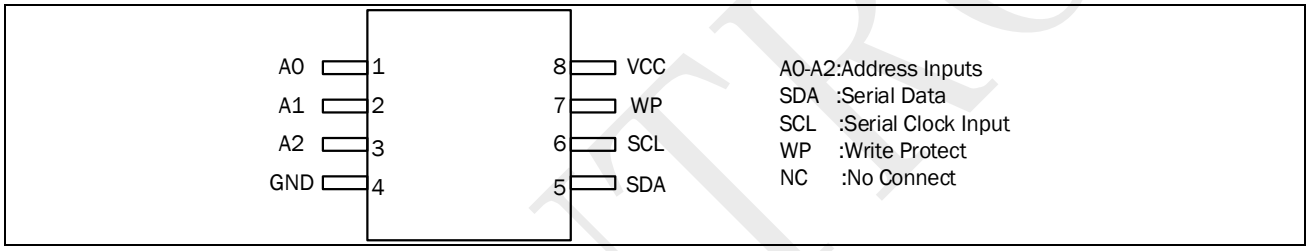
SPC2188 读写 AT24C02 EEPROM 使用指南

版本 A/0 – 2023 年 9 月

概述

AT24C02 是一个 2K Bit 的串行 EEPROM 存储器(掉电不丢失),内部含有 256 个字节。在 AT24C02 里面有一个 8 字节的页写缓冲器，具有 I2C 接口和用于硬件数据保护的写保护引脚（WP），最大写周期为 10ms，AT24C02 的引脚配置如图 1-1：所示。

图 1-1：24C02 引脚配置



目录

图片列表

表格列表

版本历史

版本	日期	作者	状态	变更
A/0	2023-09-01	X.He	Released	首次发布。

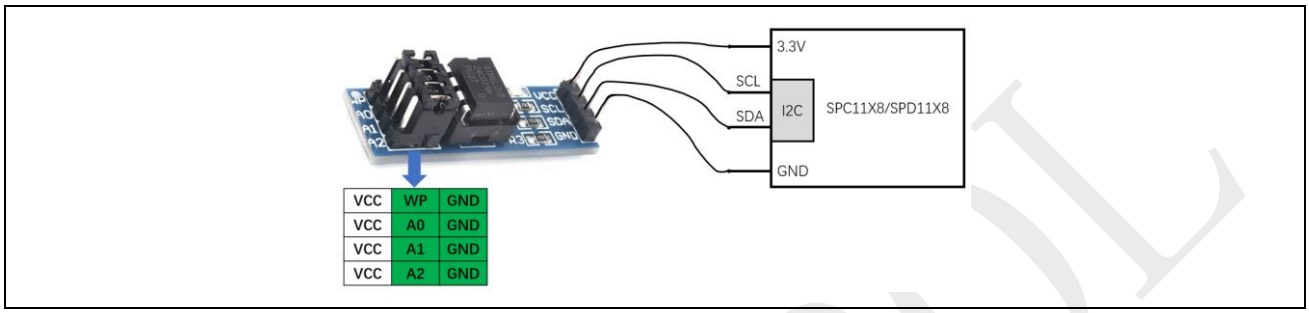
术语或缩写

术语或缩写	描述

1 使用方式简介

SPC2188 内置有 IIC 单元，可与 EEPROM（AT24C02）建立 I2C 通信，硬件连接示意图如图 1-1：硬件链接示意图所示。

图 1-1：硬件链接示意图



2 应用示例

2.1 AT24C02 单字节读写示例

按单字节读写 AT24C02 模块示例代码如下：

Example Code

```
#define AT24C02_START_ADDR      0x00
uint8_t au8WriteByteBuf[1] = {0xAB};
uint8_t au8ReadByteBuf[1] = {0x00};

int main()
{
    CLOCK_InitWithRCO(CLOCK_CPU_MAX);

    Delay_Init();

    /*
    * Init the UART
    *
    * 1.Set the GPIO34/35 as UART FUNC
    *
    * 2.Enable the UART CLK
    *
    * 3.Set the rest para
    */
    PIN_SetChannel(PIN_GPIO62, PIN_GPIO62_UART0_TXD);
    PIN_SetChannel(PIN_GPIO63, PIN_GPIO63_UART0_RXD);
    UART_Init(UART0, 38400);

    /* AT24C02 Init */
    AT24C02_Init();

    /* Read and write a byte data to AT24C02 */
    AT24C02_Byte_Test();

  while(1)
  {

  }
}
```

