

## Technical Learning

Divide your team into groups of builders, programmers, and possibly testers. The builders will build 2 of the Constructopedia robots and the programmers will write a program to drive the robots in a square. If you have testers, they will be responsible for running the programs and making observations about the performance of the robots. However, everyone should get a chance to observe the 2 robots move around while executing the program. They can then see how the exact same program will run differently depending on the engineering design and the environment. (If you only have one RCX programmable brick, then one robot will have to be built to completion, tested, then taken apart slightly to use the RCX brick for the other robot.)

You'll have to figure out how to divide up the groups appropriately. You probably want everyone to get an opportunity to try programming and building. However, the building (especially if a group is only to make 1 robot) will probably go much faster than the programming. Therefore, you might have to come up with other activities for the builders to work on while the programmers are finishing up. If you have a testing group, then they will have to wait until both groups are finished in order to start their task.

Coaches in the past have found that assigning specific roles to each group member works best. For the builders, one can find the pieces and the other can build the robot. In programming, one can read the directions and one can operate the computer. For the testers, one can operate the robot while the other writes down observations. The kids should be trading roles so that they can have as much exposure to each aspect of building and programming as possible. For the building, roles can be swapped when building the second robot. For programming, roles can be swapped after the program has been saved for the first time. For testing, roles can be swapped between each of the two robots.

Here are some questions that you can ask the kids as the robots are being tested:

- Did the robot go in a perfect square? (It most likely won't.)
- Why didn't the robot go in a perfect square?  
A: The turns were not perfect 90 degree turns and/or the robot did not travel straight.
- What are the two things that you can change in the program to change the degrees of the turn?  
A: You can change the motor speed or the amount of time that the robot turns for.
- What could you change in the robot design to change the degrees of the turn?  
A: wheel size.
- How did the performance of the same robot differ when on different surfaces?
- How did the performance of the two robots differ when on the same surface?
- Why do you think there was a difference?

Follow the directions in the Constructopedia to build Robo 1.

Follow the directions in the Constructopedia to build Pathfinder 1.

If you only have one RCX programmable brick, you can build one robot completely, test it, then remove the brick and add it to the other robot for testing.

## GETTING STARTED – First-Time Users

- ① Start up the Robotics Invention System.
- ② Select **New User**.
- ③ Type your name and then Enter.

*After the program stops describing Guided Mode*

- ④ Cntrl-Click the porthole labeled **About**

*Cntrl-Click* means press the Cntrl key while pressing the left mouse button when the cursor is over the described area.

**You are now free to select from the main menu options.**

## Do you need to install *firmware*?

---

Firmware is a special program that must be installed onto your RCX so that other programs can communicate with the RCX. Firmware is installed if your RCX display

shows  .

If firmware is NOT installed, follow the installation directions below.

## To Install Firmware

---

- ① Place your RCX about 6 inches in front of the transmitter so that the *infrared* window is facing the transmitter. Infrared is a type of light that you can't see. Your remote control uses infrared to communicate with your TV.
- ② Select **Getting Started**.
- ③ Select **Set Up Options**.
- ④ Press the button to the right of **Download RCX Firmware**.
- ⑤ After downloading of the firmware is complete, press **Continue**.
- ⑥ After downloading of the built-in programs is complete, select **Main Menu** in the upper right corner.

---

**Once firmware is installed, you are ready to start programming!**

---

## To Get Into Programming Mode

---

- ① Select **Program RCX** from the Main Menu
- ② Select **RCX Code** from the Programming Menu

**You should now see the main programming screen as shown in Figure 1.**

---

Take a moment in your group to review each of the icons as described in Figure 1. Feel free to click the menu blocks in the upper left-hand corner so that you can see the different commands that you have access to. You can scroll through the commands by using the Up and Down Triangles located at the top and bottom of each menu. Try the Help Menu too.

---

## Designing the program

---

Together, we are going to create a program to move your robot in a square.

