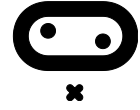


# Coding Support Sheets

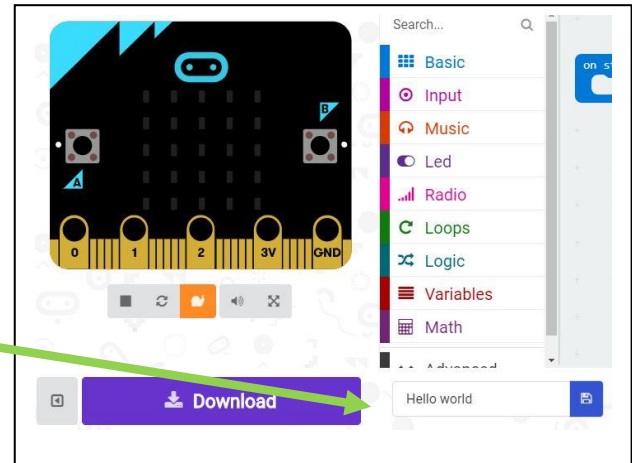
## How to create the code for Hello World



You are going to write a code which will scroll the words 'Hello world' across the LEDs on your BBC micro:bit.

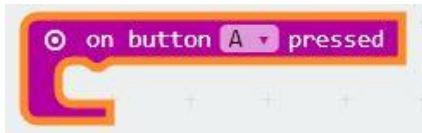
1. Click on 'New project'.

2. Rename your code 'Hello World' by typing 'Hello World' in the box shown



3. You could start the program as soon as the BBC micro:bit is plugged in to a power source but we are going to start it when Button A is pressed. To do this we need to add an **input** to our code so click on **input**.

4. Choose



5. Then go to **Basic** and choose show string. You can then type in the words you want to show when Button A is pressed. You need to type in 'Hello World'.

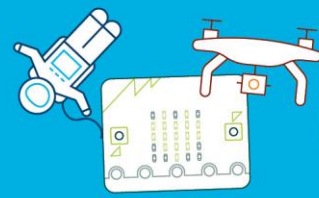


6. Put the two pieces of code together and watch the simulator.

7. Flash it to your micro:bit (see the Coding Support sheet if you are not sure how to do this).

8. Press Button 'A' on your BBC micro:bit and watch what happens.

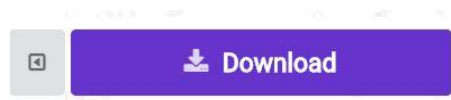




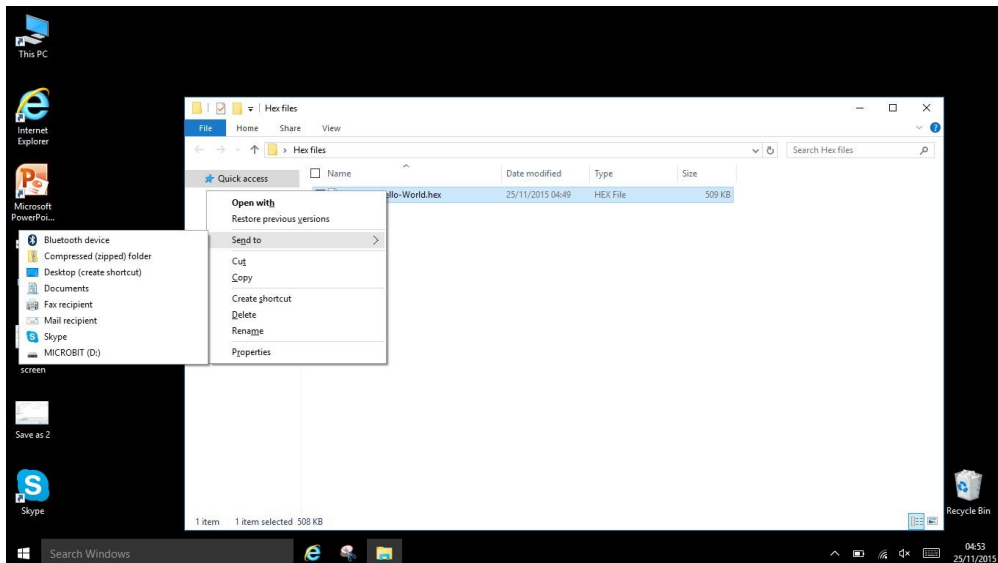
## Flashing your program to your BBC micro:bit

Once you have tested your program on the simulator you need to compile the program so that the BBC micro:bit can understand it.

1. Press 'download'

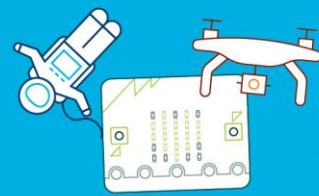


2. Your downloaded file will appear as .hex file at the bottom of your screen. Click on the arrow next to the download and choose 'show in folder'.
3. **RIGHT** click on the file you want to download to your BBC micro:bit and choose 'Send to'. Select 'Microbit' at the bottom of the list.

