GY7502 USB-SPI Adapter

产品使用说明书

产品型号：GY7502 USB-SPI Adapter
手册版本：V1.04

目　录

一、产品简介

1.1 性能与技术指标

1）USB 2.0 转 SPI 接口适配器，USB 总线供电，无需外部电源；

2）SPI 主机接口，Master 方式；

3）SPI 时钟频率最大 6MHz；

4）提供电源输出：+3.3v ,+5V

5）接口信号：SCK,MISO,MOSI,CS0,CS1,GND,+5V,+3.3V 以及 2 路 IO 口.

6）输出信号 3.3V TTL，输入 5VTTL 可承受。

7）提供 2 路片选信号，并可以编程实现 4 路从机片选；

8）读操作模式：支持连续随机地址读；

9）写操作模式：支持单字节写，以及页写模式（Page Write）；

10）提供 DLL 动态链接库，接口开发函数；

11）提供 Visual c++开发例程；

12）提供 SPI 工具软件 SPITools；

13）塑料外壳，尺寸：70*45*18mm；

14）工作温度：-40°C - +85°C

15) 此产品出厂前均经过了对最常见的 SPI 器件 X5045 的读写测试，利用 SPITools 软件。测试过的芯片如下：

E2PROM 芯片：X5045，M95160（2 字节地址）；

带 SPI 接口的光收发 CDR 芯片寄存器读写：VSC8239；

数模转换 D/A 芯片：AD5314；

12 典型应用

SPI 总线测试；

SPI 接口的元器件寄存器读写；

SPI 接口的 EEPROM 读写；

13 通信协议转换

USB 转 SPI 总线，接口转换。

14 产品销售清单

USB 转 SPI 适配器一台；

USB 连接线一根；

ISP 扁平接口线缆一根

光盘 1 张（包括 PC 驱动、接口函数、用户手册等）；

15 技术支持与服务

一年内免费维修更换。

Mail： yyd315@163.com

网址：www.glinker.cn

二、外形与接口描述

2.1 产品外形

图 1 产品外形

2.2 Pin 脚描述

对外接口为 10pin 的针式接口。

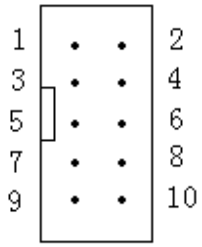


图 2 适配器对外接口（外壳）

表 1 适配器接口信号描述

引脚序号		描　　述
PIN1	GND	电源地与信号地
PIN2	SCK	时钟输出
PIN3	+5V　注 1	驱动电流 max 200mA
PIN4	MISO	主机输入，从机输出数据接口
PIN5	+3.3V　注 1	Normal 3.0-3.6V, 驱动电流 max 60mA
PIN6	MOSI	主机输出，从机输入数据接口
PIN7	CS0	片选信号 0
PIN8	IO-0	IO 口。默认为高电平。输出高 3.3V，输入 5V 可承受
PIN9	CS1	片选信号 1
PIN10	IO-1	IO 口。默认为高电平。输出高 3.3V，输入 5V 可承受

注 1：如果从机的电路上已有供电，则请不要使用适配器提供的电源。

因适配器提供电源信号的驱动电流比较小，请用户使用使谨慎评估，以防 PC 的 USB 接口损坏。

端口 IO 的直流电气特性如下：

表 14.1　端口 I/O 直流电气特性

VDD = 2.7V – 3.6V, -40℃到+85℃（除非特别说明）。

参　数	条　件	最小值	典型值	最大值	单 位
输出高电压（V_{OH}）	I_{OH} = -10μA，端口I/O为推挽方式 I_{OH} = -3mA，端口I/O为推挽方式 I_{OH} = -10mA，端口I/O为推挽方式	VDD-0.1 VDD-0.7	 VDD-0.8		V
输出低电压（V_{OL}）	I_{OL} = 10μA I_{OL} = 8.5mA I_{OL} = 25mA		 1.0	0.1 0.6	V
输入高电压（V_{IH}）		2.0			V
输入低电压（V_{IL}）				0.8	V

