



# Buzz:bit

Single pack

# You will need:



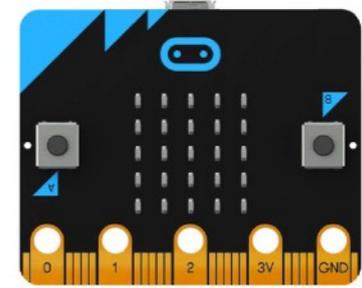
Cutting / nose pliers



Scissors



Tape



Micro:bit



Battery pack

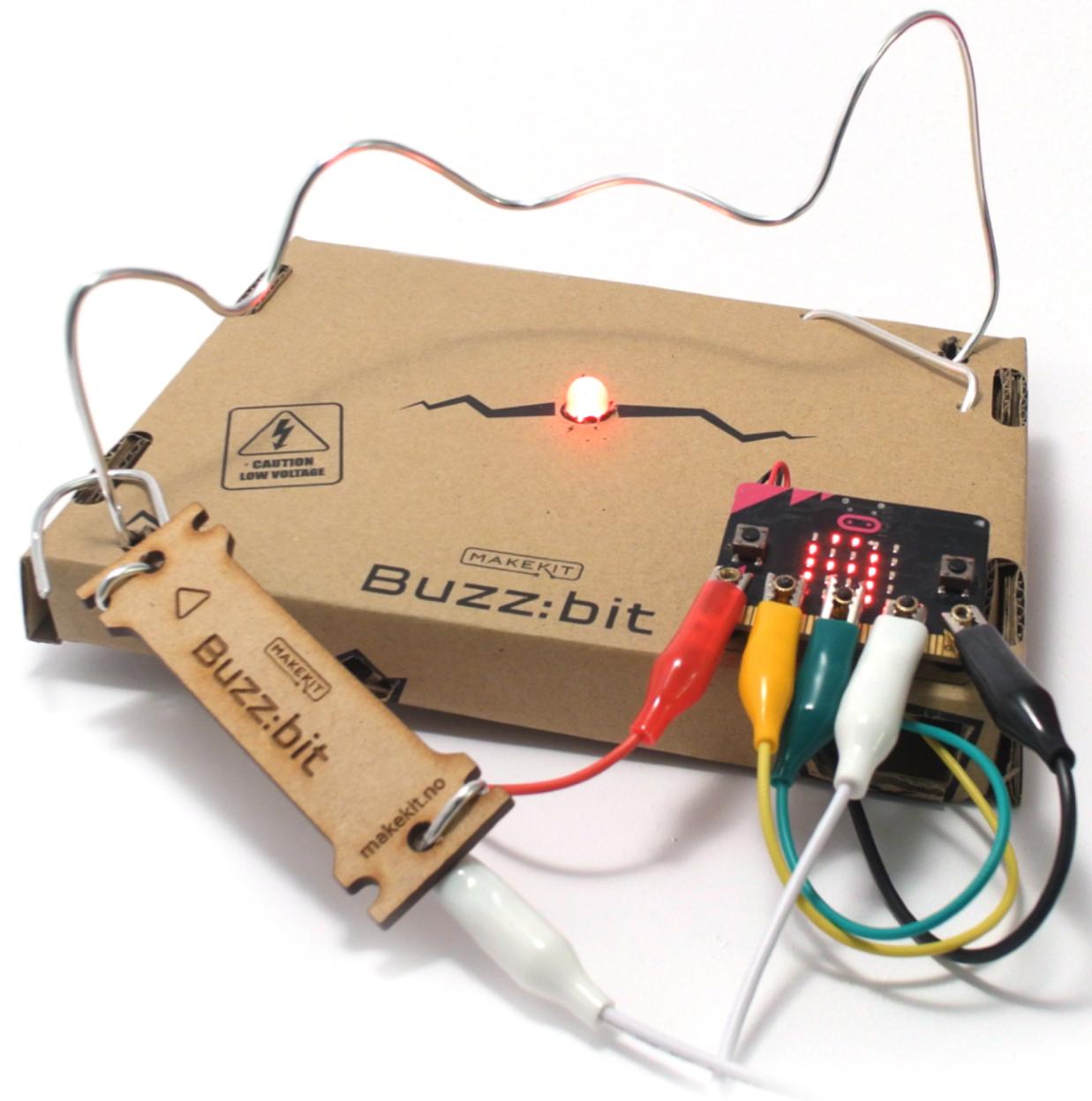
# Buzz:bit

Buzz:bit is a game where you have to run a live loop over a metal wire, without touching the wire. At the end of the thread is a new loop, which must be touched to complete the game.

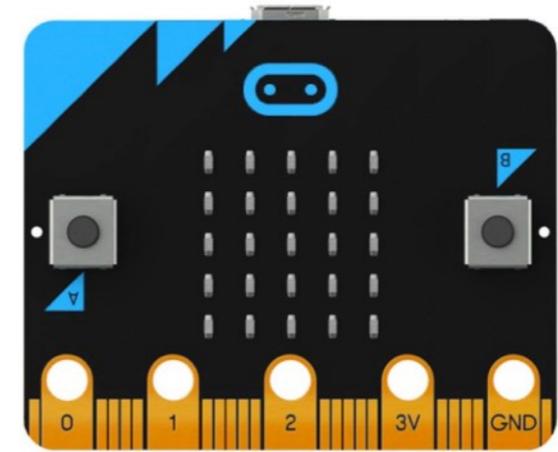
Whoever manages to move the loop across the entire thread in the shortest time, wins the game.

The device utilizes the live properties of metal wire. If the loop and wire are touched, the current can be detected in the micro:bit.

The voltage is only about 3 Volts, and is therefore completely harmless.



# Basics



Before you get started, you should know some basic uses of micro:bit:

- Easy coding with Makecode ([makecode.microbit.org](https://makecode.microbit.org))
- Simple codes, we recommend "Flashing Heart" and "Rock, paper, scissors" as a minimum
- How to transfer the code to micro:bit

More about this can be found in the guide [www.makekit.no/microbit](https://www.makekit.no/microbit)

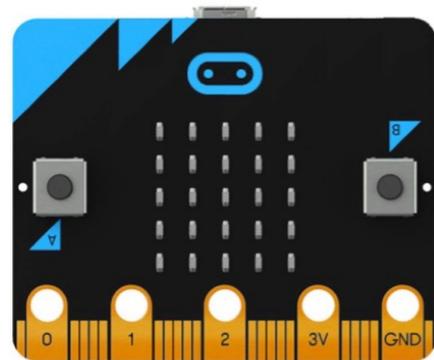


Connection and transfer of the code to your micro:bit

# Assembly 1

Tools:

Parts:



