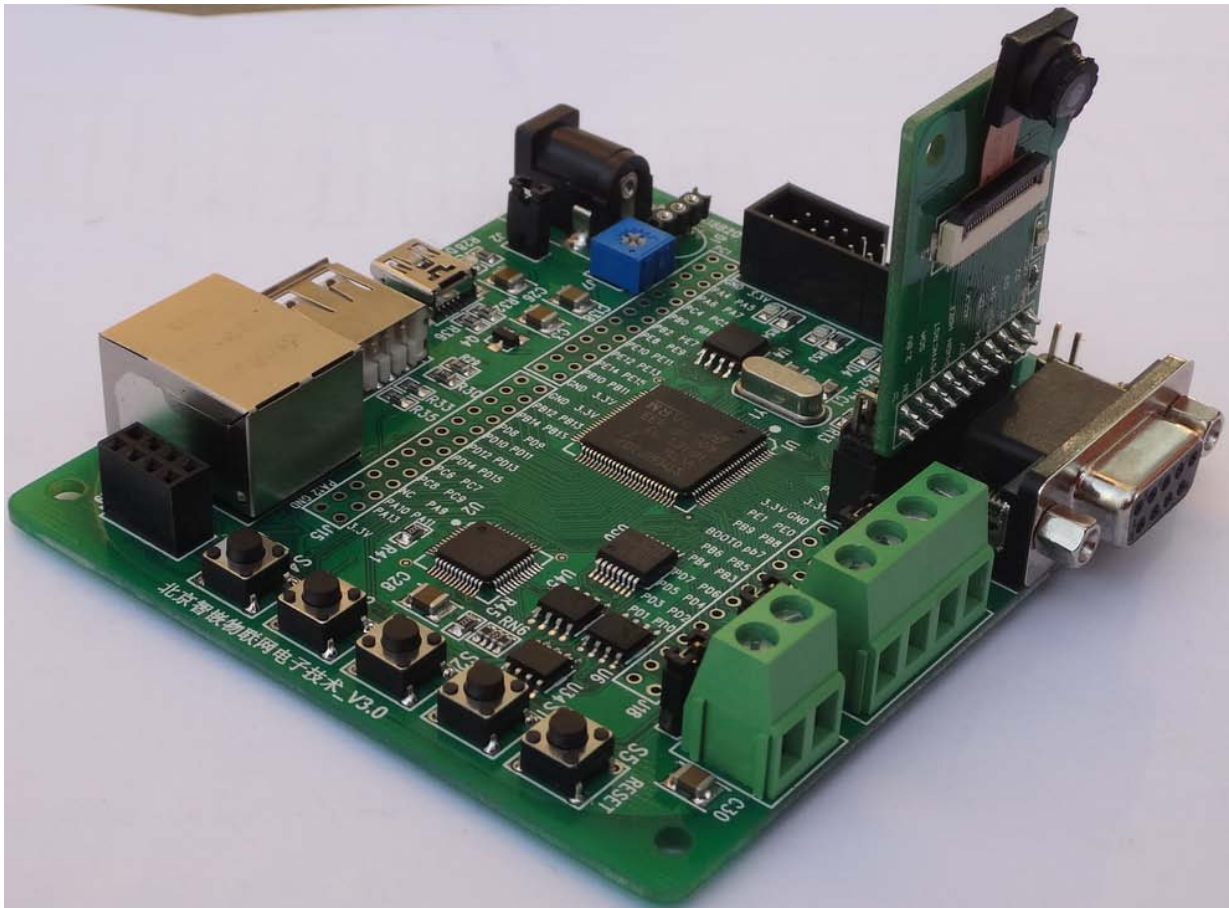


STM32F107 网络互联开发板 3.0 实验例程操作手册

版本号：A

拟制人：赵工

时 间：2013 年 7 月 1 日



目 录

| | | |
|------|--|----|
| 1 | 本文档编写目的 | 1 |
| 2 | 实验例程操作说明 | 1 |
| 2.1 | LED闪烁实验..... | 1 |
| 2.2 | KEY_LED实验 | 1 |
| 2.3 | RS232 通讯实验 | 1 |
| 2.4 | RS485 通讯实验 | 2 |
| 2.5 | CAN1 通讯实验..... | 2 |
| 2.6 | CAN1 与CAN2 通讯实验 | 3 |
| 2.7 | I2C实验—读写 24c04..... | 4 |
| 2.8 | SPI通讯实验--读写SST2508B..... | 5 |
| 2.9 | TCP服务器收发数据实验..... | 5 |
| 2.10 | 动态IP实验..... | 7 |
| 2.11 | HTTP网页服务器实验 | 7 |
| 2.12 | USB数据存储实验 1（读写STM32F107 片内flash） | 9 |
| 2.13 | USB数据存储实验 2（读写板载SST25F08B） | 9 |
| 2.14 | U盘读写实验..... | 9 |
| 2.15 | USB鼠标实验 | 11 |
| 2.16 | USB_JoyStickMouse实验..... | 11 |
| 2.17 | 2.4G无线模块NRF24I01 通信实验..... | 12 |
| 2.18 | 基于NRF24L01 的无线RS232 通讯实验..... | 12 |
| 2.19 | DS18B20 温度试验..... | 13 |
| 2.20 | UDP客户端发送数据实验 | 14 |
| 2.21 | UDP服务器收发数据实验 | 15 |
| 2.22 | TCP客户端收发数据实验..... | 18 |
| 2.23 | 16 通道ADC采集实验..... | 21 |
| 2.24 | UDP与RS232RS485 双向数据透明传输例程---开发板做服务器..... | 23 |
| 2.25 | TCP与RS232RS485 双向数据透明传输实验 | 23 |
| 2.26 | TCP与CAN双向数据透明传输实验 | 23 |

1 本文档编写目的

本手册是针对 STM32F107 网络互联开发板 V2.2 的例程而编写的，包括每个实验例程的实验原理、实验步骤、注意事项等。

2 实验例程操作说明

2.1 LED 闪烁实验

实验原理：通过控制 PE2/PE3/PE4/PE5 的电平变化实现 LED 灯的亮灭。

实验步骤：

- (1) 用 Keil uVision4 打开 LED 灯闪烁实验工程，并编译。
- (2) 给开发板上电，将生成的目标文件（HEX 文件）下载到单片机（可通过 JTAG 或串口下载）。
- (3) 按下“RESET”键（即 S5），则可以看到 4 个 LED 灯闪烁。

2.2 KEY_LED 实验

实验原理：单片机先读取按键的状态，然后根据读到的状态控制相应的 LED 灯（4 个按键和 4 个 LED 一一对应）。

实验步骤：

- (1) 用 Keil uVision4 打开 KEY_LED 实验工程，并编译。
- (2) 给开发板上电，将生成的目标文件（HEX 文件）下载到单片机（可通过 JTAG 或串口下载）。
- (3) 按下“RESET”键（即 S5），单片机复位后，再按下 4 个用户按键其中之一，则会看到相应的 LED 亮，松开后 LED 熄灭。

2.3 RS232 通讯实验

实验原理：PC 机通过“sscomv20(串口调试程序)”软件向开发板发送数据，开发板接收到一帧数据后（一帧数据以‘?’作为结尾），将该帧数据原样返回。

实验步骤：

- (1) 用串口线（usb 转串口线）将开发板和电脑相连接。打开电脑的设备管理器，查看使用的端口号。
- (2) 打开“sscomv20(串口调试程序)”软件。
- (3) 将端口号设置为第（1）步查到的端口号，波特率：115200，数据位：8，校验位：Even,偶校验。停止位：1。如图 1 所示：

