**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：数据采集模块实际设备通信**

**实验时间：2024年 5月 9日 实验教师:郭忠文**

**目录**

[一、实验目的及要求 2](#_Toc166148833)

[1. 实验目的 2](#_Toc166148834)

[2. 实验要求 2](#_Toc166148835)

[二、实验内容 2](#_Toc166148836)

[三、实验设备及软件 2](#_Toc166148837)

[四、通信协议 3](#_Toc166148838)

[五、实验步骤 3](#_Toc166148839)

[六、报告要求 7](#_Toc166148840)

[1. 实验中变送模块的作用是什么 7](#_Toc166148841)

[2. 把填写 10 组解析后传感数据的表格附在实验报告中 7](#_Toc166148842)

[3. 写出调试过程遇到的主要问题和解决方法 9](#_Toc166148843)

[七、实验总结 10](#_Toc166148844)

一、实验目的及要求

1. 实验目的

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

利用试验系统提供的 USB-RS232 转换器，为笔记本电脑扩展 RS-485 接口，通过该转换器与数据采集模块连接。由于该数据采集模块是 4-20mA 标准模拟量输入，利用变送器模块把 PT100 输出的非标准信号转换成 4-20mA 标准信号后，与数据采集模块连接。硬件接线完成后，根据数据采集模块的通信协议，通过串口助手获取 PT100 温度数据，并进行数据解析，换算成温度数据后，进行实验数据记录。

三、实验设备及软件

（1）笔记本电脑一台

（2）数据采集模块一个；

（3）PT100 温度传感器和变送器一套；

（4）DC-24V 电源两个；

（5）USB-RS485 转换器一个；

（6）串行通信调试助手软件。

四、通信协议

数据获取命令为:01 04 00 00 00 08 F1 CC，数据采集模块接收到命令后会返回测试数据，如：01 04 10 06 32 00 00 00 00 00 00 00 00 00 00 00 00 00 00 6A 1F，具体数据解析内容见表 4-1-1。

表格

描述已自动生成

五、实验步骤

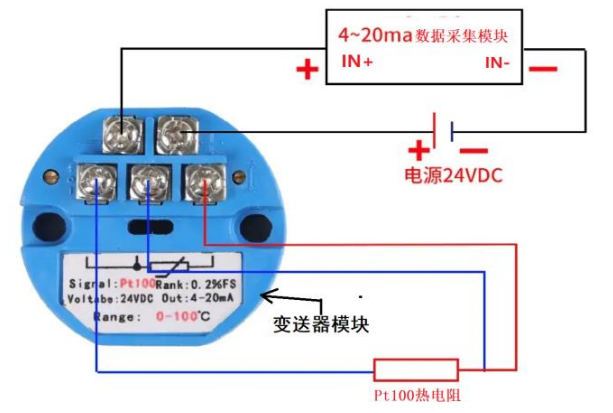
（1）实验系统提供了两个 24V 电源适配器，为了进行电源线连接，根据电 源线引出线路，如图 4.1.1 所示，建立两组电源引出电路。注意：线路全部连接 完毕，检查线路正确后再通电。

图形用户界面

描述已自动生成

**图4.1.1**  电源线引出电路

（2）系统提供了 PT100 与数据采集器硬件连线如图 4.1.2 所示，该图中 Pt100 热敏电阻传感器与变送器模块连接，该模块负责把 Pt100 的输出信号转换成 4- 20mA 的电流信号，与数据采集模块连接。采用实验系统提供的配件，根据图 4.1.2 进行硬件线路连接。数据采集模块共有 8 个输入通道，既（IN0+,IN0-）至（IN7+,IN7-） 选择任一通道连接都可以。在连接线路中需要串接一个 24V 直流电源。Pt100 温 度传感器外观如图 4.1.3 所示。



**图 4.1.2** Pt100 与数据采集器硬件连线

形状

描述已自动生成

**图 4.1.3** Pt100 传感器外观

（3）在笔记本电脑 Windows 系统中利用设备管理器，查看是否有需要使用 的串口，确认 USB-RS485 转换器驱动已经安装。RS-485 转换器的外观结构如图 4.1.4 所示，其中的 5V 电源在连接 485 接口设备较多，电路驱动能力不足时需要 连接，连接设备数量较少时可以不接 5V 电源。