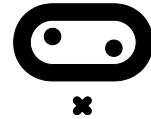


4. You will see the light on the back of your BBC micro:bit flash. When the LED stops flashing, your code has been flashed to the micro:bit and you are ready to run the program.





## How to count on your BBC micro:bit



We can use the BBC micro:bit to count up or down on an input or to set a timer which will count up or count down.

### Example 1 – creating a count code on an input.

In this example pressing button A will count up by 1 and pressing button B will count down by 1. You could use this to keep score in a game or to count the number of people coming entering a building.

```

on button A pressed
  change count by 1
  show number count

on button B pressed
  change count by -1
  show number count
  
```

You can change the input on when the BBC micro:bit adds to the score or when it takes a point away.

You can also change the number at which the count up starts.

### Example 2 – A count up code.

This code will count from 0 to 9 when you press button A. You could use it to count up to another event such as a buzzer sounding.

```

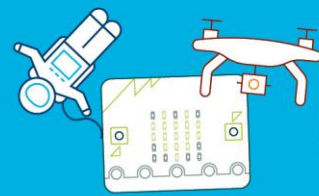
on button A pressed
  for index from 0 to 9
  do
    show number index
  
```

You can change the input for when this count up starts or change the number it will count to.

You can also change the speed at which the number scrolls across the screen.

### Example 3 – A count down code.

This code will count from 9 to 0 when you press button A. You could use it to count down to another event such as an animation.



```

on button A pressed
  for index from 0 to 9
    do
      show number 9 - index
  
```

You can change the input for when this count up starts or change the number it will count to.

You can also change the speed at which the number scrolls across the screen.

## How to use the BBC micro:bit rings

There are two ways you can connect external devices to your BBC micro:bit, the 'rings' and the 'pins'. There are three rings at the bottom of your BBC micro:bit, labelled 0, 1 and 2 whilst the pins are along the gold strip. As it is best to connect external components to the pins using an edge connector we are going to focus on using the rings.

External components, such as sensors, LEDs and buzzers, can be connected to each of the rings. We are going to explore a few examples of this.

### Example 1 – touching the rings.

We do not need to attach anything electronic to the rings, we can just attach ourselves! Try this out for yourself.

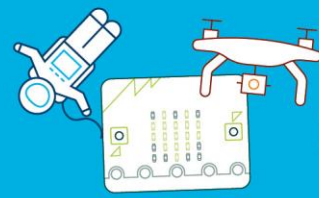
```

on pin P0 pressed
  show icon [Smiley Face]
  pause (ms) 500
  clear screen
  
```

In this example the input is ourselves. When you touch Ring '0' (known as P0) and the ground button (GND) the smiley face will show, when we let go it disappears.







**Example 2** – connecting a buzzer or an LED to the BBC micro:bit.

You can connect a buzzer or an LED to the rings to make a sound or light up if a certain condition is met.

```

forever
  if (acceleration (mg) x < -400)
  then digital write pin P0 to 1
  else digital write pin P0 to 0
  
```

In this example the buzzer will sound or the LED will light up if the BBC micro:bit is tilted too far to the left (on the X axis) but will not sound or light up if the micro:bit is held horizontal or tilted in any other direction.

You will need to attach the positive leg of the buzzer or the longer leg of the LED to ring '0' and the negative leg or the shorter leg to GND using crocodile clips.

**How to use the BBC micro:bit temperature sensor**



The BBC micro:bit does not have a dedicated temperature sensor. Instead, the temperature provided is actually the temperature of the silicon die on the main Central Processing Unit (CPU), sometimes referred to the 'brains' of the BBC micro:bit. This processor generally runs cold but you might find it heats up if you give it a lot of work to do.

The sensor can sense changes in temperature very well. Although it might not be absolutely accurate it will give us a relatively accurate reading of the temperature of our surroundings and any changes taking place.

**Example 1** – getting a numerical temperature reading.

```

forever
  show number temperature (°C)
  
```

If you leave the BBC micro:bit on your table the temperature of the room will show on the screen in degrees Celsius.

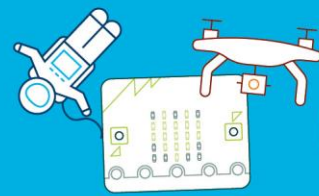
Try holding the BBC micro:bit in your hands to warm it up. What do you notice about the numbers on the screen?

How long does it take to cool down again?

**Example 2** – developing a temperature warning program.

We can change our code to show whether the temperature is too hot or just right.





```

    forever
    if (temperature (°C) > 22)
    then show string "Too Hot"
    else show icon [micro:bit icon]
  
```

Using this code our BBC micro:bit will show a smiley face when the temperature is lower than 22°C but will scroll the words 'Too hot' if the temperature rises above 22°C. Try holding the BBC micro:bit in your hands to see how it changes.

