

6

understanding sensors

The LEGO MINDSTORMS NXT 2.0 robotics kit includes three types of sensors: Ultrasonic, Touch, and Color. You can use these sensors to build a robot that makes sounds when it sees you or to build a vehicle that drives around while avoiding walls or that follows the black line on the Test Pad. This second part of the book will teach you what you need to know in order to create working robots with sensors.

To learn how to work with sensors, you'll upgrade the Explorer robot by adding several sensor attachments to it to create the Discovery robot shown in Figure 6-1. You'll learn to create programs for robots with sensors as you upgrade your robot with an Ultrasonic Sensor attachment. Once you have a good working knowledge of how to program with sensors, you'll continue creating more sensor attachments for this robot in Chapter 7.

what are sensors?

LEGO MINDSTORMS robots can't actually see or feel the way humans do, but by adding *sensors* to them, they can collect and report information about the environment around them. Your programs can interpret sensor information in ways that will make your robot seem to respond to its environment as if it is experiencing it. For instance, you could create a program that makes the robot say "Blue" when one of its sensors sees a piece of blue paper.

understanding the sensors in the NXT 2.0 kit

Your NXT kit contains three sensors, as shown in Figure 6-2. The *Ultrasonic Sensor* reads the distance to objects, the *Touch Sensors* detect button presses, and the



Figure 6-1: The Discovery robot: an enhanced version of the Explorer equipped with an Ultrasonic Sensor to make the robot "see"

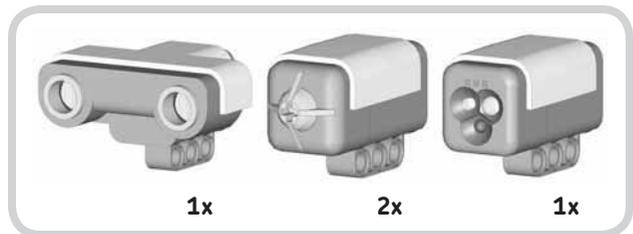


Figure 6-2: The NXT kit comes with an Ultrasonic Sensor (left), two Touch Sensors (middle), and a Color Sensor (right).

Color Sensor detects the color of a surface (among other things, as you'll learn in Chapter 7).

You connect the sensors to the NXT via *input ports*, numbered 1 through 4, as shown in Figure 6-3.

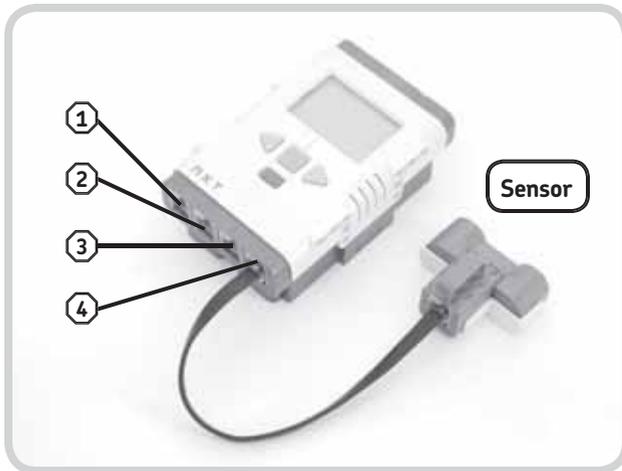


Figure 6-3: Sensors are connected to input ports.

In this chapter, you'll explore the Ultrasonic Sensor, and you'll take a more detailed look at the other sensors in Chapter 7. The programming techniques that you'll learn for the Ultrasonic Sensor can be used for all the sensors in the kit.

understanding the ultrasonic sensor

The Ultrasonic Sensor serves as your robot's eyes. To "see," the sensor measures the distance between it and other objects (as shown in Figure 6-4). The NXT retrieves the information from the sensor and uses its measurements in programs. For example, using input from this sensor, you

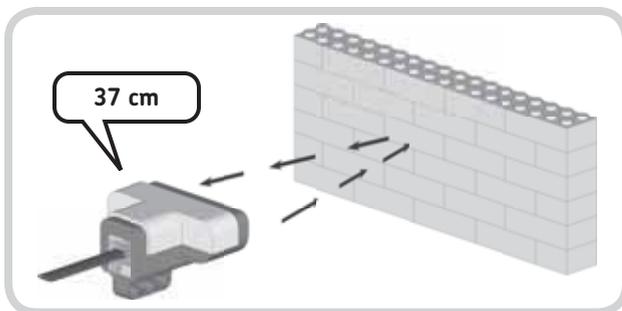


Figure 6-4: The Ultrasonic Sensor is used to detect objects by measuring the distance to them. It can see things up to 200 cm (80 inches) away, but the farther the object is, the harder it is for the sensor to see it. When the sensor doesn't see anything, it reports a value of 255 cm.

could make your robot say "Hello" when the ultrasonic sensor reports an object in front of it that's nearer than 50 cm.

As you'll learn in this and the following chapters, you can use the Ultrasonic Sensor in a number of interesting ways. For example, you can use it to make a vehicle avoid walls (Chapter 6), create an intruder alarm and detect targets to shoot at (Chapter 8), find objects to grab (Chapter 13), and even detect a ceiling so that a vertical climber knows to go down again (Chapter 15).

creating the ultrasonic sensor attachment

To begin, add an Ultrasonic Sensor to the Explorer robot as shown in the directions on the following page.

