• **Download the program to micro:bit and see what happens.**

Can OLED be used to make a more comprehensive display, showing all the data and information we need on the OLED, such as time, light, temperature and humidity?

Experiment 16 – Wireless communication

Instruction

Wireless communications are everywhere in our daily life, such as mobile phones we most often see and use. Wireless communication brings us a lot of conveniences. It enables us to communicate with each other even thousands of miles apart. Thanks to the radio/Bluetooth module on the micro: bit board, wireless communication can also be achieved between our two micro: bits or communicate with the phone.

Target

• Learn how radio work on micro:bit and use it to communicate between 2 micro:bits.
• Learn how to send and receive numbers between 2 micro:bits.

Required Parts

• Micro:bit x2
• Crowtail-Base shield for Micro:bit x2
• Crowtail-Buzzer x1
• Crowtail-Cable x1
• USB cable x1

Hardware learning and connection

Micro: bit has two wireless communication methods, one is radio communication and the other is Bluetooth communication. These two hardware modules are embedded in the upper left position behind the micro: bit, as shown in the figure. In this experiment, we will learn the standard radio on micro:bit which allows you to communicate between two or more micro:bits.
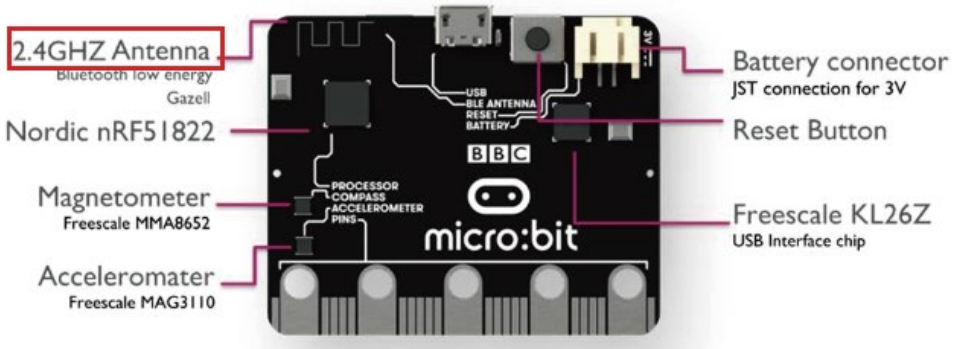
Connect Crowtail-Buzzer to P0 port of Crowtail-Base shield for Micro:bit(named bit1). Plug another micro:bit into Crowtail-Base shield for Micro:bit(named bit2, if you don't have two Crowtail-Base shield for Micro:bit, you don't have to plug this micro:bit into Crowtail-Base shield for Micro:bit). The hardware connections are as follows:
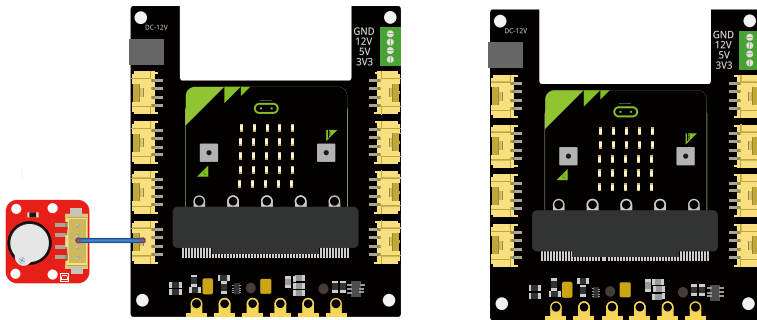
Programming and note

Send_code:

Receiver_code:

- **Radio set group:** Make a program have the group ID you tell it for sending and receiving with radio. A group is like a cable channel (a micro:bit can only send or receive in one group at a time). A group ID is like the cable channel number. If you do not tell your program which group ID to use with this function, it will figure out its own group ID by itself. If you load the very same program onto two different micro:bits, they will be able to talk to each other because they will have the same group ID. In this experiment, we set the group of two micro: bits to 1 so that they can communicate with each other.

- **Radio send number:** This block is to broadcast a number to other micro:bits connected via radio. You need to enter a parameter for this block which is the number you want to send.

- **On radio received receivedNumber:** Run part of a program when the micro:bit receives a number over the radio. "receivedNumber" is a parameter, it's the number that was sent in this packet or 0 if this packet did not contain a number. If the number received by bit1 (receiver_code) is 3, the code inside the if statement of bit1 will be run. If the number received by bit2 (send_code) is 4, the code inside the if statement of bit2 will be run.

- **Download the send_code program to bit2 and receiver_code program to bit1 and see what happens.**

> **Result**
>
> At first, the micro: bit matrix LEDs named bit2 will display a crying face, and then the micro: bit of bit1 will immediately play a melody (comfort bit2), and the crying face of bit2 will become a smiling face.

> *Does wireless communication require two or more micro:bits? Then if there are no two micro:bits, can the micro: bits be controlled wirelessly? For example, the module is used to wirelessly control the micro: bit.*

Experiment 17 – Remotely control your micro:bit

Instruction

When you're too sleepy, you definitely don't want to get up from your bed and turn off the lights. When you are busy with other things, you don't feel like being interrupted by knocking on the door. Well, in this experiment we are ready to help you solve this problem. We will use the remote control to remotely control these things, instead of having to leave for them.