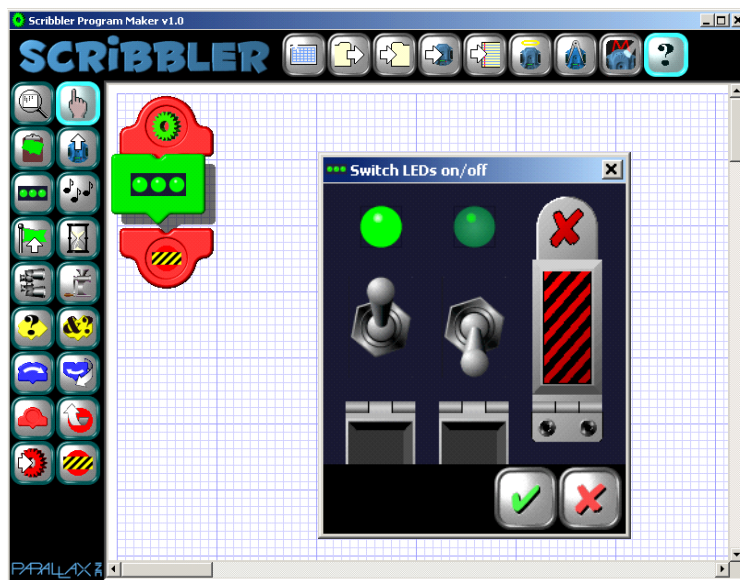Scribbler GUI Programming Guide

First Programs

Inside the Scribbler Robot, there is a tiny computer called a **BASIC Stamp® microcontroller**. It performs a list of instructions that makes the Scribbler do things. This list is called a **program**. You can change what your Scribbler does by putting a new program in its BASIC Stamp "brain."

This guide will show you how to make your first simple programs with the **Scribbler Program Maker** GUI software. "GUI" is short for **Graphical User Interface**. This is just a way of saying that you will build your programs out of pictures instead of words.

By following this guide, you can learn to make your Scribbler Robot blink its lights, make sounds, and drive in a circle. You will make your Scribbler watch for objects, and respond if it sees one.

To program your Scribbler Robot, you will need to have your Scribbler Program Maker software installed and running on your personal computer. Also, you will need to have your Scribbler Robot connected to your computer with the serial cable. If you need instructions to do these things, follow the Setting Up guide before you begin here.

Building Programs with Blinking Lights

When you first open the Scribbler Program Maker, you will see a new **worksheet** like this one. This is where you will build programs with tiles. There are already two red tiles on the worksheet.

The top tile with the green gear is the **start tile**, and the bottom tile with the striped barricade is the **end tile**. To make the Scribbler Robot do something, you will add more tiles between the start and end tiles.

You will choose tiles to add to your program with these buttons on the left side of the worksheet.

To see a hint about what each button does, hold your cursor over the button. These hints are called **tool-tips** or **flyover help**.
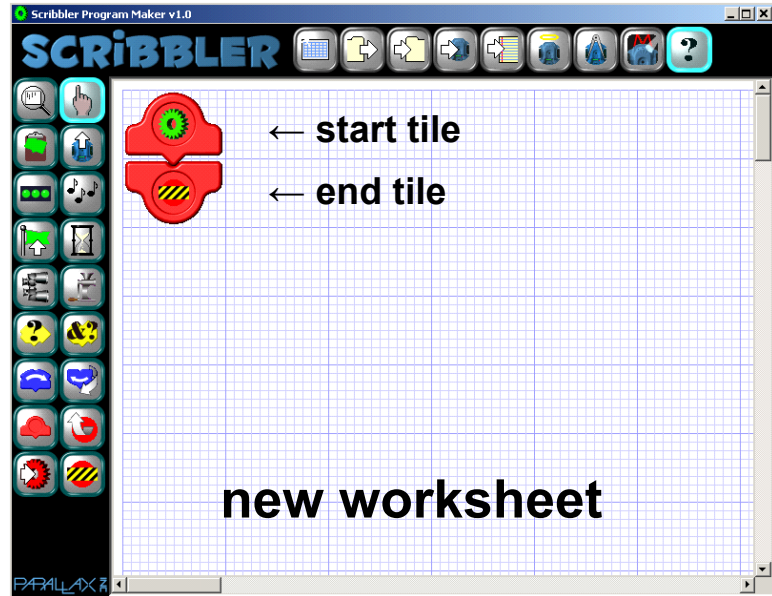
The Help File

The question mark button "**?**" at the top right opens up the **Help** file. It explains how each part of the Scribbler Program Maker works, and what every button does.
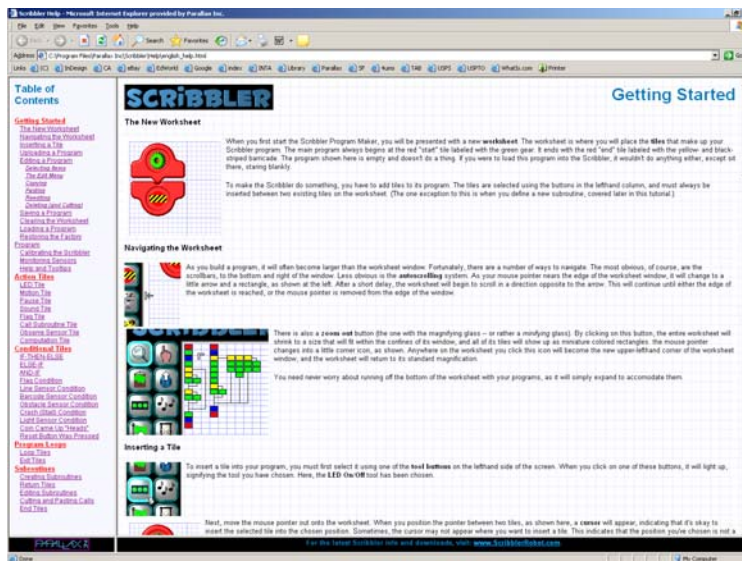
The Help file is very useful, but it is a lot of information to read all at once. If you read the whole Help file, you will discover that the Scribbler Program Maker is very powerful.

To open the Help file, click .

To close the Help file, click .

But you don't have to learn everything at once! Let's build some simple programs step by step.

Placing Tiles in your Program

To place a tile in your program, follow the steps below. For our first program, let's use the tile that controls the three lights on the back of the Scribbler. These are called **indicator LEDs.** (LED is short for **L**ight **E**mitting **D**iode.)

√ Click on the LED button.

√ Place your mouse pointer between the start tile and end tile, until a **cursor** appears. It looks like a shaded bar.

