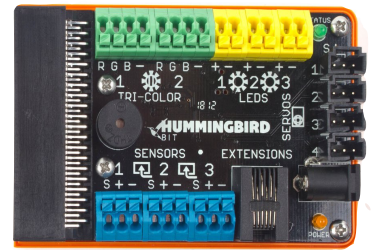


Hummingbird Bit Robotics Kit Quickstart

The Hummingbird Kit contains everything you need to build super cool robots, complete with motors, lights, and sensors out of recycled materials. The brain of the Hummingbird is the popular new BBC micro:bit. The electronics just work when you plug components into the clearly labeled ports. There is no need to understand shields, resistors, or complex circuitry.

Best of all, Hummingbird robots can be programmed with Snap! (a dialect of Logo, similar to Scratch). Not only is this language designed for learning computer science and embraced by children, but the screen and outside world may now join forces.

For example, control your robot from the screen or build a game controller via the Hummingbird. No tricky libraries are required. The software you know and love just gains new blocks.



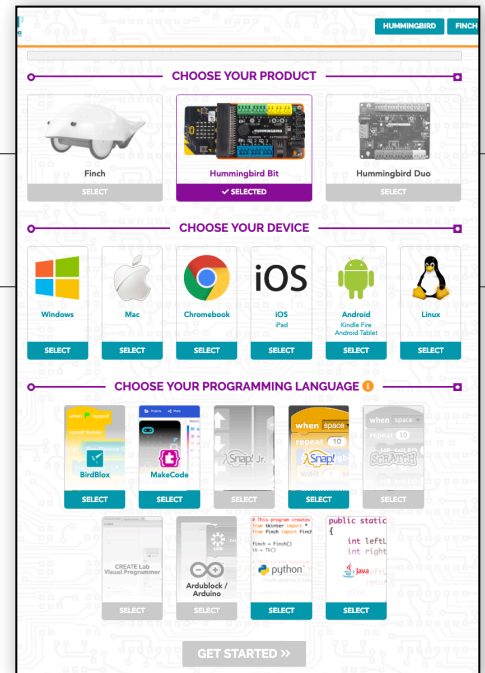
Step 1: The Birdbrain website

Birdbraintechnologies.com - click on [GET STARTED WITH OUR PRODUCTS](#)

Step 2: Select your configuration

The BirdBrain website walks you through the setup. Select the Hummingbird Bit, your device, and which programming language you want to use. Click [GET STARTED >>](#)

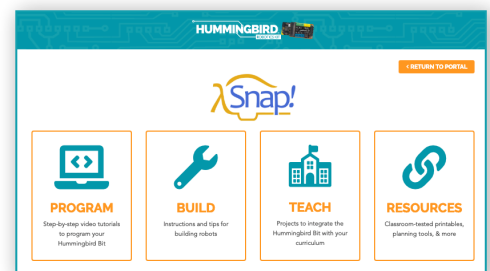
We recommend you use **Snap!** via Bluetooth. Snap! is Scratch-like, has more programming options, and lets you use the screen in your program. Edits to Snap! blocks immediately start to work on the Hummingbird. Bluetooth lets you control your robots without a cable connected to the computer. A second choice would be MakeCode. MakeCode is also block-based, but has some limitations vs. Snap! and requires the code to be downloaded to the board every time you make an edit. Believe us, this is very annoying! (*Scratch will be available soon.*)



Step 3: Connect your Hummingbird Bit

Click the **Program** button for step-by-step instructions on connecting and programming your Hummingbird. Starting with this section makes sure that you complete essential steps that you will need later.

The Hummingbird connects to your computer over Bluetooth. Follow the on-screen installation instructions. You may need to adjust your security settings to allow the Connector app to run.



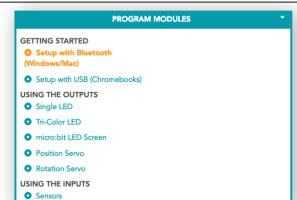
Step 4: Program, Build, or Both - Troubleshoot Along the Way

The Hummingbird Robotics Kit allows you to create programmable actions and read sensors attached to a physical robot. Start with a simple action first. See if you can make an LED blink or change the micro:bit display. Add functions one at a time, testing as you go to make sure everything still works. Add sensors to trigger your robot actions.

As you build and program, test and troubleshoot along the way. Don't build everything at once and then try to figure out what's gone wrong.



Read the User Guide!



