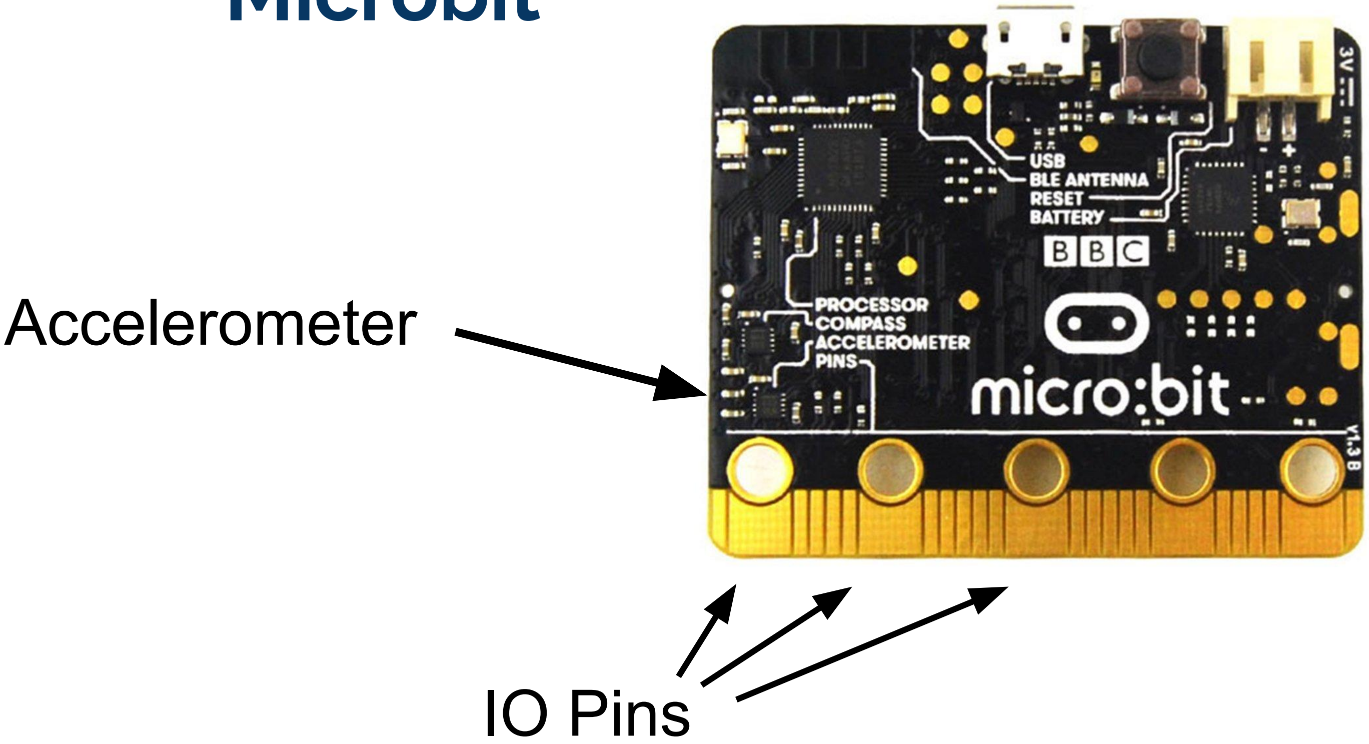
Remote Controlled Car

Get the slides

tinyurl.com/ears-remote-car

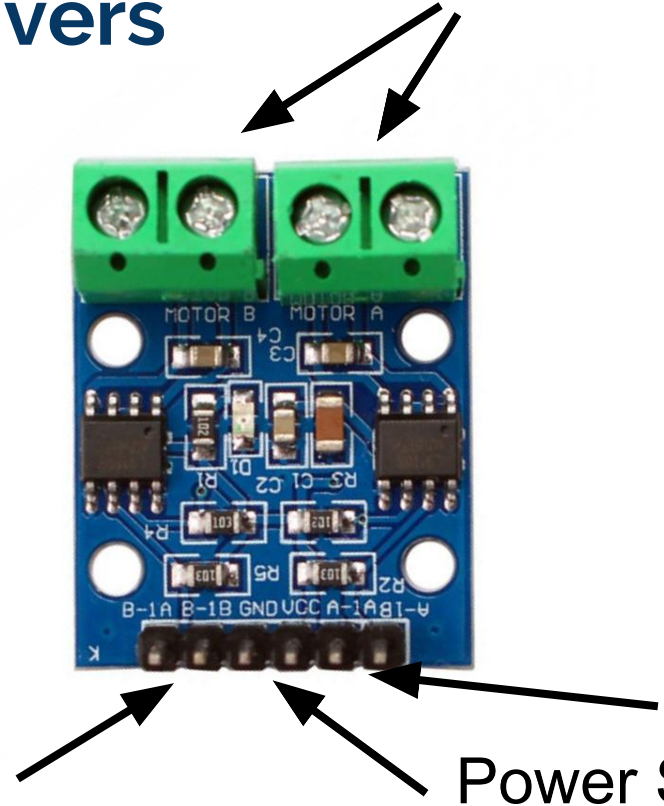