√ Click your mouse button again
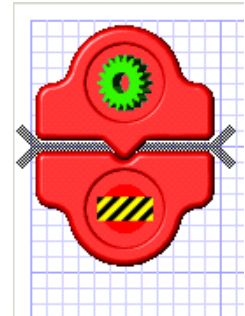
Action Tiles and Control Boxes

A green tile with the three LEDs on it will appear in your program. All green tiles are **action tiles** – they make the Scribbler perform an action when you run your program. Also, a **control box** opens up. This control box contains a switch for each LED, with all the switches turned on. All action tiles have control boxes where you choose the settings for that action.
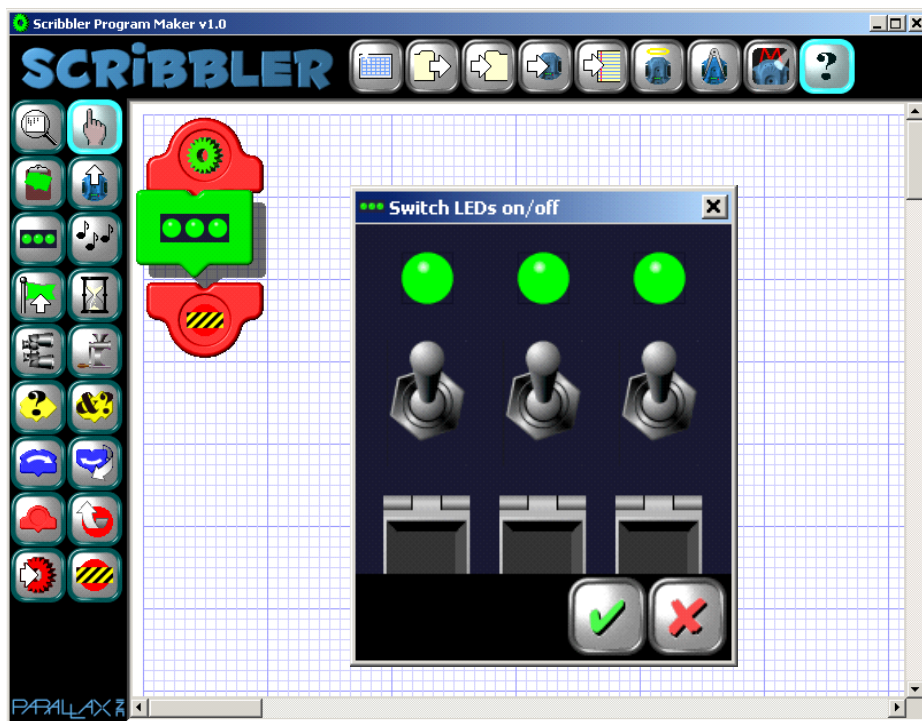
√ Click the **yes** button to accept the switch settings and close the control box.

A green checkmark always means **yes**, **okay**, or **true**.

4 · First Programs

Saving a Program

It is a good programming habit to save your program worksheet before loading the program into your Scribbler Robot.   The button for saving a worksheet is on the top button bar.

√   Click on the **Save a worksheet** button.

The **Save Worksheet as...** window will open. Your computer may automatically save programs to your **My Documents** folder. If you want to save your worksheets in a different folder, you need to browse there first.
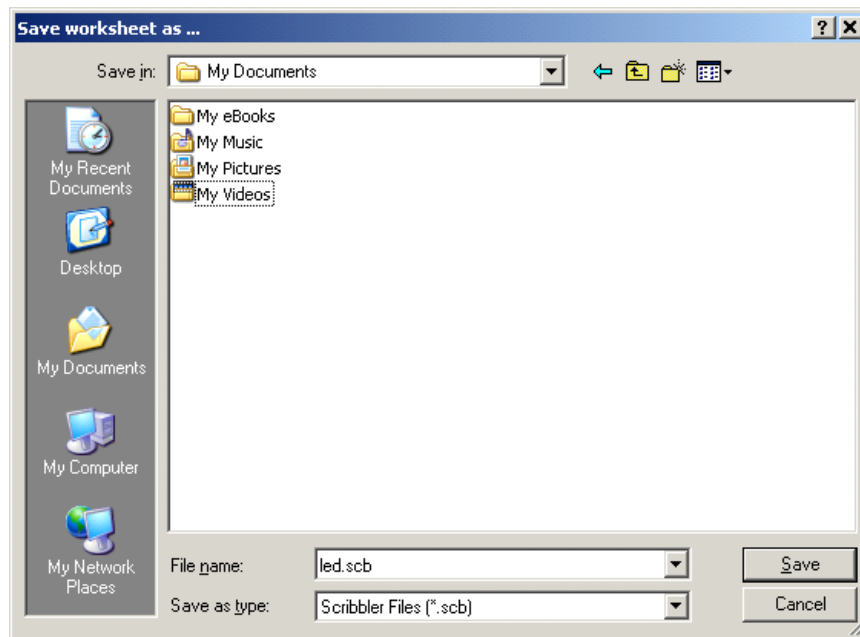
√   In the **File name** box, type in a name for your worksheet, like "led"
√   Leave the **Save as type** box set to Scribbler Files (*.scb).
√   Click on the **Save** button.

Your program worksheet is saved inside your computer. Once a worksheet has been saved, you can open it again later by clicking on the **Load a worksheet** button.

Running a Program

Now let's send a copy of your worksheet program to the BASIC Stamp microcontroller inside your Scribbler Robot. This is called **uploading** the program. First, let's make sure the Scribbler is ready to receive the program.

Programming Connection Checklist

√ Make sure your Scribbler Robot is connected to your computer with the serial cable.

√ Make sure the Scribbler power switch is in the "on" position.
   1 = ON
   0 = OFF

√ Make sure the red power light is on. If it is not, your Scribbler may need new batteries.
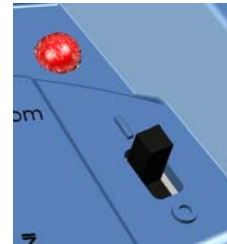
If you need instructions to connect your Scribbler Robot to your computer, follow the directions in the Setting Up guide before you continue.

Uploading the Program

√ Click on the **upload** button.

While the worksheet program is being copied to the BASIC Stamp microcontroller brain inside the Scribbler Robot, the **Uploading** message will appear.

If the Scribbler Robot receives the program, a window opens with the message:

"Upload successful!"

If you see this message, your Scribbler Robot is running your program. You can see that your program is running, because the three green LEDs will be turned on.
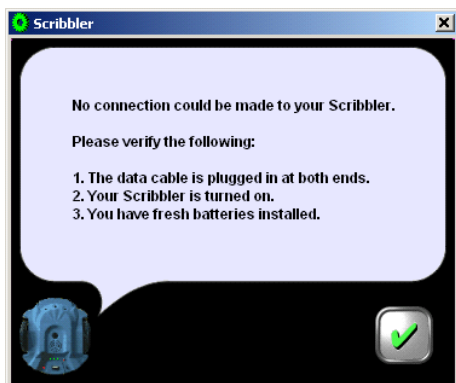
You might get this message instead:

"No connection could be made to your Scribbler."

