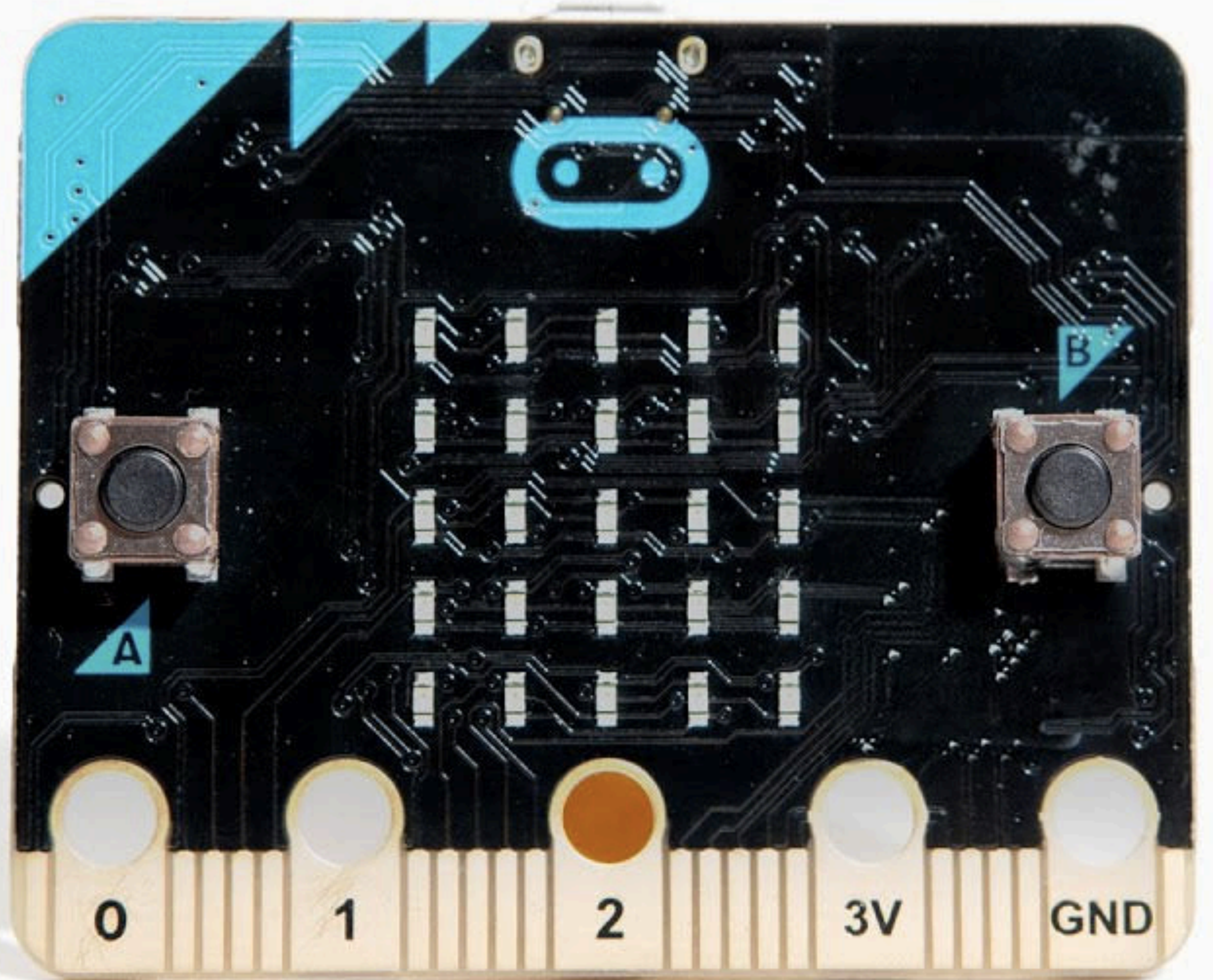




**STARLIGHT**  
**EDUCATION**

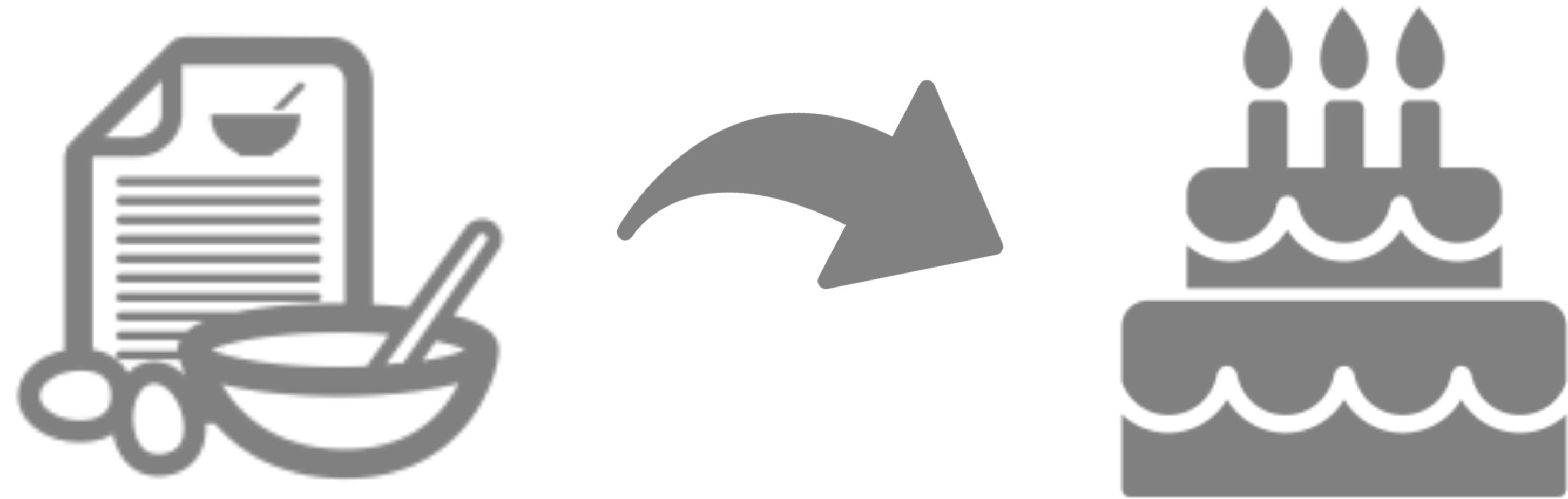


BBC  
  
micro:bit





# What is an algorithm?



# Coding concepts we will cover

- Inputs & outputs
- Variables
- Functions
- Branching (Conditional statements)
- Loops
- Radio communication between micro:bits

# Real world applications

## The Smart Garden





# Real world applications

## Smart Energy Management





# Real world applications

## Smart Homes





# Real world applications

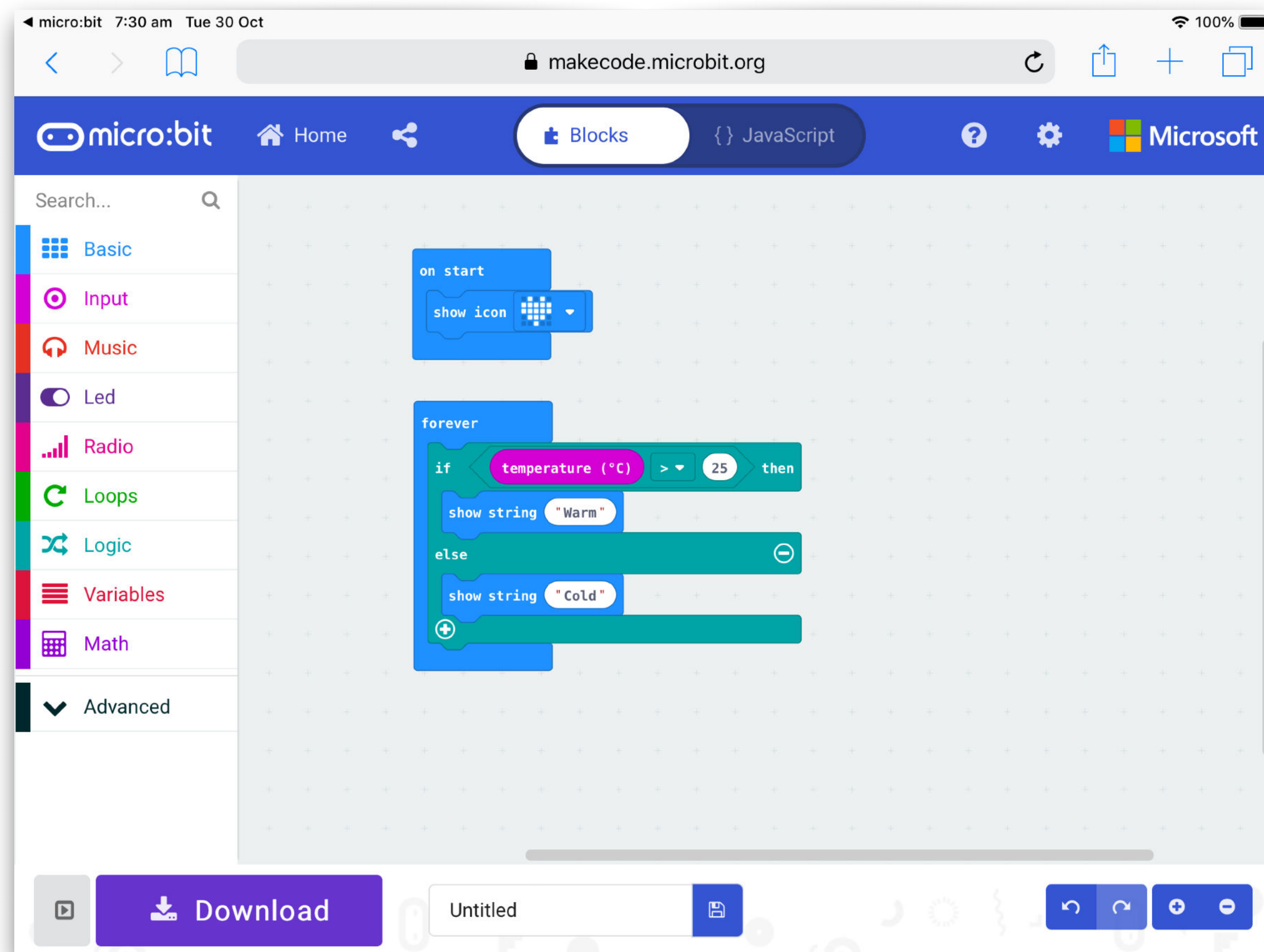
## Smart Farms





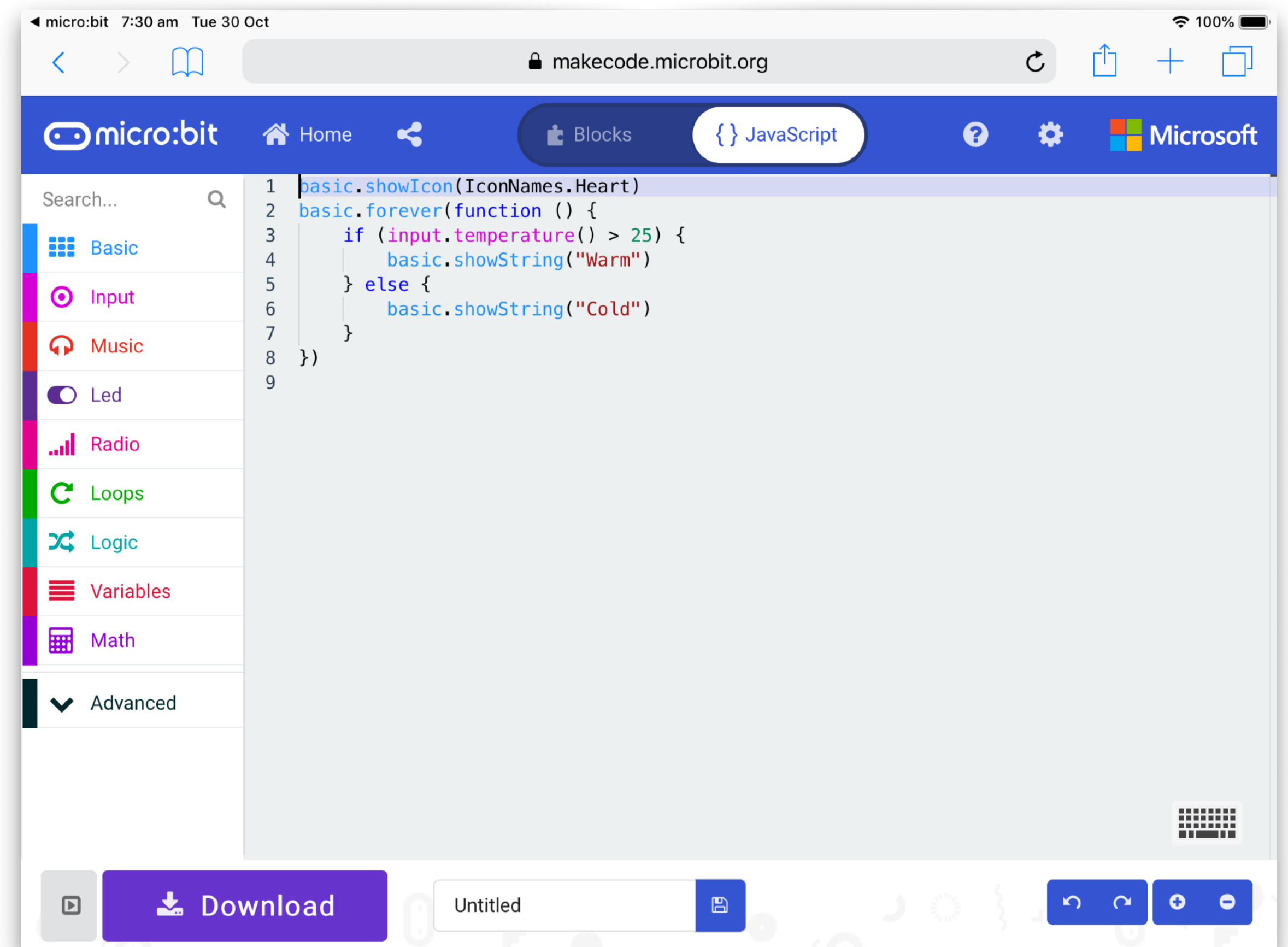
# Introduction to Coding

## Visual Block Coding



The screenshot shows the MakeCode editor interface for a micro:bit. The top navigation bar includes the 'micro:bit' logo, 'Home', 'Blocks' (selected), and 'JavaScript' tabs. A sidebar on the left lists various code blocks: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The main workspace contains a visual block program. It starts with an 'on start' block containing a 'show icon' block with a grid icon. Below this is a 'forever' loop block. Inside the loop, there is an 'if' block with the condition 'temperature (°C) > 25'. The 'then' branch contains a 'show string' block with the text 'Warm'. The 'else' branch contains a 'show string' block with the text 'Cold'.

## Syntax Coding



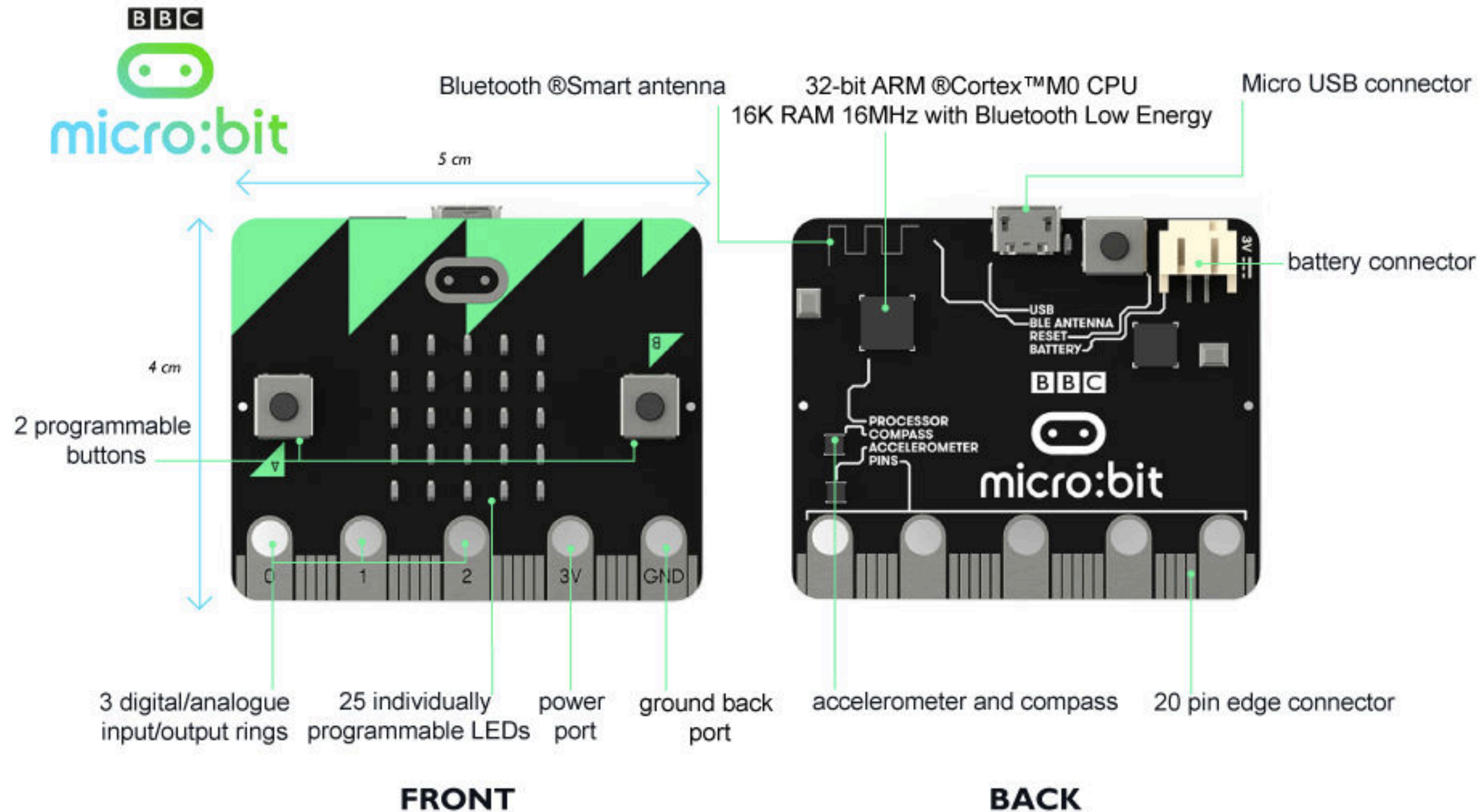
The screenshot shows the MakeCode editor interface for a micro:bit in Syntax Coding mode. The top navigation bar is identical to the visual block mode. The sidebar on the left is also identical. The main workspace shows the same program as the visual block mode, but rendered as text code. The code is as follows:

```
1 basic.showIcon(IconNames.Heart)
2 basic.forever(function () {
3   if (input.temperature() > 25) {
4     basic.showString("Warm")
5   } else {
6     basic.showString("Cold")
7   }
8 })
9
```