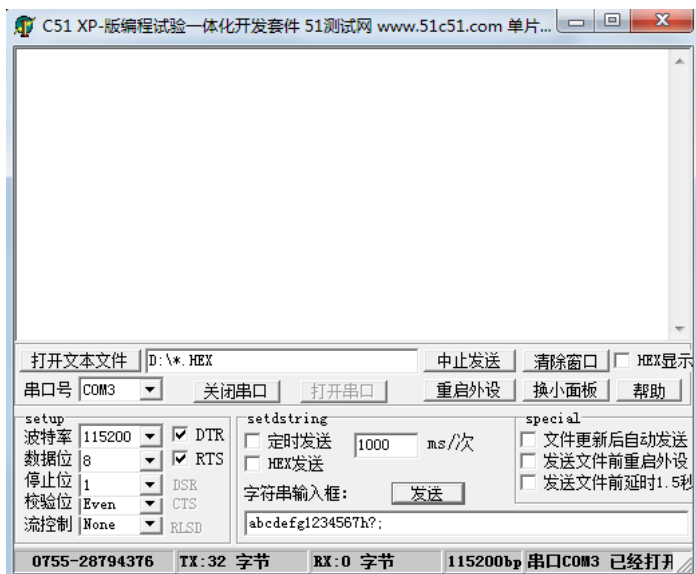


图 1 串口设置

- (4) 打开 RS232 通讯实验工程，编译，运行；并将生成的目标文件（HEX 文件）下载到单片机。

复位单片机，在串口调试软件的字符串输入框中输入：12345?;（注意?;是英文格式的）。点击发送，则会看到开发板发来的数据，如图 2 所示：

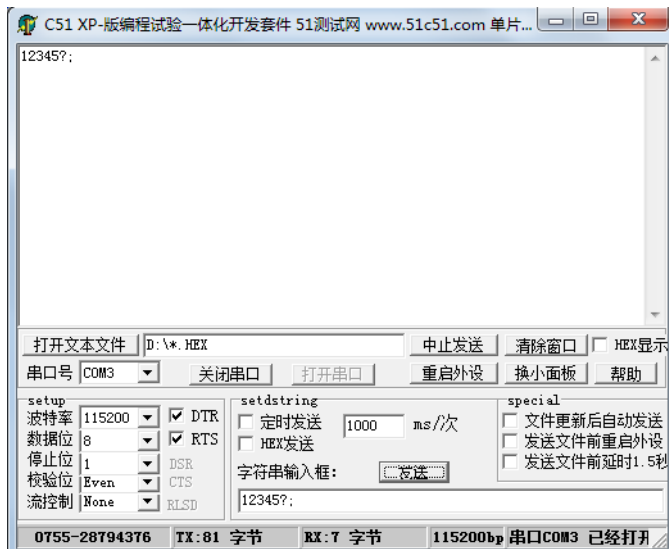


图 2 实验结果

2.4 RS485 通讯实验

实验原理：PC 机通过“sscomv20(串口调试程序)”软件向开发板发送数据，开发板接收到一帧数据后（一帧数据以'?;'作为结尾），将该帧数据原样返回。

实验步骤

- (1) 用 RS485/RS232 转换器将开发板和电脑相连接。打开电脑的设备管理器，查看使用的端口号。
- (2) 打开“sscomv20(串口调试程序)”，设置如图 1 所示。
- (3) 用 JLINK 将开发板和电脑连接，打开 RS485 通讯实验工程，编译，运行。
- (4) 在字符串输入框中输入：12345?; 点击发送，则会看到开发板发送来的数据，如图 2 所示。

2.5 CAN1 通讯实验

实验原理：CAN1 以 ID 为 0X18412345 的扩展帧向外发送数据，波特率为 250KBS.用 usb 转 can 设备可以看到板子发来的数据。

实验步骤：

- (1) 用 USB/CAN 转换器（本实例为 CAN_qm999cn）将开发板和电脑连接。
- (2) 打开 CAN_qm999cn.EXE 软件，设置如图 3 所示：



图 3 CAN 软件设置

(3) 将 can1 通讯工程打开，连接 jlink，并下载运行，点击图 3 中的“连接设备”则可以看到板子发来的数据，如图 4 所示。

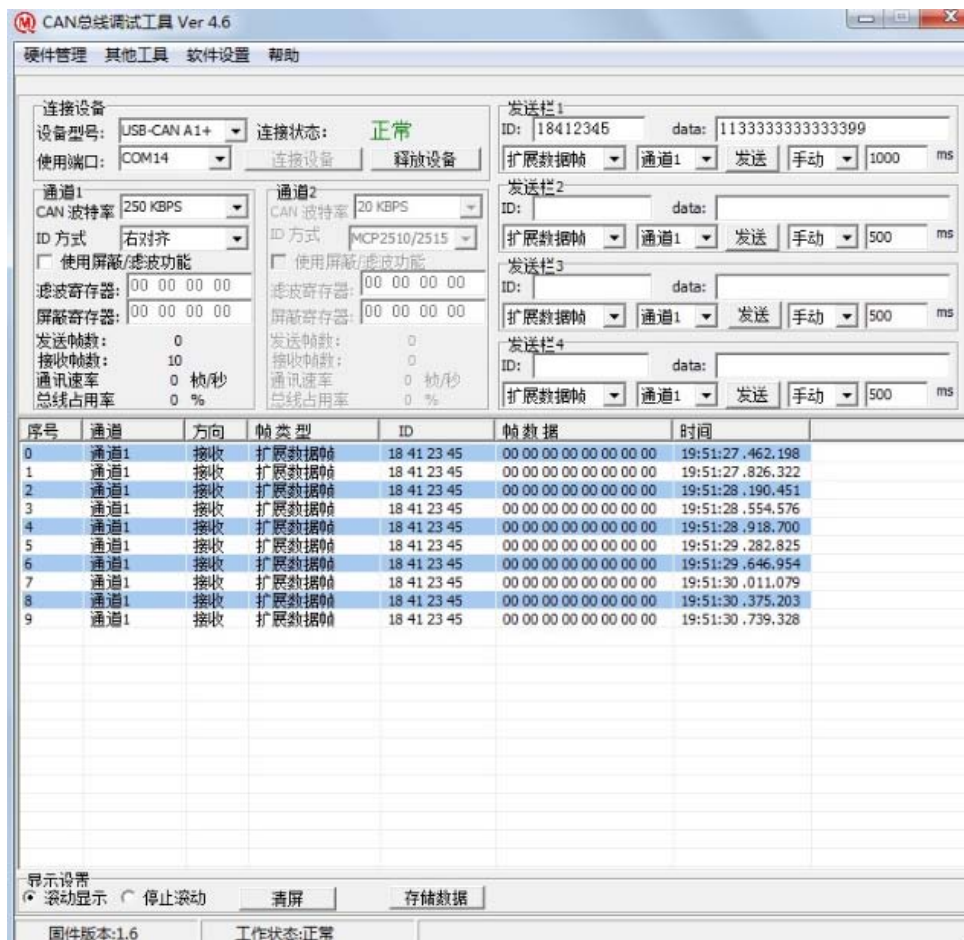


图 4 实验结果

CAN2 通讯实验和 CAN2 类似。

2.6 CAN1 与 CAN2 通讯实验

实验原理：CAN1 以 ID 为 0X18412345 的扩展帧向 can 总线上发数据，并把自己的过滤器设置为 0x18400000（即只接收和此 ID 相同的 CAN 扩展数据帧）；CAN2 以 ID 为 0X18400000 的扩展帧向 CAN 总线上发送数据，并把自己的过滤器设置为 0X18412345。如果把 can1 和 can2 的相连接，则 can1 可以收到 can2 的数据，进入自己的中断函数；can2 也可以收到 can1 的数据进入自己的中断函数。如果在 can1 和 can2 的中断函数里分别设置断点，则可以验证数据收发的正确性。当然也可以通过 usb 转 can 设备直接看结果。

实验步骤：

- (1) 用导线将开发板上的 CAN1_H 与 CAN2_H、CAN1_L 与 CAN2_L 连接。
- (2) 用 JLINK 将开发板和电脑连接，打开 CAN1 与 CAN2 通讯实验工程，编译，仿真运行。
- (3) 分别在 can1 和 can2 中断函数中（在 can.C 文件里）设置断点，如下


```
142 //CAN1中断程序
143 void CAN1_RX0_IRQHandler(void)
144 {
145     CanRxMsg RxMessage;
146     unsigned char i;
147
148     CAN_Receive(CAN1,CAN_FIFO0, &RxMessage);
149     //CAN_ID = RxMessage.ExtId;
150     for(i = 0;i < 8; i++)
151     {
152         CAN2_DATA[i] = RxMessage.Data[i];
153     }
154
155     CAN_ClearITPendingBit(CAN1,CAN_IT_FMP0);
156     // SEI();//打开总中断
157 }
158
159 //CAN2中断程序
160 void CAN2_RX0_IRQHandler(void)
161 {
162     CanRxMsg RxMessage;
163     unsigned char i;
164
165     CAN_Receive(CAN2,CAN_FIFO0, &RxMessage);
166     //CAN_ID = RxMessage.ExtId;
167     for(i = 0;i < 8; i++)
168     {
169         CAN_DATA[i] = RxMessage.Data[i];
170     }
171
172     CAN_ClearITPendingBit(CAN2,CAN_IT_FMP0);
173     // SEI();//打开总中断
174 }
```

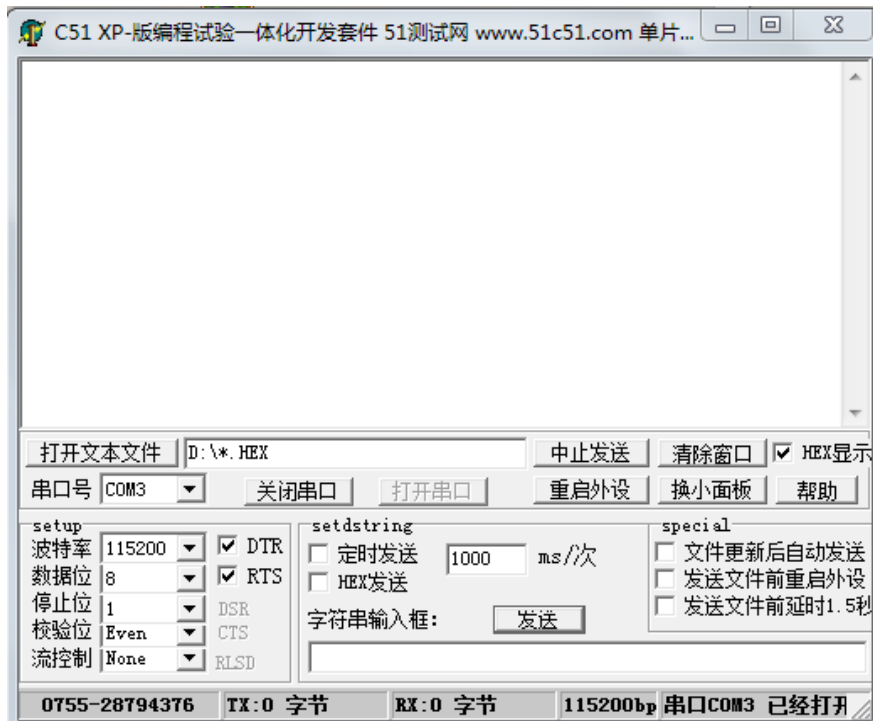
点击全速运行后，可以看到程序停在了 can1 或 can2 中断函的断点处，继续点击单步运行，可以看到程序交替的停在 can1 或 can2 的断点处，说明程序运行正确。

