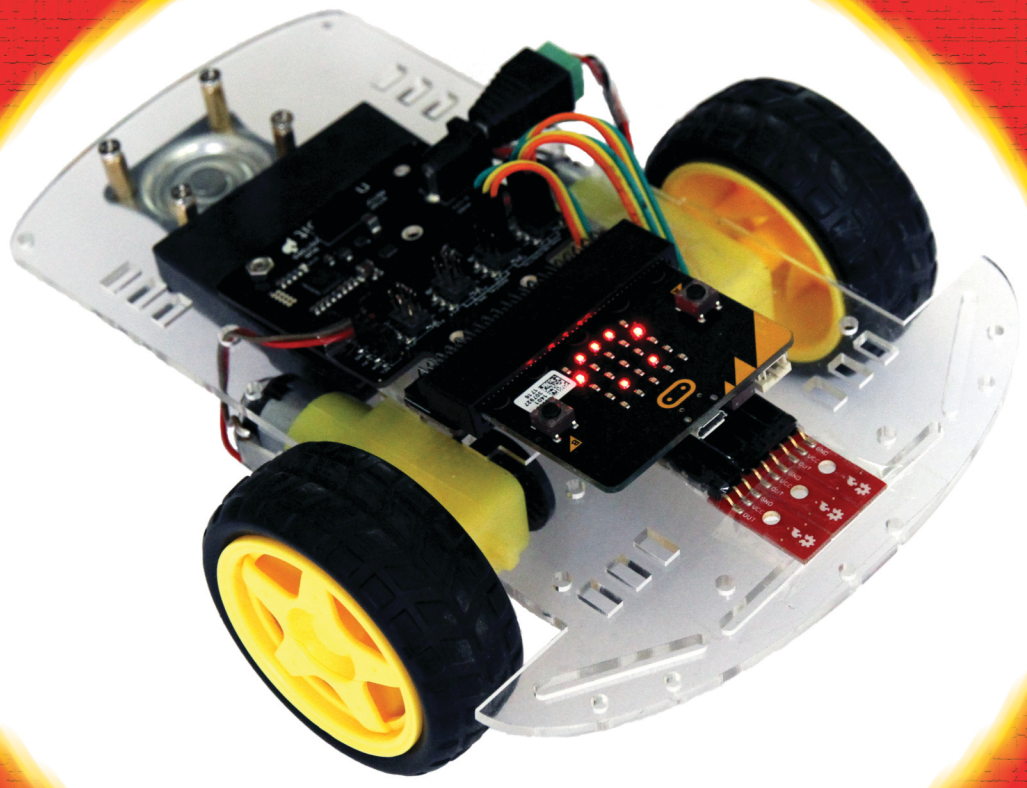
BBC

micro:bit

Robot Kit v2



Robo:bit

ABRA

www.abra-electronics.com

Table of Contents

OBJECTIVE

Welcome to the exciting world of robotics! This kit provides a step-by-step tutorial on how to create your very own autonomous robot using the *micro:bit* platform.

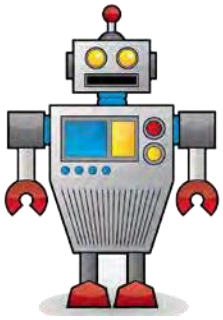
This robot will:

- Drive in all directions
- Drive within a specified perimeter
- Drive along a line
- Drive with object collision avoidance

LEVEL	Beginner-Intermediate
ASSEMBLY TIME	1-2 hour

Robot Specifications:

SIZE [cm] (LxWxH)	22 x 16 x 7.5
WEIGHT [kg]	0.390
VOLTAGE [V]	6
SENSORS	Line Following IR
MOTOR GEAR	120:1
WHEEL DIAMETER [in]	2.5
SPEED [m/s]	~ 1



PART LIST

	DESCRIPTION	ABRA PART NO.
	*micro: bit v2* controller	MICROBIT
	*moto: bit* board	DEV-14213
	Robot chassis + Assembly Manual	GK-KIT-005
	DC Power Adapter 2.1mm	369-ADA
	USB A to Micro B Cable	CAB-600- R
	F/F Jumper Wire	794-ADA
	Line Sensors [x3]	SEN-11769
	4x AA Ba&#xfffd;eries	GP15-4

INTRODUCTION

- The micro: bit v2 is an easy to use yet surprisingly powerful piece of hardware that will allow you to create many impressive gadgets.
- The board was designed in the UK by the BBC and measures only 4x5 cm.
- The module interacts with many electronic parts and runs customized scripts.
- The micro: bit v2 is perfect for beginners who are starting simple projects and for advanced users looking for new challenges.

