Chapter 5

Sensor Design

Close your eyes. Plug your ears. Hold your nose. Tie your hands behind your back. Shut your mouth. Tie your shoelaces together. Spin yourself around a few times. Now walk. How does it feel? That's exactly what your robot feels: nothing–without sensors. You have been given many types of sensors that can be used in a variety of ways to give your robot information about the world around it. In this chapter, we'll explain each of the sensors you have, how it works, what it's good for, and how to build it.

Before we can teach you what sensors do, we need to make one point very clear. Sensors are not magical boxes. The phrase "Sensors indicate a large LEGO robot 2 meters off our port bow, Captain!" will never appear. All information you get from sensors must be decoded by YOU, the human builder and programmer.

Sensors convert information about the environment into a form that can be used by the computer. The sensors that are on the robot can be related to sensors found in humans. Touch sensors embedded in your skin, visual sensors in your retina, and hair cells in your ears convert information about the environment into neural code that your brain can understand. Your brain needs to understand the neural code before you can react. Since you will be programming the robot, you will need to understand the output of the sensors before you can program your robot to react to different stimuli.

Take time to play with each of the sensors you have been given. Figure out how they work. Look at the range of values they returns, and under what conditions it gives those values. The time you spend here will greatly ease your integration of hardware and software later. The better you understand your sensors, the easier it will be for you to write intelligible control software that will make your robot appear intelligent.

Sensors provide feedback to your program about the environment. Feedback is important in any controlled situation. Rather than using open-loop, or timed programs that simply follow a pattern but have no real knowledge of the world, sensors can provide the feedback necessary to let a robot make decisions about how to act in its environment. The feedback mechanism is very important in an environment that is continually changing. During the rounds of the contest, the objects on the playing field will be changing their location (ie. the other robot moves, the drawbridge closes, or you bumb into a block). We strongly encourage you to use closed-loop feedback design when planning and implementing your strategy. There will be a smaller chance of random errors completely messing up your game if you use sensors wisely. See chapter 7 for more information on the control problems you may encounter.

5.1 Analog vs. Digital Sensors

The sensors you have can be split into two basic types: analog and digital. Analog sensors can be plugged into the analog sensor ports, which return values between 0 and 255. Digital sensors can be plugged into either the digital ports or the analog ports, but will always return either 0 or 1.

$$\text{ANALOG } 0 <= x <= 255$$
$$\text{DIGITAL } 0 \text{ or } 1$$

Each type of sensor has its own unique uses. Digital sensors, such as push-buttons, can tell you when you've hit a wall. Digital sensors always answer a question about the environment with a *YES* or *NO*. "Have I hit the wall?" Yes if the switch is closed, or No if the switch is still open.

Analog sensors, such as the bend sensors or photoresistors, can tell you how far the sensor has bent, or how much light is hitting the sensor. They answer questions with more detail. Analog sensors, however can be converted to digital sensors using thresholding. Instead of asking the question *How much is the sensor bent?* you can ask the question: *Is the sensor bent more than half way?* The threshold can be determined by playing around with the specific sensor.

5.2 Digital Sensors

Digital inputs all have *pull-up* resistors connected to them as shown in figure 5.1. Digital switches are wired such that the sensor is wired across the signal pin and ground. This means that when the digital sensors is closed, the signal is grounded or LOW. When the switch is open, the signal pin outputs +5V, or HIGH. This value is INVERTED by software, so reading the digital port with the switch open returns 0, while reading the digital port with the switch closed returns 1. With nothing plugged in, the value of a digital port should be 0.

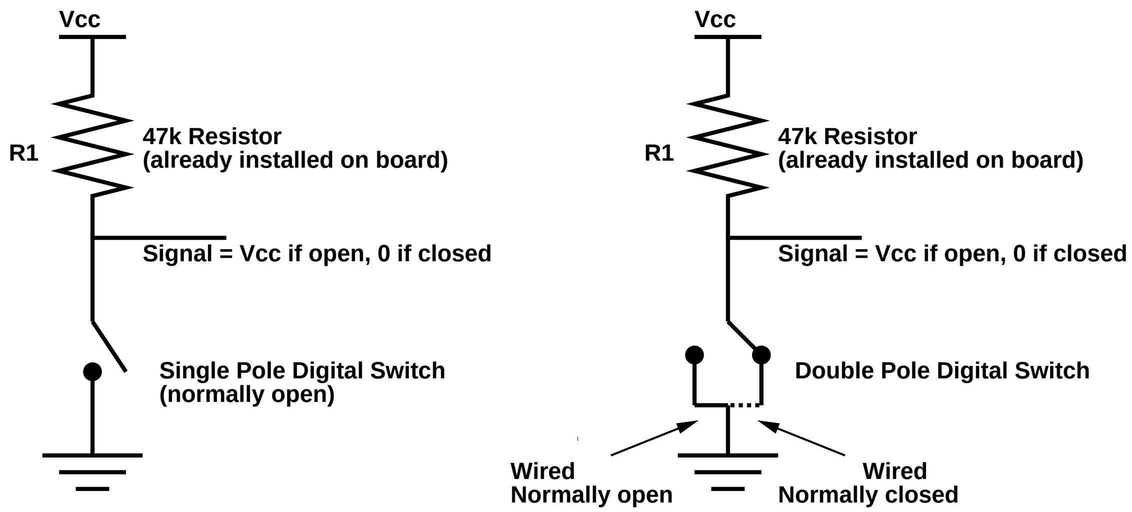

Figure 5.1: Generic Digital Sensor Schematics.

Digital sensors can be used in the analog ports on the 6.270 Controller board as well, relieving any restrictions the small number of digital inputs may cause. Typical analog values for digital sensors are somewhat above 250 for an open switch, and less than 20 for a closed switch. When using the IC command, digital(port)–where port is an analog port number (ie. greater than 7)–the sensor value is compared to a threshold value, and the command returns a 0 if the analog value is above the threshold or a 1 if the analog value is below it (remember the inversion of the actual signal that digital does?). This threshold's default value is 127, but it can be changed (See the section on IC commands for information on this).

A good way to get digital information from an analog sensor is to plug the analog sensor into a analog port and call it with the `digital (int port)` command. For example, a reflectance sensor would return a 0 for black or a 1 for white if read with the digital command–provided the threshold is properly set. This can reduce some of the programming complexity by abstracting away the thresholding. You should however experiment with the sensors to determine the range of thresholds you get and under what conditions these thresholds are valid.

It is not recommended to plug analog sensors into digital ports, however, because the digital ports threshold to conventional logic levels which cannot be adjusted to suit each analog sensor. The valid analog readings may fall into the invalid range for digital logic.

Here are some mountings and uses for some digital sensors in the 6.270 kit.

5.2.1 Dip Switches

There are four dip switches on the Expansion Board. They can be used to select user program options during testing. One dip switch will be used in the starting code for the contest to determine the side your robot starts on and at which frequencies it transmits and receives the modulated IR. They can also be useful for outside control of program parameters, like enabling certain functions or selecting programs to run. While these switches are connected to the analog port, they are really digital switches.

5.2.2 Micro-Switches

The standard kit includes three varieties of small switches, two types which are technically micro switches and a small push button type. These make great object detectors, so long as you are only interested in answering the question, *Am I touching something right now?* with a *yes or no*. This is often enough for responding to contact with a wall or the other robot or for actuator position sensing. A limit switch can be a good way to protect drive mechanisms which self destruct when over driven. This could be handy for limiting the motion of hinged joints or linear actuators by requiring that a switch be open (or closed, depending upon the situation) before running the motor and monitoring it while things are moving. They could also be used for extended user interface for testing and development purposes.