NOTE If you have trouble following these steps, try disconnecting the motor cables and then reattaching them after you connect the Ultrasonic Sensor. Be sure to connect the Ultrasonic Sensor to input port 4 on the NXT using a medium-sized cable.

polling sensors

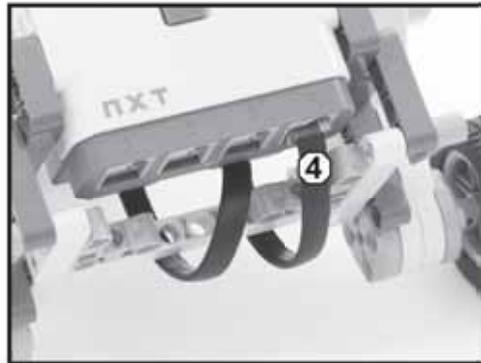
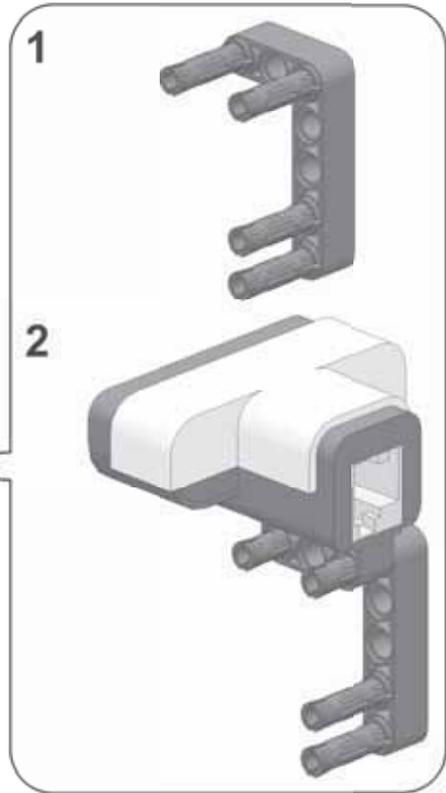
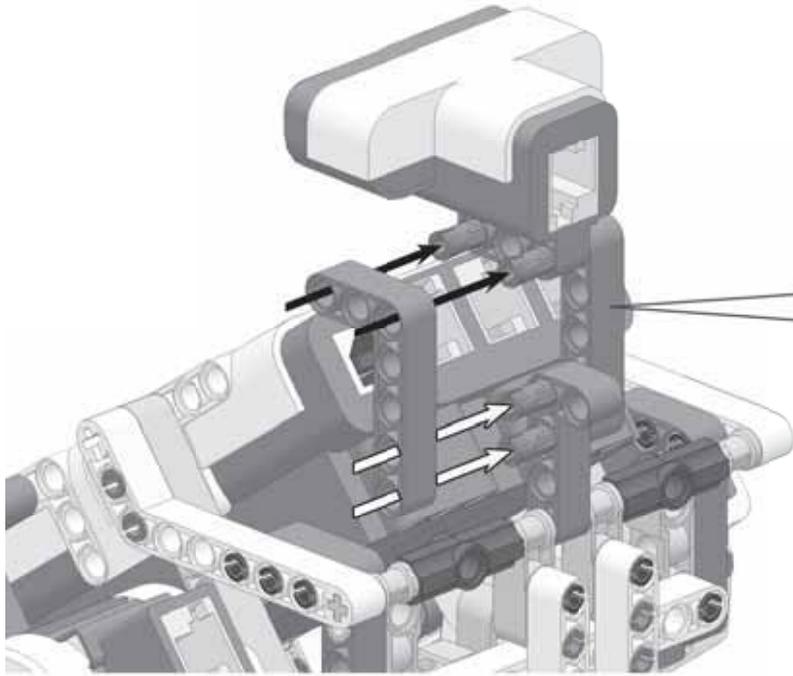
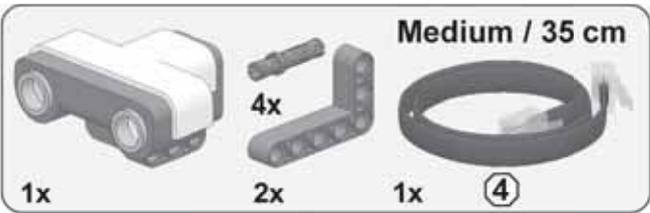
Without having to do any programming, you can view the sensor readings on the NXT's *View menu*. Gathering information from a sensor is sometimes referred to as *polling*. To poll a sensor, follow these steps:

1. Turn on your NXT, navigate to the View menu (Figure 6-5), and select the sensor you want to poll.
2. Choose **Ultrasonic cm** (or **Ultrasonic Inch**).
3. Select the input port the sensor is connected to (port 4), and you should see the sensor's measurement, which is 37 cm in this case.

DISCOVERY #19: MIND YOUR HEAD!

Difficulty: Easy

How could you find out how far your robot is from the ceiling of a room? Use the View mode and the Ultrasonic Sensor to find out, making sure to point the sensor's "eyes" toward the ceiling while measuring. If you see only question marks on the screen, you might need to hold the sensor up a little closer to the ceiling.



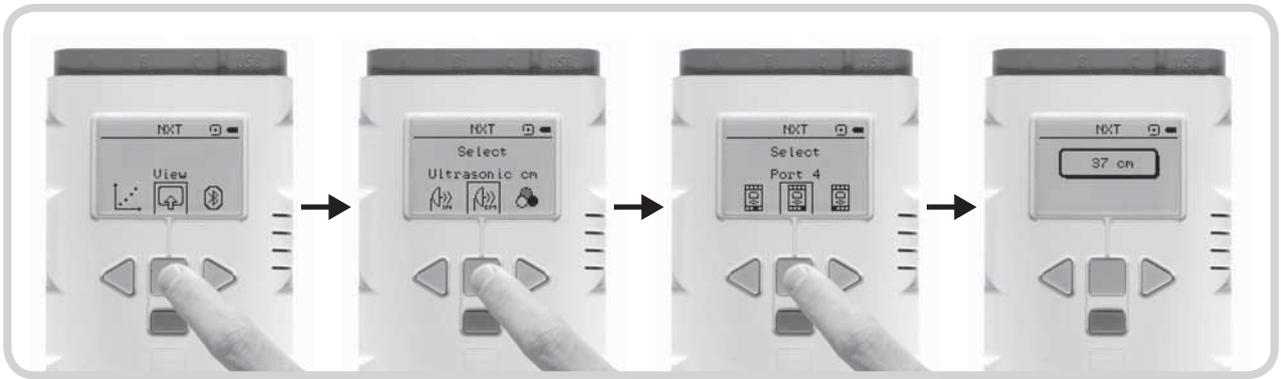


Figure 6-5: Polling a sensor with the View menu

programming with sensors

You've just seen how to poll a sensor yourself. Programs can also poll a sensor in order to use the sensor's data. As an example, you'll create a program that has the robot play a sound when the Ultrasonic Sensor sees something that's closer than 50 cm (20 inches), as shown in Figure 6-6.

