

CORTEX

QUICK START GUIDE

Software Installation Instructions:

Windows and Macintosh Users

To install Cortex, go to edventures.com/cortex. Download and install the appropriate installation package.

Android Users

Using an Internet connected tablet device, go to Google Play Store and search and install "PCS Cortex." Alternatively, you can go to edventures.com/cortex and download and install the appropriate package.

iOS Users

This application can be found on the Apple App Store under PCS Cortex using iPad only. The app does NOT show up for any other iOS devices. Contact support@edventures.com for more information.

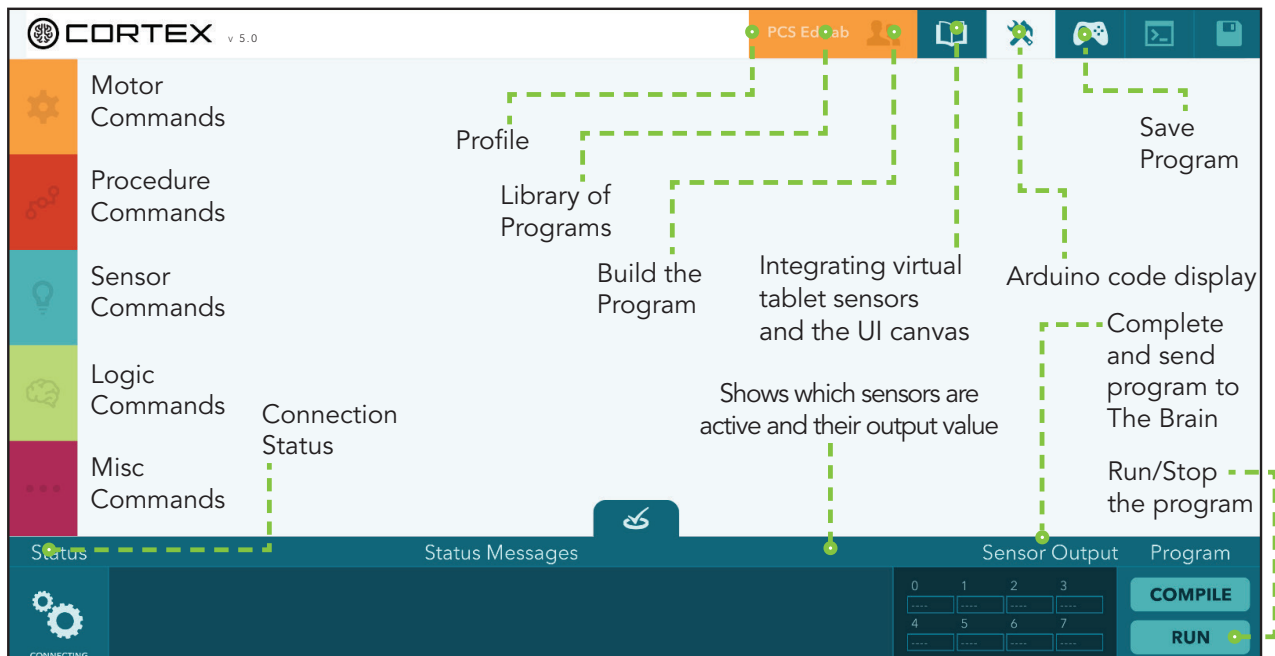


IMPORTANT NOTES ON INSTALLATION AND USE

1. A current version of Java Runtime is required to run the Cortex software. JRE 6.0+
2. There must be an active Internet connection to compile Cortex programs on a tablet.

