**中国海洋大学计算机科学与技术学院**

实验报告

**姓名：惠欣宇 年级：2021级 专业：计算机科学与技术**

**科目：物联网系统设计与开发 题目：电参表仿真设备通信**

**实验时间：2024年 5月 11日 实验教师：郭忠文**

**目录**

[一、实验目的及要求 2](#_Toc166319132)

[1. 实验目的 2](#_Toc166319133)

[2. 实验要求 2](#_Toc166319134)

[二、实验内容 2](#_Toc166319135)

[三、实验设备及软件 2](#_Toc166319136)

[四、通信协议 3](#_Toc166319137)

[五、实验步骤 3](#_Toc166319138)

[五、报告要求 7](#_Toc166319139)

[1. wt310电参表的主要用途是什么 7](#_Toc166319140)

[2. USB-RS232转换器的作用是什么 7](#_Toc166319141)

[3. 为何在笔记本Windows系统设备管理器中会有实验对应的 RS-232 接口 7](#_Toc166319142)

[4. 把填写10组解析后传感数据的4-3-1表格附在实验报告中 7](#_Toc166319143)

[5. 写出调试过程遇到的主要问题和解决方法 10](#_Toc166319144)

[六、实验总结 10](#_Toc166319145)

一、实验目的及要求

1. 实验目的

通过WT310 电参表仿真设备，学习串行通信接口智能传感设备与计算机之间的硬件互联方法，理解智能传感设备的通信协议，以及通过串行通信调试工具与智能传感设备手动交互的方法，为后续的传感设备数据采集软件编程打下基础。

2. 实验要求

预习传感设备通信协议和串口助手软件的使用方法，通过实验，熟练掌握利用串行通信助手软件进行硬件通信测试的方法，实验过程认真记录，实验完成后撰写实验报告。

二、实验内容

利用仿真器提供的WT310仿真传感设备，通过串行通信线与计算机RS-232 接口连接，在计算机中运行串口调试软件进行通信调试。

三、实验设备及软件

（1）笔记本电脑一台；

（2）仿真器一台；

（3）USB-RS232 转换器两个；

（4）串行通信调试助手软件。

四、通信协议

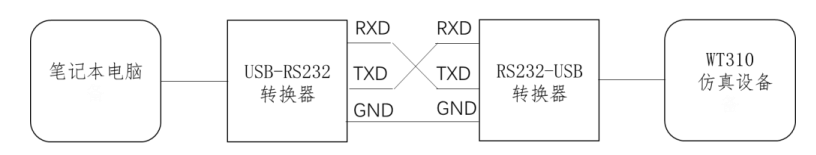
WT310 数据获取命令为**NUMeric:NORMal:VALue?**，命令后面以回车符+换行符（十六进制 0D 0A）为命令结束符。WT310 接收到命令后会返回测试数据，如：

**103.79E+00,1.0143E+00,105.27E+00,50.001E+00,655.27E+00,0.27E+00**

各参数以“,”作为分隔符，分别对应电压、电流、功率、频率、耗电量、功率因数，其数值以科学计数法方式表示（E 表示指数，E 前面的数值乘以 10 的 n 次幂，n为 E 后面三位字符串代表的数值，如：105.27E+00 表示 105.27\*10+00，仍然为 105.27）。

五、实验步骤

（1）WT310 仿真设备硬件连接如图 5.3.1 所示，根据该图将笔记本电脑通过 USB-232 转换器与 WT310 仿真设备进行硬件连接，USB-RS232 转换器外观结构如图 4.3.2 所示。



**图4.3.1** WT310 仿真设备硬件连接

图示

描述已自动生成

**图 4.3.2** USB-RS232 转换器外观结构

（2）打开仿真器桌面上的“琏雾实验系统\实验 5.3”文件夹，运行“wt310”快捷方式，仿真器屏幕显示 WT310 仿真设备，用鼠标点击仿真设备的“power”按钮(电源开关)，仿真设备开始工作，如图 4.3.3 所示：

电脑萤幕

中度可信度描述已自动生成

**图 4.3.3**  WT310 仿真界面

点击 INTERFACE 键，进行通信参数设置，如图 4.3.4 所示，默认设置为数据格式：For0，波特率：9600，结束符：Cr+LF。数据格式有四种，For0(数据位 8；停止位 1；校验位 NONE)；For1(数据位 7；停止位 1；校验位 ODD)；For2(数据位 7；停止位 1；校验位 EVEN)；For3(数据位 7；停止位 2；校验位 NONE)。波特率可以选择 1200、2400、4800、9600、19200、38400 六种。命令结束符有三种，Cr+LF（回车加换行：十六进制0D 0A）；Cr（回车：十六进制0D）；LF（换行：十六进制0A）。

