

# **GY7632 RS232-SPI Adapter**

## **产品使用说明书**

产品型号：GY7632 RS232-SPI Adapter

手册版本：V1.03

# 目 录

一、产品简介.....	3
二、外形与接口描述.....	5
2.1 产品外形.....	5
2.2 RS232 引脚定义 DB9 .....	5
2.3 SPI 侧排线引脚定义.....	5
三、电气特性.....	6
四、串口波特率设置.....	6
五、软件操作指令及举例.....	7
5.1 SPI 操作.....	7
5.2 串口命令详解.....	7
5.2.1 选择 SPI 工作模式.....	8
5.2.2 获取 SPI 工作模式.....	8
5.2.3 选择 SPI 当前通道号(CS 选择).....	8
5.2.4 获取 SPI 当前通道号.....	9
5.2.5 设置当前 SPI 通道的 SPI 速率 .....	9
5.2.6 获取当前 SPI 通道的 SPI 速率 .....	9
5.2.7 Easy SPI 写操作 .....	9
5.2.8 Easy SPI 读操作 .....	10
5.2.9 SPI 完全读写操作.....	10
5.3 常见问题现象.....	10
六、SPI 接口与时序.....	11
6.1 SPI 接口 Master--Slave 连接示意图.....	11
6.2 SPI 接口时序.....	12
七、应用系统示意图.....	13
八、常见 SPI 从器件的读写方法 .....	13
8.1 E2PROM X5045 的写和读示例 .....	13
8.2 对 M95160 的写和读示例 .....	13
8.3 对 VSC8239 寄存器的读写示例.....	14
8.4 对 AD5314 D/A 输出的控制示例 .....	14

## 一、产品简介

### 1.1 性能与技术指标

- 1) RS232 串口转 SPI 总线接口，1 组 SPI 接口输出。
- 2) 标准的 SPI 主机接口和从机接口方式，Master 方式，可发起 SPI 读，发起 SPI 写。
- 3) 同时支持 SPI 从机模式，可进行从发送，从接收。接收 SPI 主机接口发来的数据，并转发到串口。
- 4) 串口 TXD 和 RXD 信号为 RS232 电平，DB9 接口，可与 PC 串口相连。
- 5) 电源输入：+5V
- 6) SPI 接口信号:SCK,MISO,MOSI,CS0,CS1
- 7) 输出信号 3.3V TTL，输入 5VTTL 可承受。
- 8) 串口速率硬件设置，支持 9600、119200、57600、115200bps
- 9) SPI 总线速率软件设置，最高速率 12MHz。
- 10) 傻瓜式 SPI 读写模式（Easy SPI）。
- 11) 支持通过串口软件指令控制 SPI 接口的读写操作，进行二次开发。

### 1.2 典型应用

- 1) 为电脑或主控板增加 SPI 总线接口；
- 2) 通过 RS232 串口进行 SPI 接口测试；
- 3) SPI 接口的元器件寄存器读写；
- 4) SPI 接口的 EEPROM 读写；
- 5) 适用于 PC 以及嵌入式系统的串口转 SPI 需求；

### 1.3 通信协议转换

RS232 串口与 SPI 总线接口转换。SPI 转串口，串口转 SPI。

### 1.4 产品销售清单

- RS232-SPI 适配器一只；  
串口线 1 根；  
光盘 1 张（包括用户手册，相关资料等）；

### 1.5 技术支持与服务

货到 10 日内，用户不满意，无条件退货。一年内免费维修更换。

Mail: [support315@glinker.cn](mailto:support315@glinker.cn)

网址: [www.glinker.cn](http://www.glinker.cn)

