

《基础物理实验》实验报告

实验名称 _____ 虚拟仪器 _____ 指导教师 _____ 李贞杰 _____
姓名 _____ 徐博涵 _____ 学号 _____ 2023K8009908004 _____ 分班分组及座号 _____ 1-04-6 号 (例: 1-04-5 号) _____
实验日期 _____ 2024 _____ 年 _____ 11 _____ 月 _____ 11 _____ 日 实验地点 _____ 702 _____ 调课/补课 ☐ 是 _____ 成绩评定 _____

1 实验目的

1. 了解虚拟仪器的概念;
2. 了解图形化编程语言 LabVIEW, 学习简单的 LabVIEW 编程;
3. 完成伏安法测电阻的虚拟仪器设计.

2 实验器材

计算机 (含 Windows 操作系统), LabVIEW 2014, NI ELVIS II+, 导线若干, 元件盒一个 (包括 $100\ \Omega$ 标准电阻一个, 待测电阻 $100\ \Omega$ 和 $20\ \Omega$ 各一个, 稳压二极管一个)

3 实验原理

3.1 虚拟仪器的硬件和软件

本实验使用的硬件平台是个人电脑 (PC 机), 美国国家仪器公司 (National Instruments) 的教学实验室虚拟仪器套件 (Educational Laboratory Virtual Instrumentation Suite) II+ (缩写为 NI ELVISII+) 和自带的原型板.

本实验使用的用于虚拟仪器系统设计的软件开发平台是 LabVIEW (laboratory virtual instrument engineering workbench). 它将计算机数据分析和显示能力与仪器驱动程序整合在一起, 为针对仪器的编程提供了很大的便利. 而且, LabVIEW 是一种图形化编程语言, 编程过程也就是设计流程图, 即使初学者也能很快入门.

用 LabVIEW 开发平台编制的虚拟仪器程序简称为 VI. VI 包括三个部分: 前面板 (front panel)、控制框图 (Block diagram) 和图标/连线板. 前面板用于设置输入数值和显示输出量, 相当于真实仪表的前面板. 前面板上的图标, 分为两类: 输入类 (Controls, 用于输入) 和显示类 (Indicators, 用于输出), 如开关、旋钮、按钮、图形、图表等. 控制框图相当于仪器的内部功能结构, 其中的端口用来和前面板的输入对象和显示对象传递数据, 节点用来实现函数和功能子程序调用, 图框用来实现结构化程序控制命令, 连线则代表程序执行过程中的数据流.

3.2 创建一个温度测量程序

创建一个模拟温度测量程序. 假设有一个传感器, 其输出电压和温度成正比, 用它编写一个模拟温度测量的程序. 假设当温度为 80°F 时, 传感器输出电压为 0.8V , 那么我们可以编写程序, 根据电压计算温度, 并且给出摄氏度和华氏度两种显示.

3.3 创建一个电压输出和采集的程序

本实验通过编写输出/输入通道, 两个停止按钮, 可以手动调整的输出电压与测量到的电压, 连接电路, 使用 While 循环使得程序每 100ms 输出/测量一次电压, 并且在电路板上连接好两根导线, 得到的结果就可以随时返回前面板.

3.4 利用虚拟仪器测量伏安特性

本实验中利用一个模拟输出通道为整个测量电路供电, 利用两个模拟输入通道分别测量总电压和标准电阻上的电压; 利用测量得到的电压数值和标准电阻数值就可以得到电路中的电流以及待测电阻上的电压. 在程序控制下, 电路电压由 0V 开始逐渐增加到设定电压, 电压每改变一次, 测得一组电压电流值, 最后得到一个数组, 经过线性拟合后就可以得到待测电阻值. 测量原理图如图 1 所示:

