

Session 1 Part A Radio Communication & Serial Data Transfer.

Radio Communication & Serial Data Transfer

Aims of this section:

- To discuss how the radio communication on the micro:bit works.
- To illustrate the code for this using temperature data.
- Show how the received data can be graphed and downloaded as a CSV file.
- To use these ideas to complete a task on radio communication involving the viewing of received data and downloading this data to a CSV file.



Radio Communication

Pre- Coding (Design)



Radio Communication

Next Step is to code this on the micro:bit.

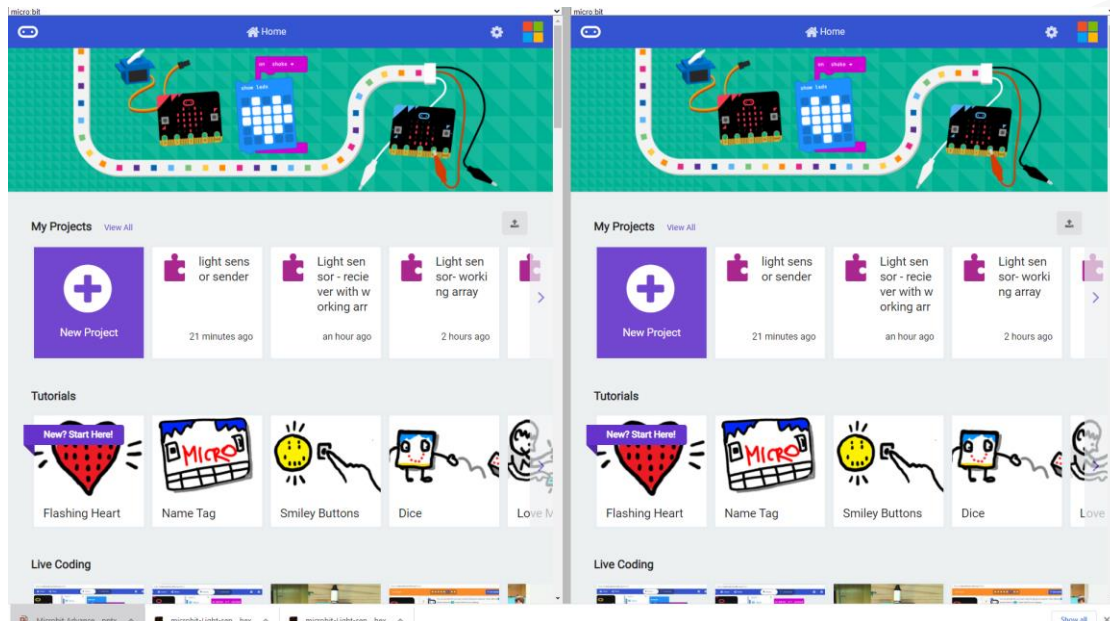
Click the link below:

[Make Code Multi Editor](https://makecode.com/multi#)

