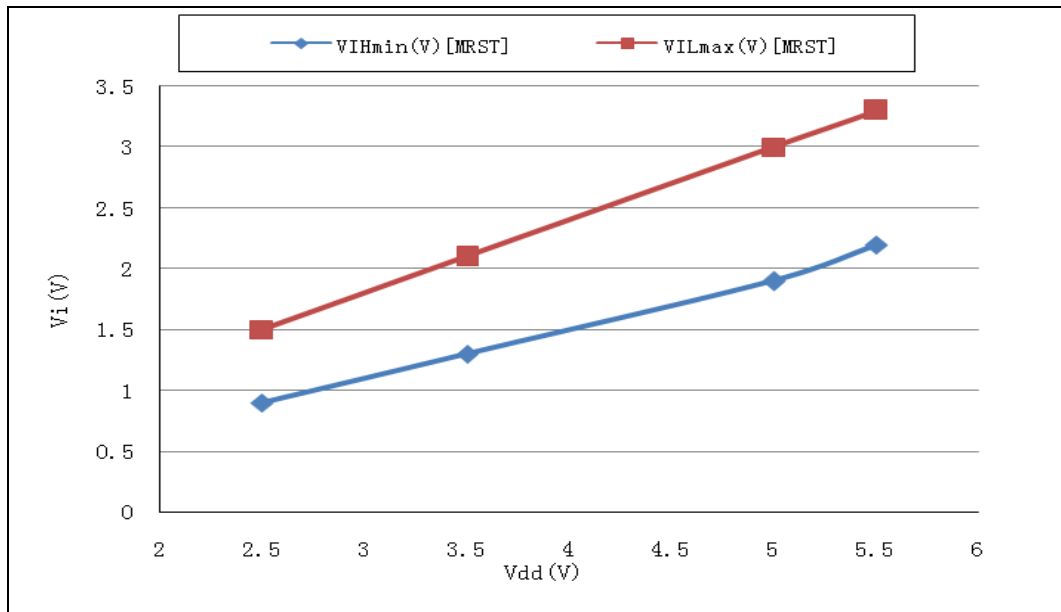


◆ 内部 1.2V 基准电压精度特性图

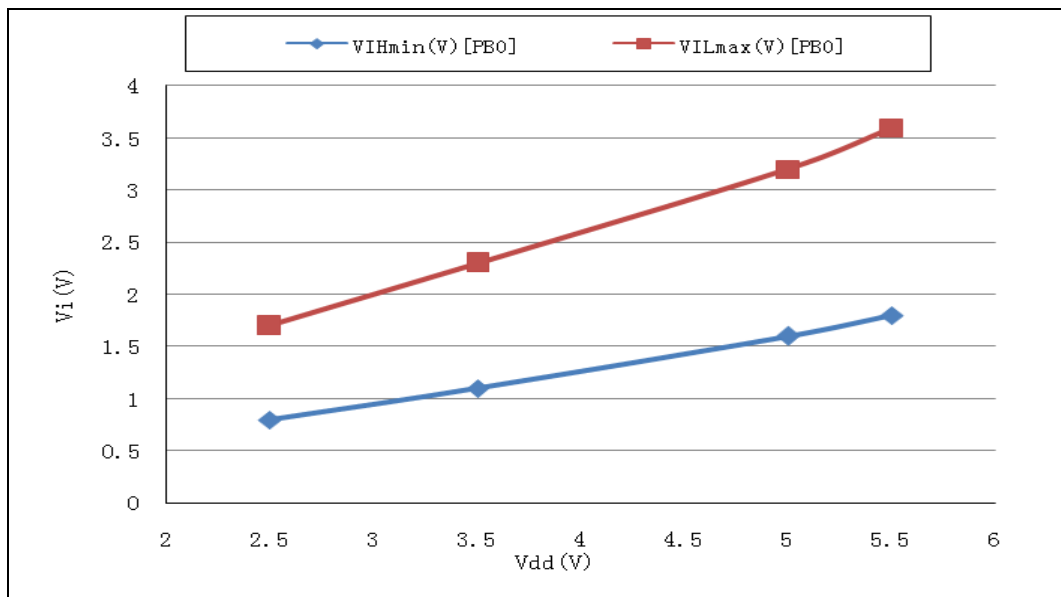




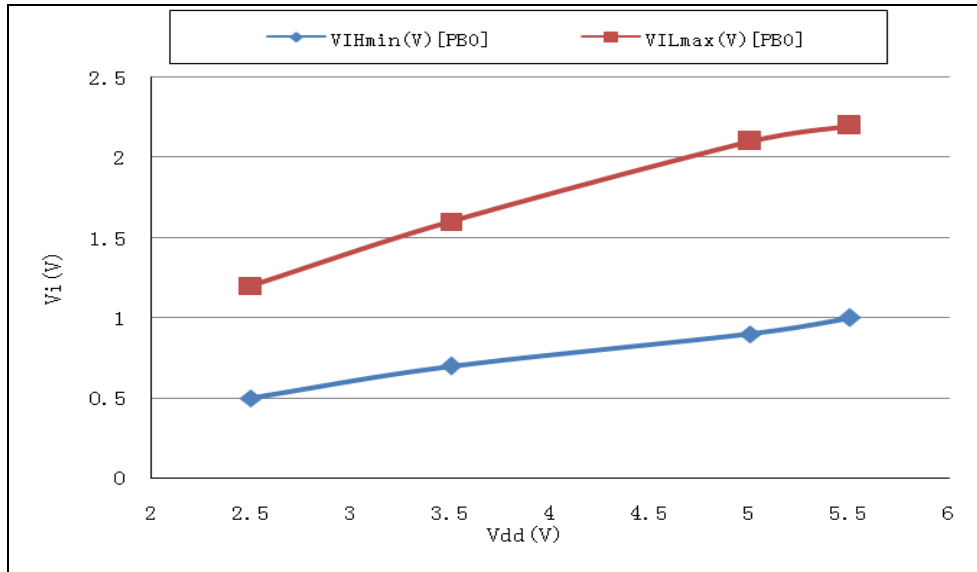
◆ 复位管脚输入特性图（室温 25°C）



