

廈門大學



软件学院

《计算机网络》实验报告

题 目 RS-232 通信程序设计

姓 名 黄勛

学 号 22920212204392

班 级 计算机网络 2 班

实验时间 2023. 3. 24

2023 年 3 月 24 日

1 实验目的

- 了解 RS-232 的通信基本原理，并尝试动手实践

本实验要求制作双机通信程序，实现两个计算机串口通过 RS-232 串口相互连接。实现发送和接收字符串的程序，支持互发信息、支持多次发送。

由甲方向乙方发送字符串（如“Hello World!”），格式如“[SENT 2015-03-26 08:01:15]Hello World!”，并在甲方界面上显示该字符串。同时，在乙方机器上显示“[RECV 2015-03-26 08:01:33] [SENT 2015-03-26 08:01:15] Hello World!”。

2 实验环境

操作系统使用 Windows 10 21H2 版本系统

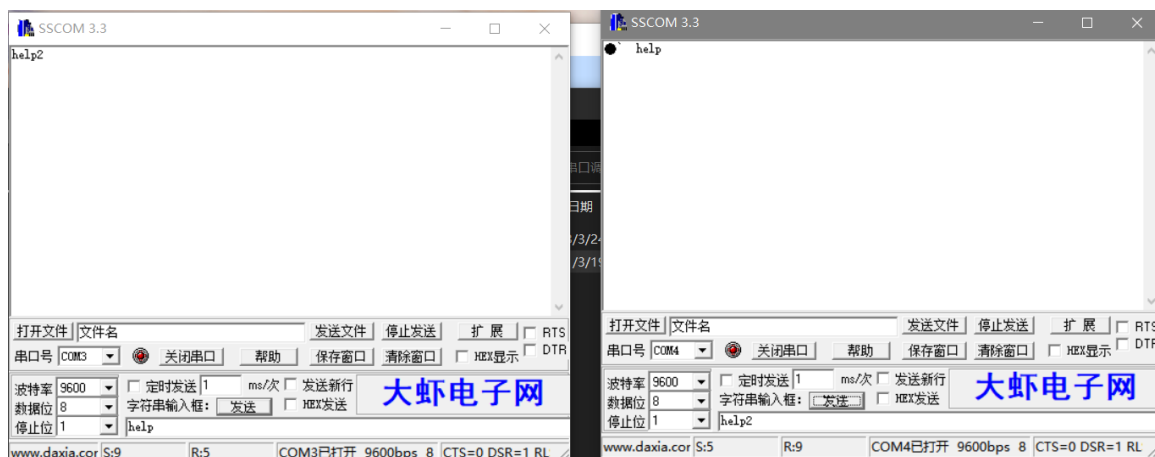
编程语言采用 Qt 框架的 C++语言