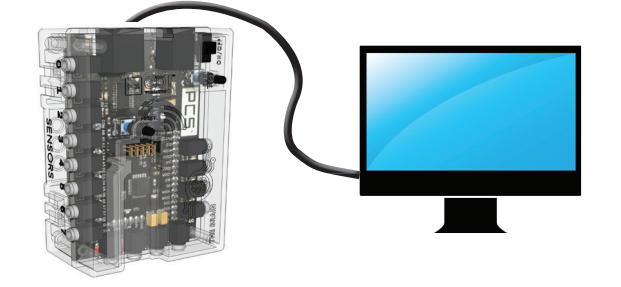
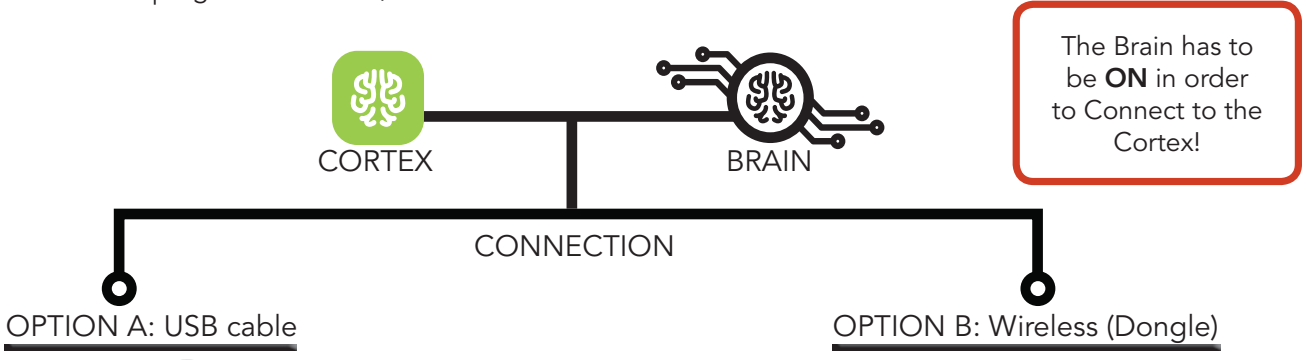
INTRODUCING CORTEX

This section introduces the basics of programming with the Cortex programming environment.

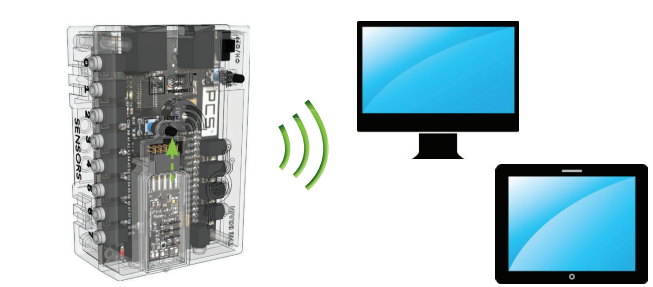


UPLOADING TO THE BRAIN

When a program is finished, it must be sent to The Brain.



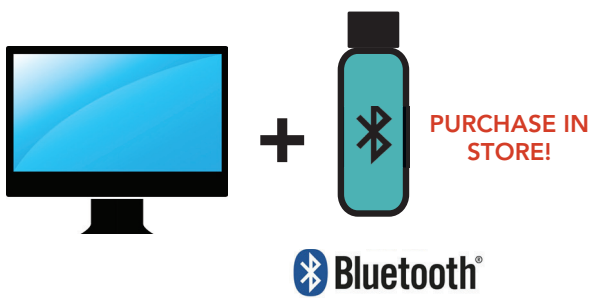
1. Make sure the Bluetooth dongle is removed. The connection does not work with the dongle attached.
2. Connect the USB cable from The Brain to the Desktop.



1. Attach the dongle to The Brain. Make sure the black circuit side is facing away from The Brain.
2. Establish a Bluetooth Connection between The Brain and the Desktop/Laptop/Tablet (see next section).

OPTION B: UPLOADING WITH BLUETOOTH

1- DESKTOP/LAPTOP with no integrated Bluetooth

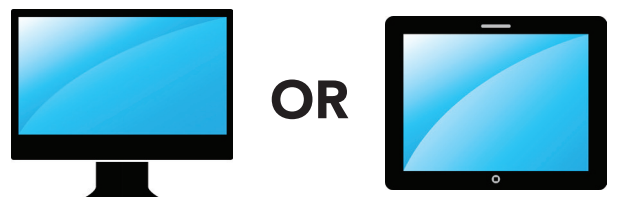


- Windows/Mac/Android**
1. **Go to:** Control Panel; Devices & Printers; Add a Bluetooth device
 2. Wait for it to Search for **PCS BT**
 3. Pair the device using **Code: 1234**

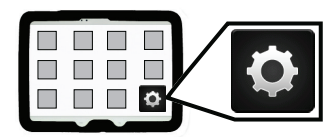
- iOS**
1. **Go to:** Applications; System Preferences; Bluetooth
 2. Wait for it to find **PCS BT 000-029**, then pair the device. The dongle number is identified by a label on its back.