If this happens, you need to go back over the Program Connection Checklist again. If you need instructions, see the Setting Up guide.

While this program is running, you may see your green LEDs flicker every few seconds. This is normal. The BASIC Stamp went to sleep at the end of the program. It wakes up every 2.3 seconds, and checks to see if everything is all right before going back to sleep. When it does this, the power gets interrupted for a moment, so you see the LEDs flicker.

## 6 · First Programs

Editing Programs

Our very short program did just one thing: turned on the LEDs. Let's make it more fun by adding more actions to the program. You can copy tiles, add new tiles, and change the settings in any action tile's control box.

Tile Edit Tools

Let's see how to edit a tile that is already in our program.

√ Click on the select tool.

√ Click on the LED tile in your program to select it. When a tile is selected, it will look like it is floating above the worksheet.

√ Right-click your mouse.
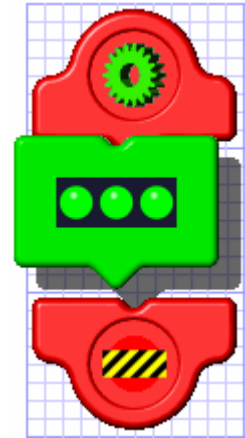
The LED tile's **edit menu** will appear.

The first button has a small picture matching the tile. This opens the tile's **control box**.

The **copy** button makes a copy of the tile so you can paste it someplace else in your program. A copy will have the same control box settings as the original.

The **cut** button lets you cut a tile out of your program, so can paste it someplace else.

The **delete** button is used to throw away a tile you no longer want in your program.

If you click on the copy, cut or delete button to select it, the button will light up around the edge. Next, you must click the **yes** button to make it happen, or you can click the **no** button if you change your mind. The edit menu closes after you choose yes or no.

Let's try copying the tile.

√ Click on the copy button.

√ Click on the yes button to approve it.

The edit menu will close, and the action tile will no longer be selected. But see that the **clipboard** button is selected. This lets you know that something is on the clipboard, and ready to be pasted in the program.
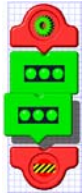
√ Place your mouse pointer below the LED tile so the cursor appears.
√ Click on the cursor.

The copied LED tile will appear in your program. You could keep pasting more and more LED tiles into your program, but for now let's keep it at two.

Changing Control Box settings

Let's change the control box settings in the bottom LED tile.

√ Click the select tool.

√ Click on the bottom LED tile to select it.

√ Right-click your mouse.

√ In the edit menu, click on the control box button.
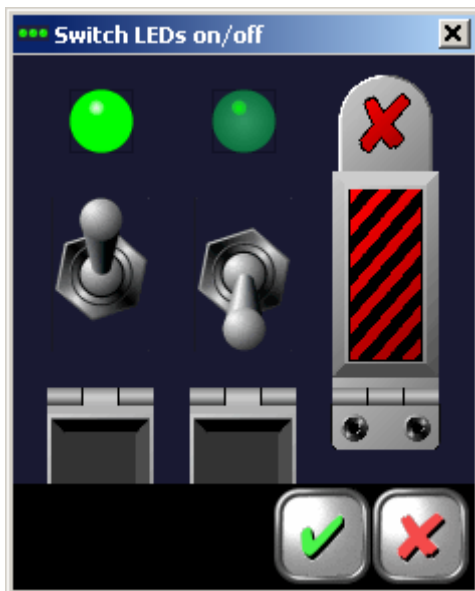
The LED control box will open.

Each LED has its own switch. Click on the switch to change its position. Each switch has three positions:
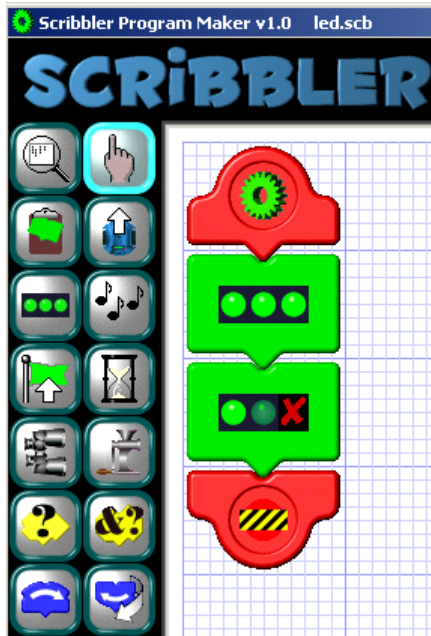
1. When the switch is up, the LED is on.
2. When the switch is down, the LED is off.
3. When the switch is covered, the LED will not change from how it was set earlier in the program.

√ Set your LED switches so they match this picture.

√ Click on the yes button.

8 · First Programs

Adding a Pause

Your program should now look like this. Notice that the LED tiles show how you have set your switches inside the control box.
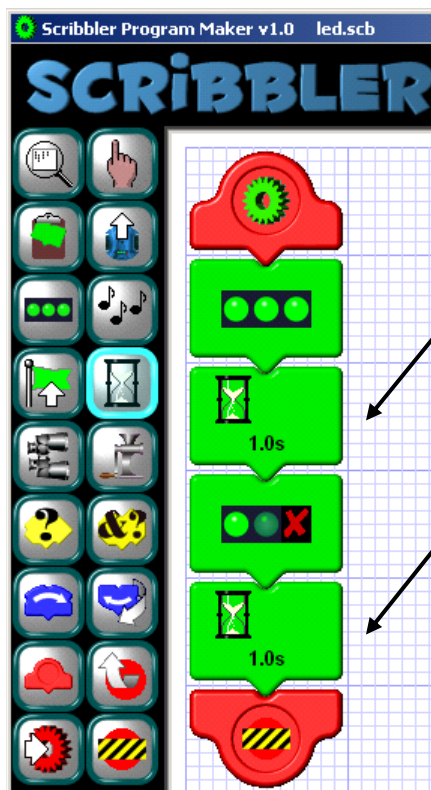
√ Run the new program.

√ Push the **reset** button on the Scribbler a few times and watch the middle LED.

The reset button makes your program run again. But you can not see the center LED turn on. Why not? The center LED turns on and then off again so fast our eyes cannot see it. Computers can work faster than our eyes.

Let's add some new tiles to slow down our program. The **pause** tile looks like an hour glass. It causes a program to pause on that tile, and wait a little while before starting again. You can set the wait time in the control box with the slider bar on the side. If you don't change it, the wait is one second.