Microbit



Accelerometer

IO Pins

Motor drivers



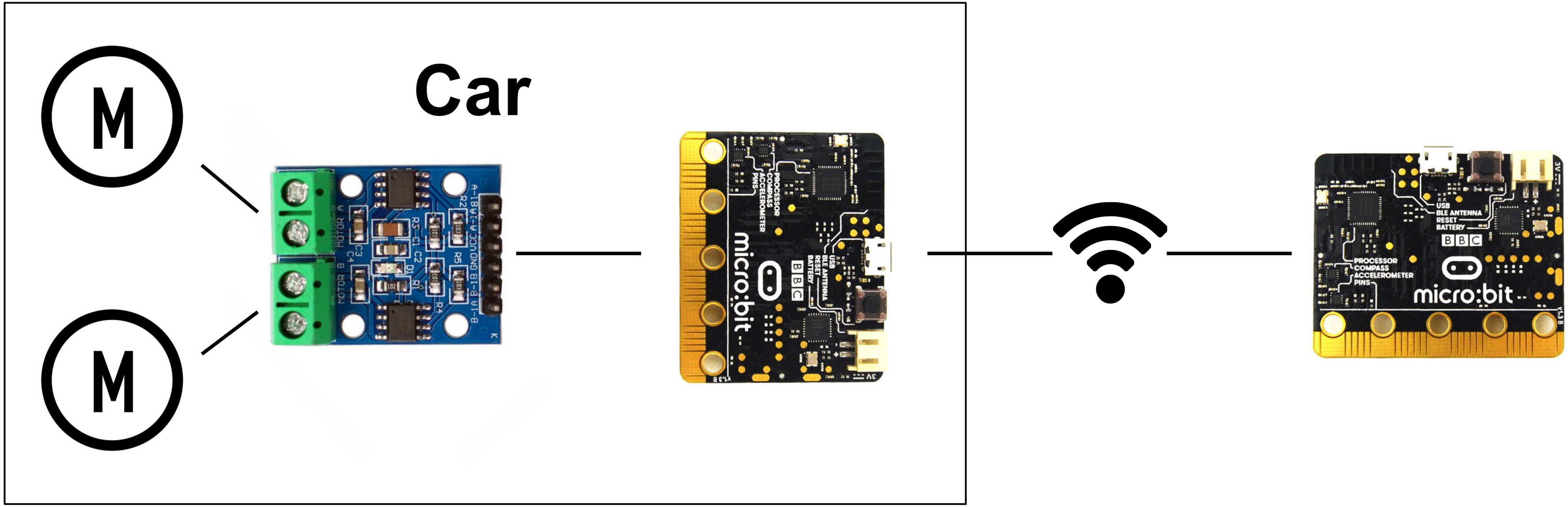
To Motors

To Microbit