◆ I/O 管脚 SIMT0 输入特性图（室温 25°C）

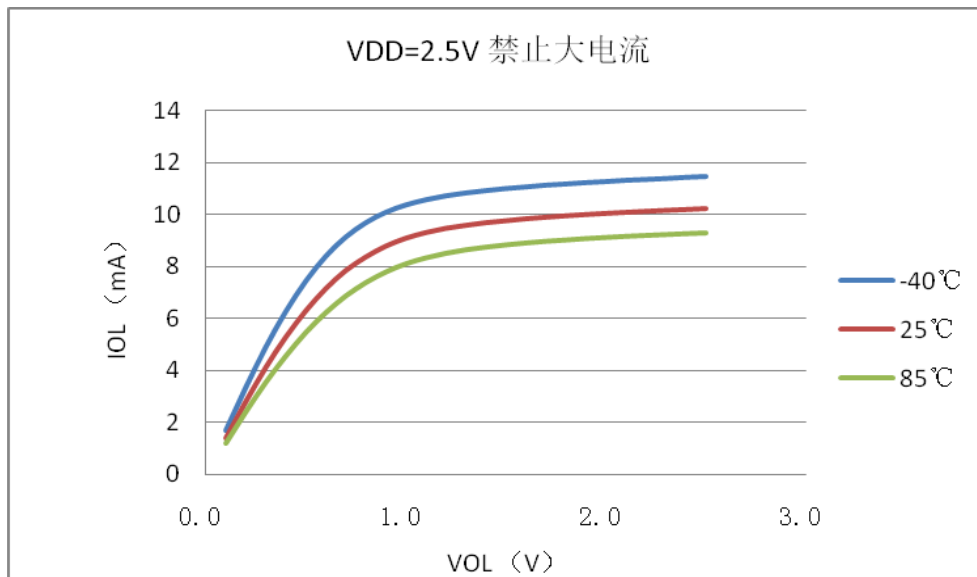


◆ I/O 管脚 SIMT1 输入特性图 (室温 25°C)

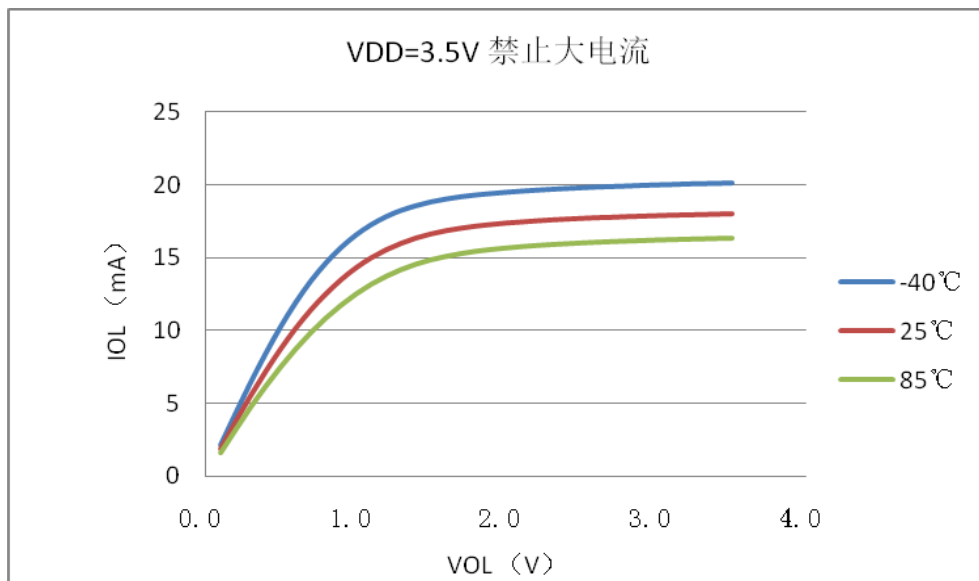


◆ I/O 端口信号输出特性图 ( $V_{OL}$  vs  $I_{OL}$  @禁止大电流驱动)

