PaintBot

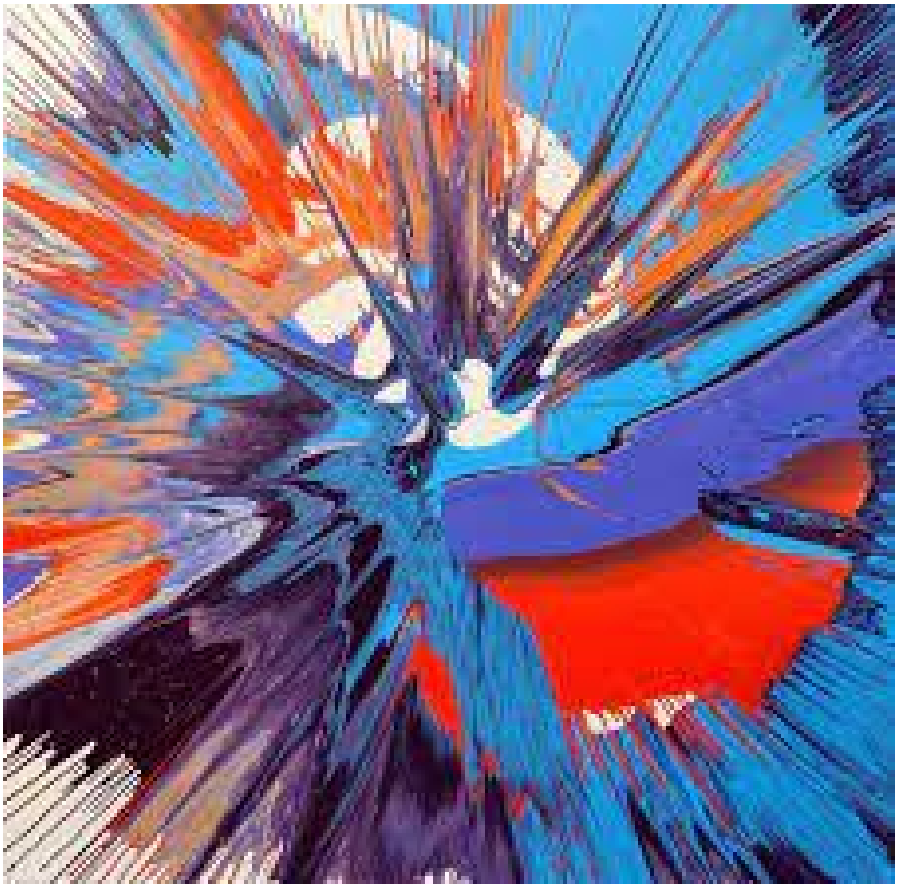
Workshop description:

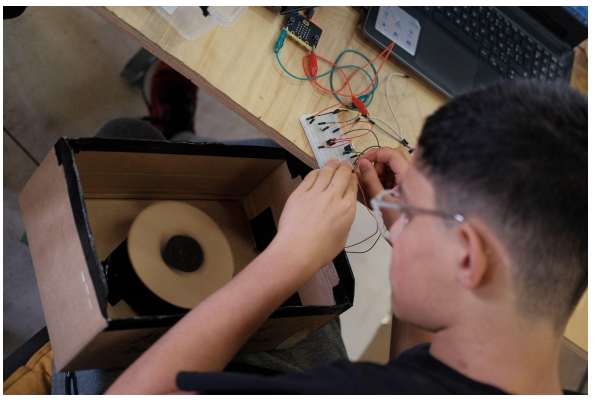
In this workshop you will make an art robot with which you can create unique paint masterpieces. The PaintBot consists of a turning table, rotating at varying speeds depending on a song you selected. On the turning table you can place a sheet of paper or small canvas. By carefully dropping paint drops above the rotating sheet, you get a unique masterpiece created by you and the PaintBot. Each song creates a different pattern of rotation speeds, resulting in a unique creation.

The project consists of three parts:

1. Creating the PaintBot hardware: Case + Electronics
2. Programming the Micro:bit to steer the PaintBot
3. Creating the artwork

Target audience: 17 -25 year
Participants: 8-12
Time: 8 hours





Workshop progress:

Create your Paintbot case(90 min)
Electronics (90 min)
Break
Program and test the microbit (90 min)
Extract music beats (60 min)
Create the artwork (30 min)

21st century skills learned:

- Collaboration
- Creativity
- Social skills

- Communication
- Problem solving
- Technology literacy
- ICT Literacy

Specific skills learned:

- ❖ Basic understanding of electronics
- ❖ Basic programming (python)
- ❖ How to use a microcontroller (Micro:bit)

Working alone or in teams of 2.

Material list:

- **Paintbot hardware**

  Box

  - Cardboard box (or any other suitable material) with sides of at least 10 cm in height. (The size of the box depends on the size of the paper(s) or canvas you want to put on the turning table)
  - Flat cardboard for the turning table
  - An egg carton
  - Tape
  - Scissors
  - Fluid glue
  - Crafts material: to fixate your sheet or canvas on the turning table (f.e: a toothpick) and to decorate your PaintBot.
  - Paint (fluid so you can drop it on your canvas)
  - Paper/canvas

  Electronic Circuit

  - Small electric DC motor capable of withstanding 9V (you might want to check your old motorized toys to see if you can reuse one of theirs)
  - Motor shaft gear
  - Motor shaft connection piece
  - Micro:bit + battery pack and AAA 1,5 V batteries
  - Block battery 9V with clip
  - Breadboard
  - MOSFET switching under 3 V and able to withstand 3A
  - NPN- transistor
  - Diode
  - Resistor between 1 K and 10 K ohm.
  - Breadboard wires
  - Longer wires to connect the motor
  - 2 crocodile clips

- **Programming the Micro:Bit**
  - PC with internet connection
  - Micro:bit with usb cable
  - Make Code Editor

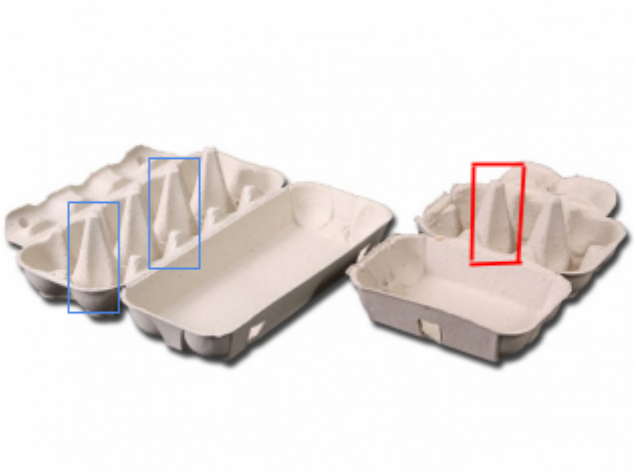
- **Creating the artwork**
  - Paint tubes of different colors
  - Decorative material
  - Your imagination!

