MASSACHVSETTS INSTITVTE OF TECHNOLOGY
Department of Electrical Engineering and Computer Science
6.01—Introduction to EECS I
Spring Semester, 2008

Assignment 8, Issued: Tuesday, April 1

Overview of this week's work

In software lab

- Work through the software lab.

Before the start of your design lab on Apr 3 or 4

- Read the class notes and review the lecture handout.
- Do the on-line tutor problems in section 8.1.
- Read the entire description of the design lab, so that you will be ready to work on it when you get to lab.

In design lab

- Do the nano-quiz.
- Work through the design lab.

At the beginning of your next software lab on Apr 8 or 9

- Submit written solutions to all the questions in this handout. All written work must conform to the homework guidelines on the web page.

Exploration 8 will be posted on the web on April 3.

You will need to have installed numpy for this lab. Instructions are in the 6.01 Software pages. Or use a lab laptop.

A circuit solver

In this software lab you will learn to use some programs for describing the components of a circuit and the way they are wired together. Once the circuit is described, you can ask it to generate a set of variables, representing voltages and currents, and equations among them, and to solve the set of equations, yielding numerical values of the voltages and currents

Circuit Constraints

In the `lab8` directory, you will find `resolveConstraints.py` and `CircuitDesign.py`, which contains programs for creating lists of constraints (equations) and their associated variables, and then once the list of constraints and variables has been generated, determining values for the variables so that the constraints are satisfied. We won't be delving into the details of this code in this lab (that's for next week); instead, we will use the code to solve for the values of voltages and currents in actual circuits.

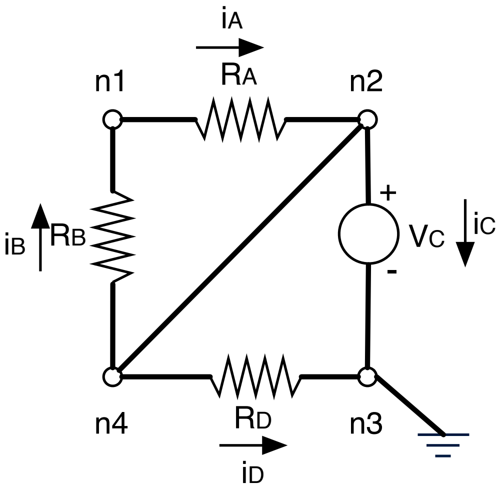


Figure 1: Circuit with three resistors and a voltage source.

Here is an example circuit specification, corresponding to the first example in the lecture notes (reproduced here in figure 1). You can find it in `example.py`.

We start by specifying values for the resistors and the voltage source. Naming them here makes them easy to play with.

```
ra = 100    # 100 ohms
rb = 200    # 200 ohms
rd = 100    # 100 ohms
vc = 10     #  10 volts
```

Next, we make an instance of the `Circuit` class:

```
c = Circuit([Resistor(ra, 'n1', 'n2'),
             Resistor(rb, 'n4', 'n1'),
             Resistor(rd, 'n4', 'n3'),
             Wire('n4', 'n2'),
             VSrc(vc, 'n2', 'n3')])
```

Each type of circuit component (Resistor, Wire, VSrc) is modeled by a Python class which takes the arguments illustrated above (generally a value and the names of the circuit nodes to which the component is connected). The Circuit class takes as its argument a list of instances of circuit components. Note that connections among components is indicated by the fact that the components share circuit nodes.