**1.6 SPI 适配器产品订购信息**

型号	名称	CS 通道	SPI Master		SPI Slave	
			Clock	缓冲区	Clock	缓冲区
GY7502	USB-SPI Adapter	2	6Mhz	主发送 520 bytes 主接收 256 bytes	不支持	不支持
GY7507	UART-SPI Module	1	12Mhz	主发送 1200 bytes 主接收 256 bytes	从发送 100kHz 从接收 2 Mhz	1200 Bytes
GY7508	RS232-SPI Module	1	12Mhz	主发送 1200 bytes 主接收 256 bytes	从发送 100kHz 从接收 2 Mhz	1200 Bytes
GY7632	RS232-SPI Adapter	2	12Mhz	主发送 1200 bytes 主接收 256 bytes	从发送 100kHz 从接收 2 Mhz	1200 Bytes

## 二、外形与接口描述

### 2.1 产品外形



图 2 GY7632 RS232-SPI 适配器外观图

外形尺寸：70×45mm

### 2.2 RS232 引脚定义 DB9

RS232 接口是一个 DB9 的母头 (Female)

DB9 引脚序号 (孔)	信号定义	描述
2	TXD	RS232 电平, 适配器串口发送
3	RXD	RS232 电平, 适配器串口接收
5	GND	信号地
1, 4, 6, 7, 8, 9	NULL	空

### 2.3 SPI 侧排线引脚定义

GY7602/GY7604 的电源与信号接口是一个 10 芯(5\*2)的排线。

排线接口上有箭头标识的是 1 脚。

引脚之间的间距为 2.54mm (100mil)

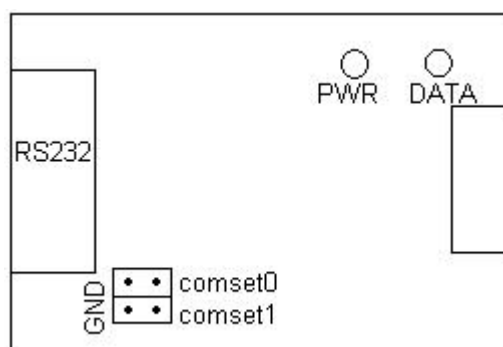
引脚序号	信号定义	描述
1	+5V	电源输入
2	GND	电源地/信号地
3	SCK	SPI 的输出时钟
4	MISO	数据端口, 主输入, 从输出
5	MOSI	数据端口主输出, 从输入
6	CS0	片选输出 0
7	CS1	片选输出 1

8	NULL	
9	NULL	
10	NULL	

### 三、电气特性

	Min	Normal	Max	备注
VDD	4.5V	5V	5.5V	电源输入
VOH	3.0V	3.25V	3.35V	输出
VOL	--	--	0.4V	
VIH	2.0V	---	5.5V	输入时, 5V 可承受
VIL	---	----	0.8V	

### 四、串口波特率设置



COMSET1	COMSET0	SPI 速率
0	0	9600
0	1	19200
1	0	57600
1	1	115200
备注：引脚悬空为高电平，1 状态		

出厂默认串口波特率 115200bps，如需要重新选择串口速率，可打开适配器外壳，COMSET0,COMSET1 信号与 GND 短接表示 0，悬空表示 1。默认为悬空状态。注：如非必要，建

议不要进行串口波特率设置。

## 五、软件操作指令及举例

对模块的参数设置以及读写 SPI 设备，均通过输入串口命令数据来进行。(电脑上可用串口调试助手等串口软件测试，8 个数据位，1 个停止位，无奇偶校验，串口波特率选被设置的值，如果 COMSET0,COMSET1 引脚悬空，则是 115200bps)