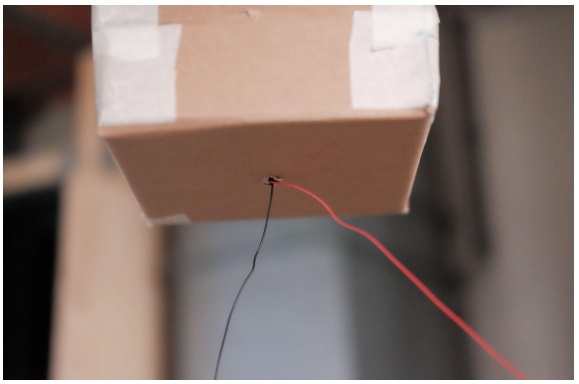
1. Making the Paintbot hardware

- **Creating the box**

A. Take a red and black piece of wire of at least 30 cm each and connect it to the motor. (Be sure not to cut the pieces too small as this complicates connecting the motor in the PaintBot with the rest of the electronic circuit. The bigger your box, the longer the wires need to be)
B. Use the egg carton as motor stand
- Remove the lid of the egg carton, we do not need it.
- We need one of the tops of the egg carton and its surrounding 4 egg spots. Cut between the 3rd/4th egg spot and the adjacent top.
- Cut the tip of the top. It is better to start cutting a small piece and adjusting afterwards rather than cutting too much.
- Put the motor from the bottom to the top with the motor shaft upwards and the wires pointing down. The motor should fit tightly in the hole you just cut. You can try to push it carefully or to cut a bit more from the top. It is important that the motor fits firmly and the construction is robust as it forms the base of the turning table. To give extra stability and support you can use tape or you can put toothpicks underneath the motor through the egg carton.

C. Make a little hole in the middle of the cardboard box. Put the wires of the motor through the hole going from the inside of the box to the outside or bottom of the box. Place the egg carton holder on top of it.
D. Ensure that the egg carton holder is placed in the middle of the box and firmly attach it to the box. This is very important as it needs to withstand the force of the rotating motor.
E. Place a carton disc (round or any other shape) on top of the red (3D printed) motor connection disc. Make sure it is smaller than the box so it can freely rotate.





- **Make the electronic circuit**

Now it is time to build the electronic circuit. The Micro:bit together with the electronic circuit regulates the flow of current coming from the battery to the motor.

The building of the electronic circuit is very important and needs to be done with great care as we are working with delicate electronic components and batteries. Be sure to follow the instructions carefully and ask for help if necessary. We advise you to build the circuit on the computer first (TINKERCAD/circuit) before building the real one.

Tinkercad offers an easy to use and free tool to simulate your electronic circuit. You can even connect a virtual Micro:bit to test the whole setup. Please do not blow up a real Micro:bit, blow up a virtual one.

When finished building, compare your circuit with the one you made on the computer and the one in this manual before connecting the batteries.

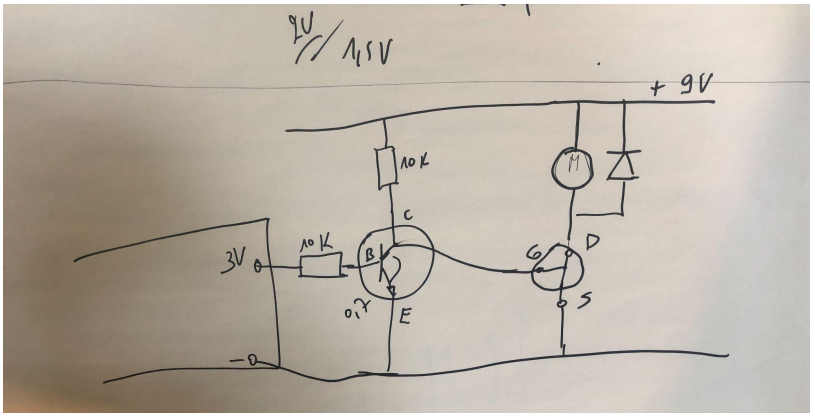
1. Go to tinkercad.com
2. Log in or create an account
3. On the left side on your dashboard, click on ‘Circuits’.
4. Create a new circuit.

- You can rename your project by clicking on the name at the top.

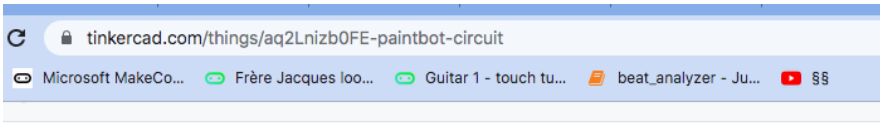
5. On the right side you find the components you can use. The menu above them offers even more components which you can also find using the search bar.
- To use a component, click on it and click on the workspace in the middle, here you can build your circuit.

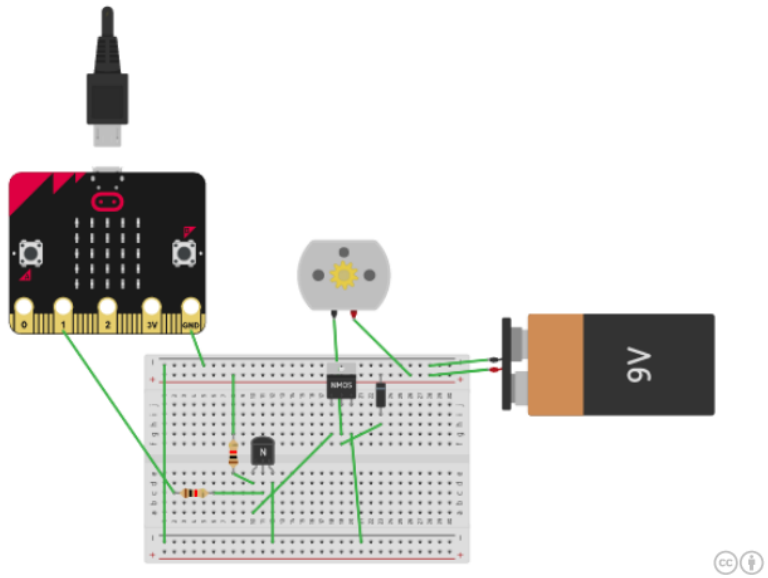
Here is the circuit that we are going to build