2.7 I2C 实验—读写 24c04

实验原理：首先通过 I2C 向 24c02 存储单元中写入 0x00~0xff，然后再读取相应的存储单元，并将读取的结果通过串口发送出去，以验证写入和读取的正确性。

实验步骤

- (1) 用串口线（usb 转串口线）将开发板和电脑相连接。打开电脑的设备管理器，查看使用的端口号。
- (2) 打开 sscomv20(串口调试程序” 软件
- (3) 将端口号设置为第（1）步查到的端口号，波特率：115200，数据位：8，校验位：Even,偶校验。停止位：1，数据格式选择 HEX，如图 5 所示。



(4)

图 5 串口设置

(4) 用 JLINK 将开发板和电脑连接，打开 I2C 实验-读写 24C02 工程，编译，运行，此时可以看到串口软件接收到了开发板发来的数据，实验结果如图 6 所示。

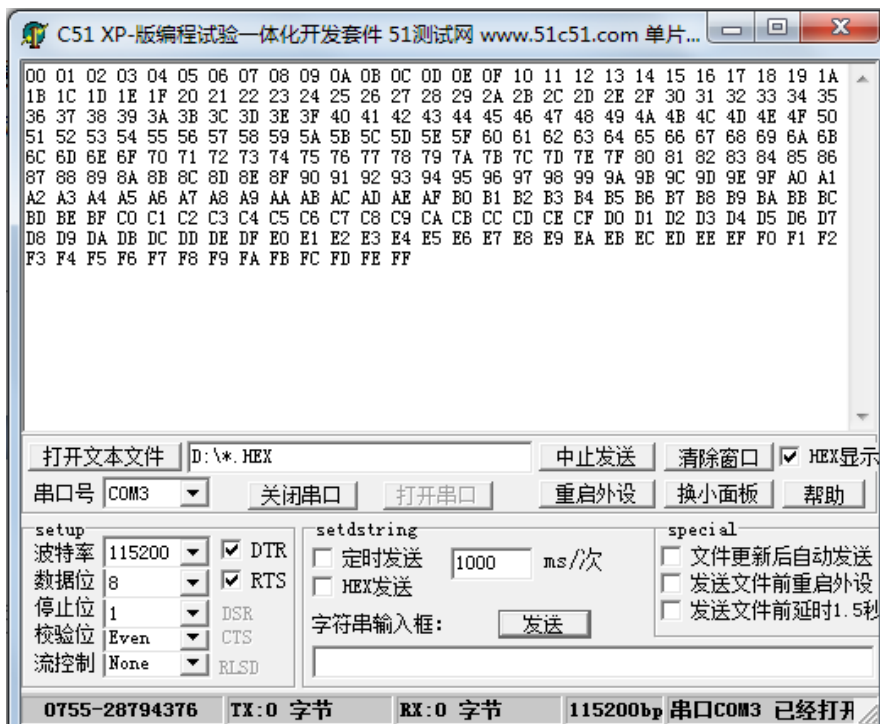


图 6 实验结果

2.8 SPI 通讯实验--读写 SST2508B

实验原理、实验步骤以及实验结果和 2.7 节类似。

2.9 TCP 服务器收发数据实验

实验原理：开发板做为 TCP 服务器，PC 机（软件）作为 TCP 客户端。首先 PC 机软件向开发板发起连接，若连接成功，PC 机软件向开发板发送数据，开发板接收到后将数据原样返回。

实验步骤：

- (1) 将电脑 IP 设置如图 7 所示。

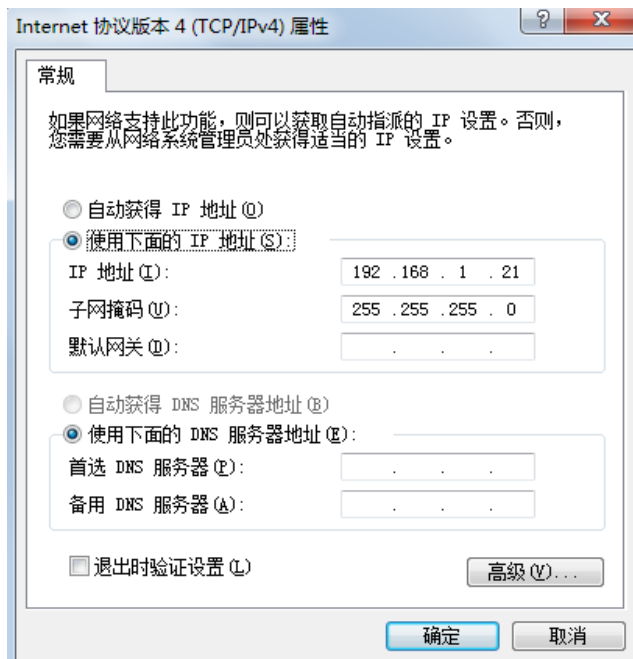


图 7 电脑 IP 设置

- (2) 打开“TCP_tester 分析”软件，默认设置如图 8 所示。

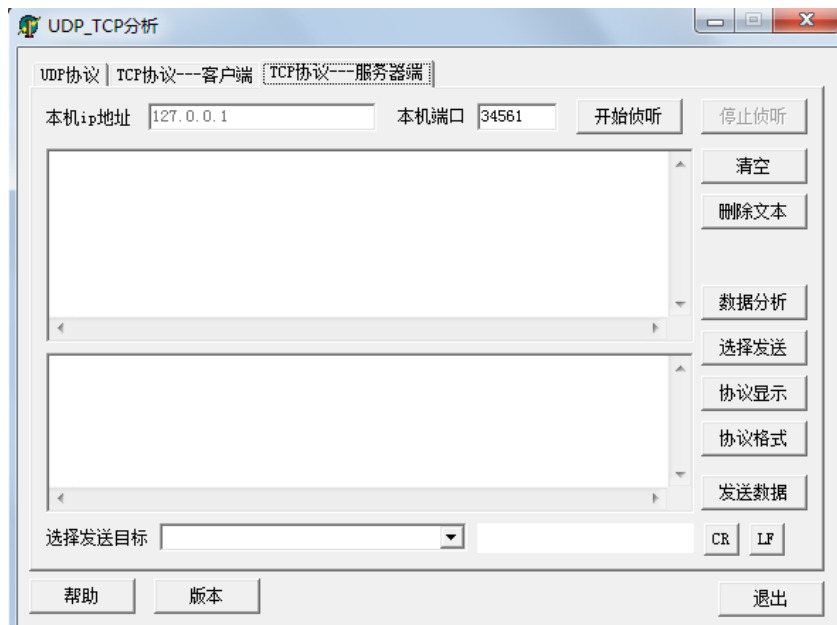


图 8 TCP_tester 软件设置

- (3) 用交叉网线将开发板和电脑连接，用 jlink 将开发板和电脑连接。
- (4) 打开 TCP_IP 收发数据实验工程，编译下载，运行。
- (5) 在图 8 中点击“TCP 协议---客户端”，并将 ip 和端口设置如图 9 所示(因为开发板的 IP 为 192.168.1.252，TCP 端口为 1030)。