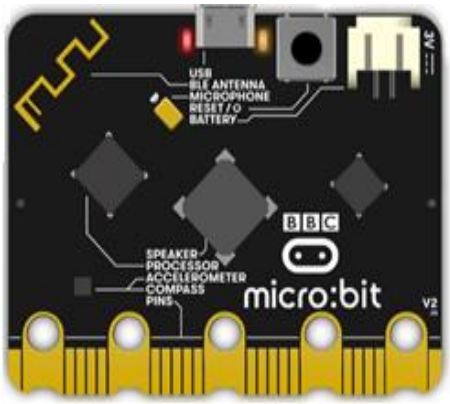


Figure 1: Micro: Bit v2

MICRO: BIT V2– HARDWARE

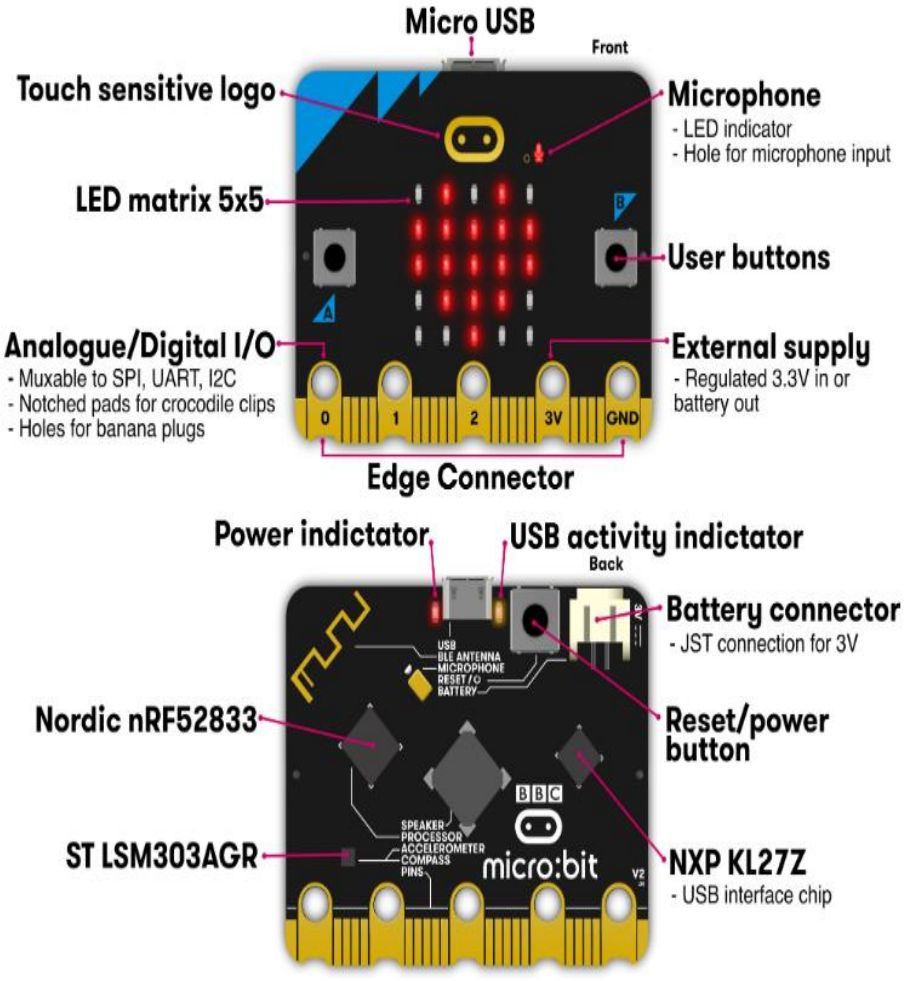


Figure 2: Hardware Components of Micro: Bit v2

Specifications:

- 16MHz 32-bit ARM Cortex M4 Nordic nRF52833
- 512 kB Flash, 128 kB RAM
- 2.4 GHz Bluetooth BLE
- USB 2.0 OTG (On-The-Go)
- 3.3V regulator (for USB only)
- 3 axis accelerometer and magnetometer (I2C) – NXP/Freescale LSM303AGR
- 5x5 LED array
- 2 programmable tactile push buttons – 1 reset button
- Ring connectors (3 x I/O, Power, Ground)
- 19 assignable GPIO pins
- PCB mounted magnetic speaker 80dB @ 5V, 10cm (2700Hz)
- Microphone with sensitivity -38dB ±3dB @ 94dB SPL

MICRO: BIT V2 – SOFTWARE

The micro: bit v2 also offers powerful software capability. The code editor is readily available online and scripts can be programmed by simply dragging and dropping block modules. Furthermore, before compiling the code, we can test our project on the micro: bit simulator. This project requires the use of a special Sparkfun package.

MakeCode JavaScript Editor:

Start coding here: https://makecode.microbit.org/#editor

Figure 3: Micro: Bit Code Editor

Steps:

1	Block mode
2	Code block selection menu
3	Program simulator
4	Name/Save file .js
5	Download hex file for uploading on board
6	JavaScript mode

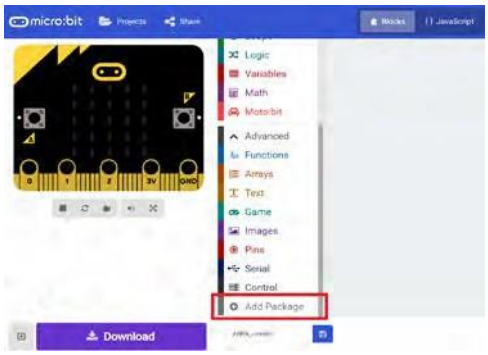
Uploading Programs:

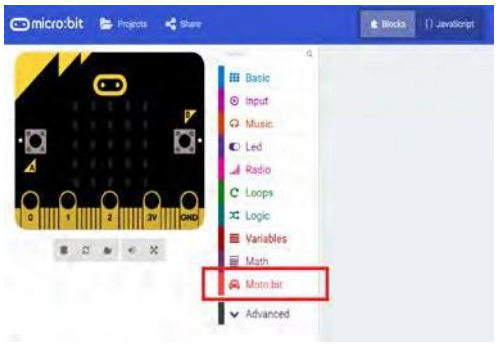
Connect the *micro:bit v2* into your computer USB. Your computer should recognize the device and create a MICROBIT drive.

1. Compile the script in MakeCode. This creates a .hex file.
2. Save file locally on your computer (ex: My Documents)
3. Locate the downloaded .hex file and folder and copy it (drag and drop) into the MICROBIT drive folder. This will compile the file onto the hardware. (You cannot do multiple files at a time)
4. The micro: bit v2 LED will flash for a few seconds. The software has been incorporated into the flash memory. This means that even after unplugging your device your program will remain. It will execute next time the board is powered.
5. Safely Remove Hardware and eject MICROBIT from your computer.

Install Moto:bit Package:

For ease of programming, we rely on the Sparkfun Moto:bit package that easily integrates into MakeCode environment and offers useful functions.





Click: **"+ Extensions"** → Search: "**Sparkfun**" → Select "**moto-bit**" → "**Moto: bit**"

Sample Code for this robot can be found:

In the ***"CODE"*** folder in ***"Attachments"*** tab of the product page.
(Search for "*micro: bit*" at https://abra-electronics.com/)

ROBOT

1- CHASSIS ASSEMBLY

Refer to the manual: Macgyver – 2WD Frame Starter Pack [GK-KIT-005] by GeeekUs included in this kit (and on the product webpage under “Attachments”)

Required Tools:

- Phillips head screwdriver
- Glue (Glue-gun or Super Glue) [Optional]
- Adhesive tape
- Soldering Iron [Optional]

Micro: bit v2 Power Connection:

Once the chassis has been assembled, we need to install the battery pack, power connector and moto: bit board. (Different configurations are possible)