To move the robot in a square, it must

1. move forward some distance
2. turn 90 degrees in one direction
3. move forward the same distance as before
4. turn 90 degrees in the same direction as before
5. move forward the same distance as before
6. turn 90 degrees in the same direction as before
7. move forward the same distance as before
8. turn 90 degrees in the same direction as before

Together, these steps are called an *algorithm*. An algorithm is a step-by-step plan of what the program should do, but it does not indicate how exactly to perform those steps. It is important that you come up with an algorithm *before* you start creating your program on the computer.

Notice that we are repeating the same two steps (move forward, turn) 4 times. Therefore, let's first create a program that moves forward then turns.

## Create a program to move forward

---

- ① Click on the green **Commands** block.
- ② Click on the lower triangle at the bottom of the Commands menu until you see **set direction**.
- ③ Click on the **set direction** tile to pick it up.
- ④ Drag this tile so that it is directly under the **program untitled** tile (fitting it like a puzzle piece).
- ⑤ Click again to lock it in place.

*Your program should now look like the one in Figure 2A.*

- ⑥ *Right-click* the **set direction** tile to turn it over. Right-click means to click the right mouse button while the cursor is over the described area.
- ⑦ Click the box next to **B** to turn that motor off (your robot won't have a motor connected to port B). Note that the selected direction arrows under A and C are pointing in the same direction.

*The screen should look like the one in Figure 2B.*

- ⑧ Click the **green checkmark** to turn the tile back over.

---

You have just programmed the motors connected to ports A and C on the robot to turn in the same direction.

---

- ⑨ Find the **set power** tile and place it below set direction.
- ⑩ Right-click **set power**.
- ⑪ Click in the box next to **B** to turn it off.

*Your screen should look like the one in Figure 2C.*

- ⑫ Click the **green checkmark** to turn the tile back over.

---

You have just programmed the power level to 4 for those motors connected to ports A and C. The power level indicates how quickly the motors will turn, thus how quickly your robot will move. You can change the power level by clicking the + or – buttons next to the power level.

---

- ⑬ Find the **on for** tile and place it below **set power**.
- ⑭ Right-click **on for**.
- ⑮ Click the **minus button** (-) until the time indicator reads 5.
- ⑯ Click the **green checkmark** to turn the tile back over.

---

You have just programmed the motors to turn on for 0.5 seconds.

---

☺ ***Congratulations!!*** ☺

*You have now programmed both motors to move forward at power level 4 for 0.5 seconds.*

## Add a turn to your program

---

- ① Add **set direction** to your program.
- ② Turn the tile over.
- ③ Turn **B** off.
- ④ Reverse the direction of the C motor by checking the **bottom circle** below **C**.
- ⑤ Turn the tile back over.
  
- ⑥ Add **set direction** to your program.
- ⑦ Turn the tile over.
- ⑧ Turn **B** off.
- ⑨ Set the power level to 1.
- ⑩ Turn the tile back over.
  
- ⑪ Add **on for** to your program.
- ⑫ Turn the tile over.
- ⑬ Set the time to 2.
- ⑭ Turn the tile back over.

*Your program should now look like the one shown in Figure 3.*

---

You have now programmed the motors to move in opposite directions, at power level 1, for .2 seconds. Moving motors in opposite directions causes the robot to turn. The power level is decreased for better control during turning.

---

😊 ***Congratulations!!*** 😊

*You have written a program to move your robot forward then turn.*

## Saving Your Program for the First Time

---

Take a moment in your group to look over Figure 4, which shows the program tile turned over. Look at the various options as described in the figure.

---

- ① Right-click the **Program Untitled** tile.
- ② Click **Save As**.
- ③ Click in the box with the word 'Untitled'.
- ④ When the box is blank, type in 'square'.
- ⑤ Click the **green checkmark** to turn the tile over again.

*You program is saved.*

- ⑥ Click the **Main Menu** button in the upper right corner to close the programming environment.