Then, we ask the circuit instance for a solution, which requires specifying which node is to be treated as ground (that is, set to 0 volts). If you don't specify an argument for the solve method, it will assume that the ground node is called 'gnd'.

```
c.solve('n3')
```

It should print the following:

```
i_n1_n2_1  =  0.0
i_n2_n3_5  =  -0.1
i_n4_n1_2  =  0.0
i_n4_n2_4  =  -0.1
i_n4_n3_3  =  0.1
n1  =  10.0
n2  =  10.0
n3  =  0.0
n4  =  10.0
```

The last four lines are the node voltages. The first five lines tell you the values of the currents in the circuit. Variables of the form i_node1_node2_x indicate the value of the current flowing from node1 to node2. Note that these currents can be negative, indicating that the current is actually flowing in the other direction (from node2 to node1). The last number in the current name is simply a sequentially generated integer to make sure that each current name is unique. If you have more than one connection between two nodes (say resistors in parallel), then node names are not enough to uniquely identify the currents.

Question 1: Run this example through Python and be sure you get the same answer as above. Explain intuitively why some of the currents are zero.

Question 2: Without changing anything else, what if we did:

```
c.solve(’n1’)
```

What changes in your solution? Is this solution physically different (could you see the difference with your multimeter) from what you got before? Explain.

Question 3: Modify this example to get rid of the wire between n_2 and n_4. Now what values do you get? Explain.

Question 4: For this version of the circuit (without the wire between between n_2 and n_4):

a. *Intuition:* If we were to take resistor `ra`, pull it out, and turn it around so that the terminals were connected the other way, would that change the voltages and currents in the actual physical circuit, that is, the values that you can measure with your multimeter? Explain.

b. *Using solver:* If you replace the entry in the `Circuit` argument that reads:

```
Resistor(ra, ’n1’, ’n2’)
```

with a entry that reverses the order of the node arguments:

```
Resistor(ra, ’n2’, ’n1’)
```

What changes in the solution printed by Python (ignoring tiny numerical differences)? Is this solution physically different from (could you see the difference with your multimeter) what you got before? Explain.

Question 5: For this same version of the circuit (without the wire between n_2 and n_4):

a. *Intuition:* If we were to take the voltage source in this same circuit, pull it out, and turn it around so that the terminals were connected the other way (without changing anything else), would that change the voltages and currents in the actual physical circuit, that is, the values that you can measure with your multimeter? Explain.

b. *Using solver:* If you were to change the entry

```
VSrc(vc, ’n2’, ’n3’)
```

to

```
VSrc(vc, ’n3’, ’n2’)
```

What changes in the solution printed by Python (ignoring tiny numerical differences)? Is this solution physically different from (could you see the difference with your multimeter) what you got before? Explain.

Question 6: Run the examples described in the files `circuit.py` and `circuit2.py`, and report the results. Draw circuit diagrams of the circuits described in those files.

Op Amps

The constraint-based program for analyzing circuits also has an op-amp model. Therefore you can use the constraint resolver to help you check your design of a circuit to control the motor during design lab.

In class we discussed the following operational amplifier circuit, referred to as an inverting amplifier.

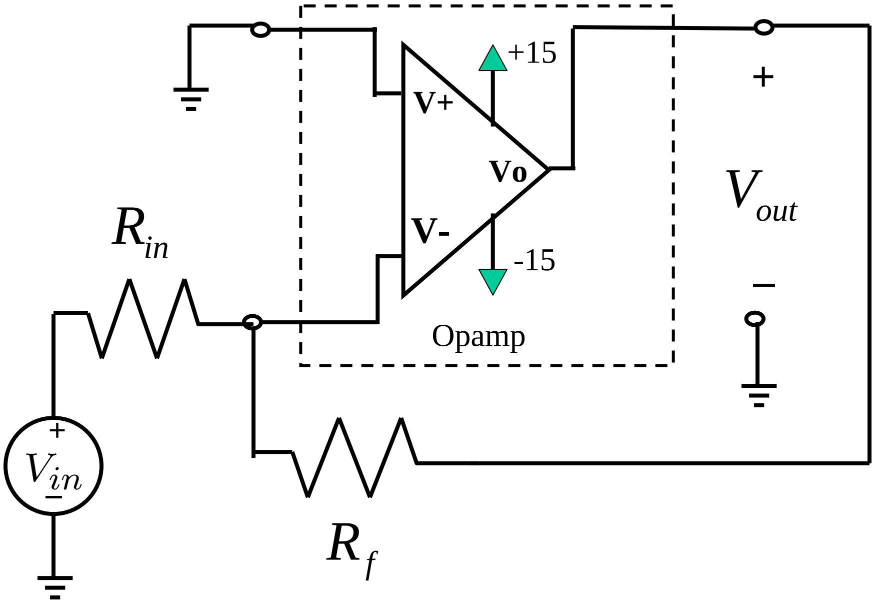


Figure 2: Inverting amplifier.