You can use several programming blocks to poll sensors including the Wait and Loop blocks as well as the Switch block.

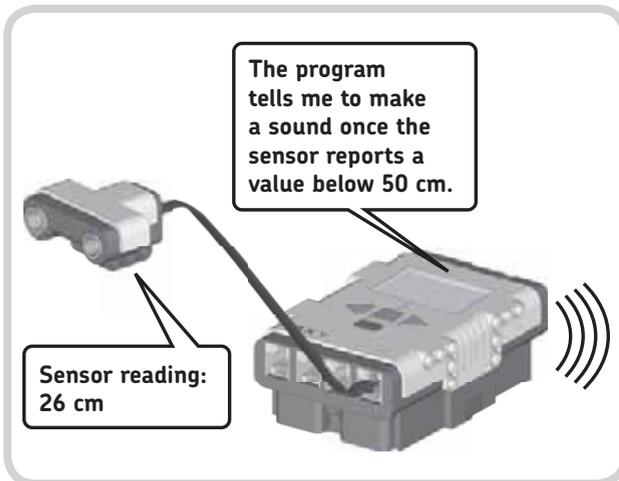


Figure 6-6: The program that runs on the NXT polls the sensor constantly. The program pauses until a sensor reading less than 50 cm is reported, at which point it plays a sound.

sensors and the wait block

You can use a Wait block to pause a program for several seconds, but also to pause until a sensor reading goes above or below a certain value. For example, Figure 6-7 shows a Wait block halting a program until the Ultrasonic Sensor detects a sensor value less than 50 cm. This value of 50 cm is called the *trigger value*. Once this trigger value is reached, the sensor is *triggered*, the Wait block stops waiting, and the next block in the program (i.e., a Sound block) runs.

using the configuration panel

You've just learned how a Wait block is used to poll a sensor. Now you'll have a look at the configurations of this block.

First select the Wait block configured to poll the Ultrasonic Sensor from the Programming Palette, as shown in Figure 6-7. (This is really just the Wait block you've been using with the Control setting set to Sensor instead of Time.)

- * You use the Port setting to select the input port to which the sensor is connected.
- * You set the trigger value in the Distance setting in the Until box by entering its value or by dragging the slider to the left (closer) or right (farther). You use the Distance setting to select whether the block should wait until a value above (>) or below (<) the trigger value is reached.
- * You use the Show setting to choose Centimeters or Inches.

seeing the sensors and the wait block in action

Now you'll create the **Discovery-Wait** program that plays a sound when the Ultrasonic Sensor sees something closer than 50 cm (20 inches), as shown in Figure 6-8.

Before running this program, make sure that there is nothing in front of your robot. When you run the program,

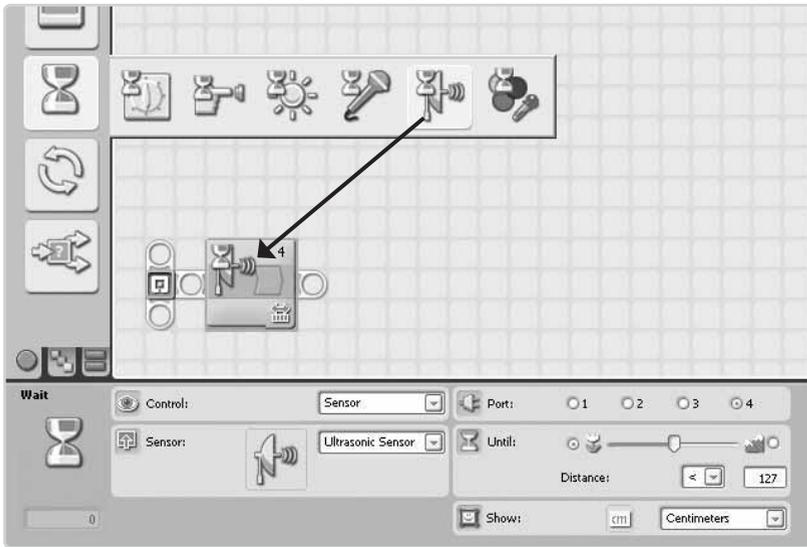


Figure 6-7: The Wait block configured to poll an Ultrasonic Sensor

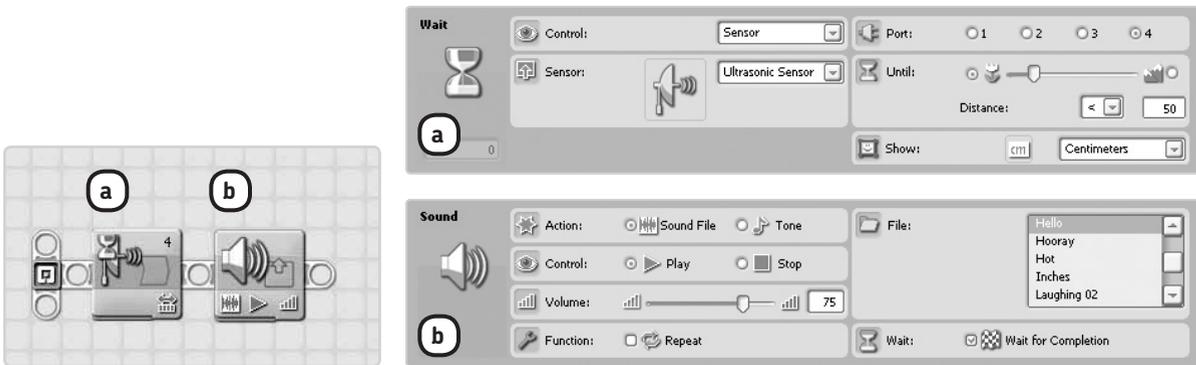


Figure 6-8: The configuration of the blocks in the Discovery-Wait program. Experiment with the trigger value (50 cm here) by changing it to make the robot respond differently.

nothing should happen at first, but as you move your hand in front of the Ultrasonic Sensor, the sensor should detect your hand once it's closer than 50 cm. Once your hand is detected, the Wait block should stop waiting, and your robot should play a sound.