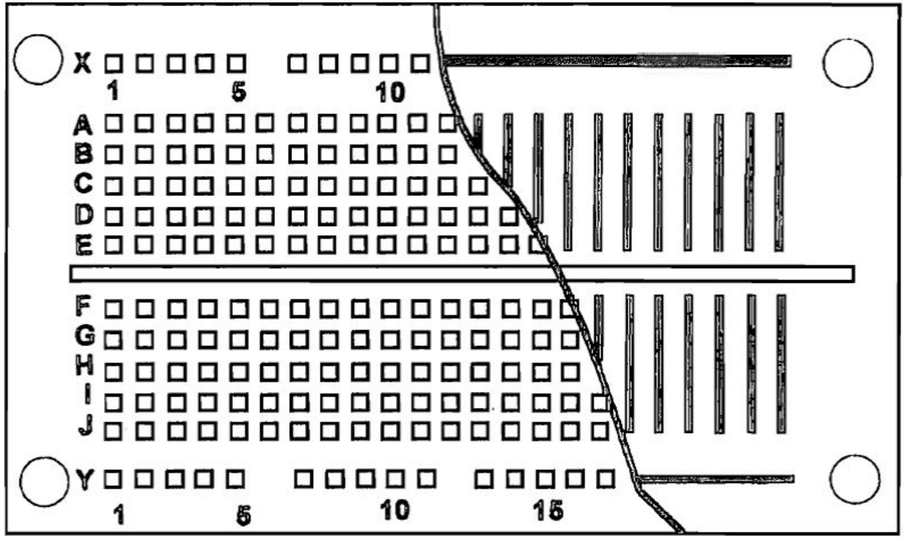
Power Supply

To Microbit

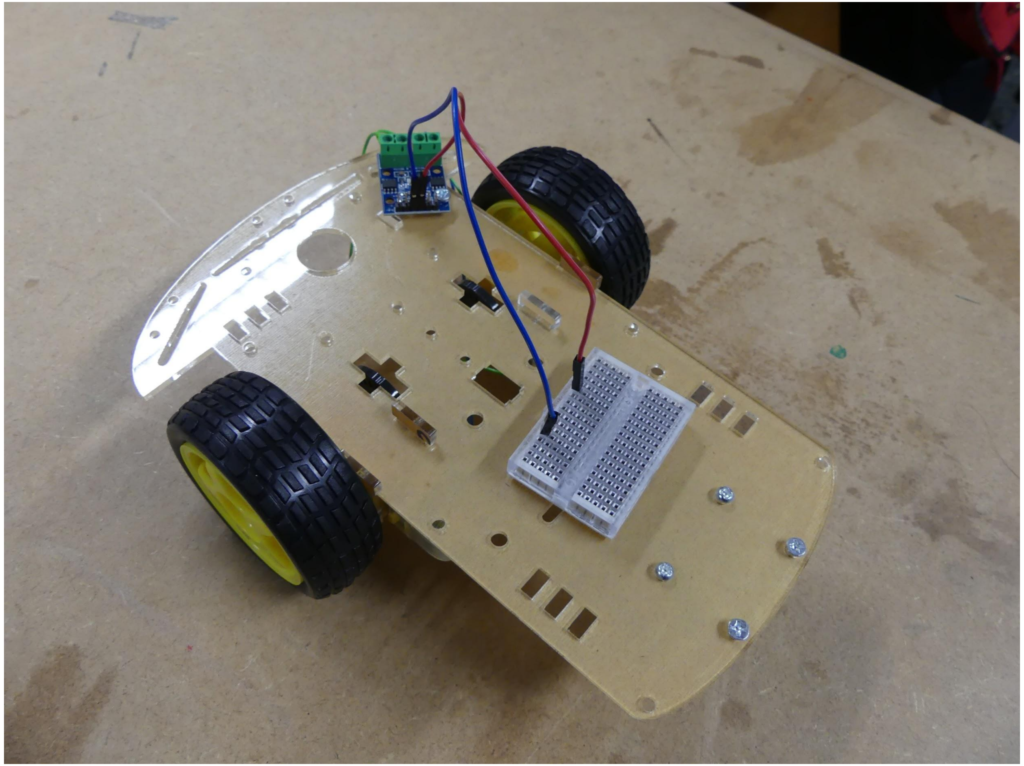
Concept

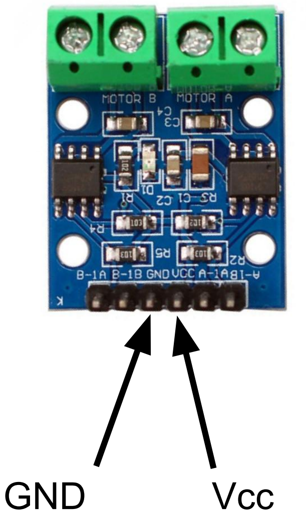


