Spooky Signals: micro:bit Ghost Throwing Challenge

Overview

The Teleporting Ghost project brings coding to life by creating an interactive ghost that can magically travel between two micro:bits! Using the micro:bit's built-in motion sensor and radio capabilities, you will learn how devices can communicate wirelessly while creating a fun and spooky experience.

What you will Learn

This hands-on project introduces important programming concepts like:

- [ ] Wireless communication between devices
- [ ] Using sensors (accelerometer) to detect motion
- [ ] Creating custom images using LED displays
- [ ] Understanding basic program flow and logic

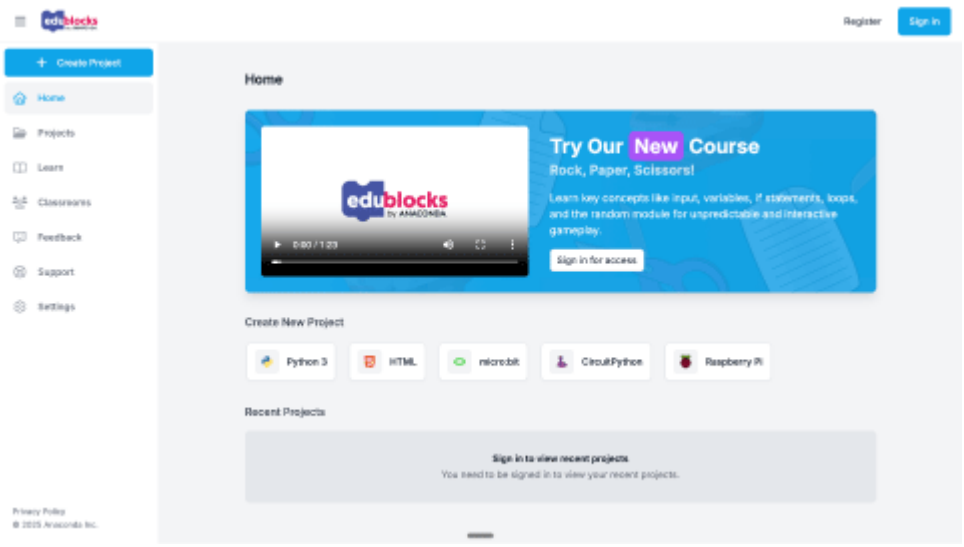
What you will Need

- 1 x micro:bit
- 1 x micro USB cable
- 1 x battery pack for the micro:bit (optional)

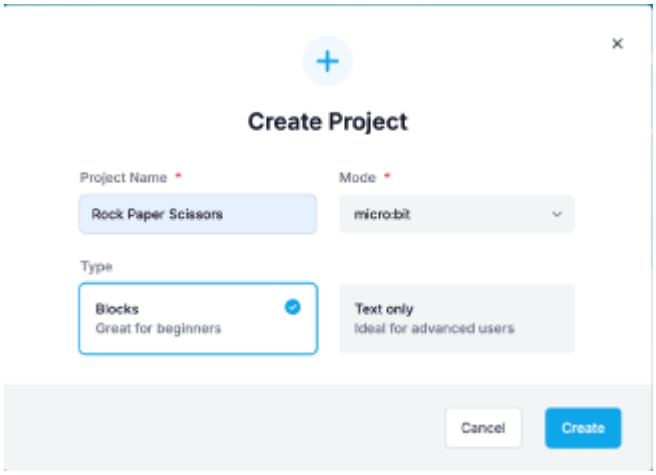
The project is not only entertaining but also demonstrates how modern devices can sense movement and share information wirelessly - the same technology used in everything from video game controllers to smart home devices.

Navigating to EduBlocks

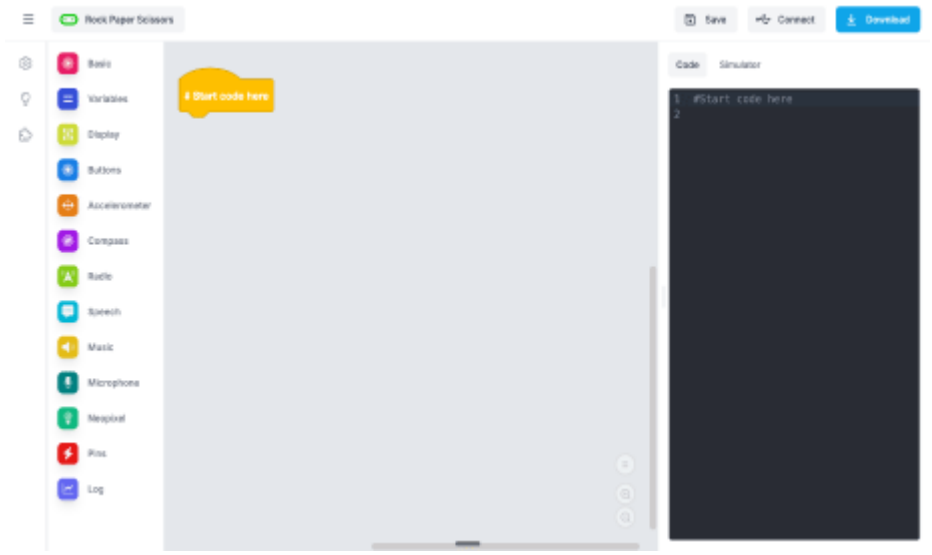
1. Open your favourite browser (we recommend Google Chrome).
2. Within the address bar of the browser type app.edublocks.org or on a tablet or phone press create code.