Now, before you proceed with other sensor programming techniques, try to solve Discovery #20 to get a better understanding of how to work with sensors.

DISCOVERY #20: HELLO AND GOODBYE!

Difficulty: Medium

Can you create a program that has the robot say “Hello” when it detects your hand in front of the Ultrasonic Sensor and then “Goodbye” when you move your hand away?

HINT Place and configure the sequence of the two blocks in the Discovery-Wait program (Figure 6-8) twice. Configure the first Wait block to wait until it sees something closer than 50 cm, and configure the second Wait block to wait until the sensor sees something farther than 50 cm. Once you've confirmed that your program works, drag all the blocks into one Loop block so that your robot will repeatedly tell you when it detects your hand.

avoiding walls with the ultrasonic sensor

The next program, **Discovery-Avoid**, will make the Discovery robot drive around a room and turn around when it sees something in order to prevent it from bumping into an obstacle such as a wall. You can see an overview of the program in Figure 6-9.

Next you'll re-create the program with programming blocks. You can accomplish each action described in Figure 6-9 with one block. You'll use a Move block to turn on the motors with the Duration option set to Unlimited, and then you'll wait for the sensor to be triggered with a Wait block. (Note that while the program waits, the robot is still moving forward.)

Once the robot sees something, you use a Move block to turn it around (and stop its unlimited movement). For turning around, the block's Duration option is set to a specific number of rotations. After the robot turns around, the program returns to the beginning, which is why you must place the three blocks used inside a Loop block that's configured to loop forever.

Create the program now, as shown in Figure 6-10.

sensors and the loop block

As you learned in Chapter 5, you can configure a Loop block in many ways. You can configure Loop blocks to loop a certain number of times, loop for a specified amount of time, or loop forever. These conditions indicate when a Loop block should stop looping.

You can use a sensor to make a Loop block stop repeating. For example, you can repeat a sequence of Move blocks inside a Loop block until the Ultrasonic Sensor sees something that's closer than 25 cm (10 inches).

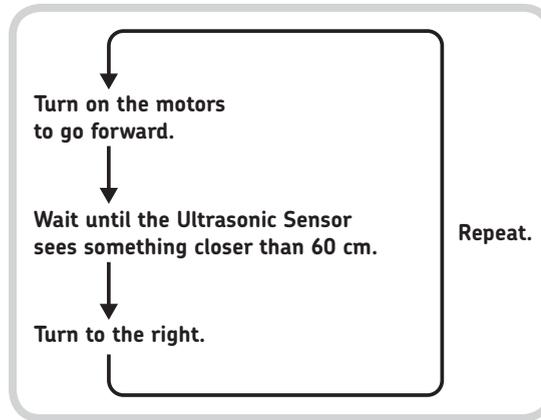


Figure 6-9: The program flow for the Discovery-Avoid program. After turning right, the program returns to the start, and the robot moves forward again.

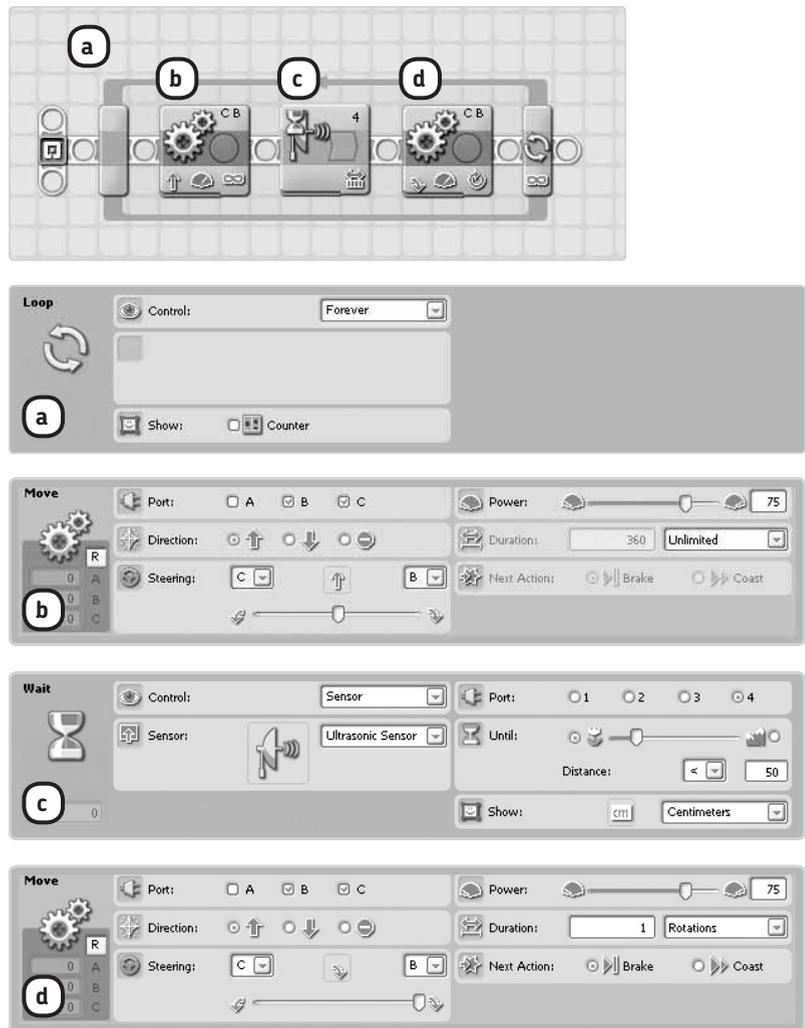


Figure 6-10: The configuration of the blocks in the Discovery-Avoid program

DISCOVERY #21: AVOID WALLS AND A BAD MOOD!