注：USB-SPI 适配器中，IO 输出采用的是推挽输出。

三、SPITools 软件

31 驱动程序安装

先将设备接入 PC 或笔记本电脑的 USB 接口，根据提示安装我们提供的驱动程序。

接入 USB-SPI 设备到 PC。

在“我的电脑”右键“属性”中选择设备管理器，可以看到

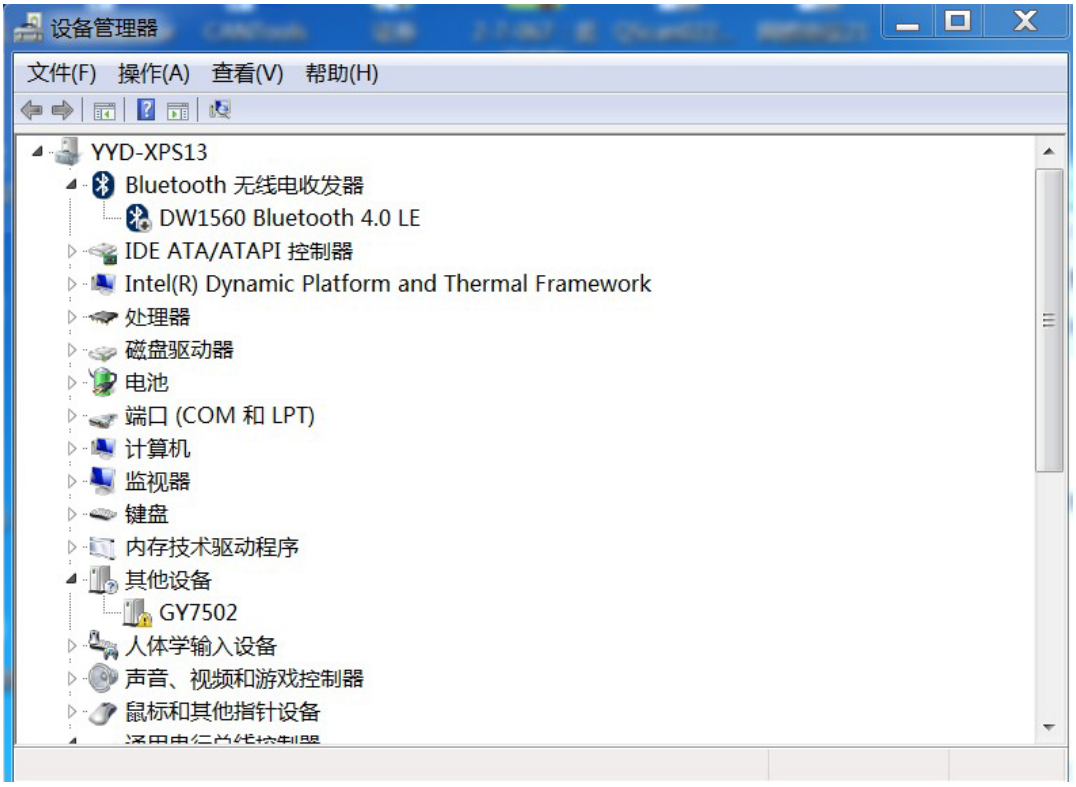


图 2 驱动安装过程

双击该 GY7502 设备，手动添加该设备的驱动程序，如下图，选择驱动程序所在的目录进行安装。过程中会提示该驱动程序是否确认安装，点确认认可。

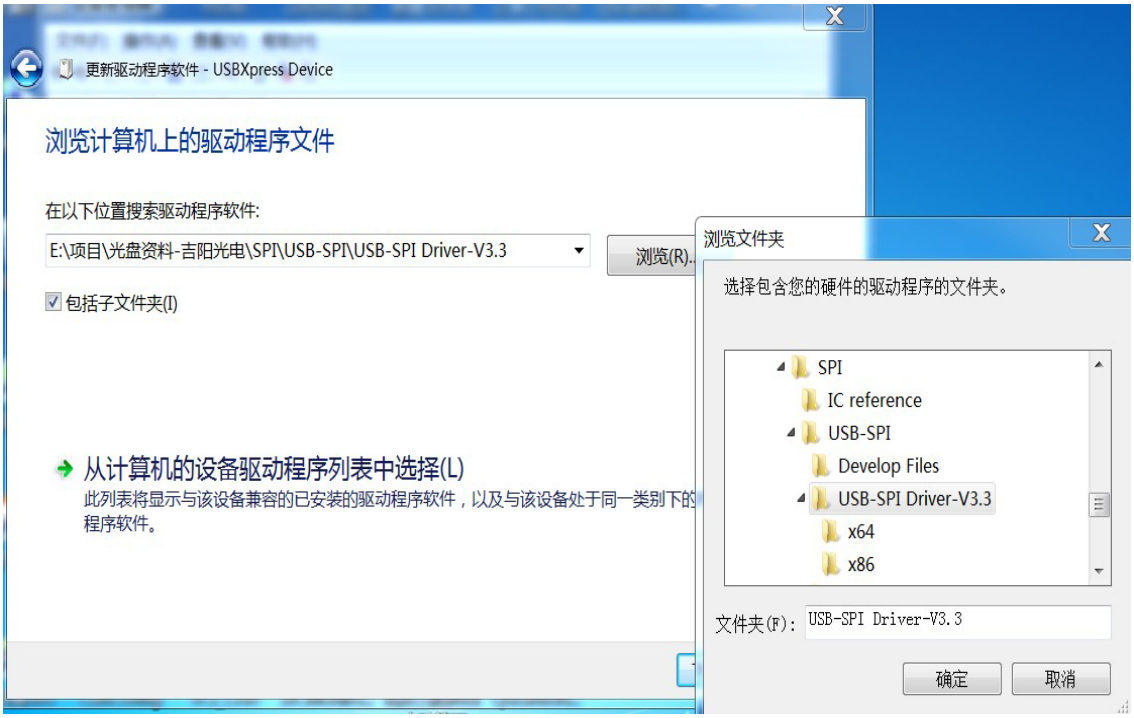


图 3 驱动安装过程

安装完成后，设备管理器中会指示有“USB-SPI Device”。

3.2 SPITools 软件安装

执行 SPITools_setup.exe ，按提示完成操作。

SPITools 主界面如下图：

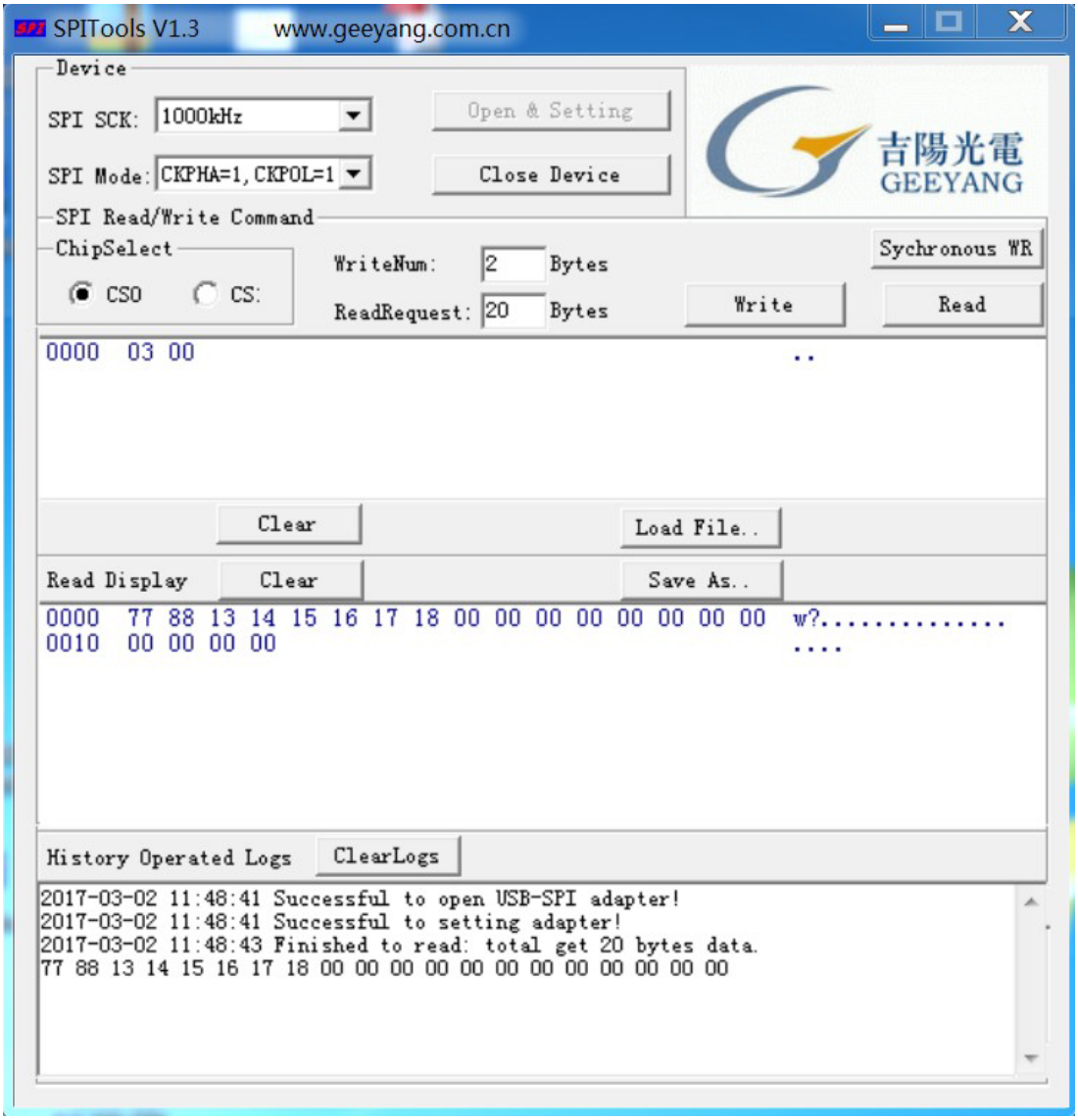


图 4　USB-SPITools 主界面

上图记录了测试 E2PROM 芯片 M95160 读操作的过程。

同时，我们为用户提供的例程中的应用程序也是一个很好的 SPI 工具软件，和上图这个软件功能基本一样,有源代码供参考。

用户也可以关注我们的网站，我们会不定期的对 SPITools 软件升级。

操作注意事项：如果要从 PC 中拔出设备，请先关闭 SPITools 软件，再拔出。否则重新插入时，会提示 USB 操作失败的信息。

33　软件功能介绍