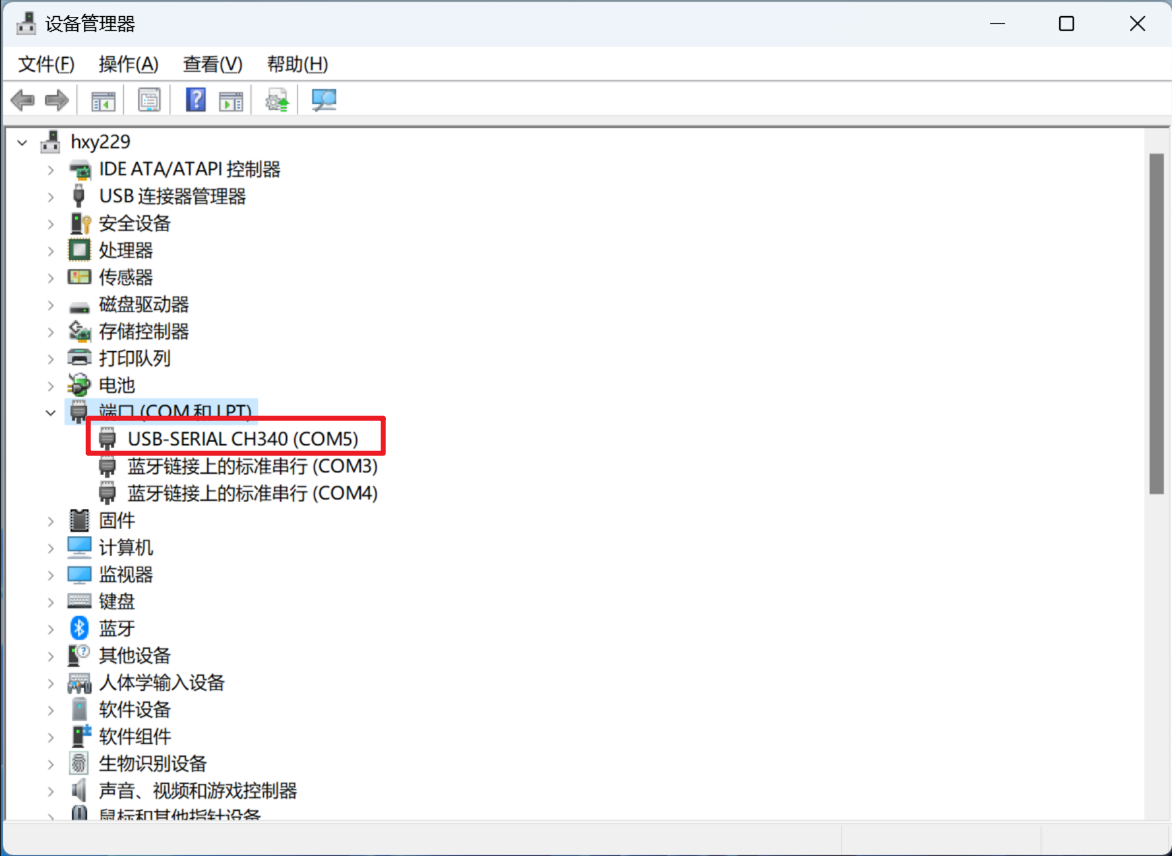
图片包含 日历

描述已自动生成

**图 4.3.4** 仿真 WT310 通信参数设置界面

（3）查看笔记本电脑windows中的设备管理器，确保USB-RS232转换器驱动已经安装，打开计算机桌面上的“琏雾实验系统\工具”文件夹，把“串口助 手”软件拷贝至笔记本电脑并运行。

这里我们还是按照串口助手的连接过程进行，将笔记本电脑通过 USB-232 转换器与 WT310 仿真设备进行硬件连接后，打开笔记本电脑设备管理器查看相应连接端口，如下图4.3.5所示，这里是COM5。



