

《基础物理实验》实验报告

实验名称 虚拟仪器在物理实验中的应用 指导教师 李凡
姓 名 唐嘉良 学号 2020K8009907032 分班分组及座号 4 -04 -8 号 (例: 1-04-5 号)
实验日期 2021 年 10 月 21 日 实验地点 702 调课/补课 ☐ 是 成绩评定 _____

一、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。

二、实验仪器

计算机(含操作系统), LabVIEW2014, NI ELVIS II⁺, 导线若干, 元件盒(包括 100Ω 标准电阻一个, 待测电阻 10Ω 和 100Ω 各一个, 稳压二极管一个)。

三、实验原理

1. 虚拟仪器的硬件

本实验使用的硬件平台是个人电脑(PC机), 美国国家仪器公司的教学实验室虚拟仪器套件 NI ELVIS II⁺ 和自带的原型版。

虚拟仪器综合实验平台 ELVIS II⁺, 如图 1 所示, 集成 8 路差分输入(或 16 路单端输入)模拟数据采集通道、24 路数字 I/O 通道, 以及多款常用的仪器。平台通过 USB 连接 PC。虚拟仪器综合实验平台是开源的, 可以在 LabVIEW 中进行定制, 同时可以使用 LabVIEW Express VI 和 LabVIEW Signal Express 的步骤对设备进行编程。

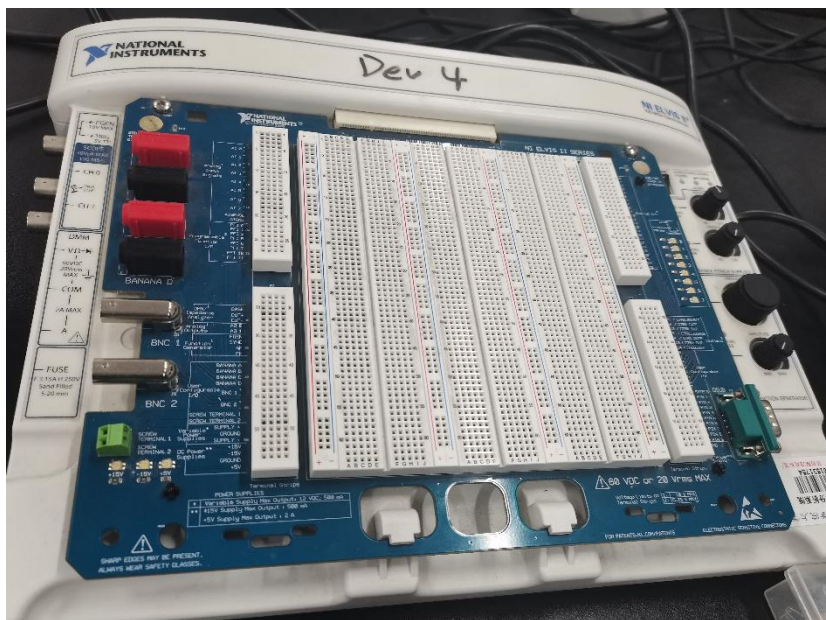


图 1 NI ELVIS II 和原型板实物图

本次实验中使用到的原型版的功能有: 模拟信号输入和可编程函数信号 I/O 阵列, 数字万用表、函数发生器、用户可配置 I/O、可变电源和直流电源阵列。

2. 虚拟仪器的软件

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。LabVIEW 是一种图形化编程语言，编程过程也就是设计流程图。

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、程序框图和图标/连线板。

2.1 前面板。前面板用于设置输入数值和显示输出量，是展示给用户的界面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（用于输入）和显示类（用于输出），具体可以是开关、旋钮、按钮、图形、图表等表现形式。

2.2 程序框图。程序框图相当于仪器的内部功能结构，类比程序设计则是程序员为实现功能而设计的代码。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。

2.3 图框和连线板。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. 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。测量原理见图 2。使用单端输入方式时，各个输入通道共用地线，各通道测量的都是对地的电压，连线时要加以注意。

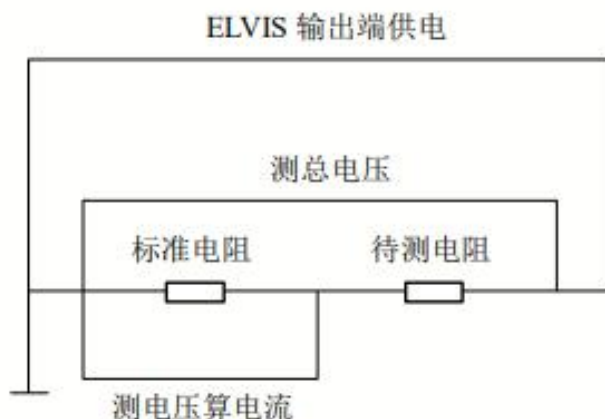


图 2 用虚拟仪器测量伏安特性原理图

四. 实验内容与步骤

1. 初步熟悉 LabVIEW 开发环境的基本操作和编程方法

打开 LabVIEW，新建.vi 文件进入开发环境界面，就看到前面板和程序框图两个窗口。

此时有两种便捷的处理方法：一是通过 Alt+Tab 快捷键不断切屏以达到对比校验的效果，二是通过 Ctrl+T 键来让两个窗口并排显示。

在前面板处，可以通过控件选板来选择我们需要的控件。控件主要分为输入控件和显示控件两种。前者允许用户输入数据，后者只允许查看。用鼠标拖拽可以把控件放在自己想要的位置，还可以对标签等文字内容进行编辑。与此同时，相应的端子和图标会出现在程序框图上，这是一个同步过程。前面板的工具栏有一个向右的箭头是运行程序的按键，红点是终止运行，还有一个暂停运行的按键——这类似我们常见的编程运行调试过程。

我们在程序框图中要对框图、连线进行设计，以实现所需的功能。显示函数选板，可以看到其中提供了很多框架，如编程需要的运行结构（for 循环、while 循环、顺序结构等），数组（创建数组等），比较、数值运算等。拖拽以放到需要的位置。打开工具选板，可以进行选择、连线、编辑文字的功能，软件也会根据预判来自动调节执行的功能。



图 3 LabVIEW 界面

发现一些细节：输入端的边框颜色甚至连线的颜色都可以反映传输数据的数据类型。while 等各种类型的循环是先执行后判断，循环体需要将器件图形拖拽放入其中才能生效，这与我们之前对编程的认识不太一样。