1）按钮“Open&Setting”：将设备打开，并配置选择的参数信息。
2）按钮“关闭设备”：关闭后，将不能操作该设备。
3) 按钮“读操作”：对 SPI 从设备的读操作一般都先要写入读命令字(可能还包括地址)，然后再读数据。成功后返回数据。
4）按钮“写操作”，将 buffer 中的数据依次通过 SPI 接口发出去。完成后会有状态返回。
5）buffer 编辑框是一个 Hex 和 ASCII 码能同时编辑显示的输入控件。内容为将要写入或者读取到的数据。
6）按钮“LoadFile”：只能导入 bin 格式的二进制码文件，内容将依次存入缓冲区。
7）按钮“SaveFile”：将依次存入缓冲区的数据存入到文件中，文件类型 bin 格式。
8）History operated logs：记录了用户的操作过程和状态，以方便用户查看。注：该编辑框有容量限制，用户应及时清空，否则可能会没有新内容进来。

四、接口函数与用户编程

4.1 数据类型定义

4.1.1 SPI 读写数据类型

```
typedef struct GY7502_DATA_INFO{
        BYTE Databuffer[256]; //地址和数据；
        UINT WriteNum;        //地址和数据的总个数
        UINT ReadNum;         //需要读的数据个数
        BYTE IoSel; //IO 口选择，低2位分别对应2路IO口。bit 值为1 表示将该口将被操作。
        BYTE IoData; //IO 口状态，只有与IoSel 位为1 的对应值有效。