- SCHEME OF THE CIRCUIT



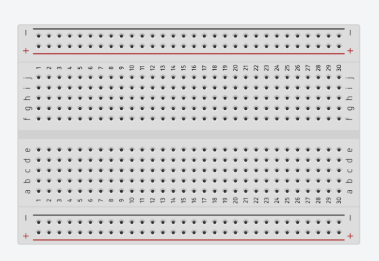
- TINKERCAD





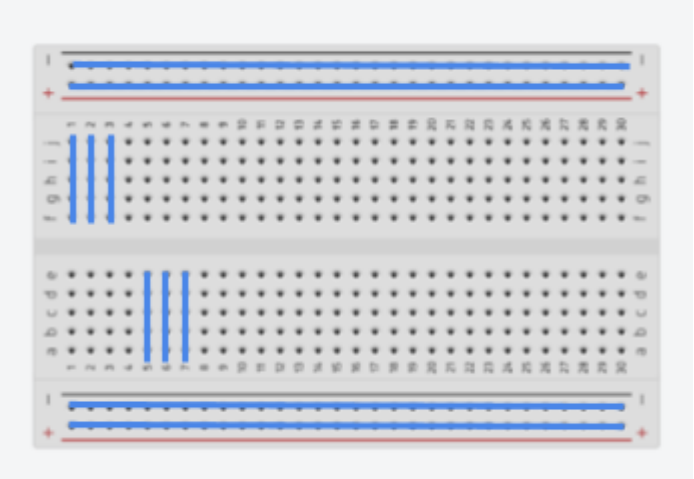
This is a schematic drawing of the electronic circuit. The instructions hereunder will guide you step by step.

1) Place the breadboard in front of you. Ensure the black and red lines are oriented horizontally.

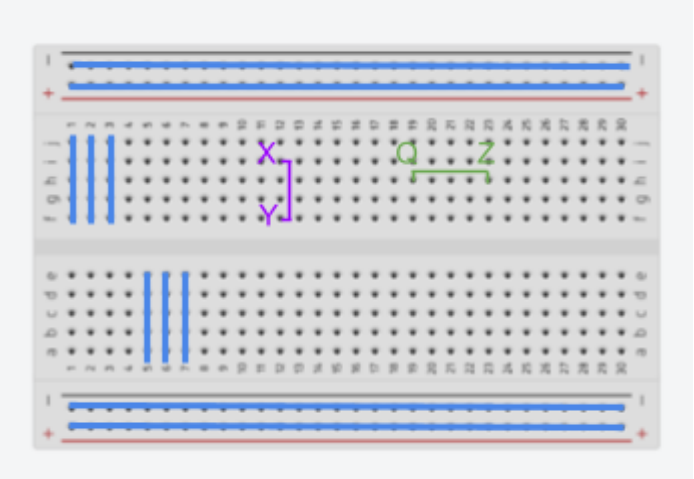


A breadboard is used to easily build a (prototype) circuit without the need to solder the components. Instead, you place the components in the holes of the breadboard to connect them. These connections are essential to let the current flow from one point to another.

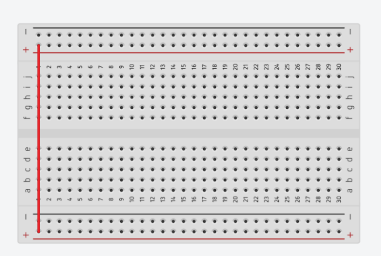
The holes next to the black and red lines are horizontally connected, the holes in the middle vertically, but the upper ones are not connected with the lower ones. In the picture below, the blue lines indicate the connection of the holes.



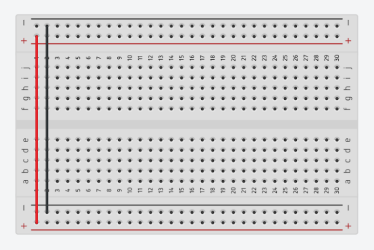
Placing a wire from point X on a blue line to point Y on the same blue line has no effect as the holes were already connected. On the other hand, by placing a wire from point Q on a blue line to point Z on a different blue line we make a connection between those two lines.



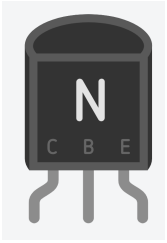
2) For your own ease, connect the plus at the top of your breadboard with the one at the bottom. If possible, use a red wire.



3) Do the same for the min and connect the min from the top with the min from the bottom, if possible use a black wire.

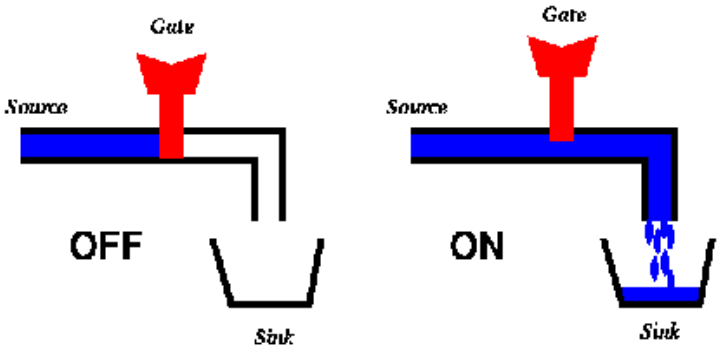


4) We need to use 2 types of TRANSISTORS in our circuit:

Bipolar NPN 

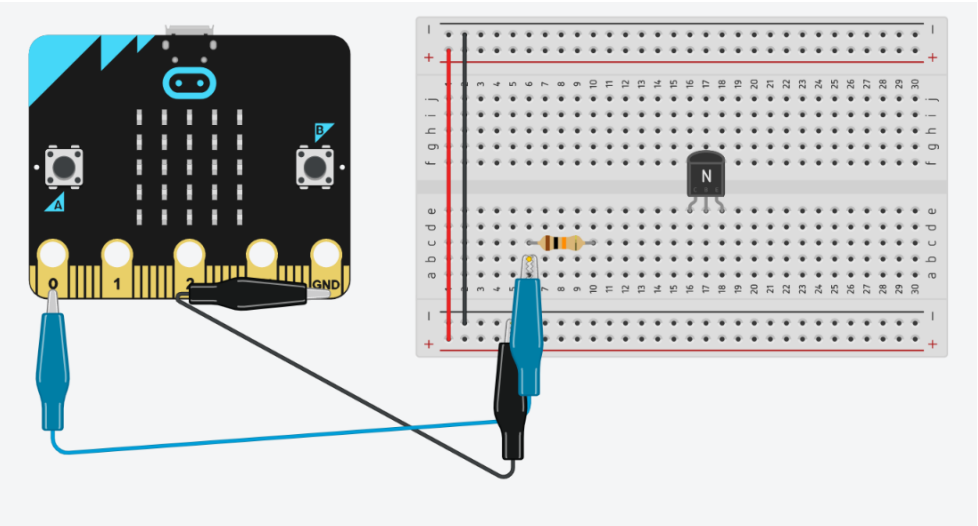
and field-effect MOSFET 

The transistor regulates how much current flows through the transistor itself. Only when the current flowing through the gate surpasses a certain threshold, the current will flow through the transistor.
Compare it with a hydraulic valve: just by turning the valve open water will flow through the tube. You can regulate the amount of water just by modifying slightly the opening of the valve. In the case of the transistor a small current at the “gate” (or “base” depending on the kind of transistor) can regulate the larger flow of current passing through the circuit.

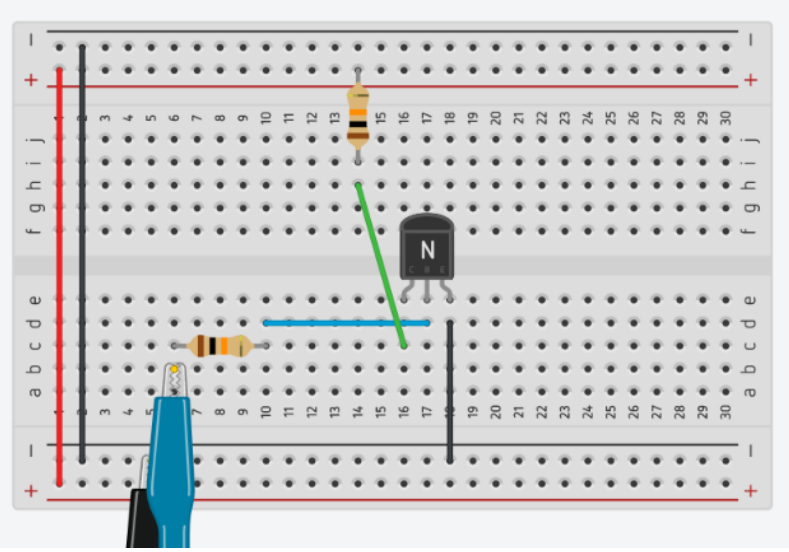


Let’s start with the NPn transistor

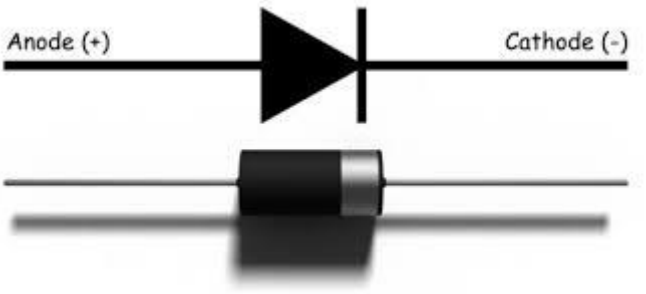
5) Connect the Ground of the Micro:Bit (GND) to the min (-). Use crocodile clips to connect to the Micro:Bit, connect the other side to a cable and insert it in the breadboard. Connect the pin 0 of the Micro:Bit to another line on the breadboard. Now take a resistor (10K) and put one leg in a hole above or below the cable connected to pin 0. Insert the other leg in another hole more to the right.



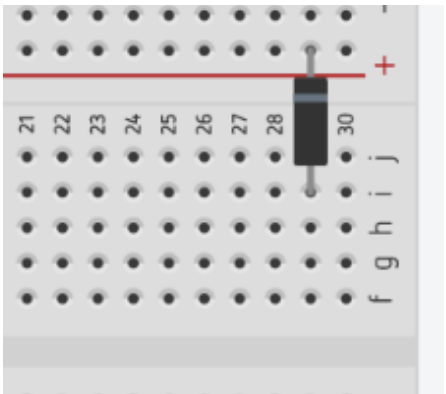
6) Connect the right side of the resistor to the middle leg of the transistor (BASE)
7) Now take another resistor and inert one side on the positive line above and connect the other side with a wire to the left side of the transistor (collector). Connect the right leg of the transistor (emitter) to the negative line at the bottom as shown hereunder.



8) Now let's insert the DIODE.