2. 创建一个模拟温度测量程序

我们可以用温度传感器来测量温度，把输出的电压转化成对应的摄氏、华氏温度。然而，

现在我们没有传感器，通过手动输入采集的电压，并假设电压与温度成正比，依然可以通过虚拟仪器来测得温度。

前面板中需要有温度单位切换的开关，采集的电压输入框，温度值的显示框，温度计这几个控件。我们设置摄氏度单位为 Boolean 值 1。于是在程序框图的设计中，需要添加条件选择运算符，Boolean 值为 1 时运行的是摄氏度输出路径，Boolean 值为 0 时运行华氏度输出路径。华氏度的输出值把采集的电压乘 100 即可，摄氏度的值则是把华氏度的值减 32 后除以 1.8 得到。

以下是我在本实验的前面板和程序框图：

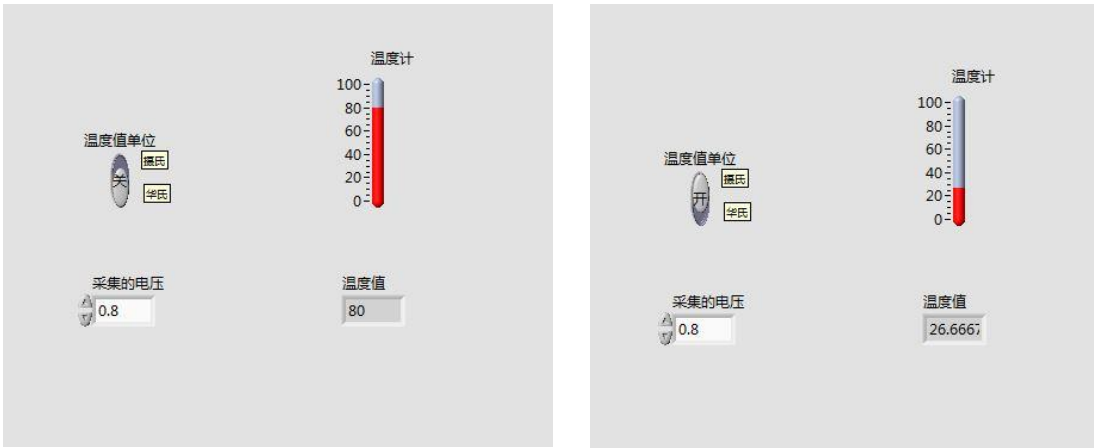


图 4 温度测量程序之前面板示意图

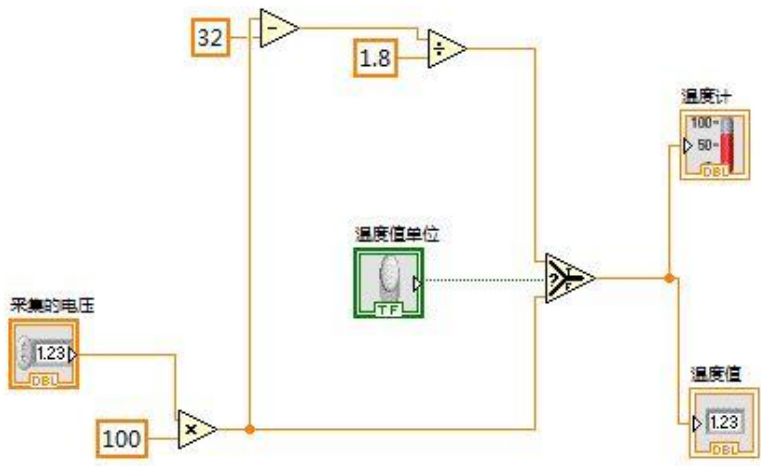


图 5 温度测量程序之程序框图示意图

3. 创建一个电压输出和采集的程序

在本实验中，我们把仪器的输出端口看作是电源输出端，把输入端口看作是电压表的输

入端，在仪器板上连接成回路，即做成一个测量电源两端电压的实验。通过改变前面板“输出电压”的数值来改变输出的电压值，判断采集到的电压值和输出的电压值在误差范围内是否相等。

在程序框图的设计中，利用测量 I/O 中的 DAQmx 数据采集来写入电压、采集电压。利用 while 循环和定时，来每隔 100ms 写入和采集电压。在前面板中，我添加停止输出和停止测量开关控件来中断程序中的 while 循环。通道设置时输出通道和输入通道的设备、端口需要和实际的虚拟仪器套件的端口匹配否则会发生错误。输出的电压不宜过高，超过 5V 仪器会自动保护，导致测量电压的不准确。

外部电路连接时，根据我在面板中设置的通道，只需要将 ao_0 端口和 ai_0+ 端口相连，ground 端口和 ai_0- 端口相连即可，注意到我的仪器是 Dev4，那么物理通道就应该选取 Dev4 所对应的通道。