        BYTE ChipSelect;      /片选输出选择。值：0 或者 1
        BYTE Reserved[4];     //Reserved 系统保留。
        }GY7502_DATA_INFO,*pGY7502_DATA_INFO;
```

4.1.2 配置信息数据类型

```
typedef struct GY7502_CONFIG_INFO{
```

```
BYTE kFreq;             //SPI 时钟频率定义，
                        //0-5 依次代表 100kHz,200kHz,500kHz,1000kHz,2000kHz,6000kHz
BYTE SpiMode;           //工作模式 CKPOL,CKPHA 定义
BYTE Reserved[4];       //Reserved  系统保留。
}GY7502_CONFIG_INFO,*pGY7502_CONFIG_INFO;
```

4.1.3 时钟频率对照表：

表 2 SPI 时钟频率设置

kFreq 值	表示
0	100KHz
1	200KHz
2	500KHz
3	1000KHz
4	2000KHz
5	6000KHz

4.1.4 SPI 模式 SpiMode

此参数涉及 SPI timing 定义。

表 3 SPI 工作模式设置

SpiMode Value	描述
0	CKPHA=0, CKPOL=0
1	CKPHA=0, CKPOL=1
2	CKPHA=1, CKPOL=0
3	CKPHA=1, CKPOL=1