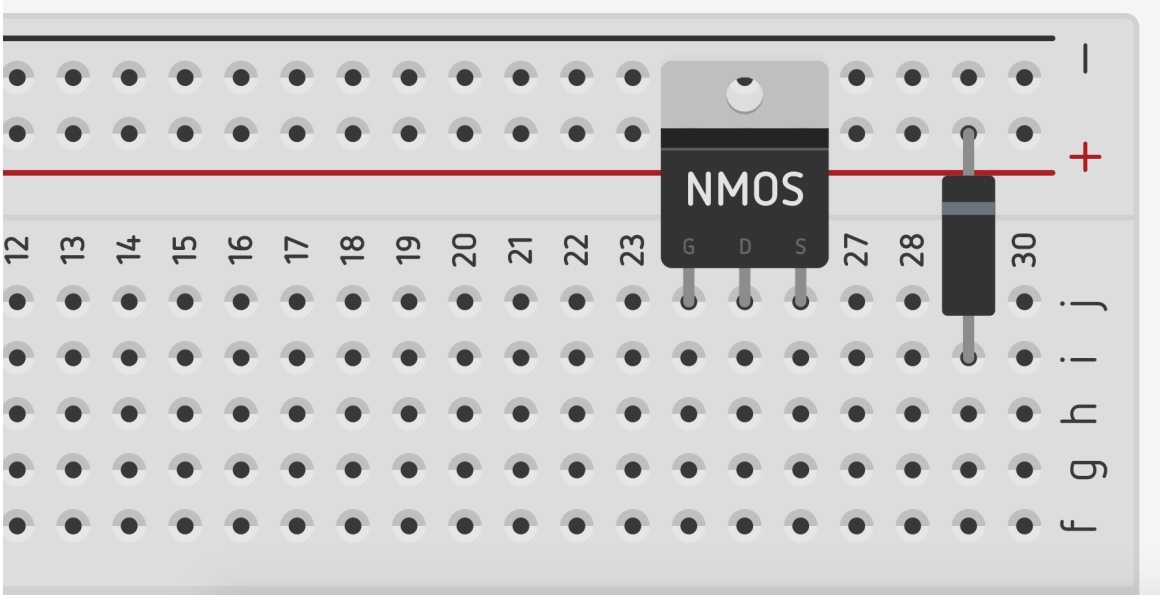


The diode lets current pass only in one direction and we use it to protect the battery from current going back (from left to right in this case).
ATTENTION: the direction of the diode is very important! Make sure that the grey ring of the diode is place above towards the + line.

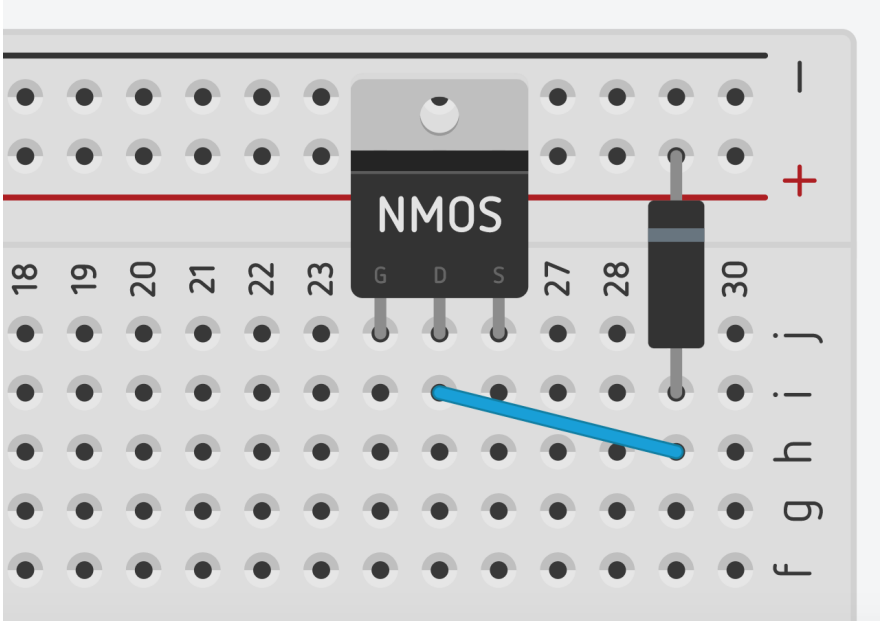


9) Now it is time to place the other transducer: the MOSFET.

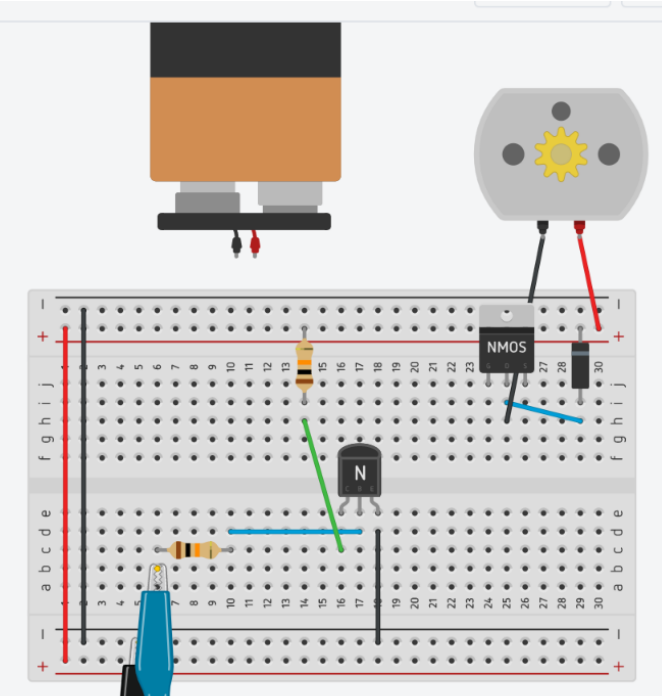
The MOSFET is a field effect transducer that can carry higher current than the bipolar NPN transistor. The metal part serves to cool down the transistor that can become very hot if large currents are flowing...RELAX!! This is not our case!
Make sure the flat part of the MOSFET is facing the lines and that you can read the descriptions above the legs (G-gate on the left, D-drain in the middle and S-source on the right)



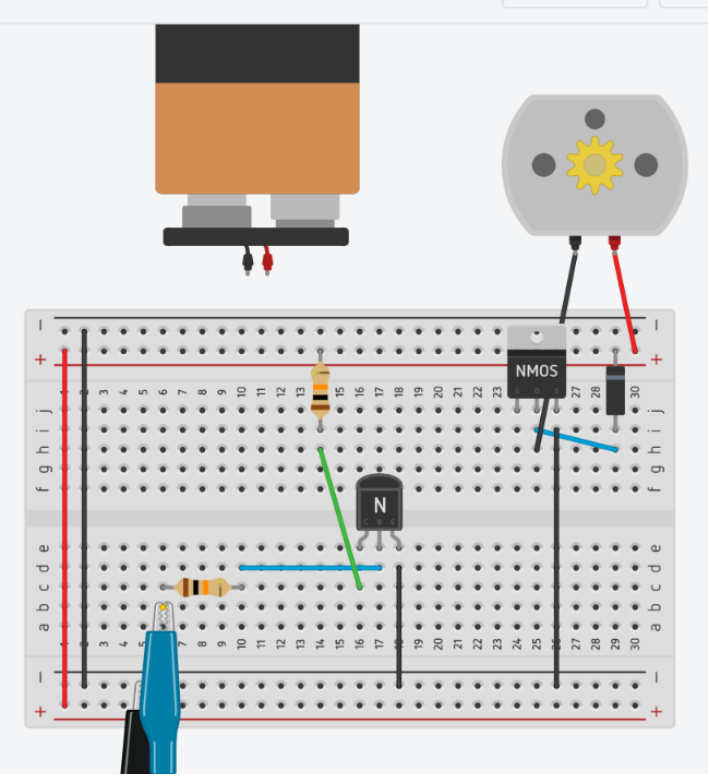
10) Connect the middle leg of the MOSFET (drain) to the bottom of the diode.