以下是我的运行中的前面板和程序框图：



图 6 电压输出和采集的前面板示意图

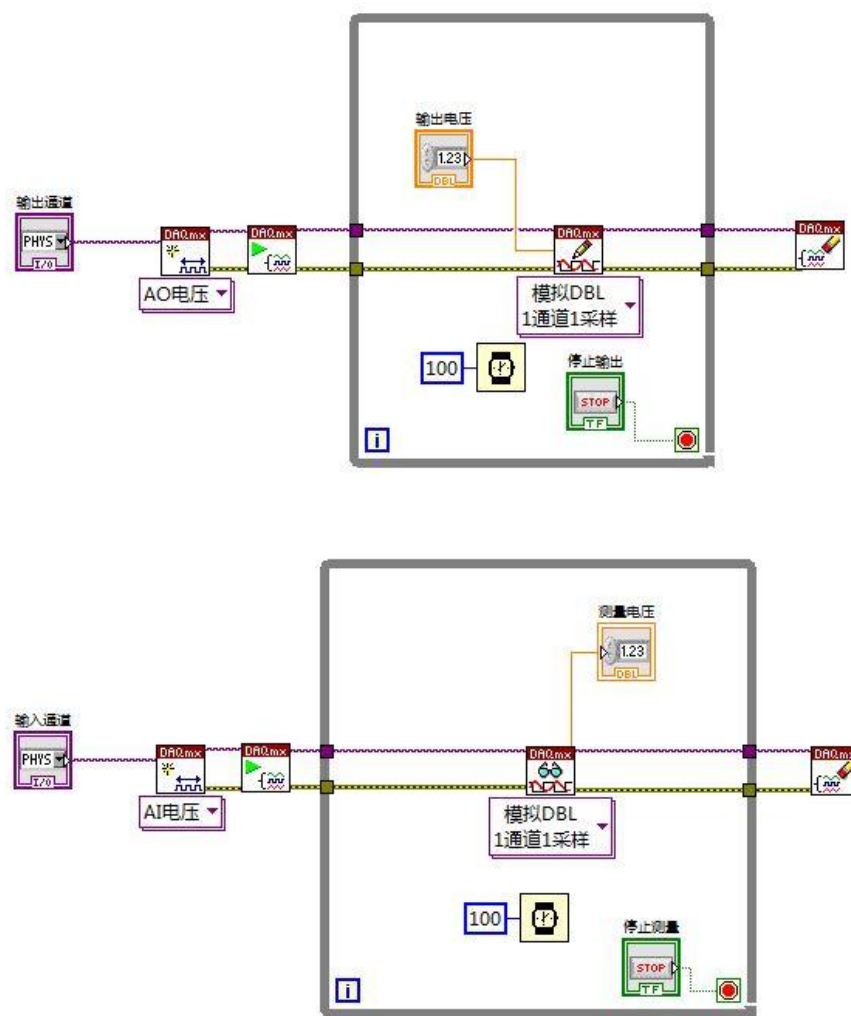


图 7 电压输出和采集的程序框图示意图

4.用虚拟仪器测量伏安特性

作为实验三的延续，同样利用软件提供的写入电压作为外部电路的电源电压，只不过本次需要测量的不再是电源两端的电压，而是待测元件两端的电压。同时我们还要获取流经元件的电流。多次测量从而得到待测元件的伏安特性曲线。经线性拟合，我们可以得到元件的阻值。

外部电路的连接需要将待测元件和标准电阻进行串联，ao_0 端口作为电源输出端，通过 ai_0 和 ai_1 端口测量总电压、标准电阻两端的电压。相减得到待测元件两端的电压，标准电阻两端的电压除以标准阻值得到流经元件的电流。

程序框图设计思路即是：先输出一个电压，等待几秒稳定后进行数据的采集。需设置电压步长，在 while 循环中，每次电压增加一个固定步长（在编程中我们也有循环中“步长”的概念，直接类比就很好理解），采集结果，利用数组进行存储、显示在前面板上，并且在前面板绘制元件的伏安特性图，直到一共有 20 组数据为止。最后通过数组中的多组电压-电流数据进行线性拟合得到阻值。（在 LabVIEW 中 for 循环与 while 循环其实等价，类似 C 语

言)

在前面板中设置输入控件：输出电压步长、测量数据点数、标准阻值、时间间隔、开关，显示控件：待测阻值、测量数据组、伏安特性曲线图。以下是我的前面板、程序框图和外部电路连接：