CKPHA 表示在 SCK 的第几个边沿采样。0 表示第一个边沿，1 表示第二个边沿。

CKPOL 表示 SCK 空闲时候的电平状态。0 表示低电平，1 表示高电平。

数据/时钟时序模式如下图 5 所示。

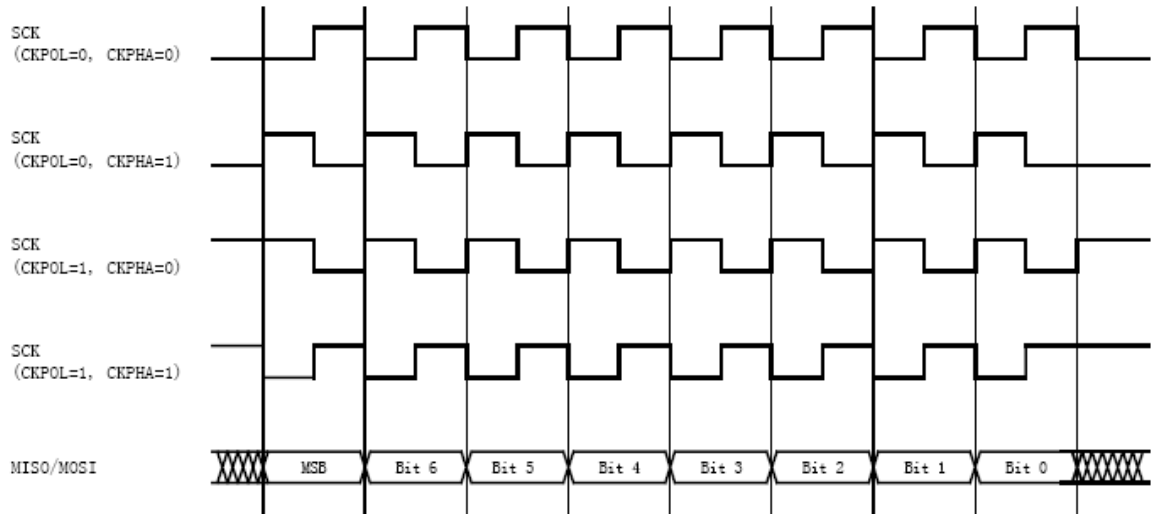


图 5　数据/时钟时序模式示意图

4.2 接口函数描述

4.2.1 打开设备

DWORD GY7502_USBSPI_Open()

返回值：1 表示成功，

0 表示失败。

4.2.2 关闭设备

DWORD GY7502_USBSPI_Close()

返回值：1 表示成功,

0 表示失败。

4.2.3 设备工作参数设置

DWORD GY7502_USBSPI_SetConfig(pGY7502_CONFIG_INFO pConfigInfo)

参数：　pConfigInfo-> kFreq

pConfigInfo-> SpiMode

返回值：1 表示成功，

0 表示失败，

-1 表示设备未打开。

4.2.4 读取当前参数

DWORD GY7502_USBSPI_GetConfig(pGY7502_CONFIG_INFO pConfigInfo)

参数：　pConfigInfo-> kFreq

pConfigInfo-> SpiMode

返回值：1 表示成功，当前参数保存在 pConfigInfo 中，

0 表示失败，

-1 表示设备未打开。

4.2.5 SPI 读操作

DWORD GY7502_USBSPI_Read(pGY7502_DATA_INFO pDataInfo)

参数：　pDataInfo->Databuffer[256]

pDataInfo->ChipSelect

pDataInfo->WriteNum

pDataInfo->ReadNum

工作过程：先将 pDataInfo->Databuffer 的内容向 SPI 从机写出，长度为 pDataInfo->WriteNum，之后紧接着从 SPI 总线读取 pDataInfo->ReadNum 个字节。

返回值：返回读到的总字节长度

0 表示失败，

-1 表示设备未打开。

4.2.6 SPI 写操作

DWORD GY7502_USBSPI_Write(pGY7502_DATA_INFO pDataInfo)

参数：　pDataInfo->Databuffer[256]

pDataInfo->ChipSelect

pDataInfo->WriteNum　　//每次最多只能写入 60 字节。

工作过程：将 pDataInfo->Databuffer 的内容向 SPI 从机写出，长度为 pDataInfo->WriteNum

返回值：1 表示成功，读取的结果将保存在 pConfigInfo-> Databuffer 中

0 表示失败，

-1 表示设备未打开。

注：写 I2C 操作时，如果数据个数大于 1 时，是一个连续写过程。数据依次被通过 MOSI 信号发出去。一般的 SPI 接口从器件都支持。如果从设备不支持连续写，则用户可以分开逐个字节的写。

4.2.6 SPI 同步写读操作

DWORD GY7502_USBSPI_WriteRead(pGY7502_DATA_INFO pDataInfo)

