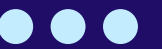


ROBOTRONIX

Module

micro:bit



By Mohd Yazid bin Mohd Anas Khan
Zalina binti Ibrahim
Zainon binti Haji Bibi

Team Member

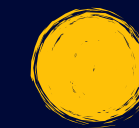
Mohd Yazid
Mohd Anas Khan
Editor

Zalina
Ibrahim
Editor

Zainon
Haji Bibi
Designer

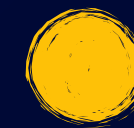
CONTENTS

- Introduction
- Must-have items
- Parts
- Overview of makecode Interface
- Overview of Input/Output



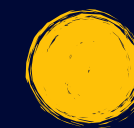
Input

- Ultrasonic Sensor
- Light Sensor
- Line Sensor
- Accelerometer
- Temperature Sensor

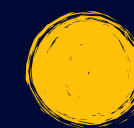


Output

- Servo
- DC Motor
- RGB on board



Potential Project Ideas with Micro:bit



Innovation Project with Micro:bit



Introduction

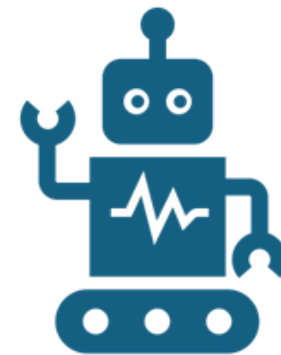
Mini Robot



Must-have items



Laptop (Windows /
macOS)



ZOOM:BIT Robot



4 x AA batteries
(Eneloop Pro)