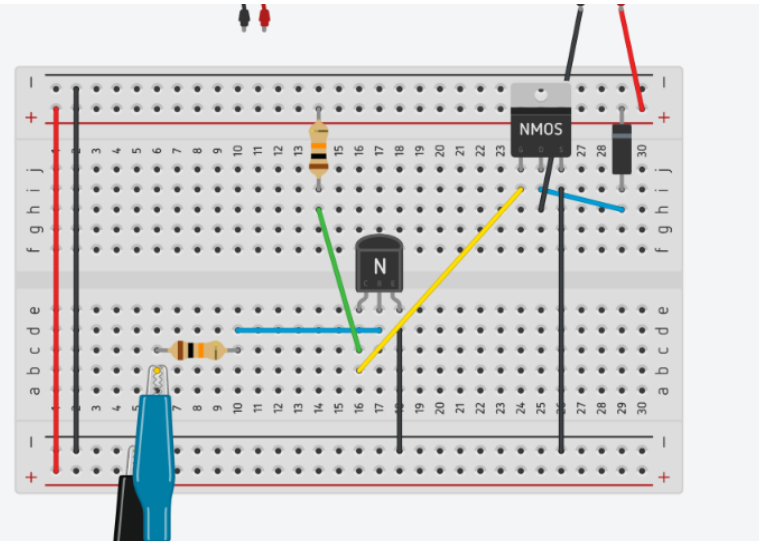
11) Take the motor and insert the cables in the circuit. The red one (positive) in the positive line (+) and the negative (black) under the middle leg of the MOSFET.



12) Connect the source (right leg) to the min line (-) at the bottom (or above)

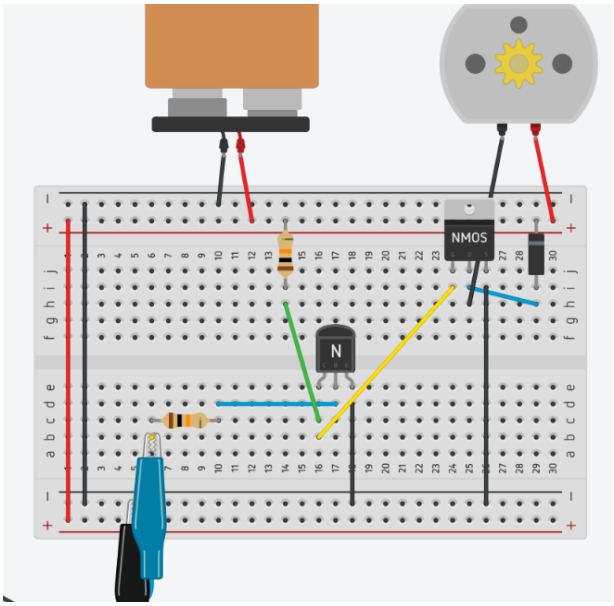


13) Connect the left leg of the MOSFET (gate) to the left leg of the NPN transistor



14) Now it is time to connect the battery.
IMPORTANT: Check if you connected everything correctly, preferably ask someone to double check your work, because now we need to connect the battery and there will be 9V in the circuit. If everything is correct, you can proceed with the next step.

15) Clip the battery clip on the 9V battery and connect the positive side of the battery (red!) to the plus. Connect the negative side of the battery (black!) to the min.

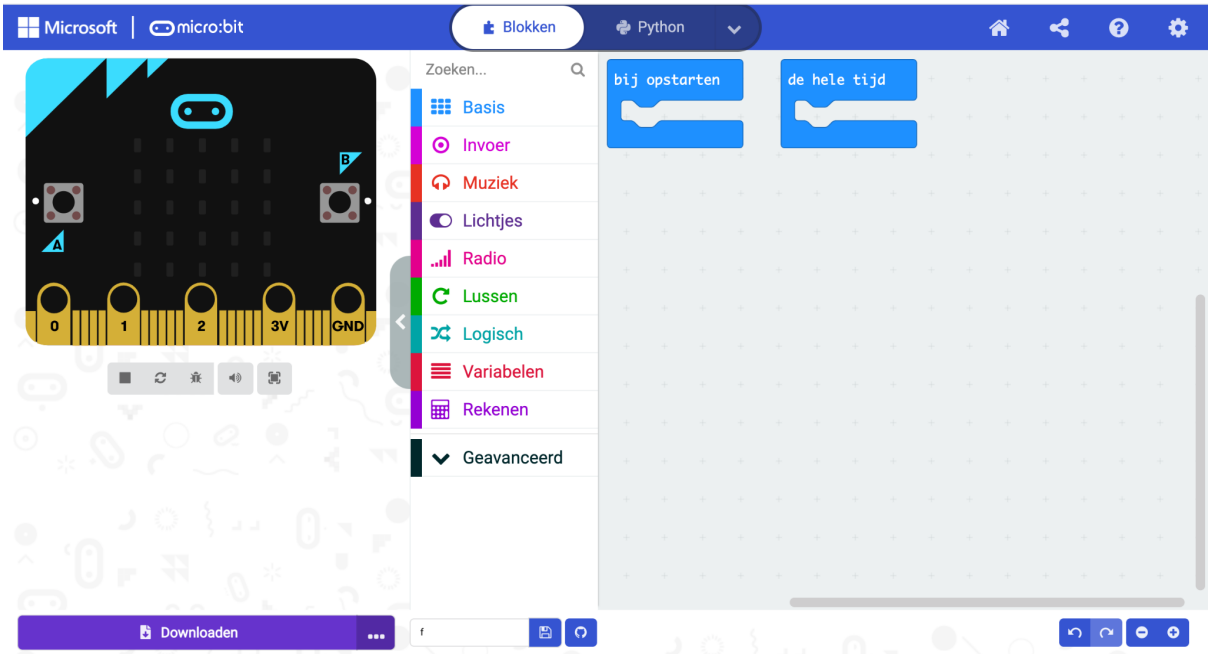


The electronic circuit of the paintbot is READY to rumble. Now let's program the Micro:Bit!