参数：　pDataInfo->Databuffer[256]

pDataInfo->ChipSelect

pDataInfo->WriteNum

pDataInfo->ReadNum

工作过程：将 pDataInfo->Databuffer 的内容向 SPI 从机写出，长度为 pDataInfo->WriteNum。同时读取 MISO 信号线上的数据并返回

返回值：返回读到的总字节长度，读取的结果将保存在 pConfigInfo-> Databuffer 中。

0 表示失败，

-1 表示设备未打开。

4.2.7　IO 接口设置

低 2 位分别代表 2 路 IO 口。

DWORD GY7502_USBSPI_SetIO(pGY7502_DATA_INFO pDataInfo)

参数：pDataInfo->IoData （低 2 位有效）

pDataInfo->IoSel　(IO口选择，为1表示被选择)

返回值：1 表示成功，

0 表示失败，

-1 表示设备未打开。

4.2.8　IO 接口状态读取

低 2 位分别代表 2 路 IO 口。

DWORD GY7502_USBSPI_GetIO(pGY7502_DATA_INFO pDataInfo)

参数：pDataInfo->IoData （低 2 位有效）

pDataInfo->IoSel　(IO口选择，为1表示被选择)

返回值：1 表示成功，返回值将保存在 pConfigInfo-> IoData 中

0 表示失败，

-1 表示设备未打开。

五、SPI 接口与时序

5.1 SPI 接口 Master--Slave 连接示意图

下图中 NSS 是片选信号。

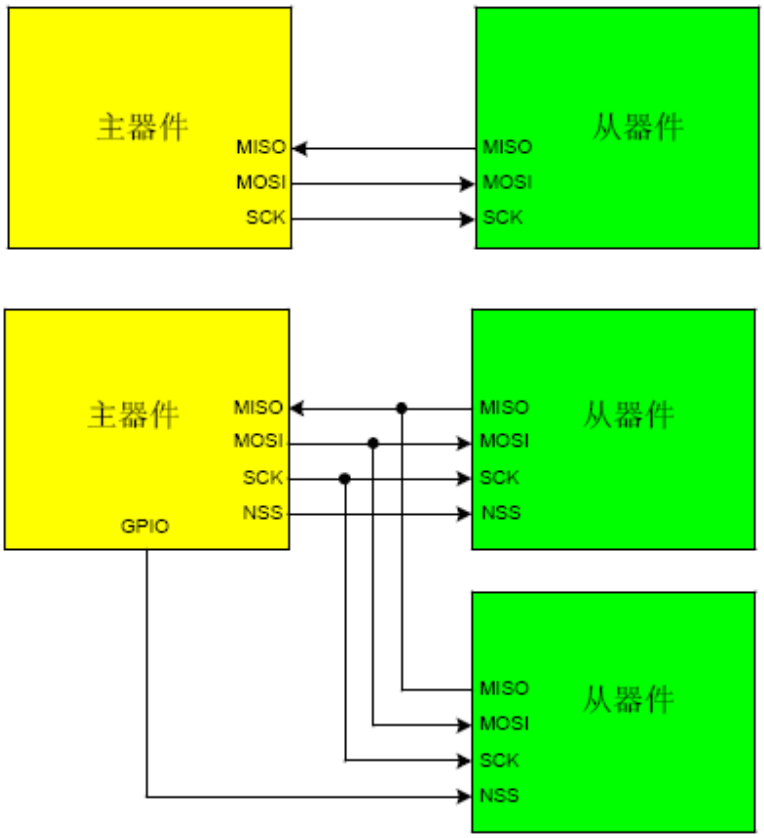
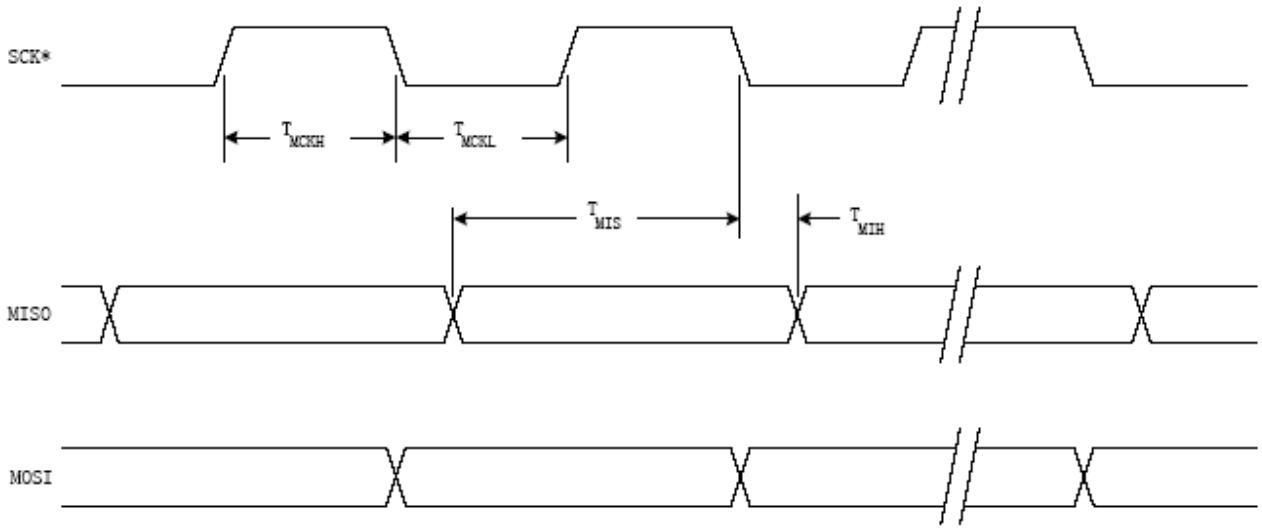