在 AT24C02_Byte_Test 测试函数中，对 AT24C02 进行单字节数据的读写和数据比对。

Example Code

```
ErrorStatus AT24C02_Byte_Test(void)
{
    ErrorStatus eStatus;
    EEPROM_INFO("Write a byte data: ");

    printf("0x%02X \n", au8WriteByteBuf[0]);

    /* Send one byte of data to 24C02 */
    eStatus = AT24C02_WriteByte(EEPROM_PAGE_SIZE, au8WriteByteBuf[0]);
    if(eStatus == ERROR)
    {
        EEPROM_DEBUG("[Write Data ERROR]\n");
        return ERROR;
    }
    else
    {
        EEPROM_DEBUG("[Write Data SUCCESS]\n");
    }

    EEPROM_INFO("Read a byte data: ");

    /* Read one byte of data from AT24C02 */
    AT24C02_ReadByte(EEPROM_PAGE_SIZE, au8ReadByteBuf);

    printf("0x%02X \n", au8ReadByteBuf[0]);

    /*  To check the data sent and recieved are the same */
    if(au8ReadByteBuf[0] != au8WriteByteBuf[0])
    {
        EEPROM_INFO("ERROR: AT24C02 inconsistent data was written and read");
        return ERROR;
    }
    else
    {
        EEPROM_INFO("AT24C02 read and write a byte test successful\n");
        return SUCCESS;
    }
}
```

2.2 AT24C02 多字节读写示例

按多字节读写 AT24C02 模块示例代码如下：

Example Code

```
#define         AT24C02_START_ADDR   0x00
uint8_t         au8WriteBuf[256];
uint8_t         au8ReadBuf[256];

int main()
{
    CLOCK_InitWithRCO(CLOCK_CPU_MAX);

    Delay_Init();

    /*
    * Init the UART
    *
    * 1.Set the GPIO34/35 as UART FUNC
    *
    * 2.Enable the UART CLK
    *
    * 3.Set the rest para
    */
    PIN_SetChannel(PIN_GPIO62, PIN_GPIO62_UART0_TXD);
    PIN_SetChannel(PIN_GPIO63, PIN_GPIO63_UART0_RXD);
    UART_Init(UART0, 38400);

    /* AT24C02 Init */
    AT24C02_Init();

    /* Read and write multiple bytes data to AT24C02 */
    AT24C02_MultiByte_Test();

  while(1)
  {
  }
}
```

在 AT24C02_MultiByte_Test 测试函数中，对 AT24C02 进行多字节数据的读写和数据比对。

Example Code

```
ErrorStatus AT24C02_MultiByte_Test(void)
{
    uint16_t i;
    ErrorStatus eStatus;
    EEPROM_INFO("Write data:");

    /* Initialize au8WriteBuf and au8ReadBuf*/
    for ( i = 0; i < 256; i++ )
    {
        au8WriteBuf[i] = i;
        au8ReadBuf[i] = 0xFF;

        printf("0x%02X ", au8WriteBuf[i]);

        if(i % 16 == 15)
        {
            printf("\n\r");
        }
    }

    /* Write 256 bytes of data sequentially to AT24C02 */
    eStatus = AT24C02_Write(au8WriteBuf, AT24C02_START_ADDR, 256);
    if(eStatus == ERROR)
    {
        EEPROM_DEBUG("[Write Data Error]\n");
        return ERROR;
    }
    else
    {
        EEPROM_DEBUG("[Write Data Success]\n");
    }

    EEPROM_INFO("Read data:");

    /* Read 256 bytes of data sequentially from AT24C02 */
    AT24C02_Read(au8ReadBuf, AT24C02_START_ADDR, 256);

    for (i = 0; i < 256; i++)
    {
        if(au8ReadBuf[i] != au8WriteBuf[i])
        {
            printf("0x%02X ", au8ReadBuf[i]);
            EEPROM_INFO("ERROR: AT24C02 inconsistent data was written and read");

            return ERROR;
        }

        printf("0x%02X ", au8ReadBuf[i]);

        if(i % 16 == 15)
        {
            printf("\n\r");
        }
    }

    EEPROM_INFO("AT24C02 read and write all test successful\n");

    return SUCCESS;
}
```