Breadboard



Connect Motor Drivers



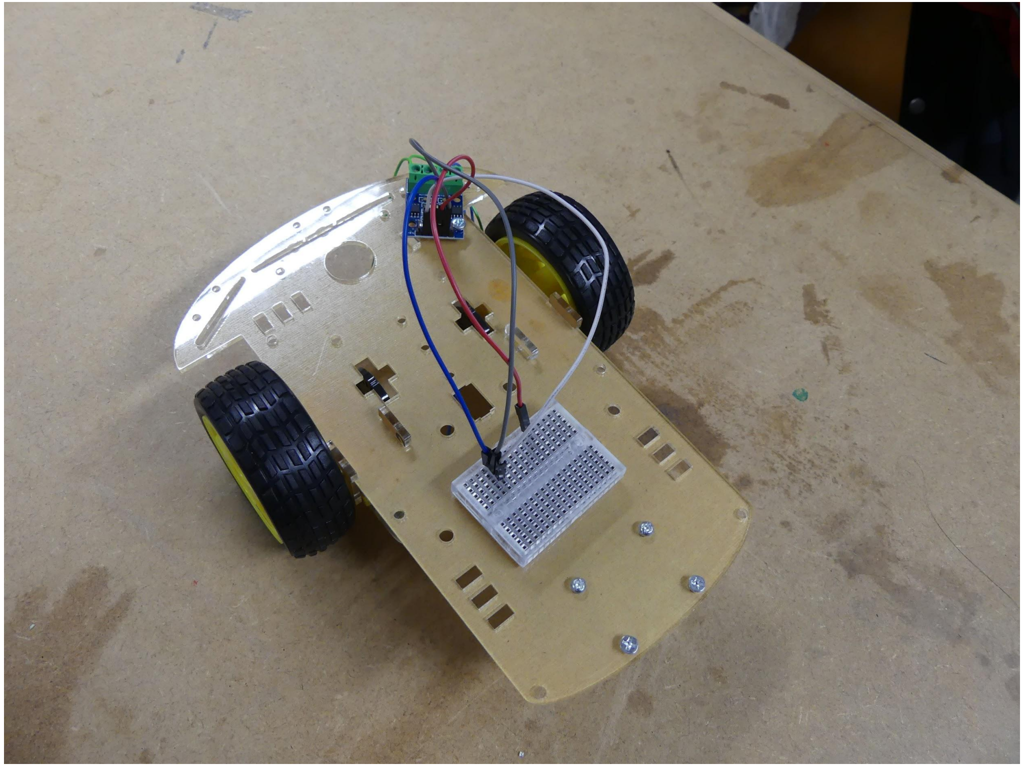


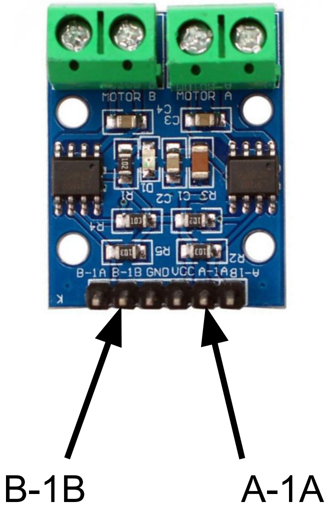
Connect:

Motor driver Vcc to breadboard

Motor driver GND to breadboard

Connect Motor Drivers



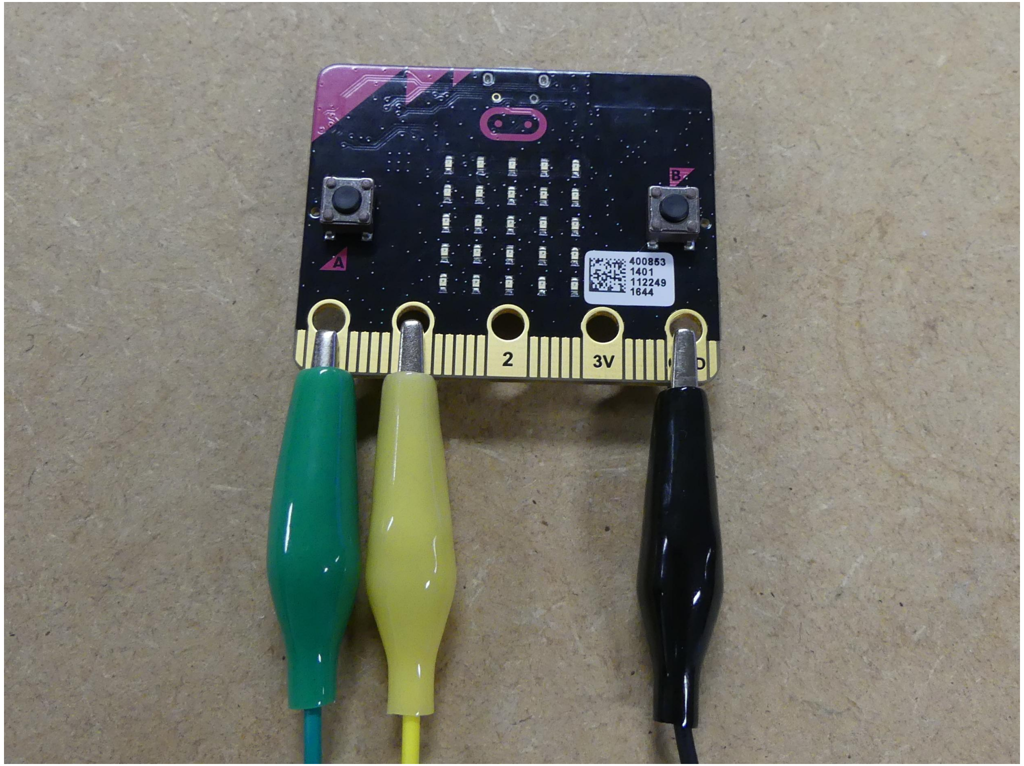


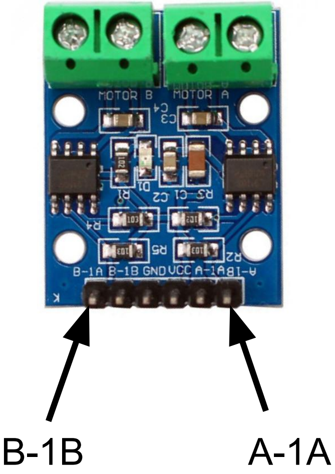
Connect:

Motor driver B-1B to breadboard (same line as GND)

Motor driver A-1A to breadboard (same line as GND)