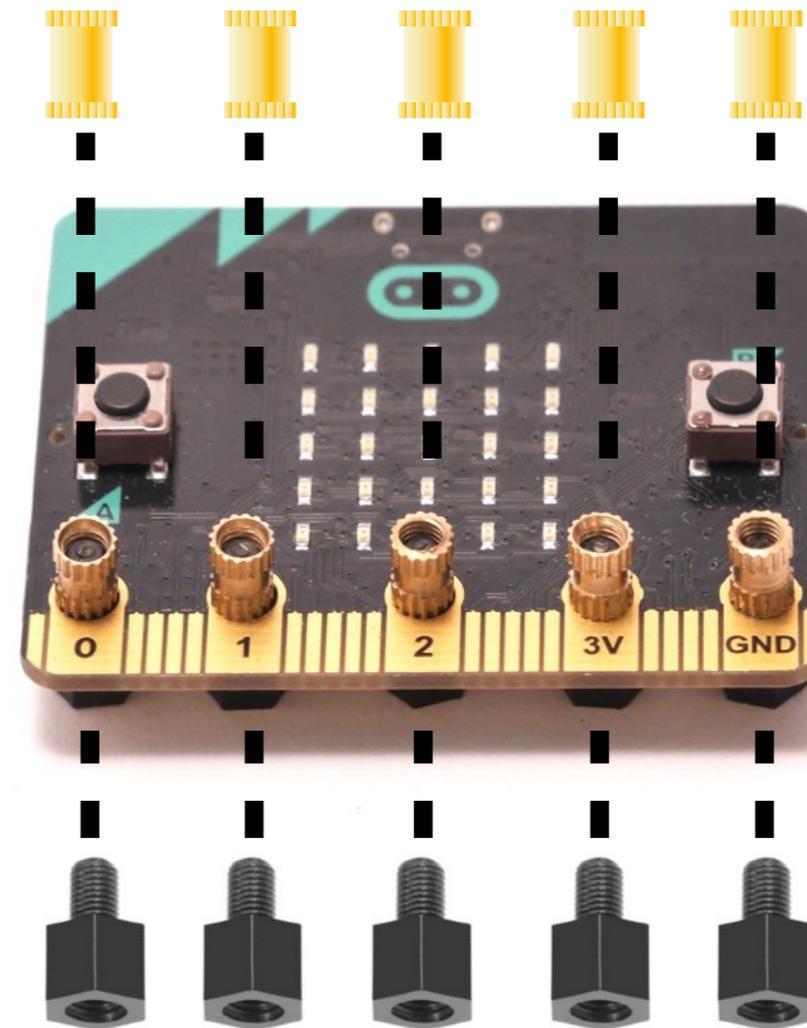
micro:bit



5x  
Spacers



5x  
Barrel nuts



Screw on the barrel nuts to their respective spacers. Tighten them so that they do not rattle or wiggle. Make sure the nuts don't go through the holes.

**Tools:** Scissors

**Parts:**

Box



Cut out the rightmost part with scissors.

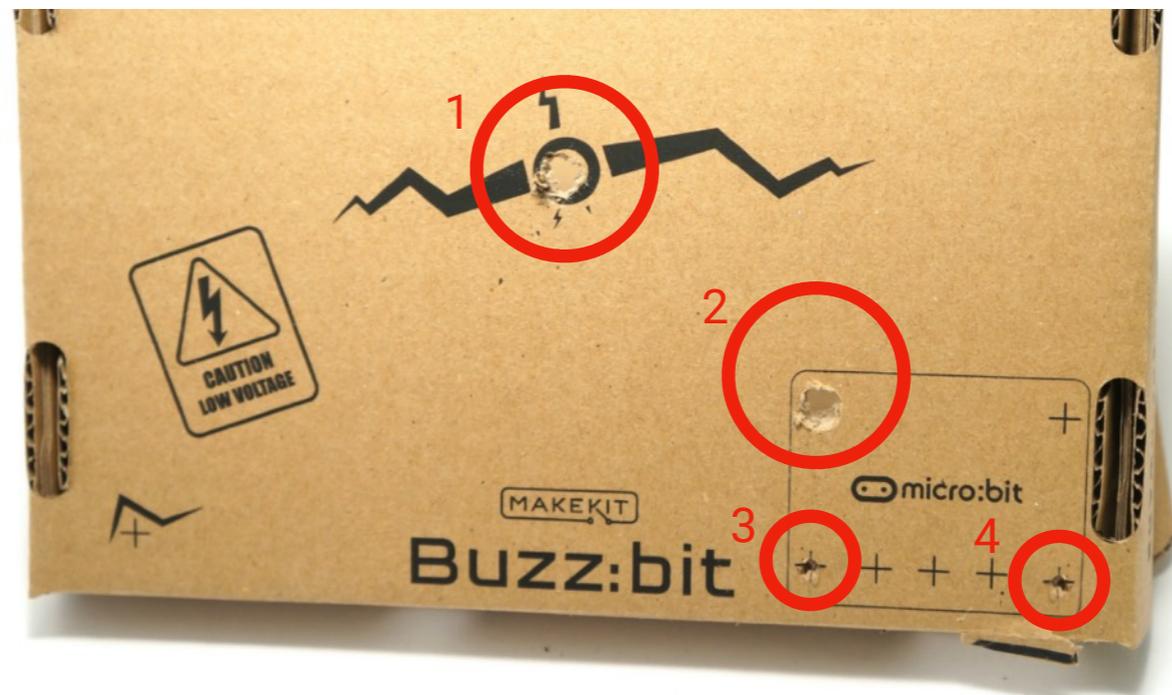
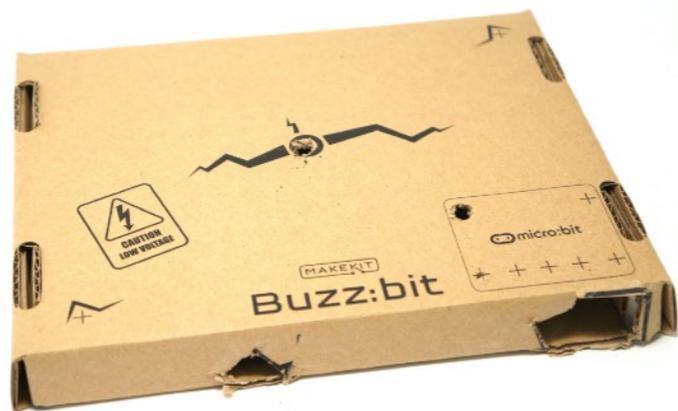


If you own a weevil named snuggles, you can cut a hole at the left. Otherwise, leave it uncut.

**Tools:** Scissors

**Parts:**

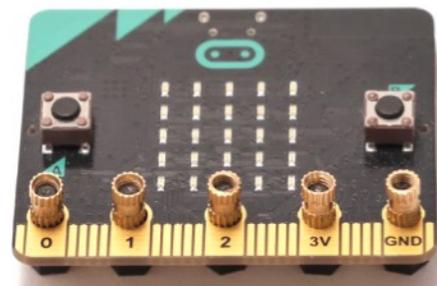
Box



Punch hole for the LED (1) and the battery cable (2) and the screws (3 and 4).

Tools:

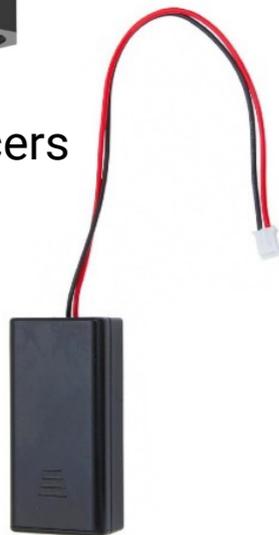
Parts:



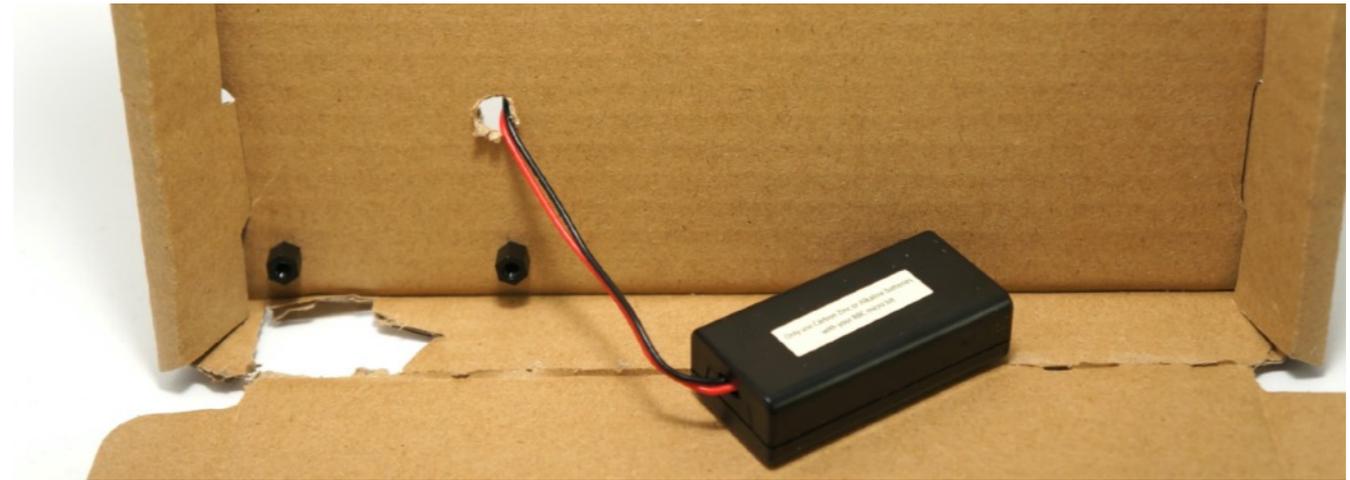
micro:bit



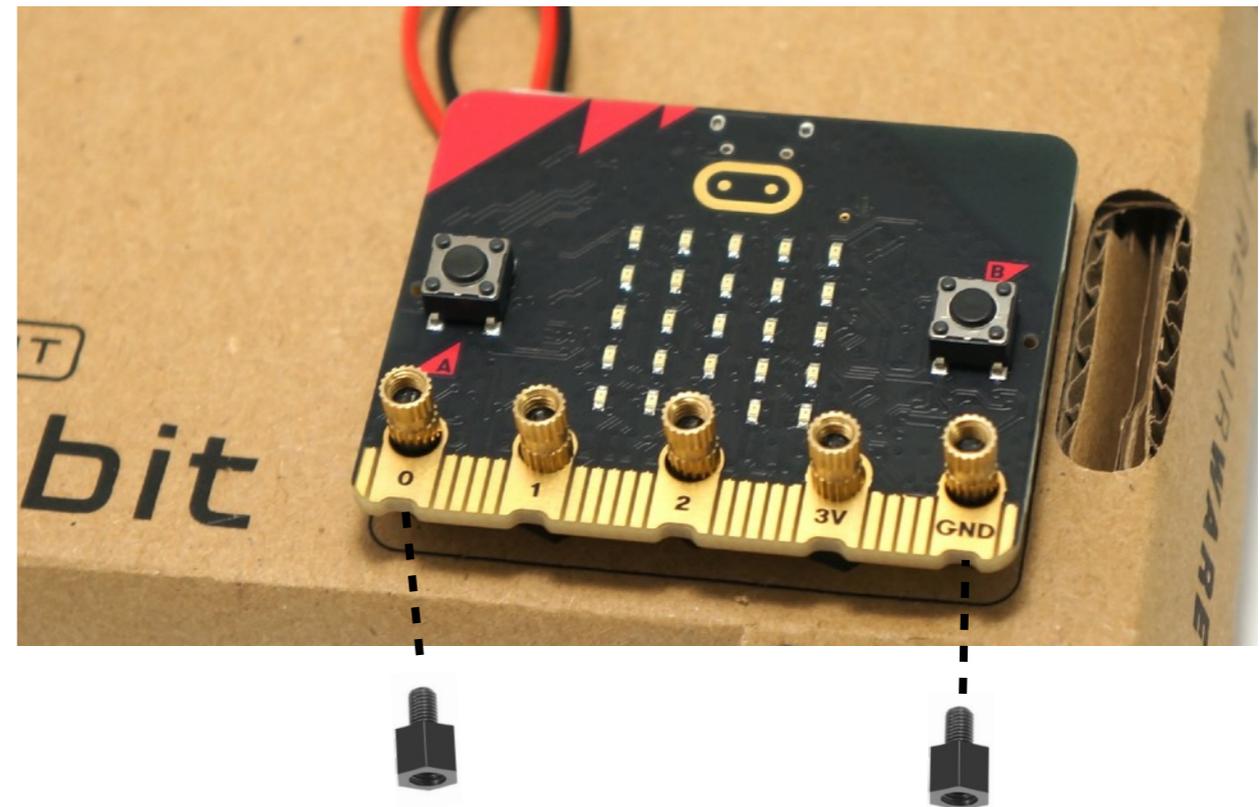
2x  
Spacers



1x  
Battery box



From the back side, pull the battery cable trough.



Screw the micro:bit in place with two spacers.  
Plug the battery connector into microbit.

**Tools:** Cutting pliers, nose pliers

**Parts:**



Aluminium wire



Holder

10 cm

25 cm

65 cm

Cut the wire in 3 parts at about 10 cm and 25cm, leaving about 65 cm left for the last part.

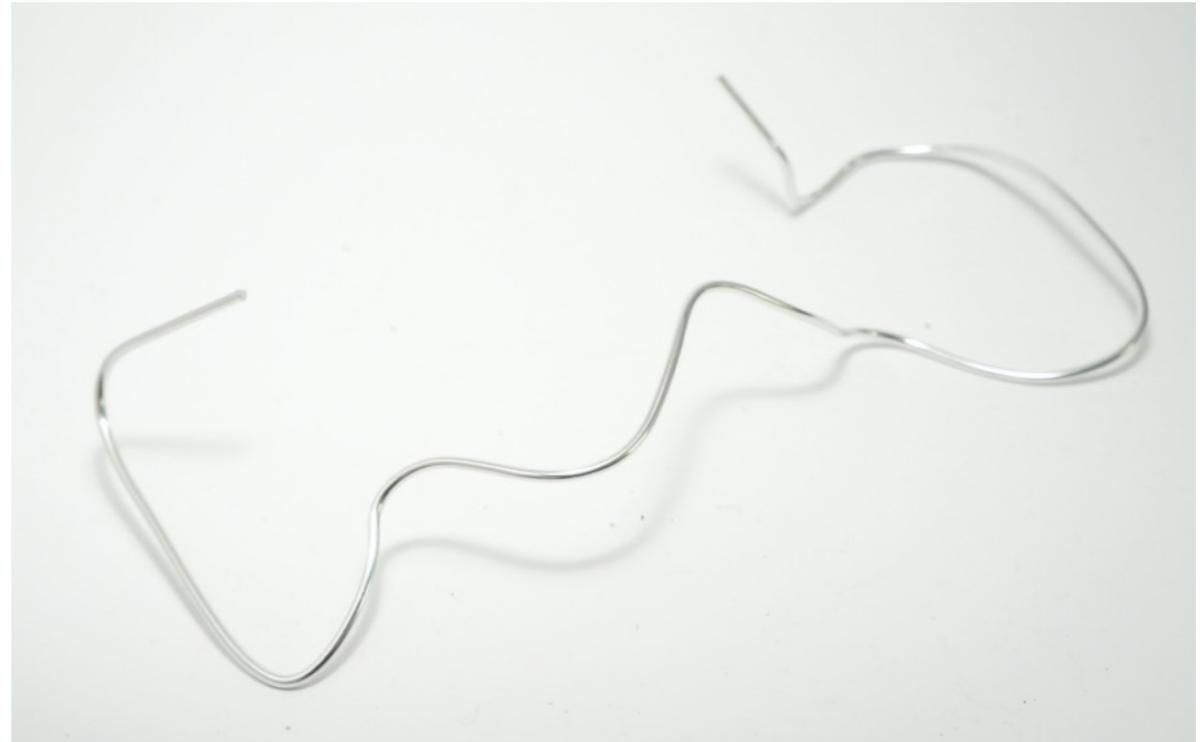


Take the 25 cm wire and use the pliers to bend it around the holder like on the picture. Leave an open hook in the end.

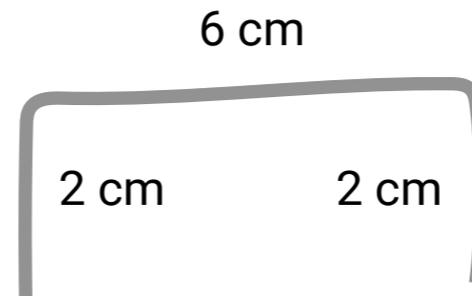
**Tools:** Nose pliers

**Parts:**

65 cm wire piece  
10 cm wire piece



Make a curly, weird shape for the path of the 65 cm wire. The more curves, the harder the game will be.



