# Proximity beacon

## Step 1: Make it

### What is it?

Use radio to sense how close another micro:bit is and then make a treasure hunt game or use it to help people know they're at a safe social distance.

### How it works

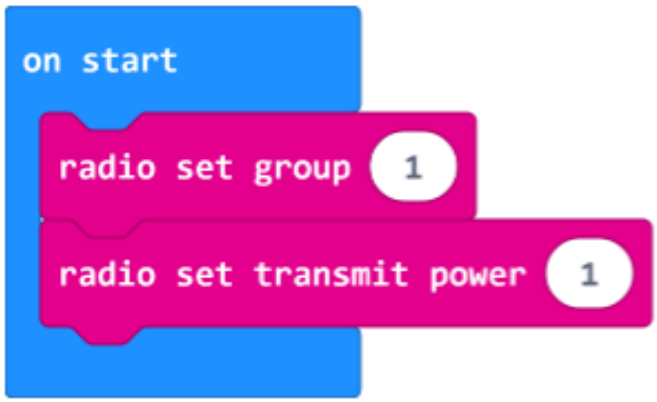
- You need at least 2 micro:bits for this. We'll create two different programs, one for the beacon which constantly sends a low-power radio message. The other program goes on the receiver.
- When the receiver picks up a message from the beacon, it stores its strength in a variable called **signal** and shows it on its LED display.
- Radio signals get stronger the closer you are to the transmitter, so if the signal is strong it means the other micro:bit is probably close.
- If the radio signal is weak, the other micro:bit is probably further away.
- It displays a bar graph which gets bigger the stronger the signal and the closer you are. It uses the maths **map** block to map radio signal strength numbers from the range -95 (weak) to -42 (strong) to a range 0 to 9 we can use to draw a bar graph.

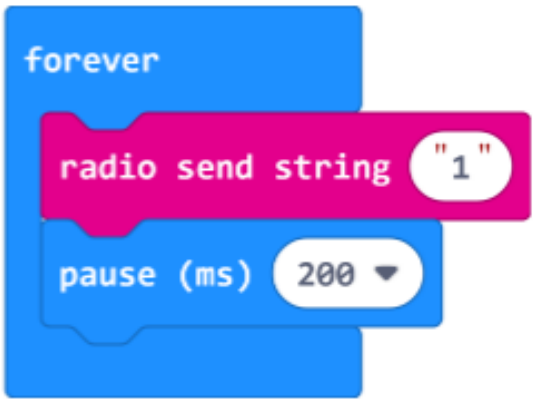
### What you need

- 2 micro:bits and battery packs
- MakeCode or Python editor
- battery pack (optional)

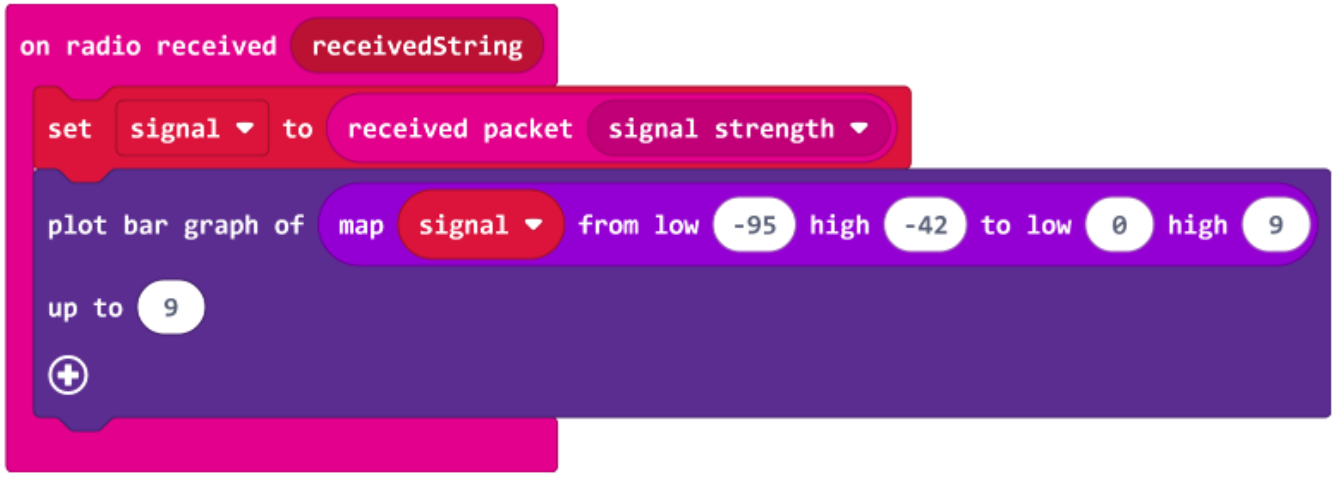
## Step 2: Code it

### Transmitter / beacon




### Receiver

## Step 3: Improve it

- Combine the beacon and receiver code so you can have one micro:bit that does both tasks.
- Make wrist bands so you can wear your proximity detectors.
- How strong is the signal when you're 1 or 2 meters apart? Modify the code to trigger a visual or audible alarm when someone is too close.
- Use these programs to make a treasure hunt game: hide the beacon and put the receiver code on lots of micro:bits
- If you're outdoors or in a large space, experiment by changing the transmitter power. It can be any number from 0 to 7