The two micro switches are double pull, which means they can be wired so that they return a one or a zero when not depressed. The only major difference is how you think about the device in your code. Reading a sensor can be thought of as asking a question. Here, the question could be, "Are you open?" or "Are you closed?" If you wire the switch normally open, the answers are yes and no, respectively, where they would be no and yes for a switch wired normally closed, all for the same situation where the switch is not depressed.

In some cases, a slight advantage may result from one arrangement, because there may be a difference between the position where the open side makes contact and the closed side breaks contact. When this is the case, the choice

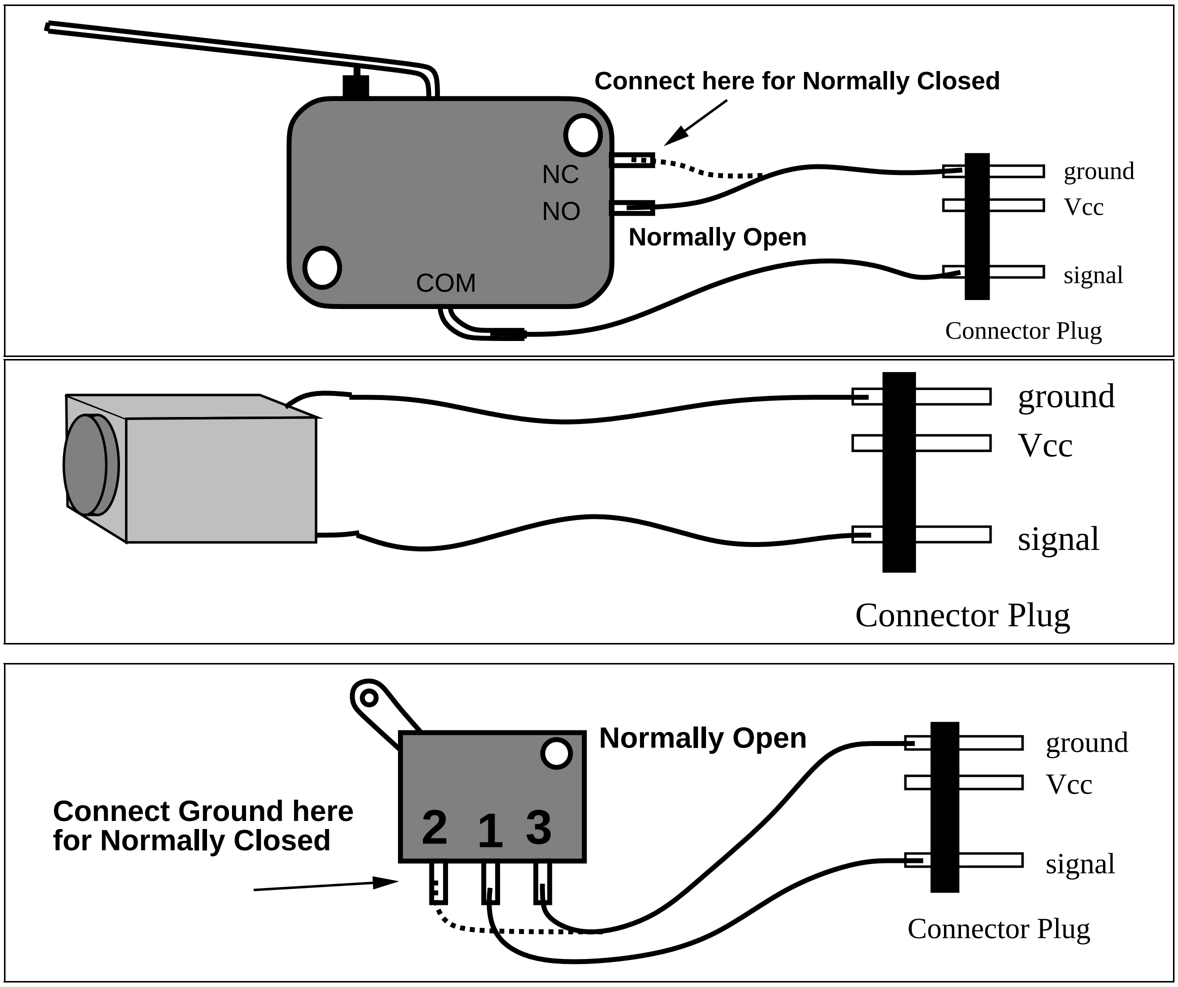

Figure 5.2: Microswitch Assemblies

of normally open or normally closed will affect how sensitive the switch is to outside forces. This can allow you to make a very touchy sensing device or help block out noise. The small black switches with the white lever arm respond to a shorter arm movement when wired normally open and require a little more movement to cause a transition in the normally closed configuration.

Bouncing is a problem found in many switches. At the point where the switch goes from open to close or vice versa, the output from the switch is very glitchy. The switch may output several transitions. Bounciness occurs especially when the switch is used in a sensitive mode. One way to *debounce* the switch is to add a delay between samples of the digital input. If the sampling is sparce enough, the bouncing section of the data will not be collected.

5.2.3 Sharp IR Detector

While it may not seem like a digital sensor because most of the light sensor we deal with are analog, it is a bonafide digital sensor because it detects infrared light modulated at 40kHz. Inside the tin can, there is a IR detector, amplifier, and a demodulator. The sensor returns a HIGH when there is no 40kHz light, and is LOW when it see the 40kHz light.

There is a lot of infrared light that is ambient in the air. Some components of this light are at 40kHz, and straight output from the sensor would look very glitchy. The sun produces a lot of IR light, and in the sun, the sensor output bounces all over the place. To eliminate the effect of the stray IR light, the IR emitters are modulated at 100 or 125 Hz (see section A.7 for more information on the IR transmission) and the output of the IR Detectors is demodulated to look for these frequencies.

5.2.4 Ball Sensor

The ball sensor can be used to measure crude angular elevations. This year you can use it to determine whether or not your robot has tipped over. The way the sensor works is that the metallic ball connects two of the metal leads in the clear casing. These sensors are just for fun, and you can play around with them to see if there you can use them.

The robots will move with a very jittery motion. This will cause the ball to bounce around. The readings from the sensor will be very unstable. You will need to average the sensor data and possibly take data when the machine is stopped.