In order to simplify the analysis of the op-amp circuit, we assumed that $V_+ - V_- \approx 0$ and that no current enters the op-amp inputs. Using those assumptions, we determined that the relationship between V_{in} and V_{out} was

$$V_{out} = -\frac{R_f}{R_{in}} V_{in}.$$

The main reason we could assume that $V_+ - V_- \approx 0$ was that the op-amp was connected in a feedback configuration and the gain of the op-amp was very large. Now we can investigate the accuracy of that approximation using the constraint resolver.

Another model that we have used for the op-amp is a voltage-controlled voltage source model. Using that model in the inverting amplifier yields a somewhat different looking schematic (shown in figure 3), where the gain of the voltage-controlled voltage source, K, is a very large number.

You can construct a `Circuit` with an op-amp by adding an entry like this:

```
OpAmp('v+', 'v-', 'vo')
```

where `'v+'`, `'v-'` and `'vo'` are node names in your circuit. Note that this assumes that the negative output node is `'gnd'` and the that $K = 1000$. You can give a more complete specification if you want to:

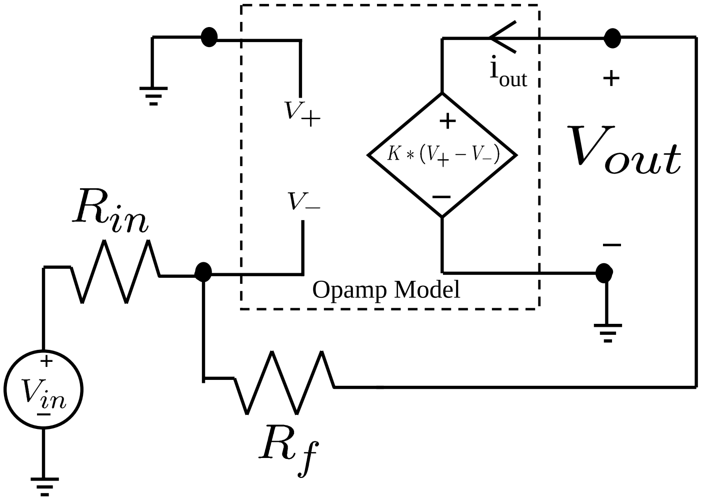


Figure 3: Op Amp model: the voltage-controlled voltage source.

```
OpAmp('v+', 'v-', 'vo', 'gnd', 1000)
```

or, if you only want to specify the gain:

```
OpAmp('v+', 'v-', 'vo', K = 1000)
```

Note that in real life you don't get to specify the op-amp gain: it's determined by the implementation of the op-amp part. The gain is part of our *model* of the op-amp.

Question 7: Use the software to solve the inverting amplifier circuit in figure 2 using $R_{in} = 10,000$, $R_f = 100,000$, and the op-amp gain equal to $K = 1000$ (which is the default value in the code).

Question 8: Roughly determine the smallest value of op-amp gain, K, for which

$$\frac{V_{out}}{V_{in}} = -\frac{R_f}{R_{in}}$$

is accurate to within ten percent.

Question 9: Model the circuit in figure 4 (from design lab 7) and solve. Use a `Wire` entry for the feedback connection. Use Vs = 12V and R = 4,700, as you did last week.

When you implemented this circuit in lab, you should have expected that V_{out} would be one quarter of V_s (remember why?). Using the solver, test how good this approximation is, for various values of K, the op-amp gain.