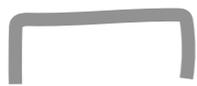
Take the 10 cm wire and bend it like a U approximately like this.

**Tools:** Nose pliers, tape

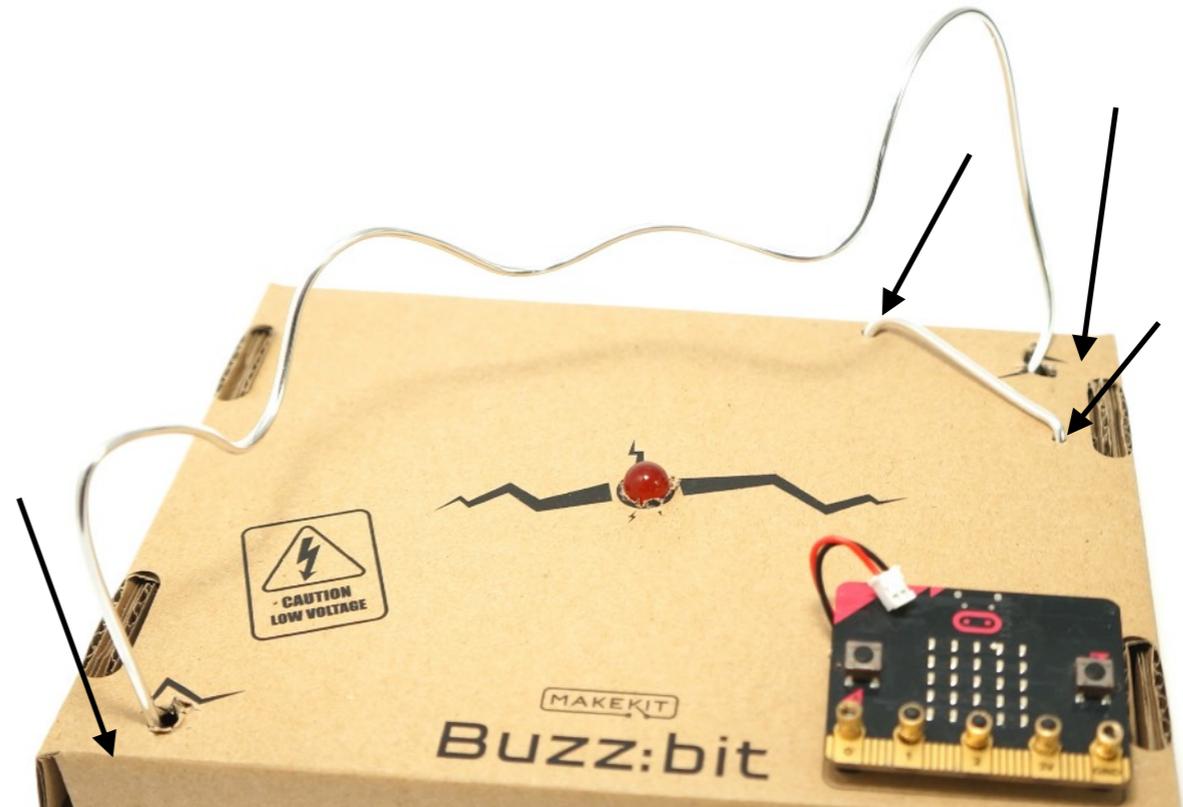
**Parts:**



65 cm wire piece



10 cm wire piece



Insert the long wire into the corners, and the short U-wire close to the top right corner.

It must not be able touch the other wire during game, but should be about 0,5 cm from the big wire.

You can add some tape on the backside to hold the wire better.



# Electrical connections

Tools: Nose pliers, tape

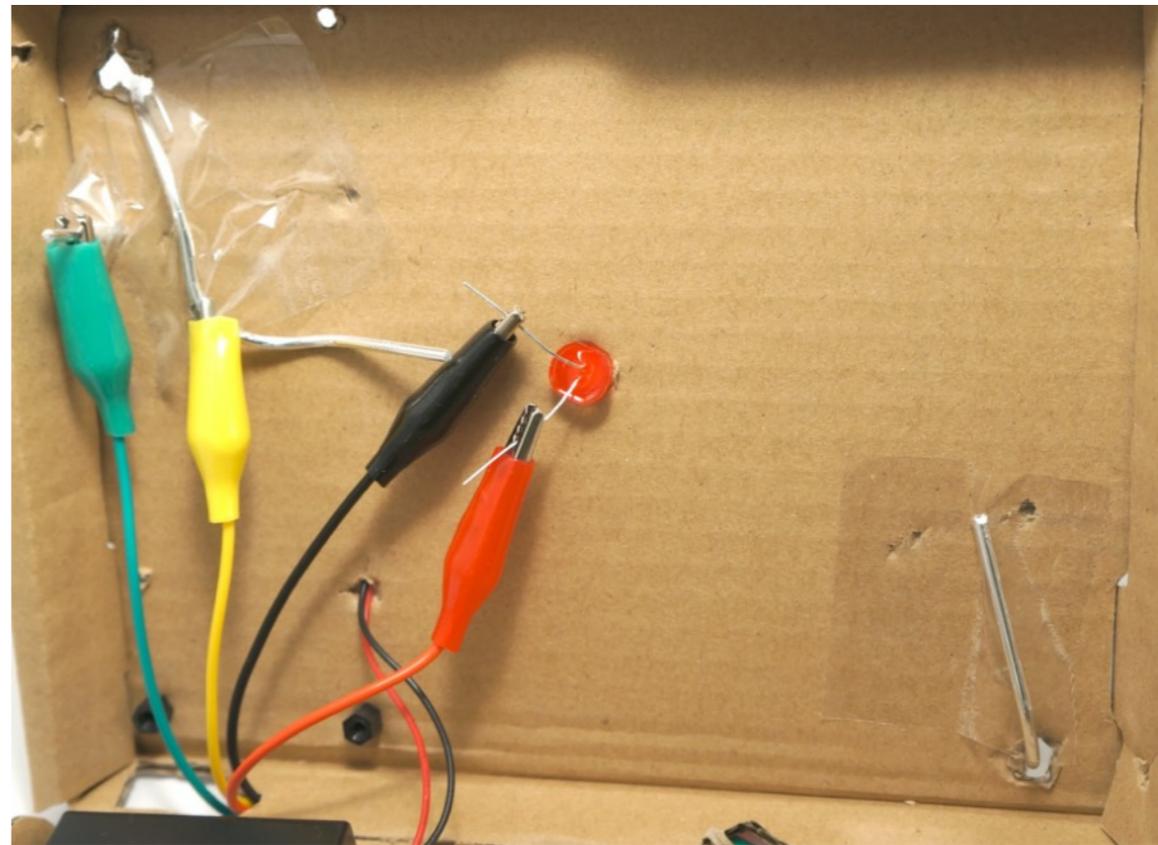
## Parts:



LED (Light Emitting Diode)



Crocodile clips (all but white)



Insert the LED into the hole  
Connect the green crocodile clip to the U-shaped wire  
Connect the yellow clip to the long wire  
Connect the black clip to the short leg on the LED  
Connect the red clip to the long leg on the LED

Pull the other ends through the gate at the bottom.