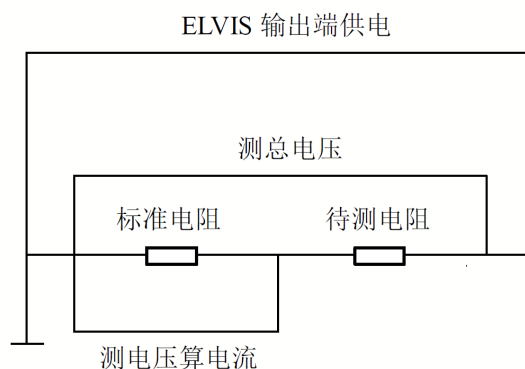


图 1: 伏安特性曲线测量原理图

4 实验内容

4.1 初步熟悉 LabVIEW 的开发环境

打开 LabVIEW 2014; 在“文件”菜单中选择“新建 V1”, 我们可以看到前面板和控制框图. 直接使用快捷键 Ctrl + T 并排两个窗口, 方便编程. 或者点击窗口, 可以实现二者之间的切换. 在前面板的“查看”菜单中打开“控件选板”和“工具选板”, 可以从“工具选板”中选择“自动选择”工具, 方便操作. 可以在“控件选板”中新建“温度计”并将其显示出来.

在控制框图窗口中, 在“查看”菜单中打开“函数选板”来显示函数选板, 利用“函数选板”新建“加法”, 并且尝试为之连线.

在下列实验开始之前, 打开面板上的两个开关, 即 ELVIS 电源 (在仪器后面) 和原型板电源 (在仪器上面的右上方). 并学习选择并放置控件、点击右键查看快捷菜单, 学习使用标签工具、定位工具、连线工具, 熟悉各种快捷键.

注意:

- 由于程序特性, 不要将上一个同学做好的程序直接复制进自己的程序, 可能会出现无法运行的情况
- 如果找不到对应的按键, 可以直接利用搜索框输入名称查找

4.2 创建一个模拟温度测量程序

(1) 创建前面板

依次放入温度计 (控件 - 数值 - 温度计)、垂直滑动杆开关 (控件 - 布尔 - 垂直滑动杆开关, 并改名为温度值单位, 显示开关状态, 继续使用标签工具使其显示“摄氏”与“华氏”)、数值显示控件 (控件 - 数值 - 数值显示控件, 并改名为温度值)、数值输入控件 (控件 - 数值 - 数值输入控件, 并改名为采集的电压)

(2) 创建控制框图

利用控制框图在函数选板中放入乘法函数、减法函数、除法函数（函数 - 数值）、选择函数（函数 - 比较, 并且设置根据温标选择开关的值输出华氏温度或者摄氏温度数值）、移动位置并利用连线工具连接起来, 并在需要的地方创建数值常量. 可以再整理一下图标位置和连线.

值得注意的是, 这里由于选择函数的特性, “温度计” 和 “温度值” 位置不能调换.