# BBC MicroBit

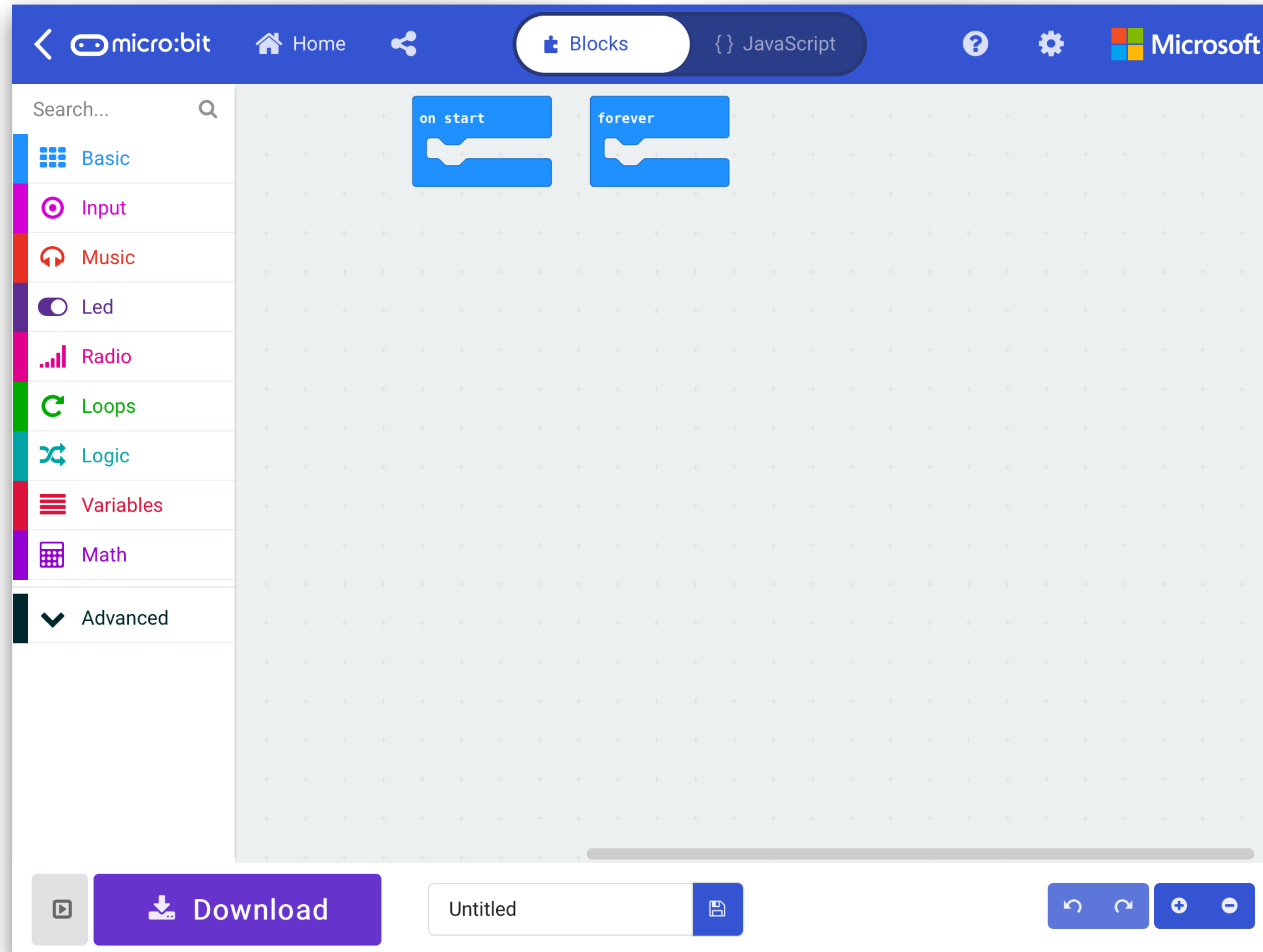
## Hardware Introduction - Interfaces & Sensors





# micro:bit coding environment

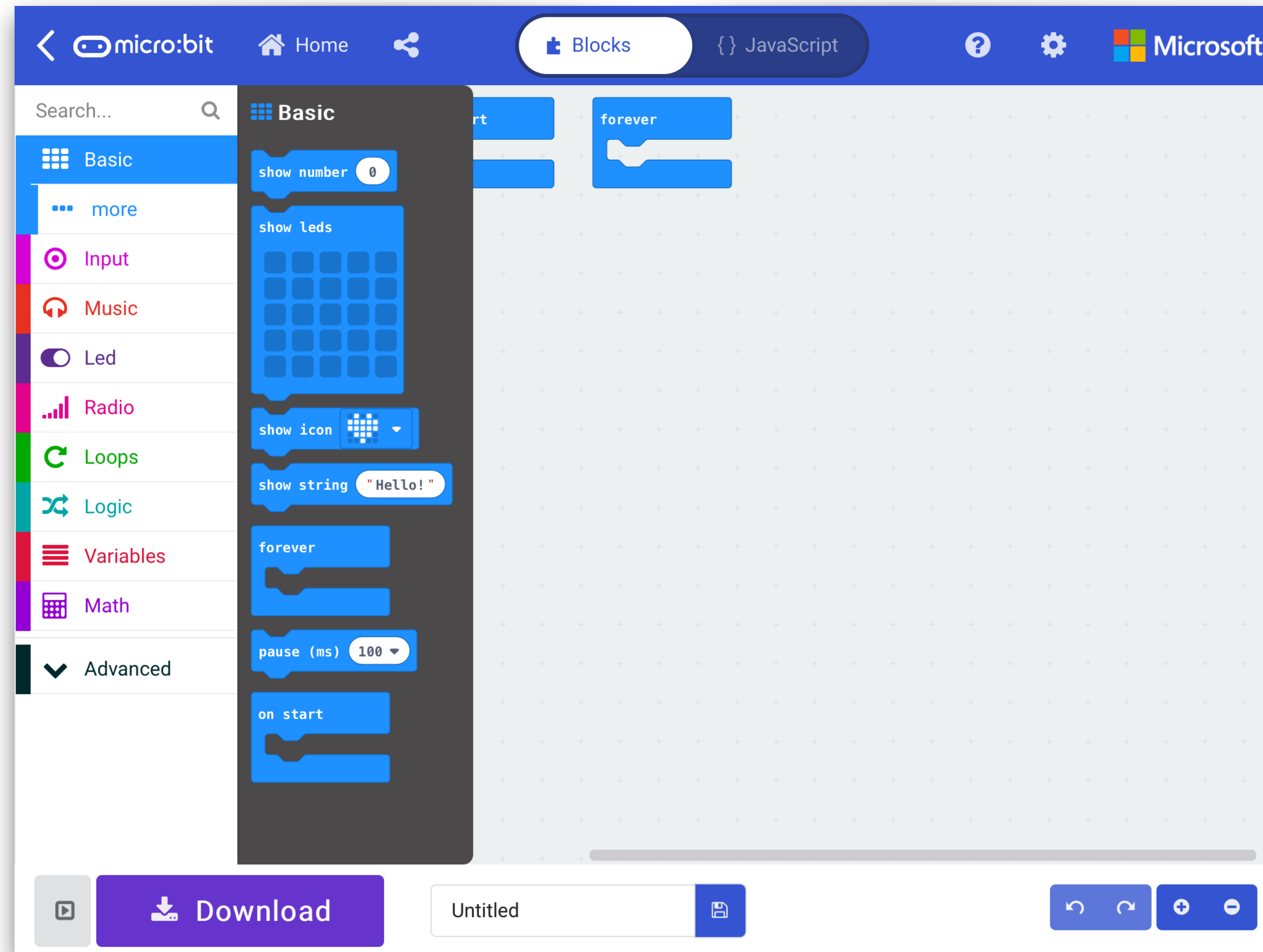
## User Interface





# micro:bit coding environment

## Basic blocks

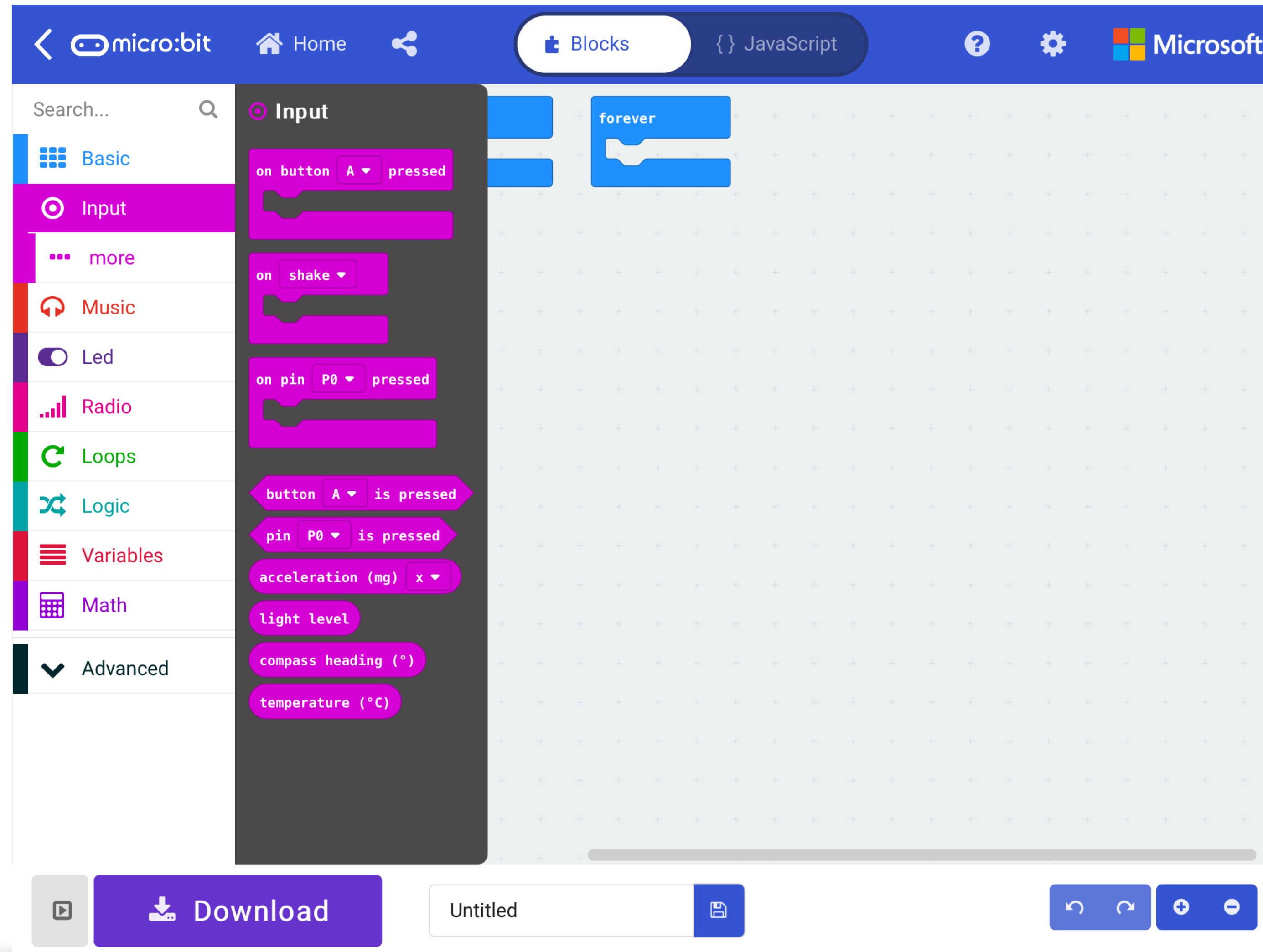


The screenshot displays the micro:bit coding environment interface. At the top, a blue navigation bar includes a back arrow, the 'micro:bit' logo, a 'Home' button, a 'Blocks' button (highlighted), a 'JavaScript' button, a help icon, a settings gear, and the Microsoft logo. Below the navigation bar is a search bar labeled 'Search...'. On the left, a vertical menu lists various block categories: Basic (selected), more, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The 'Basic' category is expanded, showing a list of blocks: 'show number' (with a value of 0), 'show leds' (with a 5x5 grid), 'show icon' (with a dropdown menu), 'show string' (with the text 'Hello!'), 'forever' (with a keyhole icon), 'pause (ms)' (with a value of 100), and 'on start' (with a keyhole icon). The main workspace is a grid where several blocks are already placed: 'show number' (0), 'show leds' (5x5 grid), 'show icon' (dropdown), 'show string' ('Hello!'), 'forever' (keyhole icon), and 'on start' (keyhole icon). At the bottom, there is a 'Download' button, a text field containing 'Untitled', and a set of navigation icons (undo, redo, zoom in, zoom out).



# micro:bit coding environment

## Input blocks

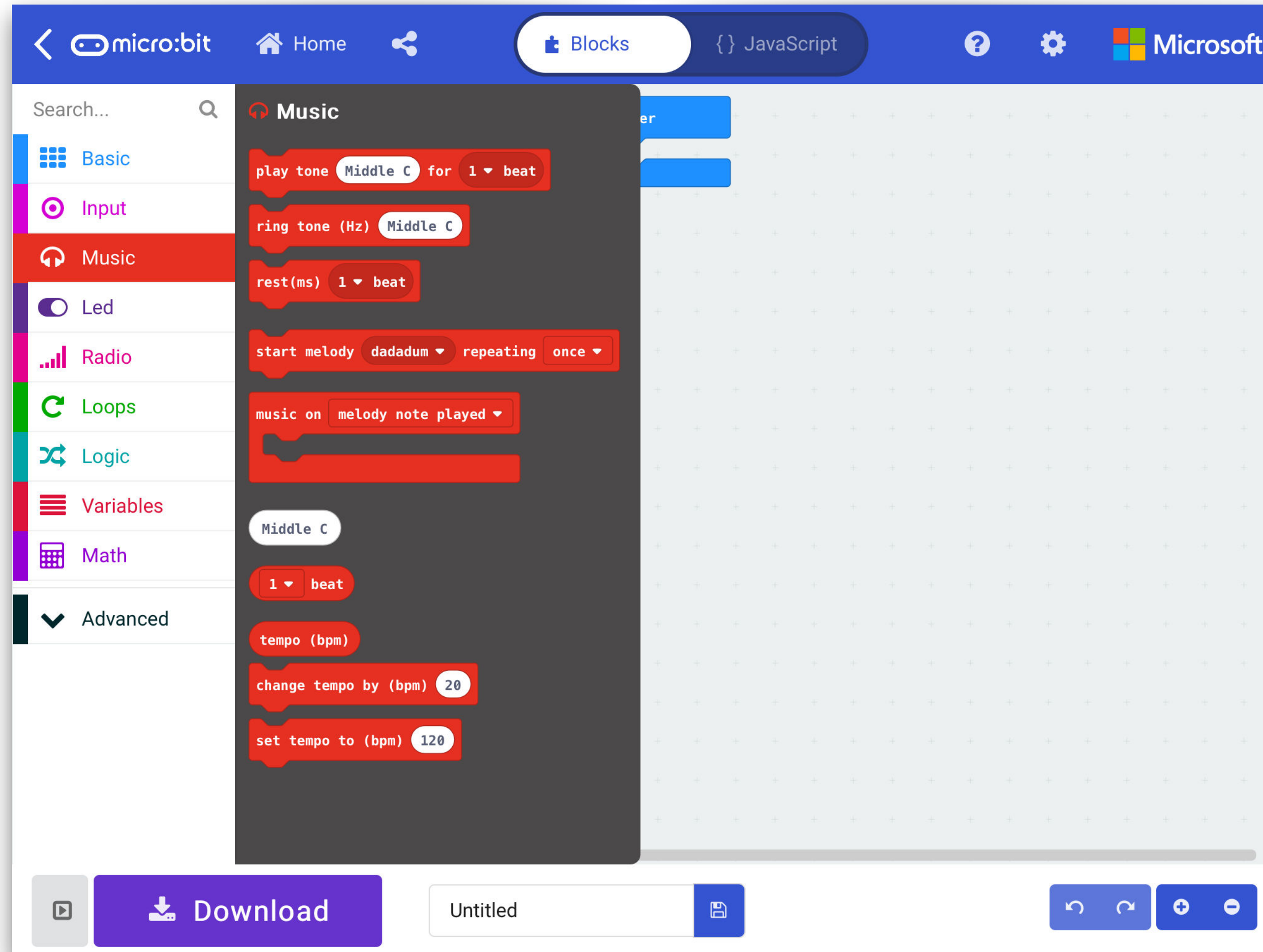


The screenshot displays the micro:bit coding environment interface. At the top, there is a navigation bar with a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" button, a "JavaScript" button, a help icon, a settings icon, and the Microsoft logo. Below the navigation bar is a search bar labeled "Search...". On the left side, there is a vertical menu with categories: Basic, Input (highlighted in pink), more, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The "Input" category is expanded, showing a list of blocks: "on button A pressed", "on shake", "on pin P0 pressed", "button A is pressed", "pin P0 is pressed", "acceleration (mg) x", "light level", "compass heading (°)", and "temperature (°C)". The main workspace is a grid with a "forever" loop block and a "button A is pressed" block. At the bottom, there is a "Download" button, a text field containing "Untitled", and a save icon. On the right side of the bottom bar, there are icons for undo, redo, and zoom in/out.