2. Programming the Micro:bit

The Micro:bit sends electrical signals to the motor telling how fast it should rotate, but first we need to tell the Micro:bit how and when to send these signals. We will do so by programming the Micro:bit using the Make Code Editor. It's an online application specially made to make coding as accessible as possible, using colorful code-blocks.
We assume here that you are already familiar with the Micro:Bit.

Go to the Make Code Editor and create a new project. Name your project.
Your screen should look like this.



You can rename the project at the bottom of your screen (PaintBot or any other funny names), next to the save disk.

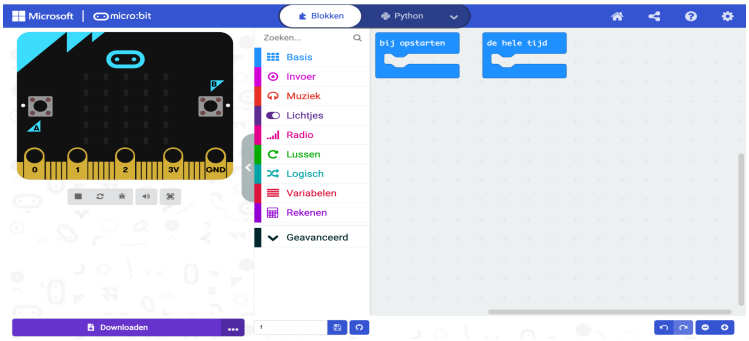
Main Code

This is the algorithm to program the Micro:Bit

- Text "PaintBot" appears on start
- After pressing button A the PaintBot starts to spin
- The Micro:Bit sends electrical signals (one number between 0 and 1023) via the analogue pin P0. This signal determines the rotational speed of the disc.
- Every cifer disappears after one second.

Let's make it

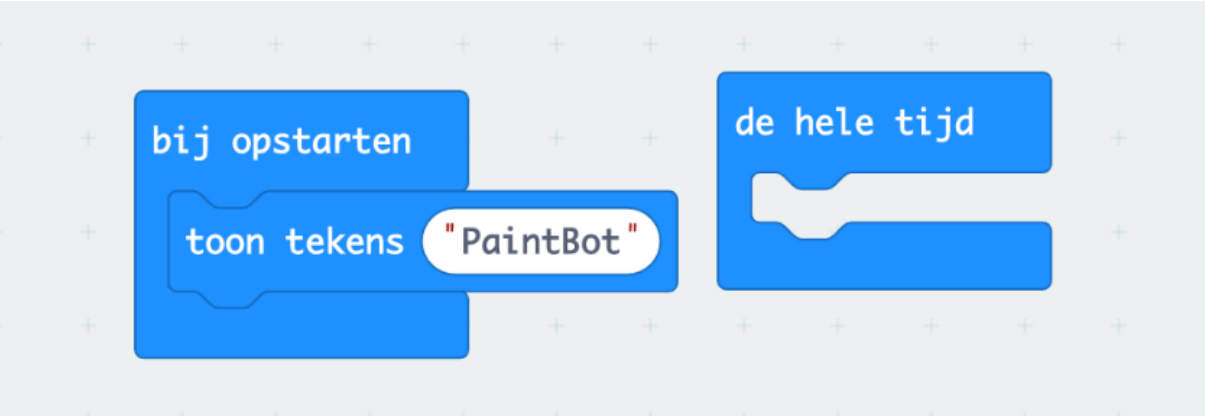
1) Open the editor



In the code editor you see two blue blocks; “on start” and “forever”. Code you place in these blocks will respectively be executed when starting the program (i.e. when starting up the Micro:bit) and during the whole operation of the program.

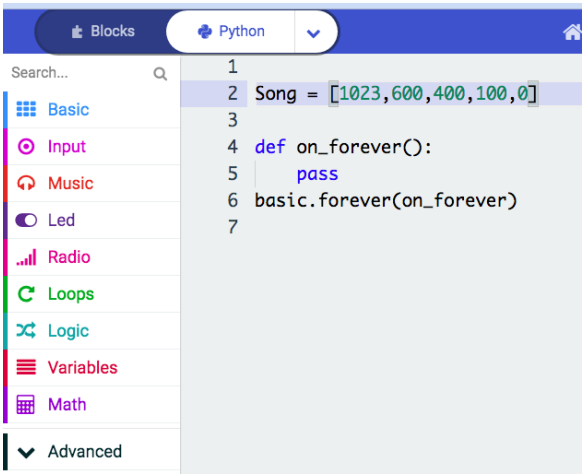
All the blocks inside the ‘on start block’ will be executed once. The blocks inside the forever block will be looped until the micro:bit is restarted or shut down.

2) Display the word “PaintBot” at start-up.

