

1.3.2: Resistors and Ohms Law – Voltage-Current Characteristics

Overview:

In this lab, we will continue to explore the characteristics of resistors. In this lab, we measure several combinations of voltage and current for a resistor and plot the resulting voltage-current characteristic curve measured for the resistor. The resistance of the resistor will be estimated from the slope of the voltage-current characteristic. The slope of the curve will be estimated using linear regression techniques; MATLAB commands used to perform linear regression are provided in the background material associated with this lab assignment.

Before beginning this lab, you should be able to:

- State Ohm's law from memory
- Use a digital multimeter to measure current and voltage (Lab 1.2.1)
- Use color codes on resistors to determine the resistor's nominal resistance
- Use the Analog Discovery's arbitrary waveform generator (AWG) to apply constant voltages to a circuit (Lab 1.2.2)

After completing this lab, you should be able to:

- Determine the least-squares best fit straight line approximating a set of data
- Calculate the correlation coefficient between a set of data and a line approximating the data
- Estimate resistance from measured voltage-current data

This lab exercise requires:

- Analog Discovery
- Digilent Analog Parts Kit
- Digital multimeter

Symbol Key:

DEMO

Demonstrate circuit operation to teaching assistant; teaching assistant should initial lab notebook and grade sheet, indicating that circuit operation is acceptable.

ANALYSIS

Analysis; include principle results of analysis in laboratory report.

SIM

Numerical simulation (using PSPICE or MATLAB as indicated); include results of MATLAB numerical analysis and/or simulation in laboratory report.

DATA

Record data in your lab notebook.

General Discussion:

We have previously noted that the resistance of a component is the slope of the current vs. voltage curve for the component. In this part of the lab assignment, we will measure a current-voltage characteristic curve for a resistor and estimate a resistance from this data. We will compare this resistance from the resistance measured by an ohmmeter.

In order to experimentally determine the current-voltage characteristic for our resistor, we will use the circuit shown schematically in Figure 1. The arbitrary waveform generator will be used to apply the voltage v_s . We will measure the voltage across the resistor, v_R , and the current through the resistor, i_R , using our DMM. By varying v_s , we can measure a set of values for v_R and i_R and plot v_R vs i_R , as shown in Figure 2(a). The slope of the line that “best fits” the measured data can then be used to estimate the component’s resistance, as shown in Figure 2(b).

Note:

Do not use the values displayed by the power supply as the resistor voltage and current, v_R and i_R . The values displayed by the power supply may differ from the resistor’s voltage and current due to non-ideal power supply effects, such as the power supply internal resistance.

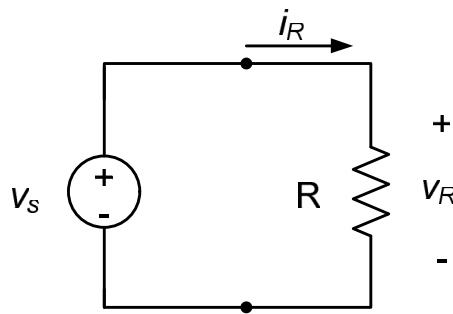
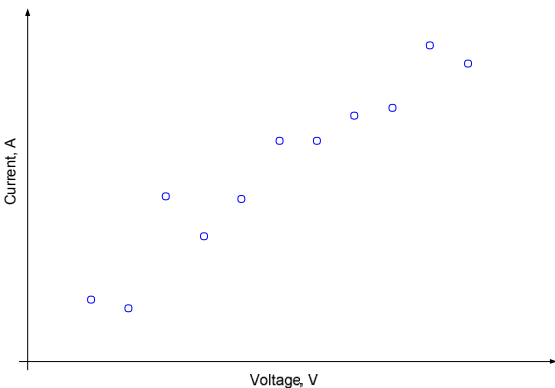
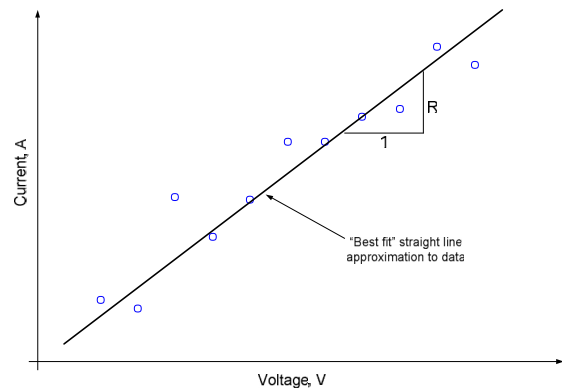


Figure 1. Circuit schematic.



(a) Current vs. voltage data.



(b) Data with straight-line approximation.