# micro:bit coding environment

## Music blocks

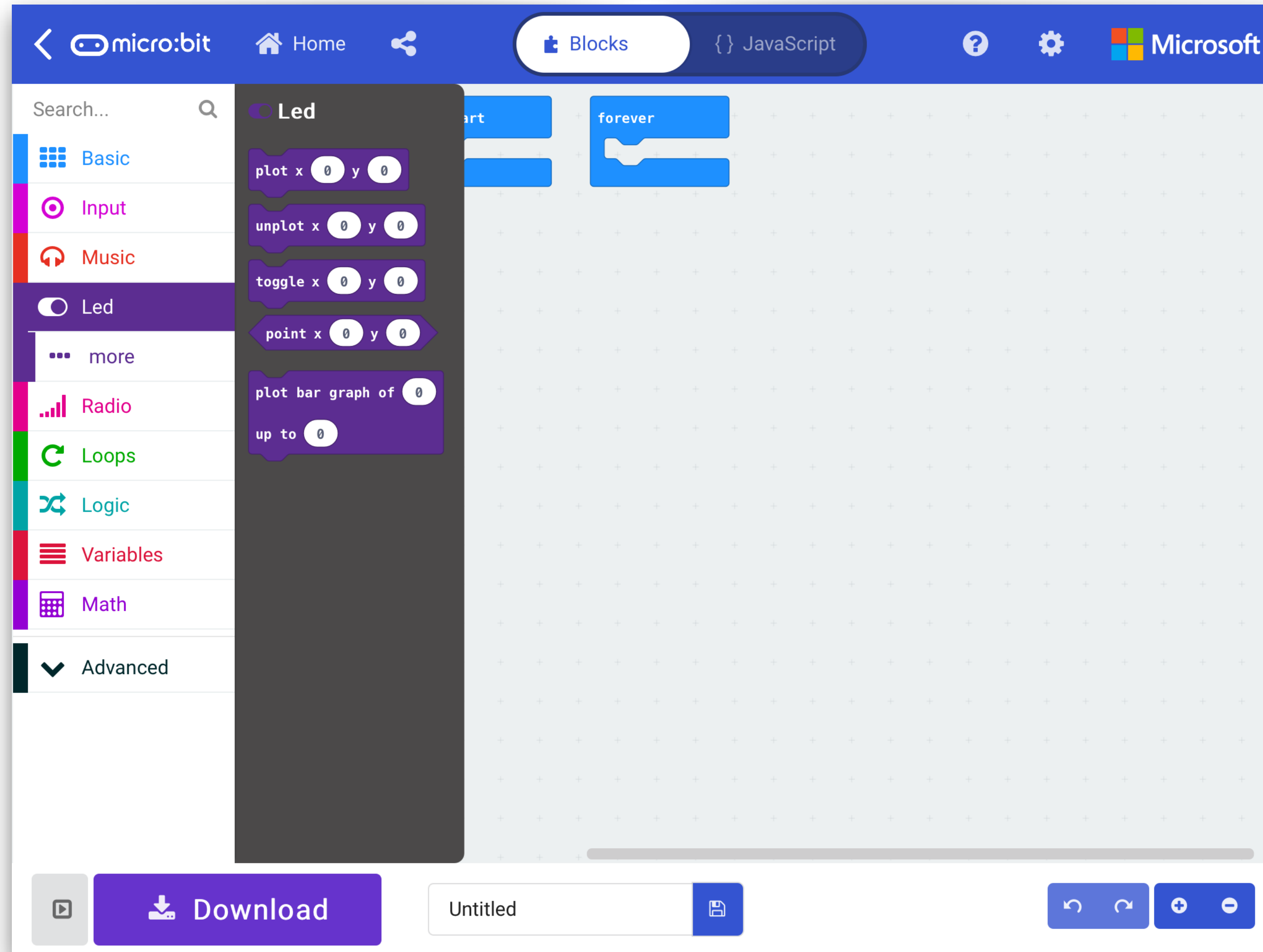


The screenshot displays the micro:bit coding environment interface. At the top, there is a navigation bar with a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" button, a "JavaScript" button, a help icon, a settings icon, and the Microsoft logo. Below the navigation bar is a search bar and a left-hand menu with categories: Basic, Input, Music (highlighted in red), Led, Radio, Loops, Logic, Variables, Math, and Advanced. The main area is divided into a palette of Music blocks on the left and a code editor on the right. The Music blocks palette includes: "play tone" (with "Middle C" and "1 beat" selected), "ring tone (Hz)" (with "Middle C" selected), "rest(ms)" (with "1 beat" selected), "start melody" (with "dadadum" and "repeating" selected), "music on" (with "melody note played" selected), and a "Middle C" block. Below these are "1 beat", "tempo (bpm)", "change tempo by (bpm) 20", and "set tempo to (bpm) 120". The code editor on the right shows a grid with two blue blocks already placed. At the bottom, there is a "Download" button, a text field containing "Untitled", and a save icon. On the far right, there are icons for undo, redo, zoom in, and zoom out.



# micro:bit coding environment

## LED blocks

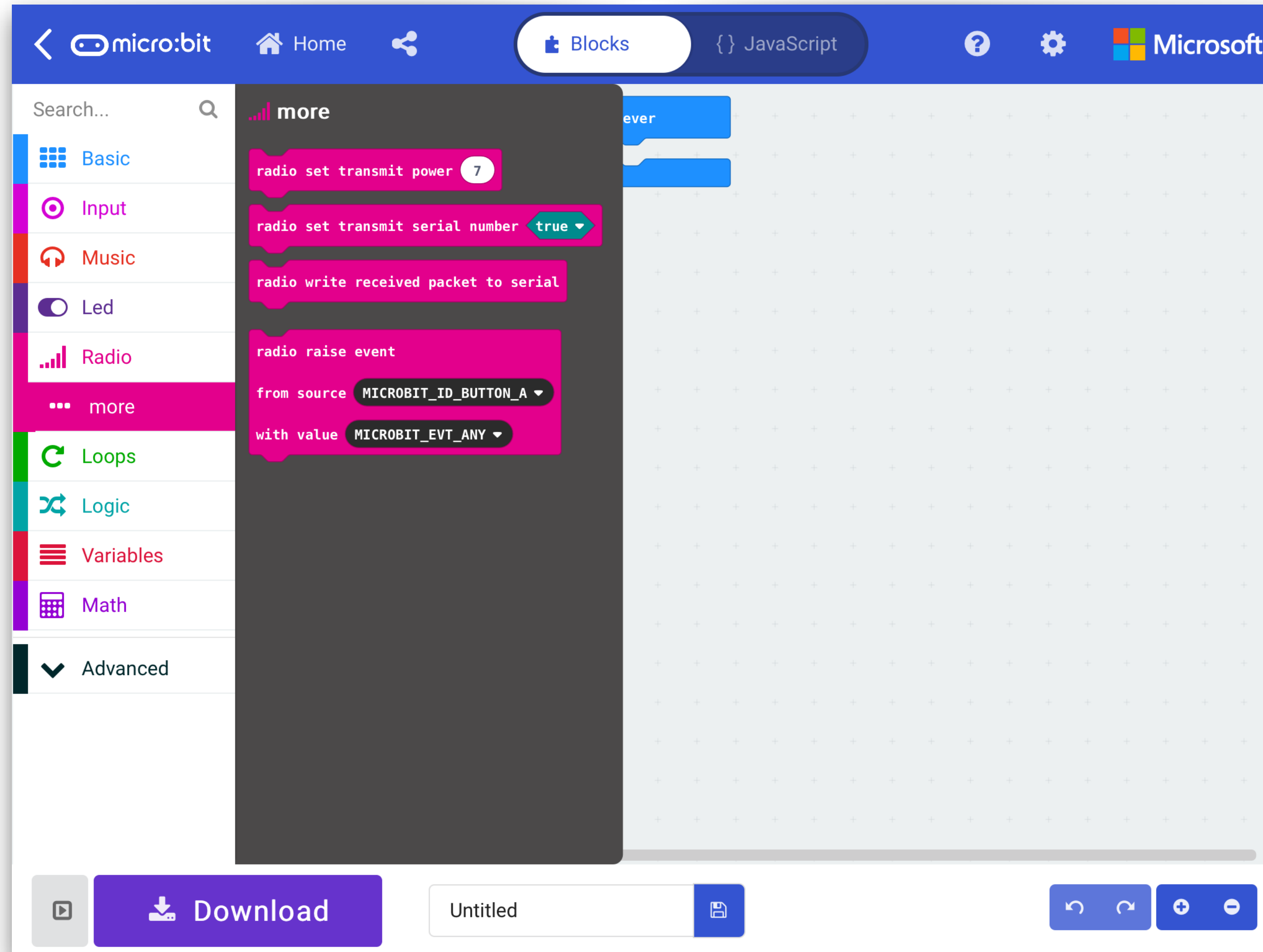


The screenshot displays the micro:bit coding environment interface. At the top, there is a navigation bar with a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" button, a "JavaScript" button, a help icon, a settings icon, and the Microsoft logo. Below the navigation bar is a search bar labeled "Search...". On the left side, there is a vertical menu with various categories: Basic, Input, Music, Led (highlighted in purple), more, Radio, Loops, Logic, Variables, Math, and Advanced. The "Led" category is expanded, showing a list of blocks: "plot x 0 y 0", "unplot x 0 y 0", "toggle x 0 y 0", "point x 0 y 0", "plot bar graph of 0 up to 0", and "plot bar graph of 0 up to 0". The main workspace is a grid with a few blue blocks: "start", "forever", and "plot x 0 y 0". At the bottom, there is a toolbar with a play button, a "Download" button, a text field containing "Untitled", a save icon, and a set of navigation buttons (undo, redo, zoom in, zoom out).



# micro:bit coding environment

## Radio blocks

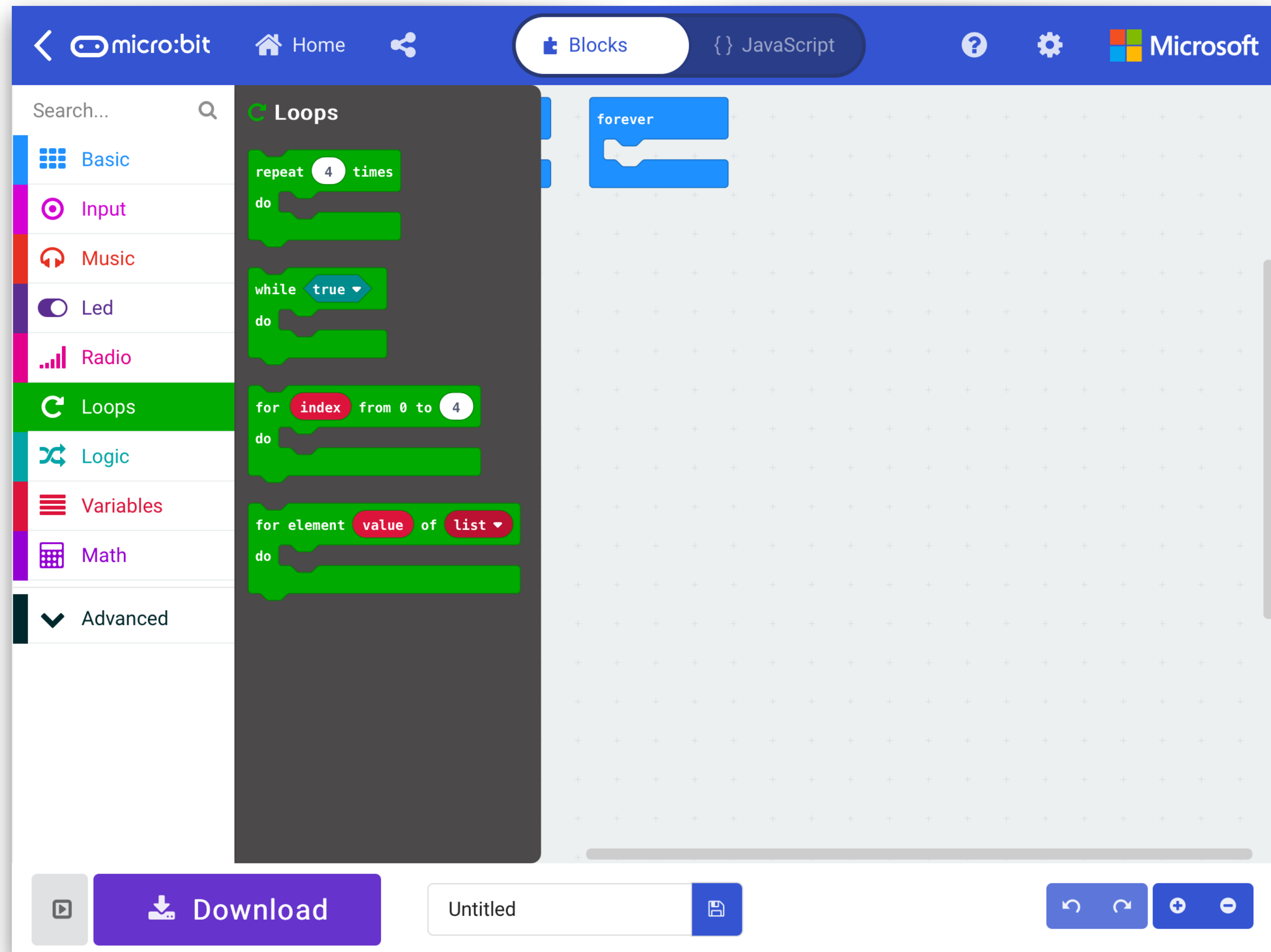


The screenshot displays the micro:bit coding environment interface. The top navigation bar includes a back arrow, the 'micro:bit' logo, 'Home', a share icon, a 'Blocks' tab, a 'JavaScript' tab, a help icon, a settings gear, and the Microsoft logo. On the left, a sidebar lists various block categories: Basic, Input, Music, Led, Radio, more, Loops, Logic, Variables, Math, and Advanced. The 'Radio' category is selected, and a 'more' dropdown menu is open, showing several radio-related blocks: 'radio set transmit power' (set to 7), 'radio set transmit serial number' (set to true), 'radio write received packet to serial', and 'radio raise event'. The 'radio raise event' block is expanded to show 'from source' set to 'MICROBIT\_ID\_BUTTON\_A' and 'with value' set to 'MICROBIT\_EVT\_ANY'. The main workspace is a grid where a few blue blocks are partially visible. At the bottom, there is a 'Download' button, a text field containing 'Untitled', and navigation icons for undo, redo, and zoom.