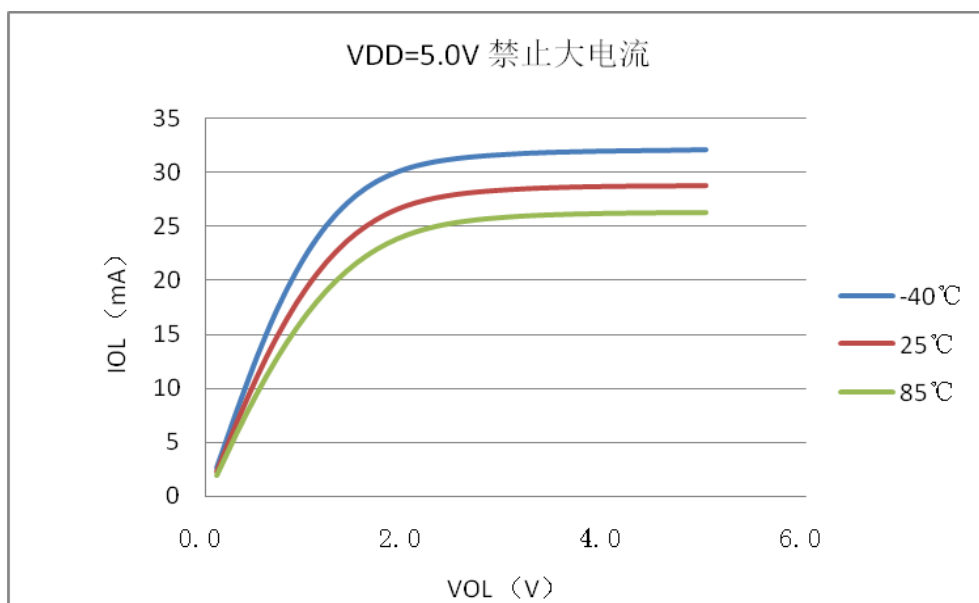
A:  $V_{OL}$  vs  $I_{OL}$  @VDD=2.5V



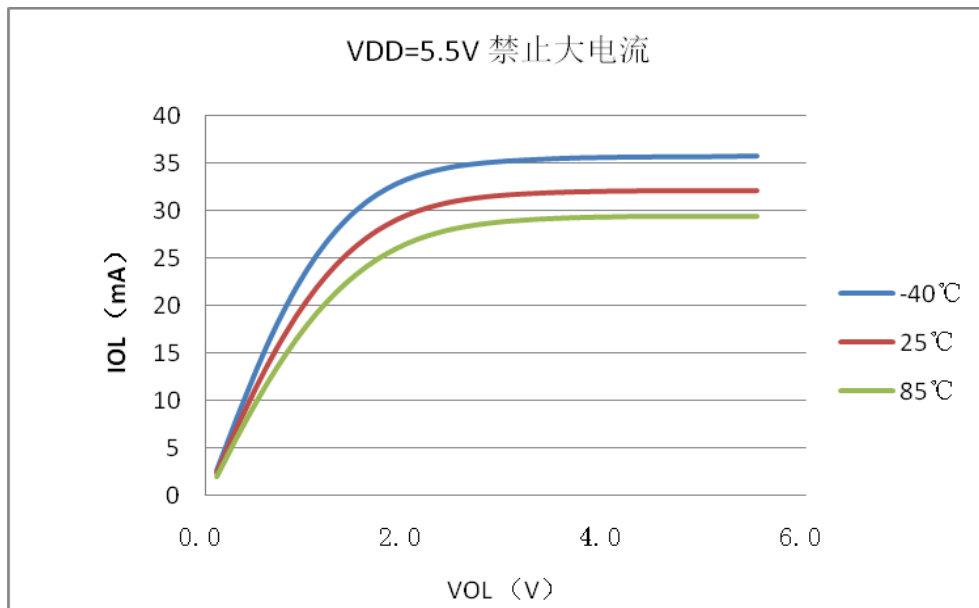
B:  $V_{OL}$  vs  $I_{OL}$  @VDD=3.5V



C:  $V_{OL}$  vs  $I_{OL}$  @VDD=5.0V

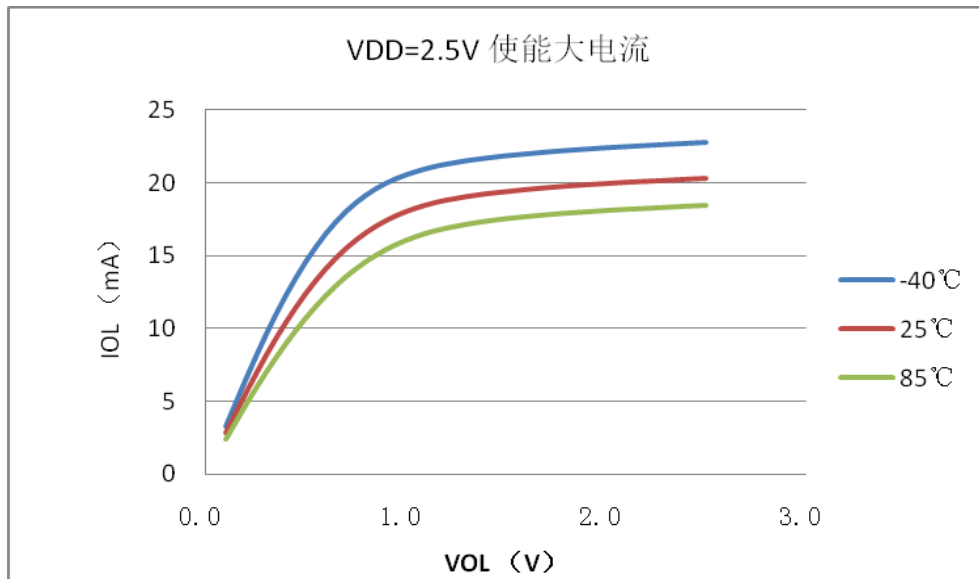


D:  $V_{OL}$  vs  $I_{OL}$  @VDD=5.5V

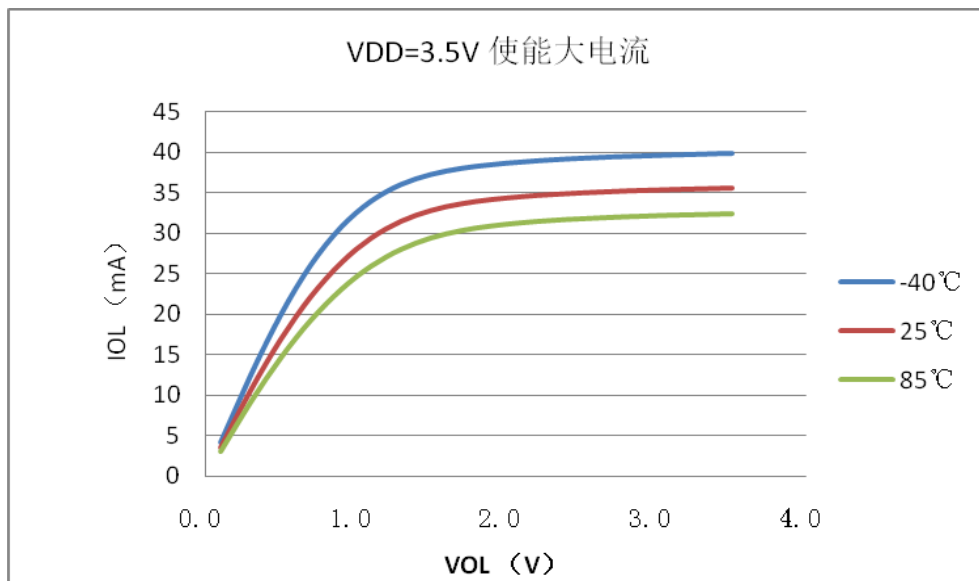


◆ I/O 端口信号输出特性图 ( $V_{OL}$  vs  $I_{OL}$  @使能大电流驱动)