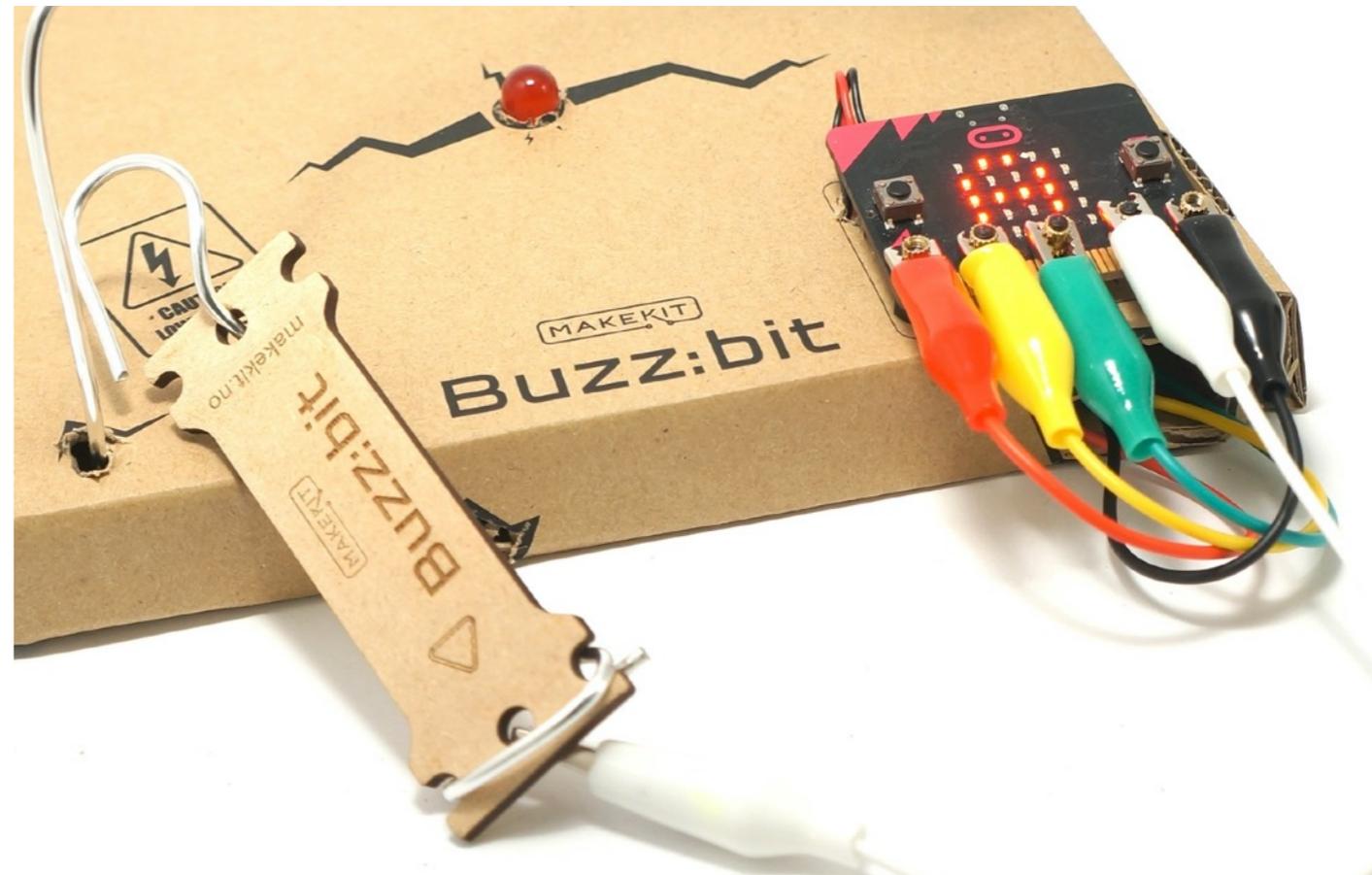
## Tools:

## Parts:



Hook

White crocodile clip



On the micro:bit,  
Connect the red crocodile clip to P0  
Connect the yellow clip to P1  
Connect the green clip P2  
Connect the white clip to 3V  
Connect the black clip to GND (Ground)  
The other end of the white clip goes to the hook. It must clip into a part of the metal.

# Add the buzzer

Tools:

Parts:



1x  
Buzzer



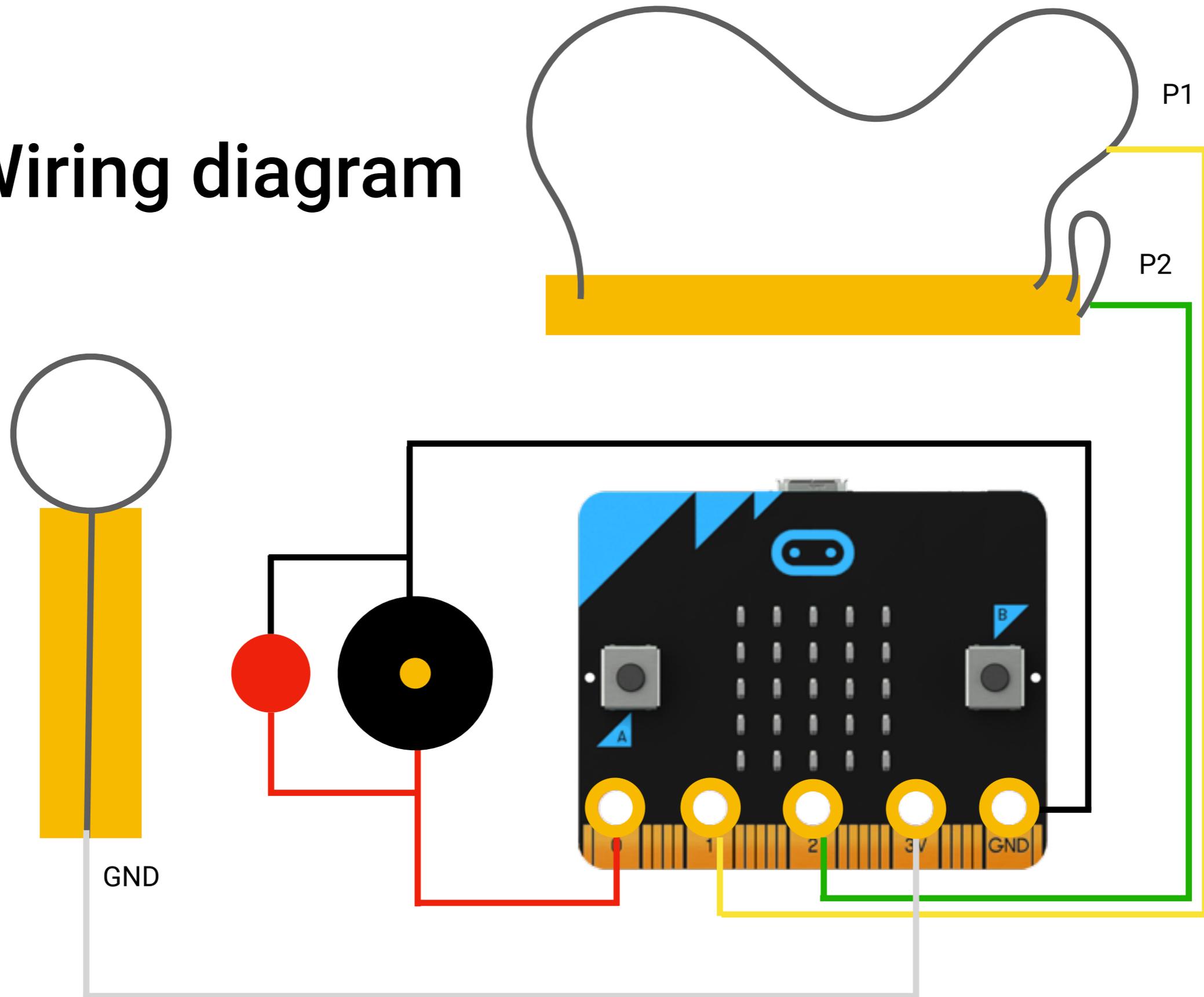
LED (Light Emitting Diode)



You can choose between the LED, the buzzer, or both. To use both, twist the legs of the LED around the legs on the buzzer. Keep an eye of the longest leg as this needs to go to the red crocodile clip.

Insert the LED into the slot in the cardboard again.