3. Select micro:bit under **Create New Project** to open the micro:bit coding editor. Name your project **Spooky Signals** and make sure you have **Blocks** selected under **Type**. See the image below.



4. Select **Create** to open the micro:bit editor.



We are now ready to start coding!

Coding

Importing Libraries

1. From Imports within the Basic menu, select and drag a `from microbit import *` block to the code area and attach it under the `# start code here` block.

2. From Imports within the Basic menu, select and drag an `import radio` block to the code area and attach it under `from microbit import *` block.

Configuring the Radio Module

1. From the Radio menu, select and drag a `radio.on()` block to the code area and attach it under the `import radio` block.
2. From the Radio menu, select and drag a `radio.config(group=23)` block to the code area and attach it under the `radio.on()` block. Change the **23** to **30**.

If you are working with more than one pair of micro:bits make sure the Channel value is different per pair.

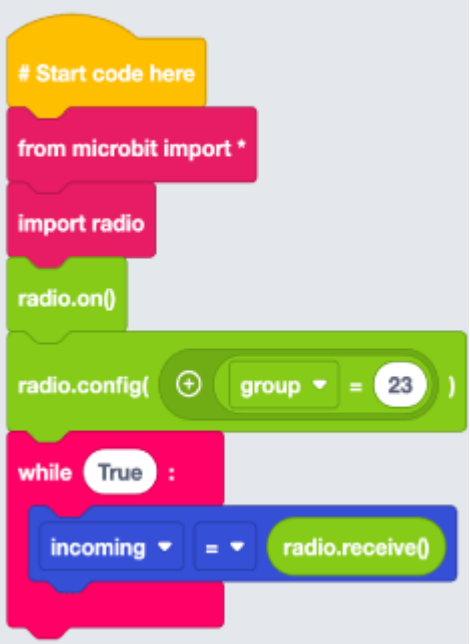
Creating the While Loop

From Loops within the Basic menu, select and drag a `while True:` block to the code area and attach it under the `radio.config(group=30)` block.

Create Incoming Variable

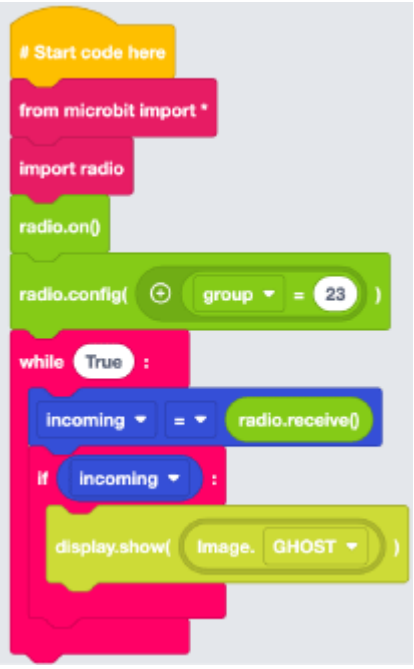
> WHAT IS A VARIABLE
> Think of a variable as a box that stores information that can be used throughout our program. We give variables a descriptive name so we and others can understand what is going on within our program.

1. From Variables select **Create variable...**. Name the variable **incoming**.
2. From Variables select and drag `incoming = 0` block to the code area and attach it under the `radio.config(group=30)` block.
3. From Radio select and drag a `radio.receive()` block and attach it within the **0** of the `incoming = 0` block.



Configuring Incoming Message

1. From Logic within Basic, select and drag an `if True:` block to the code area and attach it under the `incoming = radio.receive()` block.
2. From Variables select and drag an `incoming` block to the code area and attach it within the **True** of the `if` block.
3. From Display, select and drag a `display.show("Hello")` block to the code area and attach it within the `if incoming:` block.
4. From Display, select and drag a `Image.RABBIT` block to the code area and attach it within the **Hello** of the `display.show` block. Select **RABBIT** and choose **GHOST**



Configuring Outgoing Message

1. From Logic within Basic, select and drag an `if True:` block to the code area and attach it under the `if incoming:` block.
2. From Accelerometer select and drag an `accelerometer.was_gesture('shake')` block to the code area and attach it within the **True** of the `if` block.
3. From Display select and drag a `display.clear()` block to the code area and attach it within the `if accelerometer.was_gesture('shake')` block
4. From Radio select and drag a `radio.send("Hello")` block to the code area and attach it under the `display.clear()` block. Change `**"Hello"** to **"GHOST"**.

Completed Code:

Now that we have completed our code. Lets download it to the micro:bit.

Downloading the code to the micro:bit

1. Take the micro USB cable and connect the micro:bit to the computer.
2. Select **Connect** and follow the pop-ups on screen to pair the micro:bit to the web browser.
3. Select the **Download** button to download your code to the micro:bit.

How to Play

Using the micro:bit

When you shake one micro:bit, the ghost displayed on its LED screen will vanish and mysteriously appear on the other micro:bit's display. It's like playing a game of supernatural catch!

Grab a friend and have a go.