## Closing and Opening a Saved Program

---

- ① Get back into the programming environment (if you forgot how, look at the beginning of this document).
- ② Click the **Open** button in the lower left corner to see all programs.
- ③ Scroll through the programs and find the one named 'square'.
- ④ Click the number next to the program to open it.

Recall that our algorithm required that we go straight and turn 4 times. Now we are going to add the *program control structure* called a *repeat block* to our program. A program control structure (those commands found in the Stack Controllers menu) is something that directs the flow of your program. In other words, it determines which of the tiles to perform and when. A repeat block is a group of tiles that are repeatedly performed for as many times as you indicate.

## Adding the Control Structure Repeat

---

- ① To pick it up, click the first **set direction** tile that is directly underneath the **program square** tile.
- ② Drag it to any location on the screen and click again to drop it.
- ③ Click the **stack controller** block to see the control structure menu.
- ④ Find the **Repeat** tile and place it under the **Program square** tile.
- ⑤ Click the **plus button** (+) until 4 appears.

*Your screen should now look like the one in Figure 5.*

- ⑥ Click the first **set direction** tile that you just placed to the side.
- ⑦ Drag it directly underneath the **Repeat** block (not the **End Repeat** block) and click again to drop it. Note that the **End Repeat** tile automatically moves to the bottom.

*Your screen should now look like the one in Figure 6.  
Notice that you cannot see all of the program at once.*

- ⑧ Move your cursor to the bottom of the screen and then back to the top. Notice how the program tiles scroll up and down so you can see all of them.
- ⑨ Click the gray square in the upper left corner of the red **Repeat** tile. Note that all the code shrinks into one single tile. This makes it easier to see more of your code as you keep adding tiles.
- ⑩ Click the gray square to see all the tiles again.

## **Saving Your Program after Changing It**

---

- ① Right-click the **Program square** tile.
- ② Click **Save**.
- ③ Click the **green checkmark** to turn the tile over again.

*You have now saved the changes to your program.*

Your program is ready to be *downloaded* to the robot. Downloading is the process of transferring something from one computer to another.

## **Download Your Program**

---

- ① Place your RCX about 6 inches in front of the transmitter so that the infrared window is facing the transmitter.
- ② Right-click the **Program square** tile.
- ③ Click the **download** button.
- ④ When the RCX beeps, downloading is complete.
- ⑤ Click the **green checkmark** to turn the tile over again.

