Are you wondering how that little robot is going to open the tomb door? Good question. And you're going to answer that question by following along using a page out of Evan's design journal.

The Robot Description

Okay, now that you've named your robot, it's time to describe it. No, I'm not talking about "Short, grey and white, with wheels." What I mean is, what is this robot supposed to do? At this point, I hope you've read Chapter 1. If not, I'll wait . . . Go back and read it. Okay, have you finished it? Good. Now, what is this robot supposed to do? Don't say it, write it.

Look on your design journal page, and you'll see `Robot Description` and a large blank box. Don't be shy here. This is where you're going to try your hardest to describe accurately what this robot will do for you. Look back to Figure 1-2 in Chapter 1 if you need a reminder about the path the robot needs to follow. Let me show you what I wrote down, and you can compare it to your description, okay? Here goes (see Figure 2-2).

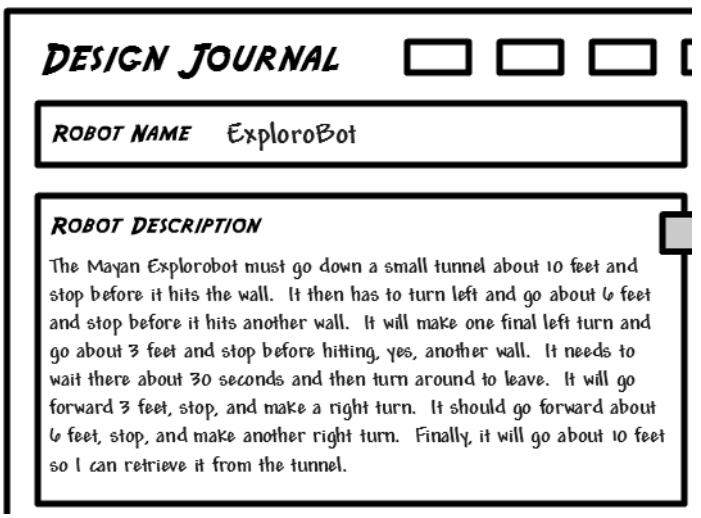

Figure 2-2. *Robot description*

If your description isn't exactly like mine, that's okay. What *is* important is that you got the major points: Robot moves forward about ten feet, stops, turns. Robot moves six feet, stops, turns. And on and on. Trust me—without an accurate description of the robot, it will be more difficult to build (Chapter 3) and program (Chapter 4). Don't worry if your description missed something; you'll get better at this, I promise. You're going to have more opportunities to write robot descriptions later in the book. By the time you're finished, you'll be an expert.

So, what's next, you ask? Okay, I'll tell you—you're going to take the description you wrote and break it down into small, single-item tasks.

The Task List

On your design journal page, locate the `Task List` section. This section is where you're going to list each individual task that the robot must perform. The *good news* is that if you wrote down a detailed description (see the previous section), then this section is almost already done.

What do I mean by "individual task?" An individual task is something like "Walk forward five feet" or "Turn doorknob." Something like "Press the button and turn the wheel" is not an individual task. Your goal is to list the actions your robot will perform, one at a time. Take a look at my task list (see Figure 2-3).

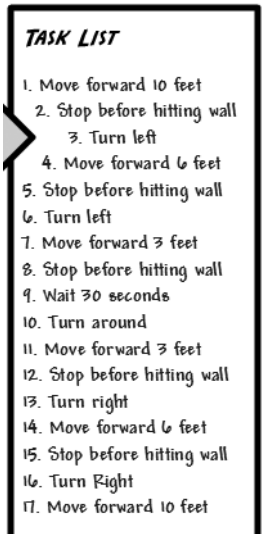

Figure 2-3. *The ExploroBot task list*

Compare your task list to mine. Were you able to break down the robot description into individual tasks? These individual tasks will help you in many ways, including assembling the correct form for your bot, picking the appropriate sensors to be used, and later when programming the bot.

I'll give you a small preview of how we'll use the task list later. Look at steps 2, 5, 8, 12, and 15—"Stop before hitting wall." Are you already thinking about how to do this? You've got options, of course. There's the Touch sensor that can be programmed to stop the robot when it's triggered. And what about the Ultrasonic sensor? The sensor sends out a signal that's detected when it bounces back off an object in front of it, such as our wall. So you can see that this task list will help you to start thinking about the NXT components you'll use. For now, let's leave the `Task List` and move on to the next section of the Design Journal.

Limitations and Constraints

You're going to encounter one obstacle quickly when you begin to design your robots using the Lego Mindstorms NXT kit. What is it? It's the number of parts in your kit.