# micro:bit coding environment

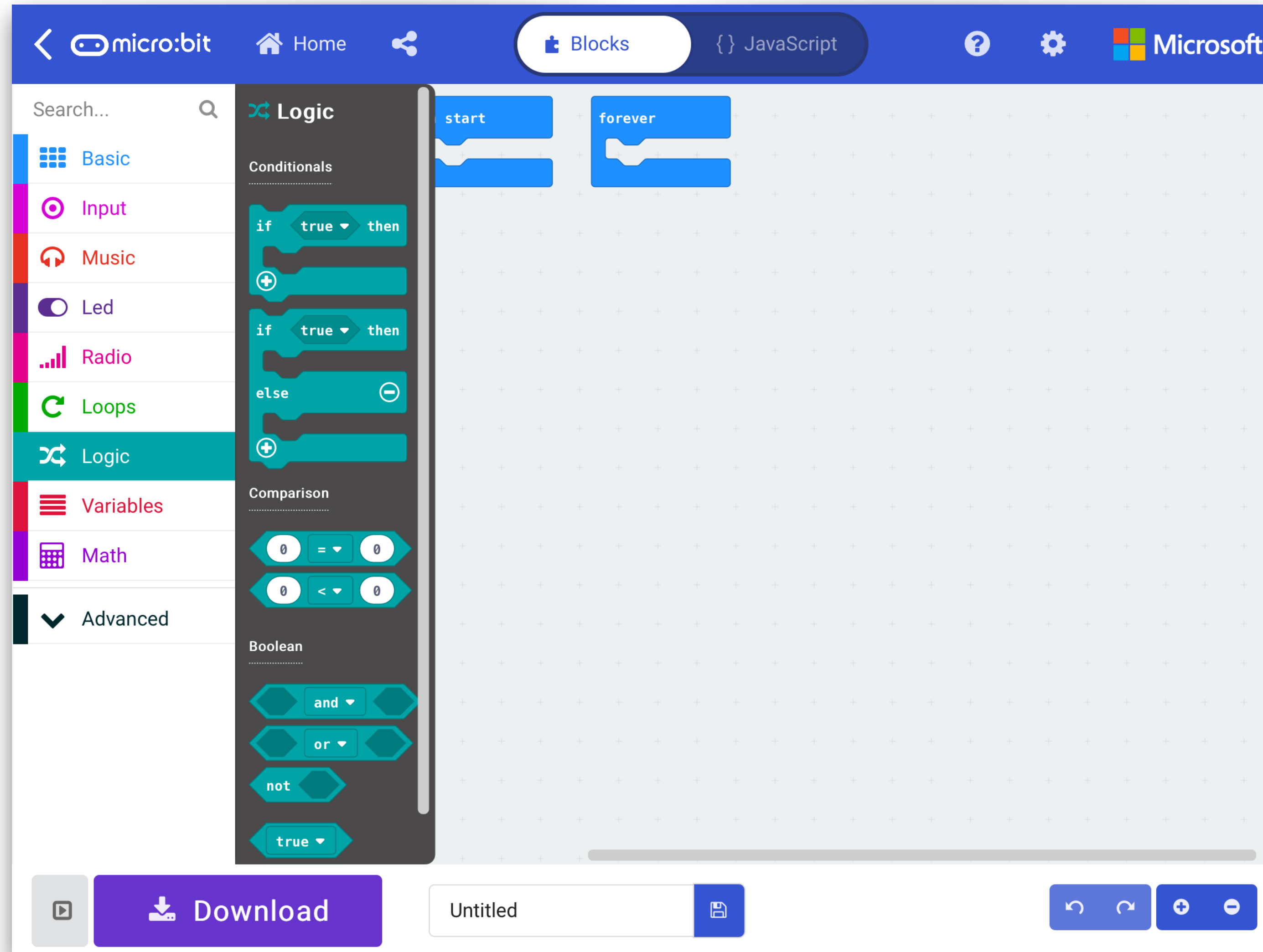
## Loop blocks



The screenshot displays the micro:bit coding environment interface. The top navigation bar includes a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" tab, a "JavaScript" tab, a help icon, a settings icon, and the Microsoft logo. On the left, a sidebar contains a search bar and a list of categories: Basic, Input, Music, Led, Radio, Loops (highlighted in green), Logic, Variables, Math, and Advanced. The "Loops" panel is open, showing four block types: "repeat 4 times", "while true", "for index from 0 to 4", and "for element value of list". The main workspace is a grid where a "forever" loop block is placed. The bottom bar features a "Download" button, a text field containing "Untitled", a save icon, and navigation icons for undo, redo, zoom in, and zoom out.

# micro:bit coding environment

## Logic blocks

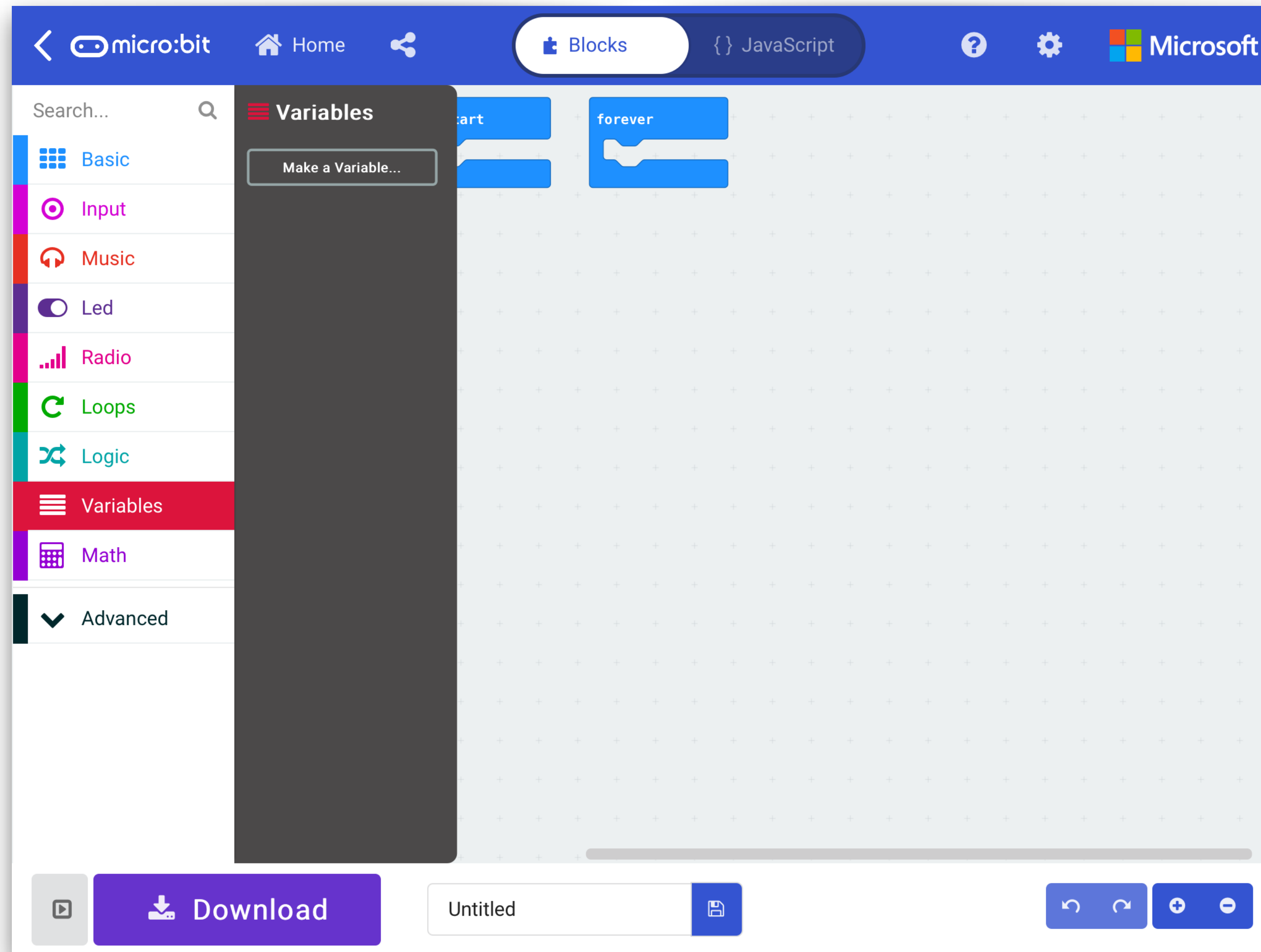


The screenshot displays the micro:bit coding environment interface. At the top, there is a navigation bar with a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" button, a "JavaScript" button, a help icon, a settings gear, and the Microsoft logo. Below the navigation bar is a search bar labeled "Search...". On the left side, there is a vertical menu with categories: Basic, Input, Music, Led, Radio, Loops, Logic (highlighted in teal), Variables, Math, and Advanced. The main workspace is a grid where logic blocks are placed. The "Logic" category is expanded, showing sub-sections: Conditionals (with "if true then" blocks), Comparison (with "=" and "<" comparison blocks), and Boolean (with "and", "or", "not", and "true" blocks). The workspace currently contains a "start" block and a "forever" loop block. At the bottom, there is a "Download" button, a text field containing "Untitled", and a save icon. On the far right, there are undo, redo, and zoom in/out buttons.



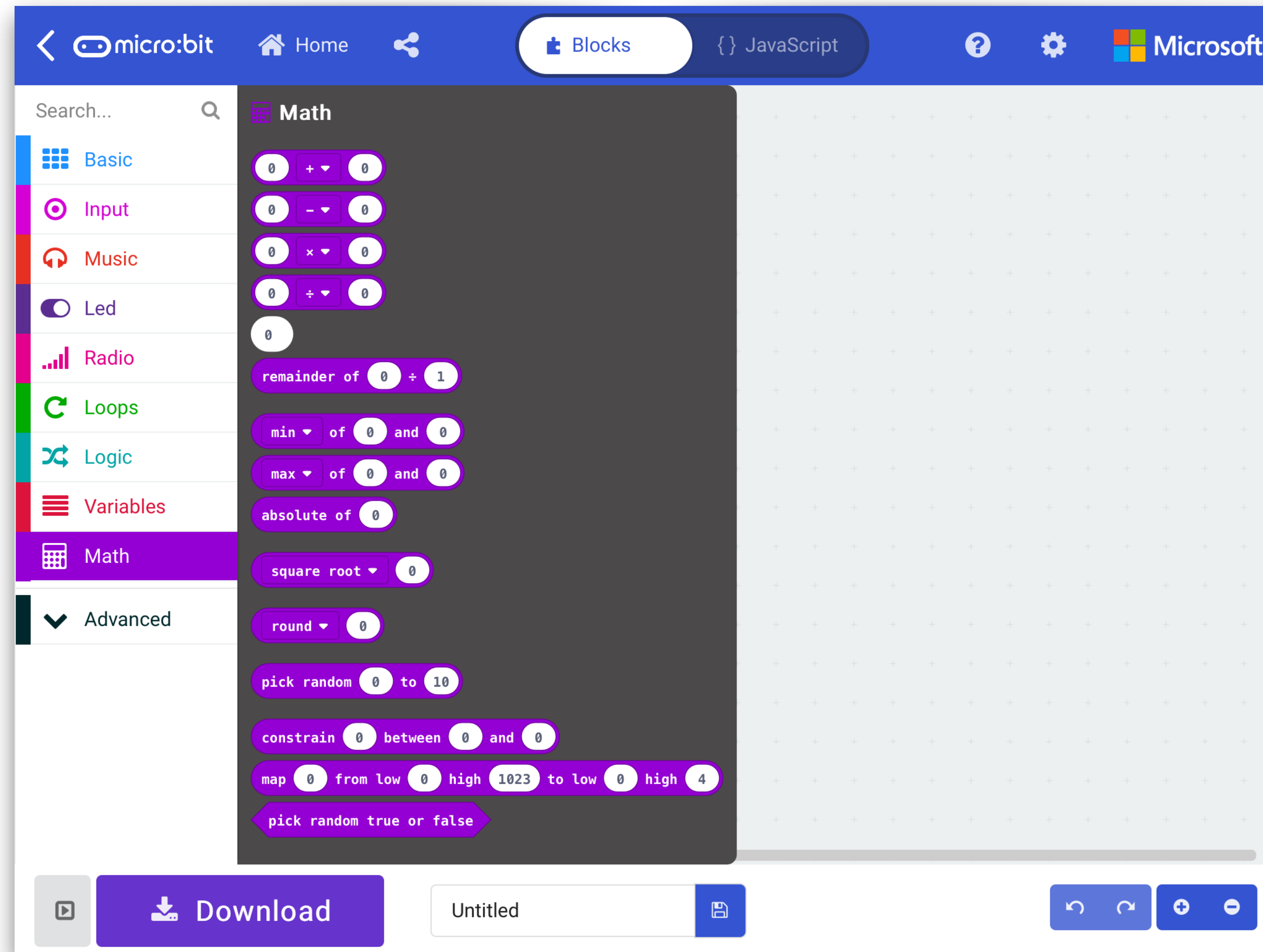
# micro:bit coding environment

## Variables block



# micro:bit coding environment

## Math block



The screenshot displays the micro:bit coding environment interface. At the top, there is a navigation bar with a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" button, a "JavaScript" button, a help icon, a settings icon, and the Microsoft logo. Below the navigation bar is a search bar labeled "Search...". On the left side, there is a vertical menu with various category icons and labels: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math (highlighted in purple), and Advanced. The main area shows a "Math" block palette with various mathematical operations and functions, each with a numeric input field set to "0":

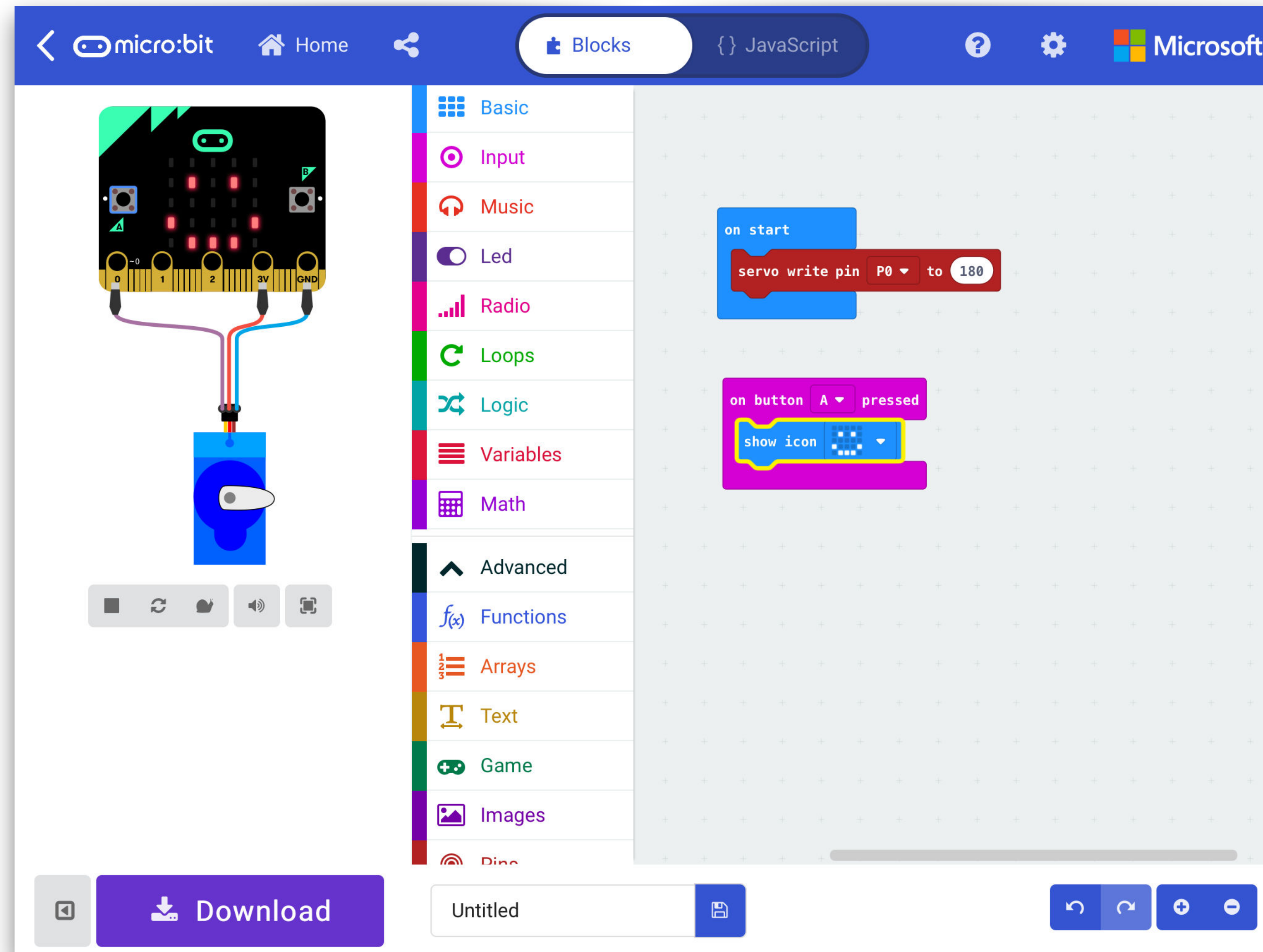
- 0 + 0
- 0 - 0
- 0 × 0
- 0 ÷ 0
- 0
- remainder of 0 ÷ 1
- min of 0 and 0
- max of 0 and 0
- absolute of 0
- square root of 0
- round of 0
- pick random 0 to 10
- constrain 0 between 0 and 0
- map 0 from low 0 high 1023 to low 0 high 4
- pick random true or false

At the bottom of the interface, there is a "Download" button, a text field containing "Untitled", and a set of navigation controls including undo, redo, and zoom in/out buttons.