Parts

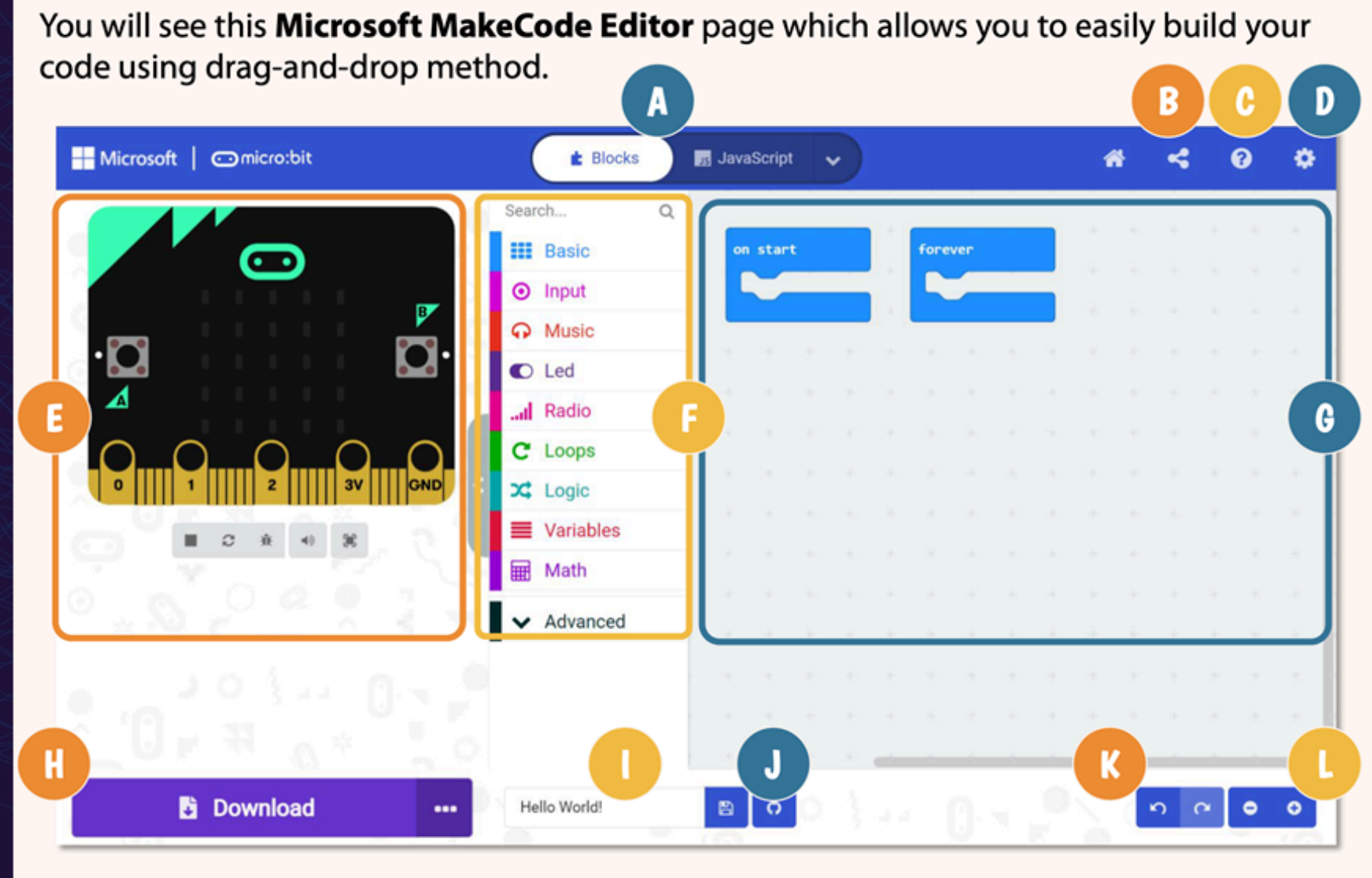


What's in the Box?



Overview of makecode Interface

- A** Choose to program in Blocks, JavaScript or Python.
- B** Publish and share your project.
- C** Open Help menu.
- D** Change settings, add extensions, connect device, etc.
- E** **Simulator** - Show you a simulation of your code.
- F** **Toolbox / Category Drawers** - Get the coding blocks that you need here. Click to see available coding blocks for each category.
- G** **Programming Workspace** - Build your code here by snapping blocks together.
- H** Click to download your code to ZOOM:BIT.
- I** Name and save current project to your computer.
- J** Create GitHub repository.
- K** Undo / Redo
- L** Zoom in / out.



Overview of Input/Output

INPUT / DERIA RASA MASUKAN

- *Ultrasonic Sensor*
- *Light Sensor*
- *Microphone*
- *Line Sensor*
- *Accelerometer*
- *Temperature Sensor*
- *Compass*
- *Touch Logo*