Connect Microbit



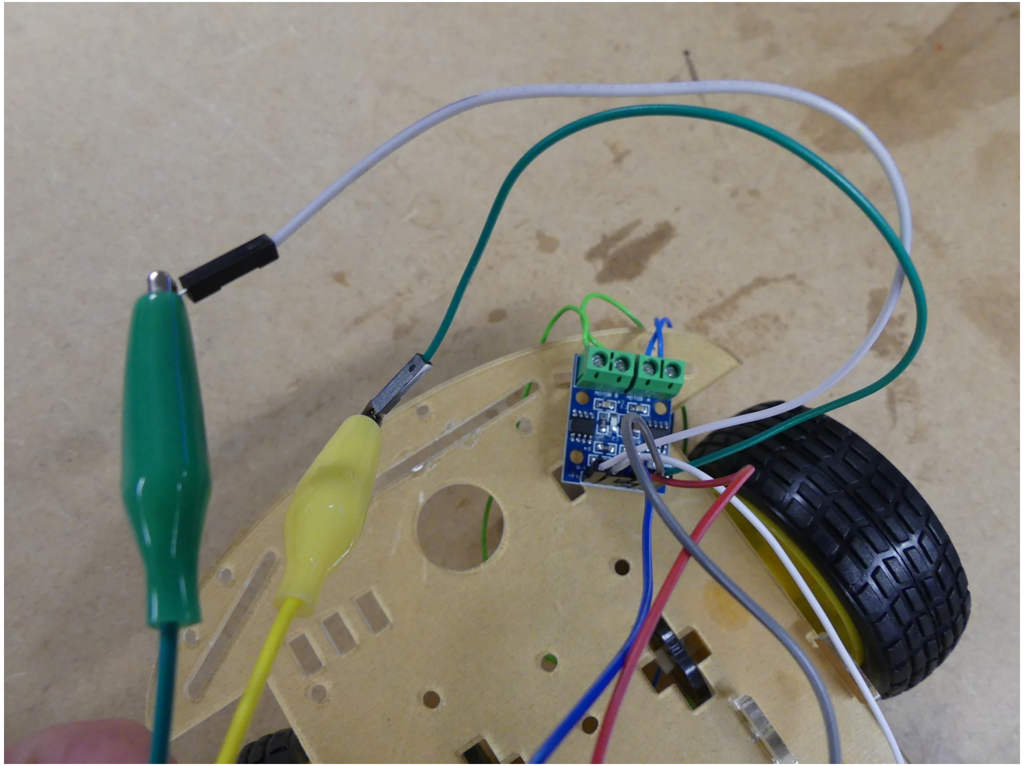


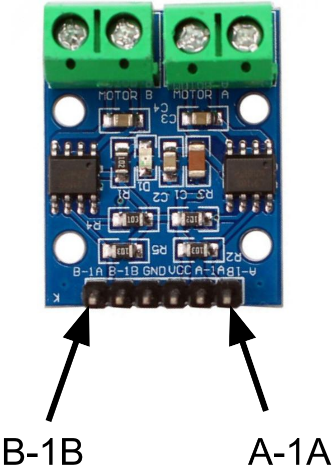
Microbit Pin 0 to Motor Driver B-1A

Microbit Pin 1 to Motor Driver A-1B

Microbit GND to GND on breadboard

Connect Microbit



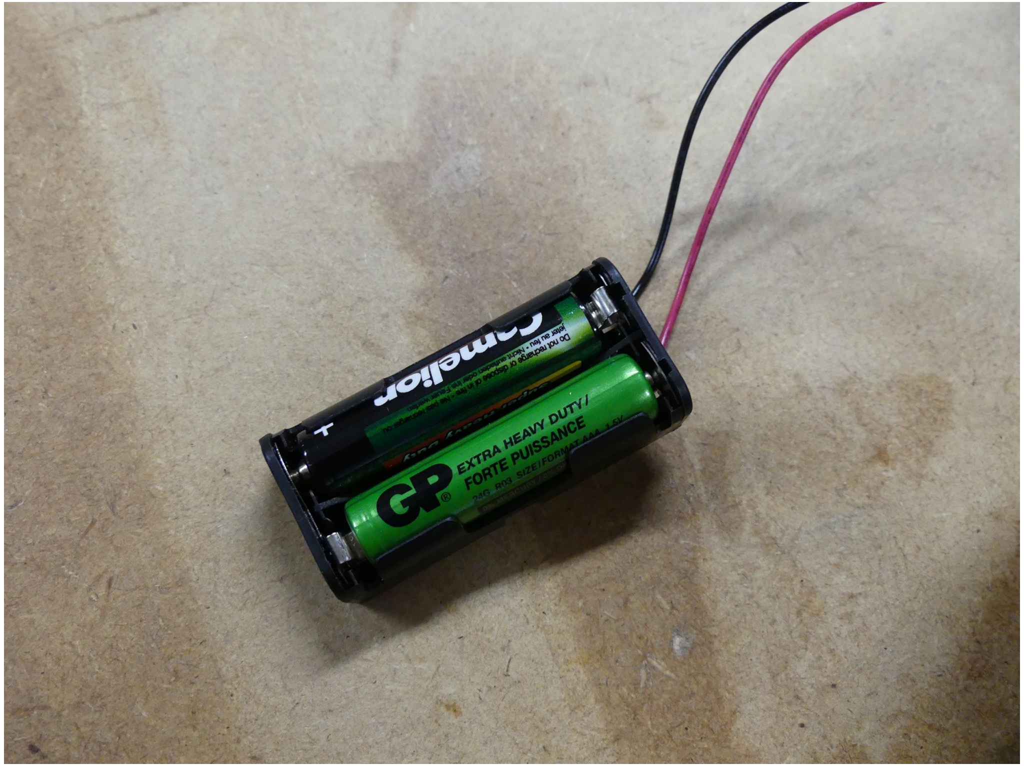


Microbit Pin 0 to Motor Driver B-1A

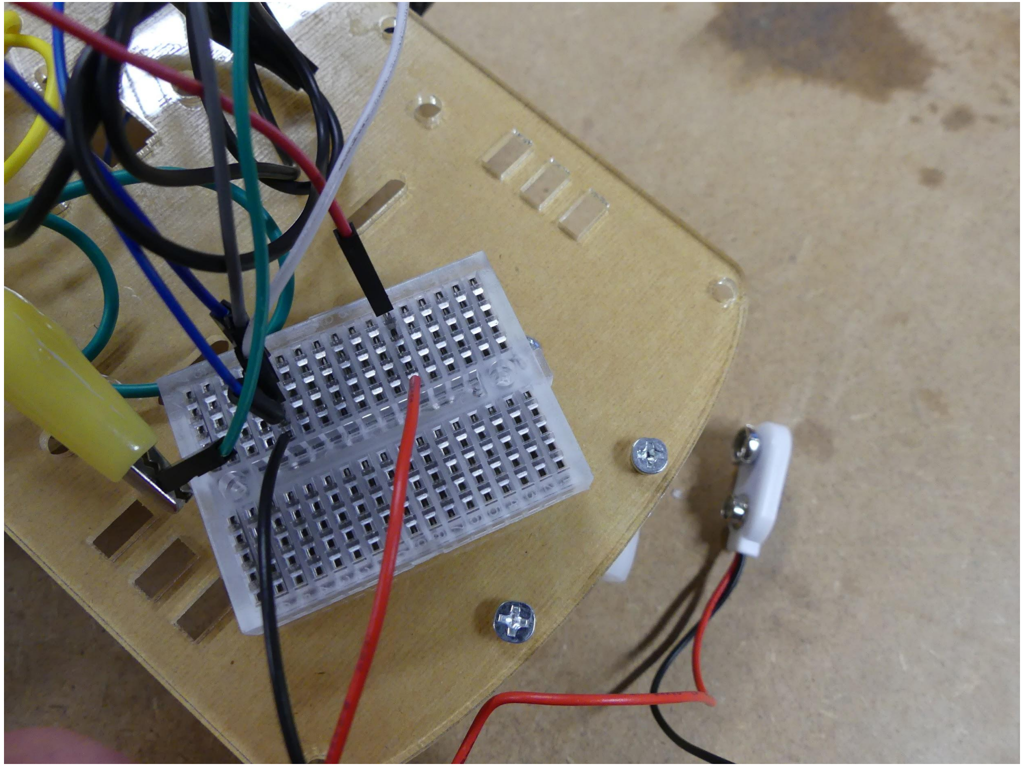
Microbit Pin 1 to Motor Driver A-1B

Microbit GND to GND on breadboard

Connect Microbit Battery



Connect Battery Clip



Battery clip black wire to GND on breadboard

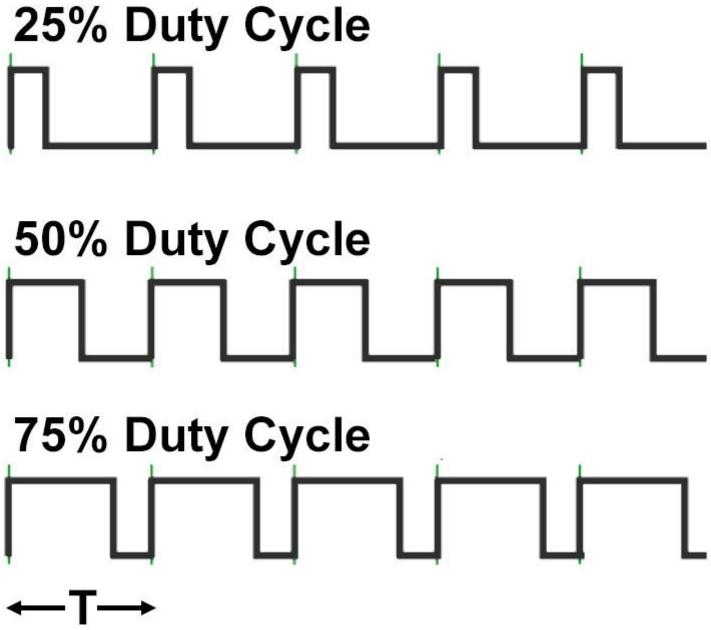