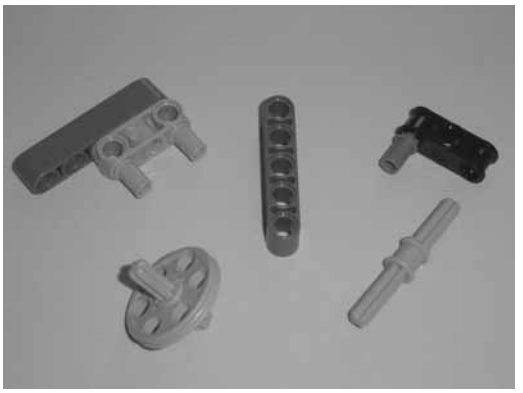

Figure 3-21. *It's not quite a rear-wheel yet, but keep going . . .*

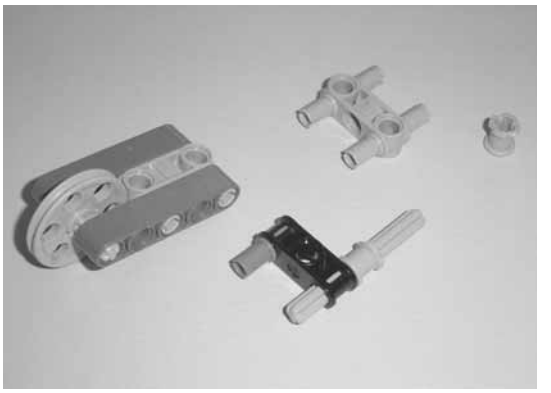

Figure 3-22. *The rear-wheel is starting to come together . . .*

■ **Note** In Figure 3-22 you'll notice that I've added two new parts that weren't seen in Figure 3-20. This will happen again in later figures, so be on the lookout for it. As I mentioned, if you set out the parts that you need for each section, when I add parts like this to a figure it will be very obvious to you—"Hey, that's not in my parts pile!"

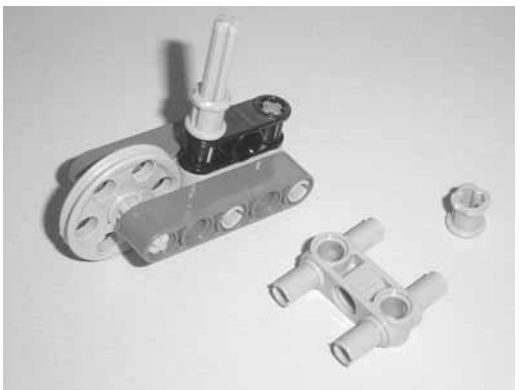

Figure 3-23. *The rear-wheel is almost done . . .*

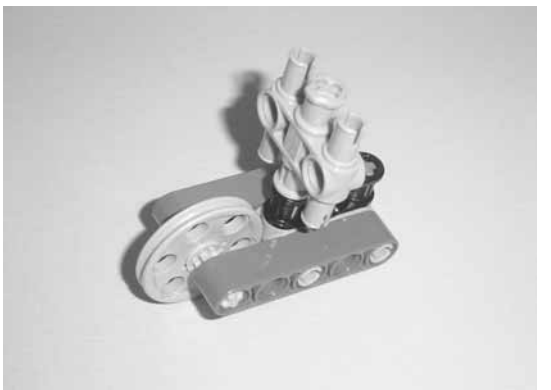

Figure 3-24. *The final rear-wheel assembly—set this aside for now*

Finally, Figure 3-25 shows the parts needed to start the rear-wheel base.

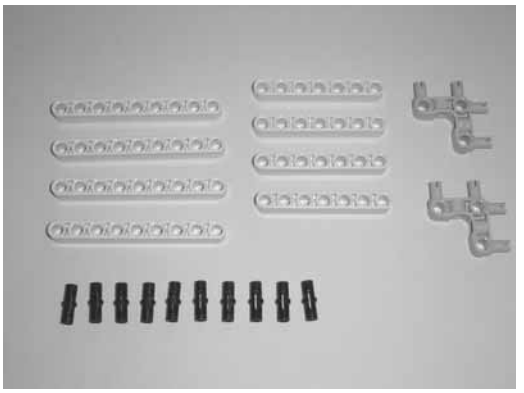

Figure 3-25. *Locate these parts to start the base that will hold the rear-wheel.*