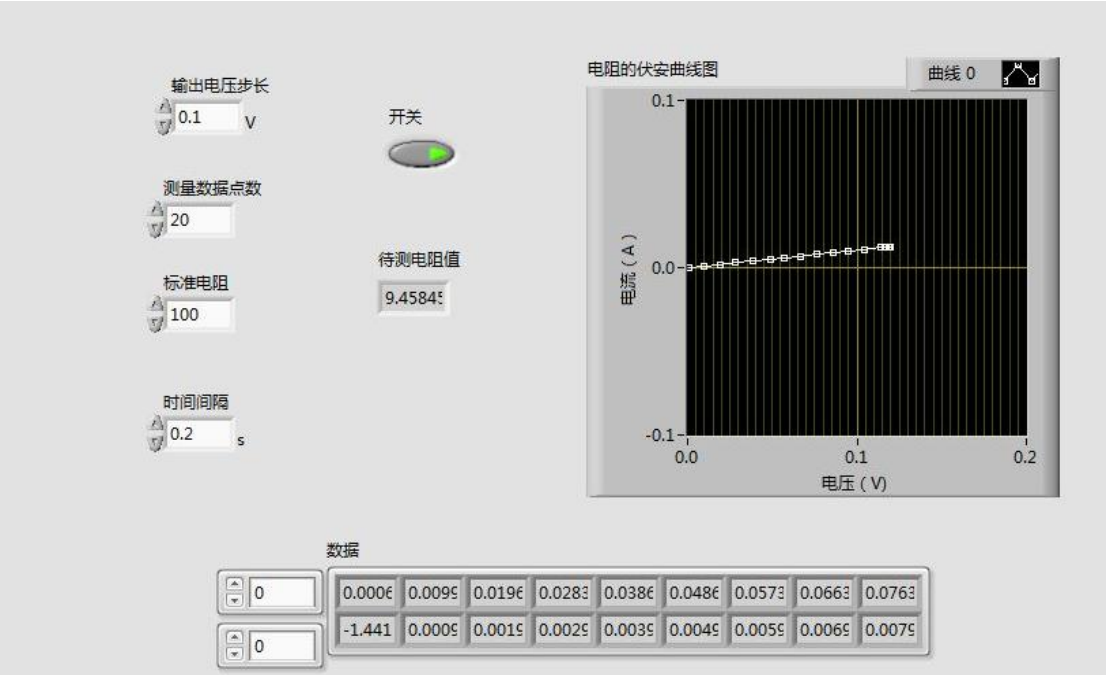


图 8 测量 10 Ω 的电阻阻值的前面板示意图

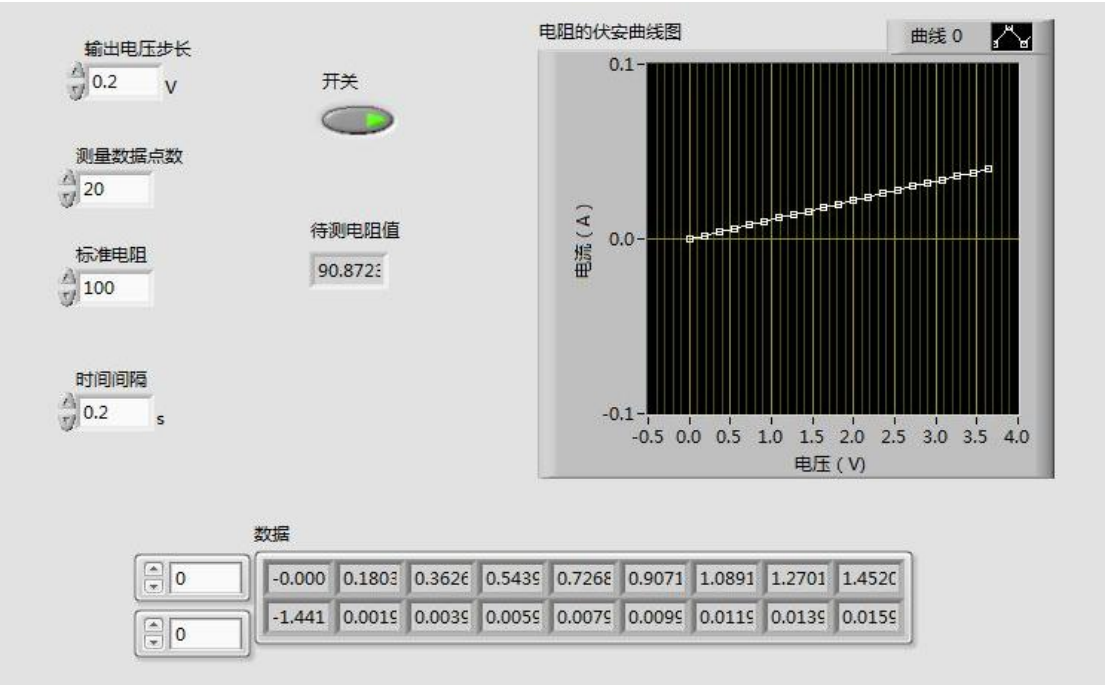


图 9 测量 100 Ω 的电阻阻值的前面板示意图

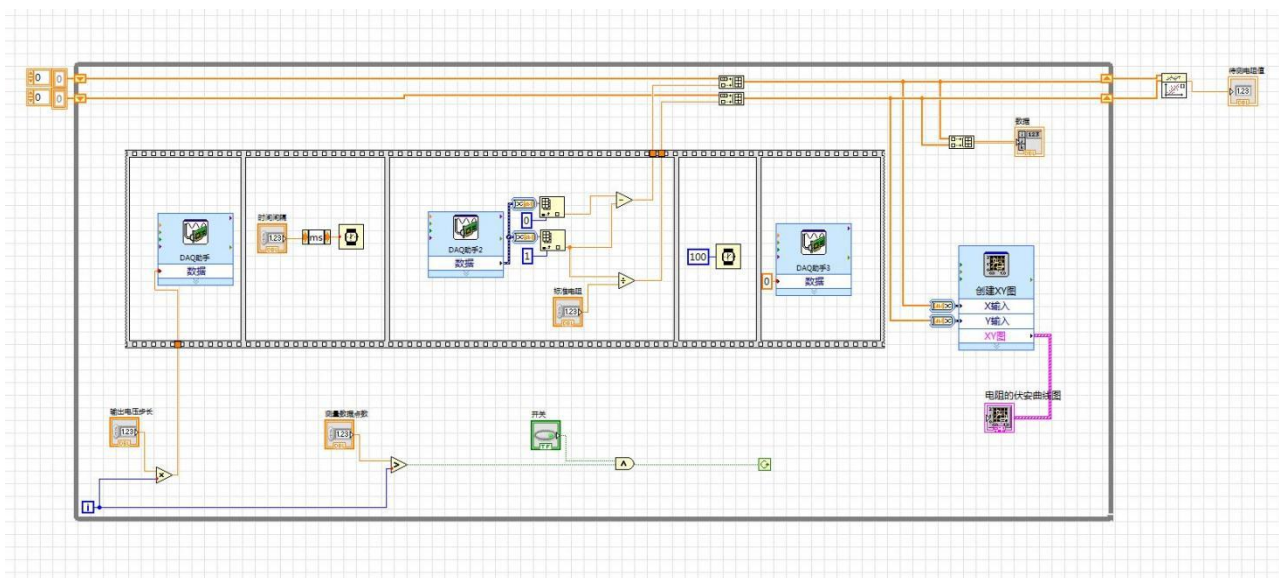


图 10 测量伏安特性的程序框图示意图

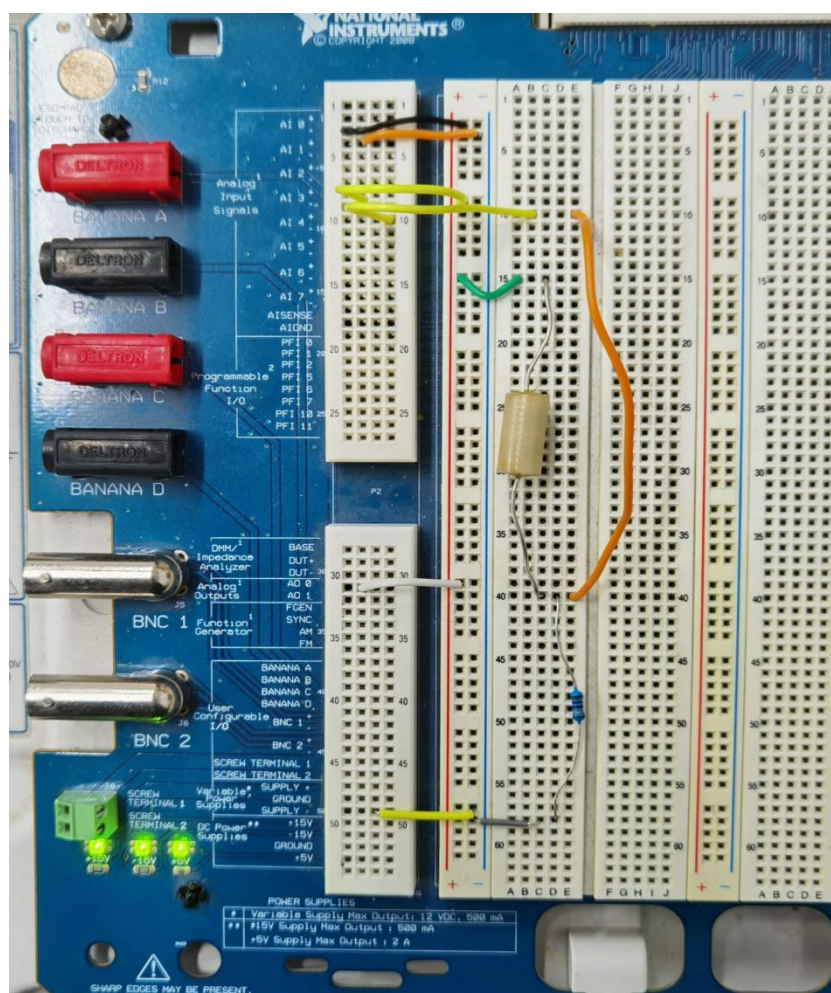


图 11 测量伏安特性的导线连接示意图（以二极管的连接为例）

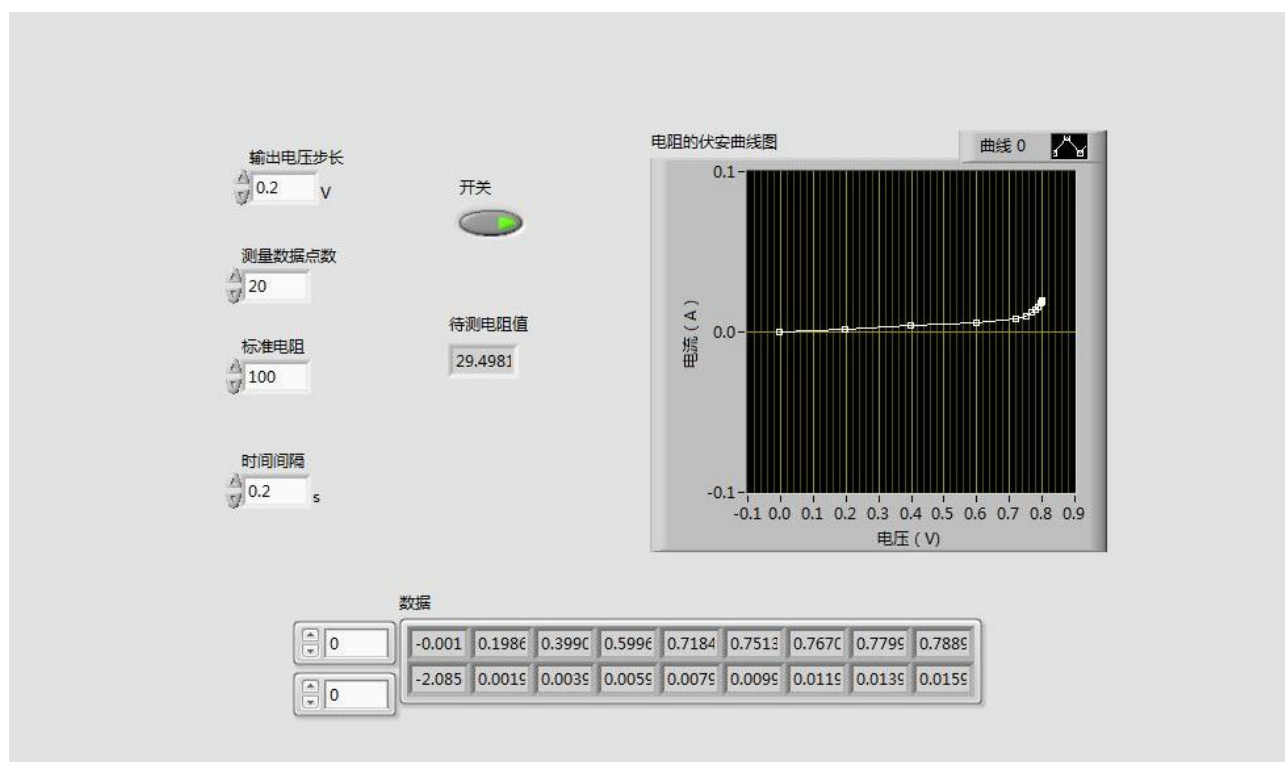


图 12 测量二极管正向阻值的前面板示意图

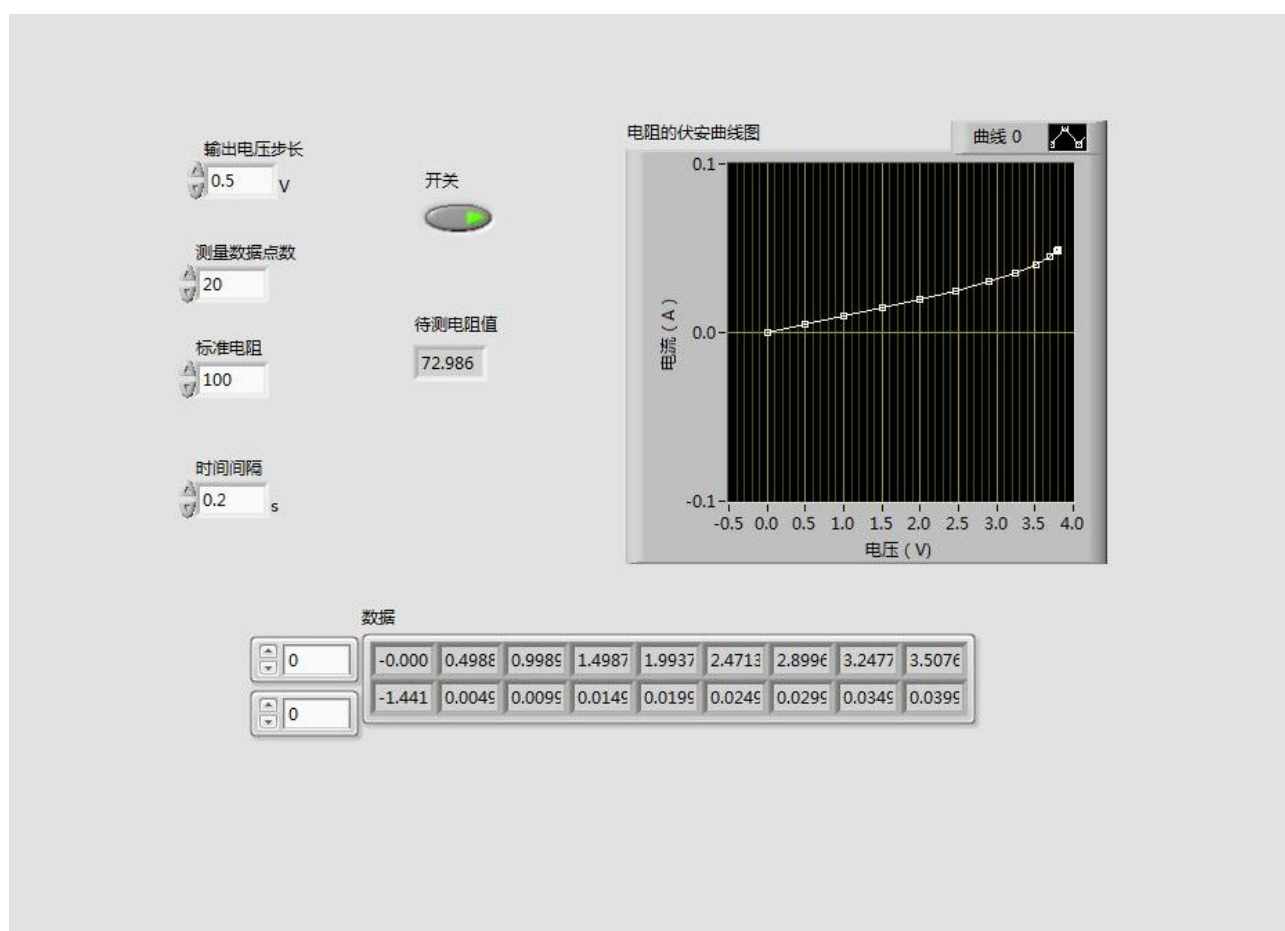


图 13 测量二极管反向阻值的前面板示意图

五. 实验数据处理与分析

1. 测量 10Ω 电阻

标准电阻为 10Ω ，步长为 $0.1V$ ，时间间隔为 $0.2s$

序号	电压 (V)	电流 (A)
1	0.000644	-0.000014
2	0.009983	0.000977
3	0.019644	0.001969
4	0.028339	0.002958