Difficulty: Easy

Expand the Discovery-Avoid program by making it display a happy face on the NXT's display as it moves forward and by making it display a sad face when it is turning around to avoid the wall.

HINT To do so, try putting two Display blocks somewhere in the Loop block.

DISCOVERY #22: FOLLOW ME!

Difficulty: Medium

Make the Discovery robot follow you in a straight line. When you place your hand in front of the robot, it should stop, but when you move your hand away, it should move forward until it sees your hand again.

TIP Have one block wait for the hand to come close (and then use a Move block to stop the robot's movement) and another block wait for the hand to move away, and then continue the robot's movement. Put all the blocks in a Loop block.

DISCOVERY #23: HAPPY TUNES!

Difficulty: Medium

Use a Loop block to have the robot play a tune until the sensor spots someone watching, at which point the robot should scream and turn its head the other way.

HINT You can use the My Block that you made in Discovery #16 in Chapter 5 for your tune. If you have yet to create your own tune, simply select a sound file from the list in a Sound block.

To use a sensor to control when a loop should stop, select **Sensor** in the Control box, and then select the sensor (**Ultrasonic**) you would like to use and the port (**4**) to which the sensor is connected. Set the trigger value in the Until box to specify exactly when the loop should stop. You can see how to make these configurations in the following sample program.

seeing the sensors and the loop block in action

Now make the **Discovery-Loop** program shown in Figure 6-11. When you run this program, the robot should move back and forth repeatedly until the sensor spots something closer than 25 cm (10 inches).

This program doesn't always respond to objects visible within 25 cm of the robot because it polls the sensor only once per repetition, just after all the blocks inside the loop complete. For this reason, when you quickly move your hand in front of the robot while the first block is running, the robot won't notice your hand.

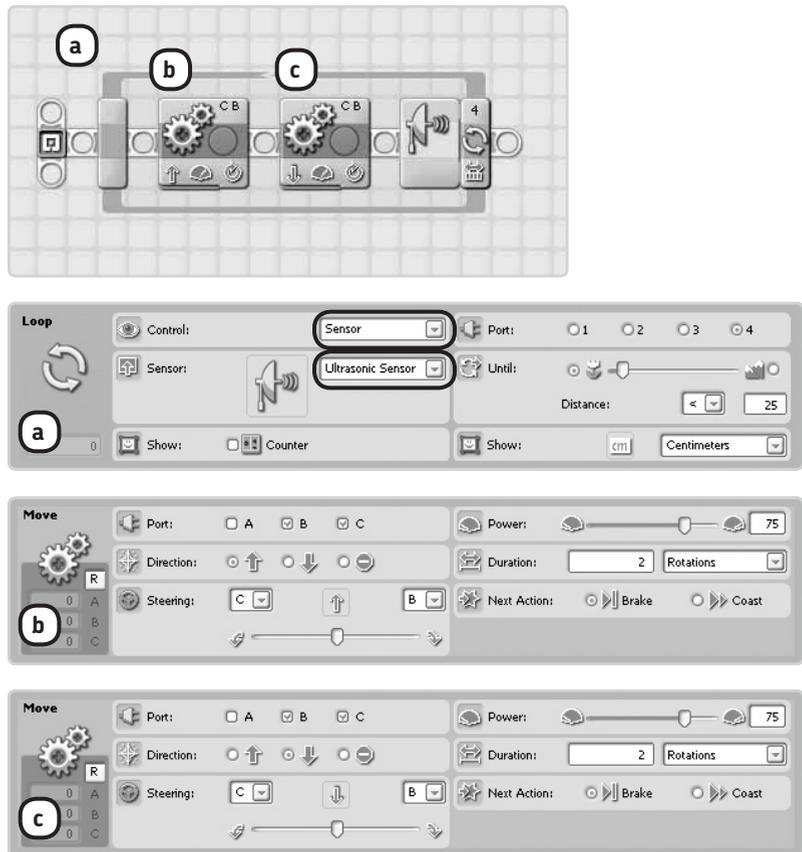


Figure 6-11: The configuration of the blocks in the Discovery-Loop program

9

strider: the six-legged walking creature

You can make interesting models that move on wheels, but it is also possible to make robots that walk. Such creations are slightly more challenging to make, but in this chapter you'll find instructions to build Strider, a six-legged walking creature, as shown in Figure 9-1. Once you finish building it, you'll program it to walk around and respond to human interaction.