√ Click on the pause button to select it.

√ Click between the LED tiles to add a pause tile.

√ Click the yes button in the control box.

√ Click below the bottom LED tile to add another pause tile.

√ Click the yes button in the control box.

√ Run the new program.

√ Press the Scribbler's reset button a few times and watch the middle LED.

Now you can see the middle LED turn on for 1 second each time you press the reset button. The pause tiles make the program slow enough for our eyes to see.

Repeating Actions with Loops

With this program, we can make the LEDs blink over and over again by pushing the reset button. But what if we don't want to keep pushing the button? To make actions repeat over and over again, we can use a programming trick called a **loop**. Let's make a new program that uses a loop.

Opening a New Worksheet

Let's begin with a new worksheet.

√ Click on the new worksheet button on the top button bar.

A window will open, asking you if you want to save your program.

√ If you want to save it, click yes.

√ If you don't want to save your program, click no.

If you wanted to just close the window without leaving your program, you could click **cancel.**

Putting a Loop in your Program

√ Click on the **insert a loop** button.

√ Click between the start and end tiles to put the loop in your program.

See that this placed two new tiles in your program, not just one. The top blue tile is the beginning of the loop, and the bottom blue tile is the end. Now, any action tiles you put in between will happen over and over again when you run your program.

10 · First Programs

The Loop Counter

How many times will your loop repeat? When you put your loop in your program, the **loop counter** box opened up. You can make your loop run forever (until you turn off the Scribbler.) This is called an **infinite loop.** Or, you can set the counter to repeat a certain number of times, then stop, from 1 time up to 255 times.

To set the counter, click on the little white buttons above or below each black window. Click above to make the number in it larger, or below it to make the number smaller. If you enter 000, the counter will go dark, and your loop will run until you turn off the Scribbler.
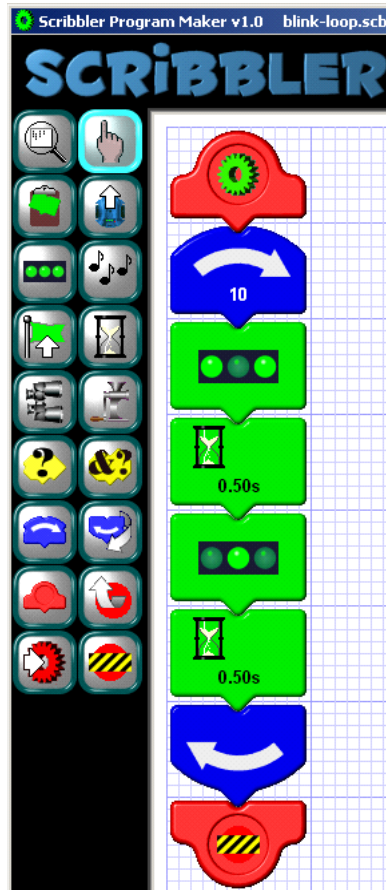
√ Set your loop counter to 10. It will look like 010.

√ Click the yes button to close the loop counter.

This will make your loop repeat 10 times. If you want to change the counter later, you can open the counter box again through the edit menu, just the same way you can with a control box.

Filling In the Loop

Now let's put some action in this loop! We can use the LED tiles and pause tiles to make the lights blink in a pattern. Let's see if you know which buttons to click to build this program.

√ Place two LED tiles in your program.

√ Set the top LED control box so the right and left LEDs are on, and the middle LED is off.

√ Set the bottom LED control box so the right and left LEDs are off and the middle LED is on.

√ Place a pause tile between the LED tiles.

√ Set the wait awhile time to 0.50 seconds.

√ Copy the pause tile, and paste it below the second LED tile.

√ Save your program worksheet as blink-loop.scb.

√ Upload your program to your Scribbler Robot.

How did you do? If everything went right, your Scribbler's LEDs should blink on and off in a pattern 10 times.

New Actions - Sound and Motion

You have learned a lot. You know how to put tiles in a program, how to copy tiles, how to edit tiles, and how to change control box settings. You know how to use a program loop. You know how to open a new worksheet, save your worksheet, and how to load your program into the Scribbler to run it. Now you are ready to learn some new actions to make your programs more exciting.
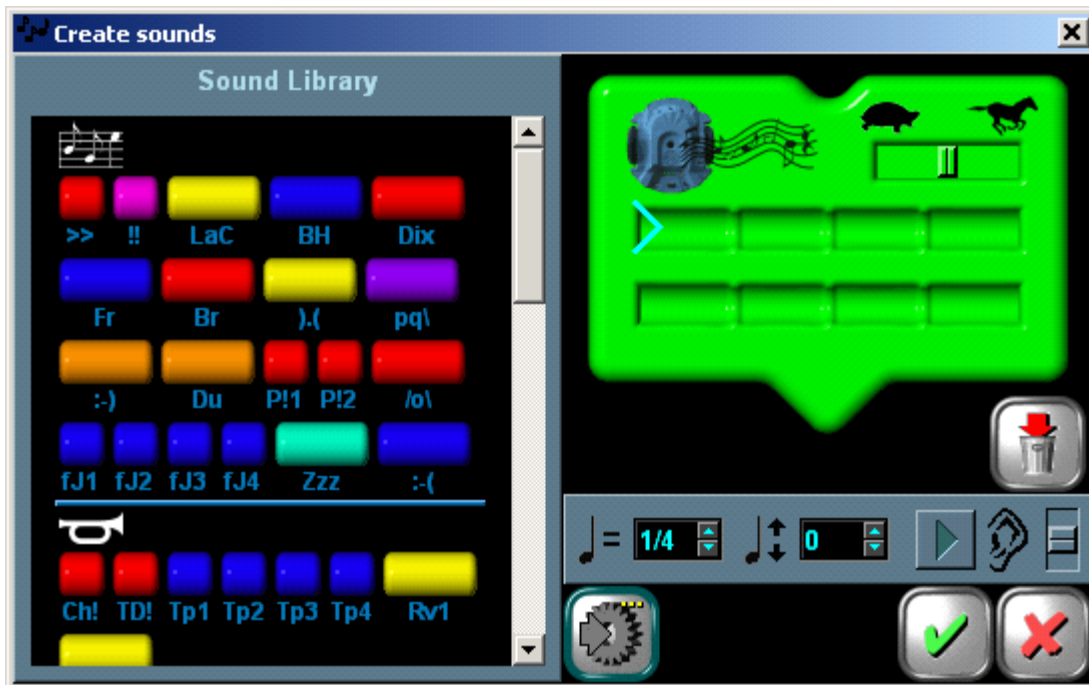
Sound Sequences

Let's try out the sound tile. This tile lets you chose sounds, noises, and songs, put them in the order you like, and place them in your program.

√   Open a new worksheet.