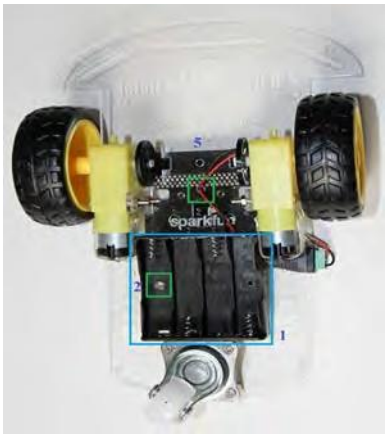


Figure 4: Bottom

Figure 5: Top

1. Place the battery pack on the bottom side of the robot between both motors.
2. Using a long screw and nut from the chassis hardware bag (GK-KIT-005), feed the screw through the hole of the battery pack up through the chassis.
3. On the top side of the robot, feed the same screw through the bottom right hole of the moto: bit and tighten down the battery and moto: bit with a nut. Be careful not to over-tighten.
4. The micro: bit v2 should now sit at the very front of the robot. (It is normal that the moto: bit is slightly slanted and does not sit flush on the chassis.)
5. Feed the red and black battery wires from underneath the chassis through a small hole.
6. Screw them into the DC adapter (Red: +, Black: -) and plug into the moto: bit
7. Wind and tape the wires to the chassis.

2- MOTOR ASSEMBLY

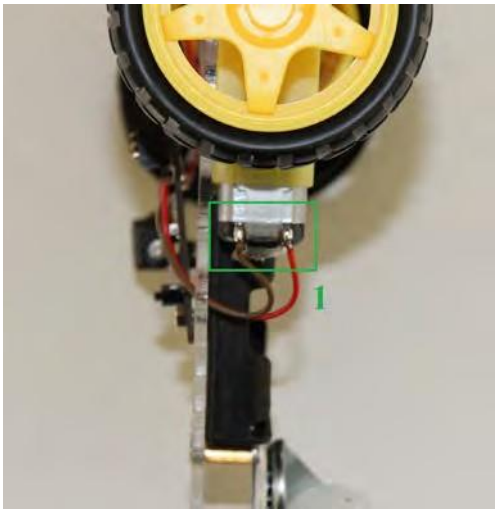


Figure 6: Motor Assembly Side View

Figure 7: Motor Assembly Front View

1. Solder the Brown(black) (-) and Red (+) wires to the brass motor tabs (You can use a glue-gun for this step, do not use Super Glue) (You can find tutorials for soldering on YouTube)
2. Feed the wires through the bottom of the robot through a hole.
3. Connect the red and brown(black) wires in the motor slots as labeled (Taping the wires down will help)
4. ***If motors turn in the incorrect direction, inversing the red and brown wire on the moto:bit will reverse the direction of the motors. (Or invert the motor in the code)

A- TEST DRIVE: BASIC

Recreate or download *motor.hex* onto your *micro: bit v2* or https://makecode.microbit.org
Experiment with code values and test one direction at a time

This code will:

- When “A” is pressed:
  - Turn on motors and display checkmark
  - Drive (Forward and Reverse)
  - Left turn (Forward and Reverse)
  - Right turn (Forward and Reverse)
  - Turn off motors and display X

* Make sure, “Run Motor” switch is enabled on your *moto: bit*

* If motors turn in the incorrect direction, inversing the red and brown wire on the moto: bit will reverse the direction of the motors. (Or invert the motor in the code)

Figure 8: Forward

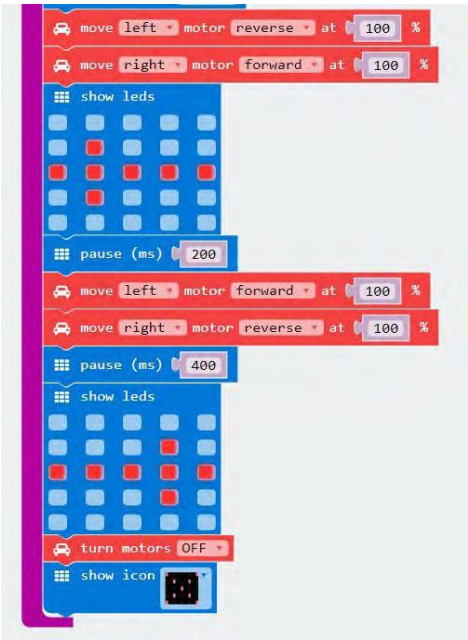


Figure 9: Reverse

B- TEST DRIVE: MOVING WITH SPEAKERS

Recreate or download *speakers.hex* onto your *micro: bit v2* or *https://makecode.microbit.org*

Experiment with code values and you can also try changing directions and set your own music

This code will:

- When “A” is pressed:
  - Speakers and Motors turn on and display checkmark.
  - Drive (Forward and Reverse)
  - Left turn (Forward and Reverse)
  - Right turn (Forward and Reverse)
  - ‘Giggle’ music plays when moving forward.
  - ‘happy’ music plays when moving reverse.
  - Turn off motors and display X.