☺ *Congratulations!!* ☺

*You have just successfully written, saved and downloaded a program.*

# RCX Programming Environment



Figure 1

**Program Appearance  
After Adding  
Set Direction  
Command**



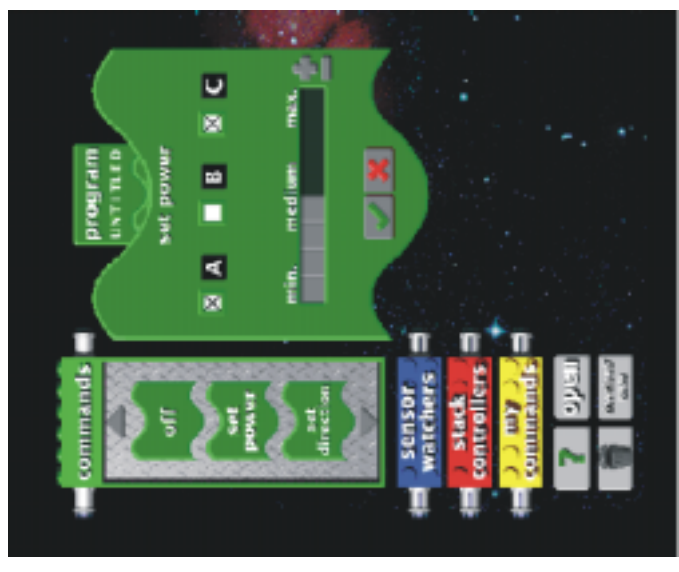
*Figure 2A*

**Program View  
After Turning Over  
Set Direction  
Command Tile**



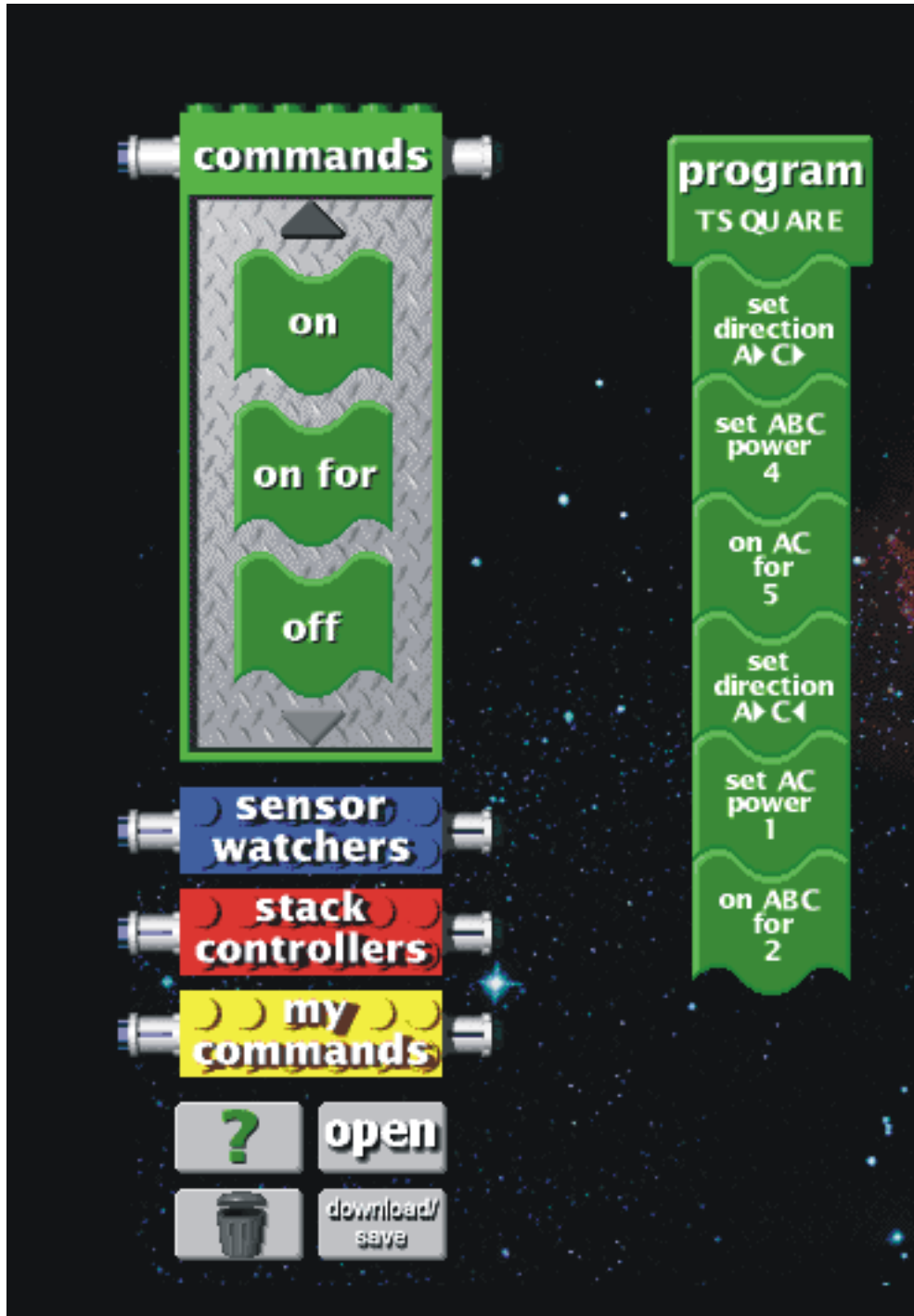
*Figure 2B*