5.3 Analog Sensors

The analog ports all have a *pull up resistor* which is a 47KΩresistor between +5 volts and the signal input. The analog readings are generated by measuring the amount of current flow through the pull up resistor. If no current flows through the resistor, the voltage at the signal input will be +5 volts and the analog value will be 255. The voltage at the signal pin can be simply calculated by:

$$V_{sig} = 5 - 47\Omega \times i$$

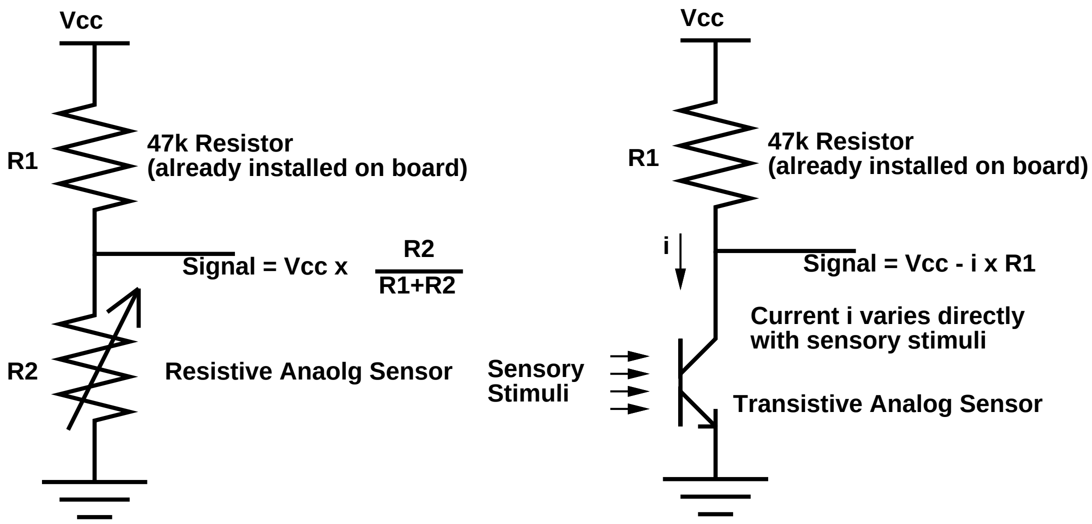

Figure 5.3: Analog Sensors Schematics

Reading the value of an analog port **without** a sensor will return a value above 250. With the sensor plugged in, the value should be less. This is one good way to check if one sensor fell out: write a piece of code that checks the values of the analog ports that you have sensors plugged into. If that value is above 250 or so, have it tell you to check the sensor.

There are two basic varieties of analog sensors.

5.3.1 Resistive Sensors

The resistance of resistive analog sensors, like the bend sensors or potentiometers, change with changes in the environment, either an increase in light, or a physical deformation. The change in resistance causes a change in the voltage at the signal input by the voltage divider relation.

$$V_{sig} = \frac{47\Omega}{47\Omega + R_{sensor}} \times 5V$$

5.3.2 Transistive Analog Sensor

Transitive analog sensors, like the photo transistors and reflectance sensors, work like a water faucet, where providing more of what the sensor is looking for opens the setting of the valve, allowing more current to flow, which makes the voltage drop across the resistor increase. As the voltage drop increases, the voltage at the signal decreases. A photo transistor reads around 10 in bright light and 240 in the dark.

One problem that may occur with transitive sensors is that the voltage drop across the resistor may not be large enough when the transistor is open. Some transitive devices only allow a small amount of current to flow through the transistor. A larger range for the sensor can be accomplished by putting a larger pull-up resistor. By having a larger resistor, the voltage drop across the pull-up resistor will be proportional to the resistance.

We will give example uses and mountings for each type of sensor. Keep in mind that these are only simple examples and are not the only possible uses for them. It's up to you to make creative use of the sensors you've been given.

5.3.3 Bend Sensors

Bend sensors were originally designed by Abrams Gentile Entertainment for use in the Nintendo(R) Power Glove. They have graciously donated the sensors to 6.270 for the past three years.

Bend sensors change their resistive value as they are bent in one direction, but do not change value if bent the other way. They are quite fragile. They can break if bent too far in either direction, and they are quite prone to damage from abrasion. There is also a limit to how far they can be bent before the value stops changing significantly. It does not take a great deal of pressure to bend the strip.

Bend sensors can be used in a variety of ways. The simplest is as a "wall feeler", where the sensor changes value as it comes in contact with other objects (not necessarily just the wall though). Another use might be a gate for the blocks

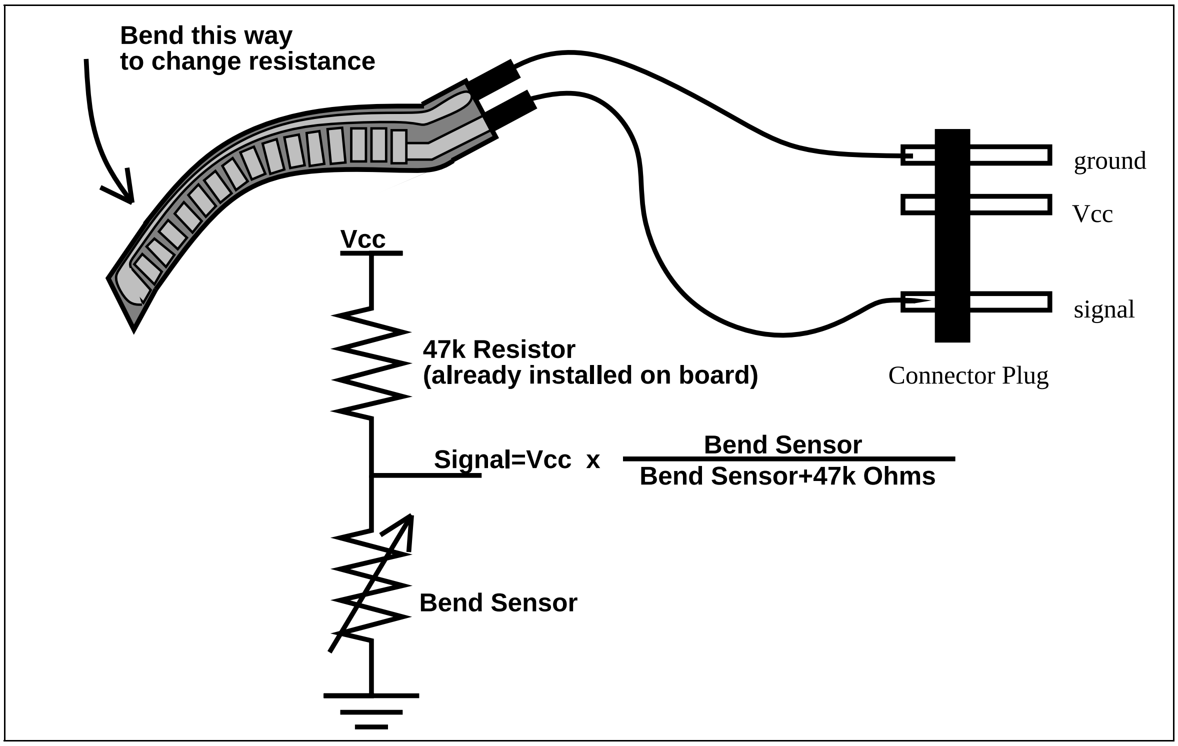

Figure 5.4: Bend Sensor Assembly

as they pass into the body of the robot, where the value changes as the blocks go through the gate.

Mounting bend sensors is always a trick. Hot glue is not recommended because it can damage the sensor. Modeling clay is allowed, but can be too flexible. One suggestion is to wedge it between two LEGO bricks whose ends have been cut a bit to leave a gap large enough for the sensor to wedge in between. The double-sided sticky tape from 3M might work well for mounting them in LEGO sized gaps. This may add some flexibility in the mounting which can distribute the strain of extreme bending over a larger portion of the strip, hopefully keeping the sensor functional, even under harsh conditions. Putting a layer of electrical tape or something similar over the lighter side may be a good preventative measure to keep scrapes and scratches from destroying your sensors.

When working with the bend sensors try to make sure that they are not located at a sharp corner. Inadvertent jamming can cause the sensor to be bent too far. This can crack the resistive etching which will break the sensor. When using the two bendsensor assembly, the sensors can be rubber banded together to make the sensor more rigid.