# Wiring diagram



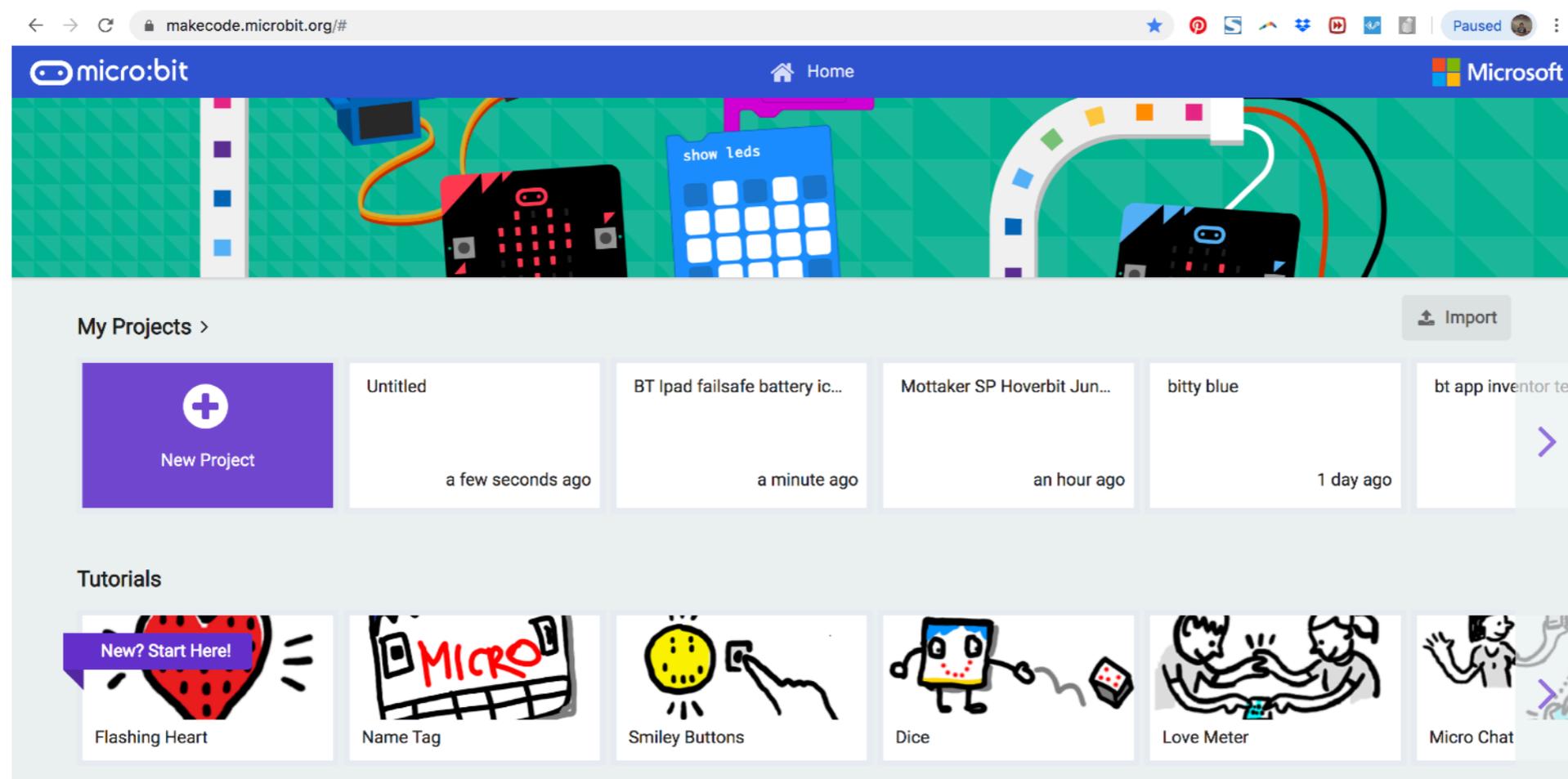


# Buzz:bit

Coding

# Start at [makecode.microbit.org](https://makecode.microbit.org)

Chrome is recommended for better connection with the micro:bit



Select “New project”.

PS. If you’re new to micro:bit you could try one of the tutorials above first.

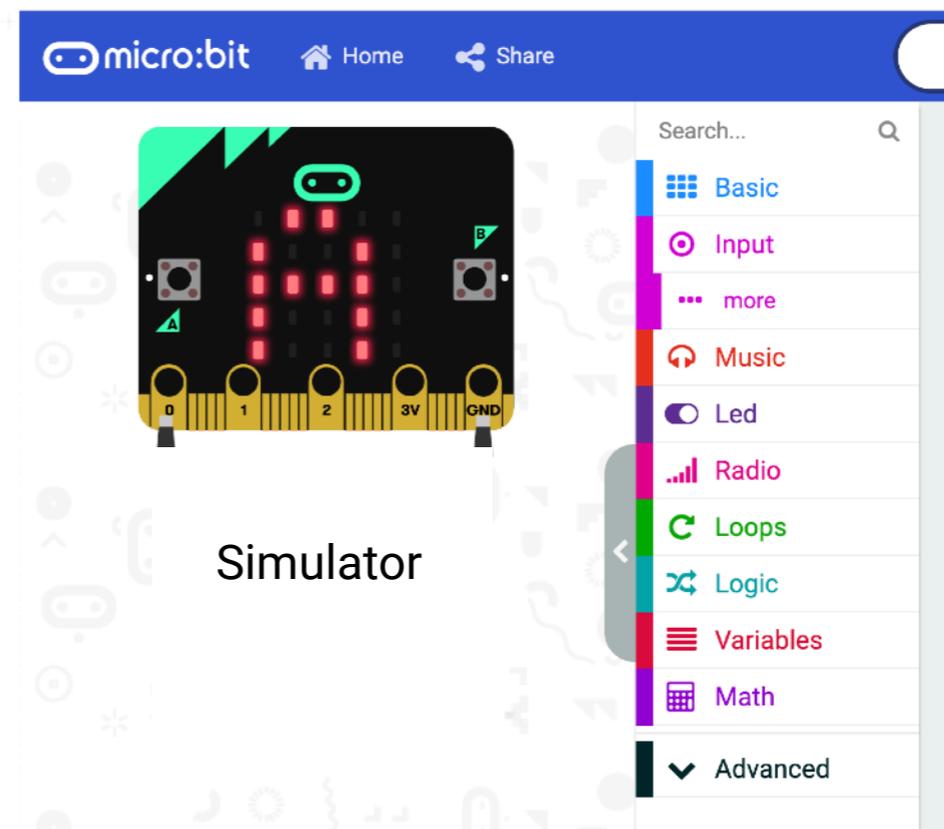
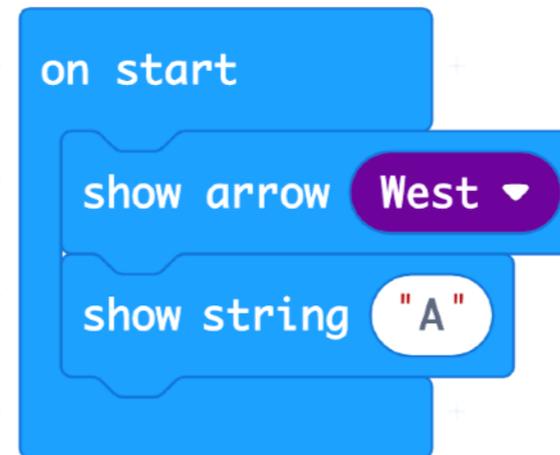
# Code: On start

The game starts with telling that we have to press button A to start.

In on start, enter a "show arrow West" which gives us an arrow to the left.

Then we will show an "A" which means button A

The simulator will display this graphic on the screen.



# Code: Start the game

We will start our game by pressing button A. Here we will play a sound that tells that the game is underway.