SPI 转换器上电后的默认参数：

工作模式：模式 3：CKPOL=1,CKPHA=1

SPI 通道号：0 号通道

SPI 时钟频率：200KHz

SPI 主从模式：默认为从模式。可自动接收外部设备发来的数据，并转发到串口

### 5.1 SPI 操作

在进行 SPI 读或写操作的时候，本设备自动转换到 SPI 主模式，操作完成后，则再切换为从模式。

#### 2) 工作于主模式

可直接通过命令读写数据，无须考虑去产生 SPI 的时序。

优点：该方式简单方便，快速。用户不需要了解 SPI 时序协议。

工作于主模式时，一次命令最多读出来的数据 256 个，一次最多写入的数据为 512~1200 个（包含命令字）。

工作于主模式的时候，SPI 时钟频率从 1k—12000kHz 可设置。（高速可支持 2M,4M,6M,8M,12M 可用）。可发起 SPI 读，写外部从设备。

工作过程：转换器/模块得到该命令以后，进行解析，然后启动内部的 SPI 读写控制时序，将上位机要求的操作完成以后，再将结果返回给上位机。

#### 2) 工作于从模式

工作于从模式的时候，SCK 时钟由外部的 SPI 主设备给出。模块处于被动响应状态。

当外设发起 SPI 读的时候，模块处于从发送状态，最大能响应的 SCK 频率为 100khz。如需要从发送功能，必须事先通过串口写入或设置 SPI 缓冲区列表。

当外设发起 SPI 写的时候，模块处于从接收状态。最大能响应的 SCK 频率为 2 Mhz。在 SPI 传输完成后，接收到的所有数据将被依次发送到串口。

### 5.2 串口命令详解

命令字	说明	备注
-----	----	----

<b>0x10</b>	设置 SPI 的时钟相位工作模式	模式类型 0~3
<b>0x11</b>	读取当前 SPI 的时钟相位工作模式	
<b>0x40</b>	选择当前的 CS 通道号	默认为 CS0
<b>0x41</b>	读取当前的 CS 通道号	
<b>0x42</b>	设置 SPI 时钟频率, 单位 KHZ	从模式时, 无效
<b>0x43</b>	读取当前 SPI 时钟频率, 单位 KHZ	
<b>0x44</b>	<b>SPI 写操作的命令头</b>	简洁操作方式
<b>0x45</b>	<b>SPI 读操作的命令头</b>	简洁操作方式
<b>0x46</b>	SPI 读写操作命令头	同步读写操作。
<b>0x48</b>	从模式下写 SPI 的缓冲区列表	通过串口事先写入, 当外部主设备发起 SPI 读的时候, 则发送该缓冲区列表数据。
<b>0x49</b>	从模式下读当前 SPI 的缓冲区列表	

(以下命令字和数据都为 16 进制表示)

### 5.2.1 选择 SPI 时钟相位工作模式

```
#define CMD_SET_MODE    0x10    //0~3
```

格式: 命令字 10 + 需要选择的 SPI 通道号

举例:

```
10 00    设置为模式 0, CKPOL=0,CKPHA=0
10 01    设置为模式 1, CKPOL=0,CKPHA=1
10 02    设置为模式 2, CKPOL=1,CKPHA=0
10 03    设置为模式 3, CKPOL=1,CKPHA=1
```

返回值: AA

### 5.2.2 获取 SPI 时钟相位工作模式

```
#define CMD_GET_MODE    0x11//查询当前 SPI 工作模式
```

格式: 命令字 11

返回值: 当前 SPI 工作模式

举例:

```
11        返回值 00        当前为模式 0
11        返回值 01        当前为模式 1
```

### 5.2.3 选择 SPI 当前通道号(CS 选择)

```
#define CMD_SET_CHANNEL 0x40    //选择当前 SPI 通道号
```

格式: 命令字 40 + 需要选择的 SPI 通道号

举例:

40 00            选择 CS0 号 SPI 通道作为当前通道

40 03            选择 CS3 号 SPI 通道作为当前通道

返回值: AA

默认设置: 如果不进行此设置, 则默认为 00

注: GY7508 只有 1 路 SPI 片选

#### 5.2.4 获取 SPI 当前的工作通道号

#define CMD\_GET\_CHANNEL 0x41    //查询当前 SPI 通道号

格式: 命令字 41

返回值: 当前工作的 SPI 通道号

举例:

41                返回值 01            当前工作的通道索引号为 01 , 即第 CS1 路 SPI 接口

#### 5.2.5 设置当前 SPI 通道的 SPI 速率

#define CMD\_SET\_CLKVALUE 0x42    //设置 SPI 时钟频率, 单位 KHZ

格式: 命令字 42 + 速率的高字节 + 速率的低字节

举例:

42 00 64            将当前 SPI 通道的速率设置为 0x0064 即 100khz

42 01 90            将当前 SPI 通道的速率设置为 0x0190 即 400khz

返回值: AA

默认设置: 如果不进行此设置, 则默认为 00 64, 即 100khz

#### 5.2.6 获取当前 SPI 通道的 SPI 速率

#define CMD\_GET\_CLKVALUE 0x43    //查询 SPI 时钟频率, 单位 KHZ

格式: 命令字 43 举例:

43        返回值 00 64            当前 SPI 通道的速率为 0x0064 即 100khz

43        返回值 01 90            当前 SPI 通道的速率为 0x0190 即 400khz

#### 5.2.7 Easy SPI 写操作

#define CMD\_WRITE\_DATA            0x44    //写命令字

格式:

命令字	命令字、地址、数据
44	待写入的数据内容

注: 一个命令帧的总长度最大为 1200 个字节

举例:

44 02 00 33 44    返回值 0xAA            依次写入地址 00, 数据 33,44。

                  返回值 0xBB            错误

注: 在 E2PROM 中, 02 表示“写命令”, 00 表示起始地址

### 5.2.8 Easy SPI 读操作

```
#define CMD_READ_DATA    0x45    //读命令字
```

格式:

命令字	命令字、待写地址等内容	长度（该命令帧的最后一个字节）
45	一般有 1-3 个字节	希望读的个数减 1

长度最大值为 1200

45 03 00 07 随机读（random read）

SPI 接口先写“读命令” 03 和地址 00，然后从发起读时序，要求读数据个数 8，正常会返回值 8 个所读到的数据。

返回值 0xBB 错误

### 5.2.9 SPI 同步读写操作

在此方式下，写入多少字节，则读出多少字节。

如果只需要读，则也要写入相应长度的无效数据。

Easy 读和 Easy 写方式可理解为是省略了部分无效数据的。

```
#define CMD_WRITE_DATA    0x46    //写命令字
```

格式:

命令字	命令字、地址、数据
46	待写入的数据内容

注：一个命令帧的总长度最大为 260 个字节

返回值：从 MISO 信号线读出的数据。读出的数据个数等于写入的数据个数。

举例:

46 03 00 00 00 00 00 00 00 00 00 依次写入 03 00，并读出 8 个字节的有效数据。

返回值 3F 3F 11 22 33 44 55 66 77 88

注：本例为读 EEPROM 的命令格式。返回的前两个字节是写 03 00 的时候 MISO 返回的内容，可被认为是无效数据，

## 5.3 常见问题现象

如果从串口发送指令后，无任何返回结果，请检查以下问题：

- 1) +5V 电源是否接入正确？电源质量是否可靠？
- 2) COMSET0,1 的设置，以及上位机的串口波特率是否设置正确？  
如果这两个脚都悬空，则默认串口波特率为 115200。
- 3) 串口 TXD, RXD 信号连接是否正确，有无接反？

如果返回结果或返回值有问题，则检查以下问题：

- 1) SPI 的 4 路信号从设备的连接是否正确和接触可靠？
- 2) 操作指令是否正确？从设备的地址是否输入正确？
- 3) 读写命令字和格式是否区分正确？