**Program View  
After Turning Over  
Set Power  
Command Tile**



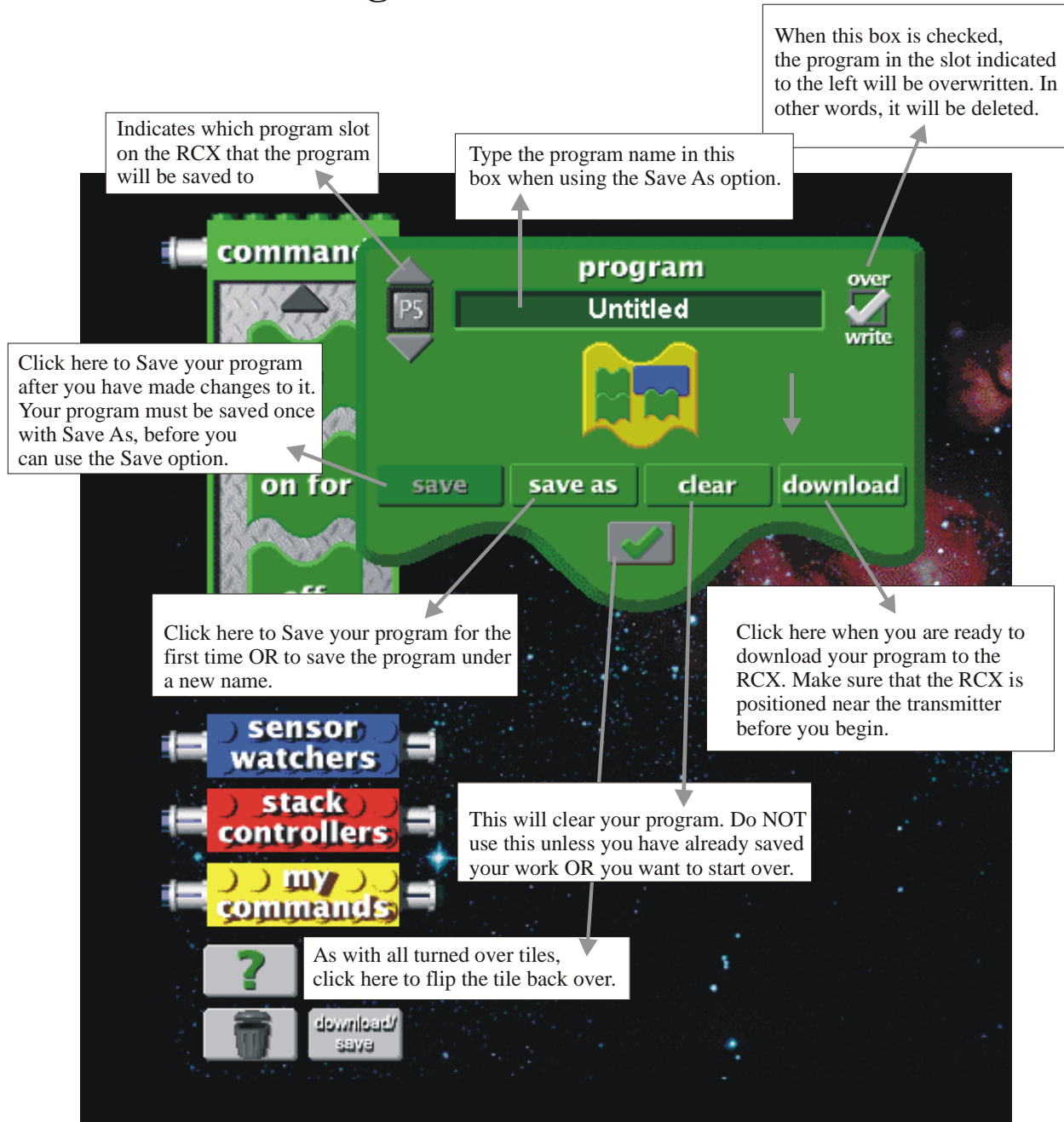
*Figure 2C*

# Program to Move Robot Forward Then Turn



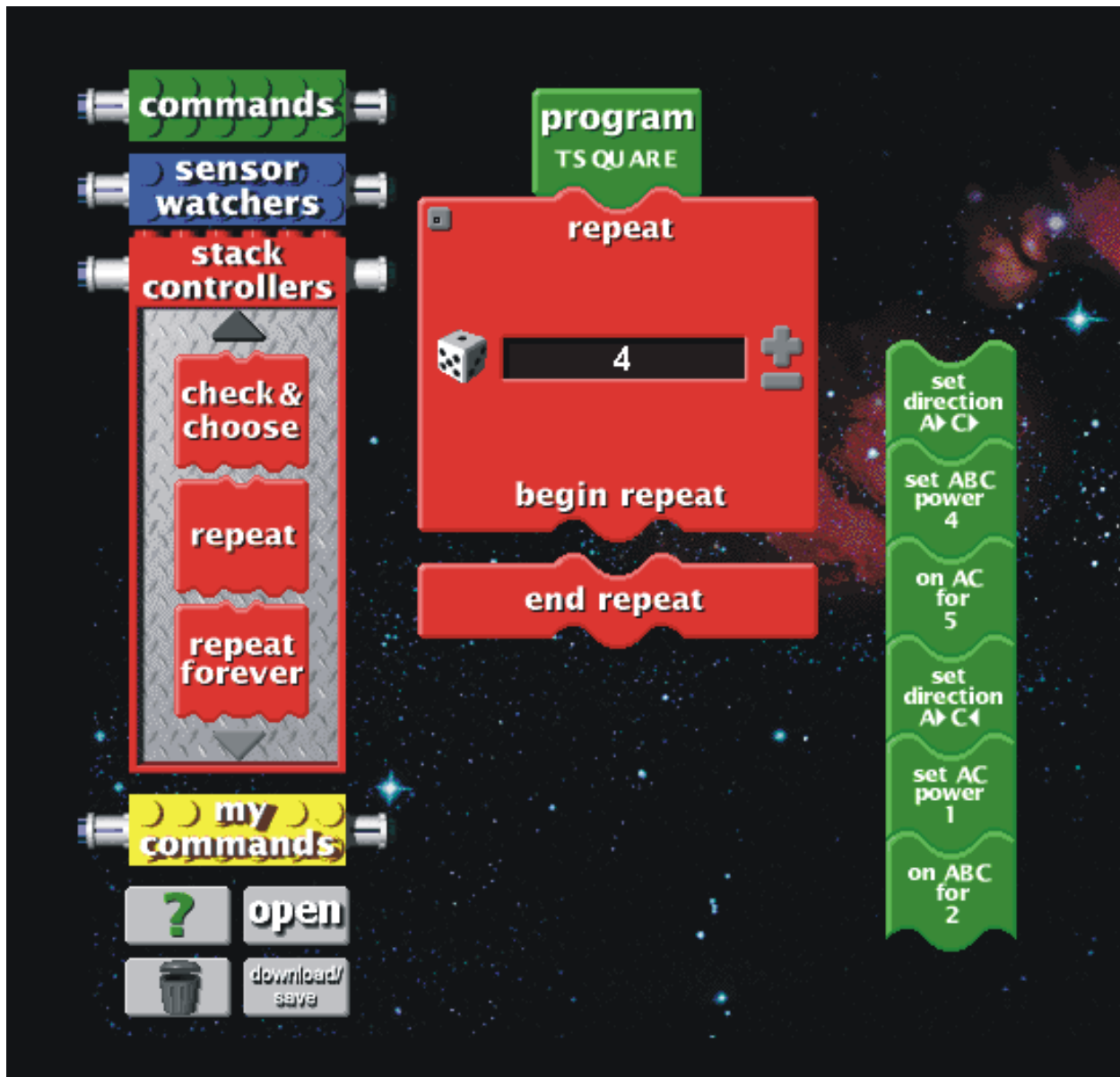