图 6　SPI 主从机连接示意图

52 SPI 接口时序

以下给出了本适配器的 SPI 时序图以及时序参数。



* 这是对应 CKPOL = 0时的 SCK 波形。对于 CKPOL = 1，SCK波形的极性反向。

图 7　SPI 主方式时序图（CKPHA=0）

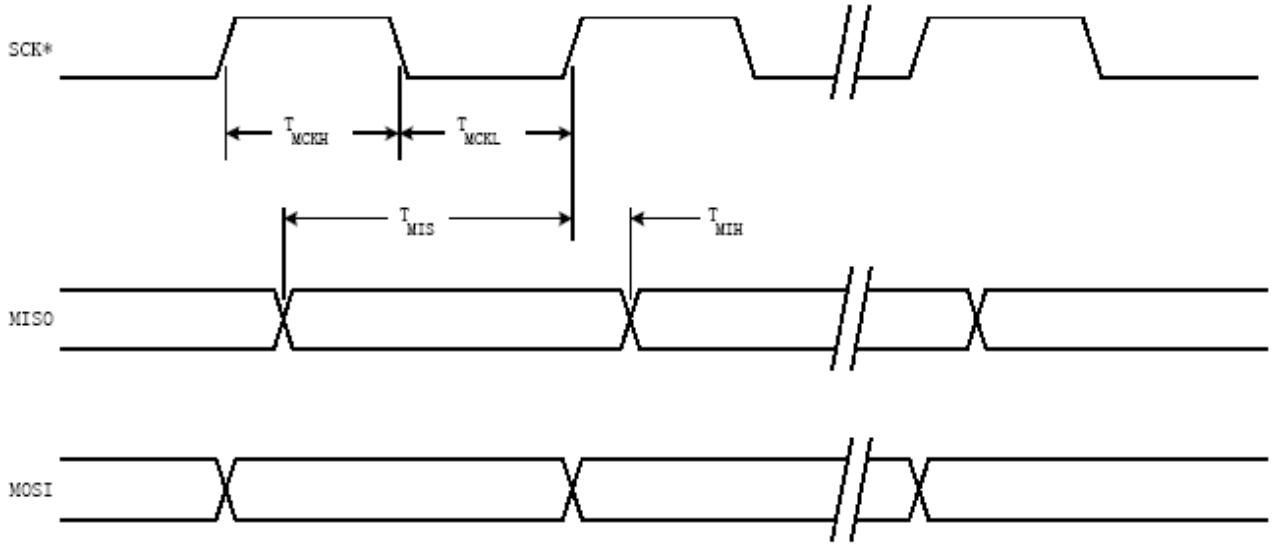


图 8　SPI 主方式时序图（CKPHA=1）

表 4　Adapter SPI 时序参数表

参 数	说 明	最小值	最大值	单位
T_{MIS}	MISO有效到SCK移位边沿	30	\	ns
T_{MIH}	SCK移位边沿到MISO发生改变	0	\	ns

六、SPI 读写示例

下面给出了通过 GY7502 USB-SPI Adapter，利用 SPITools 软件访问 SPI 接口芯片的例子，其他 SPI 接口器件请参考其 datasheet。