* Make sure, “Run Motor” switch is enabled on your moto: bit.

* If motors turn in the incorrect direction, inversing the red and brown wire on the moto: bit will reverse the direction of the motors. (Or invert the motor in the code)

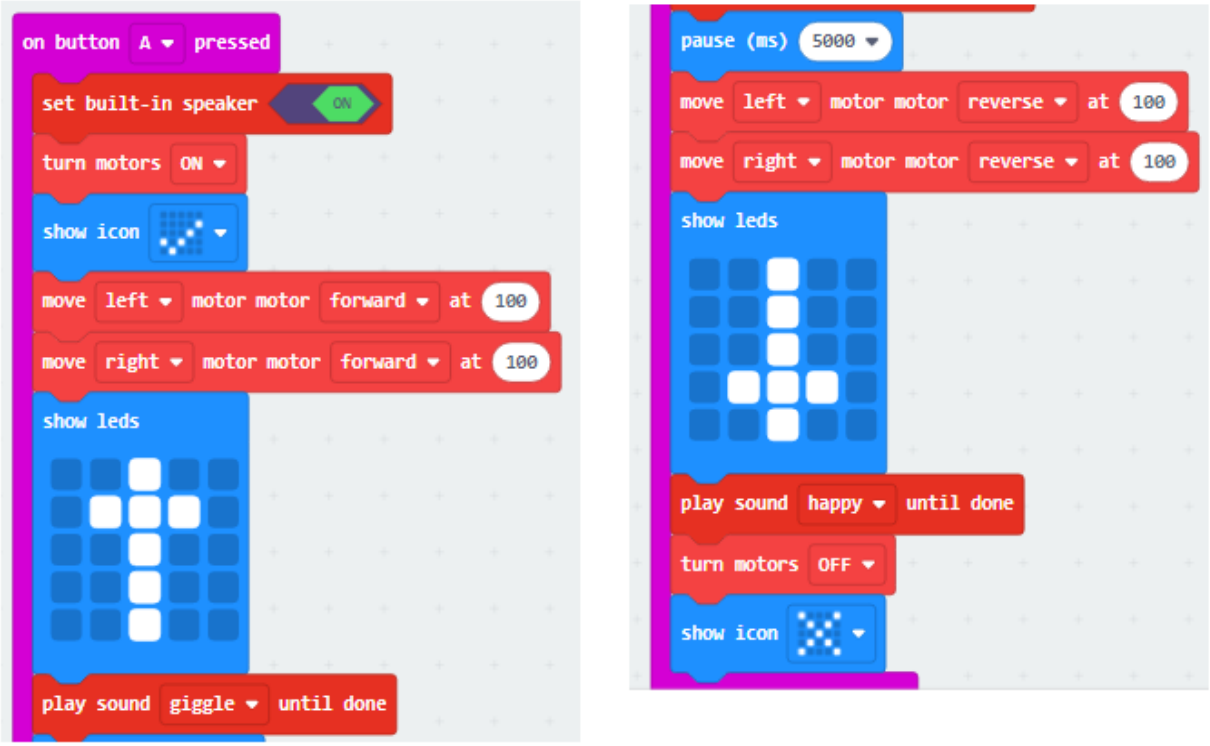


Figure 10: Block Code of Speakers

C- TEST DRIVE: ACTION WITH MICROPHONE

Recreate or download *microphone.hex* onto your *micro: bit v2* or *https://makecode.microbit.org*

Experiment with code values and you can also try changing directions

This code will:

- When “A” is pressed:
  - Turn on motors and display smiley.
  - Drive (Forward) (You can include reverse in the code)
  - Left turn (Forward and Reverse)
  - Right turn (Forward and Reverse)
  - On loud sound (you can clap), motors turn off and display X.