OR

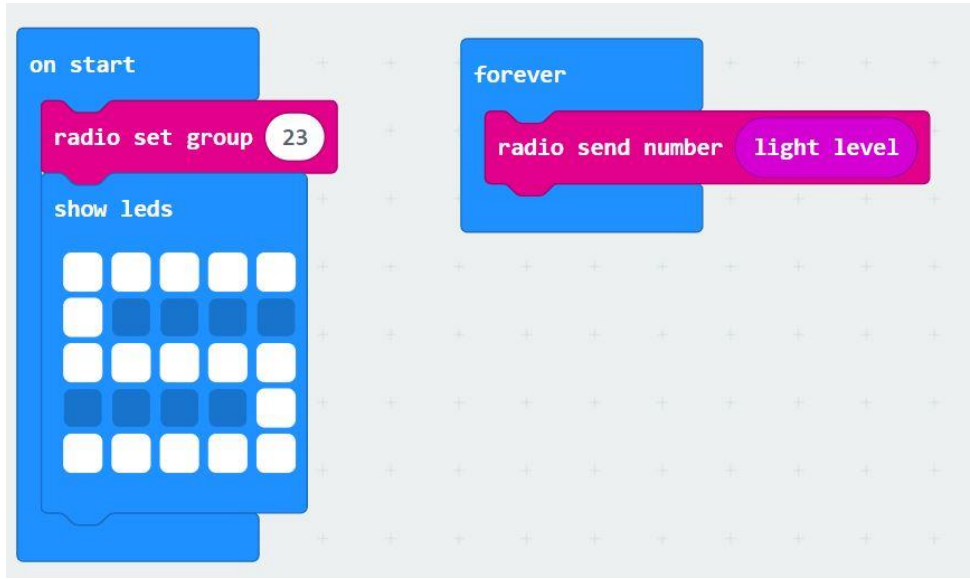
<https://makecode.com/multi#>

We will work through an example.



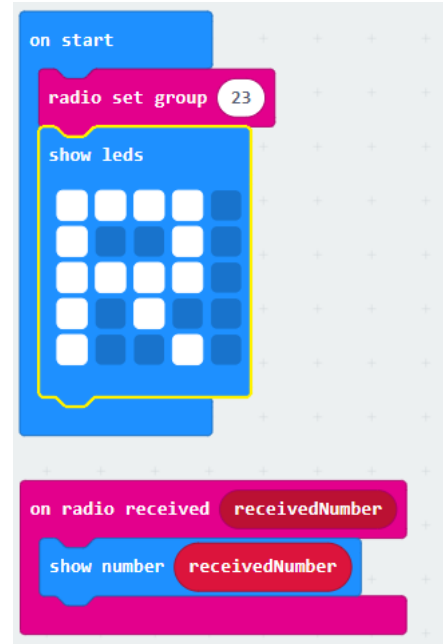
Radio Communication

SENDER CODE:



The sender code consists of two main blocks on a light gray grid background. The first block is a blue 'on start' block containing a pink 'radio set group' block with the value '23' in a white circle, followed by a blue 'show leds' block. The 'show leds' block contains a 5x5 grid of 25 squares, with the top row and bottom row being white, and the middle three rows being dark blue. The second block is a blue 'forever' loop block containing a pink 'radio send number' block with the text 'light level' in a white circle.

RECEIVER CODE:



The receiver code consists of three main blocks on a light gray grid background. The first block is a blue 'on start' block containing a pink 'radio set group' block with the value '23' in a white circle, followed by a blue 'show leds' block. The 'show leds' block contains a 5x5 grid of 25 squares, with the top row and bottom row being white, and the middle three rows being dark blue. The second block is a pink 'on radio received' block with the parameter 'receivedNumber' in a white circle. The third block is a blue 'show number' block with the parameter 'receivedNumber' in a white circle.

Group Task



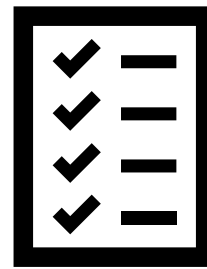
TASKS

For this and all tasks today you will need:

A person to record the group Task Reflection.

A person who is going to report how the group got on.

Learning Activity Instructions



Your task is to design and develop a solution to:

Part 1:

- Send light level via radio signal from one microbit to another.
- If the light level received is below 128 then get the receiver micro:bit to display that it is dark otherwise display it is bright. Consider the design of your display.
- Test to ensure this works on the virtual micro:bits.

Part 2:

- Open the receiver code in the normal micro:bit environment or in the offline version.
- Now update the code so that the received data is sent across the serial port.
- Don't forget to plug in and pair your device, then download the code!
- View the live data on the simulator and download the .csv file of data.
- Complete the group task Reflection Document.

TASK Solution Code

Part 1.

SENDER

The SENDER code consists of the following blocks:

- on start** block containing:
 - radio set group** block with the value 23.
 - show leds** block.
- on button A pressed** block containing:
 - radio send number** block with the variable light level.