**Some laptop/desktops have integrated Bluetooth. For these, you do not need to purchase a Bluetooth module.*

2- TABLET and DESKTOP/LAPTOP with Bluetooth



1. Go to: SETTINGS; Bluetooth
2. Turn Bluetooth ON
3. Search for the PCS-BT
4. Click on **PAIR**

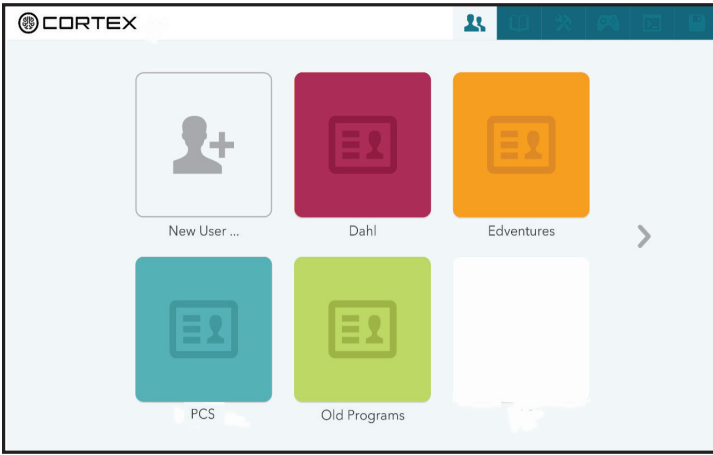


CODE: 1234

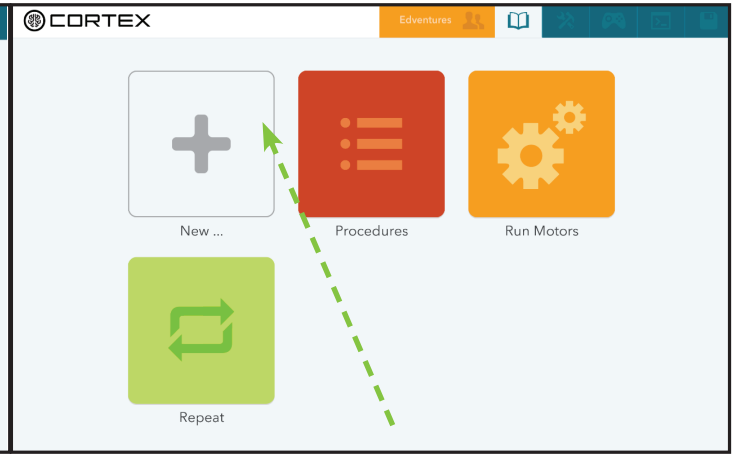
**You will have to find out where the Bluetooth connection manager is for your operating system.*

WRITING PROGRAMS

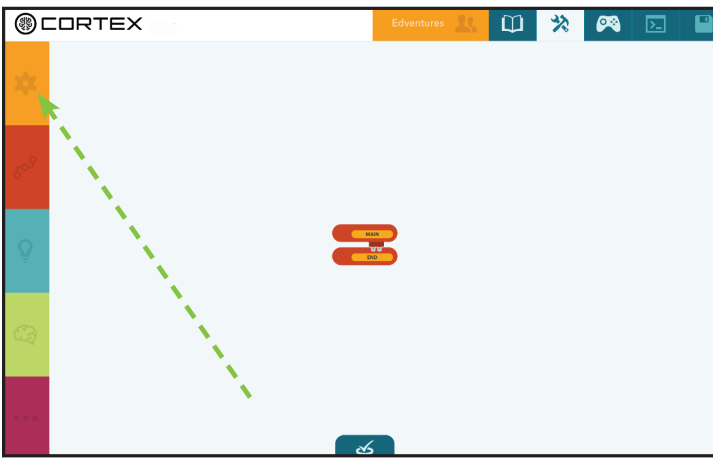
1 Add a new user or click on one of the user profiles previously made.