5.3.4 Potentiometers

Kits contain several sizes of potentiometers, also known as pots or variable resistors. There are rotary and linear pots. As the knob is turned or the handle slid, the resistance increases or decreases. This will produce different analog values.

Potentiometers should be wired with Vcc and ground on the two outside pins, and the signal wire on the center tap. This will, in effect, place the resistance of the potentiometer in parallel with the 47kiloΩpull-up on the expansion board and is more stable than just using one side and the center tab to make a plain variable resistor.

Potentiometers have a variety of uses. In the past, they have been used for menuing programs and angle measurement for various rotating limbs or scanning beacons. They can be used with a motor to mimic servos, but that's a difficult task. It is important to notice that the pots are not designed to turn more than about 270 degrees. Forcing them farther is likely to break them.

A potentiometer can be attached to a LEGO beam such that it can be used in place of a bend sensor. The rotation of the beam will produce a rotation in the potentiometer. See if you can come up with an assembly that can be used in place of a bend sensor. The advantage to such a sensor is that it is much sturdier than the bend sensor. The disadvantage is that it is bulkier.

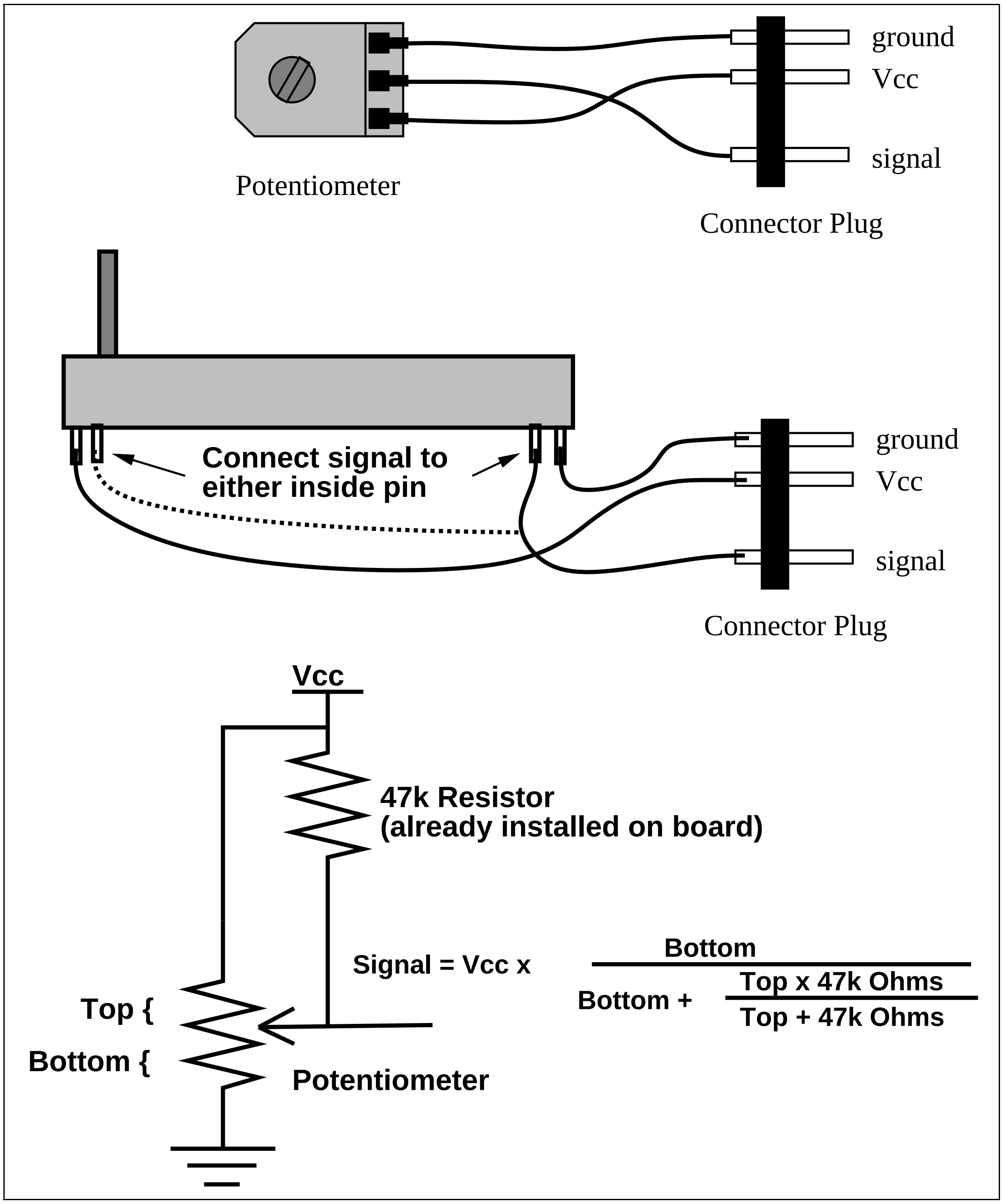

Figure 5.5: Potentiometer Assemblies

Linear Pots

A linear potentiometer can be use to measure precise linear motion, such as a gate closing, or a cocking mechanism for firing balls or blocks.

Frob-knob

The frob knob is the small white dial on the lower left corner of the Expansion Board. It returns values between 0 and 255 and provides a handy unser input for adjusting parameters on the fly or for menuing routines to select different programs. You may find it useful to glue a small LEGO piece to the frob knob to make turning it easier.

5.3.5 Photoresistors

A photoresistor changes its resistive value based on the amount of light that strikes it. As the light hitting it increases, the resistive value decreases. They are somewhat sensitive to heat, but stand up to abuse well. Try not to over heat when soldering wires to them.

As with all the light-sensing devices, shielding is VERY important. A properly shielded sensor can make the difference between valid and invalid values reported by that sensor. The idea is simple: restrict the amount of light striking the sensor to the direction you expect the light to be coming from. You do not want light from external sources (ie camera flashes or spot lights) to interfere with your robot. Black heat shrink tubing often works well to shield the photoresistor from external light sources.

One good way to get a feel for how these sensors work, and how your robot and software interact, is to make a light-sniffer. With two or more photoresistors, try to create a simple robot that can move around a room, either avoiding light or avoiding shadows in a controlled manner. Ambient light conditions play a major role in how to interpret the data from any light sensors. A combination of photocells, one pointed up and one pointed down, may be used to adjust for ambient light levels, which may be useful in some applications.

Photoresistors are probably the only sensor required to be on your robot. A starting light will be used to start each contest round, and the robot must be able to sense that light. You must place one photoresistor on the underside of your robot, probably near the center. Be sure to shield it as much as possible from the overhead ambient light. We will provide starting code that reads the value of that sensor to start the match.