√   Select the sound tile, and put it in your program.

The sound tile has a very fancy control box. On the left is the **Sound Library.** Each object in it is a short recording called a **sound bite**. Click on one, and it will appear in the green tray. To take a sound bite out of the tray, click on it, then click the delete button. The group of sound bites in the tray is called a **sound sequence**.

√   Click the scroll bar and move it up and down to see the whole Sound Library.
√   Click on sound bites until your tray is full.
√   Click the yes button to close the sound control box.
√   Upload the program to your Scribbler robot.

Did you like the sound sequence you created?

Editing a Sound Sequence

You can change your sound sequence by adding or removing sound bites, and by changing the way sound bites play.

Make a single sound bite play faster or slower. Click to select it, then set its **duration** with the up or down arrows. The fastest duration is **1/16**. The slowest is **1**.
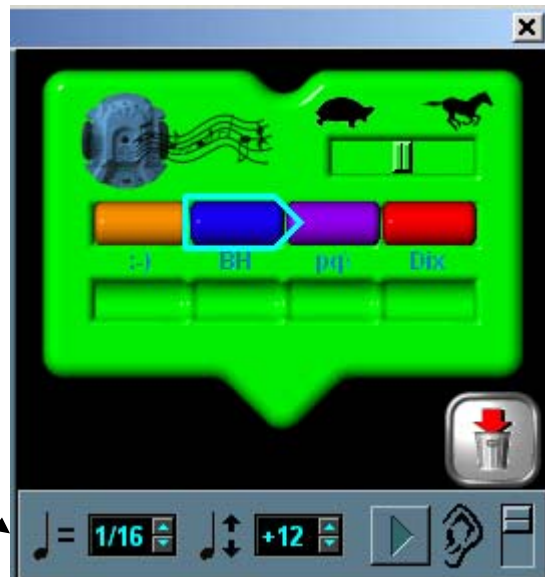
Make the whole sound sequence play faster or slower. To set the **tempo**, click and hold on this control. Slide it toward the turtle to slow the tempo, or toward the horse to speed it up.

Make a singe sound bite play higher or lower. Click on it, then change the **frequency** with the up or down arrows. You can set it high with **+12**, or set it low with **-12**.

√ Open the sound tile's control box.
√ Edit your sound sequence by deleting some sound bites and adding others.
√ Edit some sound bites in your sequence:
- Make one play faster by setting its duration to 1/16.
- Make one play slower by setting its duration to 3/4.
- Make one play higher by setting its frequency to +12.
- Make one play lower by setting its frequency to -12.
√ Change the tempo of the sound sequence.

Now we want to hear how this new sequence sounds! If your computer has speakers, you can listen to sounds before you place it in your program.
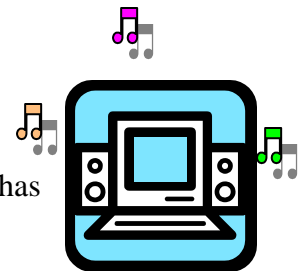
√ To listen to just one sound bite, click and hold your right mouse button on it.
√ To hear a whole sequence in the tray, click the **playback** button.

Now let's run the program so the sound sequence will play on the Scribbler's speaker.

√ Close the control box.
√ Save your program if you want to.
√ Upload your program to the Scribbler.
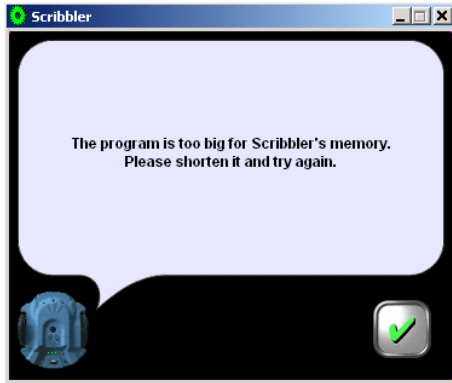
Sound tiles are fun to have in a program. But you must keep one thing in mind. Sound tiles use up lots of the Scribbler's memory.

If you use many song tiles in your program, it might get too long for the Scribbler's BASIC Stamp microcontroller brain to remember it all. If this happens, you will get an error message when you try to upload your program. So, try to use only a few sound tiles in each program.

Motion

Let's get the Scribbler Robot rolling! The **motion** tile lets you set the speed and direction for each wheel's motor.

√   Open a new worksheet.
√   Click on the motion tile, and put it in your program.

Speed and Direction

The red and green arrows show the speed and direction of each wheel. Let's try the controls.

Click and slide this control up or down to change the **velocity** (speed and direction together) of the Scribbler.

Click and drag the red and green arrows up and down. This changes the speed and direction of each wheel separately.

Keep clicking the **Quick Motion** button to see 9 different preset actions. Watch how the other controls change.
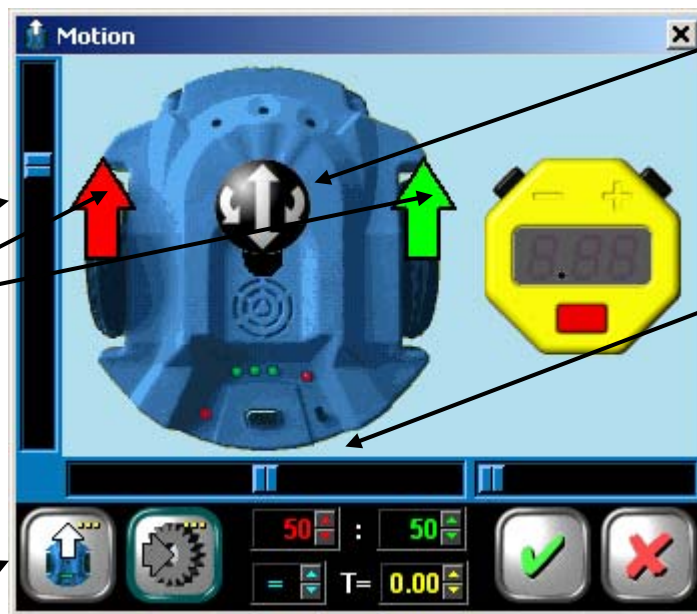
Click and hold the joystick, and then move it around. It sets both wheels at once.

Click and slide this control to make the Scribbler turn left or right. The farther you move the control, the faster the Scribbler will turn.