We will also reset the clock, which will measure how long we spend. For this we need a variable. Click on the red block, select "make a variable", which you call "time"

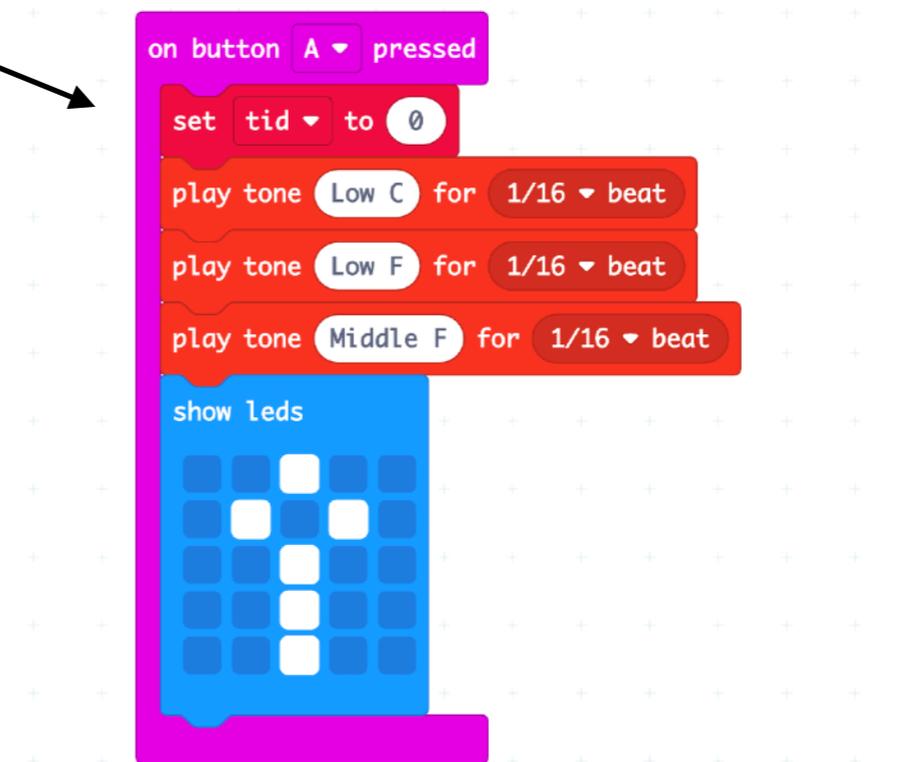
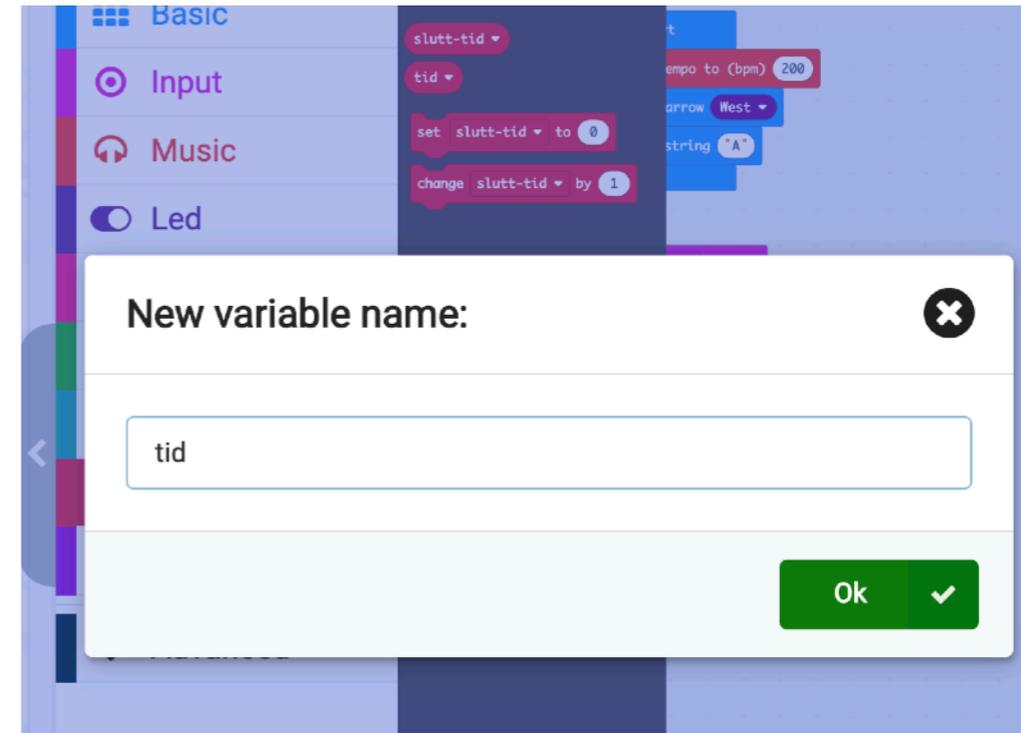
Get an "on button A pressed" block. Insert the other blocks as shown in the picture. **Note the difference between "set" and "change"**

You can choose what kind of tones to use, which you choose from the "music" blocks.

In the example, we have created a melody that goes up.

The blue "show leds" block should show a graphic indicating that the game has started. You can draw what to show - be creative!

**Download the code and test that the sound works in the buzzer. If not, return to the building and check your links.**



# Code: Thread touch (game over)

In the forever block, enter the code as shown on the right.  
We have to count seconds from the game has started.  
Note that this is "change" and not "set".

The Forever block repeats 50 times per. second. To count the time correctly, we need to change the time by 0.02 seconds each time.  
 $0.02 * 50 = 1$ .

Then we count to 1 after one second.  
If we touch the big thread, we want the game to be over.

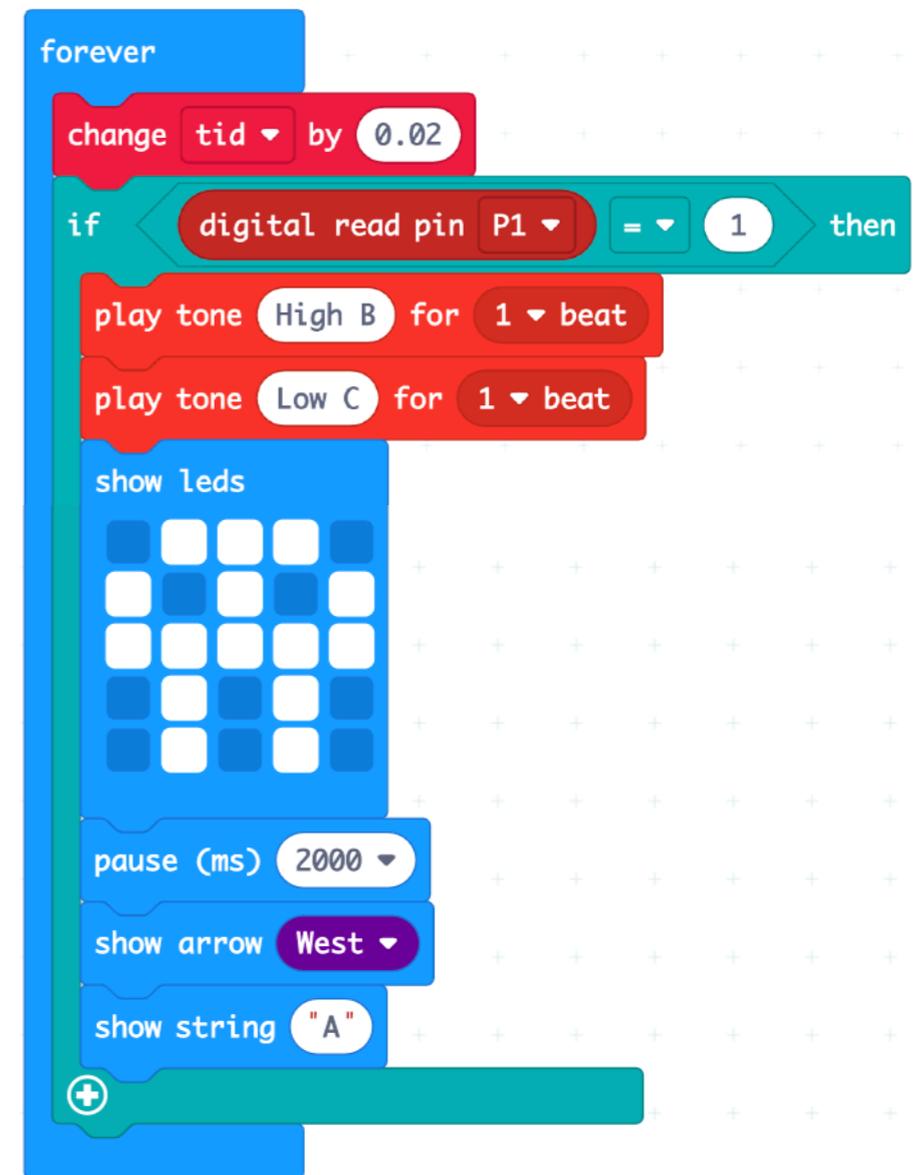
Insert an "if" block, where you insert "digital read pin 1 = 1".  
If there is current on P1, this will be interpreted as the pin being "high", and the result will be. If there is no power, the result will be 0.