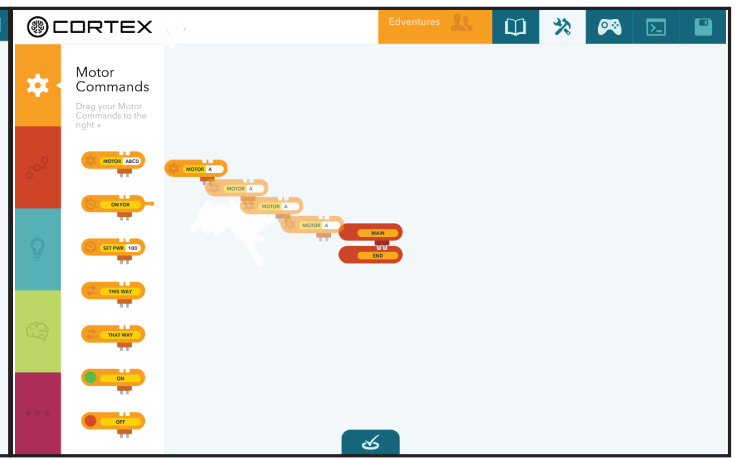
2 Once in a profile, start a new project or continue with a saved one.



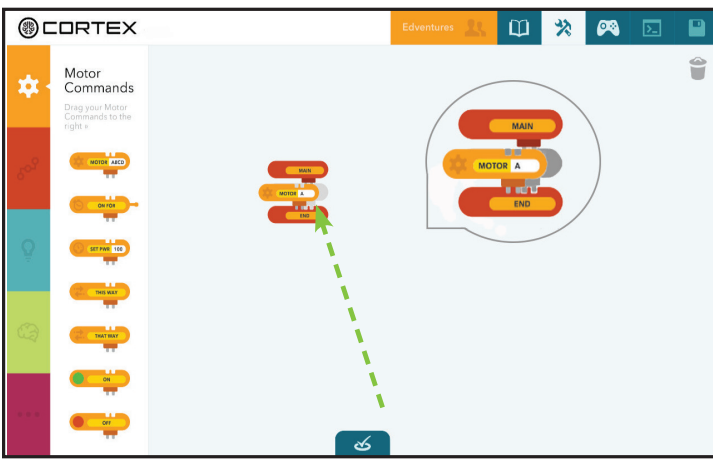
3 Click or tap the appropriate command menus to use.



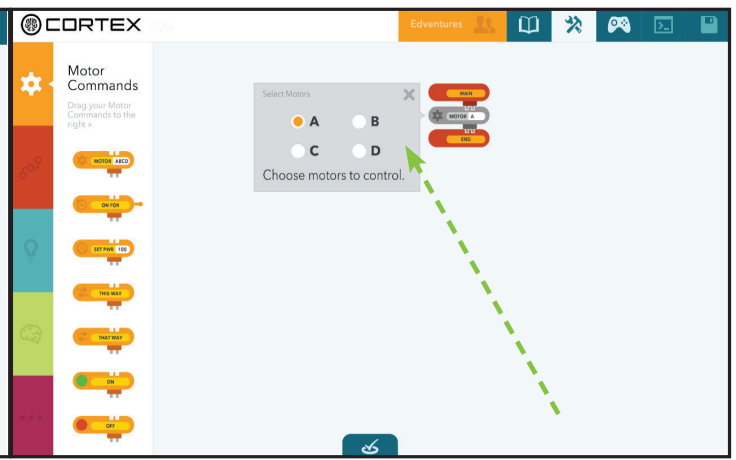
4 Drag the commands onto the workspace with a mouse or finger.