OUTPUT / KELUARAN

- *Servo*
- *LED Matrix*
- *Buzzer*
- *DC Motor*
- *Headlights*
- *RGB onboard*



INPUT

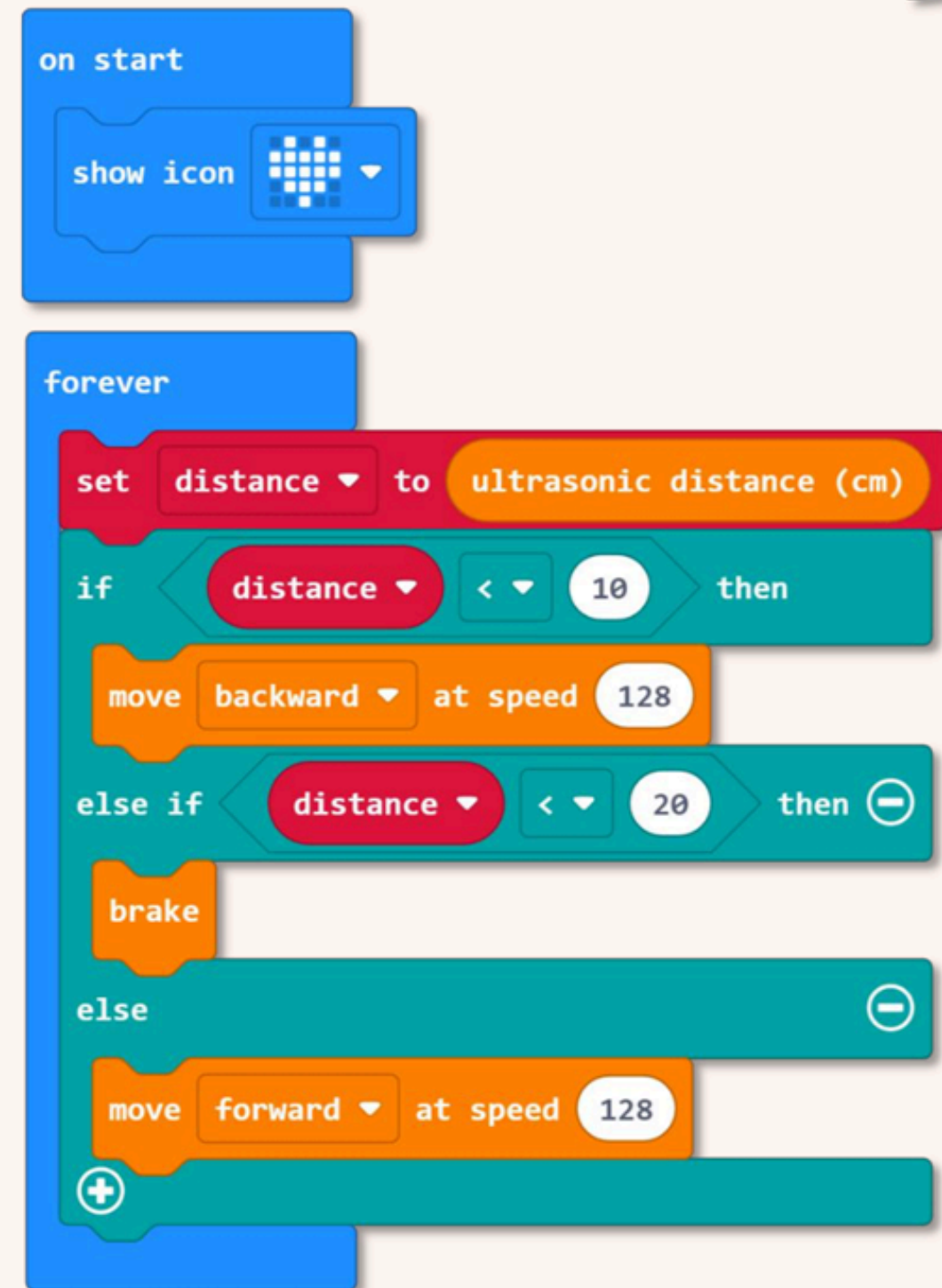
Ultrasonic Sensor

- 1 Create a new project in your MakeCode Editor and add ZOOM:BIT extension (you can refer to pp. 44-45).
- 2 Build this code. You can get the blocks you need from these category drawers:



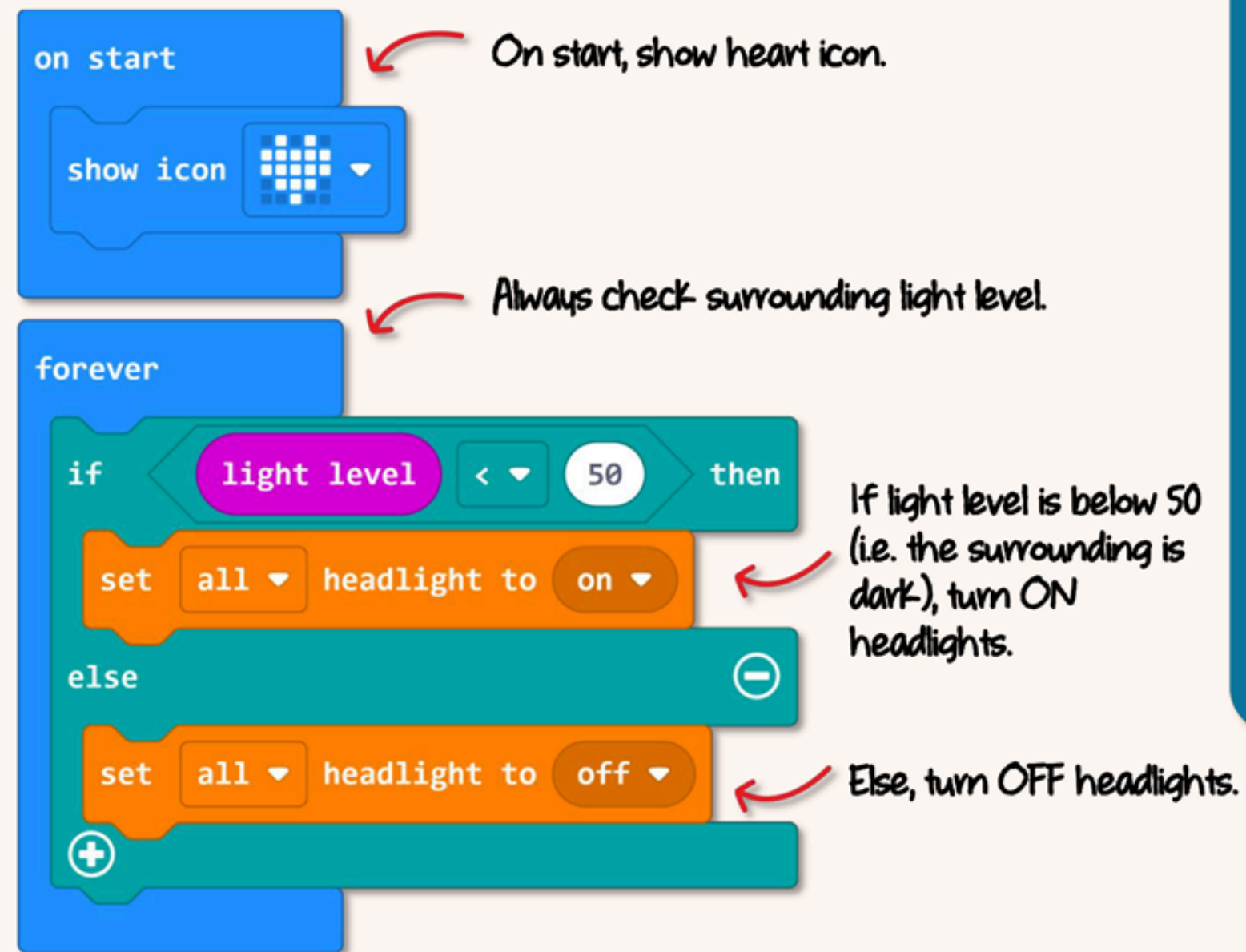
You'll need to create a new variable ("distance") and set it to always get the ultrasonic sensor value.