**图 4.3.5** 笔记本电脑上的端口信息

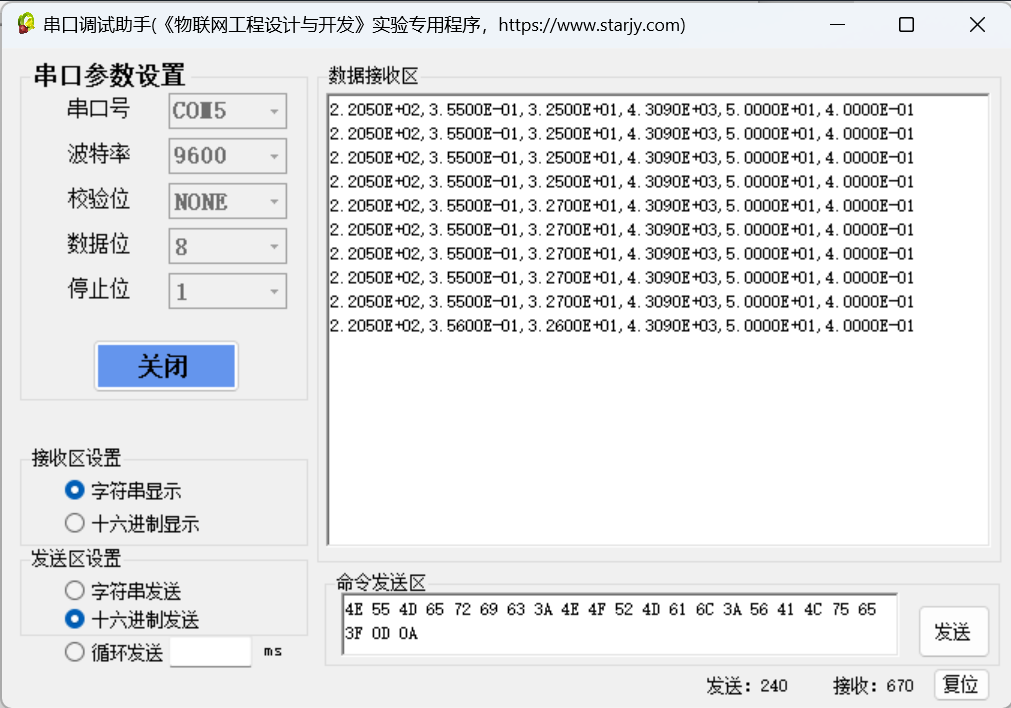
（4）进行串口参数设置。

（5）根据通信协议，在发送区输入获取数据命令。

（6）确认命令的结束符是否正确。即根据WT310电参表的通信协议，在串口助手软件下方的发送窗口，输入数据获取命令:NUMeric:NORMal:VALue?，选择 按十六进制发送，观察命令结束符和在图4.3.3中选择的是否一致，即在图4.3.3 中选择Cr+LF，命令结束符应为0D0A，若在图4.3.3中选择Cr，命令结束符应为 0D，若在图4.3.3中选择LF，命令结束符应为0A，否则要进行相应的修改。实际上，在输入数据获取命令后，若按回车键，则在选择“按十六进制发送”时，命令后面会有0D0A。

（7）利用串口助手与WT310仿真设备通信，获得10组数据。

下面我们打开笔记本电脑上的串口助手进行测试，串口号选择COM5，其余参数与上述仿真 WT310 通信参数一致，然后打开串口。在命令发送区发送命令**NUMeric:NORMal:VALue?**，WT310会给我们返回数据，我们连续向WT310仿真设备发送10次该命令，得到10组返回数据，如下图4.3.6所示。



**图 4.3.6** 测试确保硬件通信线路正常

（8）将获取的10组数据进行解析，分别对应电压、电流、功率、频率、耗电量、功率因数，并将解析的数据填写到WT310数据表中，数据表格式见表4-3-1。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **组号** | **电压** | **电流** | **功率** | **频率** | **耗电量** | **功率因数** |
| **1** |  |  |  |  |  |  |
| **2** |  |  |  |  |  |  |
| **3** |  |  |  |  |  |  |
| **4** |  |  |  |  |  |  |
| **5** |  |  |  |  |  |  |
| **6** |  |  |  |  |  |  |
| **7** |  |  |  |  |  |  |
| **8** |  |  |  |  |  |  |
| **9** |  |  |  |  |  |  |
| **10** |  |  |  |  |  |  |