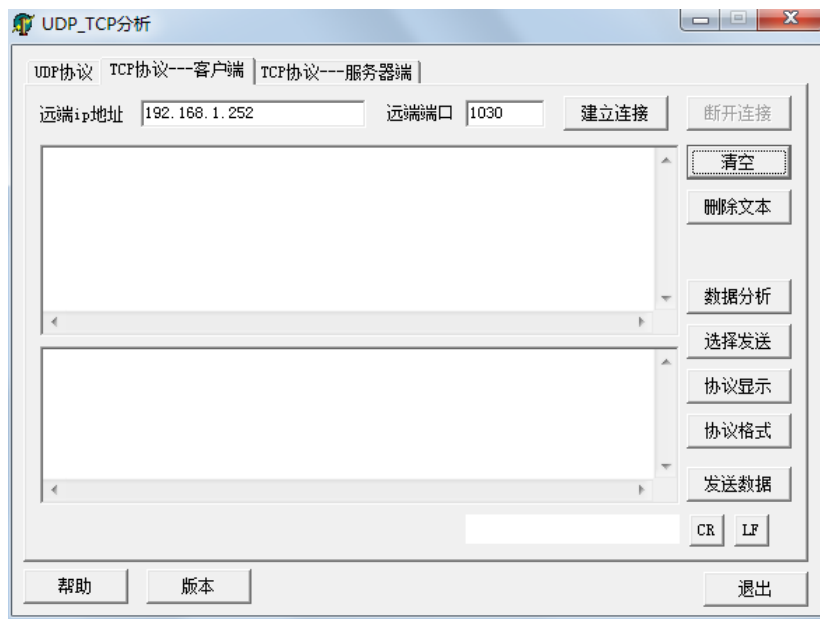


图 9 TCP_tester 软件设置

- (6) 点击图 9 中的“建立连接”，若连接成功后，在发送显示区输入任意数据，点击“发送数据”，则可以看到接收区有开发板发来的原样数据。如图 10 所示。

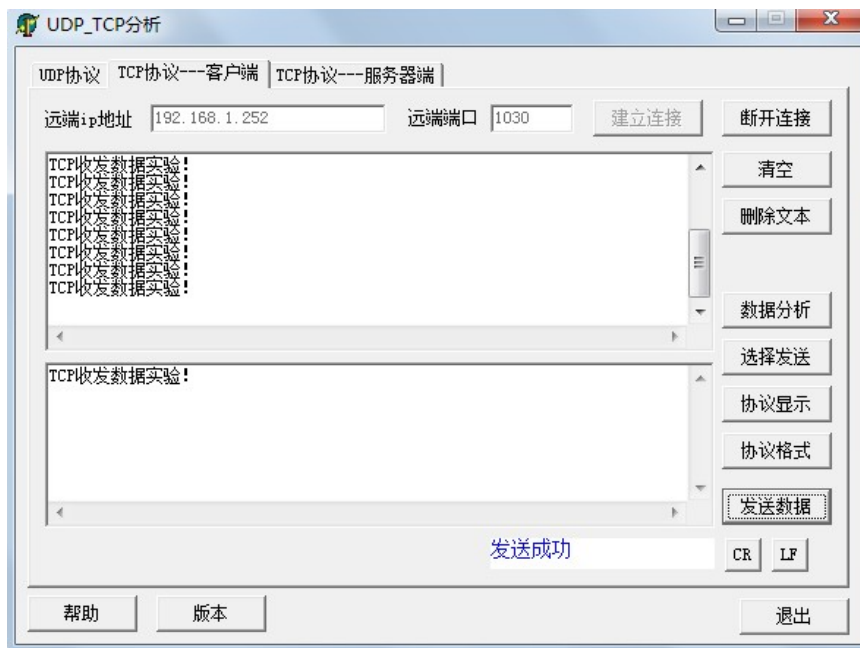


图 10 实验结果

2.10 动态 IP 实验

实验原理：开发板通与交换机或路由器连接，接收交换机或路由器分配的动态 IP 地址作为自己的 IP 地址。通过串口可以看到 IP 分配是否成功。

实验步骤：

(1) 用直通网线将开发板和交换机或路由器连接。

(2) 打开“sscomv20(串口调试程序)”，波特率：115200，数据位：8，校验位：Even,偶校验。停止位：1。如图 1 所示。

(3) 用 JLINK 将开发板和电脑连接，打开动态 IP 实验工程，编译下载，运行。

(4) 在搜寻 ip 的过程中 4 个 led 会流水显示。

如果自动获取 ip 成功则串口调试软件会收到自动获取的 ip 地址，如：ip:192,168,1,37

如果自动获取 ip 失败，开发板将 ip 设置为固定 ip (192, 168, 1, 253)。此时串口会收到：IP:FAIL! SET IP:192, 168, 1, 253

(5) 程序注释

该功能主要修改 lwipopt.h 文件中的

```
#define LWIP_DHCP 1 //自动获取 ip
```

若注释掉该句，则为静态 ip。

2.11 HTTP 网页服务器实验

实验步骤：

(1) 保证电脑的 IP 在 192.168.1.x 网段内，其中 x 不能为 253（因为 253 被开发板占用）。

(2) 用 JLINK 将开发板和电脑连接，打开 HTTP 网页服务器实验工程，编译下载，运行。

(3) 打开浏览器，在地址栏输入：http://192.168.1.253/

(4) 可以看到运行结果如图 12 所示：

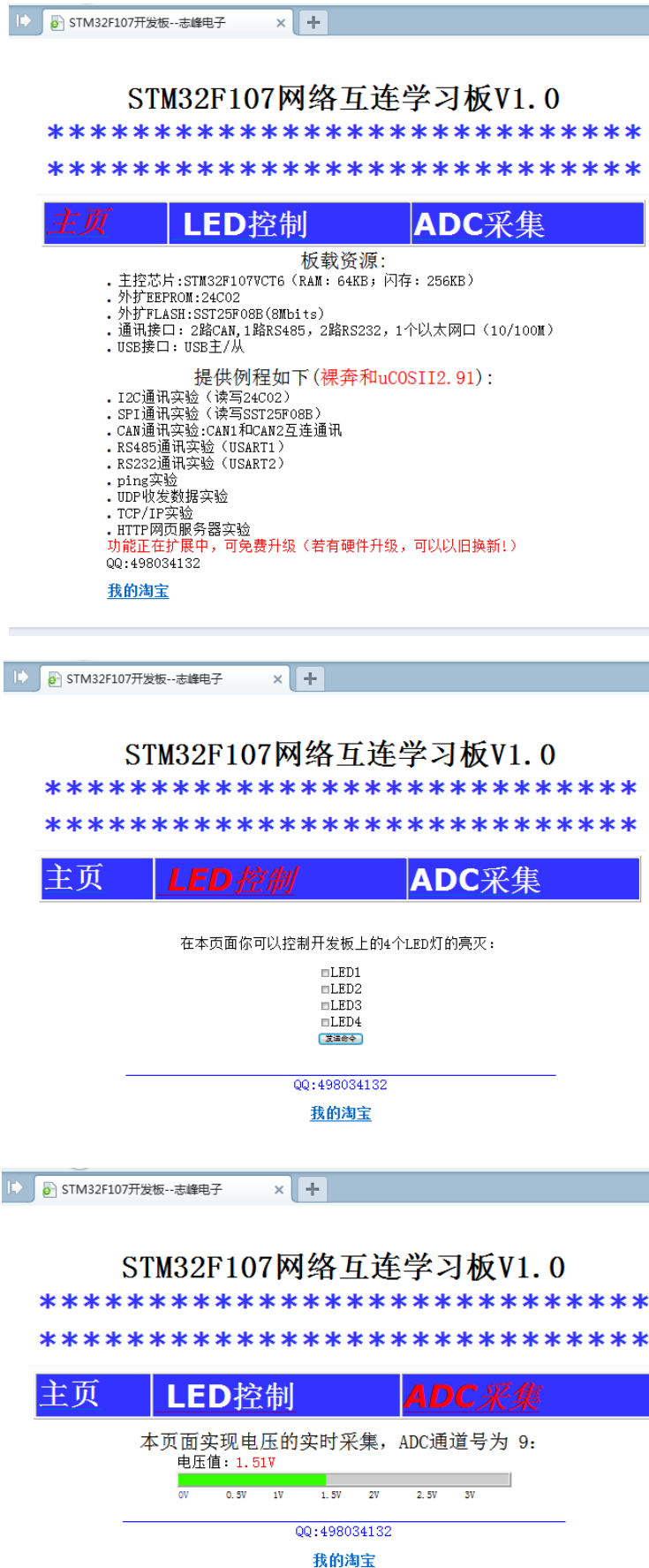


图 12 运行结果

2.12 USB 数据存储实验 1（读写 STM32F107 片内 flash）

实验步骤：

- （1）将开发板用 jlink 调试器和电脑连接，给开发板上电。
- （2）打开“打开“USB 数据存储实验 1-PC 机通过 USB 读写 STM32F107 片内 flash”实验工程，编译下载。
- （3）把开发板的电源断开，用跳线帽将 J2 短接。将 usb 线的一端插在开发板的 J10 上，另一端插在电脑上。
- （4）此时会看到电脑发现了新硬件，并自动安装驱动，如果是第一次使用片内的 flash 做该实验，则会提示格式化，按照提示操作即可。
- （5）格式完成后，可以向该“可移动磁盘”里存储文件。读写文件时 LED1 会闪烁。

2.13 USB 数据存储实验 2（读写板载 SST25F08B）

实验步骤 2.12 节类似。

2.14 U 盘读写实验

实验原理：先检查 u 盘是否插上，若插上则读取 u 盘的容量和根文件目录（注意暂时不支持长文件名），并通过串口将该信息发送到电脑上，然后判断“U 盘例程.TXT”文件是否存在，若不存则创建该文件并将数据“感谢您使用志峰开发板!! \r\n”写入该文件。

实验步骤：

- （1）设置串口软件“sscomv20(串口调试程序)”如图 1 所示。将 u 盘插到开发板的 J3 上。用串口线将开发板和电脑连接。
- （2）打开 u 盘读写实验例程，并下载到开发板，复位运行程序，此时可以看到串口输出了相关信息，如图 13 所示。