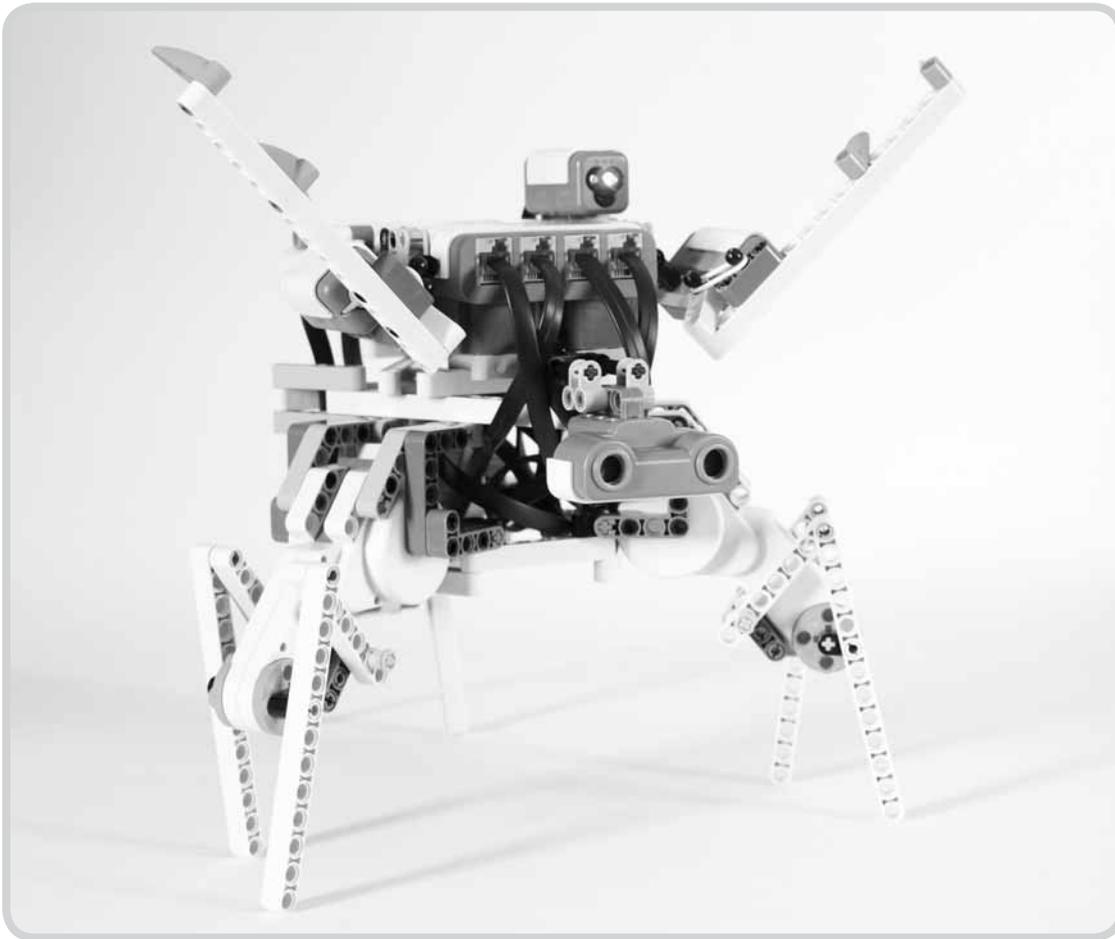
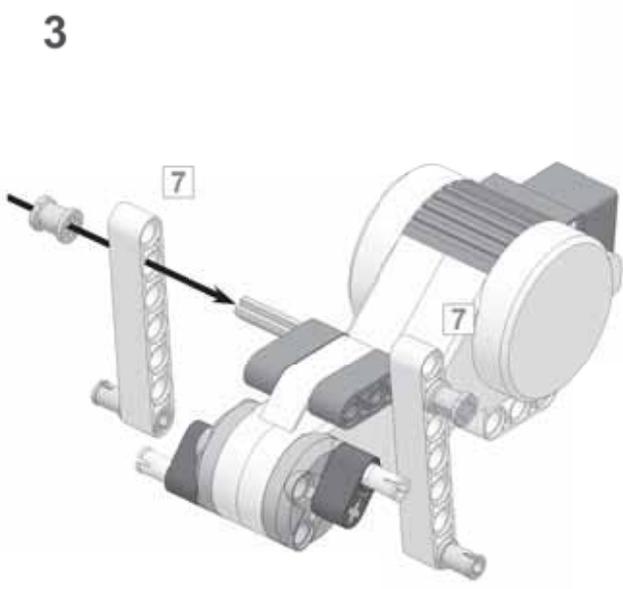
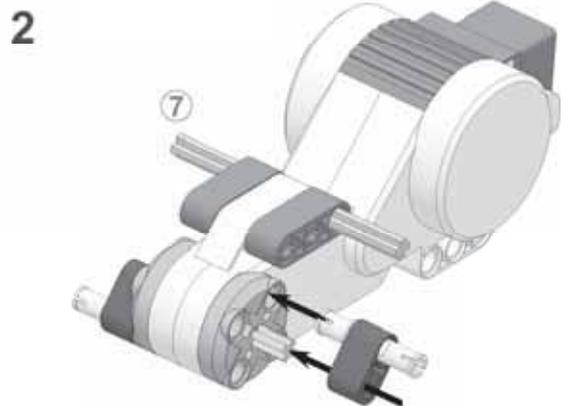
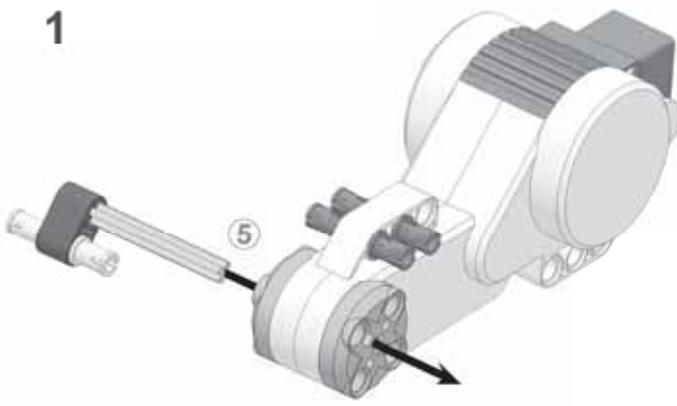
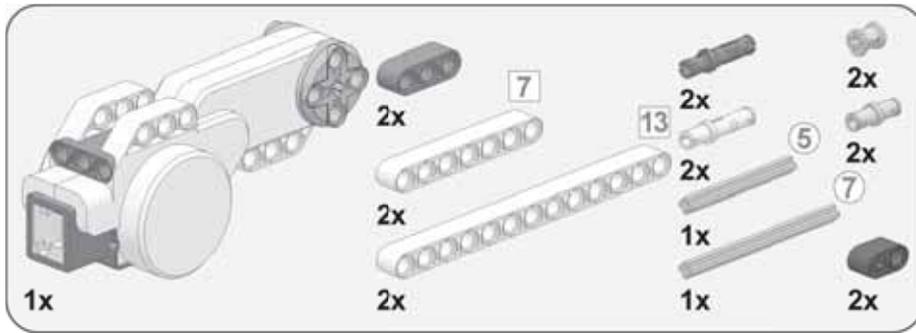