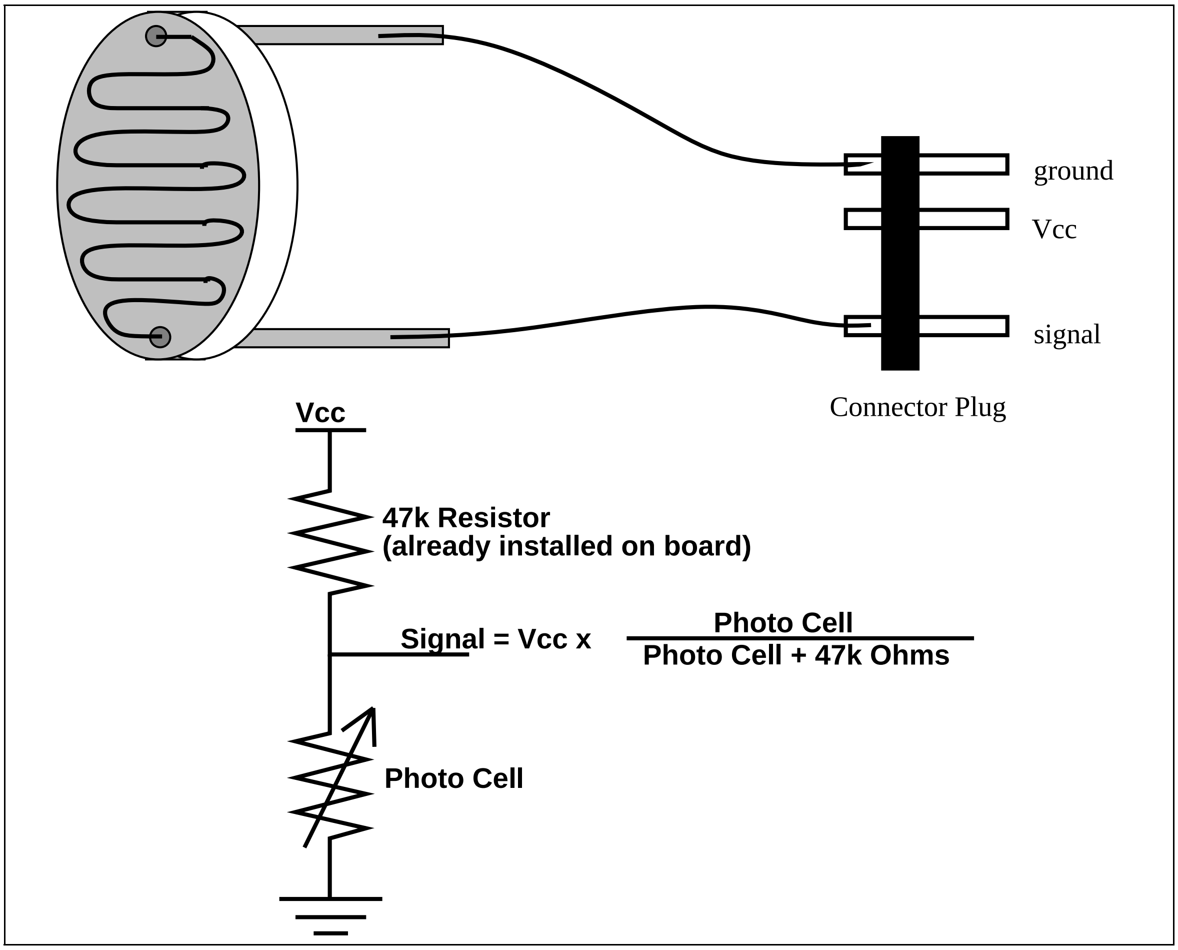

Figure 5.6: Photoresistor Cells

Mounting the photoresistors doesn't tend to be difficult. You can use a small amount of hot glue to attach the photocell to a LEGO brick, or double-sticky tape will also work. Be inventive.

5.3.6 Photo Transistors

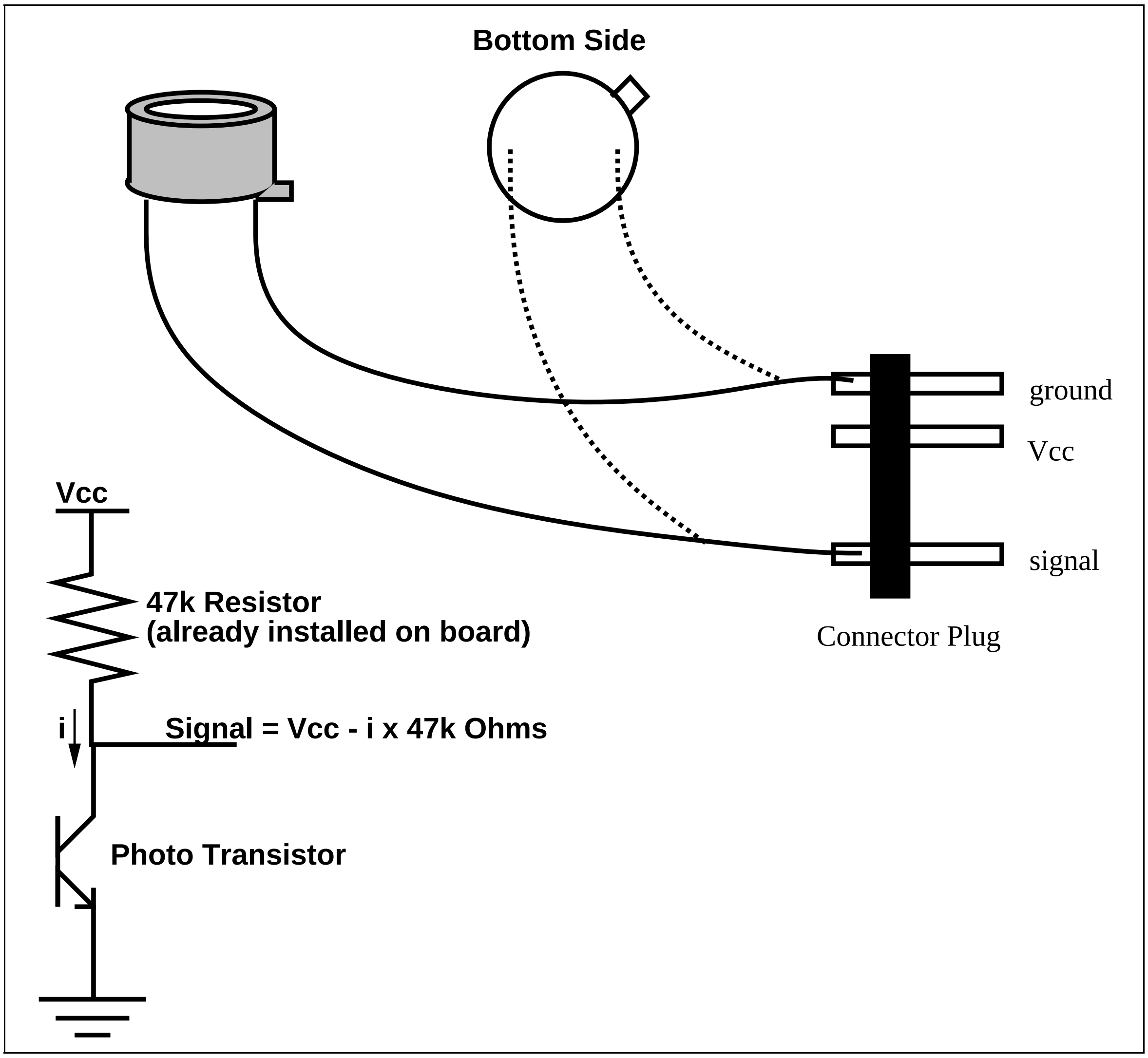

Figure 5.7: Photo Transistor Assembly

They have similar properties to the photoresistors. The main difference is that the phototransistors are usually tuned to a specific wavelength. The other important difference is that the time delay for a change in light conditions is much smaller for a phototransistor. This can be useful in doing fast control looking for polarized light.