3 实验结果

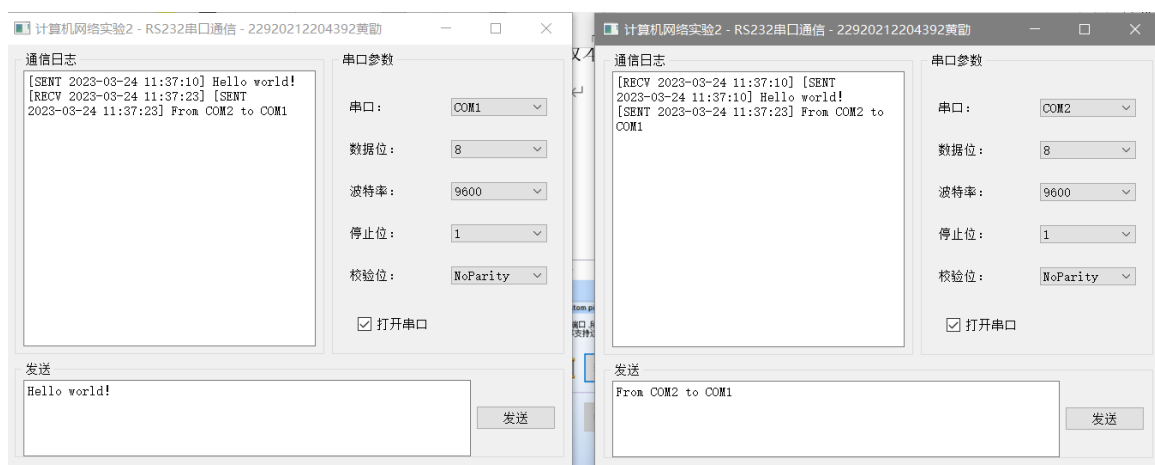
1.创建端口



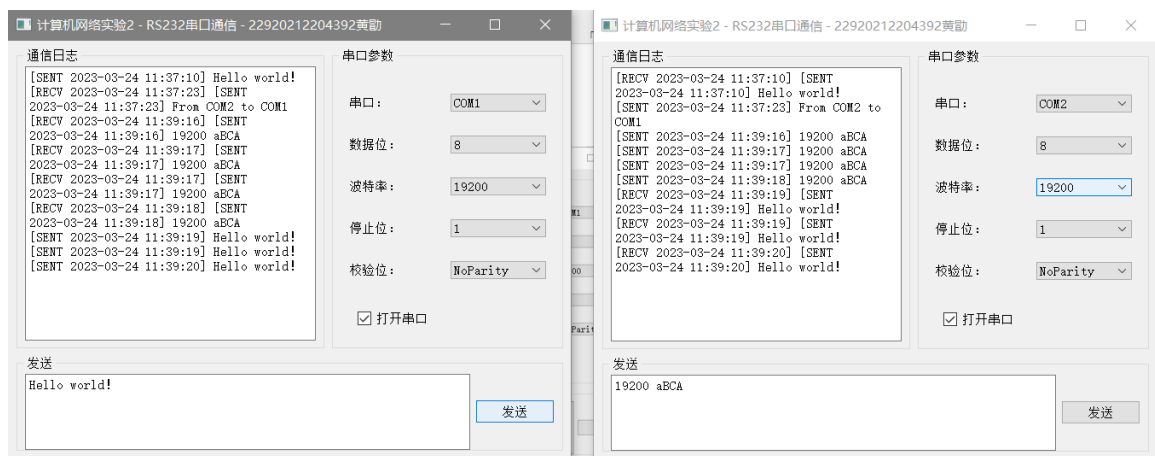
串口调试助手测试



2.编写的程序结果展示



在编写的程序中，我连接上串口 COM1 与 COM2，实现了两个互发信息、支持多次发送。

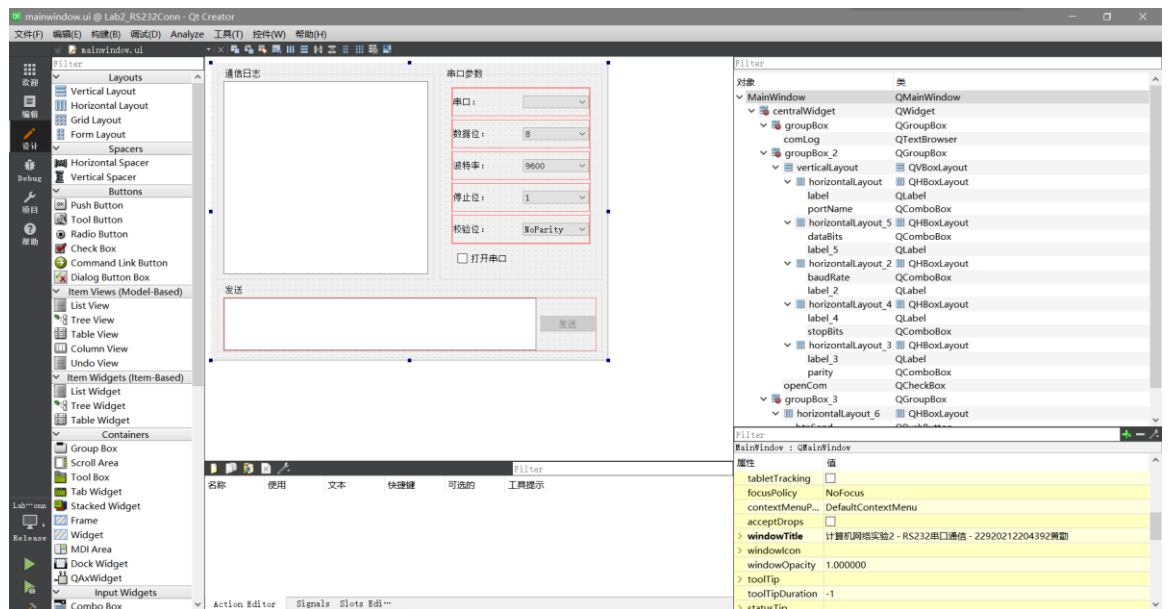


通过更改波特率和校验位，也能实现发送。

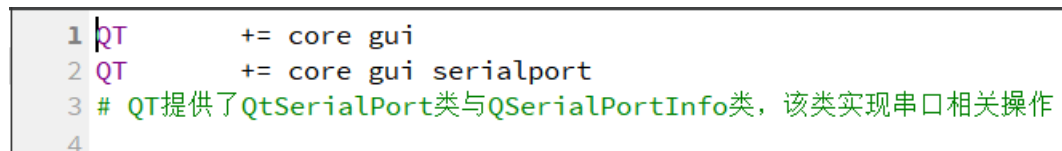


3.代码编写

- 窗口设计



- 串口通信库添加



- 相关函数编写

1. Main 函数启动窗口

```
4 int main(int argc, char *argv[])
5 {
6     QApplication a(argc, argv);
7     MainWindow w;
8     w.show();
9
10    return a.exec();
11 }
```