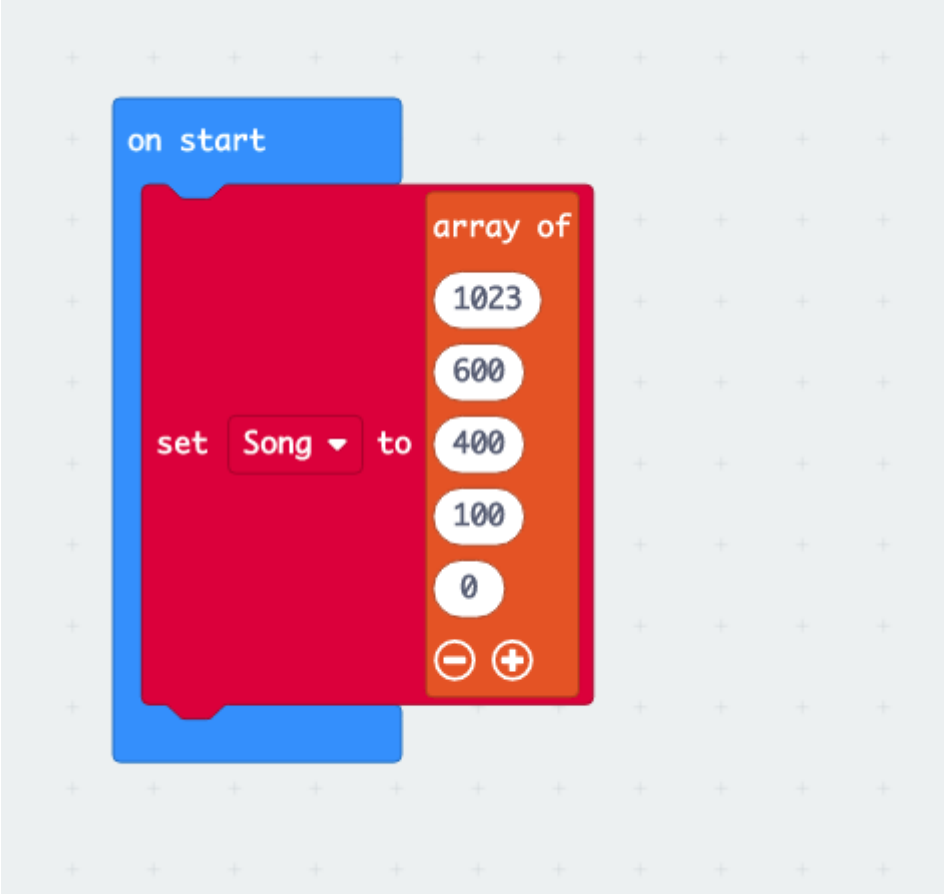


3) At the top of the page switch the editor from Block to Python and type the code below: Song = ….
This will create a new variable called Song. This variable represents the values of rotation of the motor and will depend on the beats of one song you give!

Tip: check the parenthesis and the commas!



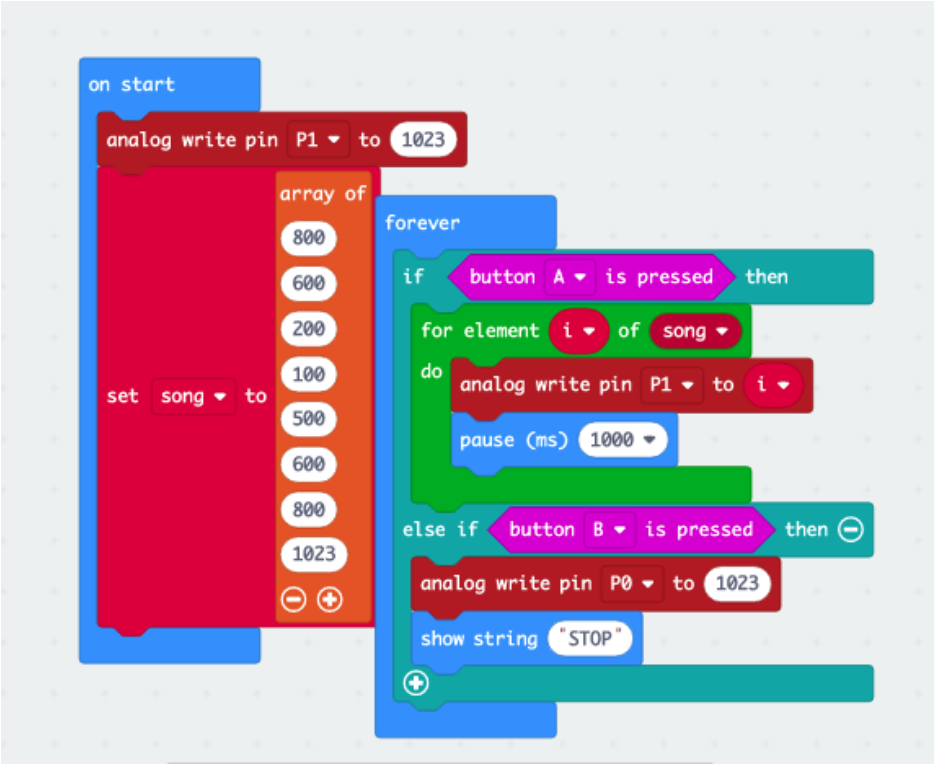
4) Go back to “blocks editor”. You should see a new variable. We will change the values of this variable using the number generator.



5) Now create the rest of the code.
6) Use a block “on start”
7) Use a logic operation: if condition 1 (do this)-- else if condition2 (do this): On this case condition1 is pressing button A condition 2 is pressing button B
8) If button A is pressed go through the values of the variable Song using a loop
9) For every element of the variable Song use a pause of 1000 ms = 1 second
10) If button B is pressed set pin0 to 1023

IMPORTANT pin0 = 1023: means the motor doesn't move
pin0 = 0 : motor maximum rotations speed

Solution



The code is now ready but.. what values for the Variable Song shall we choose, i.e what rotational speed for our motor?

The answer to this question is given by the **number generator** hereunder.

Number generation

We want to use a song as input to determine the rotation speed. Therefore we need to convert the song into numbers which the Micro:bit can understand and use as output for pin 0. We created a special number generator to do this for you.

1. First pick a nice melody or your favorite song and download it as an mp3 file (e.g. https://onlinevideoconverter.pro/nl/youtube-converter-mp3/). Save it as “song.mp3”.

2. Go to the number generator. If you see a lot of code, ignore it and focus on the instructions. You can double click on the textfield of the instructions to hide the code. On the left side, click on the folder to open a panel.

3. To use the number generator you need to be logged in with your Google account. If the students have their own, they can use it, otherwise make a general one for the class.

4. Go to the location on your computer where you saved your mp3 file. Drag and drop the file to the left side of the number generator into the panel. A pop-up will appear saying the document will only be stored temporarily and deleted afterwards. Accept to continue. Now your song.mp3 should upload and appear in the folder. If the upload is finished, you can press the play button on the left of the instructions. Be patient and a list of numbers will appear. Wait until the program finishes. Underneath the program you see the output, copy the list, including the square brackets [..]. The list starts underneath the text “`COPY THE LIST BELOW`”.

5. Go back to your Micro:Bit Make Code. In the upper middle, you see “blocks” is selected. Switch to “Python”, now you see the python code. Now at the right of “ Song = “ **paste the list you copied earlier**.

6. Switch back to “Blocks” mode. Now, a list should appear under the show “PaintBot” block, in the starting block. Now the variable “Song” contains the values relative to the BPM of the song.

Now you are all done with setting us the PaintBot now it is your time to create the artwork.

3. Create your artwork

Artwork time.
Let the students play and experiment with the system

Place the PaintBot on a desk and place some paint in the middle of the disk.
you can use different colors.
Press the button A of the Micro:Bit to activate the motor.
When the Motor spins the paint will be spread on the disk following the BPM of the music.
Repeat the process as much as you want, maybe adding different colors.

VERY IMPORTANT: ...HAVE FUN!!