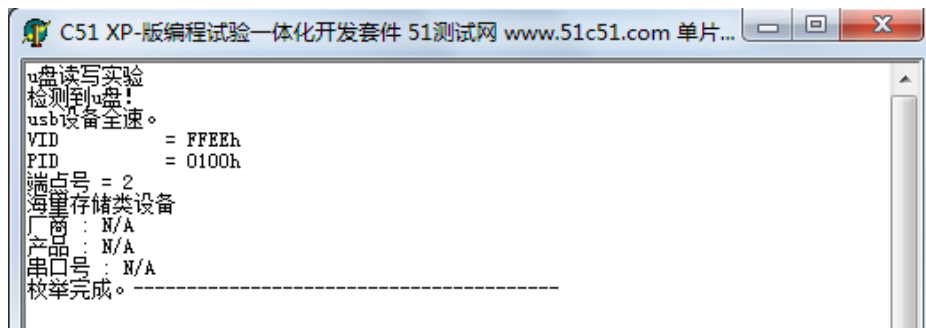
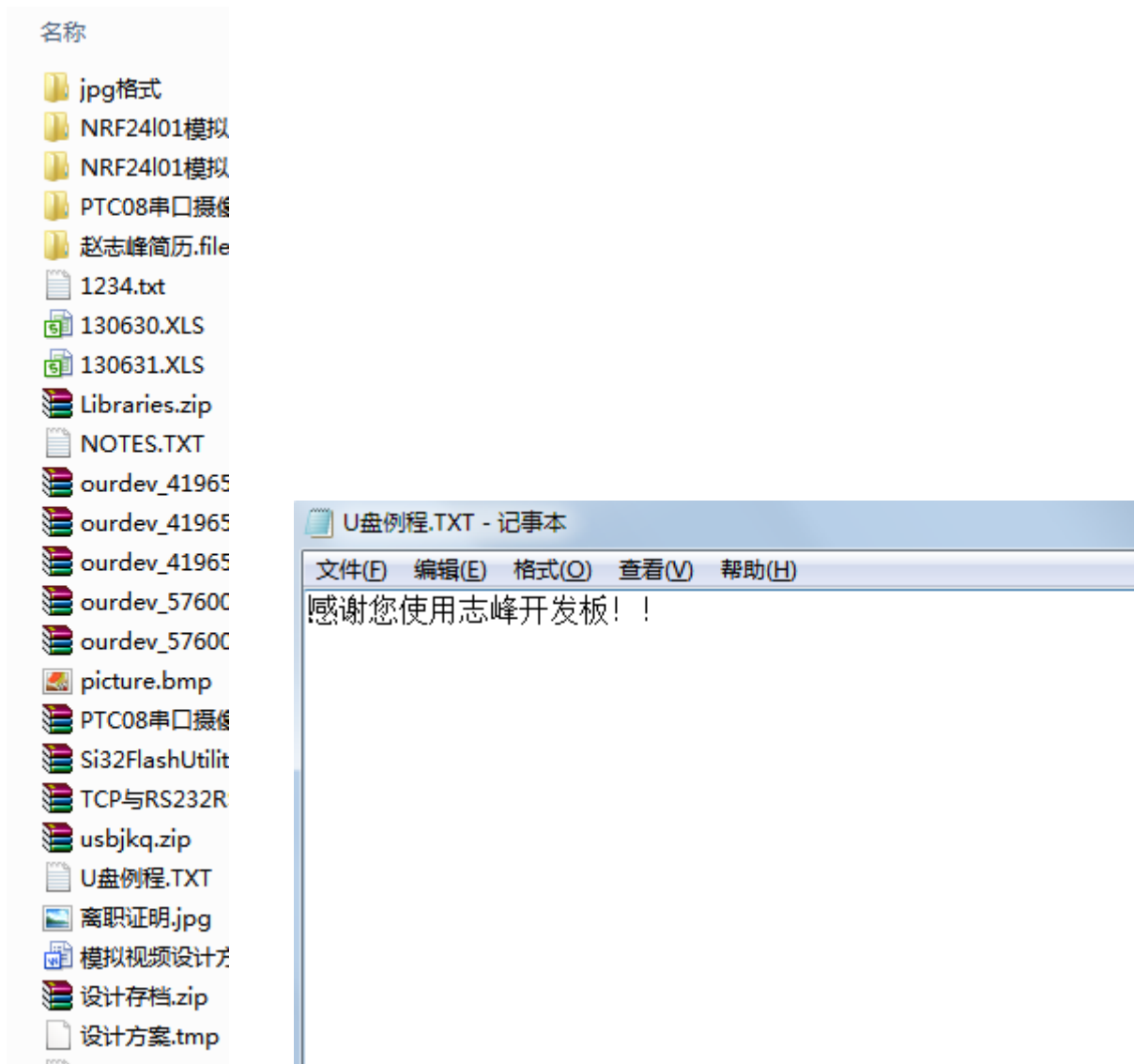




图 13 实验结果

以下为 u 盘中的内容:



根目录内容

U 盘例程.TXT 文件中的内容

2.15 USB 鼠标实验

实验原理：通过四个 led 灯分别指示鼠标的左键、中键、右键以及鼠标的移动等。

将鼠标插到 J3(USB)口上，打开本实验工程，下载，运行，操作鼠标，则可以看到相应的 led 灯亮起。

2.16 USB_JoyStickMouse 实验

实验原理：通过开发板的上四个按键来模拟鼠标的上下、左右移动（也可以将按键定义成左键按下和右键按下，请读者自行编写相应的代码）。

实验步骤：

- (1) 用 mini USB 线将开发板和电脑相连接。
- (2) 打开该实验工程，编译下载到单片机，运行。
- (3) 此时电脑会提示正在安装驱动，如图 14 所示。

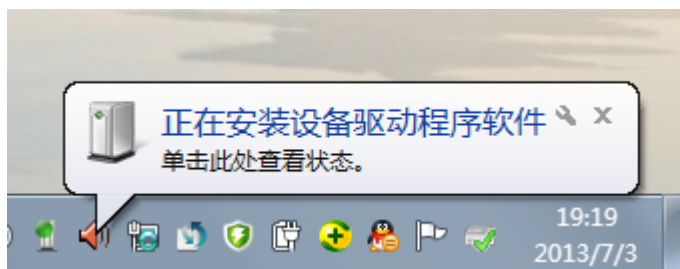


图 14 驱动安装中

为了节省驱动安装时间，可以单击上面的方框，跳过搜索，直接安装即可。

(4) 驱动安装成功后，按下按键可以看到鼠标光标在做相应的移动。

2.17 2.4G 无线模块 NRF24L01 通信实验

实验原理：两个开发板通过无线模块 NRF24L01 传输数据（按键的值），并将接收到的数据（按键值）通过 LED 显示出来。NRF24L01 的工作模式设置方式是：如果有键按下则进入到发送模式，发送完毕后，立即进入到接收模式。

实验步骤：

(1) 将无线模块 NRF24L01 插在 J19 上，如图 15 所示。



图 15 NRF24L01 的安装方式

(2) 打开该例程工程，编译，并下载到两个开发板中，复位运行。按下其中一个开发板的按键则另一个开发板的相应 led 灯会点亮。

(3) 也可以借助串口调试助手看无线模块的收发状态。

本例程的演示视频地址：http://v.youku.com/v_show/id_XNTY4MDA3MTY0.html

2.18 基于 NRF24L01 的无线 RS232 通讯实验

实验原理：如图 16 所示：

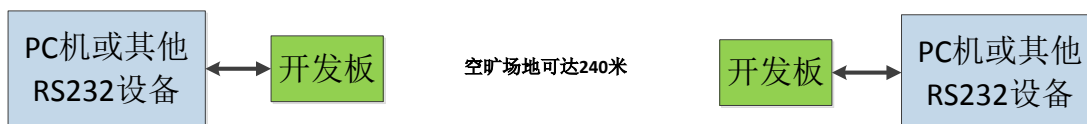


图 16 无线 RS232 示意图

实验步骤：

(1) 将无线模块安装在开发板上（两套），两个开发板的串口分别连接不通的 RS232 设备。

(2) 将该例程下载到开发板上，打开串口调试软件，设置如图 1 所示。

(3) 发送格式为：每帧数据 32 个字节，其中左后两个字节为“?;”.则一端 RS232 发送数据，另一端 RS232 设备就会收到数据，注意，两端的 RS232 设备不能同时发送，因为无线模块不能都处于发送状态。本例程仅仅为抛砖引玉，读者可以做自己的通讯协议。另外，如果按照如图 17 的方式