* Make sure, “Run Motor” switch is enabled on your *moto: bit.*

* If motors turn in the incorrect direction, inversing the red and brown wire on the moto: bit will reverse the direction of the motors. (Or invert the motor in the code)

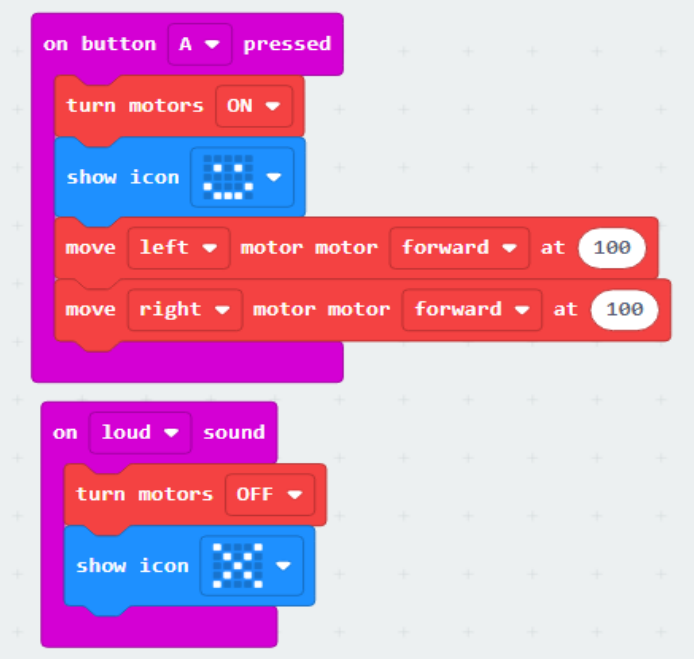
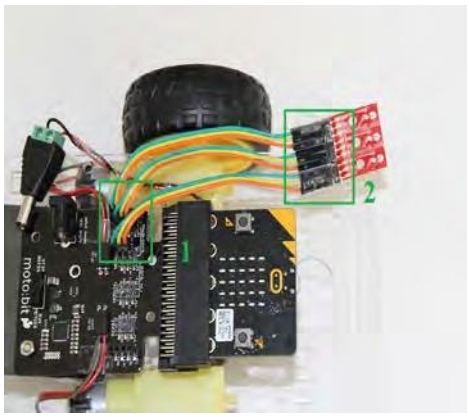
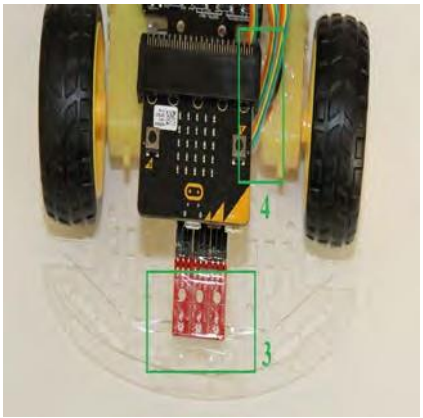


Figure 11: Block Code of Microphone

3- LINE SENSOR ASSEMBLY

The line sensors are used to detect light reflected off objects using infra-red (IR).
This robot uses line sensors to detect a change in the terrain the robot is driving on (set by a perimeter or line to follow)





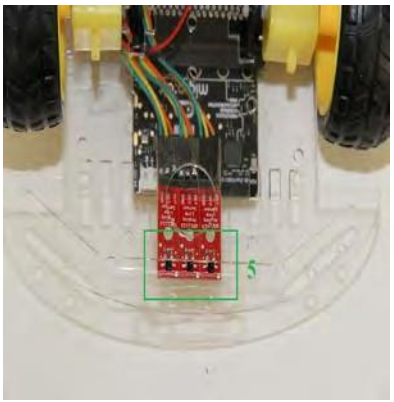


Figure 12: Line Sensor assembly

1. Connect each line sensor **(Green = GND, Yellow = VCC, Orange = OUT)** to moto: bit.
   (Left = SENSOR P0, Center = SENSOR P1, Right = SENSOR P2)
2. Tape the 3 sensors and wires together.
3. Position the sensors and tape them to the chassis.
4. Tuck the wires under the moto: bit.
5. Adjust the sensors so that the IR viewers can see through the slot in the chassis.