You can go to <https://link.cyttron.io/zoombit-tutorial-7> for step-by-step guide to build the code if you need more guidance.



Light Sensor

4 Build the following code.



You can find the blocks you need from the following category drawers:

Basic

Logic

Input

ZOOM:BIT

For more guidance, you can go to <https://link.cyttron.io/zoombit-tutorial-3> for step-by-step guide to build the code.



Line Sensor

- 7 Download the completed code to your ZOOM:BIT. Power it up, place it on the track and press Button A.

```

on start
  show icon [grid icon]
  set position to 0
  while not button A is pressed
  do
  start melody [jump up] repeating once
  
```

Let's test. Try to push ZOOM:BIT off track (until no line is detected by Maker Line). Do you notice ZOOM:BIT readjusting its position to get back on track, instead of wandering off?



```

forever
  if line detected on center then
    move forward at speed 128
  else if line detected on left then
    set motors speed: left 50 right 100
    set position to 1
  else if line detected on right then
    set motors speed: left 100 right 50
    set position to 2
  else if line detected on far left then
    set motors speed: left 0 right 100
    set position to 1
  else if line detected on far right then
    set motors speed: left 100 right 0
    set position to 2
  else if line detected on none then
    if position = 1 then
      turn left at speed 80
    else if position = 2 then
      turn right at speed 80
  
```


Accelerometer

Search...

- Basic
- Input**
- more
- Music
- Led
- Radio
- Loops
- Logic
- Variables
- Math
- Extensions
- Advanced