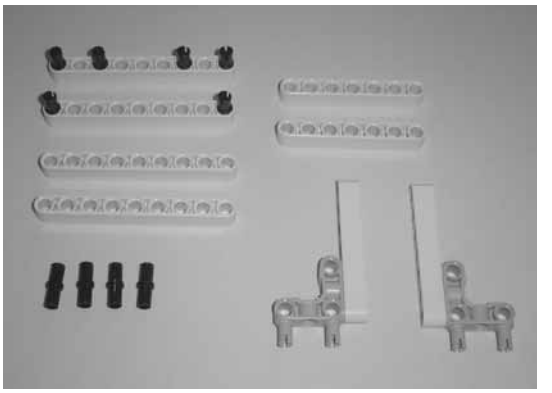

Figure 3-26. *Begin by connecting the parts as shown . . .*

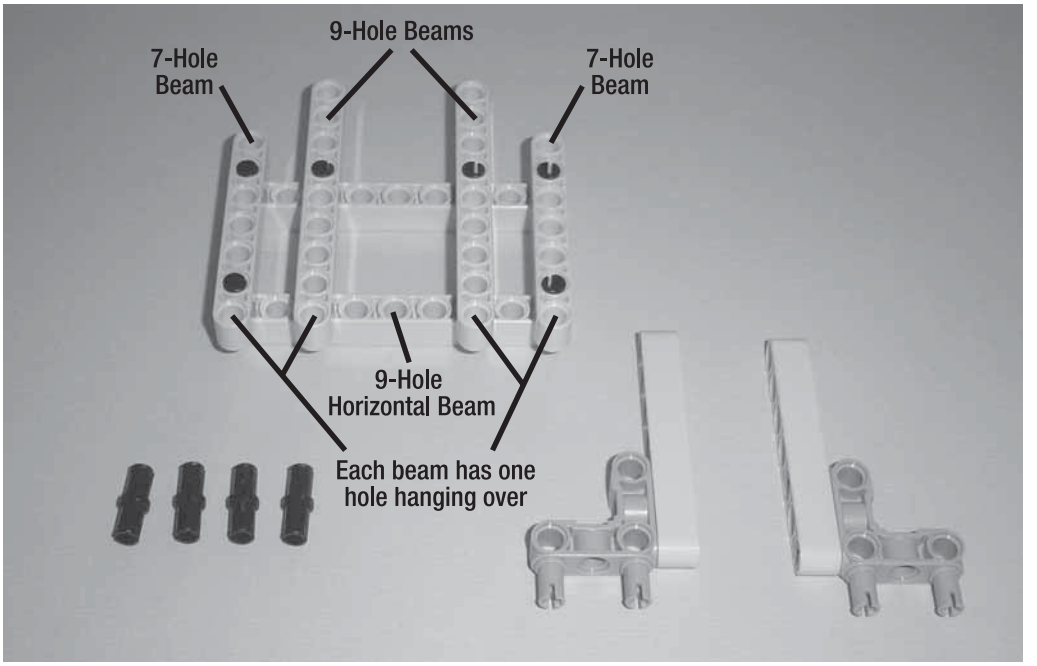

Figure 3-27. *Place the 7-hole and 9-hole beams as shown . . .*

■ **Note** In Figure 3-27, the two 9-hole beams and the two 7-hole beams are placed so that each beam has one hole "overhanging" the horizontal 9-hole beam (at the bottom). This is also a good place to mention that I'll occasionally use "callouts" in the figures. These "callouts" are simply text in the figure itself, sometimes with lines or arrows. Figure 3-27 uses callouts to indicate the 7-hole and 9-hole beams, and how they "overhang" the horizontal 9-hole beam.

Out of the Tunnel

Right now, the ExploroBot is facing west, ready to begin its exit from the tunnel. Just like the entrance, the bot will make some moves that are duplicates:

(Group 1) Forward – Detect Wall – Stop – Turn Right (second corner)

(Group 2) Forward – Detect Wall – Stop – Turn Right (first corner)

Turn Right (this final turn allows the robot to leave the tunnel)

So, once again, there's an opportunity to use two LOOP blocks, one nested inside the other. There are just a couple differences with these two LOOP blocks:

- The bot will be making right turns instead of left turns.
- The bot will make two right turns, so the outer LOOP block should only need its **Count** set to 2 instead of 3.

Knowing this information, let's place the two LOOP blocks and configure the outer LOOP for three repetitions and the inner LOOP for the Ultrasonic Sensor (see Figure 4-19). But didn't I say that it only needed to make two right turns? After that second right turn, you want the bot to keep rolling. It won't encounter another wall, but it should encounter your hands, waiting for it to come out of the tunnel. For that reason, you can configure this LOOP with a count of 3, even though it won't make an actual third turn.