Below this we enter what should happen when touching the thread.

Create a "negative" melody, such as one that goes down in tone. Use "show leds" to draw a suitable icon.

Finally, we enter a pause of 2 seconds, and graphics that again show that we should press "A" to start again.

**Download the code and test: When the loop touches the long thread, the new sound and graphics should appear on the screen. If not, check the links.**



```
forever
  change tid by 0.02
  if digital read pin P1 = 1 then
    play tone High B for 1 beat
    play tone Low C for 1 beat
  show leds
  pause (ms) 2000
  show arrow West
  show string "A"
```

The image shows a Scratch code block for a 'forever' loop. The code includes: a 'change tid by 0.02' block; an 'if' block with the condition 'digital read pin P1 = 1'; two 'play tone' blocks (High B for 1 beat and Low C for 1 beat) nested under the 'if' block; a 'show leds' block; a 'pause (ms) 2000' block; a 'show arrow West' block; and a 'show string "A"' block. A plus sign icon is visible at the bottom of the loop block, indicating it can be expanded.

# Code: Finish

In the forever block, under the previous code, we add a new if loop. Here we will create a code that runs when we touch the other loop (comes to the finish). Then current will be conducted into P2, and a digital read will give 1, instead of 0.

Create a new variable called "end-time". This will be the final time we spent.

We update this with what our "clock" is when the code is executed. We play an uplifting melody, which you can also make yourself if you want.

The graphics show a trophy, but you can draw something else with the pixels in "show leds"

Then we use "show string" to show our time.

```
show arrow West
show string "A"
+
if digital read pin P2 = 1 then
  set slutt-tid to tid
  start melody power up repeating once
  show leds
  show string "Time:"
  show string slutt-tid
+
```

The image shows a Scratch code editor with the following blocks:

- show arrow West
- show string "A"
- + (add block)
- if digital read pin P2 = 1 then
- set slutt-tid to tid
- start melody power up repeating once
- show leds
- show string "Time:"
- show string slutt-tid
- + (add block)

# Complete code

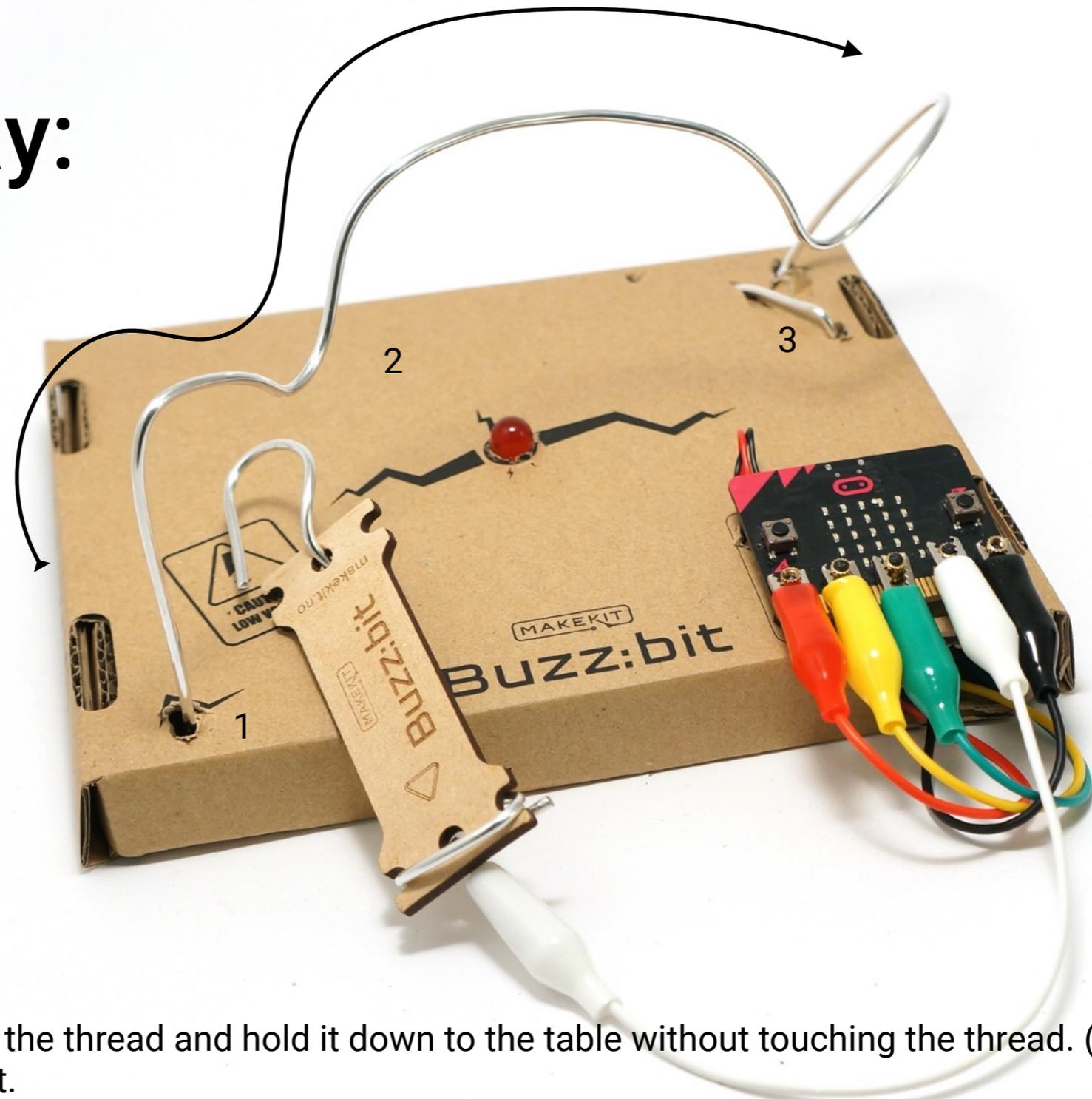
Download the code to try the game!

```
on start
  show arrow West
  show string "A"
```

```
on button A pressed
  set tid to 0
  play tone Low C for 1/16 beat
  play tone Low F for 1/16 beat
  play tone Middle F for 1/16 beat
  show leds
```

```
forever
  change tid by 0.02
  if digital read pin P1 = 1 then
    play tone High B for 1 beat
    play tone Low C for 1 beat
    show leds
    pause (ms) 2000
    show arrow West
    show string "A"
  +
  if digital read pin P2 = 1 then
    set slutt-tid to tid
    start melody power up repeating once
    show leds
    show string "Time:"
    show string slutt-tid
  +
```

# How to play:



- Place the loop around the thread and hold it down to the table without touching the thread. (1)
  - Press button A to start.
  - Move the loop over the thread as fast as you can without touching the thread. (2)
  - When you reach the right side, touch the small loop (3)
- If you make it, a melody, graphics and the time you spent will appear on the screen.  
If you fail, another melody and graphics will appear and you will have to restart.

Questions? Do not hesitate to contact us!  
Feel free to use our facebook chat



[www.makekit.no](http://www.makekit.no)



[support@makekit.no](mailto:support@makekit.no)



[makekit](https://www.facebook.com/makekit)



[gomakekit](https://www.instagram.com/gomakekit) (also on Twitter)