连接，可以做个 RS232 “网关”。

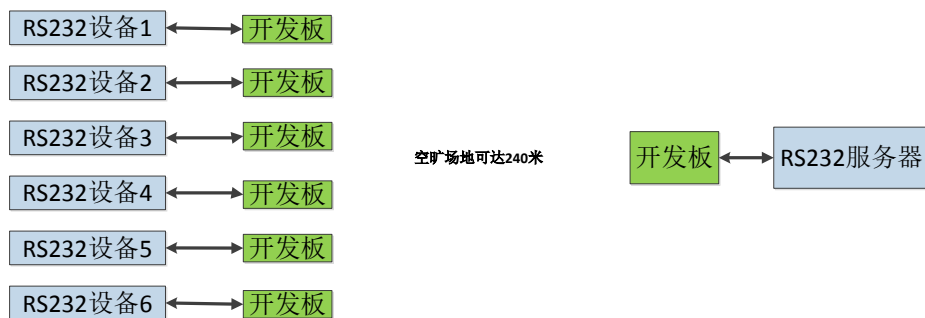


图 17 无线 RS232 网络

2.19 DS18B20 温度试验

实验原理：开发板将读取 DS18B20 的温度值通过串口发送到电脑上。

实验步骤：

(1) 用串口线将开发板与电脑连接，串口设置如图 1 所示；将 DS18B20 安装在开发板上，如图 18 所示：



图 18 DS18B20 安装方式

(2) 将该例程下载到开发板，复位单片机，则从串口显示界面上可以看到温度值（如 32 度），如下图 19。



图 19 实验结果

2.20 UDP 客户端发送数据实验

实验原理：开发板工作在 UDP 客户端模式，以固定的 IP（192.168.1.252）和端口号（1031）向固定的 IP（192.168.1.21）和端口号（1032）发送数据。

实验步骤：

- （1）将电脑 IP 设置如图 7 所示。
- （2）打开软件“USR-TCP232-Test.exe”，并设置如图 20 所示。

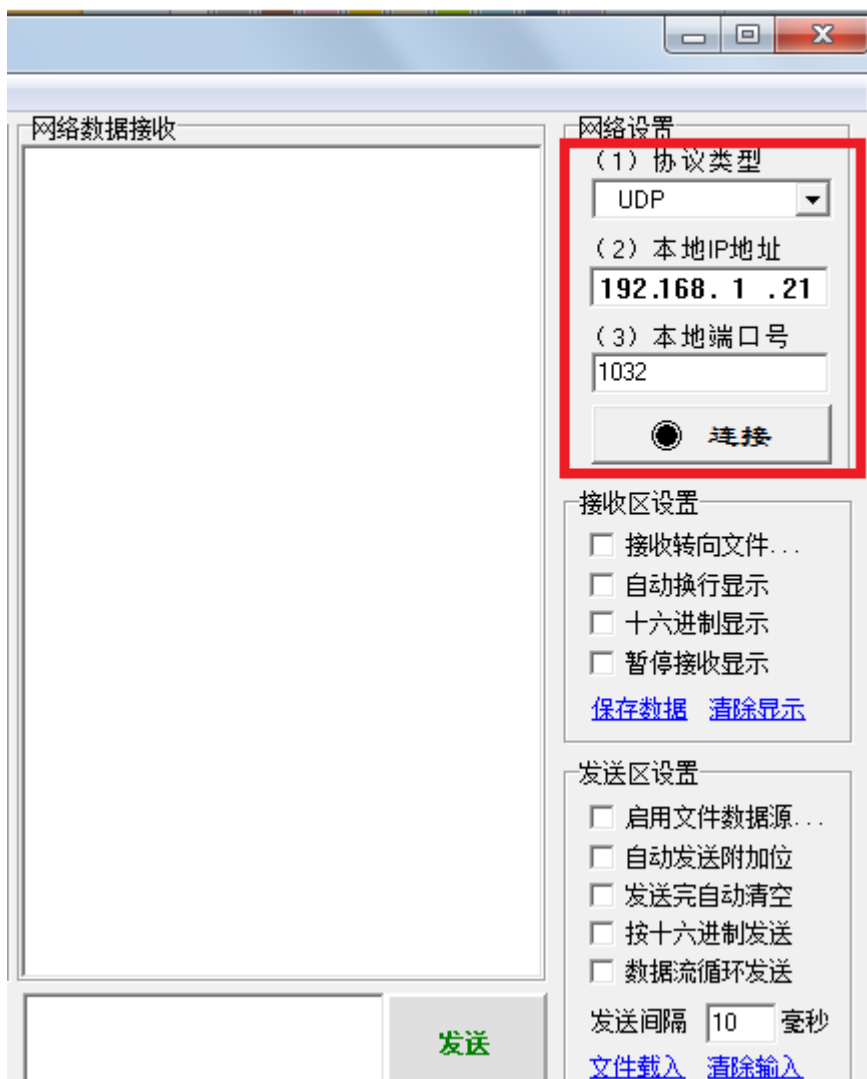


图 20 UDP 设置

- （3）用交叉网线将电脑和开发板连接，将该实验例程下载到开发板，复位运行。
- （4）点击图 20 的“连接”，则会看到开发板发来的数据，如图 21 所示。



图 21 实验结果

2.21 UDP 服务器收发数据实验

实验原理：开发板工作在 UDP 服务器模式，服务器 IP 为 192.168.1.252;端口为 1030。当接收到客户端发来的数据后，把接收到的数据原样返回。

实验步骤：

- (1) 将电脑设置成如图 7 所示。
- (2) 打开软件“USR-TCP232-Test.exe”，如 图 21。



图 21 udp 设置

(3) 用交叉网线将电脑和开发板连接，将该实验例程下载到开发板，复位运行。

(4) 点击图 21 的“连接”，在发送区中输入字符“UDP 服务器实验！”，注意后面要加回车符换行；并将“目标主机”设置成如图 22。

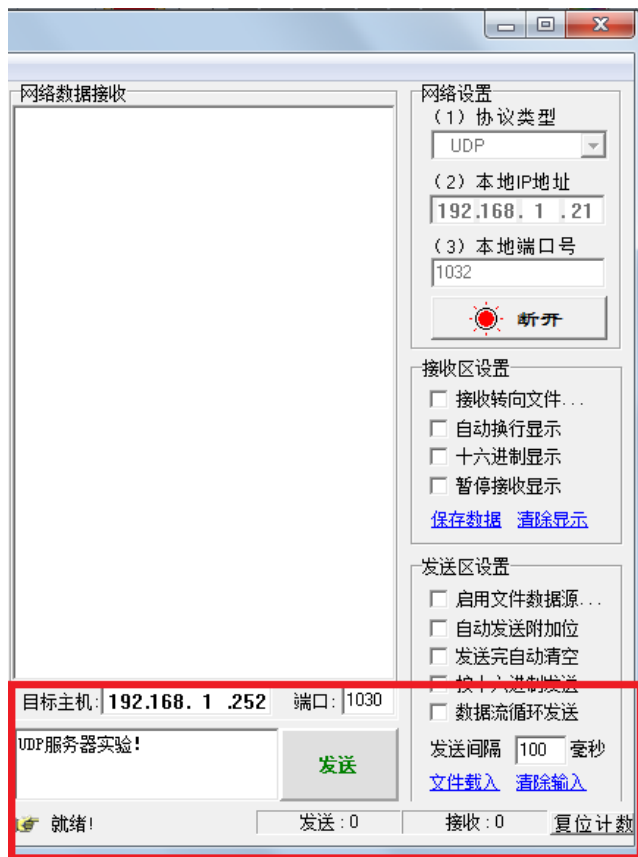


图 22 udp 设置

(5) 点击图 22 的“发送”，则可以看到开发板发来的数据，如图 23 所示。

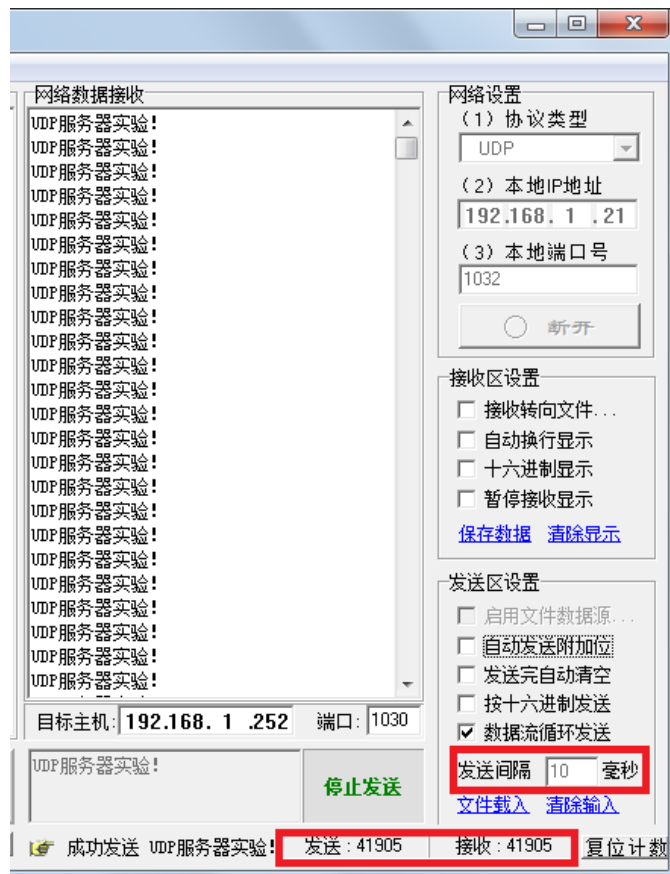


图 23 实验结果

由图 23 可以看出，发送字节数为 41905，接收到的字节数为 41905，程序运行正确。

2.22 TCP 客户端收发数据实验

实验原理：开发板工作在 TCP 客户端模式，以固定的 IP（192.168.1.252）和端口号（1030）向固定的 IP（192.168.1.21）和端口号（1031）发送数据。服务器的 IP 和端口具体可以参考程序，不同版本可能有所不同，只要电脑的 IP 和板子服务器 IP（TCP_SERVER_IP）一致即可。具体在 TCP_CLIENT.H 文件中：

```
#define TCP_LOCAL_PORT      1030
#define TCP_SERVER_PORT    1031

#define TCP_SERVER_IP      192,168,1,21//服务器IP
```