Use the red and green numbers to set each wheel's velocity one at a time.
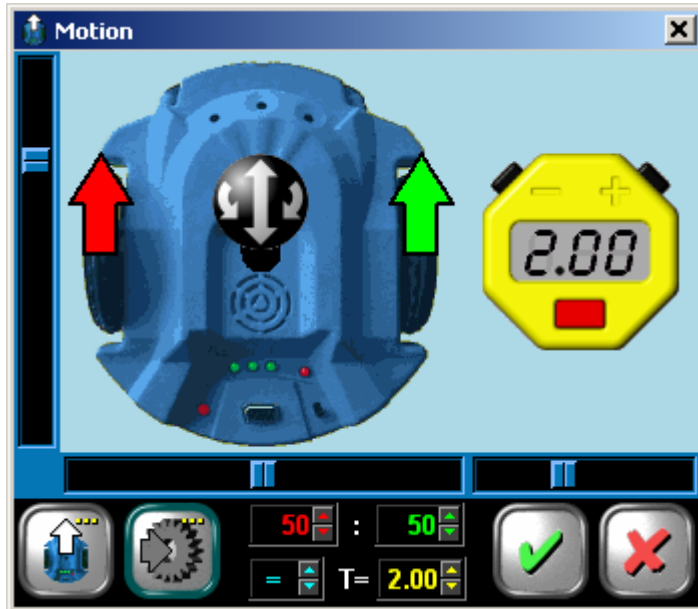
100 = full-speed forward.
0 = stop.
-100 = full-speed backward.

Setting the Move Time

In addition to setting the action tile's speed and direction, you can also choose how long you want the Scribbler to drive this way. This setting is called the **move time**.

The shortest the move time can be is **.005 seconds**. The longest the move time can be is **5 seconds**.

There are three ways to set the move time

← Click the red button on the yellow stopwatch to turn it on. Click on the "**+**" and "**-**" buttons to change the move time.

← Or, click and slide this control to change the move time.

Or, use the yellow number to set the move time. Change the move time with the yellow arrows.

Continuous Motion

If the stopwatch is turned off, the Scribbler will keep driving the way you set it. It won't change direction or stop unless you put another move tile in the program.

Avoid Accidents

Be careful! While you program your Scribbler to move, it might drive off the table.

√ Put your Scribbler Robot on a small box so the wheels don't touch anything.

√ Set the Quick Motion to "forward."

√ Set the move time to 2.00 seconds.
√ Click the yes button to close the motion control box.
√ Upload the program to the Scribbler.
√ Unplug the serial cable and put the Scribbler on the floor.
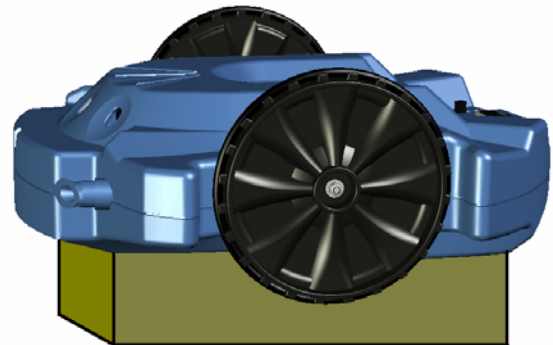
The Scribbler will drive forward for 2 seconds every time you push its reset button.

Combining Light, Sound, and Movement

Let's use three action tiles together in a program with a loop. This program will make the scribbler spin around while flashing the LEDs and making sounds. It will stop after the program loop repeats five times.

√ Open a new worksheet

√ Put a motion tile in the program.

√ In the motion control box, set the Quick Motion to "spin right."

√ Leave the move time turned off.

√ Place a program loop below the move tile.

√ Set the loop counter to 5.

√ Place two LED tiles inside the loop.

√ Set the top LED control box so the right and left LEDs are on, and the middle LED is off.

√ Set the bottom LED control box so the right and left LEDs are off and the middle LED is on.

√ Place a sound tile below each LED tile.

√ Set the top sound tile to play sound bite R21.

√ Set the bottom sound tile to play sound bite R22.

√ Place a new motion tile below the loop.

√ In the motion tile control box, set the Quick Motion to "stop."

√ Save your program as spin.scb.

√ Upload your program to your Scribbler, and then unplug the serial cable.

√ Place your Scribbler on the floor, push the reset button and watch it go!

# Test a Condition with Object Detection

Let's program the Scribbler to make a decision. The program will **test a condition** to see if it is **true** or **false.** What the program does next will depend on the answer.

Infrared Object Detection
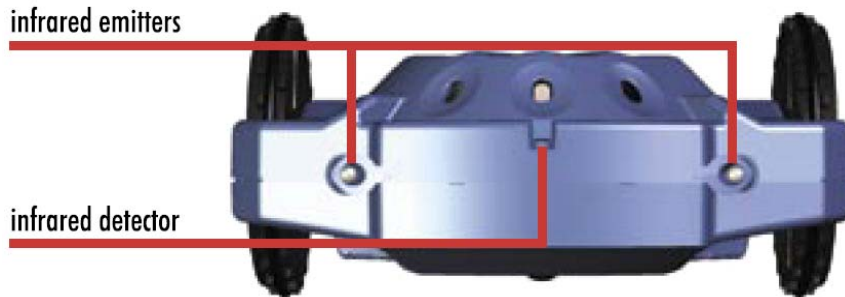
We will use the Scribbler Robot's **object detection system** for our test. The condition we will test for is **"The Scribbler does not see an object."**

Infrared is light of a color that human eyes cannot see. The Scribbler can see this light reflecting off an object and bouncing back to the infrared detector "eye." The Scribbler can use its object detection system to see if its path is clear while it drives.
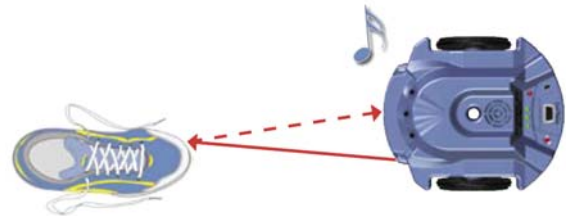
If-Then-Else
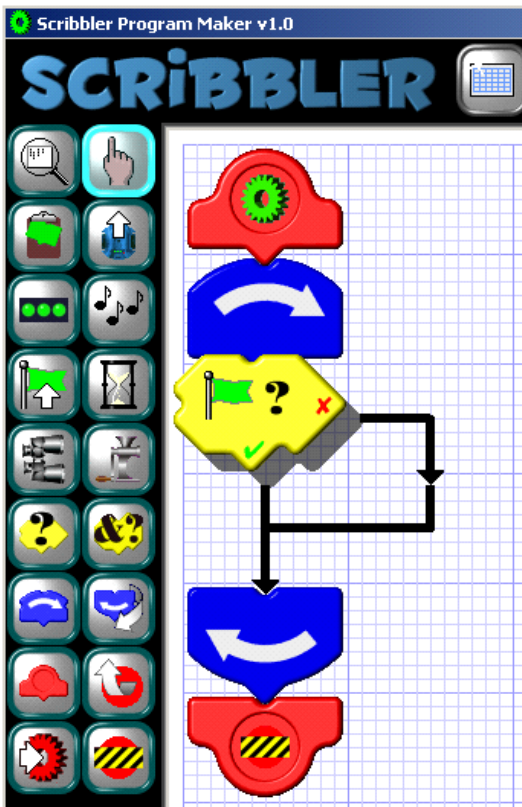
The if-then-else tile lets you choose a condition to test. Then it makes the program branch one way or another depending on the answer. Let's try it.