**表4-3-1** WT310 传感数据记录

五、报告要求

1. wt310电参表的主要用途是什么

WT310电参表的主要用途是测量用电负载的电压、电流、功率以及耗电量等参数。它能够提供准确的电能参数，帮助用户了解系统的工作状态，进行能源管理和电能质量分析。在实验中，通过仿真器提供的WT310仿真传感设备，让学习串行通信接口智能传感设备与计算机之间的硬件互联方法，以及掌握通信协议和串口调试软件的使用，为后续的传感设备数据采集软件编程提供基础。

2. USB-RS232转换器的作用是什么

USB-RS232转换器的作用是将USB接口转换成RS-232串行接口，使得设备可以通过串行通信与计算机进行连接和通信。在实验中，USB-RS232转换器起到了连接笔记本电脑和WT310仿真设备的作用，实现了计算机与仿真设备之间的硬件互联。

3. 为何在笔记本Windows系统设备管理器中会有实验对应的 RS-232 接口

在笔记本Windows系统设备管理器中会有实验对应的RS-232接口，是因为USB-RS232转换器在连接到笔记本电脑后，会被系统识别为一个虚拟的RS-232串口设备。虽然笔记本电脑通常不再直接提供RS-232串口接口，但通过USB接口连接转换器后，系统会模拟出一个虚拟的RS-232接口，使得用户可以通过该接口与外部设备进行串行通信。

4. 把填写10组解析后传感数据的4-3-1表格附在实验报告中

这里我使用Python语言编写数据解析代码将上述图4.3.6中的数据解析出来并保存到相应文本文档“解析后的数据.txt”中，解析的数据在串口助手中得到后复制粘贴至一个文本文档“WT310返回数据.txt”中。下面是代码部分，同时给出代码中函数的解释。

首先是解析主调函数**data\_parse()**，它的作用是从fpath这个路径的文件中逐行读取数据，然后分别对每一行数据调用 line\_parse() 函数与 save() 函数，line\_parse()函数对一行数据进行解析，save()函数将解析后的数据保存至“解析后的数据.txt”中。

1. def data\_parse(fpath): # 解析主调函数
2. with open(fpath, 'r') as file:
3. lines = file.readlines()  # 逐行读取文件，存入lines
4. for dataLine in lines:
5. parsed\_data = line\_parse(dataLine)  # 解析数据
6. save(parsed\_data)  # 保存原始数据与解析数据

下面是以行为单位解析数据的函数**line\_parse()**，首先使用try语句块包裹解析过程，这样就可以随时捕获解析过程中可能发生的异常。然后从串口获取的原始数据并按逗号分割，同时将分割后的字符串列表中的每个元素转换为浮点数。这样就得到了电压、电流、功率、耗电量、频率和功率因数等数据。然后使用datetime模块获取当前的日期时间，并使用strftime('%Y-%m-%d, %H:%M:%S')方法将其格式化为年-月-日, 时:分:秒的字符串格式。最后将解析后的各项数据与对应的单位以及当前时间进行格式化，并组合成一个字符串，表示解析后的数据。最后，将解析后的数据返回给data\_parse()。

1. # 解析数据的函数
2. def main\_parse(data):
3. try:  # 解析数据
4. voltage, current, power, consumption, frequency, power\_factor = map(float, data.split(','))  # 将数据按逗号分割并转换为浮点数
5. current\_time = datetime.datetime.now().strftime('%Y-%m-%d, %H:%M:%S')  # 获取当前时间
6. parsed\_data = (
7. f"{current\_time}: 电压: {voltage} V; 电流: {current} A; 功率: {power} W; 耗电量: {consumption} Wh; 频率: {frequency} Hz; 功率因数: {power\_factor}"
8. )  # 格式化解析后的数据
9. return parsed\_data  # 返回解析后的数据
10. except ValueError:  # 如果解析错误
11. return "解析数据时出错."  # 返回错误信息

下面是保存数据的函数**save()**，他主要是将原始数据和解析后的数据保存到文件中。首先使用try语句块包裹保存数据的操作，这样就可以随时捕获解析过程中可能发生的异常。打开名为“解析后的数据.txt”的文件，将解析后的数据parsed\_data写入文件中。最后，如果在保存数据到文件时发生错误（例如文件操作权限问题），则会捕获ValueError异常，并将表示错误信息的字符串"解析数据时出错"写入名为“解析后的数据.txt”的文件中。