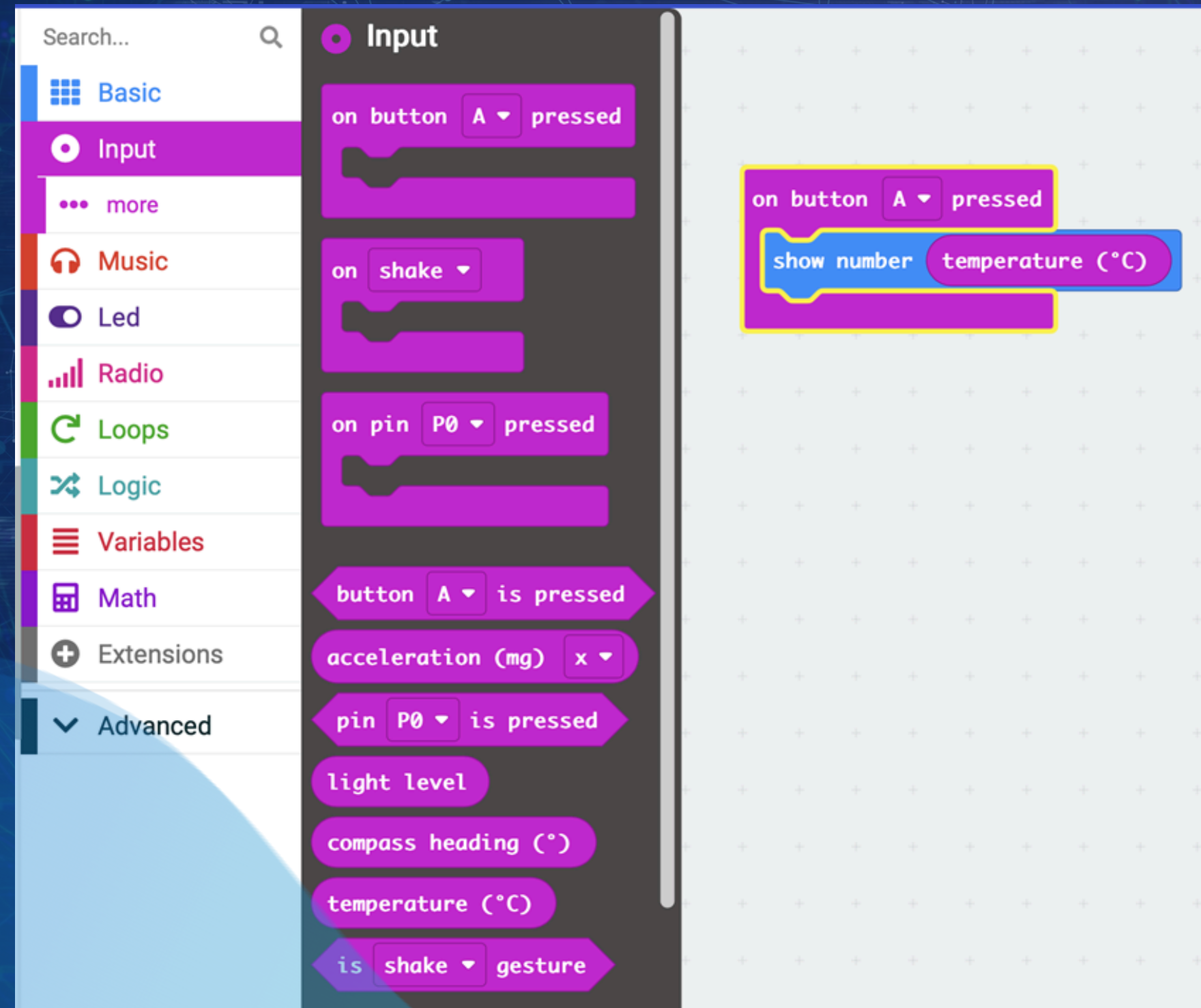
Input

- on button A pressed
- on shake
- on pin P0 pressed
- button A is pressed
- acceleration (mg) x
- pin P0 is pressed
- light level
- compass heading (°)
- temperature (°C)
- is shake gesture

forever

show number acceleration (mg) strength

Temperature Sensor





OUTPUT

Servo

The screenshot displays the Scratch Servo control interface. On the left, a wiring diagram shows a servo motor connected to a microcontroller board. The servo's brown, red, and blue wires are plugged into the 3V, GND, and a signal pin (labeled 0) respectively. The microcontroller board has pins labeled 0, 1, 2, 3V, and GND. Below the wiring diagram are icons for a stop button, a refresh button, a servo icon, a speaker icon, a camera icon, and a screenshot icon.

In the center is a block palette with a search bar and a list of categories: Basic, Input, Music, Led, Radio, Servos (highlighted), Loops, Logic, Variables, Math, Extensions, and Advanced.

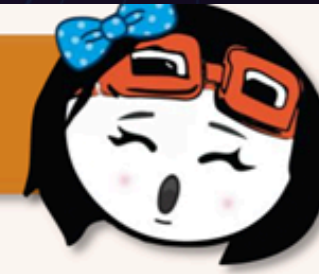
On the right is the 'Servos' control panel, divided into three sections:

- Positional:** A block 'set servo P0 angle to 90°'.
- Continuous:** A block 'continuous servo P0 run at 50%' and a block 'stop servo P0'.
- Configuration:** Three blocks: 'set servo P0 stop on neutral OFF', 'set servo P0 range from 0 to 180', and 'set servo P0 pulse to 1500 μs'.