    √   Open a new worksheet.

    √   Place a loop in the program, and leave the loop counter turned off.

    √   Place an if-then-else tile in the loop.
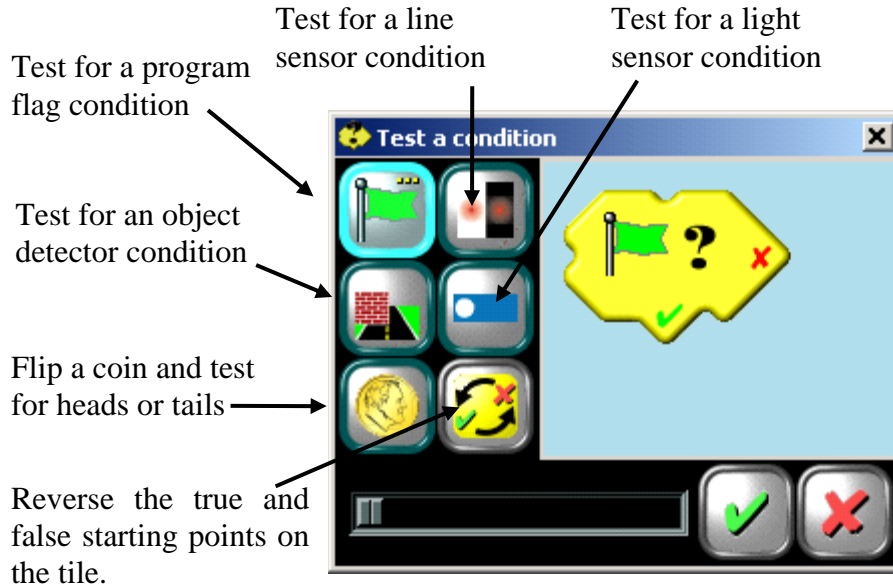
See how there are two paths coming out of this tile?

If the answer to a test is true, the program will follow the path straight down, beginning at the yes checkmark.

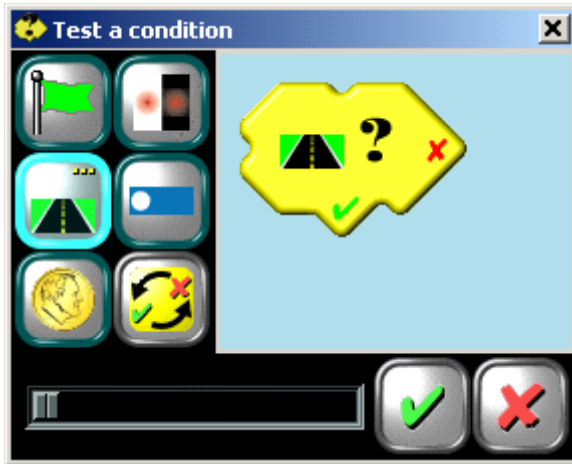If the answer to a test is false, the program will follow the other path, beginning at the no mark.

Now, we need to choose a test condition inside this tile's control box.

Choosing a Test Condition

There are five test buttons in this control box. Each button watches something. Some watch the Scribbler's sensors. Others watch for a condition in the program. Keep clicking on each of these buttons to see all conditions it can test for.

Test for a line
sensor condition

Test for a light
sensor condition

Test for a program
flag condition

Test for an object
detector condition

Flip a coin and test
for heads or tails

Reverse the true and
false starting points on
the tile.

Testing an Object Detector Condition

Our program will test for an object detector condition. Let's set the control box.

√ Click on the object detector button.

√ Keep clicking on it until it looks like a clear roadway.

√ Click the yes button to close the control box.

Remember, our test is **"The Scribbler does not see an object."** This is false if the Scribbler sees an object on the right, on the left, or on both sides. If this is true, the Scribbler has a clear road.

Making Two Action Paths
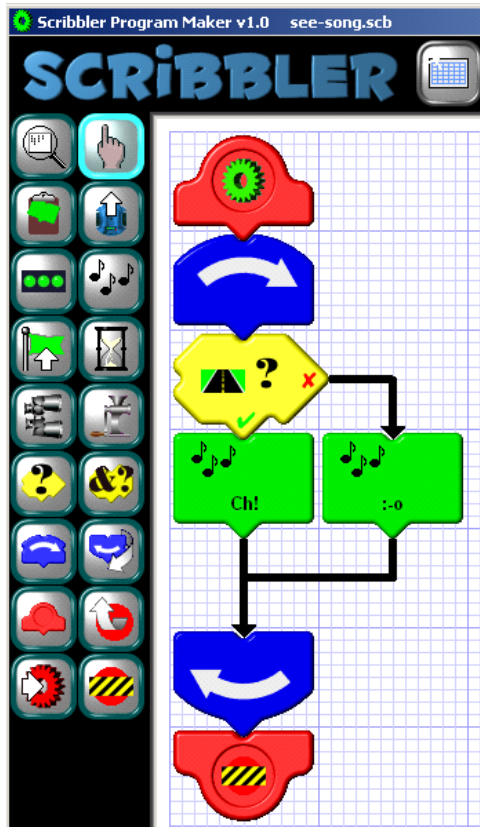
Now, let's put an action tile in the "true" path, and a different one in the "false" path.

In this program, the Scribbler will play one sound if it does not see an object. A different sound will play if the object detector sees something in front of the Scribbler.

√ Place a sound tile in the "true" path, right below the if-then-else tile.

√ Set this sound tile to play "Charge!" **Ch!**

√ Place a new sound tile in the "false" path, right below the black arrow.

√ Set this sound tile to play "Uh-oh!" **:-o**

√ Save the program as see-song.scb.

√ Upload the program to the Scribbler.

- Is your Scribbler playing "Charge?" The object detectors cannot see anything.

- Is your Scribbler saying "Uh-oh!" ? If it is, the object detectors can see something.

√ Put your hand in front of the Scribbler and remove it to make each sound play.

Combine Testing and Looping with Sound, Lights and Motion

Now let's put together all our tricks into one program. This program will make the Scribbler drive in a circle with its LED lights on. It will watch for obstacles with its infrared object detectors. If it sees an object in its path (like your hand) it will stop, turn off the LEDs, and say "uh-oh" over and over until the obstacle is removed. When the path is clear, it will start driving in its circle again.

Instead of starting from scratch, we are going to edit the last program.