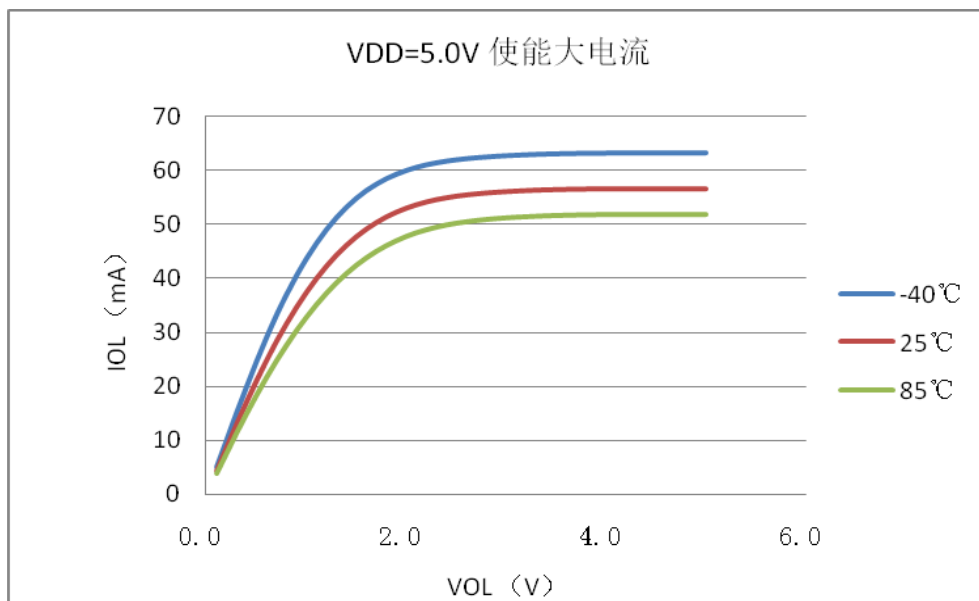
A:  $V_{OL}$  vs  $I_{OL}$  @VDD=2.5V



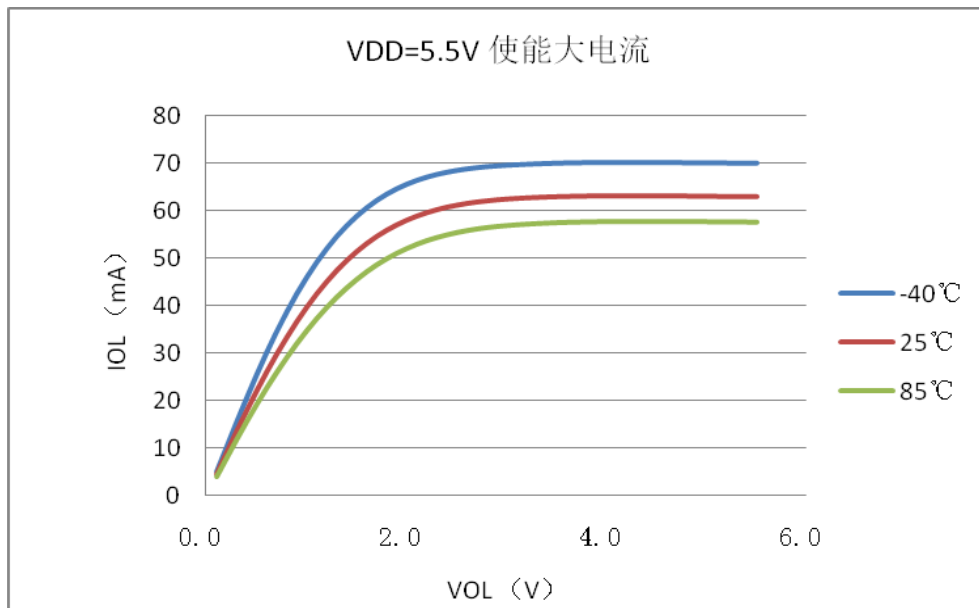
B:  $V_{OL}$  vs  $I_{OL}$  @VDD=3.5V



C:  $V_{OL}$  vs  $I_{OL}$  @VDD=5.0V

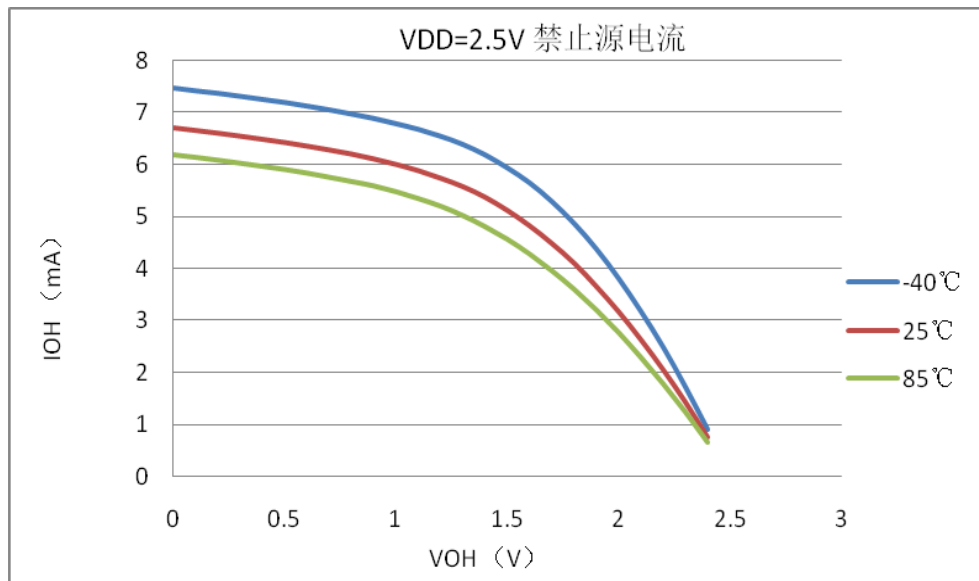


D:  $V_{OL}$  vs  $I_{OL}$  @VDD=5.5V

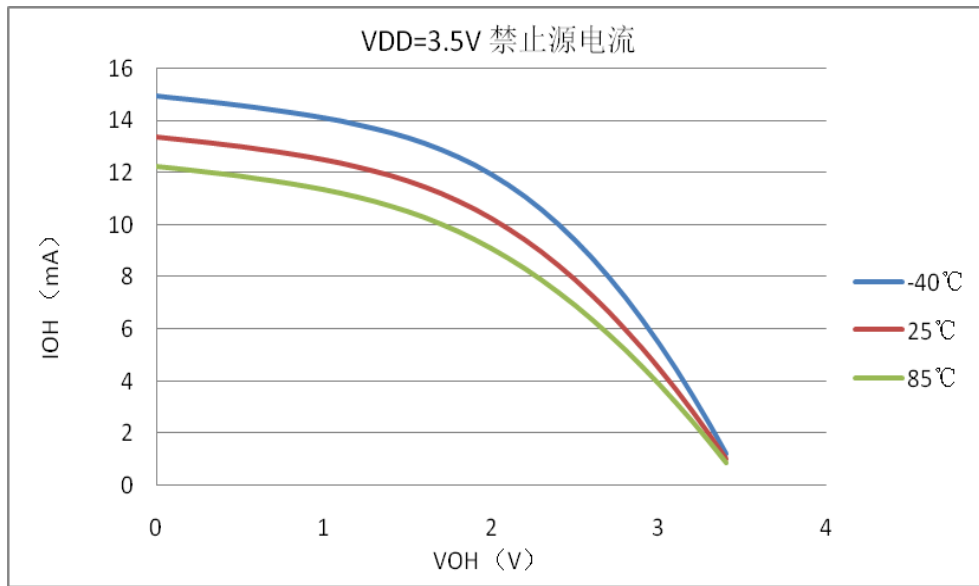


◆ I/O 端口信号输出特性图 ( $V_{OH}$  vs  $I_{OH}$  @禁止源电流)