*Figure 3*

## Turned Over Program Tile



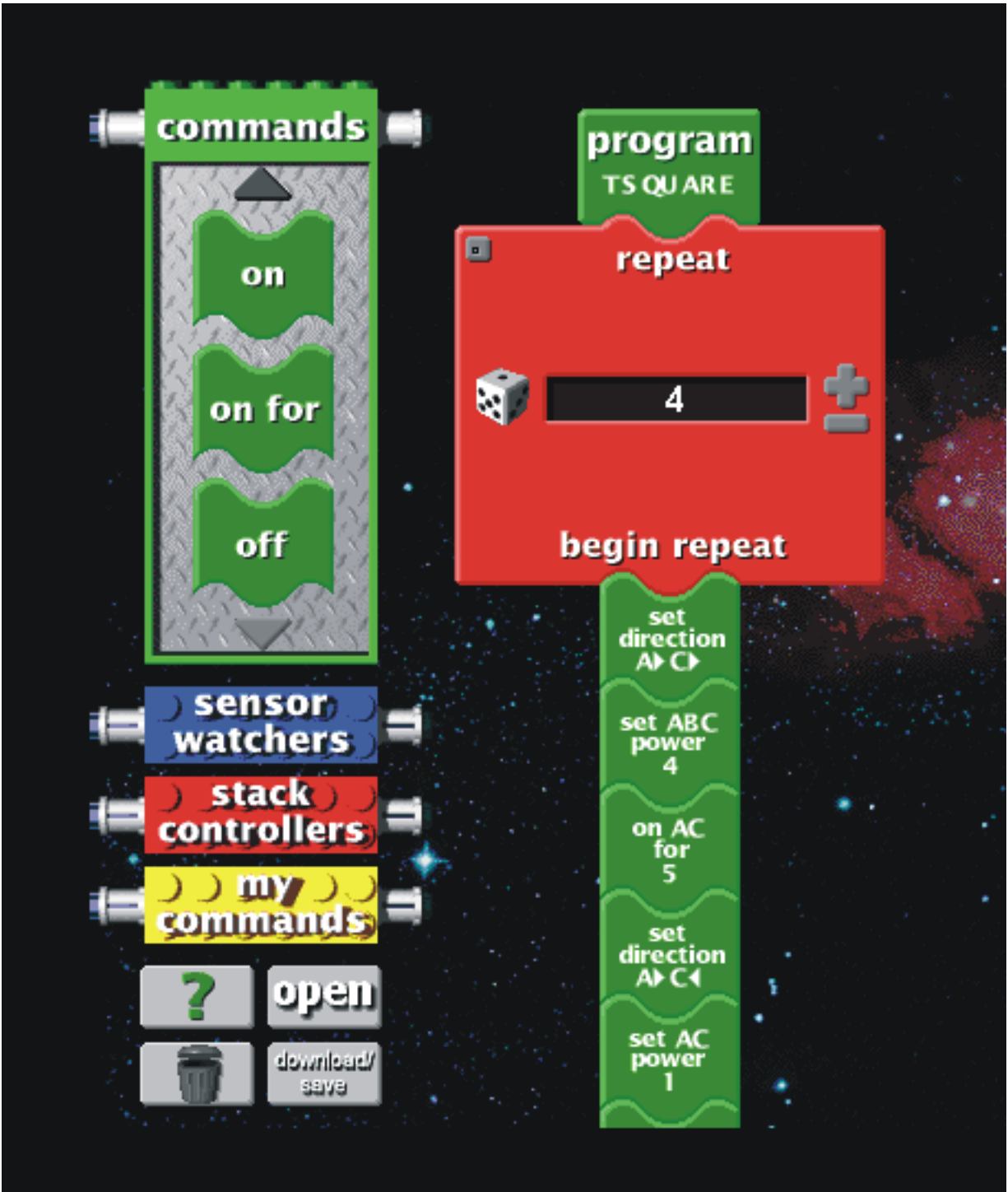
*Figure 4*

# Adding a Repeat Block



*Figure 5*

# Completed Program To Move In a Square



*Figure 6*