# BBC micro:bit

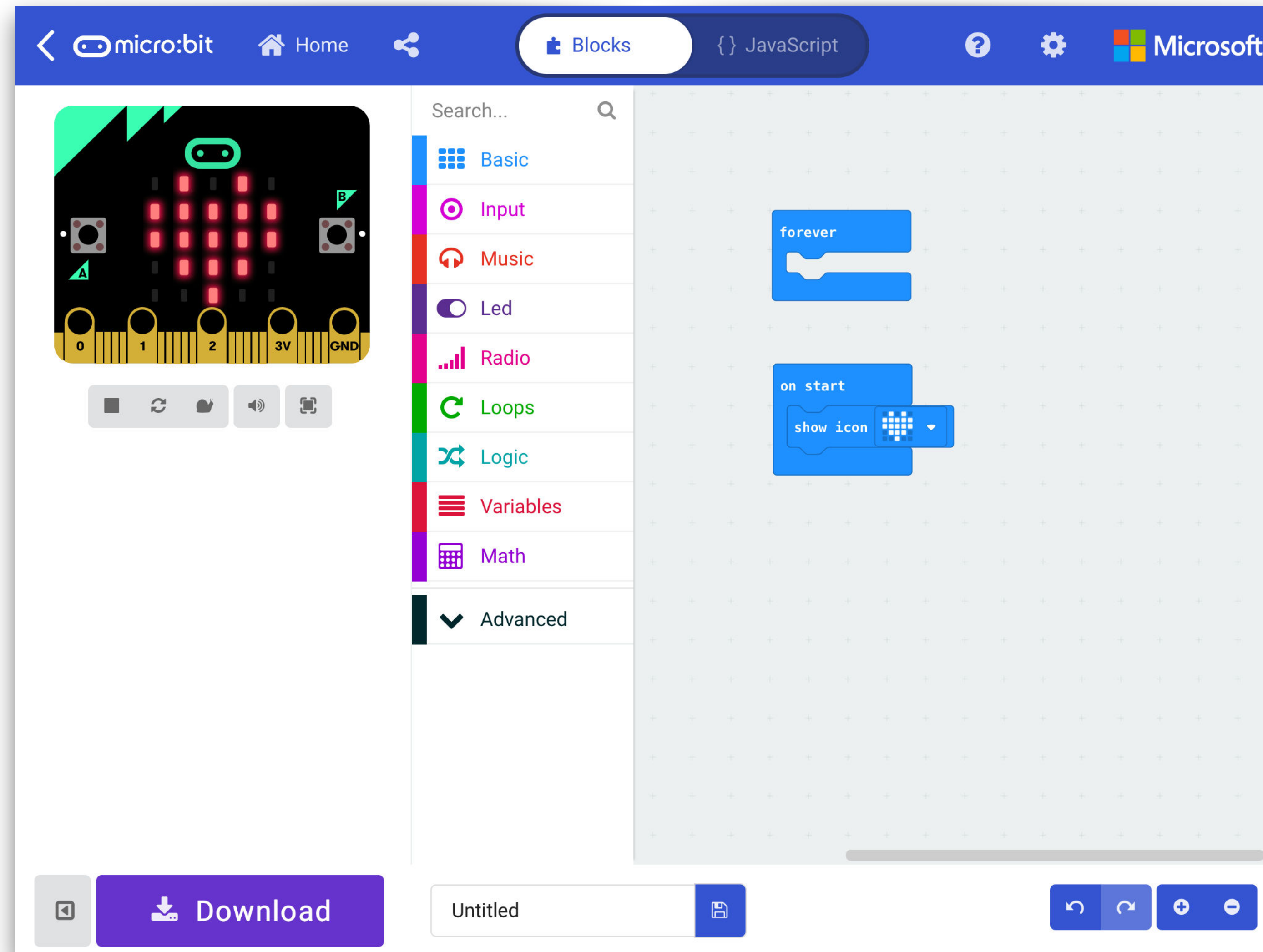
The Simulator (test before downloading)



The screenshot displays the BBC micro:bit simulator interface. At the top, a blue navigation bar includes a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" tab, a "JavaScript" tab, a help icon, a settings gear, and the Microsoft logo. The main workspace is divided into three sections: a hardware view on the left showing a micro:bit board connected to a blue servo motor; a central palette of functional blocks such as Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, Advanced, Functions, Arrays, Text, Game, Images, and Pins; and a right-hand code editor with a grid background. The code editor contains two event-driven blocks: "on start" with a "servo write pin P0 to 180" block, and "on button A pressed" with a "show icon" block. At the bottom, there is a "Download" button, a file name field containing "Untitled", and a set of control buttons for undo, redo, and zoom.

# BBC micro:bit

Download the code (via Bluetooth)

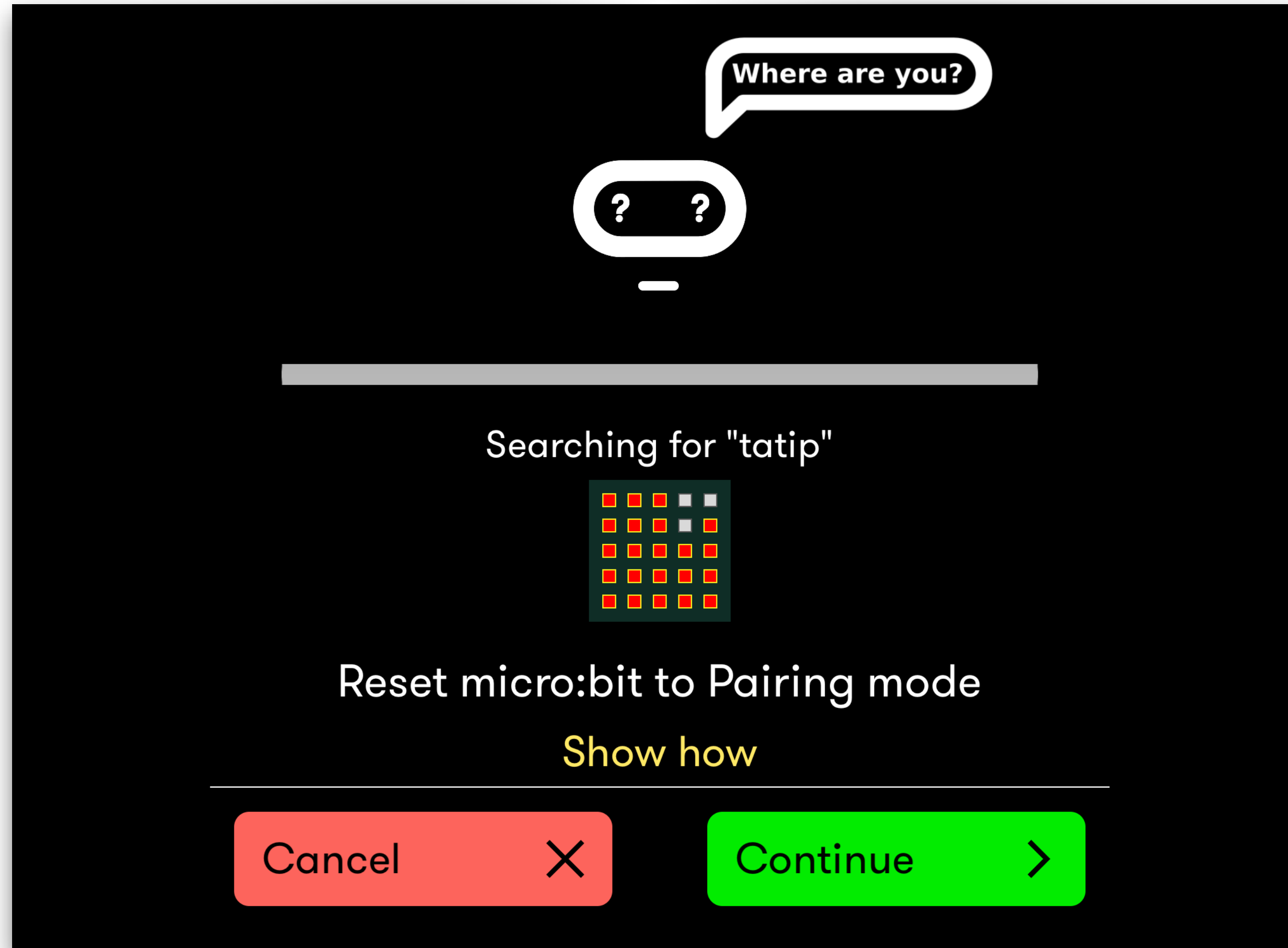


The screenshot shows the BBC micro:bit code editor interface. At the top, there is a navigation bar with a back arrow, "micro:bit", "Home", "Blocks" (selected), "JavaScript", a help icon, a settings icon, and the Microsoft logo. On the left, there is a visual representation of the micro:bit board with its LEDs lit in a pattern. Below it are icons for running, refreshing, and other actions. The central panel is a block palette with a search bar and categories: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The right panel is a workspace with a grid background, containing a "forever" loop block and an "on start" block with a "show icon" block inside. At the bottom, there is a "Download" button, a file name field containing "Untitled", and navigation icons for undo, redo, and zoom.



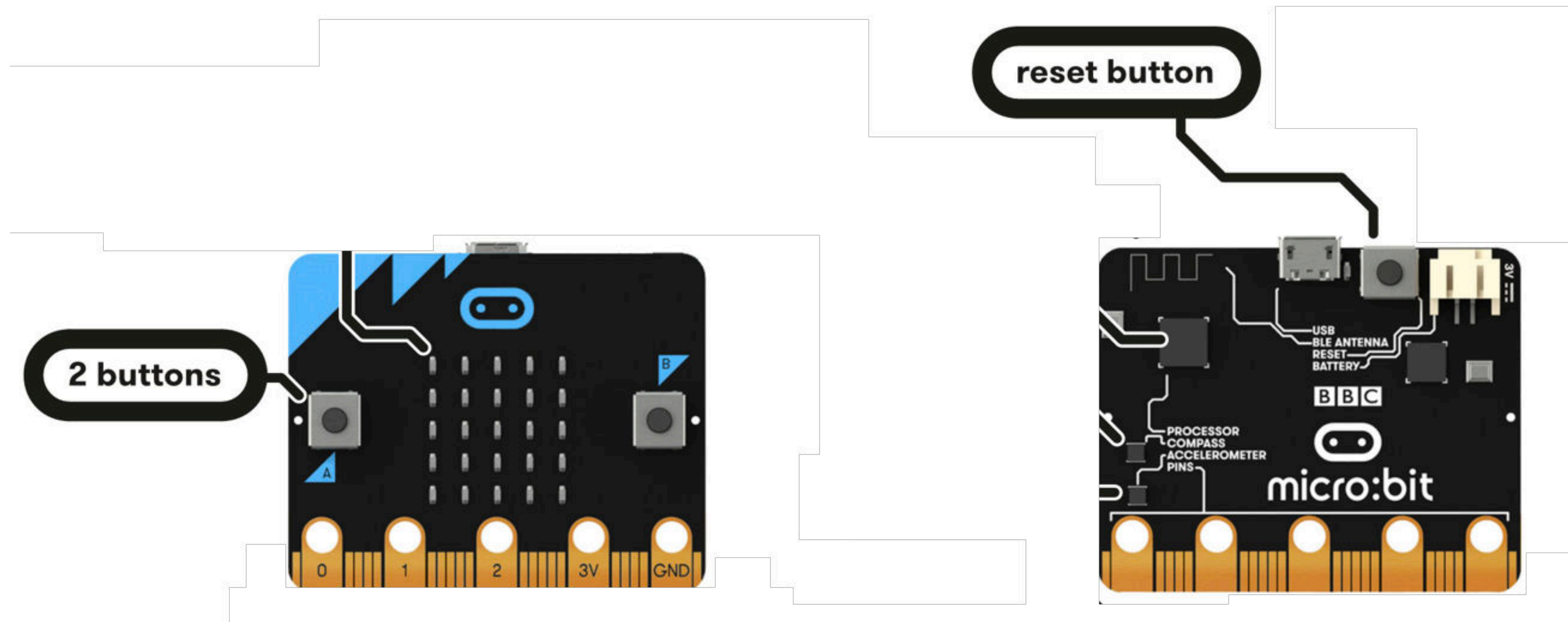
# BBC micro:bit

## Searching for a paired match



# BBC micro:bit

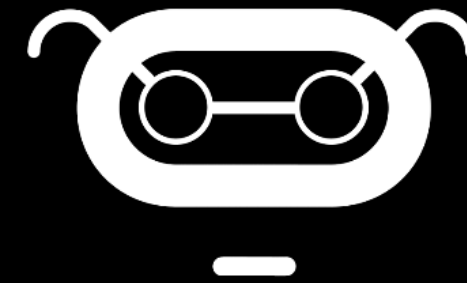
Putting the micro:bit in "pairing mode"





# BBC micro:bit

## Sending code



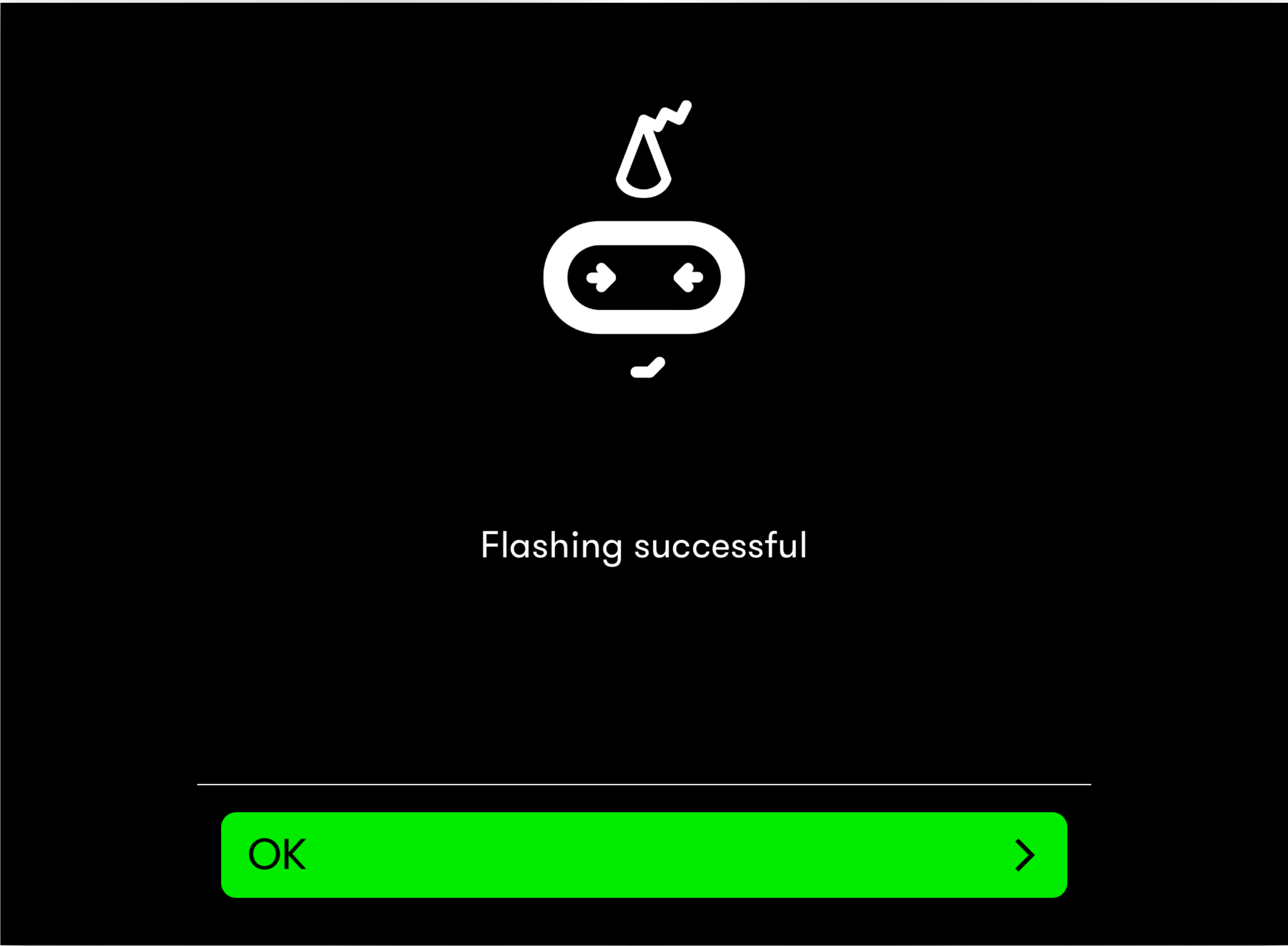
Flashing code to micro:bit

Please do not interact with micro:bit until  
the process is complete

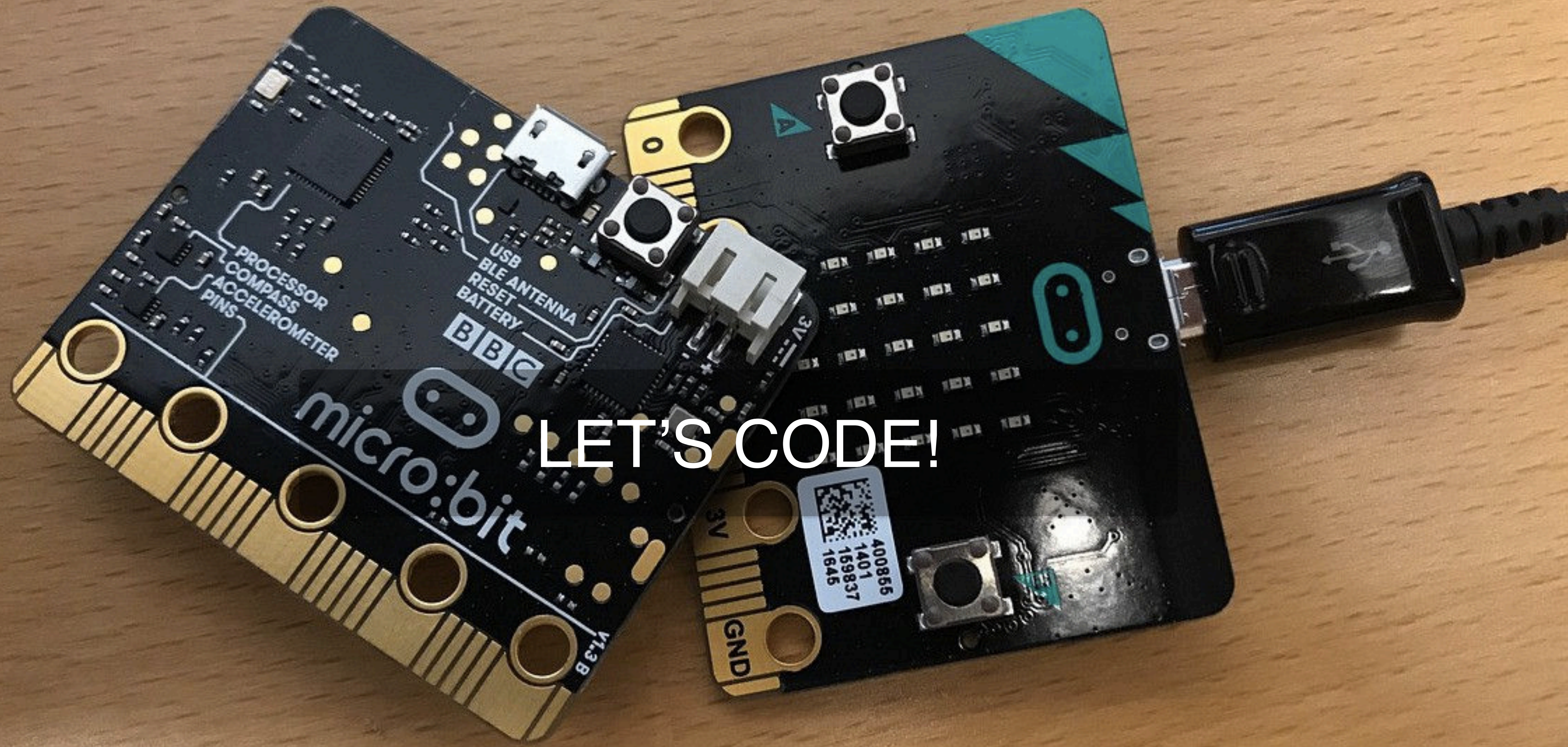
Sending...