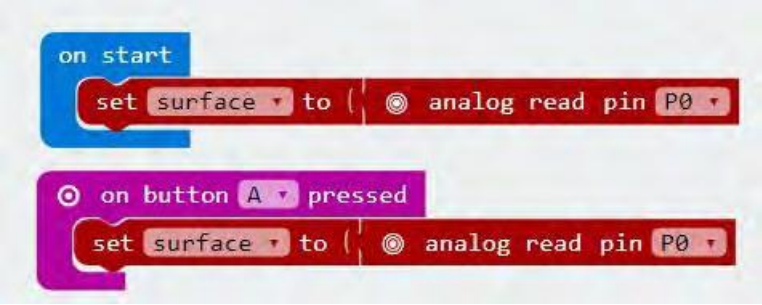
D- TEST DRIVE: PERIMETER

Recreate or download perimeter.hex onto your micro: bit v2 or https://makecode.microbit.org

Calibrating line sensors values and test each one at a time

This code will:

- When A is pressed
  - Set “surface” to the reading of sensor P0 (Left)
  - If surface is within bounds, the robot will drive in a straight line.
  - If the surface read by the sensor changes below the threshold, the robot will reverse and re-evaluate its path.
  - ***Set up a box perimeter, preferably a white surface with thick black tape border.**



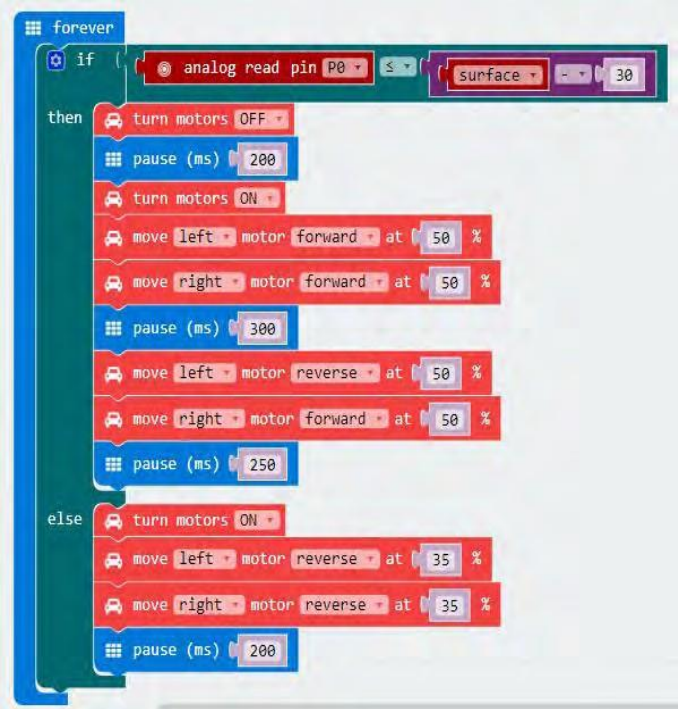


Figure 13: Block Code of Perimeter

E- TEST DRIVE: LINE FOLLOWING

Recreate or download line.hex onto your micro: bit v2 or https://makecode.microbit.org

Calibrating line sensors values and test each one at a time

This code will:

- Calibrate left and right sensors on start up
- When button A is pressed:
  - Drive robot in along a specified line.
  - Constantly adjust course based on left and right sensor values.