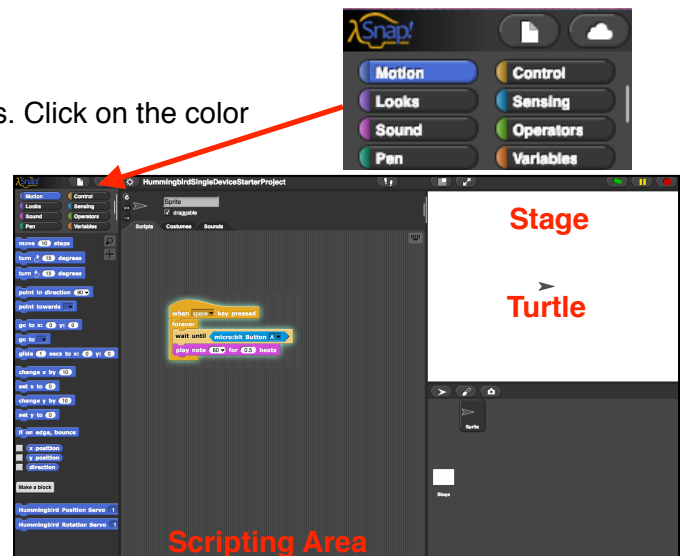
Click on the **Program Modules** dropdown to see directions for connecting LEDs, motors, and sensors.

Snap! Basics

Like Scratch, Snap! has palettes of different kinds of blocks. Click on the color coded palette buttons to see them.

- ▶ Blocks that say “Hummingbird” ONLY work with the Hummingbird.
- ▶ Blocks that say “micro:bit” ONLY work with the micro:bit
- ▶ Other blocks are used for programming or to control the actions of the on-screen Turtle.

Drag blocks out to the scripting area. Blocks will connect to form a stack that runs together. Use a control block to start your stack. The red light (top right) stops all scripts.



Connections - Inputs and Outputs

The Hummingbird board has easy-to-use connectors for the components that come in the kit. Use the orange terminal tool to press down on the connector tabs to insert wires into the correct holes. Each connector has a number printed in white on the board. The connector number is how you tell the code block which component you want to control.

Outputs

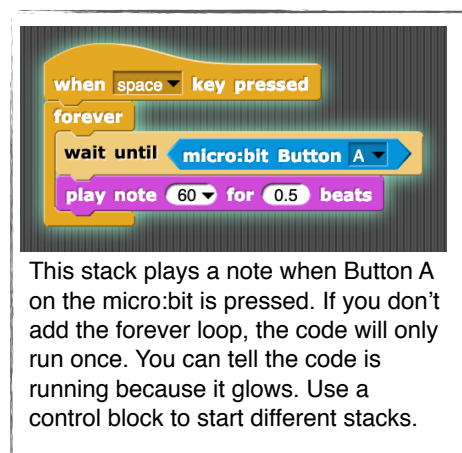
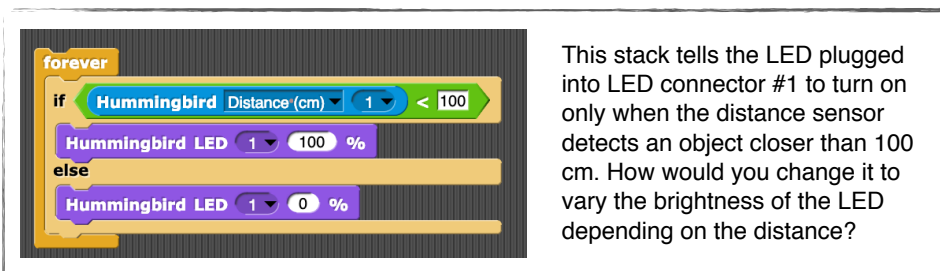
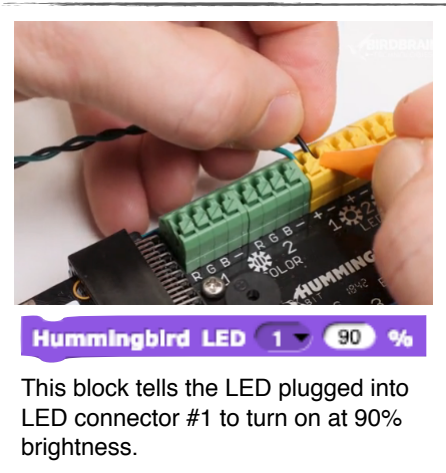
Motion Looks Sound

The program you write can send commands to the Hummingbird and the micro:bit to turn on LEDs, control the micro:bit display, make sounds, turn on/off motors, and more. There are two kinds of motors: **rotation** (turns until you tell it to stop) or **position** (turns to a fixed setting and then stops).

Inputs from Sensors

Sensing

Your program can receive numbers from sensors on the micro:bit board or connected to the Hummingbird and use those values to make decisions. For example, the value of the micro:bit compass reading can be used as numerical input to a Snap! block. (Be sure to calibrate the compass.) The input from a light sensor connected to the Hummingbird is a number that represents darkness (a low number) or brightness (a high value).



Troubleshooting

Check simple things first: is the Hummingbird powered (orange light is on)? Is the Connector app still running? Is the code block selecting the correct component in the same numbered connector? Is the component wired correctly and firmly?



Turtle to the Rescue!

Use the “say” block to show sensor readings.