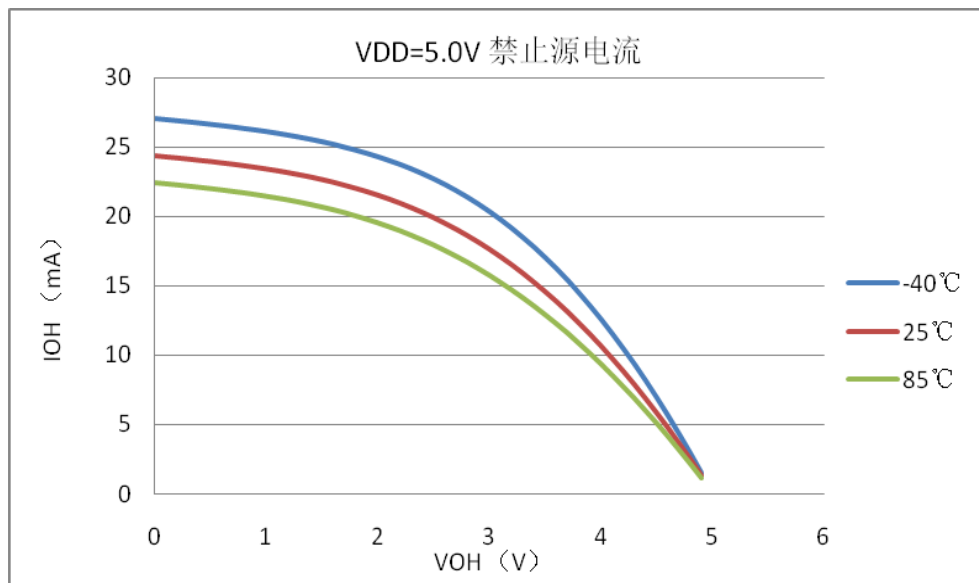
A:  $V_{OH}$  vs  $I_{OH}$  @VDD=2.5V



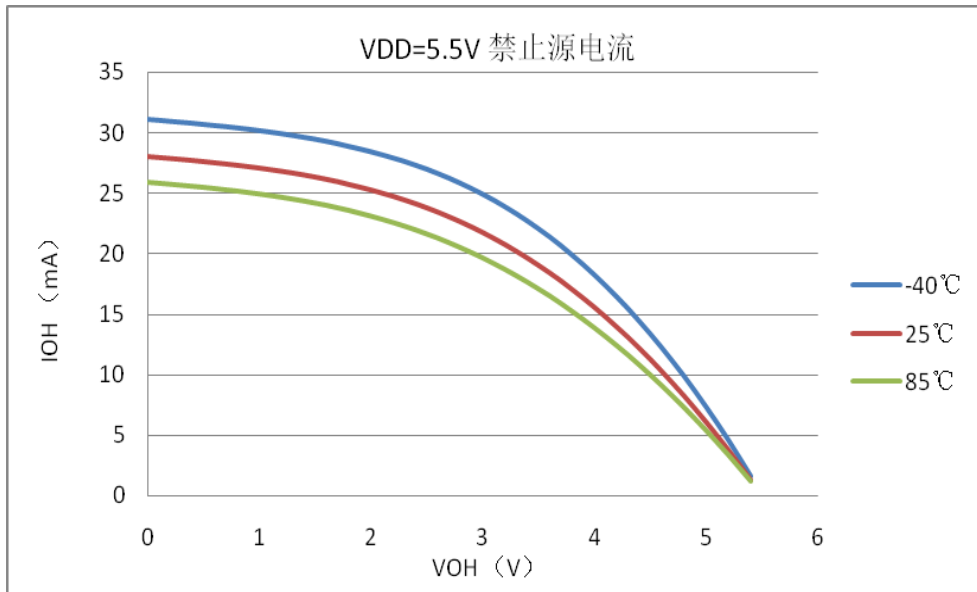
B:  $V_{OH}$  vs  $I_{OH}$  @VDD=3.5V



C:  $V_{OH}$  vs  $I_{OH}$  @VDD=5.0V

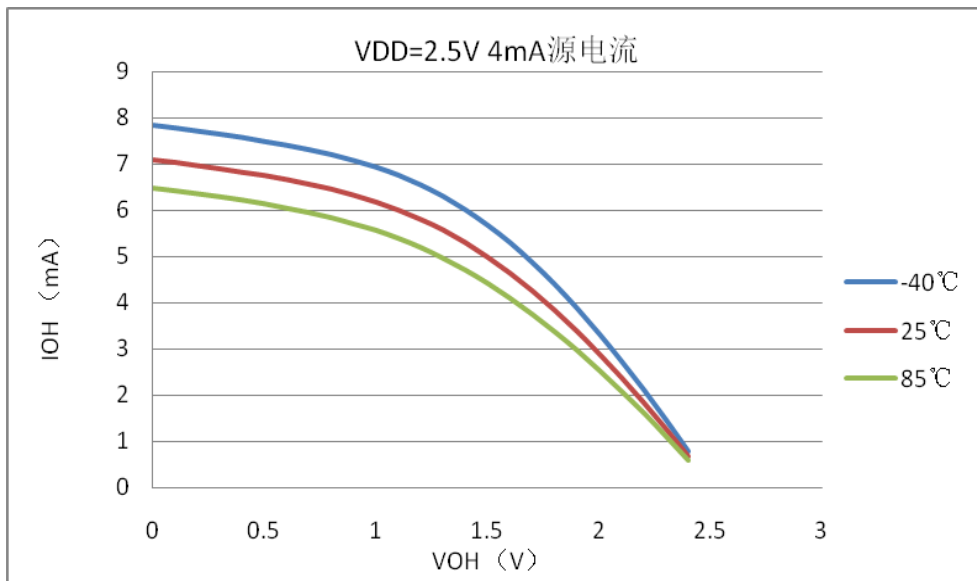


D:  $V_{OH}$  vs  $I_{OH}$  @VDD=5.5V



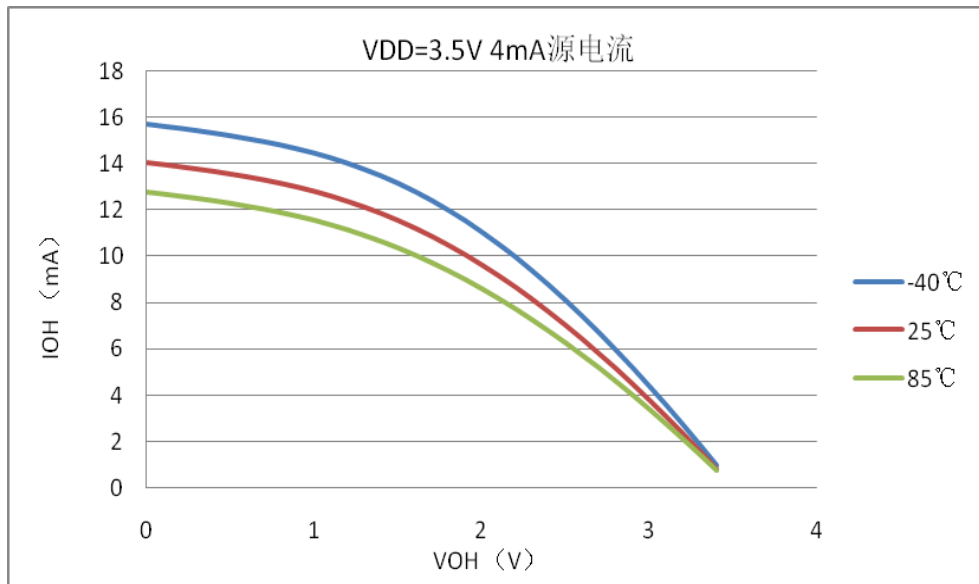
◆ I/O 端口信号输出特性图 ( $V_{OH}$  vs  $I_{OH}$  @4mA 源电流)