## 六、SPI 接口与时序

### 6.1 SPI 接口 Master--Slave 连接示意图

下图中 NSS 也是片选信号，与 CS0,CS1 同样意义。

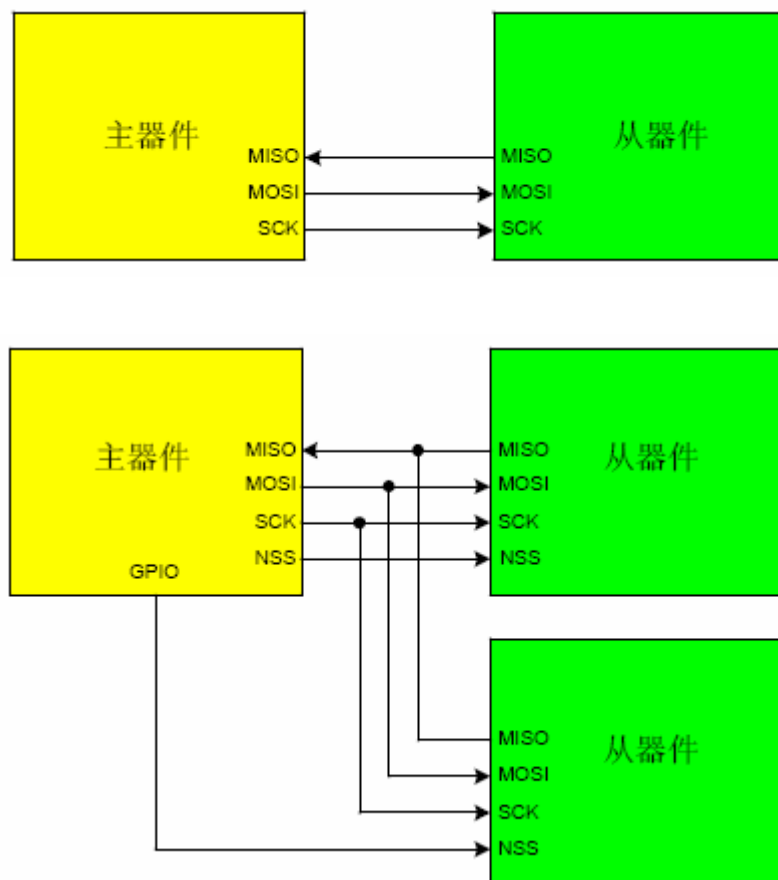
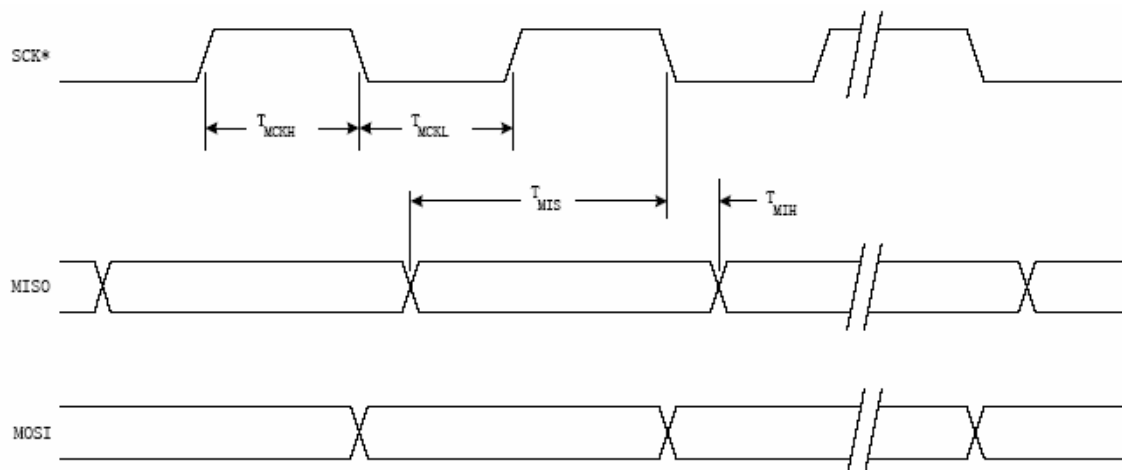