5	0.038644	0.003956
6	0.048628	0.004951
7	0.057323	0.005937
8	0.066340	0.006925
9	0.076323	0.007924
10	0.085984	0.008912
11	0.095001	0.009904
12	0.104018	0.010899
13	0.114323	0.011891
14	0.119476	0.012558
15	0.118510	0.012429
16	0.117222	0.012348
17	0.117222	0.012303
18	0.116900	0.012268
19	0.116900	0.012245
20	0.116256	0.012223
21	0.116256	0.012207

表 1 10Ω 电阻伏安特性测量数据表

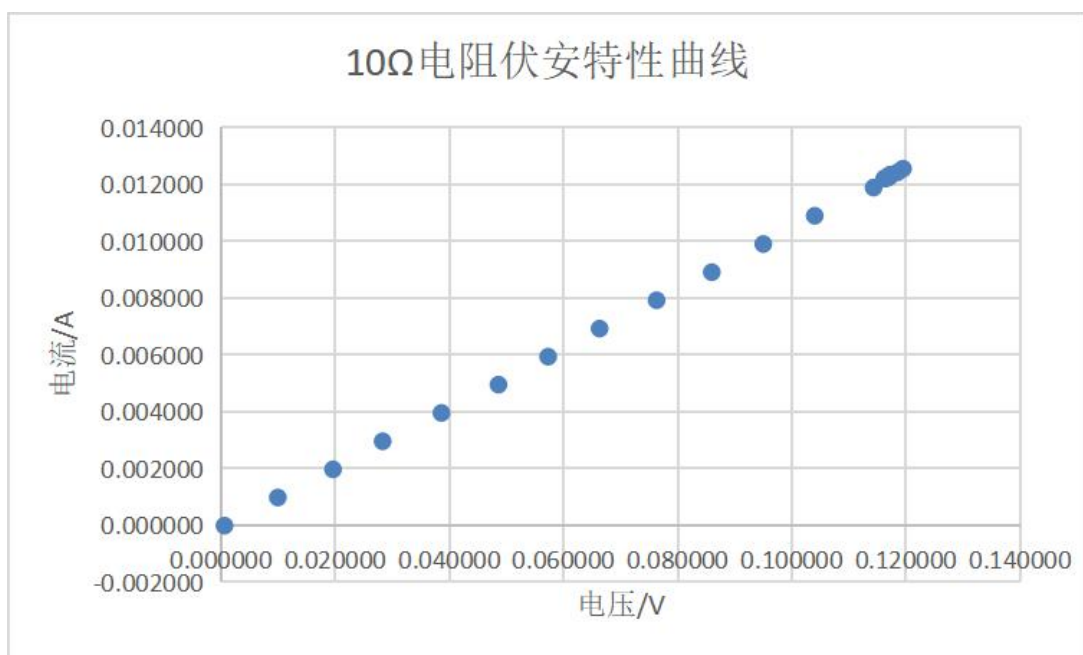


图 14 10 Ω 电阻伏安特性曲线图

【实验结果分析】

曲线虽然在 0.12V 电压处堆积了几个数据点，但是经线性拟合后仍然可以看出明显的线性关系，相关系数为 0.982。这与理想电阻的伏安特性符合得很好。直线斜率 k 为电导率 σ ，即 $1/R$ ，则 $R = 1/\sigma = 9.45845\Omega$ ，最后用万用表测得该实验电阻实际阻值为 9.42Ω ，相对误差为 $\delta = 0.40\%$ ，误差较小，数据符合较好。

2. 测量100 Ω 电阻

标准电阻为**100 Ω** ，步长为**0.2V**，时间间隔为**0.2s**

序号	电压(V)	电流(A)
1	-0.000966	-0.000014
2	0.180341	0.001979
3	0.362614	0.003979
4	0.543921	0.005975
5	0.726838	0.007982

6	0.907179	0.009978
7	1.089130	0.011975
8	1.270110	0.013975
9	1.452070	0.015972
10	1.634020	0.017971
11	1.815000	0.019965
12	1.997280	0.021968
13	2.178260	0.023965
14	2.360540	0.025961
15	2.541520	0.027961
16	2.722830	0.029958
17	2.904780	0.031954
18	3.086740	0.033957
19	3.267400	0.035951
20	3.449350	0.037954
21	3.631310	0.039951