Then, change your circuit model to remove the op-amp and simply connect the two dividers. This is the circuit you tried originally in last week's lab (before you had op-amps). Does the solver result conform to your observations of the behavior of that circuit?

The op-amp models we have considered so far have neglected one important fact: that the output

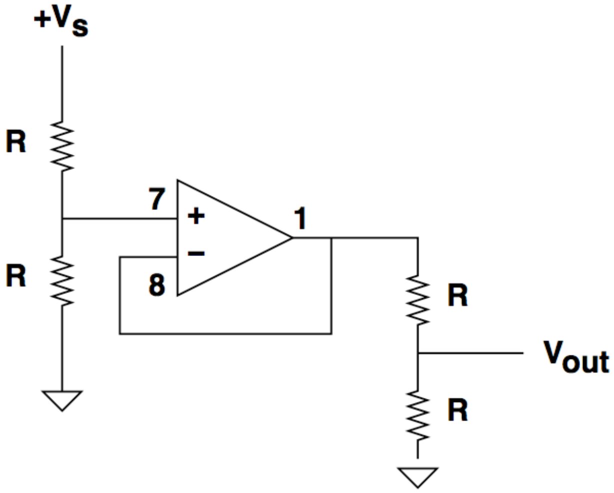


Figure 4: Buffered voltage dividers. The numbers 1, 7 and 8 are the pins on the op-amp chip we used in lab. Vs is the voltage source voltage.

voltage of an op-amp cannot exceed the range of voltages provided to it. Note that in figure 2 we show that the op-amp is connected to a two-sided power supply, providing both +15V and -15V. The output of the op-amp is limited to being in the range of the supply voltages. If we connect the op-amp to +15V and 0V (ground), then the output must be in that range (only positive voltages).

The op-amp model in the solver does not normally enforce these limits due to the supply voltages, unless you specify the voltages in the `OpAmp` instance:

```
OpAmp('v+', 'v-', 'vo', Vcc=15, Vss=-15)
```

The positive supply voltage is `Vcc` and the negative supply voltage (or ground) is `Vss`. Note that we are limited to `Vcc = -Vss` or `Vss = 0`.

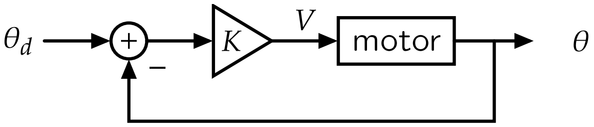
Question 10: Use the software to solve (again) the inverting amplifier circuit in figure 2 using $R_{in} = 10,000$, $R_f = 100,000$, and the op-amp gain equal to $K = 1000$ (which is the default value in the code). Try $V_{in} = 2$ and contrast what happens when you specify the supply voltages and when you don't.

If you have extra time, start working on the tutor problems.

Design Lab

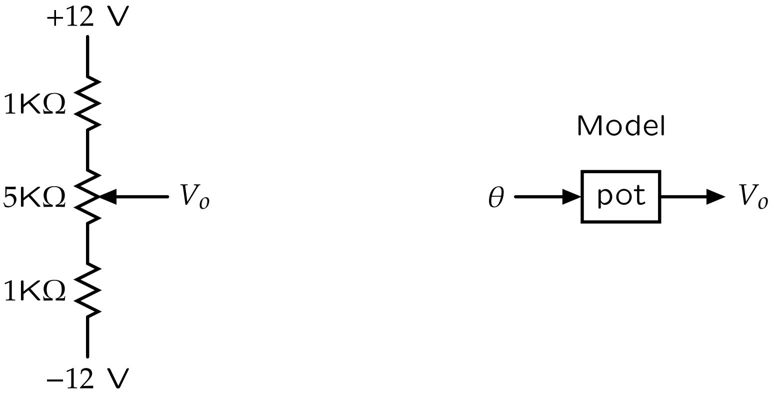
The goal of today's design lab is to build a circuit to control the position of a motor. In later labs, we will use this motor to turn the "head" of our robot.