1. # 保存数据到文件的函数
2. def save(parsed\_data):
3. try:  # 保存数据
4. with open('解析后的数据.txt', 'a') as file:
5. file.write(f"{parsed\_data}\n")  # 写入解析后的数据
6. except ValueError:  # 如果保存错误
7. with open('解析后的数据.txt', 'a') as file:
8. file.write("解析数据时出错.\n")  # 写入错误信息

最后直接调用data\_parse()函数进行数据解析的完整过程，这里需要传入相应的原始数据文件的路径信息。

1. file\_path = "E:\ouc\IoT\lab4\lab4.3\dataParse\WT310返回数据.txt"
2. data\_parse(file\_path)

上述代码运行后生成的“解析后的数据.txt”文件中的内容如下图4.3.7所示。

表格

描述已自动生成

**图4.3.7** 解析后的数据文件

下面给出10组解析后传感数据的4-3-1表格。

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **组号** | **电压（V）** | **电流（A）** | **功率（W）** | **频率（Hz）** | **耗电量（Wh）** | **功率因数** |
| **1** | 220.5 | 0.355 | 32.5 | 50.0 | 4309.0 | 0.4 |
| **2** | 220.5 | 0.355 | 32.5 | 50.0 | 4309.0 | 0.4 |
| **3** | 220.5 | 0.355 | 32.5 | 50.0 | 4309.0 | 0.4 |
| **4** | 220.5 | 0.355 | 32.5 | 50.0 | 4309.0 | 0.4 |
| **5** | 220.5 | 0.355 | 32.7 | 50.0 | 4309.0 | 0.4 |
| **6** | 220.5 | 0.355 | 32.7 | 50.0 | 4309.0 | 0.4 |
| **7** | 220.5 | 0.355 | 32.7 | 50.0 | 4309.0 | 0.4 |
| **8** | 220.5 | 0.355 | 32.7 | 50.0 | 4309.0 | 0.4 |
| **9** | 220.5 | 0.355 | 32.7 | 50.0 | 4309.0 | 0.4 |
| **10** | 220.5 | 0.356 | 32.6 | 50.0 | 4309.0 | 0.4 |

**表4-3-1** WT310 传感数据记录

5. 写出调试过程遇到的主要问题和解决方法

* **问题及解决方法1**

**问题：**在进行串口助手与WT310仿真设备通信交互时总是不成功，检测到端口但是端口无法使用，导致不能得到仿真设备的数据返回。在之前做过的实验中也遇到了类似的问题，但是当时的问题是在之前串口助手实验时我自行创建了一对虚拟端口COM5与COM6，这里的COM5正好与实验中与WT310仿真设备相连的COM5冲突，也就是虚拟端口占用了COM5，导致实验的测试不能成功。最后在Configure Virtual Serial Port Driver中删除了先前创建的COM5与COM6端口，再次进行测试就能成功了。这次并没有虚拟串口的占用情况。

**解决方法：** 使用了其他设备重新进行连接，发现通信正常，这里考虑有可能是设备驱动问题，随后尝试下载课程群中助教发的“试验箱USB转485驱动\_PL23XX-M\_LogoDriver\_Setup\_408\_20220725.exe”，发现原先设备也可以进行正常通信，问题得到解决。

六、实验总结

通过本次实验，我深入理解了智能传感设备与计算机之间通过串行通信接口进行硬件互联的方法。我学习了WT310电参表仿真设备的通信协议，并掌握了使用串口助手软件进行硬件通信测试的技巧。实验过程中，我不仅熟悉了串行通信的基本概念，还提高了解决实际通信问题的能力。

通过实际操作，我利用串口助手软件与WT310仿真设备进行10次通信并成功获取数据。然后再对数据进行解析。我使用Python语言编写了数据解析程序，该程序能够从原始数据文件中读取数据，将其解析为电压、电流、功率、频率、耗电量和功率因数等参数，并将解析后的数据保存至文本文档中。

在实验开始前，我的串口助手与WT310仿真设备通信不成功。通过检查和分析，我发现问题可能出在设备驱动上。最终，通过重新安装驱动并使用其他设备进行测试，我成功解决了这个问题。

本次实验不仅让我掌握了智能传感设备的通信协议和串行通信调试工具的使用，还提升了我的问题解决能力和编程能力。为我未来在物联网系统设计与开发课程的学习和研究打下了坚实的基础。