表 2 $100\ \Omega$ 电阻伏安特性测量数据表

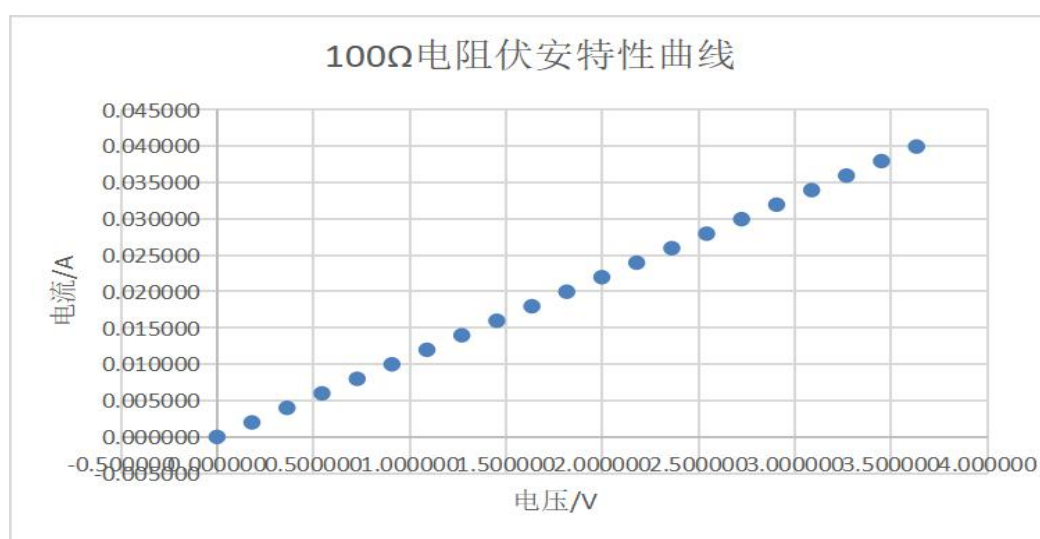


图 15 $100\ \Omega$ 电阻伏安特性曲线图

【实验结果分析】

由上面的数据可知，曲线没有明显的数据堆积区间，且经线性拟合后相关系数为 0.997，I 与 U 呈线性关系，与理想电阻的伏安特性符合很好。直线斜率为电导率 σ ，即电阻的倒数，则 $R = 1/\sigma = 90.8723\Omega$ ，最后用万用表测得实验电阻实际阻值为 90.47Ω ，相对误差为 $\delta = 0.44\%$ ，误差较小，数据符合较好。

3. 测量二极管正向导通电阻

标准电阻为**100 Ω** ，步长为**0.2V**，时间间隔为**0.2s**

序号	电压(V)	电流(A)
1	-0.001610	-0.000021
2	0.198697	0.001979
3	0.399004	0.003982
4	0.599633	0.005985
5	0.718465	0.007975
6	0.751313	0.009966
7	0.767093	0.011946
8	0.779975	0.013936
9	0.788992	0.015914
10	0.796399	0.017897
11	0.801230	0.019327
12	0.800586	0.019176
13	0.799620	0.019089
14	0.800264	0.019044
15	0.799620	0.019002
16	0.799620	0.018973
17	0.799942	0.018954
18	0.799942	0.018941

19	0.799620	0.018931
20	0.798976	0.018918
21	0.798976	0.018912

表 3 二极管正向伏安特性测量数据表

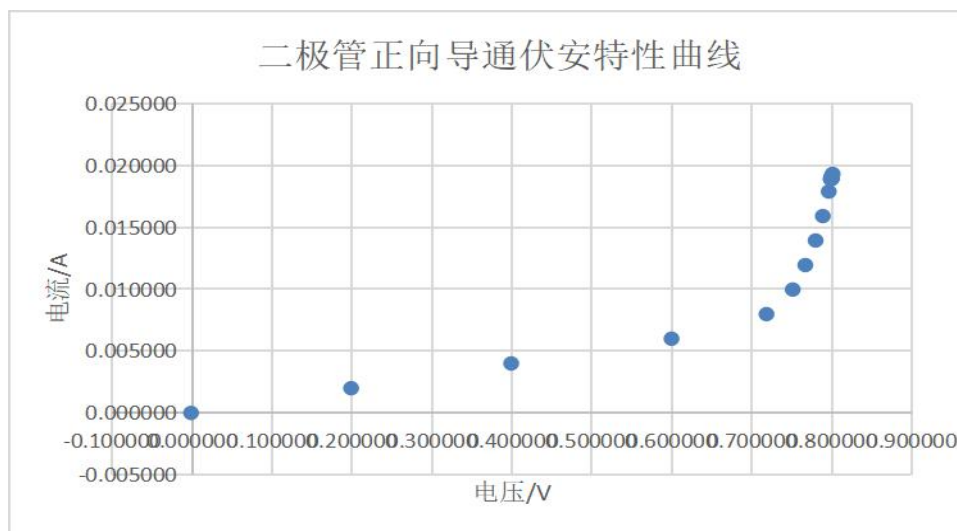


图 16 二极管正向导通伏安特性曲线图

【实验结果分析】

由上面的数据可知，曲线在低电压区域平缓，高电压区域陡峭，这代表随着电压增大，二极管正向导通电阻一开始近乎不变，且电流很小；然而电阻在电压超过死区电压（即开启电压） U_0 时迅速减小，并接近 0，这与正向导通二极管的伏安特性符合很好。

4. 测量二极管反向导通电阻

标准电阻为**100 Ω** ，步长为**0.5V**，时间间隔为**0.2s**

序号	电压(V)	电流(A)
1	-0.001610	-0.000021
2	0.198697	0.001979
3	0.399004	0.003982
4	0.599633	0.005985
5	0.718465	0.007975

6	0.751313	0.009966
7	0.767093	0.011946
8	0.779975	0.013936
9	0.788992	0.015914
10	0.796399	0.017897
11	0.801230	0.019327
12	0.800586	0.019176
13	0.799620	0.019089
14	0.800264	0.019044
15	0.799620	0.019002
16	0.799620	0.018973
17	0.799942	0.018954
18	0.799942	0.018941
19	0.799620	0.018931
20	0.798976	0.018918
21	0.798976	0.018912