图示, 工程绘图

描述已自动生成

**图 4.1.4** RS-485 转换器的外观结构

（4）数据采集模块外观如图 4.1.5 所示。笔记本电脑与数据采集模块连线 如图 4.1.6 所示，根据该图连接，利用过 USB-RS485 转换器连接笔记本电脑利用 如图 4.1.1 所示的第 2 组电源线路连接数据采集模块电源 VIN 和 GND。

图示

描述已自动生成

**图 4.1.5** 数据采集模块外观

图示

低可信度描述已自动生成

**图 4.1.6** 笔记本电脑与数据采集模块硬件连线

（5）打开计算机桌面上的“琏雾实验系统\工具”文件夹中 “串口助手”软 件，串口参数波特率选择 9600、校验位选择 NONE、数据位选择 8、停止位选择 1，接收区和发送区选择十六进制，串口设置后输入命令出现串口助手界面如图 4.1.7 所示。

图片包含 图形用户界面

描述已自动生成

**图 4.1.7** 串口助手界面

（6）利用串口调试助手进行通信调试，获得 PT100 不同温度下的 10 组数据。

（7）将获取的 10 组数据根据温度变送器量程 0-100℃、4-20mA 和数据采集 模块电流输入值计算公式进行解析，数据换算公式如下：

所求值 = 通道数据 X (量程上限 - 量程下限) / 0xFFFF + 量程下限；

例如： 通道数据是 0xCDD2(52690)，电流值量程是 0~20mA； 电流值 = 52690 x (20-0) / 65535 + 0 = 16.079mA。 通道数据是 0xCDD2(52690)，温度值量程是 0~100℃； 温度值 = 52690 x (100-0) / 65535 + 0 = 80.395℃。 将解析的数据填写到 TAS-RTU-432 数据表中，数据表格式见表 4-1-2。

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 通道值（十六进制） | 电流输入值（mA） | 测量温度值（℃） |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |
| 4 |  |  |  |
| 5 |  |  |  |
| 6 |  |  |  |
| 7 |  |  |  |
| 8 |  |  |  |
| 9 |  |  |  |
| 10 |  |  |  |

**表 4-1-2**  传感数据记录

六、报告要求

1. 实验中变送模块的作用是什么

变送模块将 PT100 温度传感器输出的非标准信号转换成 4-20mA 的标准信号。因为PT100 温度传感器输出的信号可能是非标准的电压或电阻值，通过变送模块的转换，将其转化为标准的 4-20mA 电流信号。这样的标准信号更适合于长距离传输和抗干扰能力更强，能够提高数据采集的稳定性和精度。同时，4-20mA 信号也是工业领域常用的一种标准信号格式，便于与其他设备连接和集成。

2. 把填写 10 组解析后传感数据的表格附在实验报告中

对于上述图4.1.7中的返回数据（即源数据）并结合步骤（7）中的数据解析换算公式，我使用Python实现了相关解析逻辑，解析代码如下所示。

1. def data\_parse(file\_path):
2. with open(file\_path, 'r') as file:  # 打开文件
3. lines = file.readlines()  # 读取文件的所有行并存储在列表 'lines' 中
4. i = 1
5. for line in lines:  # 遍历文件中的每一行
6. line = line.split()  # 将行按照空格拆分为字符串列表
7. hexstring = line[3] + line[4]  # 提取有效数据
8. num = hex\_string\_to\_hex\_number(hexstring)  # 将十六进制字符串转换为十六进制数值
9. I = (num \* 16) / 65535 + 4  # 使用给定的公式计算电流值 (I)
10. T = (num \* 100) / 65535 + 0  # 使用给定的公式计算温度值 (T)
11. print(f"组号{i}")
12. print("通道数据为：0x" + hexstring)
13. print("电流值为：" + '{:.3f}'.format(I))
14. print("温度值为：" + '{:.3f}'.format(T))
15. print()
16. i = i + 1
17. # 将十六进制字符串转换为十六进制数值
18. def hex\_string\_to\_hex\_number(hex\_string):
19. hex\_number = 0
20. power = len(hex\_string) - 1  # 字符串长度
21. for digit in hex\_string:
22. if digit.isdigit():  # 如果字符是数字，则将其转换为对应的十进制数值
23. hex\_number += int(digit) \* (16 \*\* power)
24. else:
25. # 如果字符是字母，则将其转换为对应的十进制数值
26. hex\_number += (ord(digit.upper()) - ord('A') + 10) \* (16 \*\* power)
27. power -= 1
28. return hex\_number
29. file\_path = "E:\ouc\IoT\lab4\lab4.1\dataParse\PT100数据返回.txt"
30. data\_parse(file\_path)