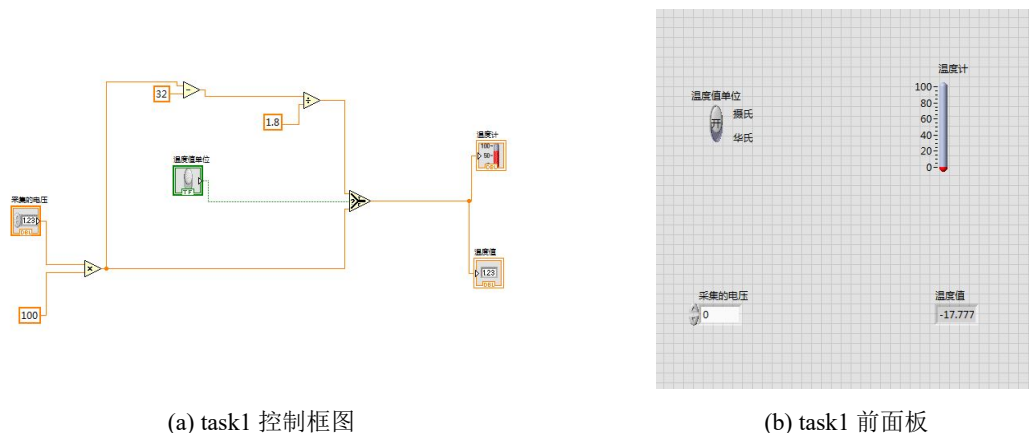


图 2: task1

(3) 运行程序

运行 VI 程序, 点击连续运行按钮, 使程序运行于连续运行模式. 改变 “采集电压” 控件输入值 (比如在 0.5 2.0 之间的任意值) 和温度值单位, 观察程序运行情况, 并解释程序每部分的功能. 停止运行, 并保存即可.

4.3 创建一个电压输出和采集的程序

(1) 新建空白 VI. 操作程序框图窗口.

(2) 对于输入部分, 我们需要进行如下操作: 新建 “DAQmx 创建虚拟通道” (测量 I/O → DAQmx - 数据采集 → DAQmx 创建虚拟通道), 选择模拟输入电压. 在其 “物理通道” 接口处右键单击, 创建 “输入控件”. 新建 “DAQmx - 数据采集”, “DAQmx 读取”, “DAQmx 清除任务”, 并在 “DAQmx 读取” 的 “数据” 接口处右键单击, 创建 “显示控件”. 新建 “While 循环”, “等待”, 在 “等待” 的 “等待时间 ms” 接口处右键单击, 创建 “常量”, 设为 100, 在 “结束条件” 处右键单击, 创建停止. 调整位置, 连线, 更改标签.

(3) 输出部分与之相似, 不同之处有: 选择: “DAQmx 创建虚拟通道” 选择模拟输入电压; 新建 “DAQmx 写入” 而非 “DAQmx 读取”; 在 “DAQmx 写入” 的 “数据” 接口处右键单击, 创建 “输入控件”.

(4) 在前面板中, 则需要修改标签, 并调整图表位置. 而对于实验所使用的 ELVIS 仪器面板, 只需要在打开两个电源后, 将 AI 0+ 与 AO 0, AI 0- 与 AIGND 用导线连接即可. 然后即可对输出通道/输入通道分别选择 Dev3/ao0 和 Dev3/ai0 改变输出电压, 观察测量电压. 最后保存并关闭文件.

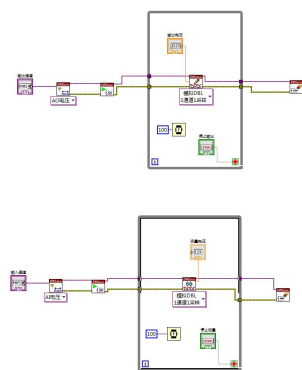
此部分的控制框图和前面板见图 3.

4.4 用虚拟仪器测量伏安特性

(1) 新建 VI 文件, 在前面板新建 “Express XY 图”, 修改标签, 选用 “点加线” 模式, 将横坐标和纵坐标标签分别修改为 “电流 (A)” 和 “电压 (V)”. 新建 “数值输入控件” × 4, 修改标签并设置单位; 新建 “数值显示控件”, 修改标签; 新建 “开关按钮”, “数值显示控件”, “数组”, 拖拽使数组成为 2 × 20 以上大小, 将 “数值显示控件” 拖放至数组框内并拖拽成 2 个.

(2) 编写程序框图的步骤繁琐, 不再赘述,

这部分程序框图的搭建尤其需要细心, 功能和连线端口等选择容易出错. 完成的控制框图见图 4, 前面板见图 5



(a) task2 控制框图



(b) task2 前面板

图 3: task2

(3) 接下来需要连接外部电路, 连接好原型板上导线和电阻即可, 其中的蓝色电阻可更换, 白色电阻为 100Ω 的标准电阻. 连接完之后即可运行程序, 每次测量只需要运行一次程序即可. 在数组箭头处按右键可以将数据导出为 Excel. 在每次实验后更换电阻/稳压二极管 (正反), 得到四项数据, 导出并存储即可.