Figure 2. Measured data with “best fit” straight-line approximation.

Pre-lab: None

Lab Procedures:

DATA

1. Connect the circuit shown in Figure 1. Use a 100Ω resistor and one of the AWG channels for the variable supply. Use an ohmmeter to measure the actual resistance of your resistor and record the value in your lab notebook.

DATA

2. Vary the supply voltage v_s from 0V to approximately 2V. Measure v_R and i_R for at least 10 different values of v_s over this range of applied voltage (e.g. measure v_R and i_R at approximately 0.2V increments in v_s . Tabulate the measured values of v_R and i_R in your lab notebook. *Note: since we only have one DMM, the voltage and current measurements will have to be performed separately. If you have access to two DMMs, the two measurements can be made simultaneously.*

DEMO

3. Demonstrate operation of your circuit to the Teaching Assistant. Have the TA initial the appropriate page(s) of your lab notebook and the lab checklist.

Related information:

Resistance is estimated in the post-lab exercises using linear regression of these data. Linear regression is discussed in Appendix A of this lab.

Post-lab Exercises:

Determine a least-squares curve fit of the v_R vs. i_R . Plot the resulting line and the measured v_R vs. i_R data on the same graph. Comment on your results. Calculate a correlation coefficient for the data. Comment on your correlation coefficient relative to the qualitative agreement between the line and the data as shown on your plot.

Appendix A – Linear Regression:

Least-squares curve fitting:

Experimental data will always contain some uncertainty, so measured current-voltage data for a resistor will never lie exactly on a straight line – see, for example, Figure 2. Thus, the notion of a “best” straight-line approximation to the data is rather nebulous. It is simplest to draw by eye a straight line through the plotted data. This approach, while used fairly often, has the drawback that no two engineers are likely to draw the same straight line. Thus, we look for a more objective and readily quantifiable approach toward fitting a line to a set of measured data. One common approach toward determining a line which provides a “best fit” to the available data is *least squares curve fitting*. The basic idea behind the least-squares approach toward fitting a curve to data is as follows:

- We have a set of x, y data, where the x data points are $\{x_1, x_2, \dots, x_N\}$ and the y data points are $\{y_1, y_2, \dots, y_N\}$
- Assume that a straight line will approximate the data. The equation for the straight line is

$$y = mx + b \quad (1)$$

where m and b – the slope of the line and its y -intercept – are unknowns to be determined.

- We define the error between the estimated line and the measured data to be the square of the distance between the line and the data at the x data points. Thus, our error is:

$$E = \sum_{i=1}^N [y_i - (mx_i + b)]^2 \quad (2)$$

- If we minimize the error of equation (2) with respect to m and b , we obtain the least-squares straight line fit to the data. We will not discuss the mathematical details of this step here – they are rather tedious.
- We can determine how well our straight line agrees with the data by calculating a *correlation coefficient*. The correlation coefficient, r , is calculated by:

$$r = \frac{1}{N-1} \sum_{i=1}^N \left[\left(\frac{x_i - \bar{x}}{s_x} \right) \cdot \left(\frac{y_i - \bar{y}}{s_y} \right) \right] \quad (3)$$

where $\bar{x}$ and $\bar{y}$ are the means of the x and y data, respectively, and s_x and s_y are the standard deviations of the x and y data. The correlation coefficient is a number between -1 and 1 ($-1 \leq r \leq 1$); it essentially tells us how well our data agrees with the straight line curve fit. If all the data lie exactly on a straight line with positive slope, the correlation coefficient will be identically one ($r = 1$). If the data have noise or follow a nonlinear relationship, the correlation coefficient will be reduced. Data which are entirely uncorrelated have a correlation coefficient of zero ($r = 0$). A correlation coefficient of -1 simply means that there is a perfect negative relation between x and y – the data will lie on a straight line with negative slope. Figure 3 provides several examples of data with various degrees of correlation.