2. 构造窗口

```
4 MainWindow::MainWindow(QWidget *parent) :    //构造函数，继承QMainWindow类
5     QMainWindow(parent),                    //父类构造函数
6     ui(new Ui::MainWindow)                  //初始化ui指针
7 {
8     ui->setupUi(this);                       //调用自动生成的setupUi()函数，创建UI
9
10    this->serialPort = new QSerialPort;       //创建QSerialPort类指针
11    findFreePorts();                         //寻找空闲状态串口
```

3. 连接事件发生的函数

```
13 connect(ui->openCom, &QCheckBox::toggled, [=](bool checked){    //开关串口按钮的clicked信号与槽的连接
14     if (checked){                                                //按钮选中
15         initSerialPort();                                         //初始化串口
16         ui->btnSend->setEnabled(true);                             //使能发送按钮
17     }else{                                                        //按钮未选中
18         this->serialPort->close();                                  //关闭串口
19         ui->btnSend->setEnabled(false);                             //禁用发送按钮
20         ui->openCom->setChecked(false);                             //将按钮状态置为未选中
21     }
22 });
23
24 connect(this->serialPort, SIGNAL(readyRead()), this, SLOT(recvMsg()));    //串口数据接收
25 connect(ui->btnSend, &QPushButton::clicked, [=]() {             //按钮被按下，发送数据
26     sendMsg(ui->message->toPlainText());
27 });
```

4. 寻找串口

```

35 //寻找空闲状态串口
36 void MainWindow::findFreePorts(){
37     QList<QSerialPortInfo> ports = QSerialPortInfo::availablePorts();
38     for (int i = 0; i < ports.size(); ++i){
39         if (ports.at(i).isBusy()){
40             ports.removeAt(i);
41             continue;
42         }
43         ui->portName->addItem(ports.at(i).portName());
44     }
45     if (!ports.size()){
46         QMessageBox::warning(NULL,"Tip",QStringLiteral("找不到空闲串口"));
47         return;
48     }
49 }

```

5. 调整串口参数

```

49 }
50 //初始化串口
51 bool MainWindow::initSerialPort(){
52     this->serialPort->setPortName(ui->portName->currentText());
53     if (!this->serialPort->open(QIODevice::ReadWrite)){ //尝试打开
54         QMessageBox::warning(NULL,"Tip",QStringLiteral("串口打开失败"));
55         return false;
56     }

57 // 设置波特率
58 this->serialPort->setBaudRate(ui->baudRate->currentText().toInt());
59
60 // 设置数据位
61 if (ui->dataBits->currentText().toInt() == 8){
62     this->serialPort->setDataBits(QSerialPort::Data8);
63 }else if (ui->dataBits->currentText().toInt() == 7){
64     this->serialPort->setDataBits(QSerialPort::Data7);
65 }else if (ui->dataBits->currentText().toInt() == 6){
66     this->serialPort->setDataBits(QSerialPort::Data6);
67 }else if (ui->dataBits->currentText().toInt() == 5){
68     this->serialPort->setDataBits(QSerialPort::Data5);
69 }
70
71 // 设置停止位
72 if (ui->stopBits->currentText().toInt() == 1){
73     this->serialPort->setStopBits(QSerialPort::OneStop);
74 }else if (ui->stopBits->currentText().toInt() == 2){
75     this->serialPort->setStopBits(QSerialPort::TwoStop);
76 }
77
78 // 设置校验位
79 if(ui->parity->currentText() == "NoParity"){
80     this->serialPort->setParity(QSerialPort::NoParity);
81 }else if (ui->parity->currentText() == "EvenParity"){

```

6. 发送信息（主要功能）

```
89 //向串口发送信息
90 void MainWindow::sendMsg(const QString &msg){
91     const QString info = "[SENT " + QDateTime::currentDateTime().toString("yyyy-MM-dd hh:mm:ss") + "]" + msg;
92     this->serialPort->write(QByteArray::fromStdString(info.toStdString() + msg.toStdString()));
93     //this->serialPort->write(QByteArray::fromHex(msg.toLatin1()));
94     ui->comLog->insertPlainText(info + msg + "\n");
95 }
```

7. 接受信息（主要功能）

```
97 //接受来自串口的信息
98 void MainWindow::recvMsg(){
99     QByteArray msg = this->serialPort->readAll();
100     ui->comLog->insertPlainText("[RECV " + QDateTime::currentDateTime().toString("yyyy-MM-dd hh:mm:ss") + "]" + msg.toStdString().data() + "\n");
101 }
102
```

4 实验总结

本实验要求制作双机通信程序，通过 RS-232 串口相互连接的两个计算机实现发送和接收字符串。程序应支持互发信息和多次发送。在本次实验我实现了相关要求，提升了我对相关串口的知识掌握，收获颇丰！