

APT32F101 I2C 应用笔记

2017.11

Author: Du Xianye



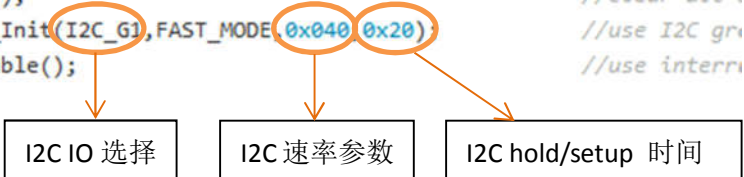
Revision History

NO	日期	描述
1	2017-11	新建

1. I²C 设置

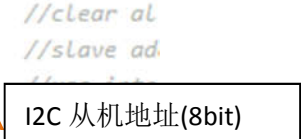
I2C 主机设置

```
void I2C_MASTER_CONFIG(void)
{
    I2C_DeInit();
    I2C_Master_Init(I2C_G1, FAST_MODE, 0x040, 0x20);
    I2C_Int_Enable();
}
```



I2C 从机设置

```
void I2C_SLAVE_CONFIG(void)
{
    I2C_DeInit();
    I2C_Slave_Init(I2C_G2, FAST_MODE, 0x4A, 0x20, 0xac);
    I2C_Int_Enable();
}
```



I2C_IO	SDA	SCL	备注
I2C_G0	PA0.12	PA0.13	与调试口共用，配置后无法直接用 CDK 调试
I2C_G1	PA0.2	PA0.1	
I2C_G2	PA0.5	PA0.4	

表 1. I2C IO 口配置

I2C 速度计算

I2C 速度=PCLK/(PRV+4)

e.g: 20Mhz/(0x040+4)=353kHz

I2C hold/setup time=Htime*PCLK

e.g:0x20*0.0417us(1/24Mhz)=1.333us

2. I²C 配置注意事项

- 在配置时主从机的配置参数基本相同。需要注意的是上图中的 0xac 为从机地址，主机地址在 apt32f101_i2c.h 中定义 I2C_Address 的值。
- 在速度配置时需要特别注意，在 FAST_MODE 时，通讯速度不仅与 PRV 的设置值有关，还与上拉电阻大小以及线上的寄生电容相关。所以当 PRV 配置得较小时，通讯若失败与另外两个参数有极大关系。

上拉电阻计算公式：

最小值 $R_{min} = \{V_{dd(min)} - 0.4V\} / 3mA$

最大值 $R_{max} = (T / 0.874) * C$, $T = 1\mu s$ 100KHz, $T = 0.3\mu s$ 400KHz

C 是寄生电容，100K 时最大容限 400pF，400K 时最大容限 200pF

3. 使用从机时,必须打开中断,并使能apt32f101_interrupt.c中的I2C_Slave_Receive();函数。此函数默认是屏蔽的,使用时需手动打开。
4. V1.20 开始的版本支持从机模式 I2C 的连续读写,以前的版本只能单次读写

3. I²C 使用方法

主机的使用

1. 硬件的写函数 I2C_WriteByte_Int; 硬件的读函数 I2C_ReadByte_Int

从机的使用

1. 从机的接收函数在 I2C_Slave_Receive()中
2. 由主机写入的地址及数据存储在 I2CRdBuffer[BUFSIZE]中;传送给主机的数据存储在 I2CWrrBuffer[BUFSIZE]中, I2CWrrBuffer[BUFSIZE]中的数据需要在从机程序中更新;
3. BUFSIZE 的值决定了从机寄存器的个数,范围是 0~255
4. 主机写入数据时,是向 I2CRdBuffer[BUFSIZE]写入数据。BUFSIZE 的值即代表操作的寄存器。
eg.主机向从机寄存器 0x20 写入数据 0x55,操作完成后 I2CRdBuffer[0x20]=0x55;
5. 主机读取值时,则是从 I2CWrrBuffer[BUFSIZE]读取数据。所以读取值寄存器值与 I2CWrrBuffer[BUFSIZE]需要一一对应;
eg.读取从机 0x20 的值,及代表读取 I2CWrrBuffer[0x20]的值。在程序中需要将 I2CRdBuffer 的值赋值给对应的 I2CWrrBuffer。