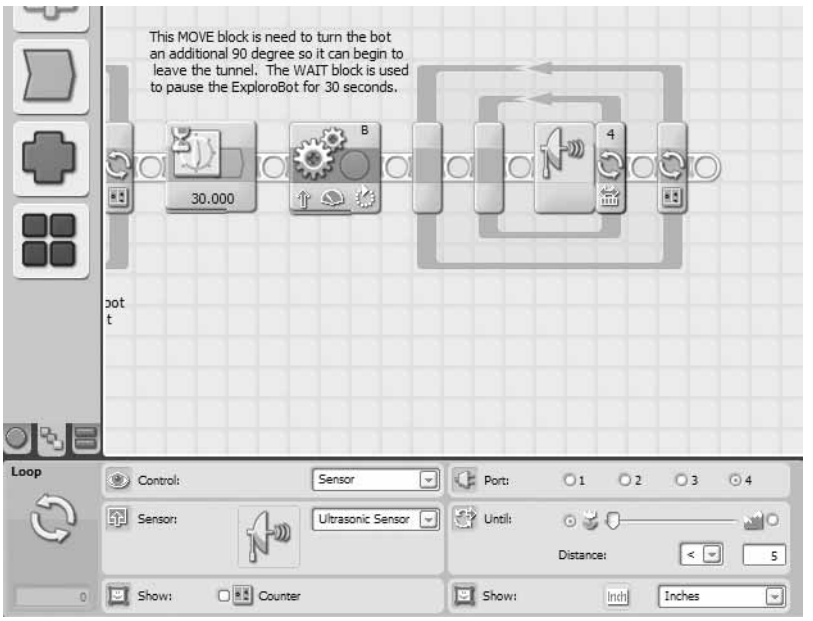

Figure 4-19. *Two LOOP blocks placed and configured*

Next, place three MOVE blocks: one for moving the bot, one for stopping the motors, and one for turning the bot. The first MOVE block is configured with **Unlimited** duration for motors

B and C. The second MOVE block is configured to stop motors B and C, and the third MOVE block is configured to turn *only* motor C at a power level of 25 and a duration of 360 degrees (see Figure 4-20).

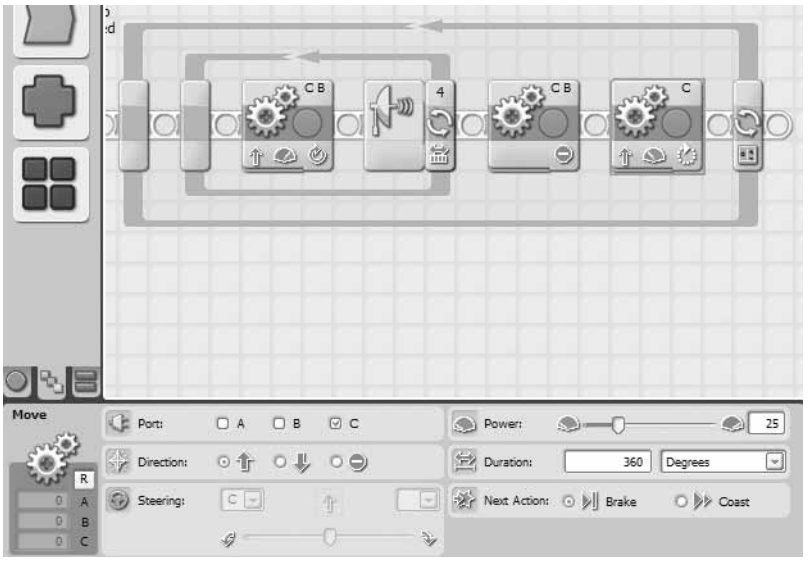

Figure 4-20. *Place three MOVE blocks here.*

Comments are placed describing these new LOOP and MOVE blocks (see Figure 4-21).

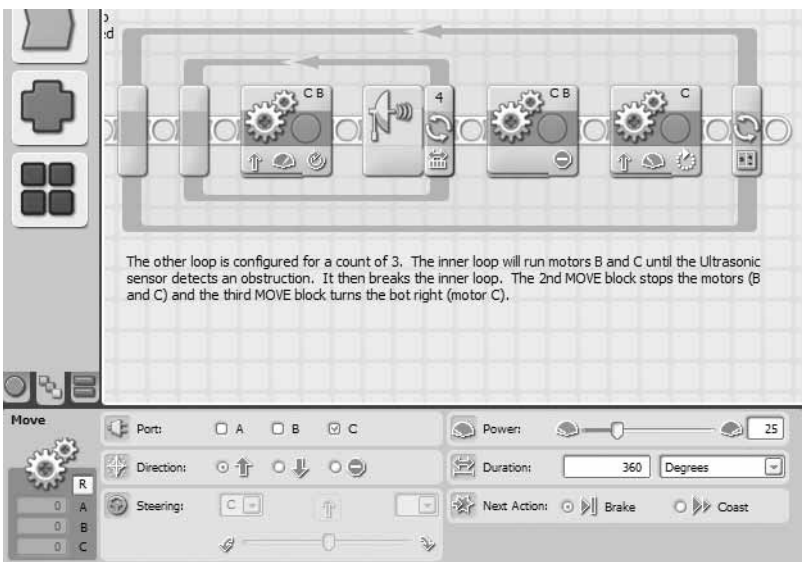

Figure 4-21. *Comments are added to describe these new blocks.*