



Pick a Card, Any Card

Have you ever had someone perform a card trick and ask you to pick a card from the deck? There are 52 cards in a standard deck (the two jokers don't count), and if it's been properly shuffled, you should be able to pick a random card from the deck.

It's the same with rolling a pair of dice. Each die has the potential to roll a number from one to six. When you roll a normal pair of dice, the numbers that appear are random. The odds of rolling a 2 on one die are the same as rolling a 6; they all have the same likelihood of appearing. What allows this to occur is simple randomness.

Your robots also have the ability to generate random numbers. You might want to build a robot that can roll a virtual set of dice or maybe pick a number between 1 and 1,000. Read on to learn how to have your bots generate random numbers *and* display them on the LCD screen.

The RANDOM Block

Your bots can use the RANDOM block to generate numbers in a range that you define. The RANDOM block is found on the Complete Palette in the Data fly-out menu. Take a look at the block and its configuration panel in Figure 14-1.

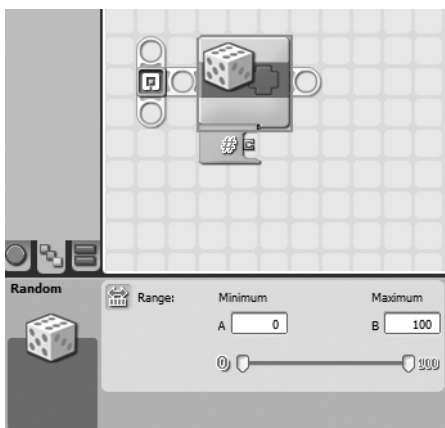


Figure 14-1. *The RANDOM block and its configuration panel*

The RANDOM block is another one of those blocks that is very simple to use. There are not a lot of options to configure, but what is there is important.

The first thing I want you to notice in the configuration panel is the Minimum and Maximum number fields in the Range section. In these boxes, you can type in the upper and lower values for the numbers you want the RANDOM block to generate. The Minimum value can never be less than 0, and the Maximum value can never be greater than 32,767.

There is one other method you can use for defining the Minimum and Maximum values. There are two small tabs on the Slider bar below the Minimum and Maximum values. You can drag the leftmost small tab to set the Minimum value. The rightmost small tab can be dragged to set the Maximum value.

Note One thing you should be aware of is that the Slider bar can only be used for defining a range between 0 and 100. If you wish to use a Maximum value greater than 100, you need to type the number into the Maximum number field.

Now, let's do an example with SPOT using pseudo-code:

Me: SPOT, show a random number on your LCD screen between 20 and 80 until I press the left button.

To do this, I'll first drop a RANDOM block on the beam (see Figure 14-1). I've told SPOT that I want the Minimum value to be no less than 20, so I'll use the Slider bar to set the Minimum value (see Figure 14-2).

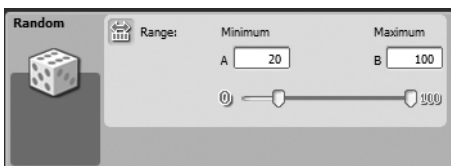


Figure 14-2. *The RANDOM block with the Minimum value set to 20*

Now, what if I had wanted the value to be between 20 and 80 but not include 20? Remember, the Minimum and Maximum values you define will be included in the possible numbers generated by the RANDOM block. So if I didn't want 20 to be a possible number, I would simply drag the Slider bar again to set the Minimum value to 21. By doing this, 21 would be a possibility, but 20 would be no longer allowed.

My final step is to set the Maximum value; I drag the Slider bar to set the Maximum value to 80 (see Figure 14-3).

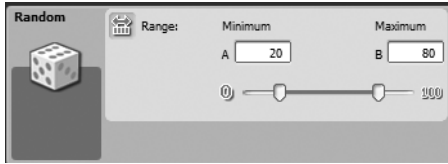


Figure 14-3. *The RANDOM block with the Maximum value set to 80*

Now all that's left to do is to have SPOT display the value on the LCD screen. To do this, I need to introduce you to another NXT-G programming block: NUMBER TO TEXT.

The NUMBER TO TEXT Block

The NUMBER TO TEXT block is also found on the Complete Palette in the Data fly-out menu. I'll drag and drop it after the RANDOM block, so you can see its configuration panel (shown in Figure 14-4).

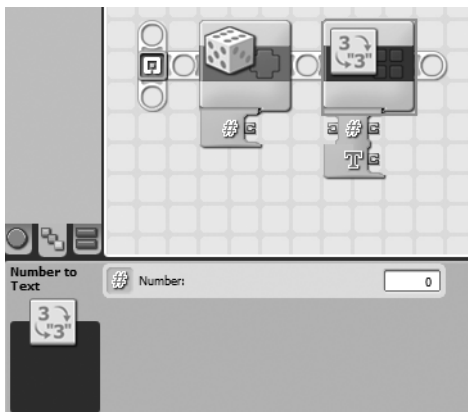


Figure 14-4. *The NUMBER TO TEXT block is for displaying numbers on the LCD screen.*

If you're wondering why the configuration panel is grayed out, it's because the NUMBER TO TEXT block requires the number it will display on the LCD screen to come into its single-input data plug. The number that is passed to this block is converted to text that can be displayed using the DISPLAY block.