On the far right is the script area, which contains a 'forever' loop block. Inside the loop, there are four blocks: 'set servo P0 angle to 90°', 'pause (ms) 1000', 'set servo P0 angle to 180°', and 'pause (ms) 1000'.

DC Motor

Now we're ready to program ZOOM:BIT to move around... Let's start! Zip zip Zoom ~



- 1 Create a new project in your MakeCode Editor and add ZOOM:BIT extension (you can refer to pp. 44-45).
- 2 Build the code below. You can get the blocks you need from these category drawers:



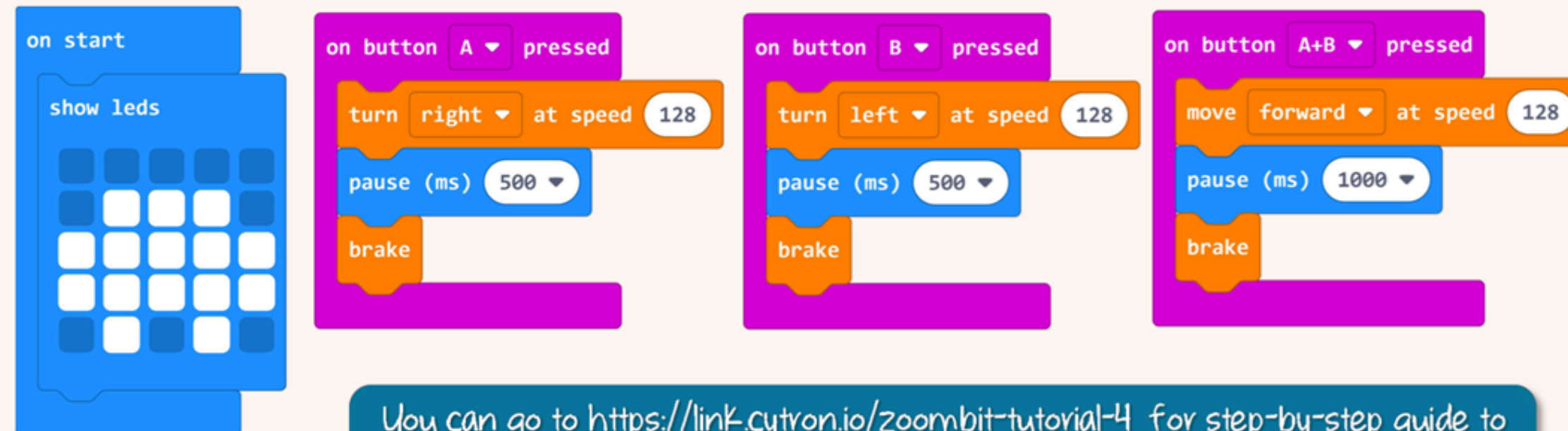
Basic



Input



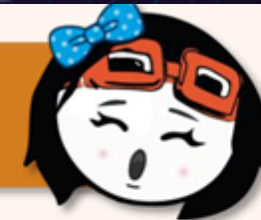
ZOOM:BIT



You can go to <https://link.cyttron.io/zoombit-tutorial-4> for step-by-step guide to build the code if you need more guidance.

RGB Onboard

There are two RGB LEDs on REKA:BIT board, labelled "0" and "1". You can program them to light up in different colours using blocks from [REKA:BIT] category drawer.