The reflectance sensor in figure 5.9 can be used as a phototransistor alone if you run out of phototransistors or do not get enough range on your phototransistor.

5.3.7 Reflectance Sensors

A reflectance sensor is made up of a combination of an infrared or red LED and a phototransistor that is sensitive to the wavelength of light being emitted by the LED. Over dark surfaces, the light is absorbed, whereas over light surfaces, the light is reflected back to the phototransistor.

A reflectance sensor (figure **??**) can be made using discrete components instead of using the OPB5447 sensor shown in figure 5.9. This sensor may give a better range at a further distance from the reflected surface.

The reflectance sensors are useful for detecting what color the floor is. They could also be used as object detectors, but they are very near sighted and quite responsive to outside lighting. In any application, good shielding is an absolute requirement if any reliability is desired. The sensors are very sensitive to distance from the reflectin=g surface. Distances greater than an inch will give very poor reading, and distances that are too small will not allow the right to be reflected. The angle at which the light is reflected to the surface is important and can produce better or worse results at different distances.

5.3.8 Motor-Force Sensors

There are four motor force sensors built into the 6.270 Controller board, attached to motors 0 through 3. These sensors are included to detect when the motors might be stalled. When the motors stall, they draw a large amount of current, which appears as a large voltage on the analog inputs to the 6811. When a motor_force value increases sharply, that's a good sign, but not guaranteed, that the motor may be stalled. The value that it reaches will depend on the load attached to the motor. Experiment by stalling the motor yourself while printing the values on the LCD to determine a threshhold that's right for your robot.

Motor force values are not very accurate when you are driving the motors at anything less than 100that is used to drive the motors at lower speeds will cause the value to oscillate wildly, so it is recommended that you only use this information when you are driving a motor at full speed.