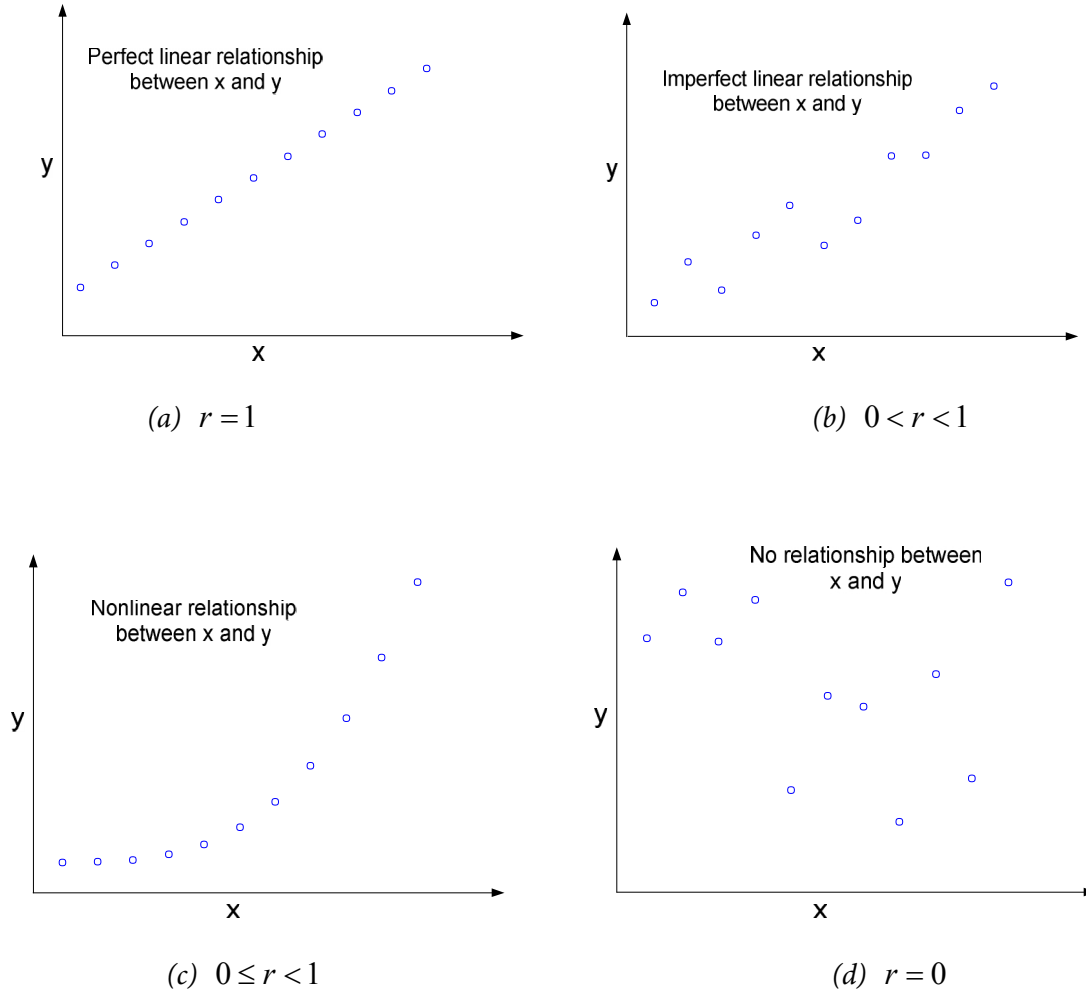


Figure 3. Example data and representative correlation coefficients.

Using MATLAB for least squares curve fitting:

MATLAB's **polyfit** function performs least-squares curve fitting. **polyfit** will fit an arbitrary-order polynomial to a set of data. Syntax for the function is

$$\mathbf{p} = \text{polyfit}(\mathbf{x}, \mathbf{y}, n)$$

where $\mathbf{x}$ and $\mathbf{y}$ are vectors containing the data to be fit, n is the order of polynomial to be fit to the data (a straight line is a first order polynomial, so we will always set $n = 1$). The function returns a vector containing the coefficients of the polynomial which provides a least-squares fit to the data. For $n = 1$ a two-element vector will be returned; the first element of the vector will be the slope of the line (m , in equation (1)) and the second element will be the y -intercept of the line (b , in equation (1)).

MATLAB's **corrcoef** function provides the correlation coefficient of two data sets. Possible syntax for using this function is:

$$\mathbf{r} = \text{corrcoef}(\mathbf{x}, \mathbf{y})$$

where $\mathbf{x}$ and $\mathbf{y}$ are vectors containing the data. This use of the function will return a 2×2 matrix; it will have the following form:

$$\mathbf{r} = \begin{bmatrix} r_{xx} & r_{xy} \\ r_{yx} & r_{yy} \end{bmatrix} \quad (4)$$

This matrix provides correlations between all possible combinations of the data provided to the function. r_{xx} is the correlation between the x data and itself. Likewise, r_{yy} is the correlation between the y data and itself. Since data is always perfectly correlated with itself, $r_{xx} = r_{yy} = 1$ always. r_{xy} is the correlation between the x data and the y data, and r_{yx} is the correlation between the y data and the x data. For us, $r_{xy} = r_{yx}$. Thus, either the r_{xy} or r_{yx} terms will give us the correlation coefficient as defined in equation (3).