5 When placing new commands a **gray** box appears. Let the command go and it connects.



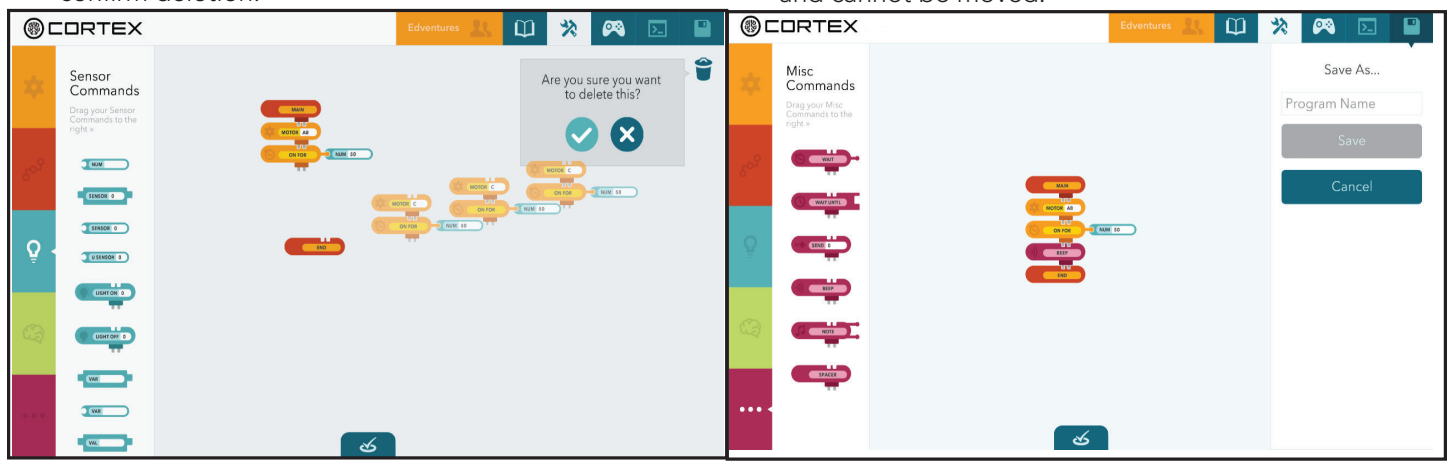
6 **Double click or tap** on motor command to select which motor(s) to use.



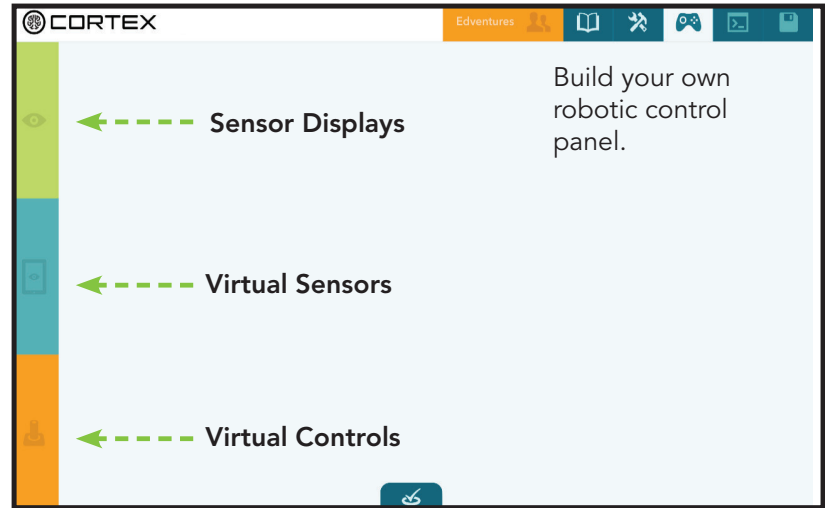
WRITING PROGRAMS (CONT.)

7 To delete commands, drag them into the trashcan. When the gray box appears click the check mark to confirm deletion.

8 **Save** programs so work can be continued on them in the future. Note: programs are saved in the application and cannot be moved.



User Interface Canvas



Sensor Displays
These displays can be used to display values of different sensors, or counters.



Virtual Sensors
Microphone, GPS, compass, accelerometer, etc. are some of the virtual sensors available on tablet devices.



Virtual Controls
Create your own control panel using joysticks, sliders, ON/OFF switches, buttons, etc.