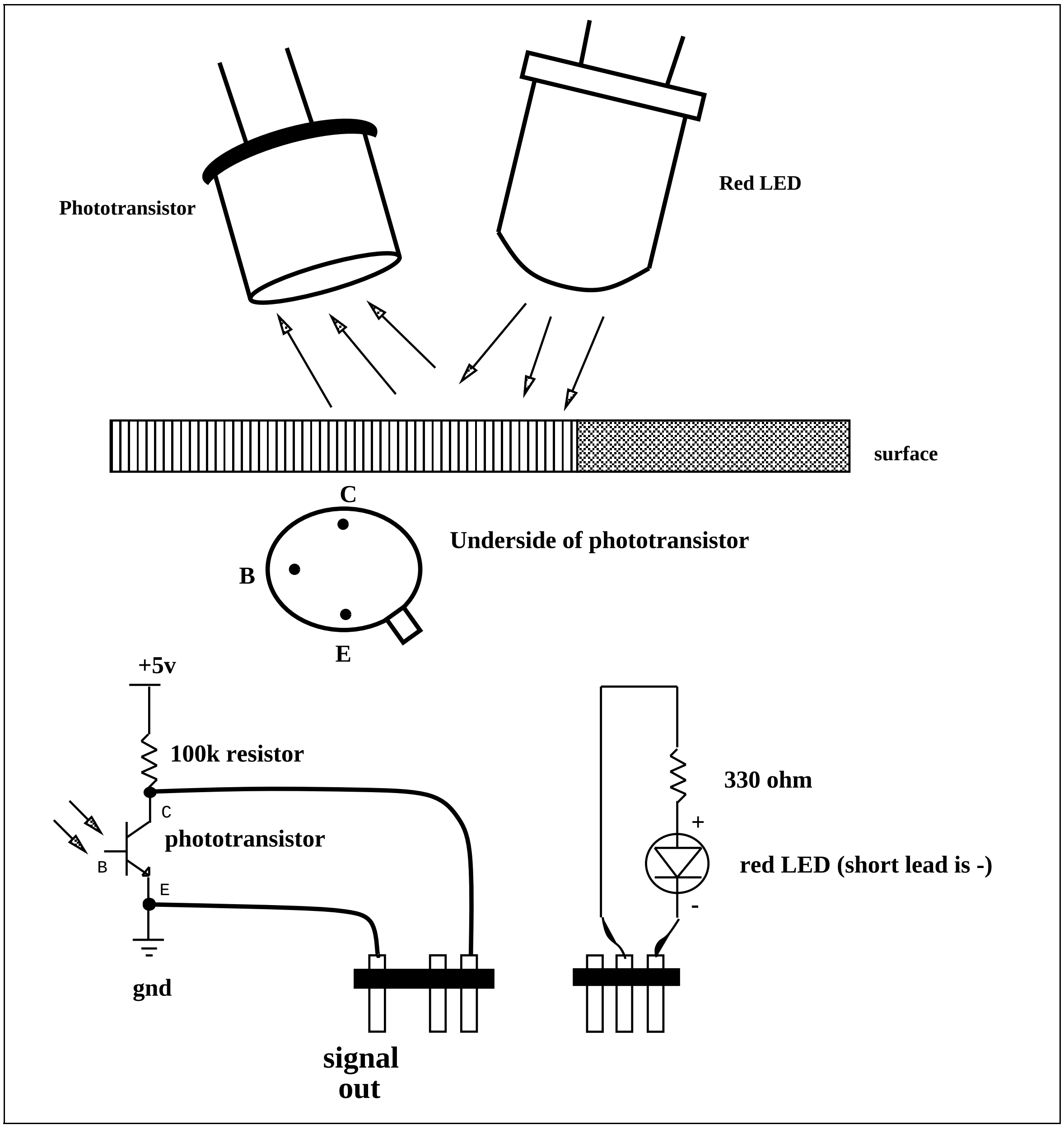

Figure 5.8: Reflectance Sensor

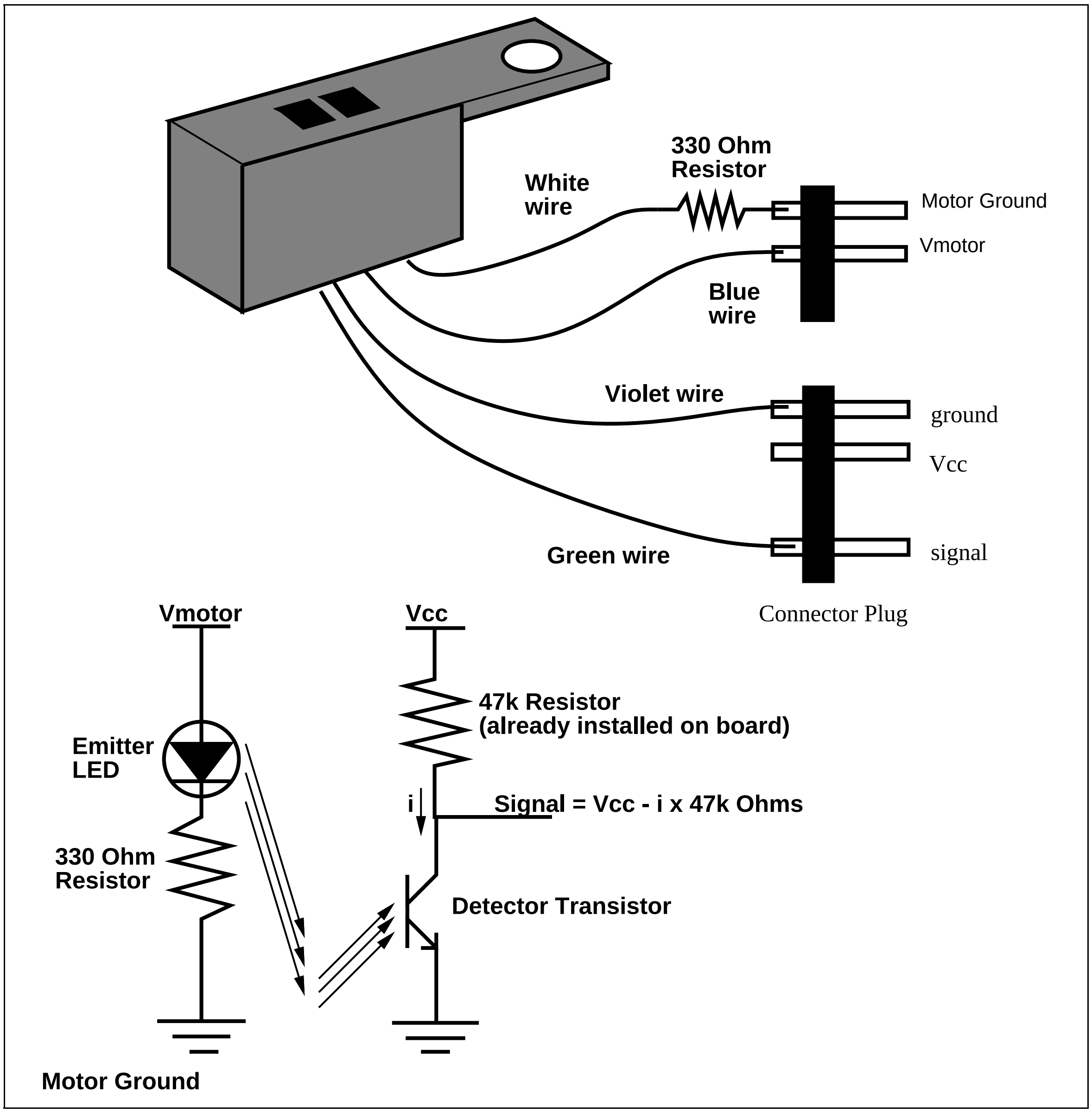

Figure 5.9: OPB5447 Reflectance Sensor

5.3.9 Breakbeam Sensors

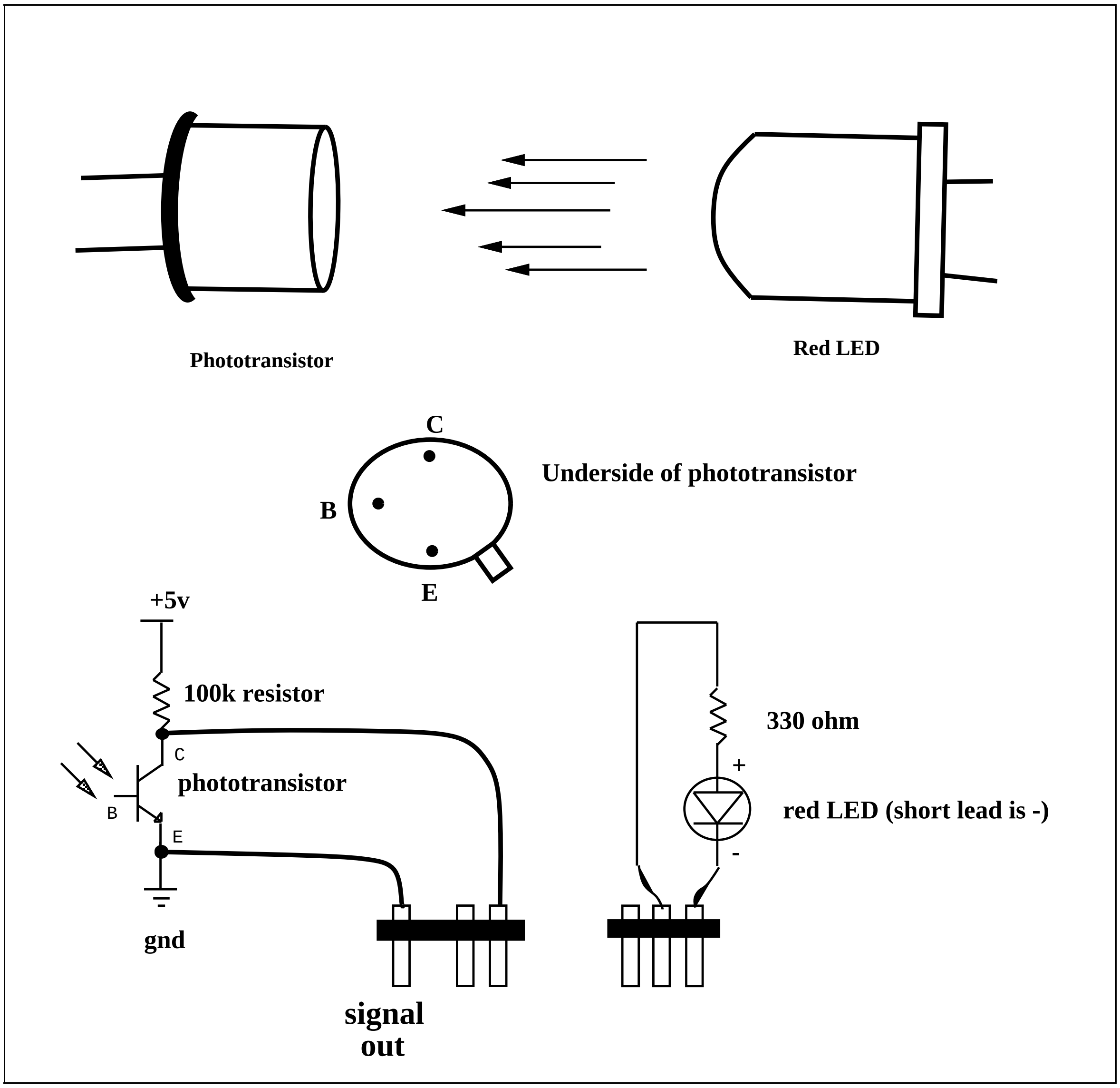

Figure 5.10: Breakbeam Sensor using discrete components.

Breakbeam sensors are another form of light sensors. Instead of looking for reflected light, the photosensor looks for direct light as shown in figure 5.10.