At a high level, we can think of our motor controller as follows:



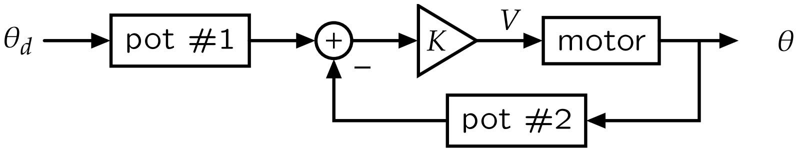
We wish to command a desired angle θ_d that is the input to a feedback loop that compares the desired angle to the measured angle θ. The difference is multiplied by a gain K to generate an output voltage V that will cause the motor to turn in a direction to make the measured and desired angles equal.

We cannot conveniently manipulate angles, so we will use potentiometers to convert the angle of their input shaft to a voltage. A potentiometer is a three terminal device. The resistance between the middle and bottom terminals increases in proportion to the angle of the input shaft (θ) and the resistance between the middle and top terminals decreases, so that the sum of the top and bottom resistors is constant (here it is 5KΩ). By connecting a potentiometer as shown below we can create a device that converts θ into a voltage V_o (measured relative to GND).



Question 11: Construct the circuit shown above (be sure to adjust the power supplies to ±12V before connecting the circuit.) Mathematically determine the range of V_0 that the device can generate. Measure the range (relative to GND) and compare the measured and calculated ranges.

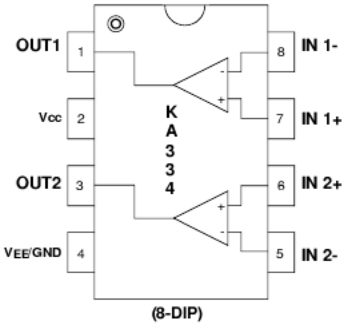
Using potentiometers, we can redraw our block diagram as follows.



As before, we have represented the motor as a device that converts an input voltage to an output angle.

Question 12: Use the potentiometer from the previous question to drive the voltage input to the motor. Use a Lego motor with adaptor cable and breadboard adapter (6 pins). The motor is connected between pins 5 and 6 of the adaptor. Connect pin 5 to the output of the potentiometer and connect pin 6 to GND. How fast can you get the motor to turn? Explain.