6.1 读写芯片的简易方法

如果在用户的从机板上未设计 10pin 的接口，则可以将我们提供的 ISP 扁平线缆剪断，将相应信号线直接焊接到芯片引脚上即可访问。特别提醒用户谨慎使用适配器提供的电源线。如果仅仅是对芯片供电，是没有问题的。如果向电路板供电，一定要考虑总的电流负载不要超过我们定义的最大电流限制，以防损坏 PC 的 USB 接口。

6.2 对 X5045 的写和读示例

首先设置 SPI Mode 为 3 或 0 都可以（根据 X5045 的 SPI 时序）。

1） 写使能：

buf[0]=0x06 //写使能命令

WriteNum=1 //写入字节数

ReadRequest=0//不读

执行“Write”

2） 从地址 0 开始写 16 个字节：

buf[0]=0x02 //写命令

buf[1]=0x00 //ROM 地址

buf[2]=0x10 //数据 0

buf[3]=0x11……

buf[17]=0x1F //数据 15

WriteNum=18 //共需写入以上 18 个字节

ReadRequest=0 //不读

执行“Write”，数据即会被写入

3） 从地址 0 开始读 16 个字节：

buf[0]=0x03 //读命令

buf[1]=0x00 //ROM 地址

WriteNum=2 //共需写入以上 2 个字节

ReadRequest=16 //读 16 次，将数据读出

执行“Read”，数据即会被读出

6.3 对 M95160 的写和读示例

首先设置 SPI Mode 为 3（根据 M95160 的 SPI 时序）。

1）写使能：

buf[0]=0x06 //写使能命令

WriteNum=1 //写入字节数

ReadRequest=0//不读

执行“Write”

2）从地址 0 开始写 16 个字节：

buf[0]=0x02　//写命令

buf[1]=0x00　//ROM 地址高位字节

buf[2]=0x00　//ROM 地址低位字节

buf[3]=0x10　//数据 0

buf[4]=0x11……

buf[18]=0x1F　//数据 15

WriteNum=19 //共需写入以上 19 个字节

ReadRequest=0 //不读

执行“Write”，数据即会被写入

3） 从地址 0 开始读 16 个字节：

buf[0]=0x03　//读命令

buf[1]=0x00　//ROM 地址高位字节

buf[2]=0x00　//ROM 地址低位字节

WriteNum=3　//共需写入以上 3 个字节

ReadRequest=16 //读 16 次，将数据读出

执行“Read”，数据即会被读出

6.4 对 VSC8239 寄存器的读写示例

VSC8239 共有 128 字节寄存器，地址 0－127。

首先设置 SPI Mode 为 2（根据 VSC8239 的 SPI 时序）。

1）将地址 0x12 写入新值，并读出：

buf[0]=0x25　//地址在高 7 位，bit0=1 表示写命令

0x12<<1 即 0x24, buf[0]=0x24 |0x01 =0x25

buf[1]=0x55　//数据值

WriteNum=2　//共需写入以上 2 个字节

ReadRequest=0 //不读

执行“Write”，数据即会被写入

2） 从地址 0 开始读 128 个字节：

buf[0]=0x00　//REG 地址以及读命令，地址在高 7 位，bit0=0 表示读命令

0x00<<1 即 0x00, buf[0]=0x00& 0xFE =0x00

WriteNum=1　//共需写入以上 1 个字节

ReadRequest=128 //读 128 次，将数据读出

执行“Read”，所有 128 个寄存器值即会被读出

6.5 对 AD5314 D/A 输出的控制示例

AD5314 的 DA 控制是 16 个 bit。共有 DA 通道 4 个,CH0-CH3。16bit 的命令字 bit16－bit0 分别为 A1,A0,PD,LDAC,D9,D8,D7,D6,　D5,D4,D3,D2,D1,D0,NULL,NULL

首先设置 SPI Mode 为 2（根据 AD5314 的 SPI 时序）。假定基准源 1.225V。

1）将 CH2 输出 0.6V：(即(0.6/1.225)*1023=501,16 进制为 0x01F5)

buf[0]=0xA7　//高字节，0x01F5 >>6=0x07, A1=1,A0=0,PD=1,LADC=0

buf[1]=0xD4　//低字节，0x01F5 <<2= 0xD4, DA 的低 6 位在 bit7-bit2

WriteNum=2　//共需写入以上 2 个字节

ReadRequest=0 //不读

执行“Write”，DA 值即被输出。