A:  $V_{OH}$  vs  $I_{OH}$  @VDD=2.5V

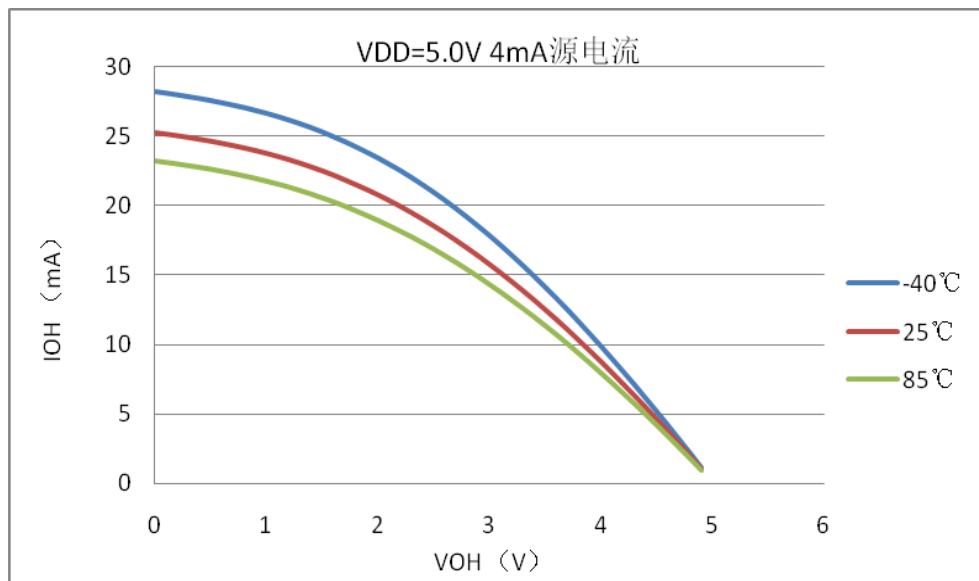




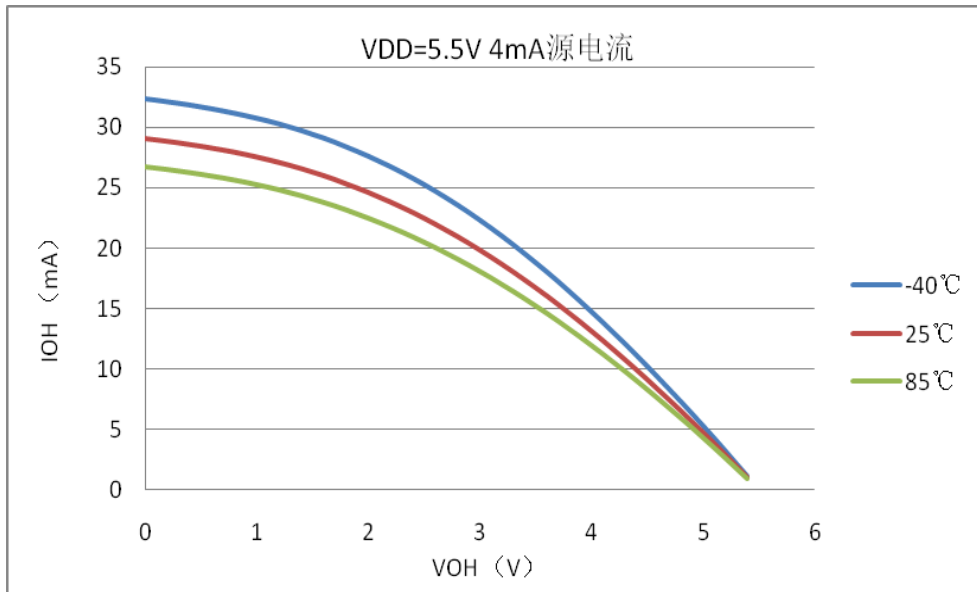
B:  $V_{OH}$  vs  $I_{OH}$  @VDD=3.5V



C:  $V_{OH}$  vs  $I_{OH}$  @VDD=5.0V

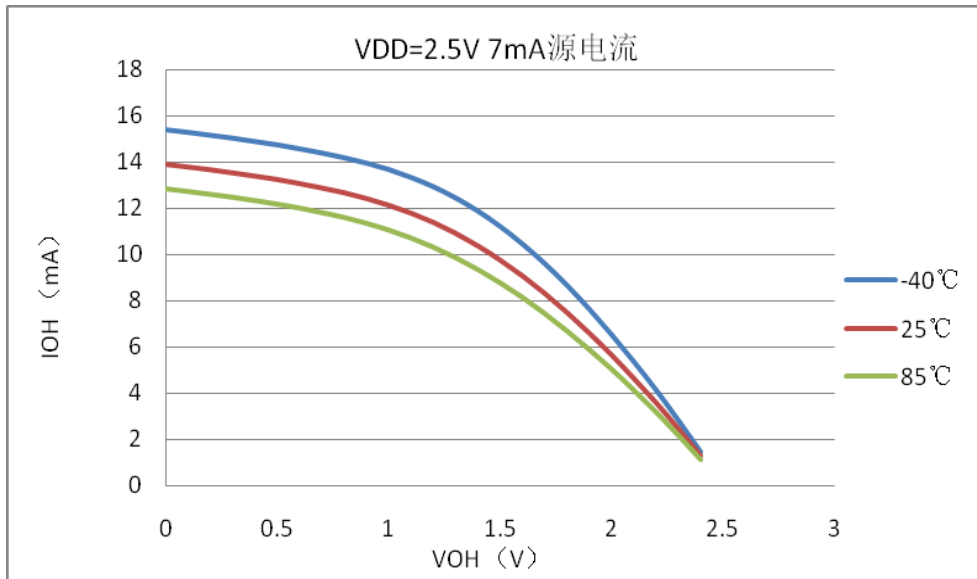