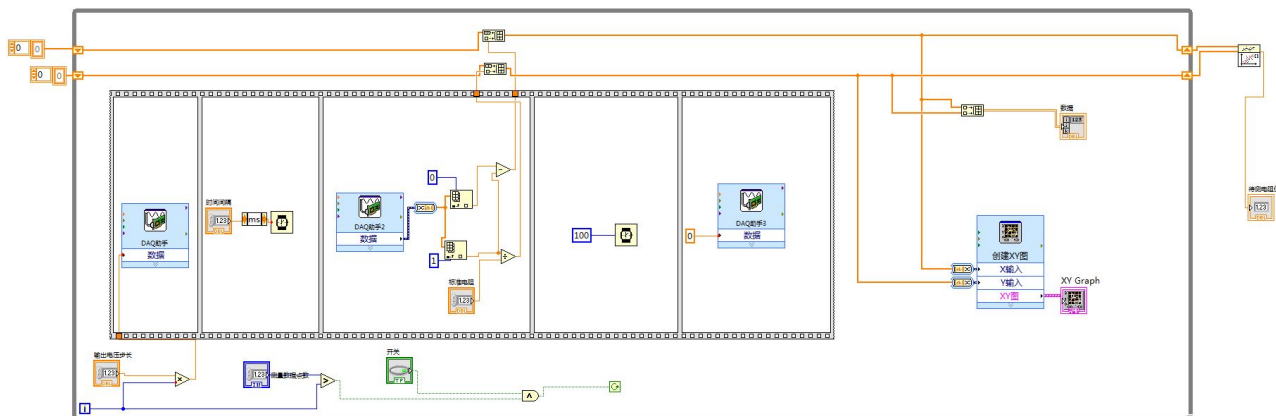


图 4: task3 控制框图

在这个实验中我遇到了不少问题:

- 在添加“数组”时, 该组件横向有两个可以拖拽的点, 一个可以改变单个数组框大小, 另一个可以增加数组列数. 我一开始只拖拽了改变单个数组框大小的那一个, 没有找到另一个. 在请教了同学后才改正.
- 控制框图极易连错: 由于某些组件的接口过多, 我一开始将测量斜率和测量截距的接口连错, 导致测量结果非常奇怪.
- “DAQ 助手”中不同的选项容易搞混, 我初始将两个选项的顺序搞混, 导致测量出的电阻值为负数.
- 与之前需要连续运行的程序不同, 这个程序仅需运行一次即可. 因此, 在选择运行时注意选择单次运行, 如果选连续运行的话无法得到稳定的数据 (程序一直在循环测量)
- 电路板上的电路图接错, 将本该连在标准电阻两端的电压表连在待测电阻两端. 当测量 100Ω 电阻时, 由于两电阻均为 100Ω , 测量结果仍为 100Ω , 但是当测量其他阻值的电阻时就会出错. 这一异常情况困扰了我接近 20 分钟, 询问老师后才得到解决.

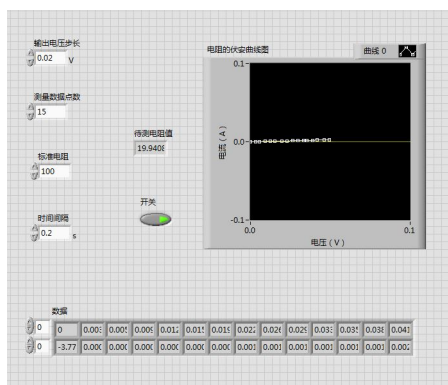


图 6: task3 20 欧前面板

测量数据如表 2

电压/V	电流/A
0	-3.78E-06
0.00321756	0.000163538
0.00579161	0.000330851
0.00933093	0.000501382
0.0125485	0.00066226
0.0151225	0.000829573
0.0196271	0.000987234
0.0228447	0.00115455
0.026384	0.00131864
0.0296016	0.00148596
0.0331409	0.00165327
0.0353932	0.00182702
0.0389325	0.0019879
0.0418283	0.00215521
0.0463329	0.00231287
0.0492287	0.00248018

表 2: 20Ω 电阻数据表

将数据输入 mathematica 绘图，得到图 7，拟合得出电阻为 19.96Ω.