√ Open see-song.scb (if it is not already open).

√ Save the worksheet again with a new name: circle.scb.
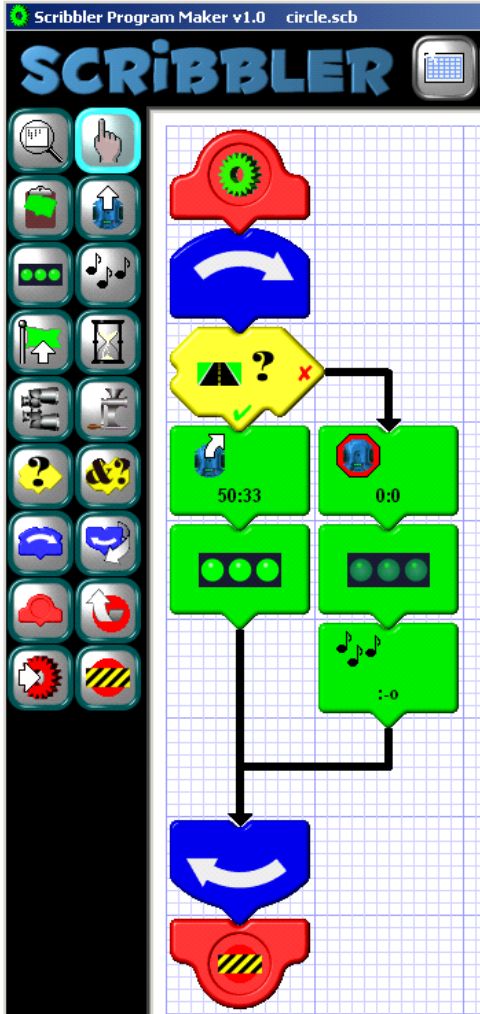
Now, let's remove a tile we won't need to use.

√ Delete the sound tile on the "true" path, below the if-then-else tile.

Next, let's add two motion tiles.

√ Place a motion tile on the "true" path below the if-then-else tile.
√ In the motion control box, choose the "forward-right" Quick Motion. Do not set a move time.
√ Place a motion tile on the "false" path above the sound tile.
√ In this motion control box, chose the "stop" Quick Motion. Do not set a move time.

Now, let's add two LED tiles.

√ Place an LED tile on the "true" path below the motion tile.
√ In the LED control box, set all the switches to on.
√ Place an LED tile on the "false" path between the motion tile and the sound tile.
√ In this LED control box, set all the switches to off.
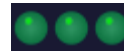
Let's make sure the Scribbler will not drive off the table while we upload the program.

√ Set your Scribbler on a small box so the wheels are free.
√ Save your program again.
√ Upload the program to the Scribbler.

Let's see the program in action! Set the Scribbler on the floor in an open area where it can drive in a circle.

√ As the Scribbler is driving, put an object in front of it.

Did it stop driving, turn off the lights, and say "Uh-oh"?

If it did, **congratulations to you!** If it did not, look carefully at the program on this page to see what parts don't match and try again.

Now that you know how to build programs for your Scribbler, experiment on your own!
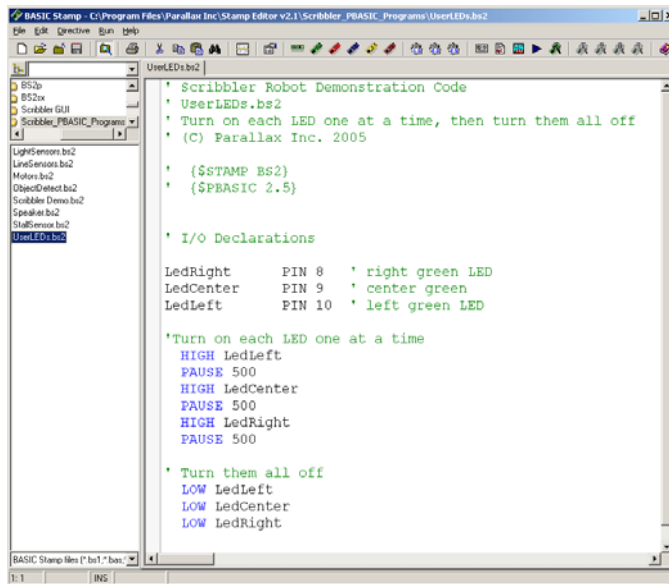
A Note to Parents and Teachers

The *Scribbler GUI Programming Guide* is a series of short modules that introduce robotics and programming concepts in an illustrated, easy to read format.

This module, First Programs, was designed to help the beginner with learning to use the Scribbler Program Maker through building simple programs. It did not use all of the features of the software, or all of the capabilities of the Scribbler Robot. Watch our website, www.ScribblerRobot.com, for more modules in this series.
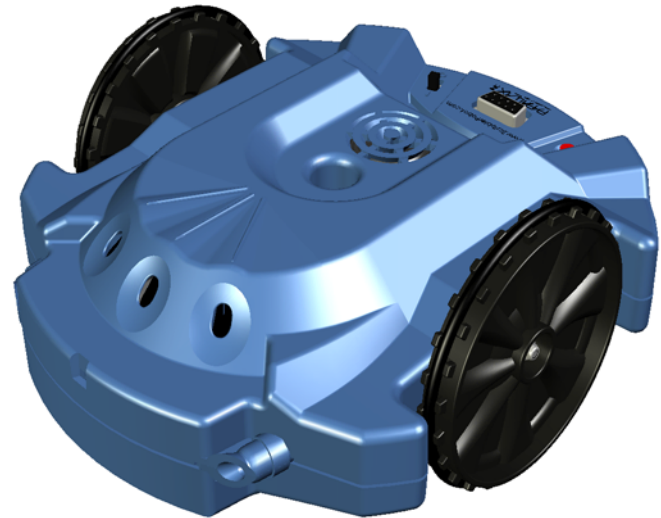
Scribbler Forums

Would you like to share your experiences with the Scribbler Robot and Program Maker with others? Visit our moderated forums at http://forums.ScribblerRobot.com where you can post questions, help others, and share ideas with other Scribbler owners.

Text Programming

Would you like to learn to directly program the BASIC Stamp 2 microcontroller brain inside the Scribbler Robot? The BASIC Stamp Editor software lets you write programs for your Scribbler with text. The straightforward PBASIC programming language is easy to learn.

You can download the latest versions of the BASIC Stamp Editor and the *Scribbler PBASIC Programming Guide* free from our website www.ScribblerRobot.com.

Have a Great Idea?

Did you design a really great Scribbler Program Maker project that you would like to share? We are inviting educators and robotics enthusiasts to write modules for the *Scribbler GUI Programming Guide*. If you are interested, email aalvarez@parallax.com.