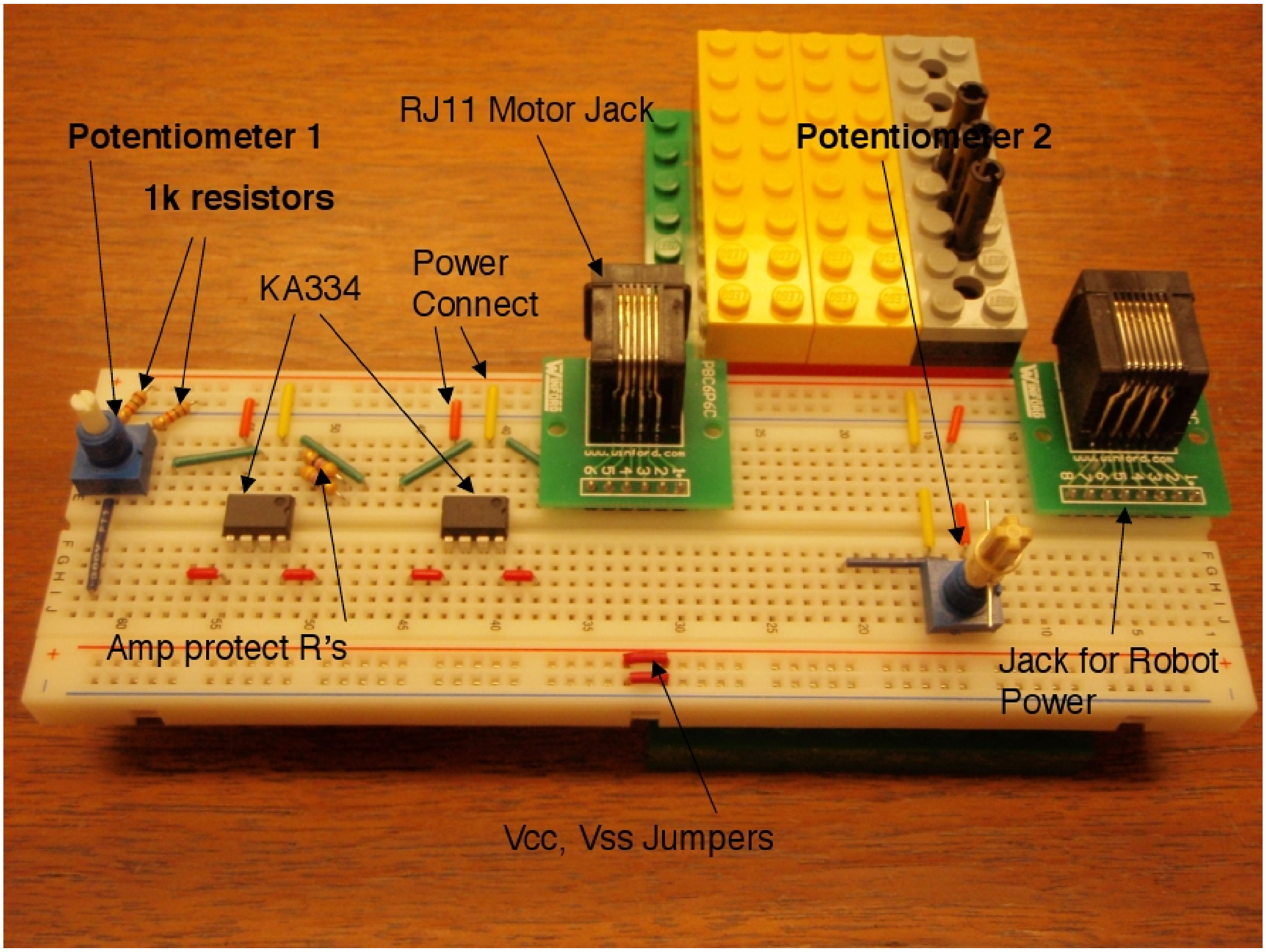
Question 13: Use an op amp to buffer the output of the potentiometer (this is similar to what we did last week when we used an op amp as a buffer between two voltage dividers). Use one of the two op amps in the KA-334 package (see schematic below). Let $V_{cc} = +12$ V (pin 2) and $V_{ss} = -12$ V (pin 4). How fast can you get the motor to turn? Can you make it go in both directions?



When we use the motor to position the robot "head," we will only have access to a +12 V supply (the robot does not have a −12 V supply). One way to get bidirectional turning from a single +12 V supply is by generating a "virtual ground" that is midway between 0 V and +12 V.

Question 14: Reconnect power to the op amp so that it uses a single +12 V supply, i.e., connect $V_{cc} = +12$ V (pin 2) and $V_{ss} =$ GND (pin 4). Use a second op amp to generate a "virtual ground" signal. Modify your previous op amp circuit so that it uses the virtual ground to allow the motor to turn in both directions.

For the remainder of today's lab, you will need a "head assembly," which you can obtain from the staff. These head assemblies have circuitry that is pre-built on the "bottom board." The "top board," which is mounted so that it can be swiveled by a Lego gear motor, has no circuitry on it right now. Examine and familiarize yourself with the circuitry on the bottom board (shown below). You will note that there are two KA334 op-amps, wired for power but not for input or output.



The photograph of the bottom board (above) shows the following:

1. Potentiometer 1: This 'pot' is connected between the power supply and ground rails by two 1k resistors. The potentiometer has a total resistance of 5kΩ, just like the example you started this lab with.