*** Set up thin black tape line along white surface for the robot to follow.**

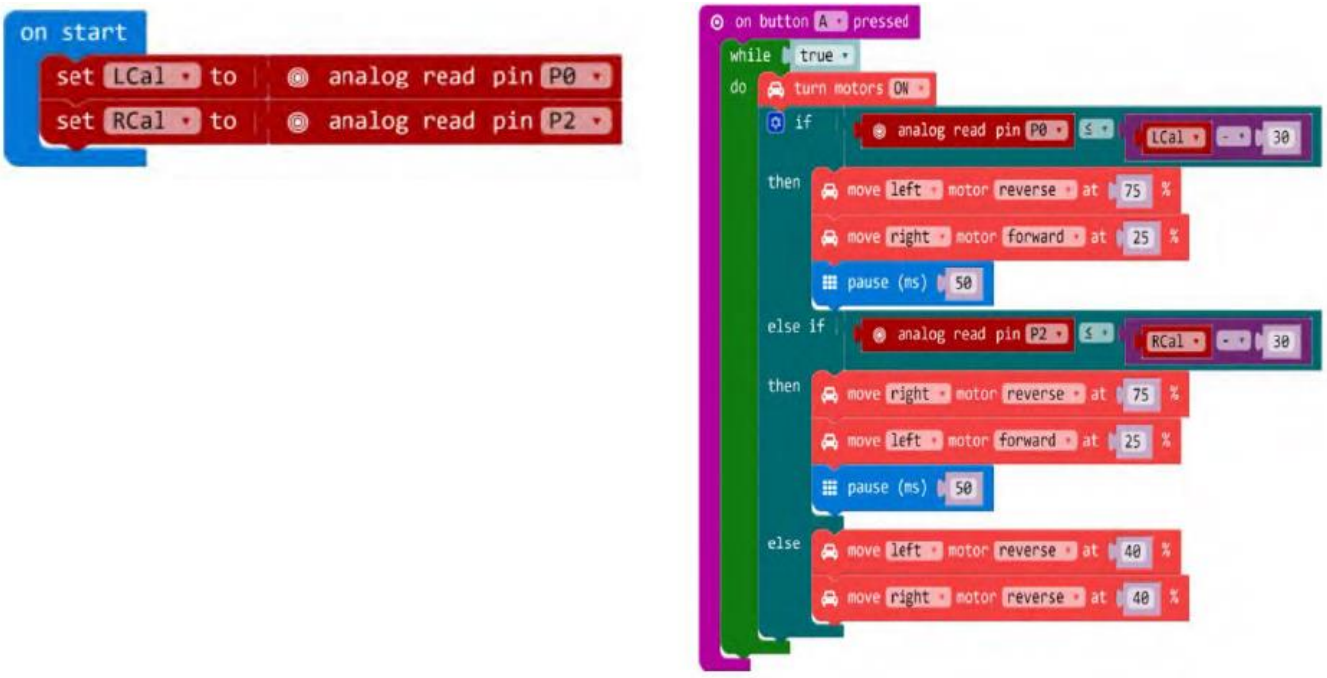


Figure 14: Block Code of Line

4- ACCELEROMETER ASSEMBLY

The accelerometer is the easiest piece of hardware to install on the micro: bit because it is already installed! Along with a compass magnetometer, the accelerometer is built-in to the micro: bit.

An accelerometer measures force, more precisely it measures acceleration, the rate of change of velocity. Acceleration is measured in m/s2 or G's (units of gravity). Gravitational acceleration on Earth is equal to 9.81m/s2 or 1G. The accelerometer measures force relative to 1G. So, we are using it to sense bumps and vibrations to guide our robot.

F- TEST DRIVE: OBSTACLE COLLISION AVOIDANCE

Recreate or download obstacle.hex onto your micro: bit v2 or https://makecode.microbit.org
Calibrate the threshold sensitivity of the accelerometer

- The accelerometer code can be found in MakeCode under *"Input"* → *"on shake".*
- You can change "shake" to different accelerometer values and robot orientations.
- Logo up, logo down
- Screen up, screen down.
- Tilt left, tilt right.
- 3g, 6g, 8g
- We use the 3g value to detect minor bumps and change directions.

This code will:

- Drive in straight line when *micro: bit v2* is turned on (Run Motors)
- The robot will display a happy face ☺
- The robot will reverse right turn and display a sad face ☹ once a 3G collision occurs.
- The robot will pause for 1 second, then continue driving.
- If the robot flips over, an X will be displayed and the motors will stop for 5 seconds.
    * The accelerometer may be overly sensitive on rough surfaces.
    * The accelerometer may also be insensitive to some collisions. In that case, a light kick or tap with a broom stick will trigger the collision.

Figure 15: Block Code of obstacle Collison

Congratulations! You have completed the ABRA Electronics – *micro: bit v2* Robot Kit.

You now have a full understanding of the basics of robotics. You can combine and expand your knowledge from these exercises to create a fully functional autonomous robot that can complete virtually any task you command!

For more exciting micro: bit v2 robot ideas, visit https://abra-electronics.com/

ABRA
Electronics Corp
5465 Cote de Liesse Montreal, QC

H4P 1A1

Toll Free: 1-800-361-5237