D:  $V_{OH}$  vs  $I_{OH}$  @  $V_{DD}=5.5V$

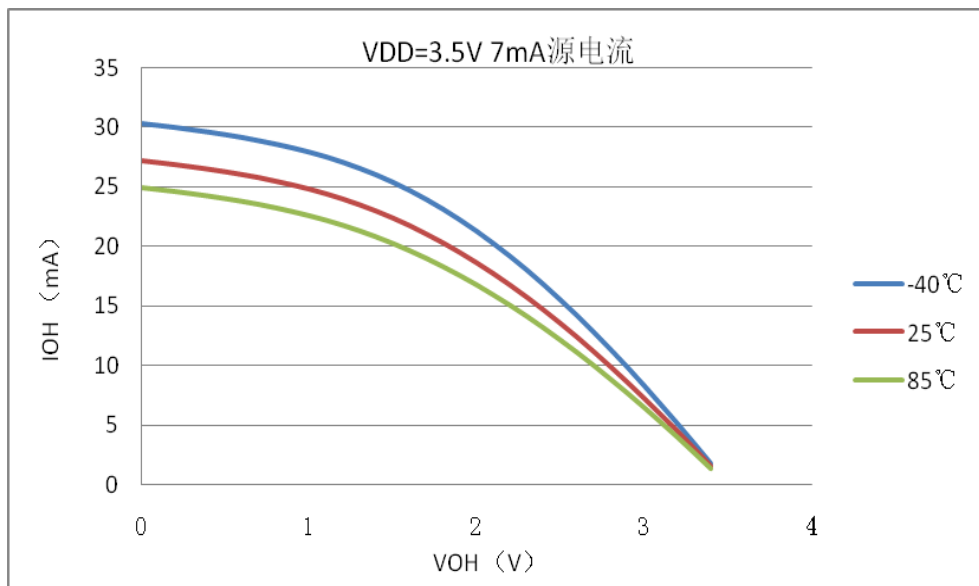


◆ I/O 端口信号输出特性图 ( $V_{OH}$  vs  $I_{OH}$  @7mA 源电流)

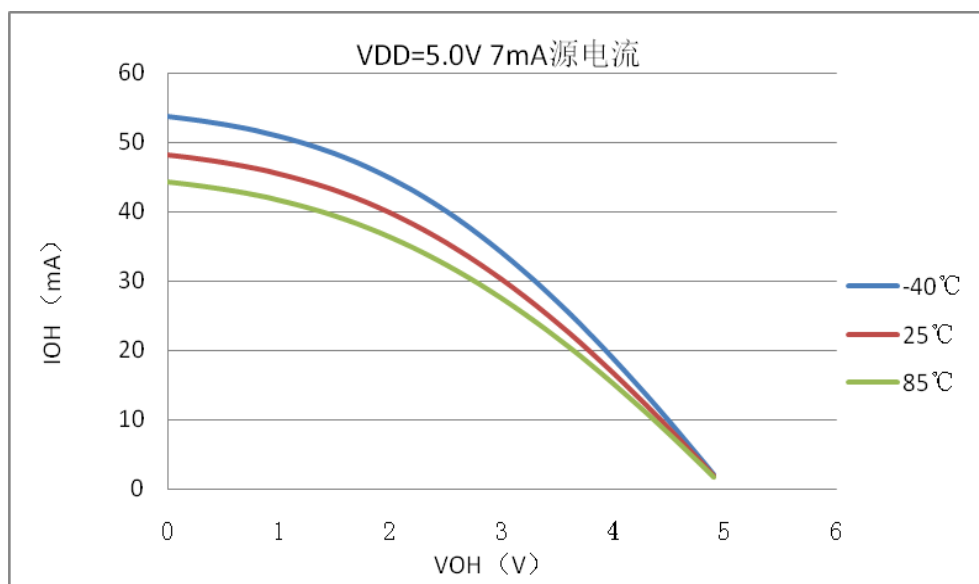
A:  $V_{OH}$  vs  $I_{OH}$  @  $V_{DD}=2.5V$