# BBC micro:bit

## Sending code





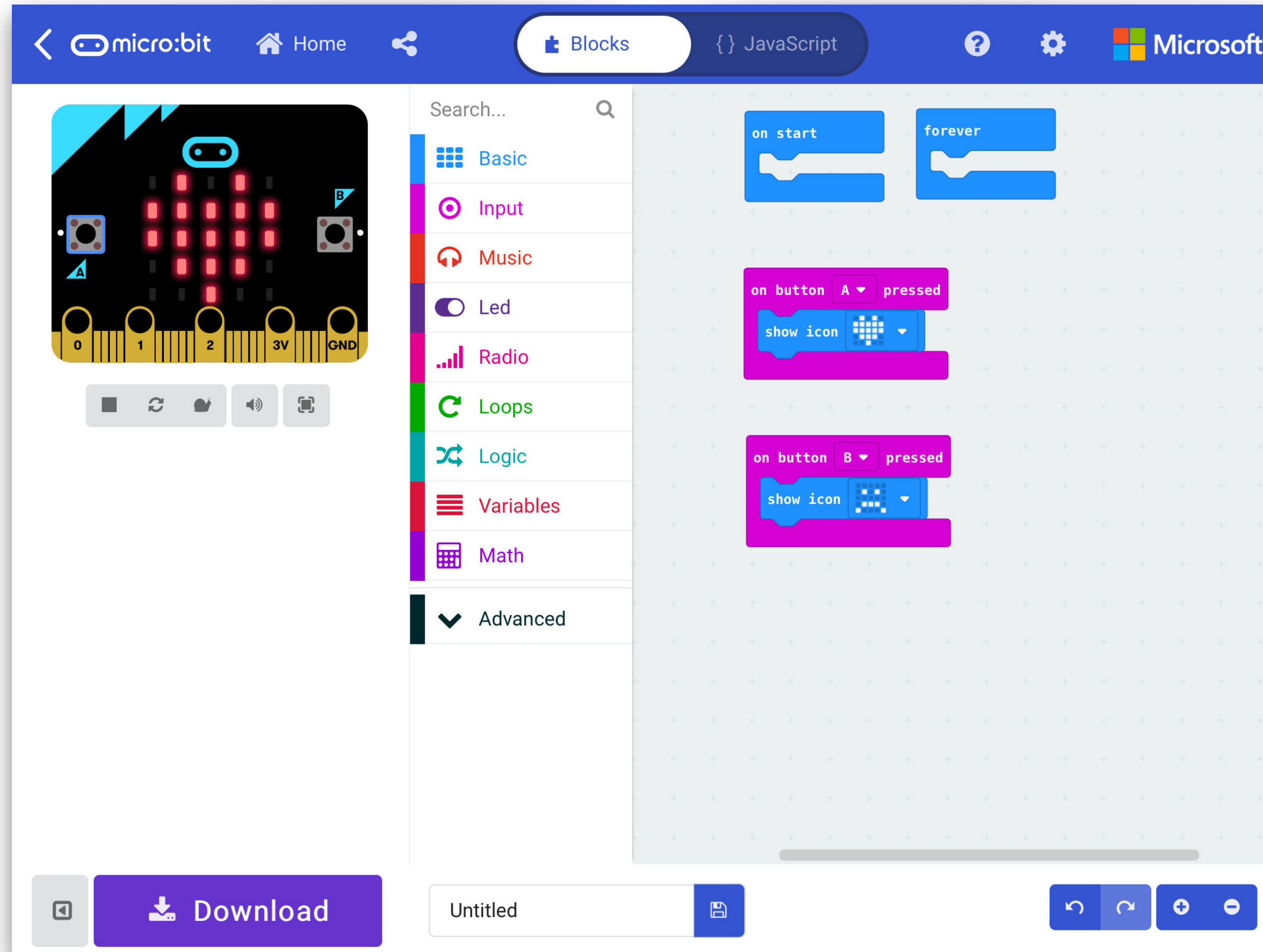


LET'S CODE!



# Challenge 1

Read an input - react with an output (Simulator)



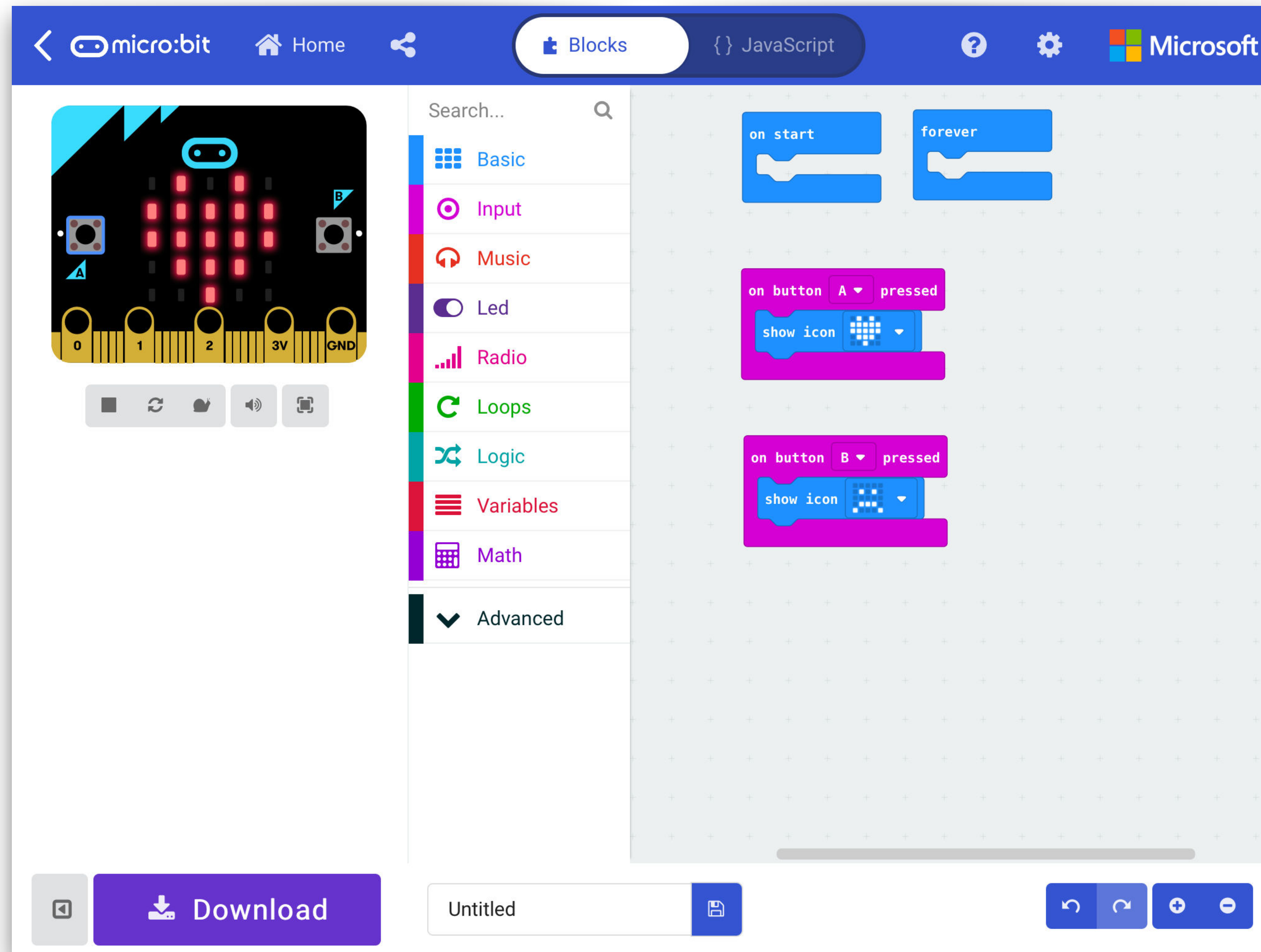
The screenshot displays the Microsoft MakeCode simulator for a micro:bit. The interface is divided into several sections:

- Top Bar:** Includes navigation icons for 'micro:bit', 'Home', and 'Blocks', along with a search bar, a 'JavaScript' tab, and the Microsoft logo.
- Left Panel:** Shows a virtual micro:bit board with a grid of red LEDs. Below the board are control buttons for play, refresh, and volume.
- Central Menu:** A vertical list of block categories: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced.
- Right Panel (Workspace):** A grid-based workspace containing a script. The script starts with an 'on start' block, followed by a 'forever' loop. Inside the loop, there are two 'on button pressed' blocks (A and B), each followed by a 'show icon' block.
- Bottom Bar:** Features a 'Download' button, a file name 'Untitled', and navigation controls for undo, redo, and zoom.



# Challenge 1

Read an input - react with an output (Download code)

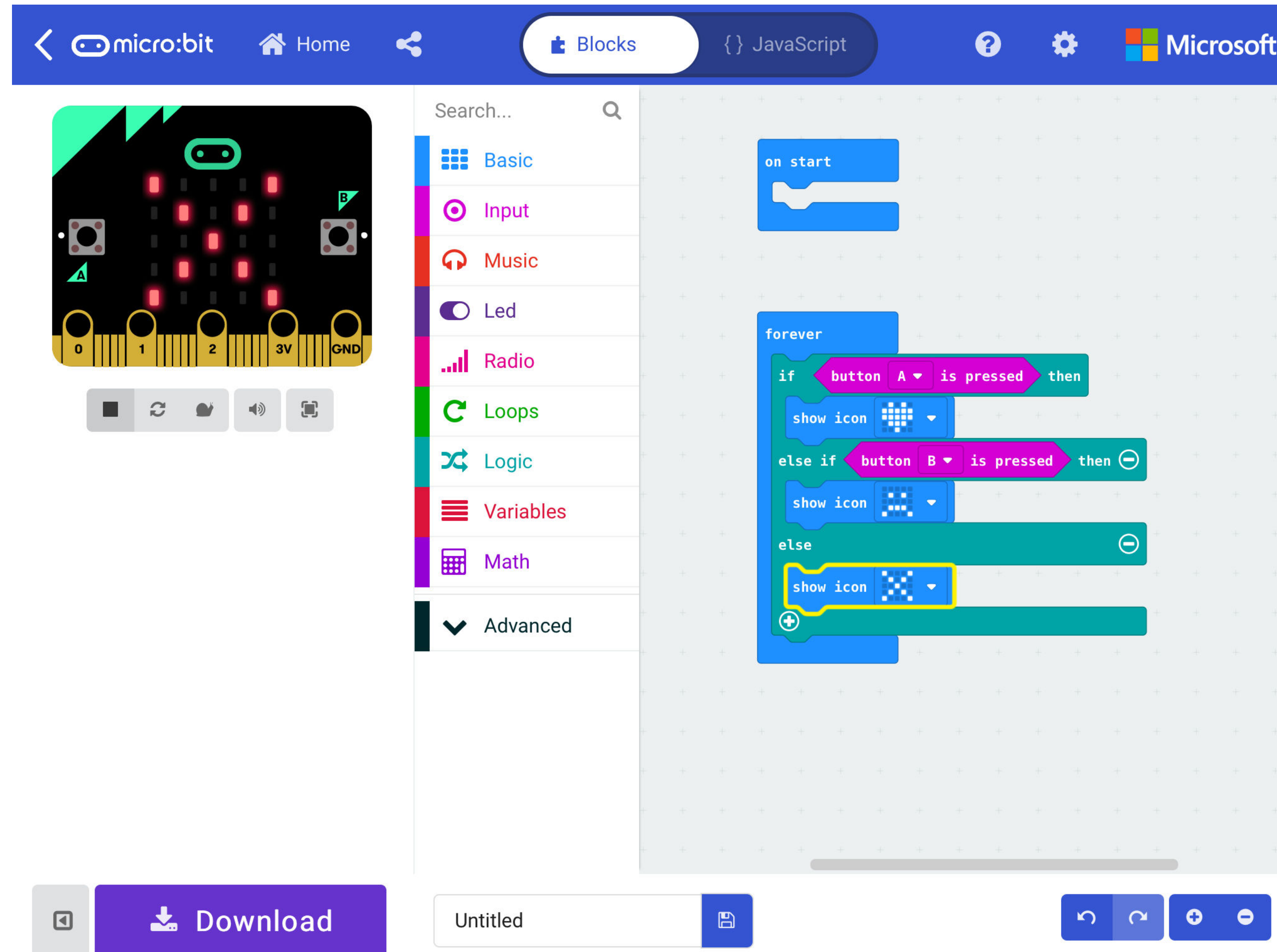


The screenshot displays the Microsoft MakeCode editor for a micro:bit. The interface is divided into several sections:

- Top Bar:** Includes navigation icons for 'micro:bit', 'Home', and 'Blocks', along with a search bar, a 'JavaScript' toggle, and the Microsoft logo.
- Left Panel:** Shows a virtual representation of the micro:bit with a grid of red LEDs. Below it are icons for running, refreshing, and other actions.
- Central Menu:** A vertical list of categories for code blocks: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced.
- Main Workspace:** A grid-based area for assembling code blocks. The current script includes:
  - An 'on start' block.
  - A 'forever' loop block.
  - Inside the loop, two 'on button pressed' blocks (one for button A and one for button B).
  - Each 'on button pressed' block is followed by a 'show icon' block with a grid icon.
- Bottom Bar:** Features a 'Download' button, a file name 'Untitled', and navigation controls (undo, redo, zoom in, zoom out).

## Challenge 2

Read an input - react with an output (Branching)



The screenshot displays the Microsoft MakeCode editor interface for a micro:bit. The top navigation bar includes a back arrow, 'micro:bit', 'Home', 'Blocks', and 'JavaScript' tabs, along with a help icon, settings gear, and the Microsoft logo. On the left, a virtual micro:bit board is shown with a grid of red LEDs. Below the board are control buttons for stop, refresh, play, volume, and share. A central sidebar contains a search bar and a list of categories: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The main workspace on the right contains a script starting with 'on start' and a 'forever' loop. Inside the loop, there is an 'if' statement: 'if button A is pressed then' followed by 'show icon' (grid icon), 'else if button B is pressed then' followed by 'show icon' (dots icon), and 'else' followed by 'show icon' (dots icon). The 'show icon' block in the 'else' branch is highlighted with a yellow box. At the bottom, there is a 'Download' button, a file name 'Untitled', and navigation buttons for undo, redo, and zoom.

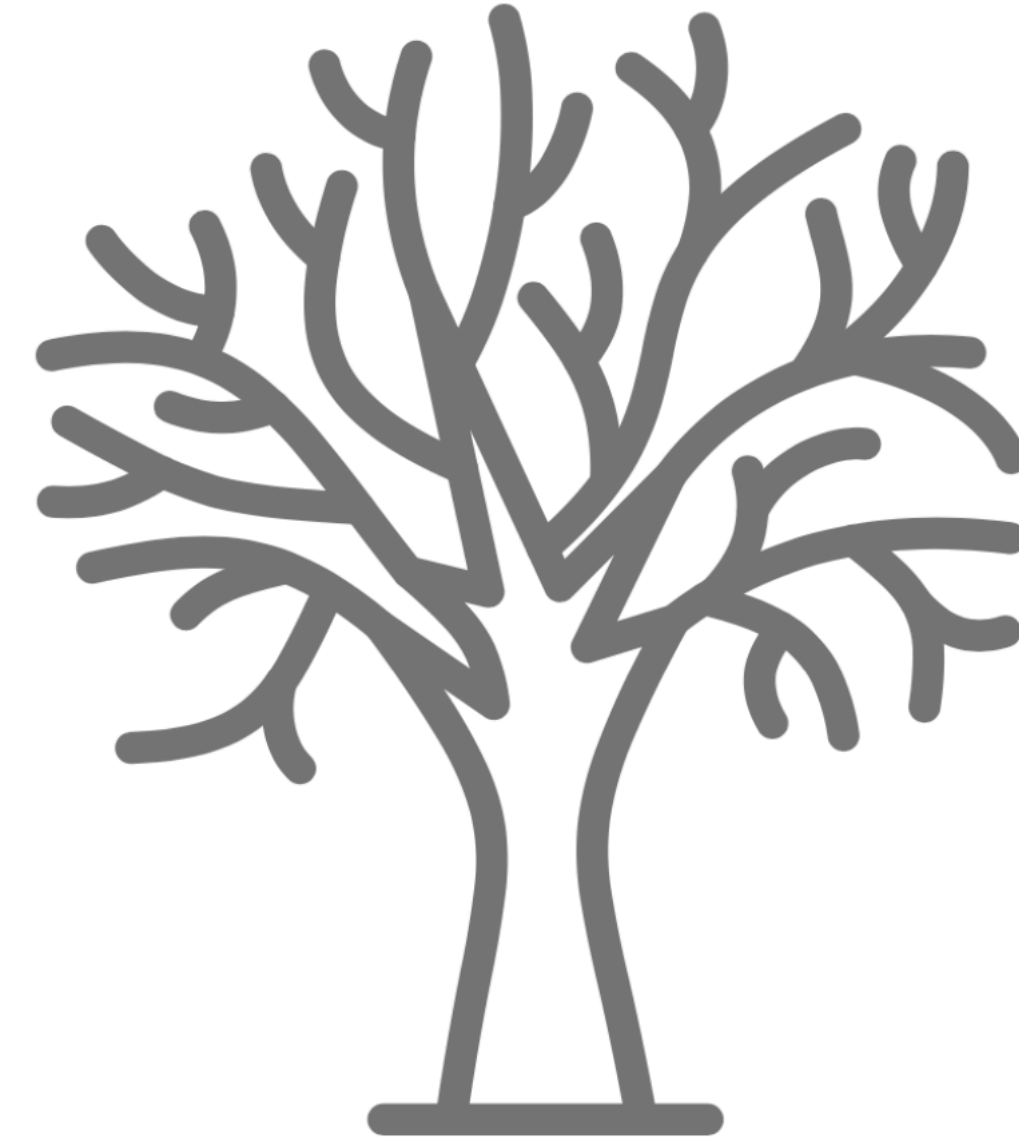


## Challenge 2

### Branching

Branching is a way of writing code to make decisions based on what is happening or what has happened.