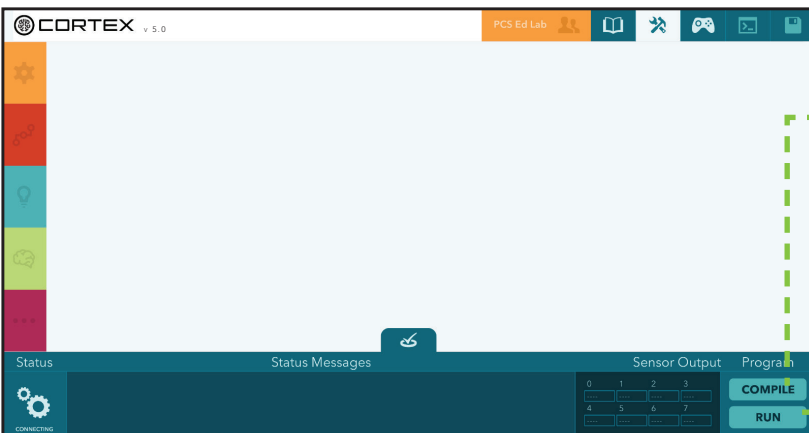
WRITING PROGRAMS (CONT.)

9 Look at the **Arduino Code** to learn Arduino language. This code can also be copied and used in the Arduino IDE.

10 Click/tap on the status icon to open up the connection status panel.



SENDING A PROGRAM TO THE BRAIN



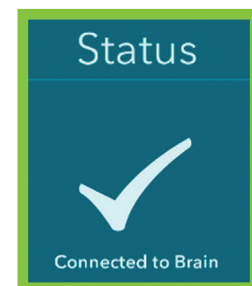
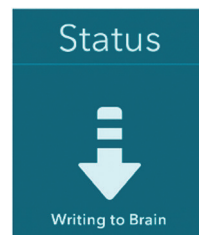
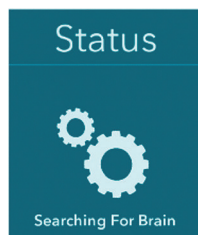
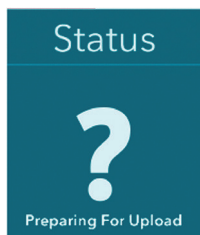
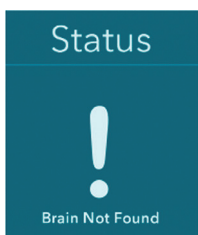
COMPILE

1. Make sure there is an Internet connection for compiling programs when using tablet devices. Computers can compile locally or online.
2. Press COMPILE to transfer the program to The Brain. It might take a few seconds.

RUN

Once the program is sent, either RUN/STOP it from the Cortex or else directly from the RUN/STOP button on The Brain.

Connection Status



If The Brain and Cortex are connected correctly and communicating, the **LINKED** icon displays. Now it's ready to compile the program and transfer it to The Brain!

NOTE: If using a tablet there must be an active Internet connection to COMPILE.

NOTE: Click/tap on the icon to refresh the connection if it is NOT LINKED.

MOTOR COMMANDS



Choose motor(s) to control.



Sets direction of selected motor(s) to run THIS WAY. Global command, so the motors always turn THIS WAY unless otherwise specified.

This is a **GLOBAL COMMAND**.



Turns selected motor(s) on for specific time.



Sets direction of selected motor(s) to run THAT WAY. Global command, so the motors always turn THAT WAY unless otherwise specified.

This is a **GLOBAL COMMAND**.



Turns on selected motor(s). To be most useful, used in a loop command.



Reverses direction of selected motor(s).

This is a **GLOBAL COMMAND**.



Turns off specific motor(s).



Sets power level on selected motor(s).

This is a **GLOBAL COMMAND**.

PROCEDURE COMMANDS



Create and name procedure (section of code) which can be called into the main program.



Create and name a new global variable.



Used to end procedure(s) and program(s).



Set the value of a variable using a number.



Call a defined procedure to run. Double click to choose procedure.



Set the value of a variable using a sensor.

GLOBAL COMMAND: A command that remains on until specified otherwise.

SENSOR COMMANDS



Turns ON the specified port which powers an LED.



Represents the value of a selected sensor for variables.



Turns OFF the specified port.



Represents the value of a selected virtual sensor for variables.



Enter a number value.



Represents the value of a selected sensor for command(s).



Represents the value of the ultrasonic sensor on the selected port.



Represents the value of a selected virtual sensor for command(s).

READING SENSOR VALUES

Knowing how to use sensors when programming is important. They send certain values to The Brain, and depending on the value they report, it acts one way or another. It is very important to find out which values are to be expected from each sensor such as the ultrasonic, infrared or light sensor as well as other tablet built-in controls like the microphone.

