

java认证辅导:Java串口通信简介Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_java_E8_AE_A4_E8_AF_81_c104_644608.htm 嵌入式系统或传感器网络的很多应用和测试都需要通过PC机与嵌入式设备或传感器节点进行通信。其中，最常用的接口就是RS-232串口和并口(鉴于USB接口的复杂性以及不需要很大的数据传输量，USB接口用在这里还是显得过于奢侈，况且目前除了SUN有一个支持USB的包之外，我还没有看到其他直接支持USB的Java类库)。SUN的CommAPI分别提供了对常用的RS232串行端口和IEEE1284并行端口通讯的支持。RS-232-C(又称EIA RS-232-C，以下简称RS232)是在1970年由美国电子工业协会(EIA)联合贝尔系统、调制解调器厂家及计算机终端生产厂家共同制定的用于串行通讯的标准。RS232是一个全双工的通讯协议，它可以同时进行数据接收和发送的工作。

1 常见的Java串口包 目前，常见的Java串口包有SUN在1998年发布的串口通信API：`comm2.0.jar`(Windows下)、`comm3.0.jar`(Linux/Solaris).IBM的串口通信API以及一个开源的实现。鉴于在Windows下SUN的API比较常用以及IBM的实现和SUN的在API层面都是一样的，那个开源的实现又不像两家大厂的产品那样让人放心，这里就只介绍SUN的串口通信API在Windows平台下的使用。串口包的安装(Windows下)到SUN的网站下载`javacomm20-win32.zip`，包含的东西如下所示：按照其使用说明(`Readme.html`)的说法，要想使用串口包进行串口通信，除了设置好环境变量之外，还要将`win32com.dll`复制到`\bin`目录下.将`comm.jar`复制到`\lib`。

把javax.comm.properties也同样拷贝到\lib目录下。然而在真正运行使用串口包的时候，仅作这些是不够的。因为通常当运行“java MyApp”的时候，是由JRE下的虚拟机启动MyApp的。而我们只复制上述文件到JDK相应目录下，所以应用程序将会提示找不到串口。解决这个问题的方法很简单，我们只须将上面提到的文件放到JRE相应的目录下就可以了。值得注意的是，在网络应用程序中使用串口API的时候，还会遇到其他更复杂问题。有兴趣的话，你可以查看CSDN社区中“关于网页上Applet用javacomm20读取客户端串口的问题”的帖子。

2 串口API概览

2.1 javax.comm.CommPort

这是用于描述一个被底层系统支持的端口的抽象类。它包含一些高层的IO控制方法，这些方法对于所有不同的通讯端口来说是通用的。SerialPort和ParallelPort都是它的子类，前者用于控制串行端口而后者用于控这并口，二者对于各自底层的物理端口都有不同的控制方法。这里我们只关心SerialPort。

2.2 javax.comm.CommPortIdentifier

这个类主要用于对串口进行管理和设置，是对串口进行访问控制的核心类。主要包括以下方法

- 确定是否有可用的通信端口
- 为IO操作打开通信端口
- 决定端口的所有权
- 处理端口所有权的争用
- 管理端口所有权变化引发的事件(Event)

2.3 javax.comm.SerialPort

这个类用于描述一个RS-232串行通信端口的底层接口，它定义了串口通信所需的最小功能集。通过它，用户可以直接对串口进行读、写及设置工作。

2.4 串口API实例

大段的文字怎么也不如一个小例子来的清晰，下面我们就一起看一下串口包自带的例子---SerialDemo中的一小段代码来加深对串口API核心类的使用方法的认识。

2.4.1 列举出本机所有可用串口

void

```
listPortChoices() { CommPortIdentifier portId. Enumeration en =
CommPortIdentifier.getPortIdentifiers(). // iterate through the
ports. while (en.hasMoreElements()) { portId =
(CommPortIdentifier) en.nextElement(). if (portId.getPortType()
== CommPortIdentifier.PORT_SERIAL) {
System.out.println(portId.getName()). } }
portChoice.0select(parameters.getPortName()). } 以上代码可以列
举出当前系统所有可用的串口名称，我的机器上输出的结果
是COM1和COM3。 2.4.2 串口参数的配置 串口一般有如下参
数可以在该串口打开以前配置进行配置：包括波特率，输入/
输出流控制，数据位数，停止位和齐偶校验。 SerialPort sPort.
try { sPort.setSerialPortParams(BaudRate,Databits,Stopbits,Parity).
//设置输入/输出控制流
sPort.setFlowControlMode(FlowControlIn | FlowControlOut). }
catch (UnsupportedCommOperationException e) {} 2.4.3 串口的
读写 对串口读写之前需要先打开一个串口：
CommPortIdentifier portId =
CommPortIdentifier.getPortIdentifier(PortName). try { SerialPort
sPort = (SerialPort) portId.open("串口所有者名称", 超时等待时
间). } catch (PortInUseException e) { //如果端口被占用就抛出这
个异常 throw new SerialConnectionException(e.getMessage()). } //
用于对串口写数据 OutputStream os = new
BufferedOutputStream(sPort.getOutputStream()). os.write(int
data). //用于从串口读数据 InputStream is = new
BufferedInputStream(sPort.getInputStream()). int receivedData =
is.read(). 读出来的是int型，你可以把它转换成需要的其他类
```

型。这里要注意的是，由于Java语言没有无符号类型，即所有的类型都是带符号的，在由byte到int的时候应该尤其注意。因为如果byte的最高位是1，则转成int类型时将用1来占位。这样，原本是10000000的byte类型的数变成int型就成了1111111110000000，这是很严重的问题，应该注意避免。

100Test 下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com