2. Potentiometer 2: When you get the head assembly, this potentiometer will be connected to the bottom of the Lego motor. It is connected to the power and ground rails directly (not through resistors).

3. Two KA334 op-amps. As you can see, they are both connected to the power and ground rails. Two of the input pins (numbered 5 and 8) are connected by short red wires to rows of holes beyond the ends of the op-amps. This is just for your convenience in wiring. The other two input pins are unconnected. The two op-amp outputs (pins 1 and 3) are also connected to outboard rows of pins. Note that pin 1 of one of the op-amps is connected to two 47 ohm resistors in parallel. You should use that connection to the motor when it comes time to connect to the motor. DO NOT DO THIS YET.

4. There are two connectors from pins to cables. One of these is an RJ-11 like the one you have already used to connect to the motor. As you already know, pins five and six are the motor

connections. The other cable connection accepts the RJ-45 cable that is on the robot, and is wired so that this board will get power and ground connections from the robot battery. For the moment you can ignore this connector and derive power and ground from the same power supply you have been using.

5. Note that, on some but not all protoboards, the left and right power rails must be connected by jumpers. The jumpers for the bottom two power rails are shown in the photograph as V_{CC}, V_{SS} Jumpers. There must be similar jumpers for the top two power rails. The upper jumpers cannot be seen in the photograph because they are beneath the RJ-11 Motor Jack. **Please make sure that there are jumpers at the right spots beneath the RJ-11 connector.**

To complete our motor controller, we must design a circuit to calculate v_1, the voltage from potentiometer 1, minus v_2, the voltage from potentiometer 2 (under the motor) and then offset this difference to "virtual ground," i.e., add 6 V. This resulting voltage should be multiplied by $K = 2$ and delivered by an op amp to the motor through the two parallel 47 ohm resistors.

Question 15: Construct an op amp circuit to compute $2(v_1 - v_2 + 6)$. Change the positions of the potentiometers and measure the change in output voltage that is produced. Can you set the two potentiometers so that the output voltage is 6 V? Explain why this is important. Set potentiometer 1 so that θ_d is greater than θ. Will the resulting voltage V move the motor in a direction to increase θ?