5.4.2 100Ω 电阻

测量 100Ω 电阻时的前面板如图 8所示

测量数据见表 3.

将数据输入 mathematica 绘图，得到图 9，拟合得出电阻为 94.16Ω.

注意到这个拟合结果与标定值 100Ω 差距较大，再对比数据可以发现至少有 9 个电压值分布在 0.95 ~ 1.05 这一区间. 推测在实验时可能不小心做了某些操作，导致电压不能超过 1V，而此处输出的电压步长为 0.15V，因此很快达到最大值，这些堆积的数据点导致了测量的偏离.

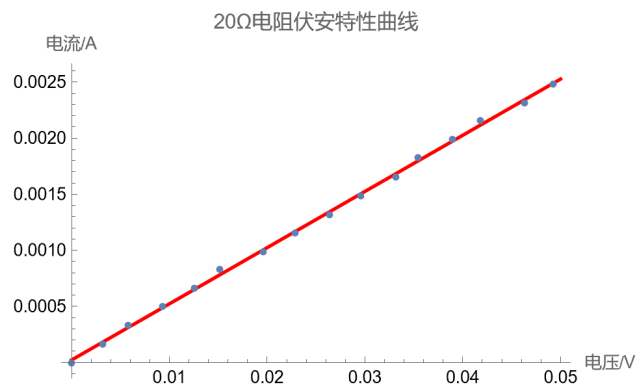


图 7: 20Ω 电阻伏安特性曲线

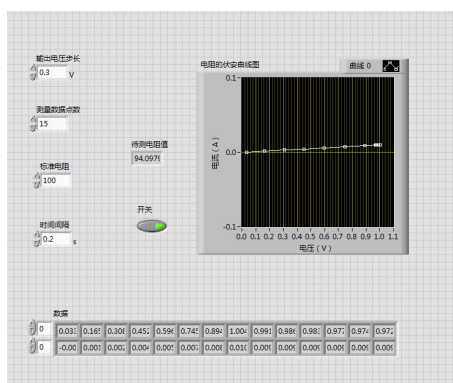


图 8: task3 100 欧前面板

电压/V	电流/A
0.0337844	-0.000351272
0.165383	0.00132186
0.308886	0.00287594
0.452068	0.00443003
0.596859	0.0059648
0.745511	0.00746741
0.894485	0.00896679
1.00453	0.0100833
0.991656	0.00999642
0.986186	0.00993206
0.98329	0.0098838
0.97782	0.00988058
0.974602	0.00987737
0.972672	0.00986771
0.970741	0.00986771
0.968167	0.00987737