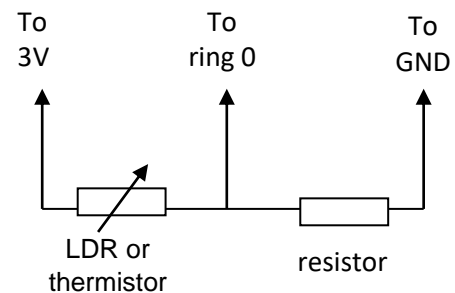
But what if the temperature dropped too low? How could you adapt your code to show this?

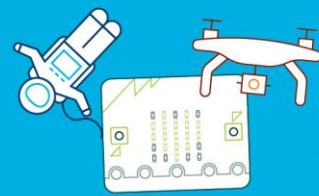
## How to connect a sensor to the BBC micro:bit rings

You can connect external sensors to the BBC micro:bit such as a Light Dependent Resistor (LDR), which senses the level of light, or a thermistor, which senses changes in temperature. The code you write is similar for each as they both work in the same way within a circuit

To connect an LDR or a thermistor;

1. Get three crocodile leads.
2. Twist together one leg of the LDR or thermistor and one leg of the resistor to make a connection as in the photograph
3. Connect the crocodile leads as shown in the diagram. You can use the 0, 1 or 2 ring but this diagram shows the connection between the LDR/thermistor and the resistor connected to the 0 ring (P0).
4. Write your code (see the examples below) and flash it to the BBC micro:bit. You will need to either cover the LDR or hold it with your hand to alter the light level or the temperature.





**Example 1 - light level alert**

```

forever
  set light to (analog read pin P0)
  if (light > 400)
  then
    show icon [smiley face]
  else
    show icon [frowny face]
  
```

In this example the BBC micro:bit will tell us when it is above a set light level by showing a smiley face. If you cover the LDR the frowny face appears to show us the light level has dropped.

One way to use this would be in a car to tell you when you need to turn your lights on (or even turn them on for you).

You can change the light level at which the output would alter.

What happens if you swap round the crocodile clips attached to the GND and 3V rings?

**Example 2 – temperature alert**

```

forever
  set temp to (analog read pin P0)
  if (temp > 325)
  then
    show icon [smiley face]
  else
    show icon [frowny face]
  
```

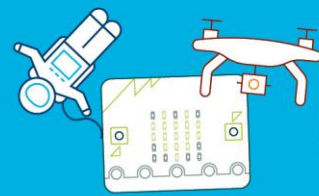
In this example the BBC micro:bit will tell us when the room is too cold by showing the frowny face. As you hold the thermistor the temperature increases and the smiley face will light up letting you know the room is warmer.

If the room is very cold or very warm you may need to try a different number than '> 325' (or more than 325) until you can adjust to the temperature of the room.

What happens if you swap round the crocodile clips attached to the GND and 3V rings?

To sense changes in temperature you can also use the temperature sensor which is built in to the BBC micro:bit. See the 'How to use the BBC micro:bit temperature sensor' guide to help you do this.





# How to use the BBC micro:bit compass

The BBC micro:bit has an inbuilt compass. This can be used to tell the user which direction they are facing. The compass records the angle between 0° and 360° that the BBC micro:bit is at. Three examples from the online simulator are shown below.



**IMPORTANT NOTE:** The first time you use the compass function the words 'Draw a circle' will scroll across your micro:bit. Move the micro:bit around to get a a circle of LEDs. Once you have done this you will see a smiley face. Now you can start using your compass.

## 1. Using the compass heading in degrees

You can tell your BBC micro:bit to display the degrees of the compass angle using the following script



To begin with you will need to tell the BBC micro:bit to show the reading forever. To do this use the Basic command and choose 'show number'.

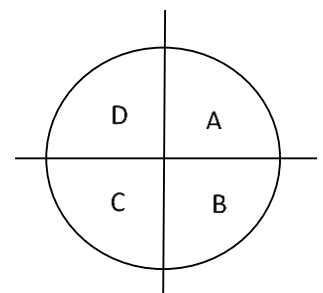
Then choose Input and select compass heading. Move the blocks together and your simulator will start to show the compass reading. Now try it on your BBC micro:bit. Remember you should hold your BBC micro:bit horizontally like a compass.

## 2. Using the compass heading for direction

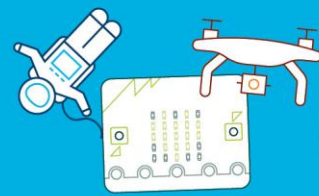
Rather than displaying the compass heading in degrees you can display a symbol or letter e.g. N, S, E, W when the compass heading is between certain values. We can split a circle into sections and then assign each section a symbol or letter, for example:

The code on the next page shows how you could display the letter A depending on which direction the BBC micro:bit is heading.

Notice the use of > (greater than), < (less than) and greater than or equal to and less than or equal to signs in this code.







```
forever
  set degrees to compass heading (°)
  if (degrees ≥ 0 and degrees ≤ 90)
  then show string "A"
  else clear screen
```

This code will only show the letter 'A' when the compass is between 0° and 90°. At all other times the screen will be blank. How could you develop the code to show all the letters or show the points on a compass?