So, let me go ahead and drag a wire from the output Number data plug on the RANDOM block into the input Number data plug on the NUMBER TO TEXT block. You can see this in Figure 14-5.

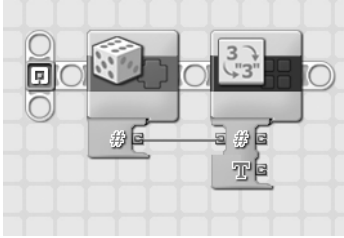


Figure 14-5. Drag a wire from the *RANDOM* block to the *NUMBER TO TEXT* block.

As you can tell from Figure 14-5, the text you want to display will come from the output data plug on the *NUMBER TO TEXT* block. I'll drop in a *DISPLAY* block and drag another wire from the output Text data plug on the *NUMBER TO TEXT* block into the input Text data plug on the *DISPLAY* block (see Figure 14-6). Remember to set the *DISPLAY* block to display Text, not Image or Drawing, in its configuration panel.

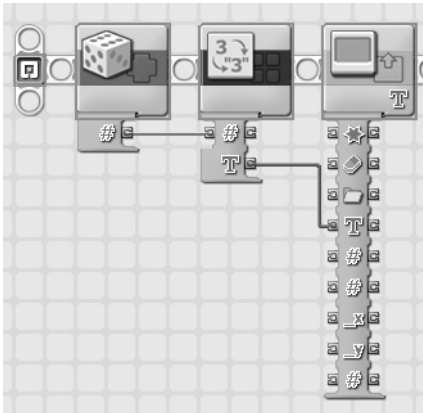


Figure 14-6. Drag a wire from the *NUMBER TO TEXT* block to the *DISPLAY* block.

Instead of using a *TIME WAIT* block to keep the text on screen as I've done with past examples, I'm going to use an *NXT BUTTON WAIT* block this time. The *RANDOM* number generated will stay on the LCD screen until I press the Left button.

To do this, I drop in an *NXT BUTTON WAIT* block and configure it as shown in Figure 14-7.

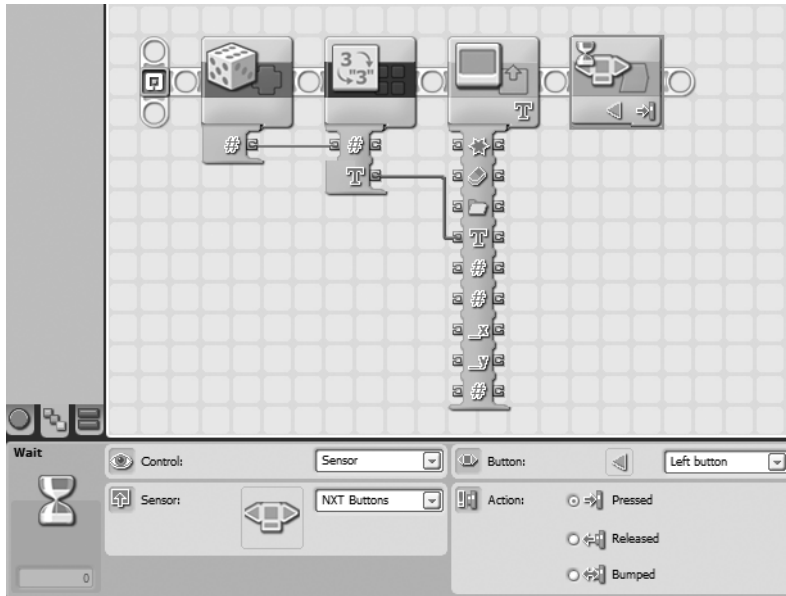


Figure 14-7. *The generated number will stay on the LCD screen until the left button is pressed.*

I've used a WAIT block with the Control section set to Sensor. In the Sensor section, I selected NXT Buttons from the drop-down menu. For the Button section, select Left button from the drop-down menu, and finally, in the Action section, I selected Pressed. By configuring the WAIT block this way, the random number will stay on the LCD screen until I press the left button on the NXT Brick. Then the program ends.

Not too difficult, huh?

One final thing I want to point out on the NUMBER TO TEXT block is the output Number data plug. This block will still allow you to keep and use the random number you generated in a Number format. You might need that random number later in the program. If so, you can drag a data wire out of the output Number data plug—it will still be a number and not changed to text.

So now you know how the RANDOM block is configured and used. If you want your robot to move randomly around the room, for example, you could configure a RANDOM block to generate a number between 1 and 4. Program your bot to go left if the number is 1, right if the number is 2, forward if the number is 3, and in reverse if the number is 4. By using a random number to control the bot's direction, you can give the bot some unpredictable behavior. Refer to Chapter 12's discussion of the SWITCH block for using conditions such as a random number to control movement.

Next, in Chapter 15, we'll look at the COMPARE block.