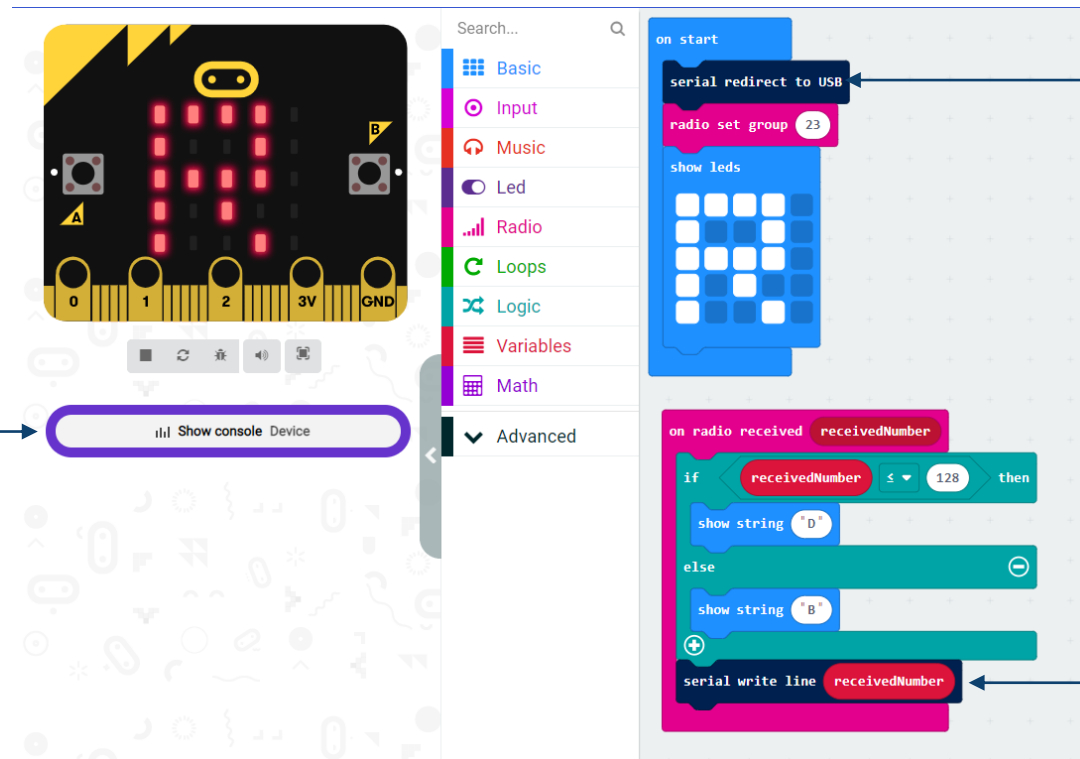
RECIEVER

The RECIEVER code consists of the following blocks:

- on start** block containing:
 - radio set group** block with the value 23.
 - show leds** block.
- on radio received** block with the variable receivedNumber, containing:
 - if** block with the condition `receivedNumber ≤ 128`.
 - then** block containing **show string** "D".
 - else** block containing **show string** "B".

TASK Solution Code

Part 2.



The screenshot shows the Scratch IDE interface. On the left is a visual representation of a microcontroller board with a grid of red LEDs. Below it is a 'Show console Device' button. The central pane shows a search bar and a category list including Basic, Input, Music, Led, Radio, Loops, Logic, Variables, and Math. The right pane shows the script area with the following code:

```

on start
  serial redirect to USB
  radio set group 23
  show leds

on radio received receivedNumber
  if receivedNumber <= 128 then
    show string "D"
  else
    show string "B"
  serial write line receivedNumber
  
```

Added code to allow transmission via USB.

This takes a few minutes to display but then allows us to track the live data.

Added code to allow send the data.

TASK Solution Code

Visual of data tracking.

The screenshot displays the MakeCode for micro:bit IDE interface. On the left, a virtual micro:bit board is shown with its LEDs. Below it, a button labeled "Show console Device" is highlighted with a purple border. The central workspace contains the following code blocks:

- on start** block containing:
 - `serial redirect to USB`
 - `radio set group 23`
 - `show leds` block with a 5x5 grid of LEDs.
- on radio received receivedNumber** block containing:
 - `show number receivedNumber`
 - `if receivedNumber <= 128 then` block:
 - `show string "D"`
 - `else` block:
 - `show string "B"`
 - `serial write line receivedNumber`

At the bottom left, a purple "Download" button is visible. The bottom right corner shows a file name "Recliver solnpart1" and a file icon.

Reflection