表 3: 100Ω 电阻数据表

5.4.3 二极管

二极管测量时的前面板见图 10

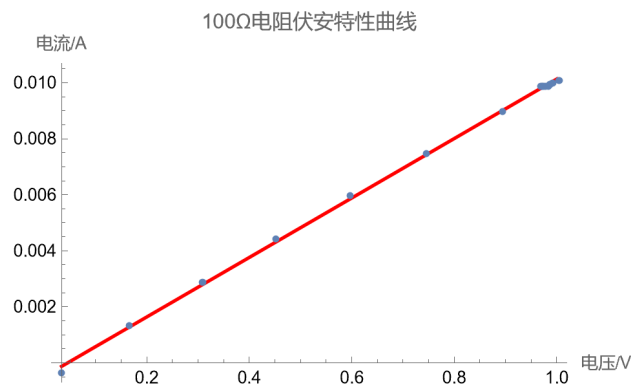
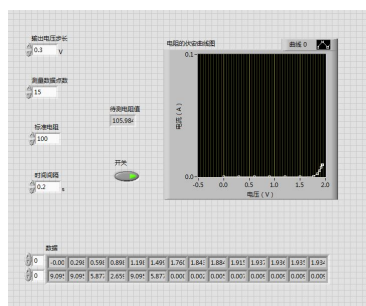
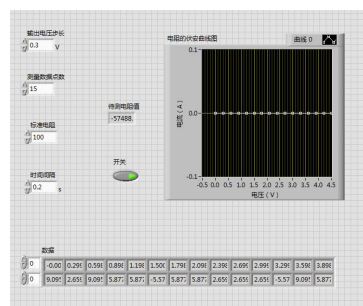


图 9: 100Ω 电阻伏安特性曲线



(a) 二极管正向前面板



(b) 二极管负向前面板

图 10: 二极管前面板

二极管的数据表见表 4

负向		正向	
电压/V	电流/A	电压/V	电流/A
-0.000965269	9.10E-06	-0.00160878	9.10E-06
0.299233	2.66E-06	0.29859	9.10E-06
0.598145	9.10E-06	0.598467	5.88E-06
0.898666	5.88E-06	0.898344	2.66E-06
1.19886	5.88E-06	1.19854	9.10E-06
1.50003	-5.58E-07	1.49939	5.88E-06
1.79862	5.88E-06	1.76097	0.000382
2.09882	5.88E-06	1.84302	0.002535
2.3987	2.66E-06	1.88453	0.005096
2.69922	2.66E-06	1.91574	0.007763
2.99942	2.66E-06	1.93795	0.009958
3.2993	-5.58E-07	1.93602	0.0098484
3.59854	9.10E-06	1.93537	0.009787
3.89874	5.88E-06	1.93408	0.009749
4.1983	9.10E-06	1.93376	0.00972
4.49882	2.66E-06	1.93312	0.009697

表 4: 二极管数据表

将数据输入 mathematica 绘图，得到图 11

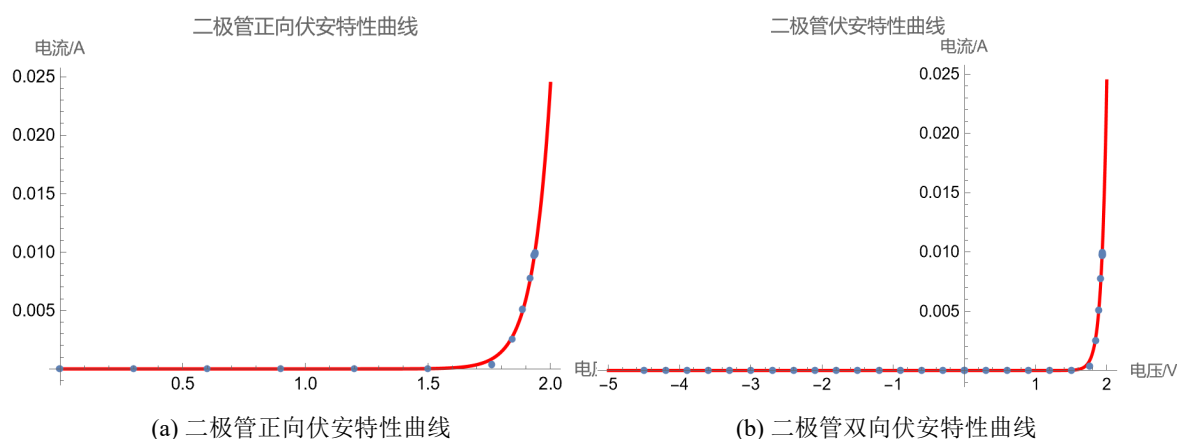


图 11: 二极管伏安特性曲线

6 实验感想

这个实验我接触了虚拟仪器，尝试用 LabVIEW 进行编程，创建了模拟温度测量的程序，电压输出和采集的程序，并利用虚拟仪器测量了多个元件的伏安特性曲线。

我自己在实验过程中遇到了不少问题，例如在 4.4 中和 5.4.2 中提到的，这些问题多是由于我没有理解实验内容，仅仅依照讲义一步一步搭程序而忽略了某些点造成的。不过，幸运的是，在老师和同学的帮助下，我还是收集到了数据。