实验步骤：

- （1）将电脑 IP 设置成如图 7 所示，打开软件“USR-TCP232-Test.exe”，并设置如图 24 所示。

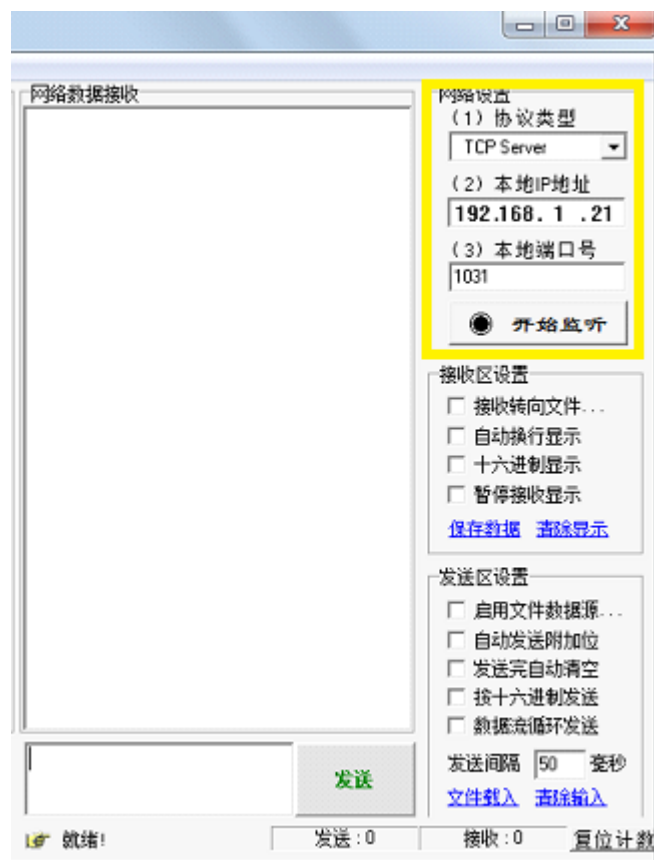


图 24 软件设置

- （2）打开该实验工程，编译下载到开发板，复位运行。
- （3）网口灯亮后，点击图 24 的“开始监听”，稍等片刻后，则可以看到开发板发来的数据，如图 25 所示。

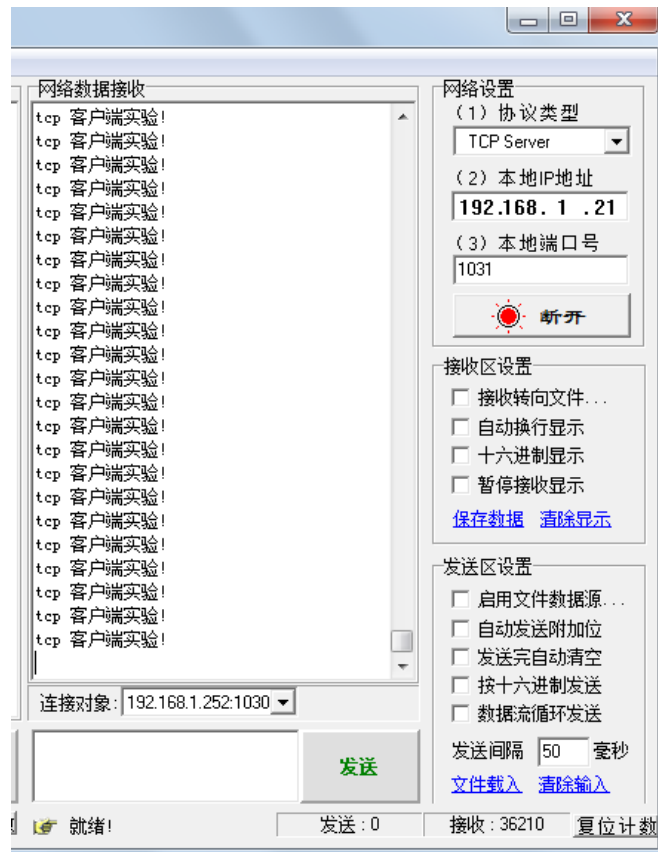


图 25 TCP 发送实验结果

(4) 在 main 函数中注释掉 TCP 发送子函数,即修改后如下所示:

```
053 while(1)
054 {
055     pcb = Check_TCP_Connect(); //检查是否建立连接,若未建立则开始建立
056     if(pcb != 0) //连接建立成功
057     {
058         //TCP_Client_Send_Data(pcb,tcp_data,sizeof(tcp_data)); //该函数为主动向服务器发送函数,
059     }
060 }
061 Delay_s(0xffff); //必要的延时
062 System_Periodic_Handle(); //作为服务器的例行事件服务,主要是更新TCP timers
063 }
```

编译,下载,复位,待网口灯亮后,在图 24 界面的发送区中输入“这是 TCP 客户端收发实验!”,最后回车换行;点击图 24 的“开始监听”后,稍等片刻(需要一点时间),则会出现如图 26 所示界面,表明开发板已经和电脑连接成功。