表 4 二极管反向伏安特性测量数据表

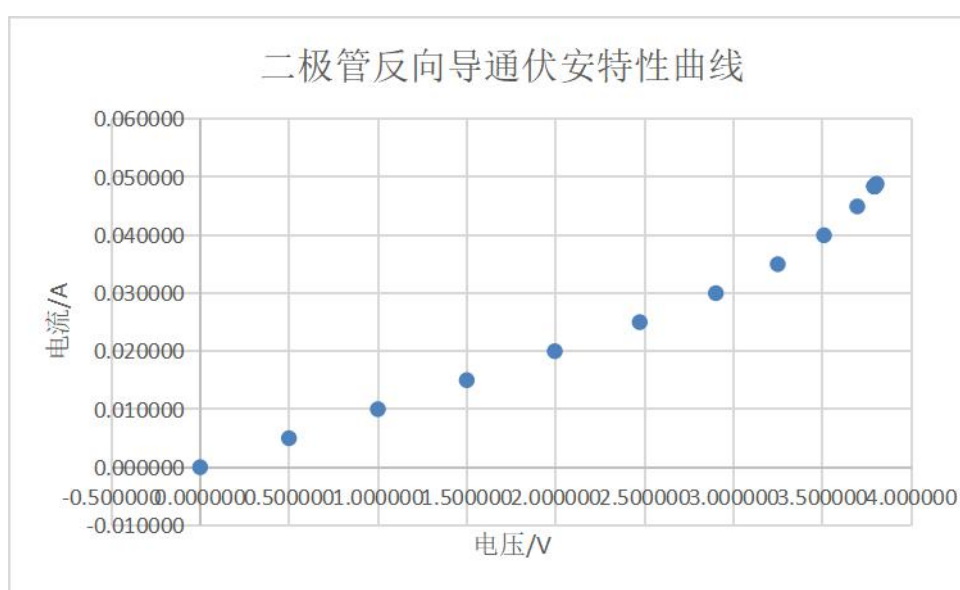


图 17 二极管反向导通伏安特性曲线图

【实验结果分析】

理想情况下二极管加反向电压，反向电流很小，而且反向电流不再随着反向电压而增大，即达到了饱和，这个电流即称为反向饱和电流。但是此处由于电流很小，受到噪声的影响，其变化并非很小。反向电压继续升高并超过反向击穿电压时，反向电流急剧增大，此时二极管被击穿。此处由于数据点选取不够合理的缘故，未能很完美地体现击穿现象。但是仍然可以看出数据点末尾处的击穿趋势。

六. 思考题

1. 虚拟仪器与传统仪器的区别：

虚拟仪器的功能，主要通过我们编写图形程序而实现，因为虚拟仪器平台中已经设定了很多的模块、工具和函数，所以可以实现的功能很多，集成化程度高。而且当我们编写好程序让计算机运行时，无需手动改变输入，机器可以自动执行，就像常规的计算机编程一样，减少人力投入，很便捷。还有，计算机的测量精度比我们人眼去观察要好很多，测量误差小。

传统仪器的功能已经全部设定好，只能使用特定的功能，灵活性非常差，且几乎没有任何自主性，对实验参数的调节等操作均需要人力完成。这也导致传统仪器支撑的实验容错率低、难以发现 bug（只能肉眼 debug）等缺点。

但是，虚拟仪器有一个致命的问题，那就是只能够进行确定性实验，在探索性方面虚拟仪器就是个灾难。因为我们只能够基于已知的物理定律给虚拟仪器“设定”好一个确定的运行模式，每一个合法输入都将产生“意料之中”的输出，但是对于一种暂且未知甚至仅仅是比较模糊的领域，我们不能确保我们根据现有知识设计而来的虚拟实验，是否能正确地反映未知现象！它在对现实世界的模拟中是极不可靠的。

2. 本实验内容 3 中的电压输出和采集哪个先执行：

借助 LabVIEW 中“高亮显示执行过程”工具跟踪运行过程，可以发现电压输出部分和采集部分的数据同时产生，因此两个部分同时开始执行。

七. 实验总结与思考

图形化的编程模式非常直观形象。对物理实验而言，它降低了做实验的要求限制，让计算机代替我们做事。比如每次增大一定的电压，人来做很有可能产生较大误差。比如测量元件两端电压，测量误差比人来测会小很多；再比如绘制伏安特性曲线，它能自动描点、自动更改横纵坐标的单位长度等等，如果是我们手动来做会很费劲。由此可见计算机对各个学科进步的推动作用。数字化、自动化的时代，我们所能够使用的工具也可以更加给力。

另外，正如我在思考题中提及的，虚拟仪器有一个致命的问题，那就是只能够进行确定性实验，在探索性方面虚拟仪器就是个灾难。因为我们只能够基于已知的物理定律给虚拟仪器“设定”好一个确定的运行模式，每一个合法输入都将产生“意料之中”的输出，但是对于一种暂且未知甚至仅仅是比较模糊的领域，我们不能确保我们根据现有知识设计而来的虚拟实验，是否能正确地反映未知现象！

举个不够恰当的例子，倘若我们只拥有微薄的光学知识，并不知道光的相干性叠加，或是从未意识到光的波动性，那么我们用虚拟仪器就根本无从设计也无从观察光的衍射实验（问题就出在你到底该用什么样的虚拟模型去描述、模拟现实世界的光？你永远不知道这种虚拟的模型是否一定正确）。也就是说，虚拟仪器仅仅只是基于已有知识的、对现实世界的一种不完备的模拟，这种不完备性使得我们不能永远相信它，因为每次使用它做模拟实验时候都必须在足够确定的理论支撑下才行。只有传统仪器才是永远符合物理世界、现实世界的规律的，也只有传统仪器才拥有探索未知规律的实验能力。

我们可以看到，在伏安特性曲线测量中，我们仍然借助了传统的电阻、导线等工具，虚拟仪器只是完成了集成的测量工作，这种虚拟仪器没有较高的模拟性，而是偏测量性的虚拟仪器，这才是它的拿手工作。我们实验中使用的也主要是它的测量性。现代科学就应该利用好虚拟仪器在测量、统计等领域的力量，而不是指望它进行物理上的模拟，这一点我们完成得十分出色。所以现在的虚拟仪器概念一般是指测量性、运算性工具，而鲜有模拟性仪器前文所提及的问题也仅限于广义虚拟仪器。

这与常规编程异曲同工。或许很多人会认为计算机很聪明，但其实它是非常“笨”的，它只能在人类确定好的运算规则下运行，而不能自主决定什么样的运算模式才能得到正确的结果或是自己创造算法。所以它所能做的只是帮助人类在他们自己接受的事实下提供强大的算力，至于运行模式的设计则是它永远无法自主优化并确保正确性的。我相信这种巧妙、充满智慧的任务，至少在未来很长一段时间内都会是人类的工作。

总的来说，虚拟仪器确实为现代科学的发展提供了足够的便利，但是也相应地产生了“智慧陷阱”，一旦给定运行模式，它就将拥有对现实理解程度的上限。人类还远没到能够指望虚拟事物在理论拓展、算法设计、结构优化等研究型、理论型领域有所建树的时代。这是一个陷阱，人类的智慧仍旧是外物难以企及的，虚拟仪器只是应用性、测量性工具，也只能是应用性、测量性工具。它不在智慧的殿堂内，它只是智慧的帮手。