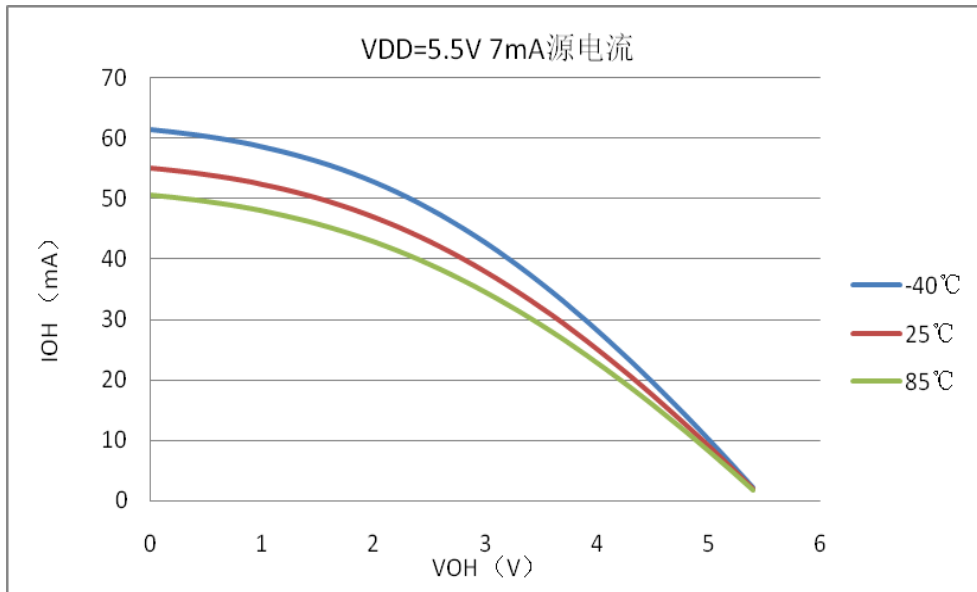
B:  $V_{OH}$  vs  $I_{OH}$  @VDD=3.5V



C:  $V_{OH}$  vs  $I_{OH}$  @VDD=5.0V

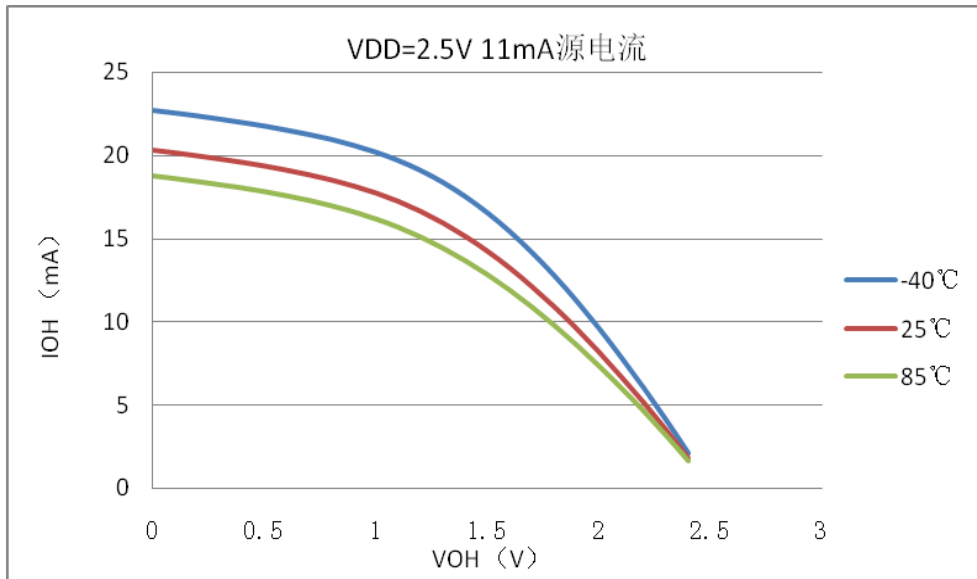


D:  $V_{OH}$  vs  $I_{OH}$  @VDD=5.5V

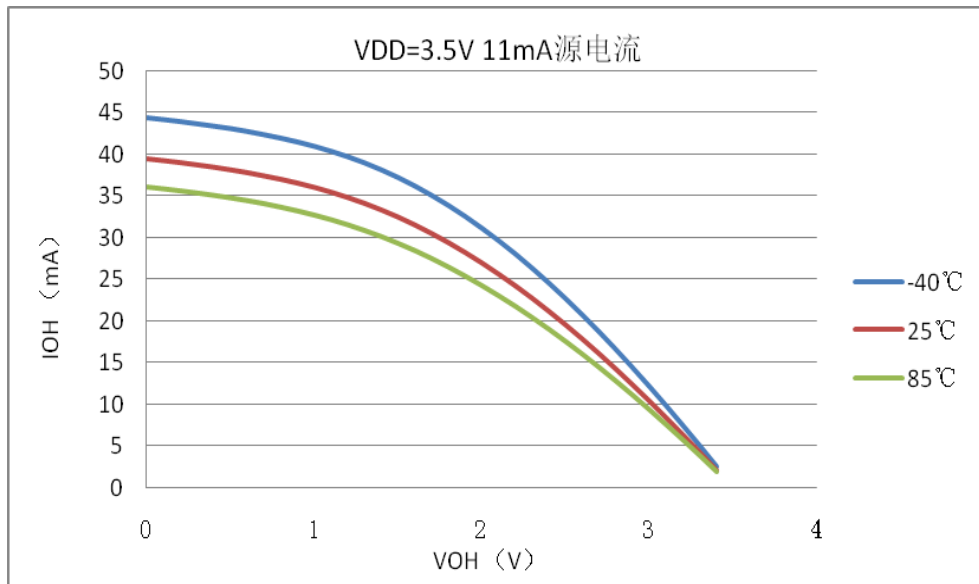


◆ I/O 端口信号输出特性图 ( $V_{OH}$  vs  $I_{OH}$  @11mA 源电流)

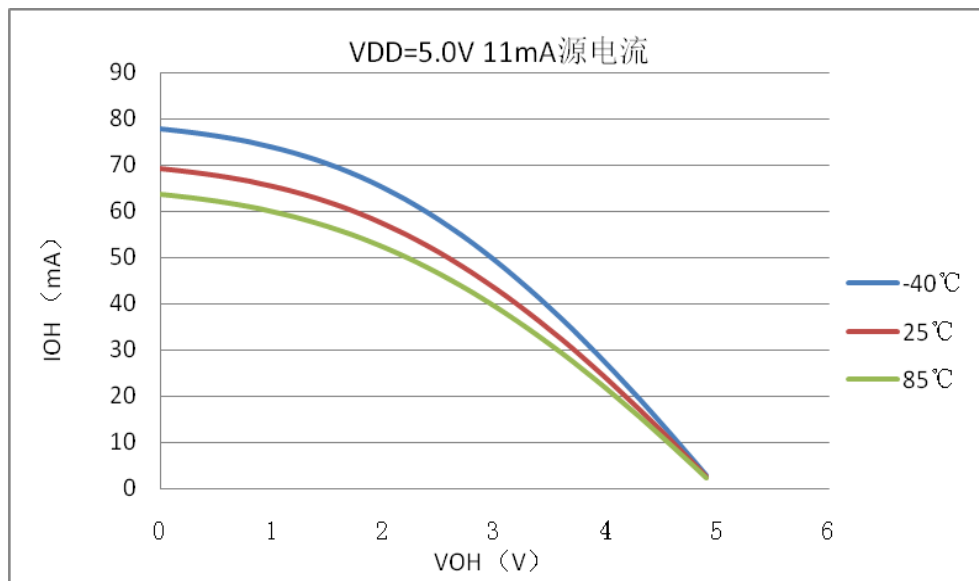
A:  $V_{OH}$  vs  $I_{OH}$  @VDD=2.5V



B:  $V_{OH}$  vs  $I_{OH}$  @VDD=3.5V



C:  $V_{OH}$  vs  $I_{OH}$  @VDD=5.0V



D:  $V_{OH}$  vs  $I_{OH}$  @VDD=5.5V