相关**源数据**如下：

01 04 10 04 73 00 00 00 00 00 00 00 00 00 00 00 00 00 00 64 3F

01 04 10 04 3F 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3F 3F

01 04 10 05 0C 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3F 7A

01 04 10 05 19 00 00 00 00 00 00 00 00 00 00 00 00 00 00 4F 75

01 04 10 05 26 00 00 00 00 00 00 00 00 00 00 00 00 00 00 30 65

01 04 10 05 28 00 00 00 00 00 00 00 00 00 00 00 00 00 00 3F 61

01 04 10 05 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 67 6B

01 04 10 05 36 00 00 00 00 00 00 00 00 00 00 00 00 00 00 20 69

01 04 10 05 41 00 00 00 00 00 00 00 00 00 00 00 00 00 00 17 4F

01 04 10 05 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 38 63

通过上述代码解析后的输出结果如下：

文本, 信件

描述已自动生成文本, 信件

描述已自动生成 文本, 信件

描述已自动生成 文本, 信件

描述已自动生成 文本, 信件

描述已自动生成 文本, 信件

描述已自动生成 文本, 信件

描述已自动生成 文本, 信件

描述已自动生成 文本, 信件

描述已自动生成 文本, 信件

描述已自动生成

下面填写表4-1-2：

|  |  |  |  |
| --- | --- | --- | --- |
| 组号 | 通道值（十六进制） | 电流输入值（mA） | 测量温度值（℃） |
| 1 | 0x0473 | 4.278 | 1.738 |
| 2 | 0x043F | 4.265 | 1.659 |
| 3 | 0x050C | 4.315 | 1.971 |
| 4 | 0x0519 | 4.319 | 1.991 |
| 5 | 0x0526 | 4.322 | 2.011 |
| 6 | 0x0528 | 4.322 | 2.014 |
| 7 | 0x0531 | 4.324 | 2.028 |
| 8 | 0x0536 | 4.326 | 2.036 |
| 9 | 0x0541 | 4.328 | 2.052 |
| 10 | 0x052E | 4.324 | 2.023 |

**表 4-1-2**  传感数据记录

3. 写出调试过程遇到的主要问题和解决方法

* **串口通信设置问题**：在设置串口助手软件时，没有正确配置波特率和串行通信参数，导致无法与数据采集模块建立连接。通过仔细检查实验指导书，我重新设置了波特率为9600，数据位为8，停止位为1，无校验位，并确保发送和接收区都选择了十六进制格式，最终成功建立了通信。
* **数据解析错误**：在使用串口助手软件获取数据后，我手动解析数据时发现很麻烦也很容易出错。通过编写Python代码来自动化解析过程，更准确地转换和计算出电流值和温度值。在编写代码时，我遇到了一些简单的逻辑错误，但通过调试和修正代码，最终实现了数据的准确解析。

七、实验总结

本次数据采集模块实际设备通信实验不仅加深了我对物联网系统中数据采集模块通信协议的理解，而且提升了我解决实际问题的能力。在实验过程中，巩固了之前实验中使用串行通信助手软件进行硬件通信测试的步骤，并实现了智能传感设备与计算机之间的硬件互联。

在实验过程中，我遇到了包括串口通信设置、硬件连接、数据解析等在内的多个问题。通过不断尝试和调整，最终问题得到解决，这是一次很好的锻炼。为了自动化数据解析过程，我还编写了Python代码来实现相关解析逻辑。在编写和调试代码的过程中，不仅需要实现相关解析逻辑，也要完全符合通信协议。

本次实验经历为我未来在物联网领域的学习和研究打下了坚实的基础。为我在物联网课程的继续探索和学习添砖加瓦，希望在未来的项目和工作中能够运用所学知识和技能，解决更复杂的问题。