To do this we use conditional statements.



### Conditional Statements

Conditional statements are the way computers can make decisions after evaluating conditions.

Conditional statements always have an **if** part, which tells the device what to do when the condition is true. Conditional statements also usually have an **else** part, which tells the device what to do when the condition is false.

## Challenge 2

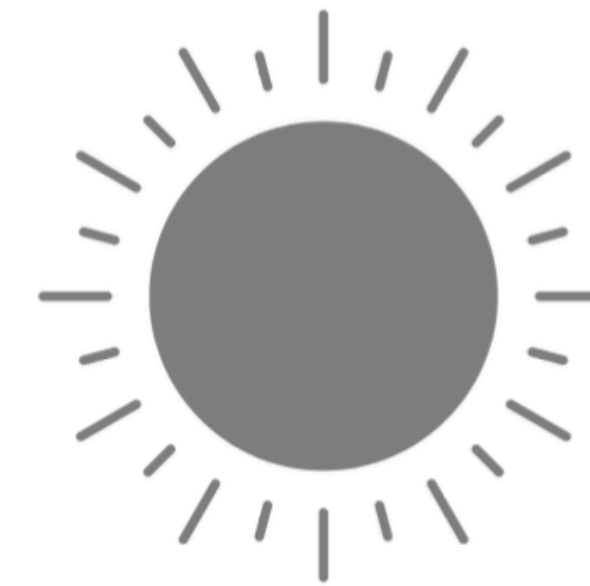
Complete these conditional statements that you use to make decisions every day.

***If you are playing in the sun then...***

***If it is cold outside, then...***

***If your phone is dead, then...***

***If your fridge has no food in it, then...or else...***

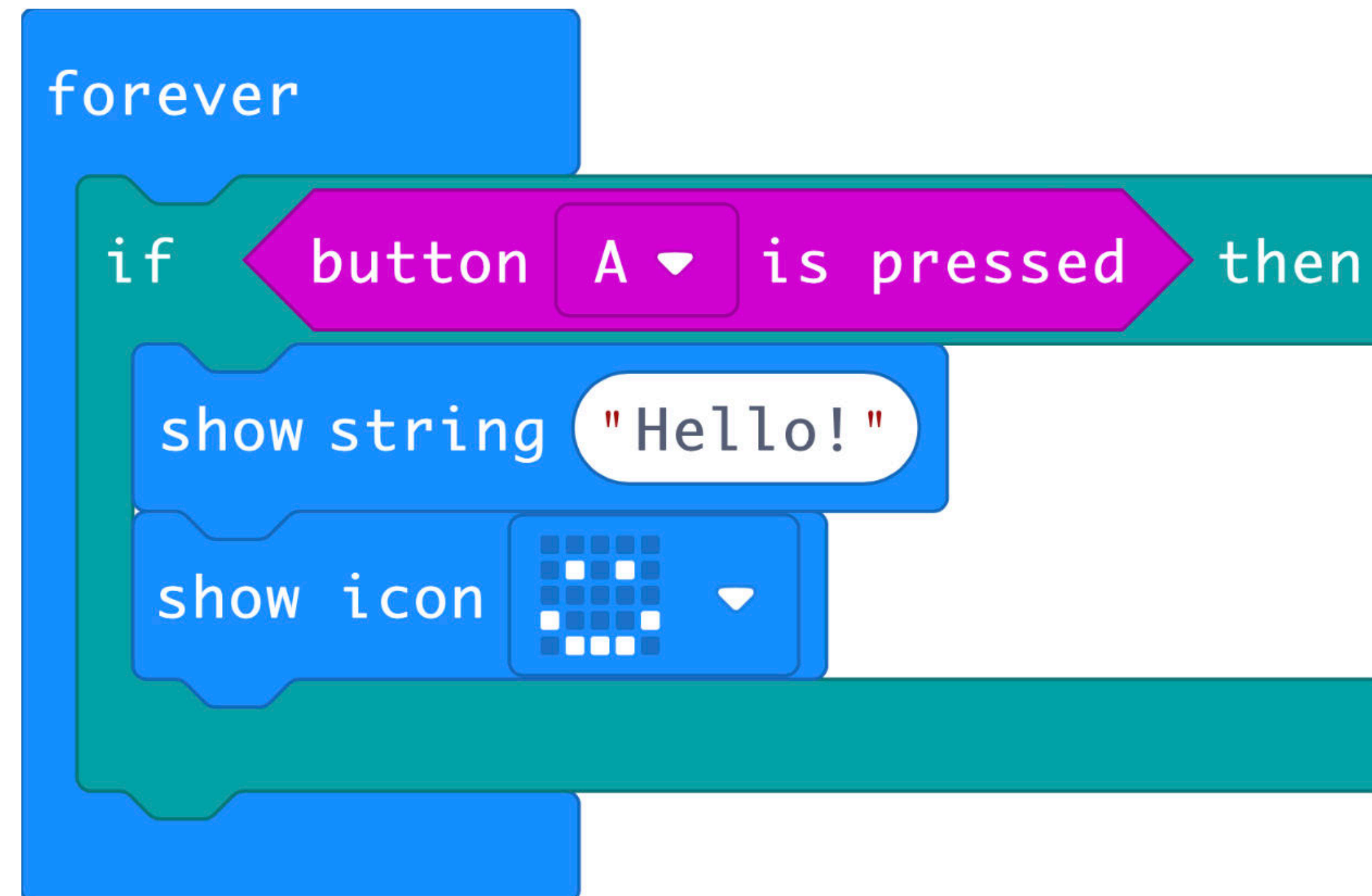




# Challenge 2

## Conditional Statements

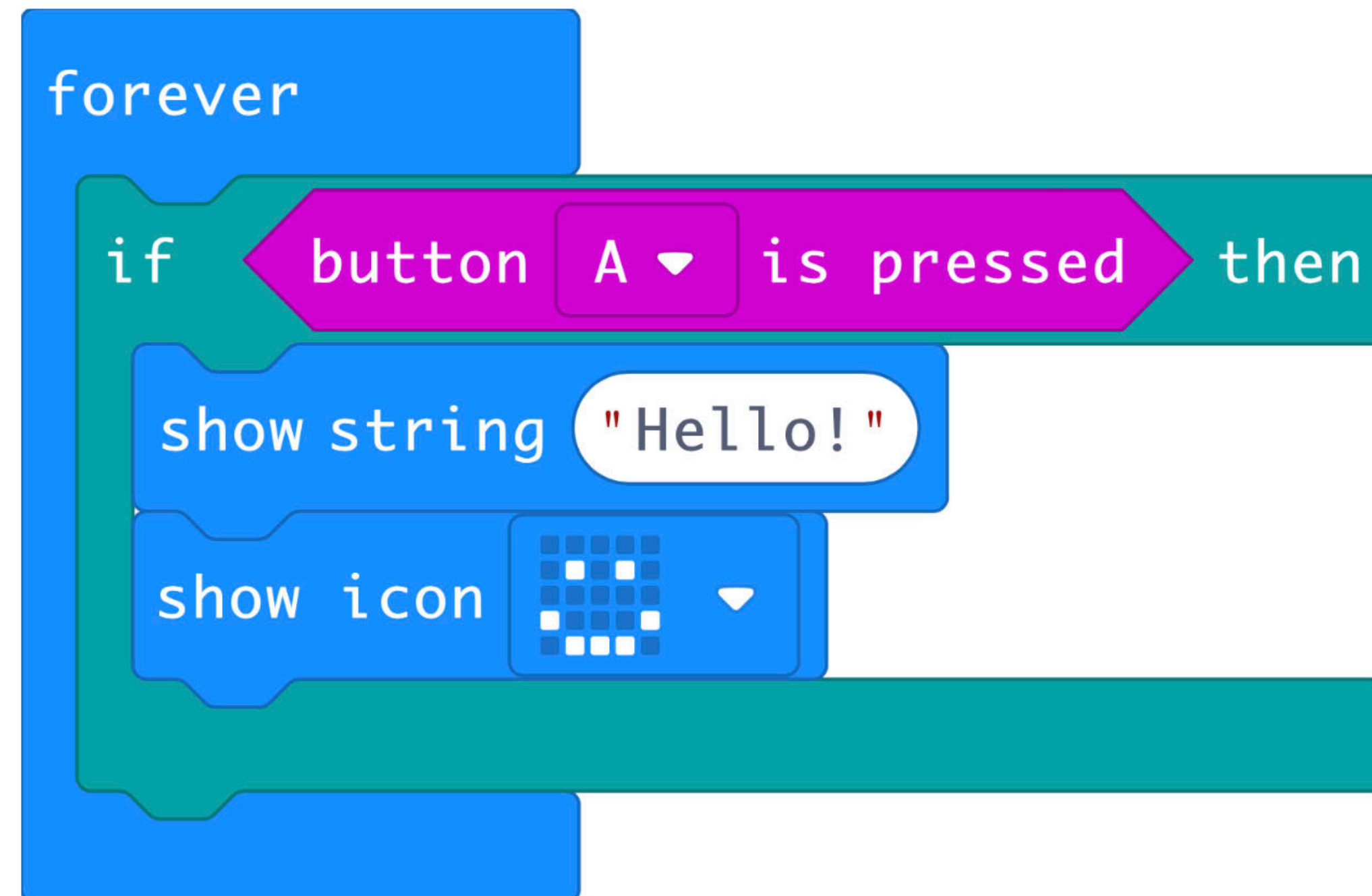
Write an algorithm for your micro:bit to execute code based on a decision being made. You will need to use an 'if' statement (you will find these in the logic blocks'.



# Challenge 2

## Conditional Statements

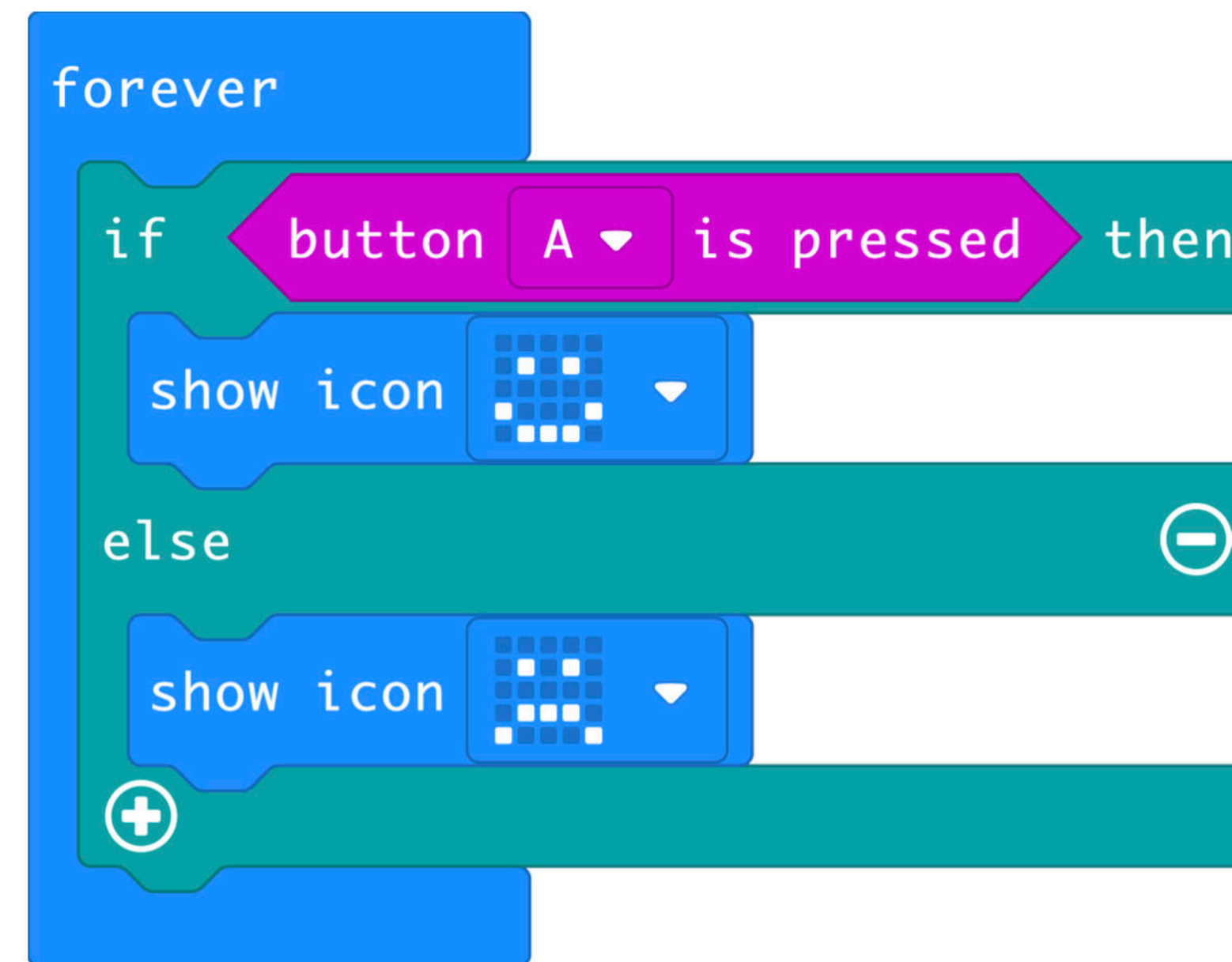
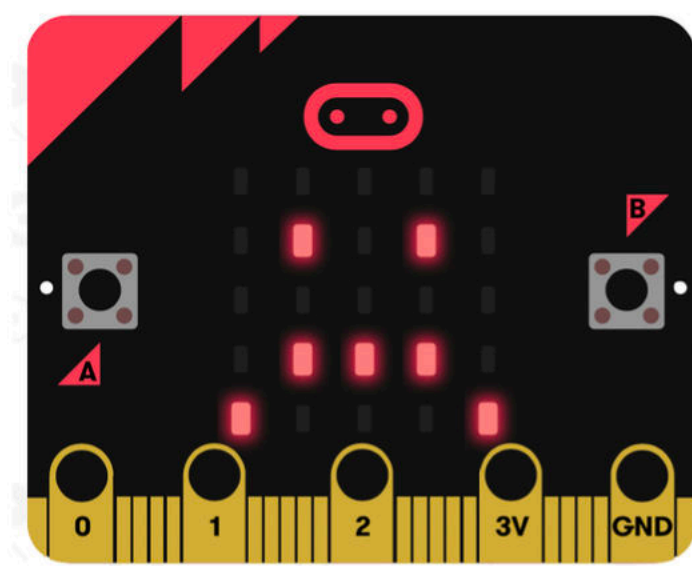
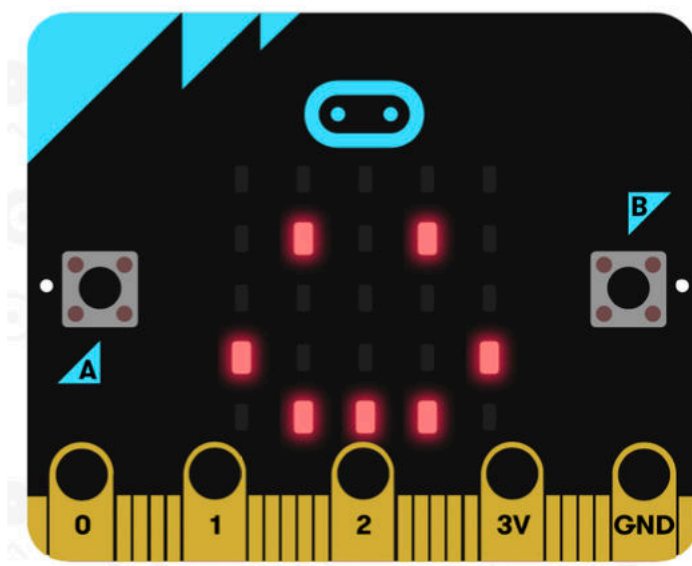
Write an algorithm for your micro:bit to execute code based on a decision being made. You will need to use an 'if' statement (you will find these in the logic blocks'.



# Challenge 2

## Conditional Statements

Write an algorithm for your micro:bit to execute different code based on a decision being made. You will need to use an 'if - else' statement (you will find these in the logic blocks'.



You can use the simulator to run your code, or download and test it on your micro:bit.