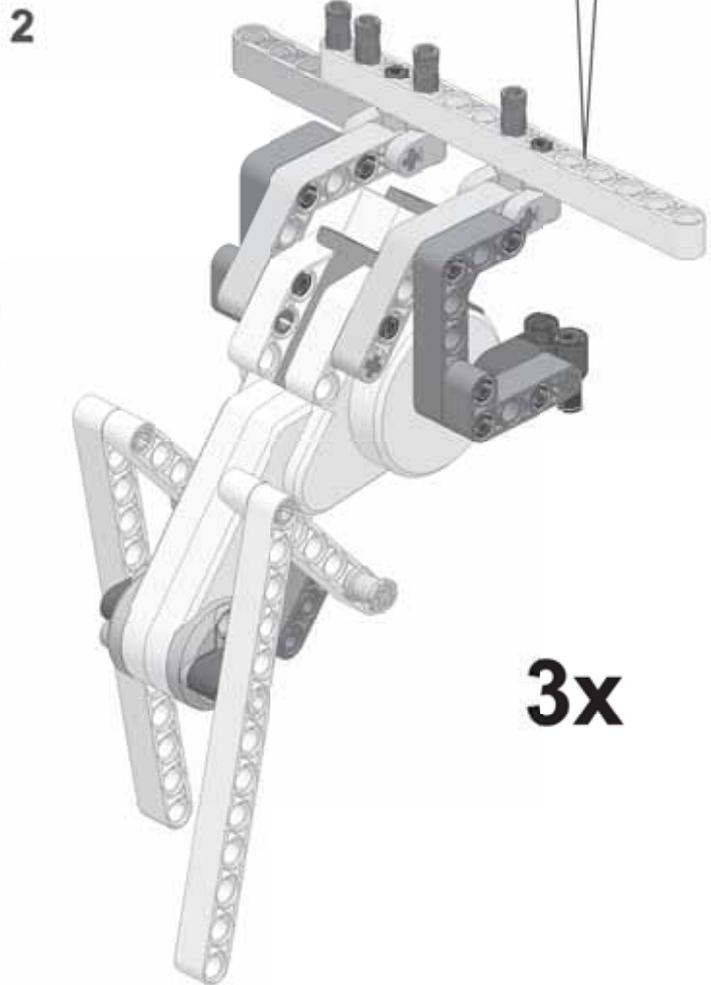
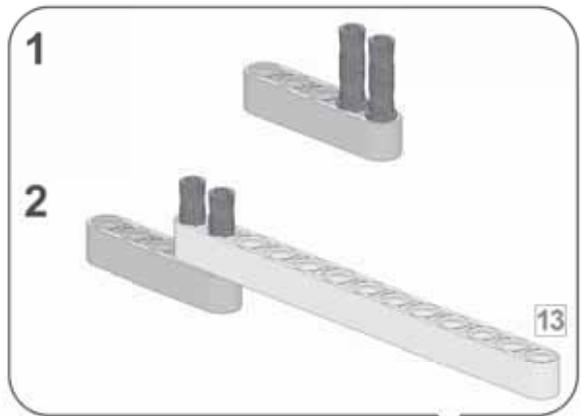
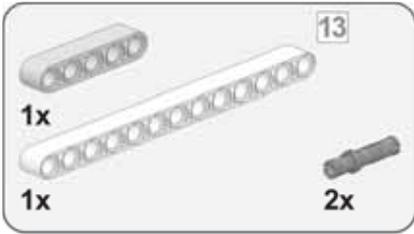
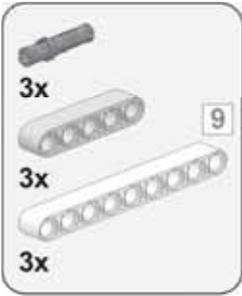


