

文档版本	V1.0
发布日期	20210524

APT32F102 I2C 应用指南

APTCHIP



目录

1 概述	1
2. 适用的硬件.....	1
3. 应用方案代码说明	1
3.1 I2C 配置	1
3.2 I2C 主机模式	2
3.3 I2C 从机模式	3
4. 程序下载和运行	6

1 概述

本文介绍了在APT32F102中使用I2C的应用范例。

2. 适用的硬件

该例程使用于 APT32F102 开发板 APT-DB102

3. 应用方案代码说明

3.1 I2C 配置

基于 APT32F102 完整的库文件系统，对 I2C 进行配置。

- 硬件配置：

I2C 主-从设备常见硬件连接电路

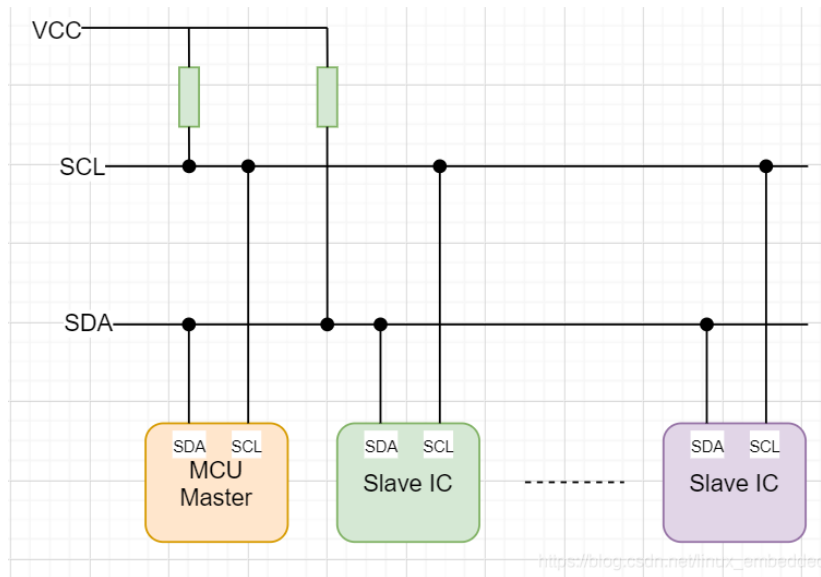


图 3.1.1 I2C 硬件

通讯引脚：

I2C	I2C_SCL(B)	PB0.0/PB0.2/PA0.1/PA0.4/PA0.6/PA0.15
	I2C_SDA(B)	PA0.0/PA0.3/PA0.7/PA0.13/PA0.14

三种传输模式：标准模式传输速率为 100kbit/s ，快速模式为 400kbit/s ，高速模式下可达 1000kbit/s。

支持 7 位、10 位设备地址

● 软件配置：

可在 apt32f102_initial.c 文件中进行初始化的配置；

编程要点：

1. 通过 SYSCON_CONFIG() 配置时钟
2. 通过 GPIO_CONFIG()设置目标引脚为开漏模式
3. 通过 I2C_MASTER_CONFIG 配置 I2C 主机或者通过 I2C_SLAVE_CONFIG 设置 I2C 从机。

```

/*****/
//GPIO Initial
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void GPIO_CONFIG(void)
{
    GPIO_OpenDrain_EN(GPIOA0,13); //漏极开漏输出
    GPIO_OpenDrain_EN(GPIOB0,0);
}

```

3.2 I2C 主机模式

```

/*****/
//IIC MASTER Initial
//EntryParameter:NONE
//ReturnValue:NONE
/*****/
void I2C_MASTER_CONFIG(void)
{
    I2C_DeInit();
    //address=0X43,SCL=IIClk*250=5us
    I2C_Master_CONFIG(I2C_SDA_PA013,I2C_SCL_PB00,FAST_MODE,I2C_MASTRE_7BIT,0Xae,0Xa0,0Xa0);
}

```

```

I2C_SDA_TSETUP_THOLD_CONFIG(0x40,0x40,0x40);

//TSETUP=0X20*IICLK,RX_THOLD=0X20*IICLK,TX_THOLD=0X20*IICLK

I2C_Enable();

}

```

● 代码说明:

I2C_DeInit();---用于复位寄存器

I2C_Master_CONFIG();---用于配置 I2C 参数

I2C_SDA_TSETUP_THOLD_CONFIG();---用于配置 SDA 保持数据时间

I2C_Enable();----用于使能 I2C 模块

● 参数说明:



● 测试

主程序中测试发送，写入对应 I2C 设备。

```

int main(void)
{
    APT32F102_init();

    While(1)
    {
        I2C_WRITE_Byte(0x01,0x55);

        delay_nms(500);
    }
}