The sensor is useful in detecting opaque objects that prevent the light beam from passing through. This can be useful in detecting block between gripper, or when block passes through a passageway. The sensor does not need to detect the block very quickly so the phototransistor can be plugged into the analog

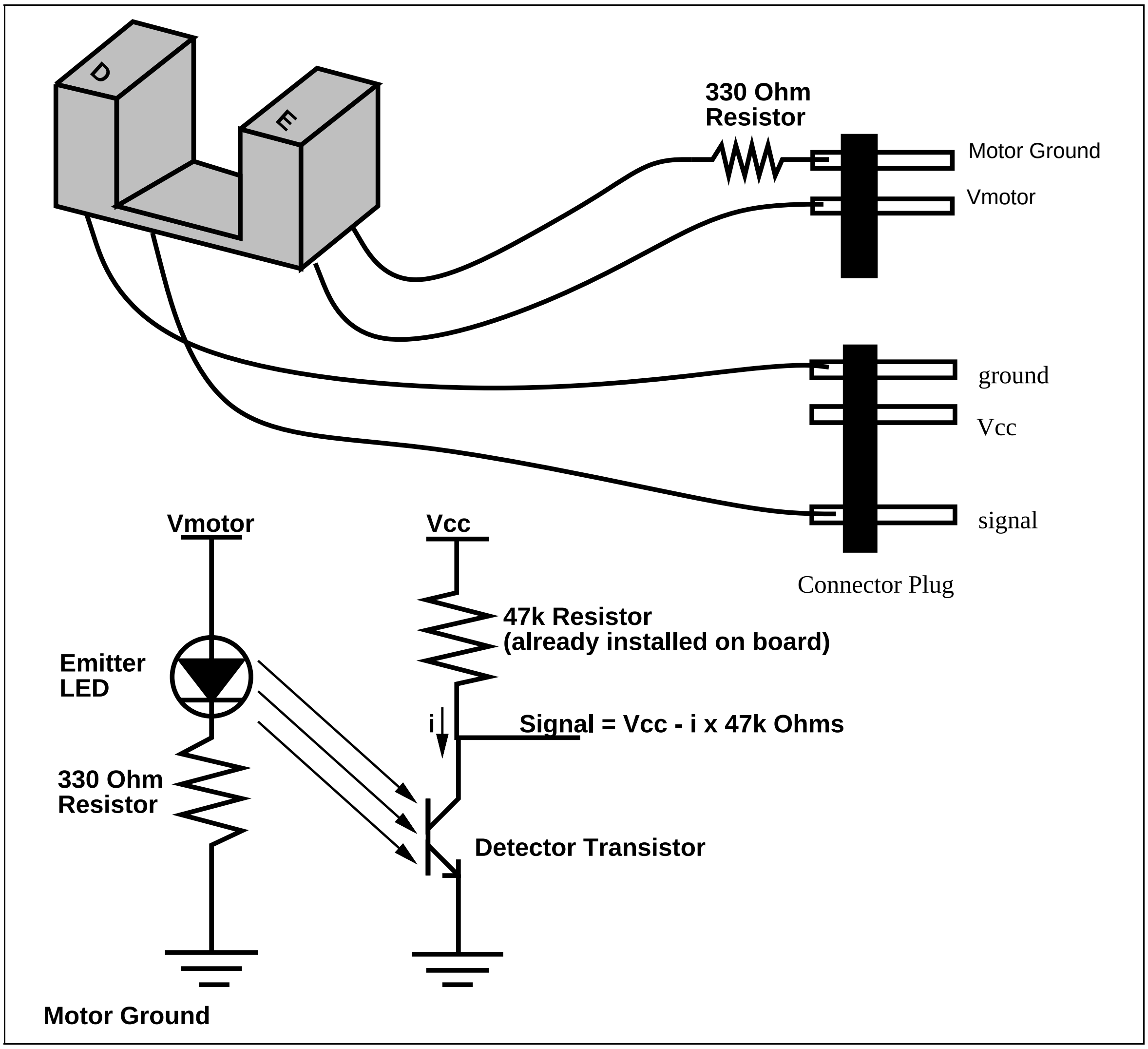

Figure 5.11: Breakbeam Assembly

port.

The breakbeam sensors can also be used for counting holes or slots in a disk as it rotates (see figure 5.12), allowing distance traveled to be measured. Since this requires a very fast sampling, the sampling needs to be done at the assembly language level. We have implemented shaft-encoder routines to do the fast sampling. But inorder to use these routines the sensors should be plugged into the lower two digital ports if the rate at which the holes or slots go by is very high.

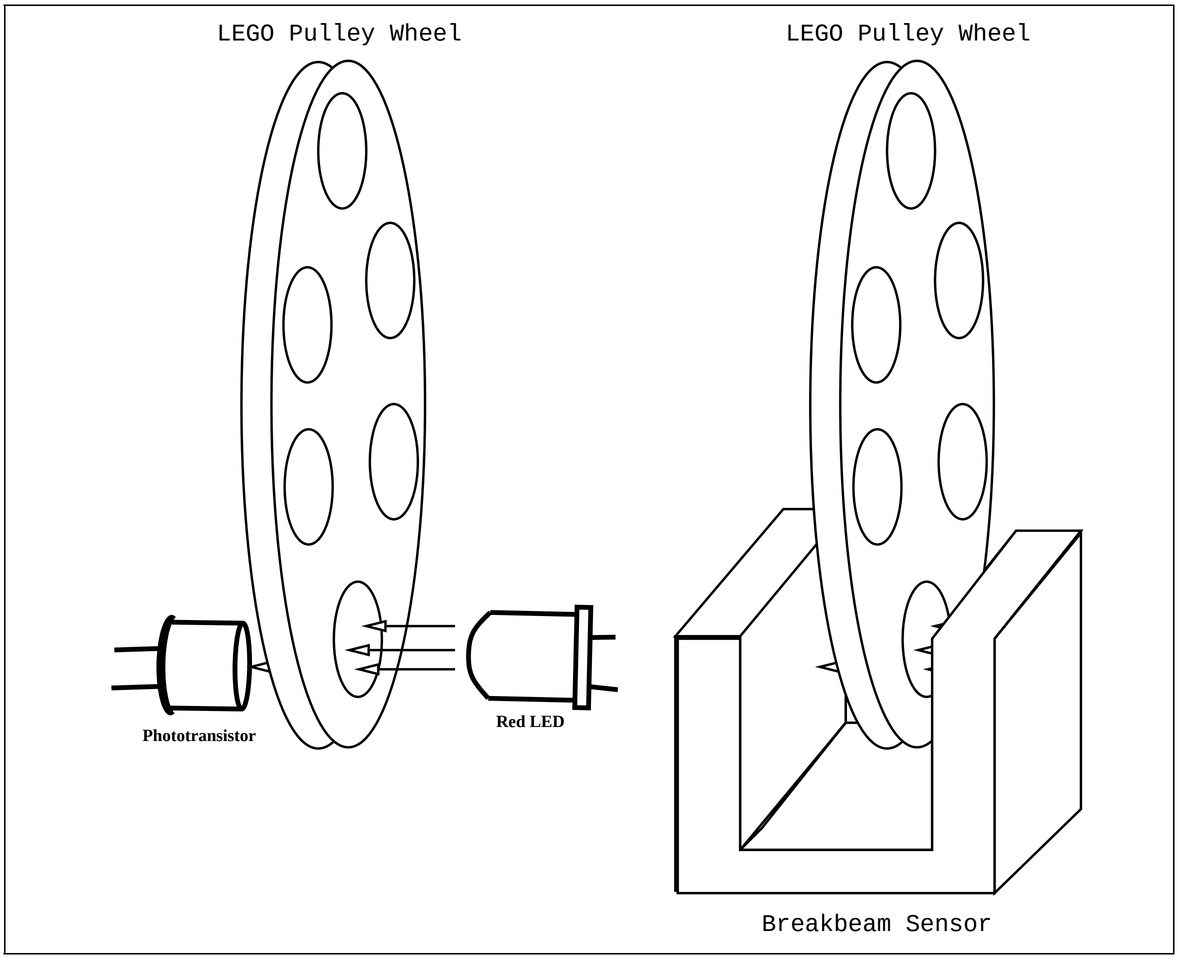

Figure 5.12: Shaft encoding using a LEGO pulley Wheel

Before you use the analog sensors in the digital switch you must make sure that there is a full swing in the analog reading from when the light goes through to when the light is blocked.