我认为虚拟仪器在未来的科研中也是很有意义的：

- 通过计算机程序控制测量，可以大幅节省重复性工作的时间。
- 引入计算机程序后，整个实验的可重复性提高，便于他人复现实验结果。
- 传统仪器的测量按钮等都由厂家定义好，在虚拟仪器中可以自己定义测量函数，提升效率。

7 思考题

1. 虚拟仪器系统与传统仪器有什么区别？请简要说明。

- 传统仪器需要手工操作，虚拟仪器仅需在电脑上编程操作。
- 虚拟仪器是由计算机硬件资源和软件组成的测控系统，用户可以自定义仪器功能，而传统仪器的功能是固定且由厂商定义的。
- 虚拟仪器利用计算机强大的软件资源，具有强大的信号处理能力，而传统仪器的信号处理能力相对有限。
- 虚拟仪器的功能、性能、指标可以根据用户需求通过软件定义，且可以由后续需要进一步升级，而传统仪器一经设计、制造完成后，就很难改变。

2. 本实验内容 3 中的电压输出和采集哪个先执行？

由于没有指明哪个先执行，因此两者应当并行，即电压输出和采集同时执行。

附录

A mathematica 拟合绘图代码

修改输入数据和绘图范围即可得到不同图像.

```
data = {{0.0337844`, -0.000351272`}, {0.165383`,  
      0.00132186`}, {0.308886`, 0.00287594`}, {0.452068`,  
      0.00443003`}, {0.596859`, 0.0059648`}, {0.745511`,  
      0.00746741`}, {0.894485`, 0.00896679`}, {1.00453`,  
      0.0100833`}, {0.991656`, 0.00999642`}, {0.986186`,  
      0.00993206`}, {0.98329`, 0.0098838`}, {0.97782`,  
      0.00988058`}, {0.974602`, 0.00987737`}, {0.972672`,  
      0.00986771`}, {0.970741`, 0.00986771`}, {0.968167`,  
      0.00987737`}};  
fitpara2 = FindFit[data, a x + b, {a, b}, x]  
1/a /. fitpara2  
fitted2 = a x + b /. fitpara2 ;  
  
Show[{Plot[fitted2, {x, 0.0337844`, 1},  
  AxesLabel -> {"电压/mV", "电流/mA"}, PlotStyle -> Red,  
  PlotLabel -> "100\[[CapitalOmega]\]电阻伏安特性曲线"  
  , ListPlot[data]]}
```

绘制二极管图像的代码略有不同

```
ClearAll["Global`*"]  
rawdata1 =  
Import["D:\\UCAS\\LaTeX\\phys_ex\\virtual\\task3 二极管正向.csv"] //  
ToExpression;  
rawdata2 =  
Import["D:\\UCAS\\LaTeX\\phys_ex\\virtual\\task3 二极管负向.csv"] //  
ToExpression;  
data1 = Transpose[rawdata1];  
data2 = Transpose[rawdata2];  
data2[[All, 1]] = -data2[[All, 1]];  
data = Join[data1]  
fitPara = FindFit[data, Exp[a*x + b], {a, b}, x];  
fitted = Exp[a*x + b] /. fitPara;  
Show[Plot[fitted, {x, 0, 2}, AxesLabel -> {"电压/V", "电流/A"},  
PlotStyle -> Red, PlotLabel -> "二极管正向伏安特性曲线", PlotRange -> All],  
ListPlot[data]]
```