图 6 SPI 主从机连接示意图

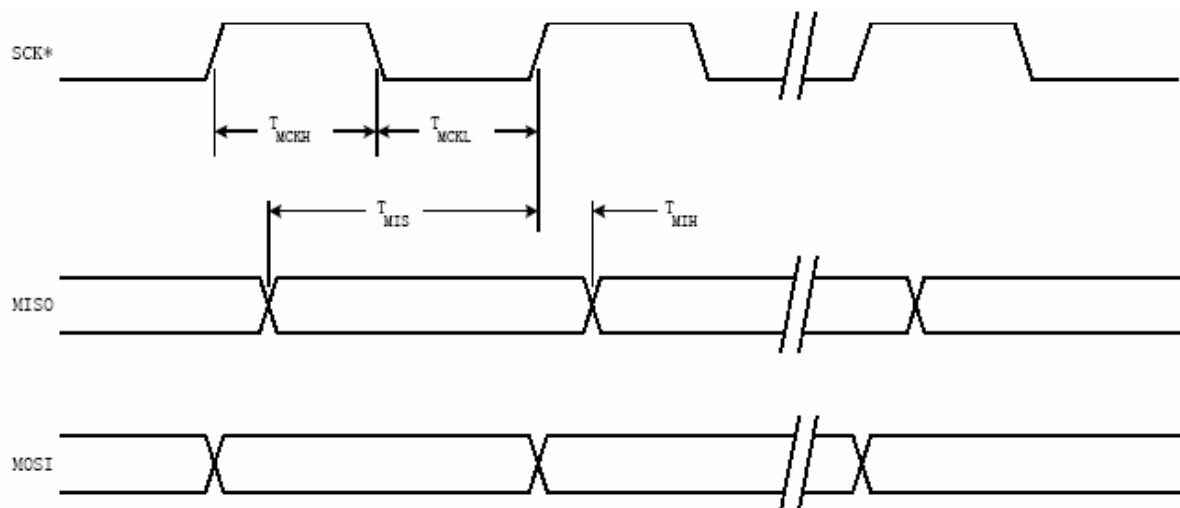
## 6.2 SPI 接口时序

以下给出了本适配器的 SPI 时序图及时序参数。



\* 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)



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

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
$T_{MCKH}$	SCK高电平时间	10	\	ns
$T_{MCKL}$	SCK低电平时间	10	\	ns
$T_{MIS}$	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

## 七、应用系统示意图

主控制器可以是电脑或者带 RS232 串口的单片机/嵌入式系统板卡。

SPI 从设备一般是带 SPI 接口的芯片或设备。

GY7508 是单路 RS232 转 SPI 接口模块。

GY7507 是单路 UART 转 SPI 接口模块。

GY7508 因封装较小，可以作为一颗芯片焊接或安装到用户的 PCB 板上。

GY7632 是 RS232 转 2 路 SPI 适配器。

系统的连接请参考下图：

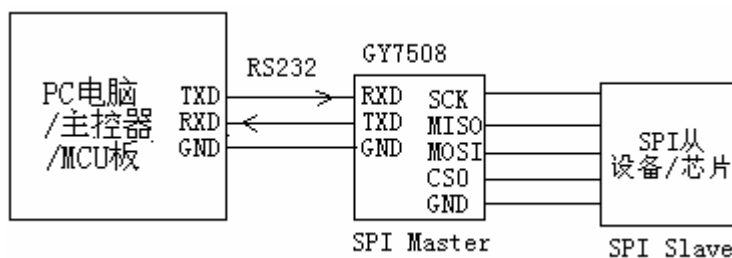


图 8.1 GY7508/GY7632 SPI 应用示意图

## 八、常见 SPI 从器件的读写方法

### 8.1 E2PROM 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：

44 06

2) 从地址 0 开始写 8 个字节：

44 02 00 11 12 13 14 15 16 17 18

3) 从地址 0 开始读 16 个字节：

45 03 00 0F

### 8.2 对 M95160 的写和读示例

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

内部 ROM 地址是 2 个字节

1) 写使能:

44 06

2) 从地址 0 开始写 8 个字节:

44 02 00 00 11 12 13 14 15 16 17 18

3) 从地址 0 开始读 16 个字节:

45 03 00 00 0F

### 8.3 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:

44 25 55

说明:

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

2) 从地址 0 开始读 128 个字节:

45 00 7F

说明: 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

### 8.4 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

命令字格式: 44 A7 D4