```

3.3 I2C 从机模式

```

/*****/

//IIC SLAVER Initial
//EntryParameter:NONE
//ReturnValue:NONE

```

```

/*****/

void I2C_SLAVE_CONFIG(void)
{
    I2C_DeInit();

    //选择 I2C PA0.14 PA0.15 .高速模式从机地址不能设置为 0x00~0x07,0x78~0x7f

    I2C_Slave_CONFIG(I2C_SDA_PA014,I2C_SCL_PA015,FAST_MODE,I2C_SLAVE_7BIT,0X57,0X50,0X50);

    //I2C setup time=0X40/48M=1.3uS

    I2C_SDA_TSETUP_THOLD_CONFIG(0x40,0x40,0x40);

    //发送/接收 FIFO 阈值:该寄存器的值 0 表示有 1 个数据, 7 表示阈值为 8 个数据。

    I2C_FIFO_TriggerData(0,0);

    //SCL 锁死状态 表示 SCL 是否被拉低锁死, 锁死时间超过了 I2C_SCL_TOUT 个 PCLK 周期的时长。

    I2C_ConfigInterrupt_CMD(ENABLE,I2C_SCL_SLOW);

    //停止位检测状态

    I2C_ConfigInterrupt_CMD(ENABLE,I2C_STOP_DET);

    //当 I2C 工作在从机模式并且另外一个 I2C 主机正在尝试从该从机设备读取数据时, 该位置 1

    I2C_ConfigInterrupt_CMD(ENABLE,I2C_RD_REQ);

    //RX_FULL 状态 :当缓冲区数据个数等于或者大于 I2C_RX_FLSEL 寄存器中 RX_FLSEL 阈值时置 1

    I2C_ConfigInterrupt_CMD(ENABLE,I2C_RX_FULL);

    //TX_ABRT 状态 1:I2C_TX_ABRT 寄存器会保存发送中止的原因。

    I2C_ConfigInterrupt_CMD(ENABLE,I2C_TX_ABRT);

    I2C_Enable(); //开启 I2C

    I2C_Int_Enable(); //开启 I2C 中断

}

```

● 代码说明:

I2C_DeInit(); --- 用于恢复默认寄存器

I2C_Slave_CONFIG();----用于配置 I2C 从机模式参数

I2C_SDA_TSETUP_THOLD_CONFIG();----用于设置 SCL 保持时间

I2C_FIFO_TriggerData(); ----用于配置发送\接收 FIFO 阈值

I2C_ConfigInterrupt_CMD();----用于开启 I2C 相应的中断使能

I2C_SCL_SLOW: SCL 锁死状态

I2C_STOP_DET: 停止位检测状态

I2C_RD_REQ: 从机模式下读请求状态

I2C_RX_FULL: RX 缓冲区状态

I2C_TX_ABRT: 发送被中止状态

I2C_Enable();----用于使能 I2C 模块

I2C_Int_Enable();--用于开启 I2C 中断

- 102x 系列做从机时，请提升 I2C 的中断优先级。

```
/*-----*/
//Interrupt Priority setting
//EntryParameter:00/40/80/C0
//CORET_INT   IRQ0
//SYSCON_INT  IRQ1
//IFC_INT     IRQ2
//ADC_INT     IRQ3
//EPT0_INT    IRQ4
//****DUMMY  IRQ5
//WWDT_INT    IRQ6
//EXI0_INT    IRQ7
//EXI1_INT    IRQ8
//GPT0_INT    IRQ9
//****DUMMY  IRQ10
//****DUMMY  IRQ11
//RTC_INT     IRQ12
//UART0_INT   IRQ13
//UART1_INT   IRQ14
//UART2_INT   IRQ15
//****DUMMY  IRQ16
//I2C_INT     IRQ17
//****DUMMY  IRQ18
//SPI_INT     IRQ19
//SIO_INT     IRQ20
//EXI2_INT    IRQ21
//EXI3_INT    IRQ22
//EXI4_INT    IRQ23
//CA_INT      IRQ24
//TKEY_INT    IRQ25
//LPT_INT     IRQ26
//****DUMMY  IRQ27
//BT0_INT     IRQ28
//BT1_INT     IRQ29
//ReturnValue:None
/*-----*/
void SYSCON_INT_Priority(void)
{
    INTC_IPR0_WRITE(0X40404040); //IQR0-3
}
```

```
INTC_IPR1_WRITE(0X40404000); //IQR4-7
INTC_IPR2_WRITE(0X40404040); //IQR8-11
INTC_IPR3_WRITE(0X40404040); //IQR12-15
//INTC_IPR4_WRITE(0X40404040); //IQR16-19

INTC_IPR4_WRITE(0X40004040); //IQR16-19
INTC_IPR5_WRITE(0X40404040); //IQR20-23
INTC_IPR6_WRITE(0X40404040); //IQR24-27
INTC_IPR7_WRITE(0X40404040); //IQR28-31

}
```

● 测试：

I2C 从机测试,中断中接收数据

```
/*-----*/
//I2C Interrupt
//EntryParameter:NONE
//ReturnValue:NONE
/*-----*/
void I2CIntHandler(void)
{
    // ISR content ...
    I2C_Slave_Receive();
}
}
```

4. 程序下载和运行

1. 将目标板与仿真器连接，分别为 VDD SCLK SWIO GND
2. 将主机 I2C 功能脚与对应的从机设备进行连接。
3. 程序编译后仿真运行
4. 写入数据，并读出进行查看验证。