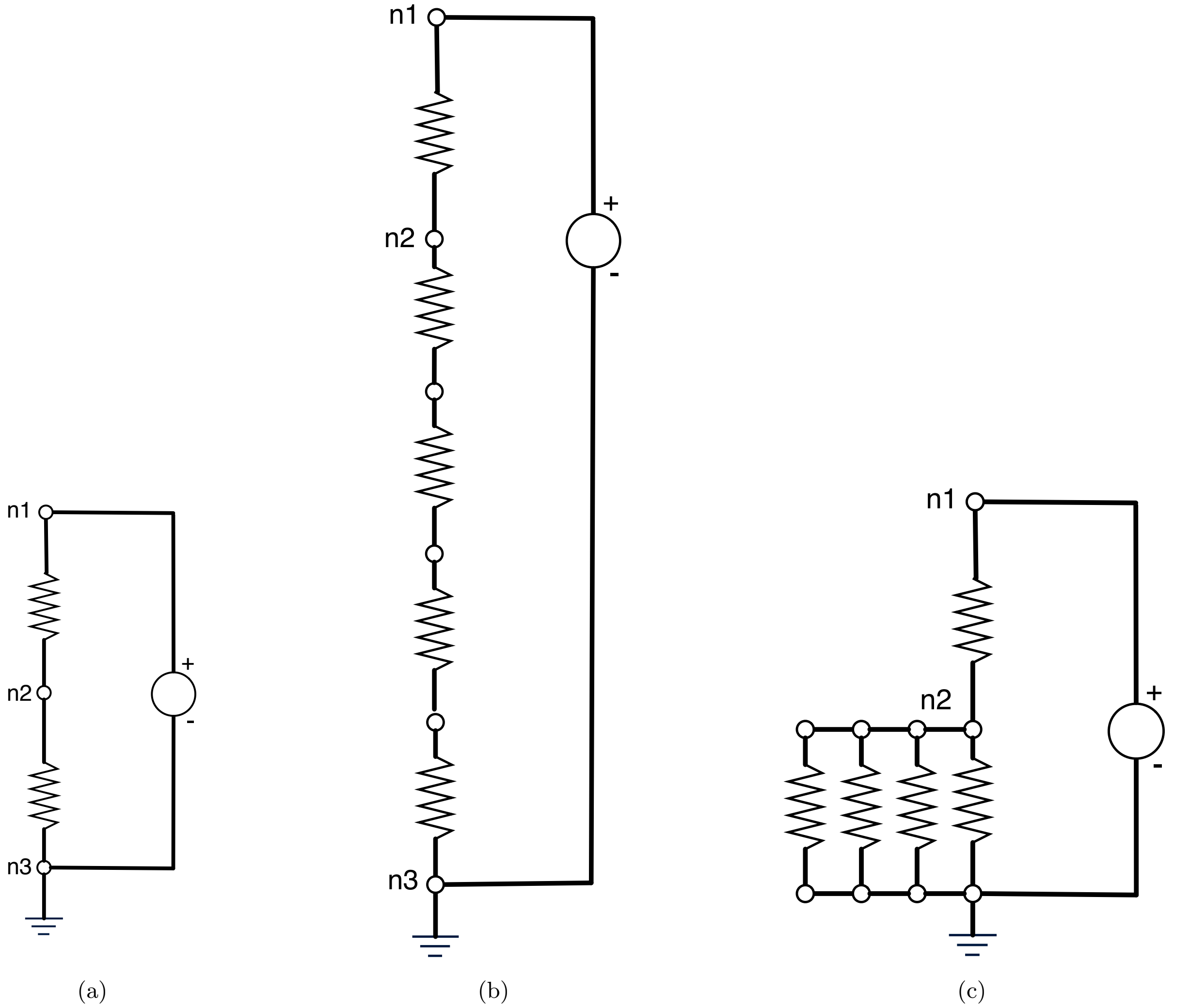
Question 16: Wire up the motor so that the feedback system drives the motor angle θ to equal θ_d. Does the motor turn to a stable position when you turn on power? Change the angle of potentiometer 1. Does the motor position change accordingly?

Post-lab Homework

Write up answers to all questions in the lab handout, including the ones below.

Practice with resistor circuits

Use the circuit-solving software to help answer the following questions. For each one, submit the the `Circuit` you constructed and the output it generated when you solved it. Show which values in the output correspond to answers to the questions we ask below.



(a) (b) (c)

Figure 5: Resistor Circuits

Question 17: Figure 5(a) shows a circuit with two 100Ω resistors and a 10V source. What is the voltage at node n_2?

Question 18: Figure 5(b) shows the circuit in Figure 5(a), but with four 100Ω resistors, wired "in series" in the place of $r_{2,3}$ (the resistor between nodes n_2 and n_3). What is the voltage at node n_2 in this circuit?

Question 19: How could you choose the resistance of $r_{2,3}$ in the circuit in figure 5(a) so that the voltage at node n_2 would be the same as it is in the circuit in figure 5(b)?

Question 20: Figure 5(c) shows the circuit in Figure 5(a), but with four 100Ω resistors, wired "in parallel" in the place of $r_{2,3}$. What is the voltage at node n_2 in this circuit?

Question 21: How could you choose the resistance of r_{2_3} in the circuit in figure 5(a) so that the voltage at node n_2 would be the same as it is in the circuit in figure 5(c)?

Concepts covered in this lab

- Design of simple buffers and differential amplifiers
- Drive of the motor from two different sources
- Design of an offset differential amplifier
- Your very first servomechanism