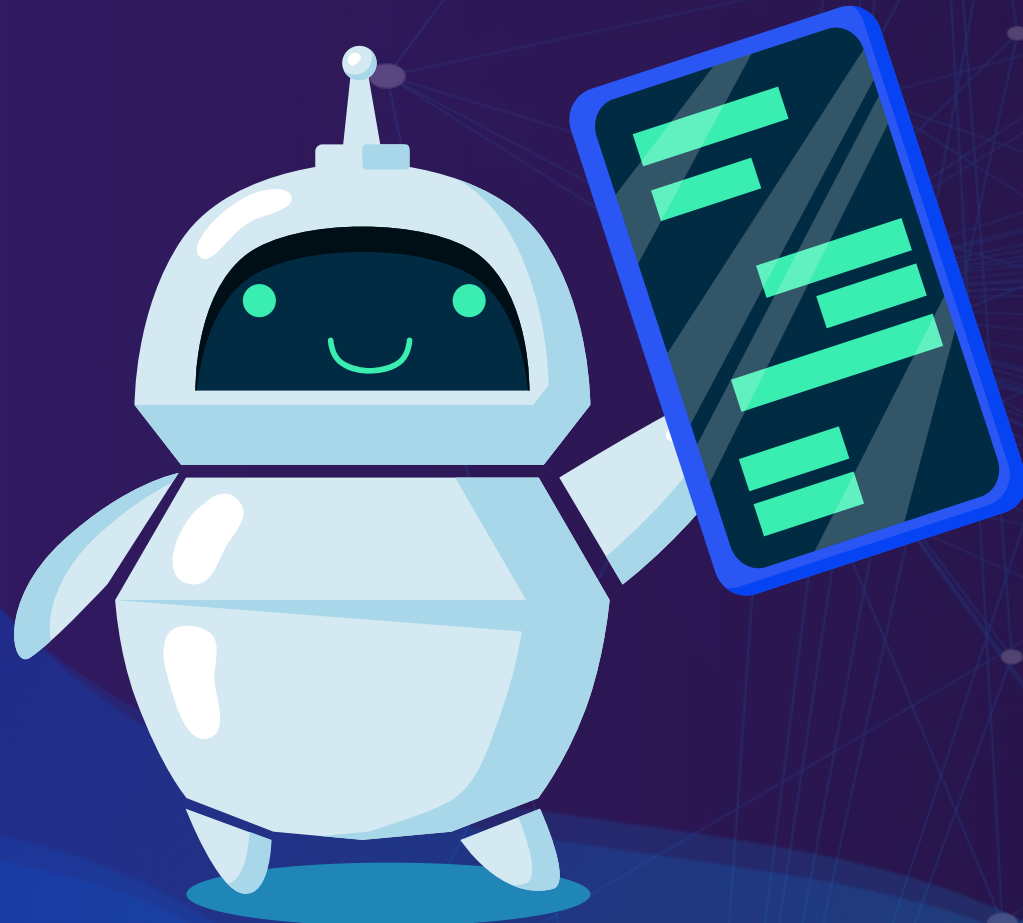
- 1 Add the following highlighted blocks from [Basic], [Loops] and [REKA:BIT] category drawers to your code from the previous lesson.

The code consists of four event-driven blocks:

- on start** (blue):
 - show leds (blue block with a 4x4 grid)
 - clear all RGB pixels (orange block)
- on button A pressed** (pink):
 - repeat 4 times (green loop block):
 - do (green block):
 - set RGB pixel 0 to (orange block with a red circle)
 - pause (ms) 100 (blue block)
 - set RGB pixel 0 to (orange block with a black circle)
 - pause (ms) 100 (blue block)
 - turn right at speed 128 (orange block)
 - pause (ms) 500 (blue block)
 - brake (orange block)
- on button B pressed** (pink):
 - repeat 4 times (green loop block):
 - do (green block):
 - set RGB pixel 1 to (orange block with a red circle)
 - pause (ms) 100 (blue block)
 - set RGB pixel 1 to (orange block with a black circle)
 - pause (ms) 100 (blue block)
 - turn left at speed 128 (orange block)
 - pause (ms) 500 (blue block)
 - brake (orange block)
- on button A+B pressed** (pink):
 - set all RGB pixels to (orange block with a red circle)
 - move forward at speed 128 (orange block)
 - pause (ms) 1000 (blue block)
 - brake (orange block)
 - set all RGB pixels to (orange block with a black circle)

For more guidance, you can refer to <https://link.cytron.io/zoombit-tutorial-5> for step-by-step guide to build the code.

Potential Project Ideas with Micro:bit



- *Agriculture*
- *Smart Home*
- *Home Security*
- *Health Monitoring*
- *Automation*
- *Robotics*
- *Traffic Survey Data Logger*
- *Traffic Survey Data Logger*
- *Energy Cost Calculator*
- *LED Matrix Sketch*
- *Solar Weather Station*
- *Physical Training Recognition*
- *Cat Feeder*
- *Digital Measuring Roller*
- *Robot Hand*

Innovation Project with Micro:bit



Things to consider when building a project with micro:bit :-

- *Based on recyclable materials*
- *Low-cost prototyping*
- *Practicality*
- *Ease-of-use*
- *User-friendly*

Thank You

Robotronix

Module

micro:bit



Published by:

Perpustakaan Tun Abdul Razak
Universiti Teknologi MARA
Cawangan Sarawak