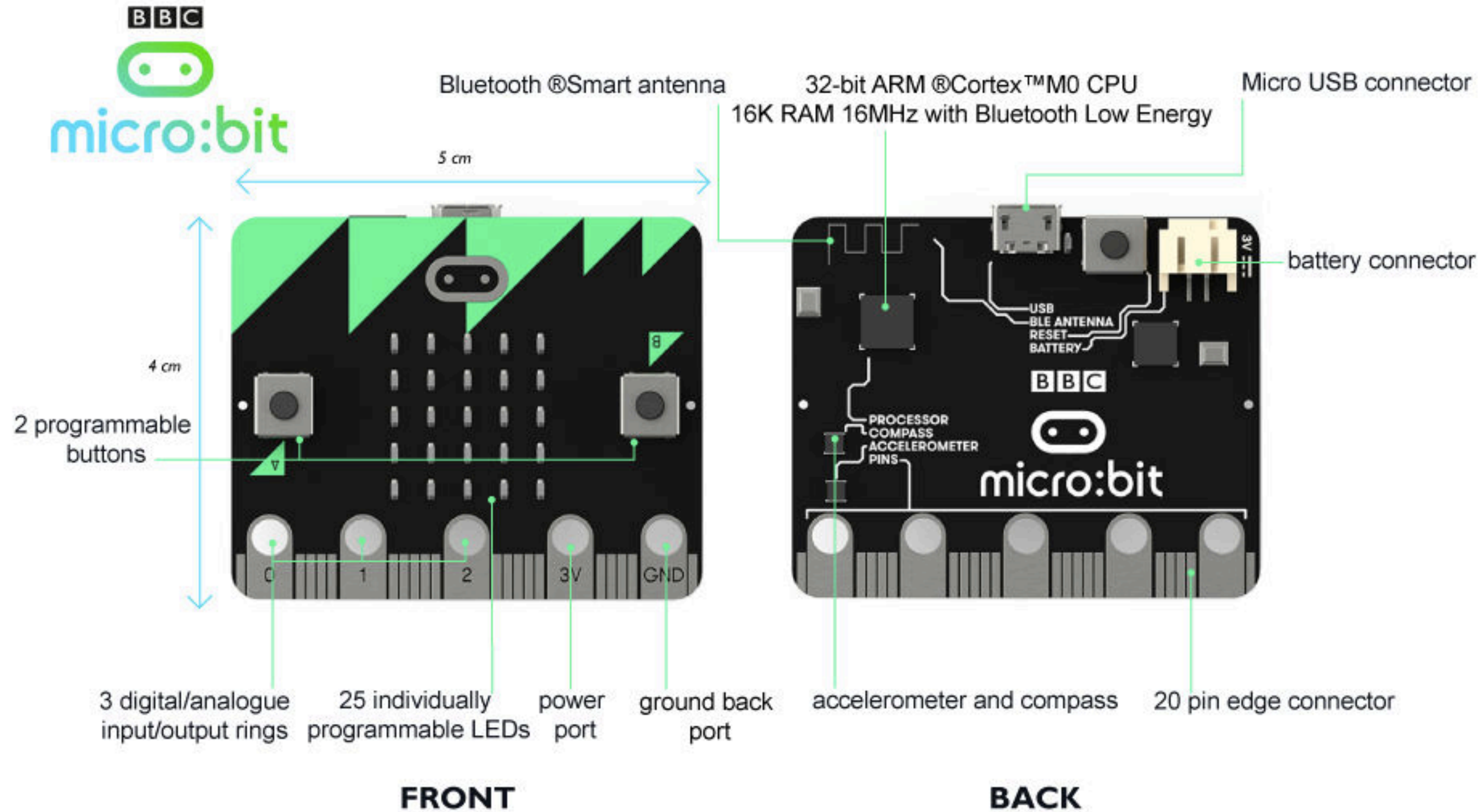
## Challenge 2

```
forever
  if button A is pressed then
    show string " Boo! "
  else
    show icon [grid icon]
```

```
forever
  if button A is pressed then
    show icon [grid icon]
  else if button B is pressed then
    show icon [grid icon]
  else
    show icon [empty icon]
```

# Challenge 3

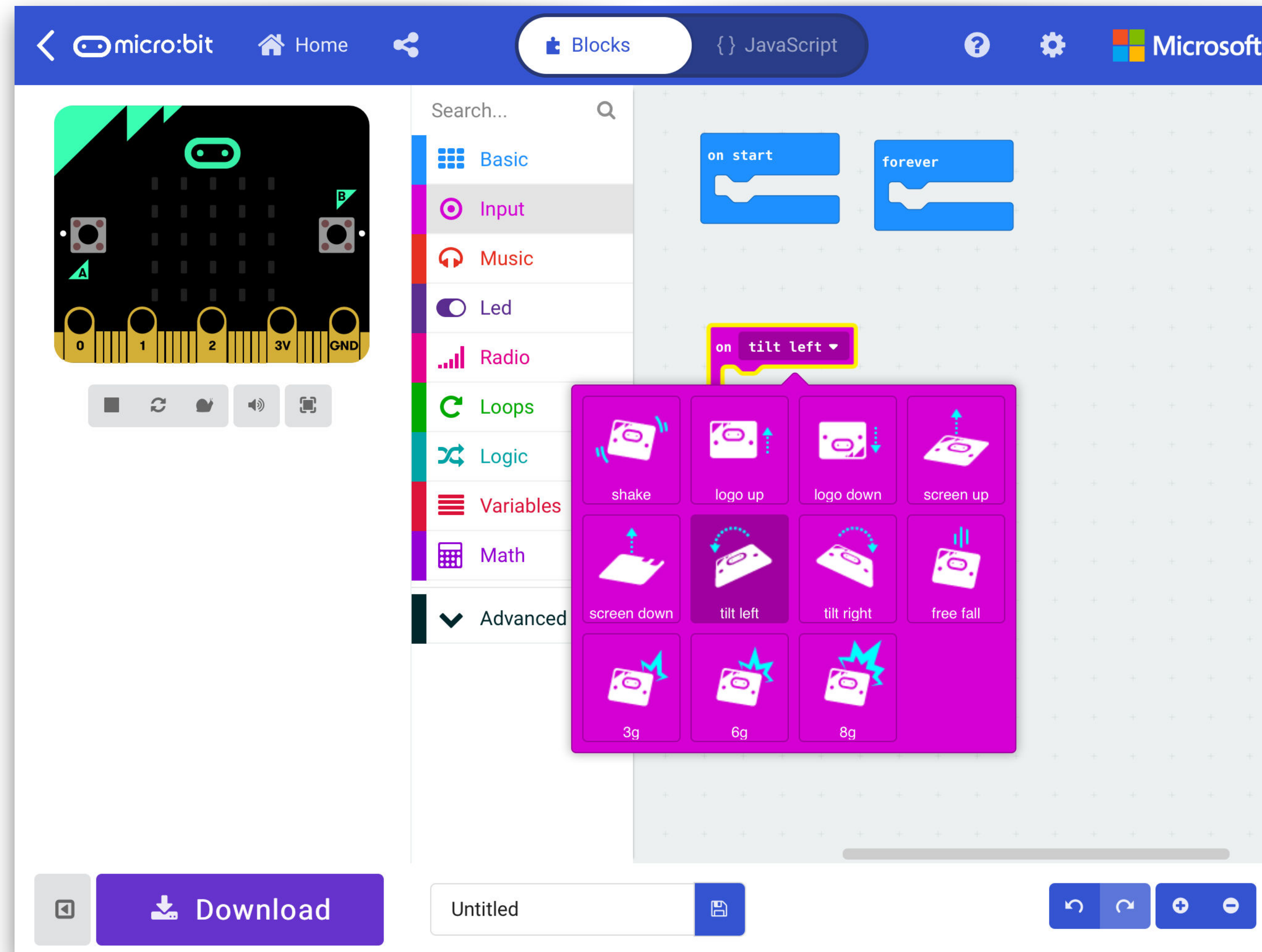
## Reading sensor inputs





# Challenge 3

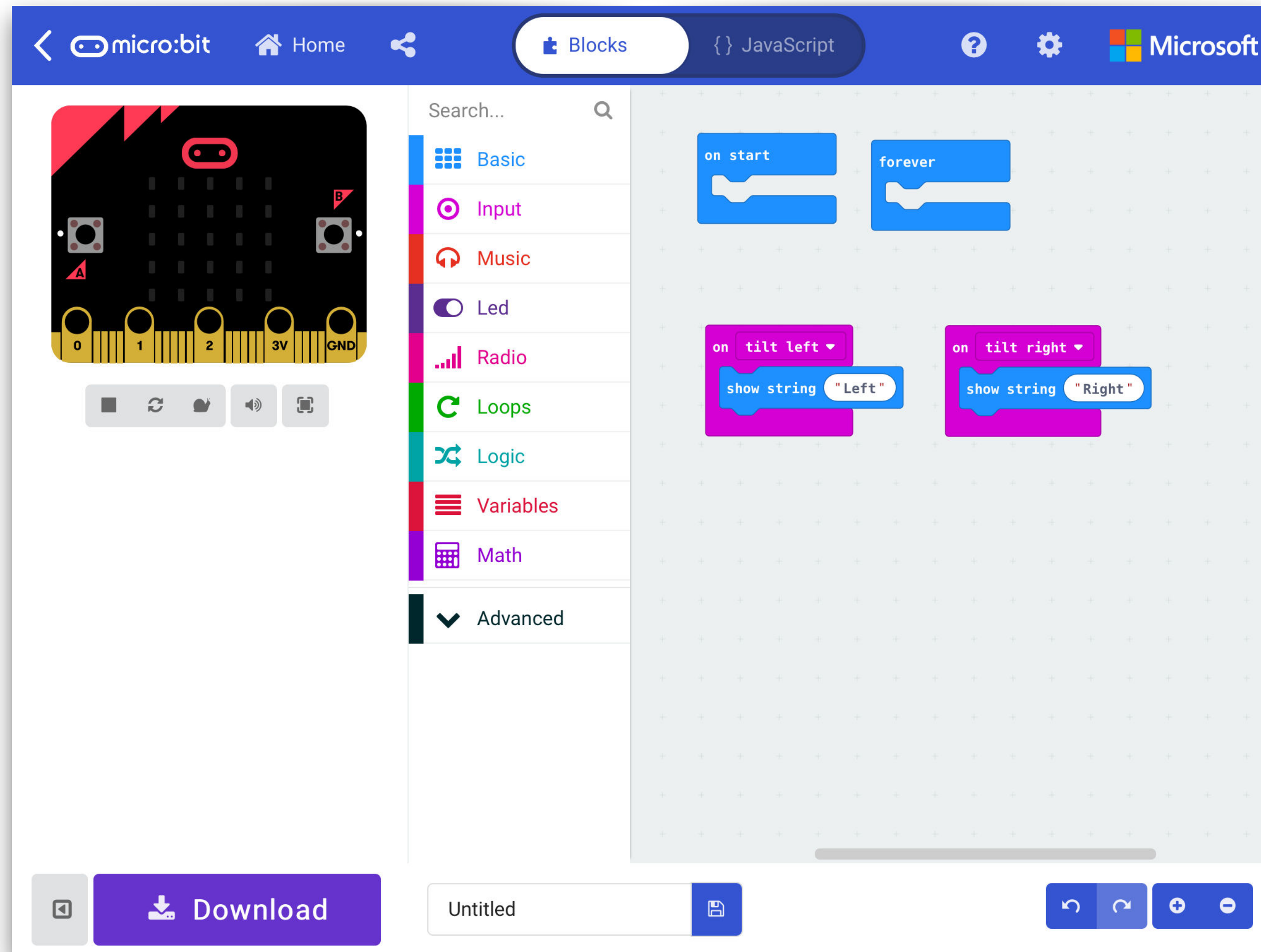
## Sensor inputs (Accelerometer)



The screenshot shows the Microsoft MakeCode editor interface for a micro:bit. The top navigation bar includes a back arrow, the text "micro:bit", a home icon, a share icon, a "Blocks" tab, a "JavaScript" tab, a help icon, a settings icon, and the Microsoft logo. On the left, there is a visual representation of the micro:bit board with pins labeled 0, 1, 2, 3V, and GND. Below the board are icons for running, refreshing, erasing, and saving. The central panel features a search bar and a category menu with options: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The right panel is a workspace with a grid background. The program consists of an "on start" block followed by a "forever" loop. Inside the loop, there is an "on tilt left" event block. A dropdown menu is open for the "on tilt left" block, showing various accelerometer events: shake, logo up, logo down, screen up, screen down, tilt left (highlighted), tilt right, free fall, 3g, 6g, and 8g. At the bottom, there is a "Download" button, a text field containing "Untitled", a save icon, and navigation controls for undo, redo, zoom in, and zoom out.

# Challenge 3

## Sensor inputs (Accelerometer)



The screenshot shows the Microsoft MakeCode editor interface for a micro:bit. The top navigation bar includes a back arrow, 'micro:bit', 'Home', 'Blocks', 'JavaScript', a help icon, a settings gear, and the Microsoft logo. On the left, there is a visual representation of the micro:bit board with pins labeled 0, 1, 2, 3V, and GND. Below the board are icons for running, refreshing, and other actions. A central sidebar contains a search bar and a list of block categories: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The main workspace is a grid where the following code is written:

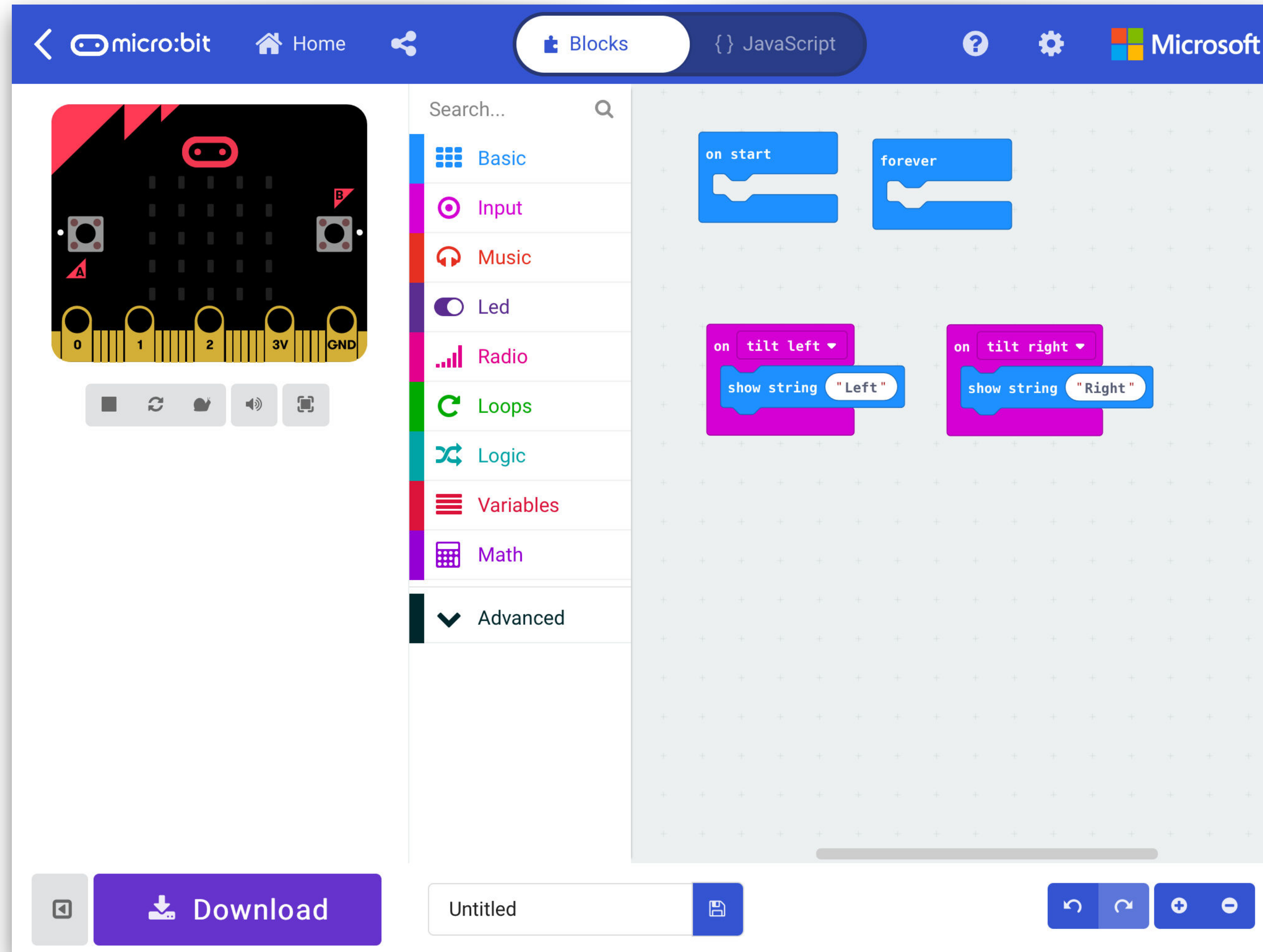
- An **on start** block followed by a **forever** loop block.
- Inside the **forever** loop, there are two **on tilt left** and **on tilt right** event blocks.
- The **on tilt left** block is connected to a **show string** block with the value **"Left"**.
- The **on tilt right** block is connected to a **show string** block with the value **"Right"**.

At the bottom of the editor, there is a **Download** button, a text field containing **Untitled**, and a save icon. On the far right, there are navigation icons for undo, redo, and zoom in/out.



# Challenge 3

Sensor inputs (Accelerometer) - Add "Shake detector"



The screenshot displays the Microsoft MakeCode editor for a micro:bit. The top navigation bar includes a back arrow, 'micro:bit', 'Home', 'Blocks', and 'JavaScript' tabs, along with help, settings, and Microsoft logos. On the left, a virtual micro:bit board is shown with pins labeled 0, 1, 2, 3V, and GND. The central category menu lists various blocks: Basic, Input, Music, Led, Radio, Loops, Logic, Variables, Math, and Advanced. The main workspace contains a script starting with 'on start' and 'forever' loops. Below these, there are two 'on tilt' blocks: 'on tilt left' with a 'show string' block containing 'Left', and 'on tilt right' with a 'show string' block containing 'Right'. At the bottom, there is a 'Download' button, a file name 'Untitled', and navigation icons.



**STARLIGHT**  
EDUCATION