Figure 9-1: Strider

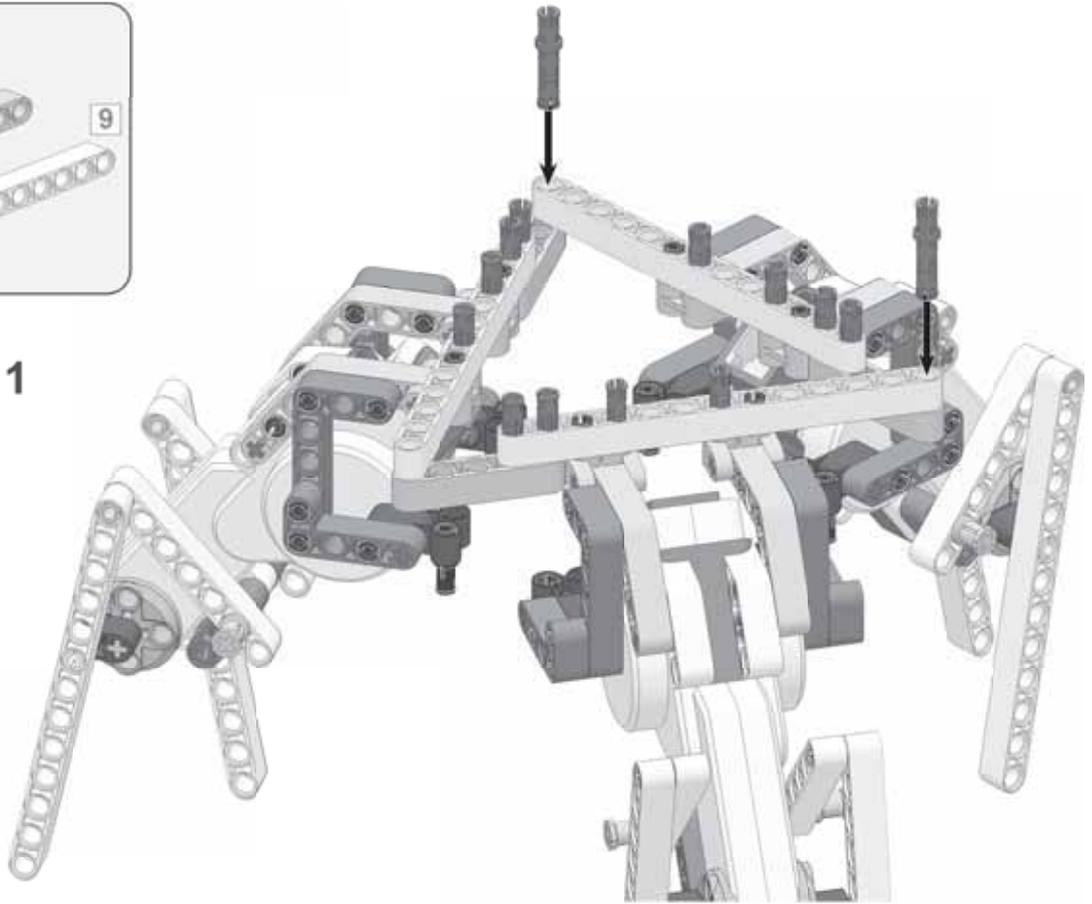




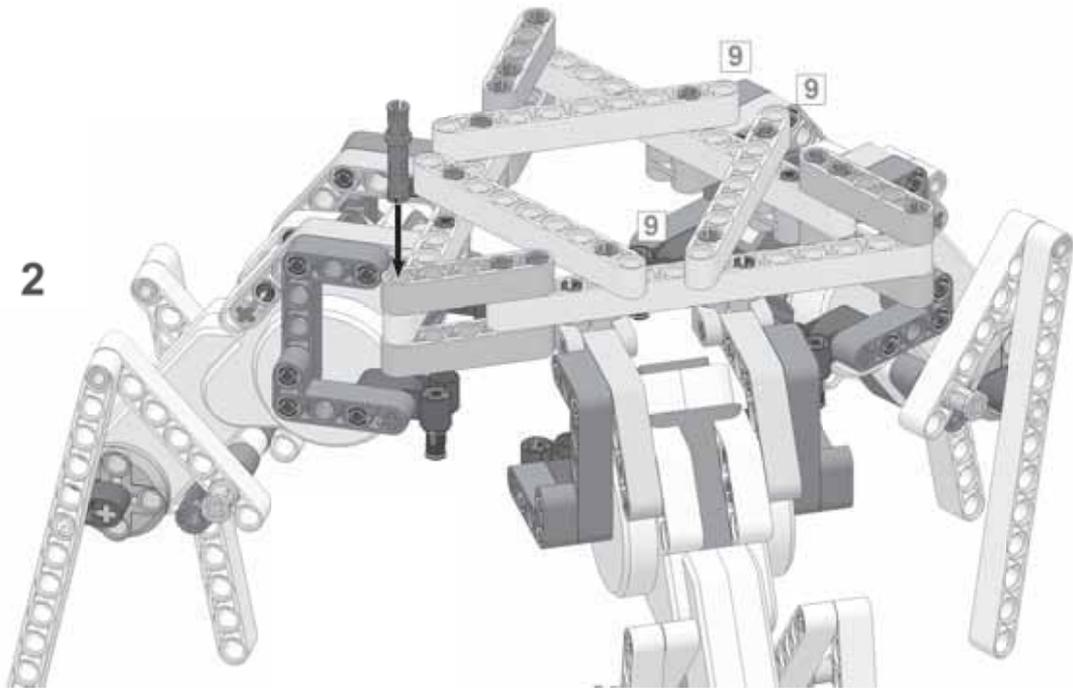
3x

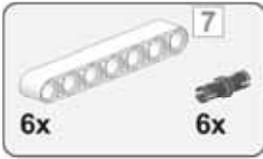


1

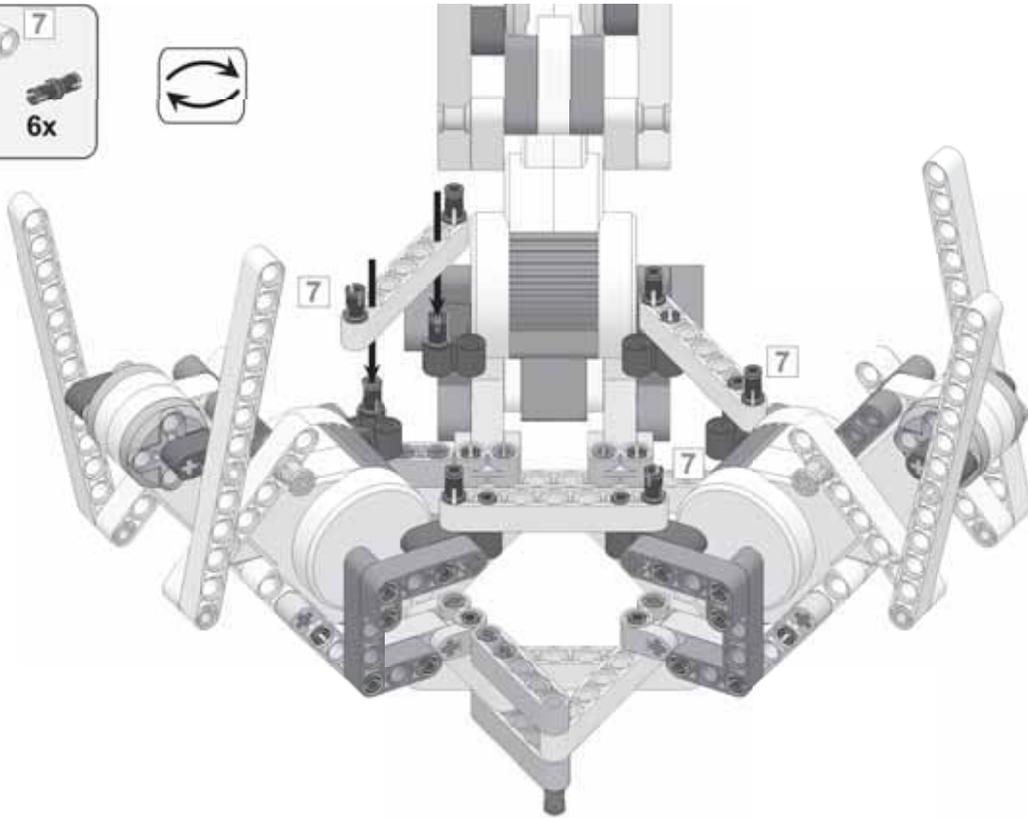


2





3



4