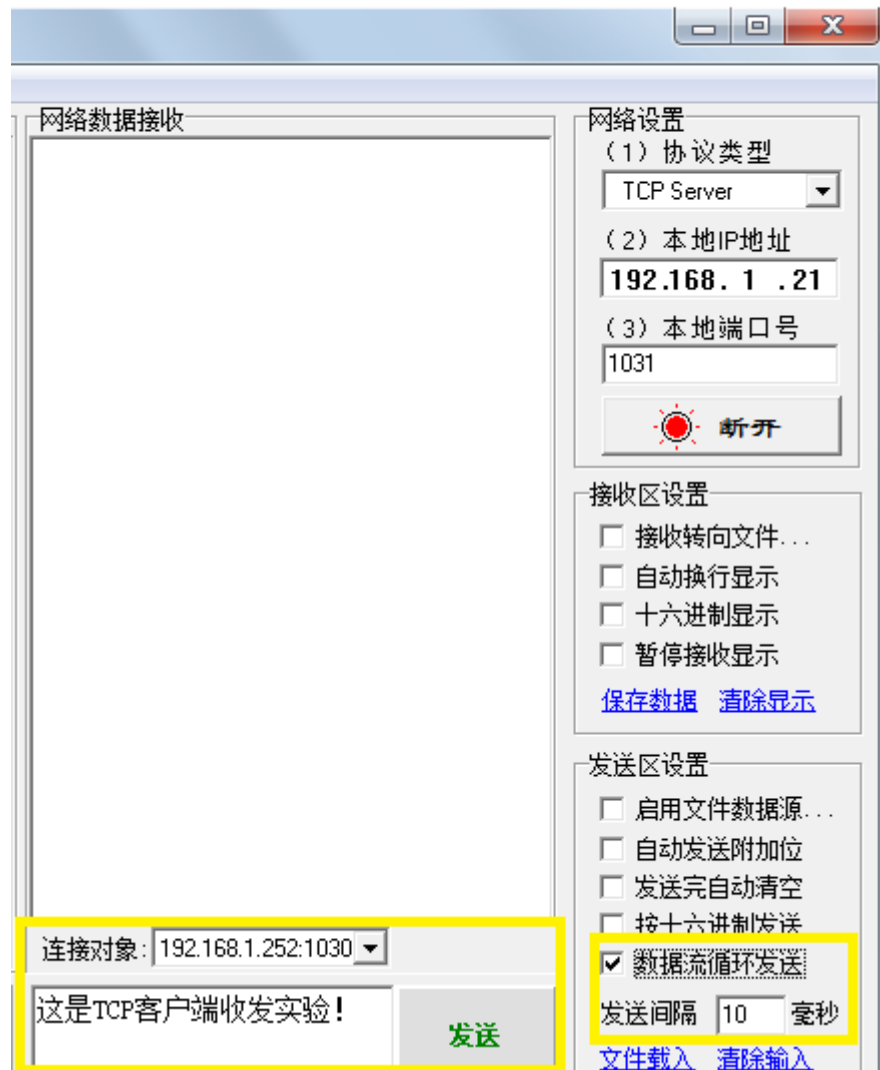


图 26 TCP 客户端收发数据连接成功

连接对象即是开发板的 IP 和端口号。此时即可点击图 26 的“发送”，结果如图 27 所示。

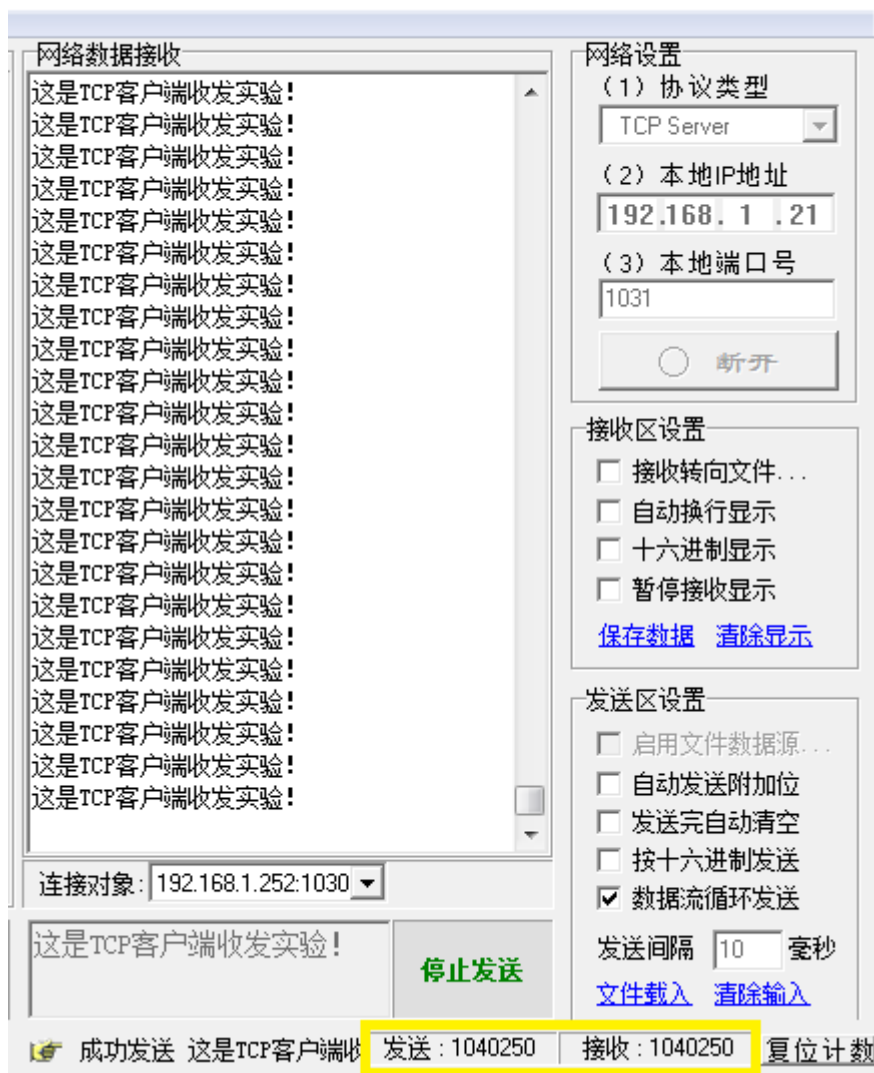


图 27 TCP 客户端收发数据实验结果

由图 27 可以看出，服务器端向开发板发送 1040250 字节数据，开发板向服务器返回的字节数也是 1040250，表明程序运行正确。

2.23 16 通道 ADC 采集实验

因为该配置是采集 16 通道的，但是板子上有些 adc 管脚和其他功能复用（接了其他器件）所以在使用过程中需要注意，板子上有一个电位器，可以通过调节该电位器观察该路变化。

实验原理：开发板使用 ADC1 以 DMA 方式采集 16 通道，然后通过串口输出。

ADC 对应管脚：

| | |
|---------------|-----|
| ADC_Channel_0 | PA0 |
| ADC_Channel_1 | PA1 |
| ADC_Channel_2 | PA2 |
| ADC_Channel_3 | PA3 |
| ADC_Channel_4 | PA4 |
| ADC_Channel_5 | PA5 |
| ADC_Channel_6 | PA6 |
| ADC_Channel_7 | PA7 |

| | |
|----------------|-----|
| ADC_Channel_8 | PB0 |
| ADC_Channel_9 | PB1 |
| ADC_Channel_10 | PC0 |
| ADC_Channel_11 | PC1 |
| ADC_Channel_12 | PC2 |
| ADC_Channel_13 | PC3 |
| ADC_Channel_14 | PC4 |
| ADC_Channel_15 | PC5 |

实验步骤：

(1) 串口设置

设置如图 28 所示：

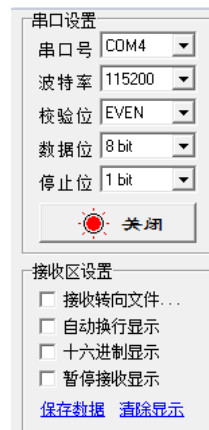


图 28 串口设置

(2) 下载验证

将该例程下载到开发板，复位运行，则可以看到串口调试软件有如下显示：

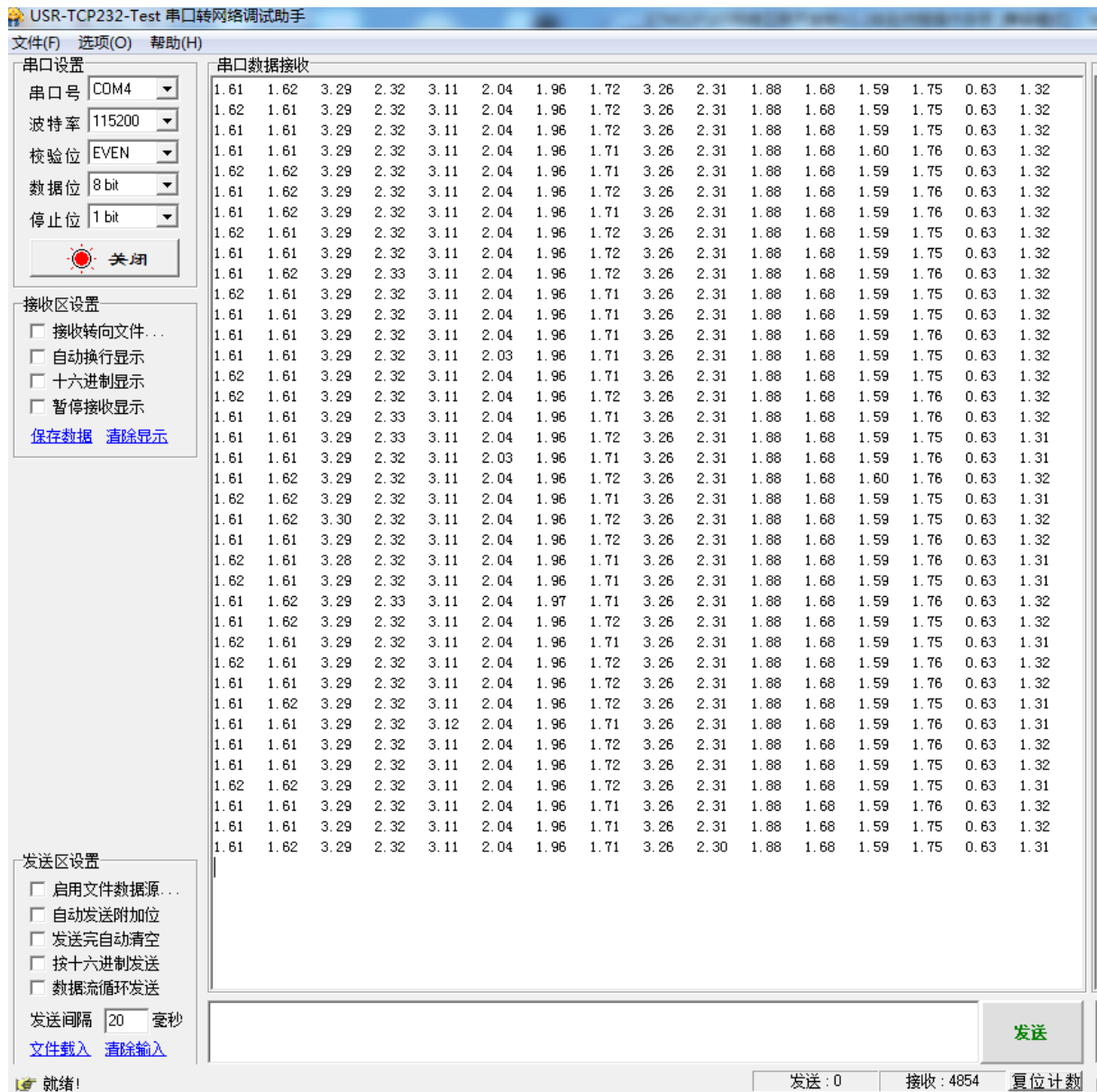


图 30 16 通道 ADC 采集实验结果

2.24 UDP 与 RS232RS485 双向数据透明传输例程---开发板做服务器

参考<<UDP 与 RS232RS485 双向数据透明传输例程(开发板做服务器)使用说明>>.

2.25 TCP 与 RS232RS485 双向数据透明传输实验

参考《TCP 与 RS232RS485 双向数据透明传输例程使用说明》。

2.26 TCP 与 CAN 双向数据透明传输实验

参考《TCP 与 CAN 双向数据透明传输例程使用说明》。

-----以下无正文。