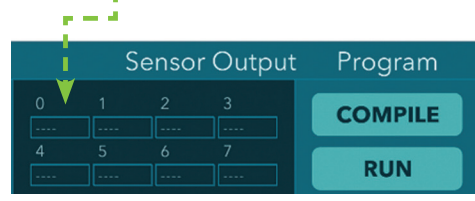
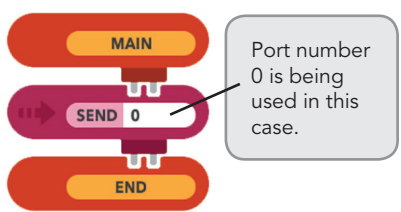
Obtaining the values

1. Create this simple program (make sure it is connected to The Brain).

2. Compile and Upload it to The Brain.
3. Run the program and see how the value changes.

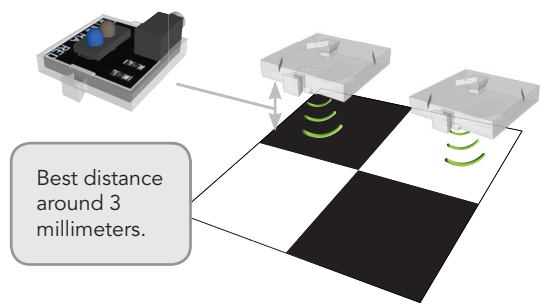
For example:

- Ultrasonic: values change according to how far an object is from the sensor.
- Infrared: values change by detecting the changes of color nearby.
- Light: value changes with brightness.
- Microphone: value changes with sound intensity.



EXAMPLE: This is an example of how to analyze the sensor values

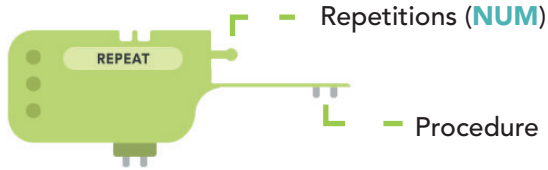
1. Connect an infrared sensor to any port on The Brain.
2. Make the connection between The Brain and Cortex.
3. Build the "SEND" program shown above.
4. Press Compile, when it is done uploading, put the front of the sensor facing the white area very close (3 mm). While that is done look at the "SENSOR OUTPUT" on the Cortex.
5. Do the same thing with the black area. See how the values change?



TEST THE SENSOR HERE!



LOGIC COMMANDS



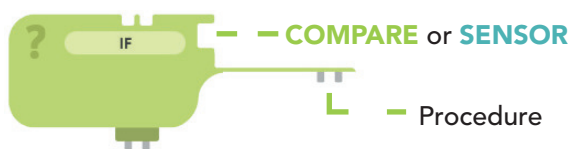
Repeats the attached code the number of times defined.



Repeats the attached code infinitely until a "BREAK" command is reached, or the user stops the program.



Stops a Loop from continuing and sends the program to the next command after the Loop.



If the described condition is met, the attached code is run.

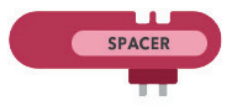


If the described condition is met, the attached code is run. Else, if the condition is not met, the second branch of defined code runs.

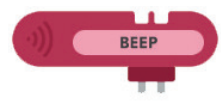


Used to compare two values. Can be used to check if two values are equal, greater than, less than, etc.

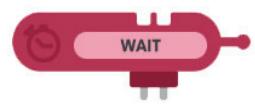
MISCELLANEOUS COMMANDS



Add a space in the program where needed (Cosmetic).



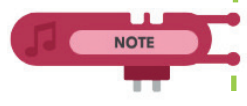
Play a short beep.



Delay for a given time. No commands in the program run during this.



Delay for a time defined by a sensor or variable.



Play a note of a specific frequency for a specific duration.

Note	Frequency:
C	261
D	294
E	329
F	344
G	392
A	440
B	493
C	523

In the grey square to the left are the main note frequencies. If you multiply or divide any frequency it will raise or lower by an octave.

For example, 344 and 688 are both F's, but 688 is one octave higher. Check it out at: riq.edventureslab.com