Battery clip red wire to Vcc on breadboard

Don’t connect battery for now

Test the Motors

Go to python.microbit.org

PWM - Pulse Width Modulation



PWM for the Microbit

```
pin0.write_analog(val)
```

val can be between 0 (stop) to 1023 (full speed)

```
sleep(100)
```

Microbit pauses for 100ms

Test the motors

TODO:

- Turn on the motors
- Wait for ~ ½ second
- Turn off motors

Click Download, save the microbit.hex file and copy it to the microbit

Use Reset Button on Microbit to restart program

Compare your solution

Compare your solution

tinyurl.com/ears-test-motor

```
from microbit import *

pin0.write_analog(500)
pin1.write_analog(500)
sleep(500)
pin0.write_analog(0)
pin1.write_analog(0)
```

Motors Turning Backwards

Left motor backwards:

Swap motor driver pins B-1A and B-1B

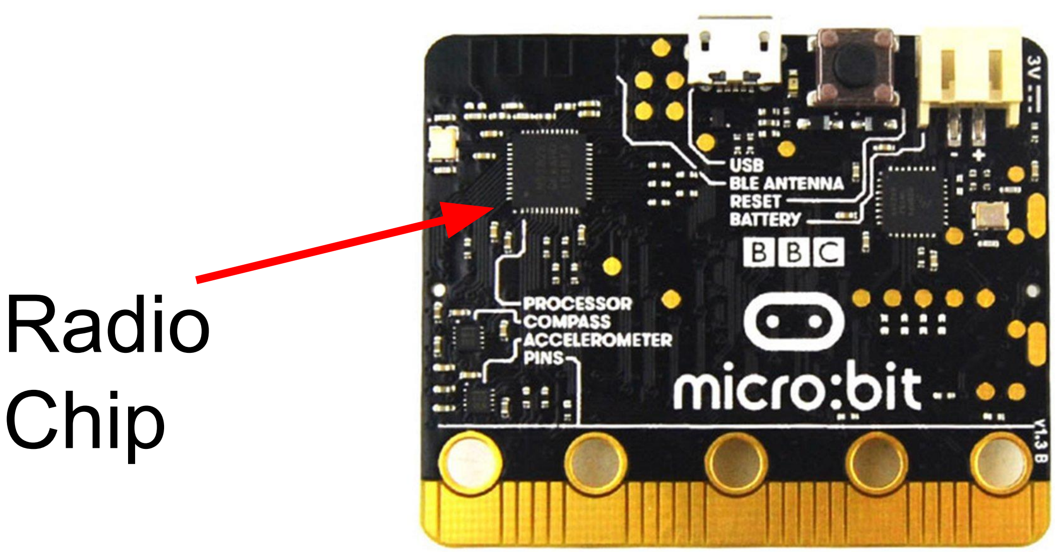
Right motor backwards:

Swap motor driver pins A-1A and A-1B

UNPLUG 9V BATTERY FIRST!

Test Radio

Radio Module



Easy way to transmit data

100 different channels

All devices on same channel receive same packages

Radio Test

What we need: (on both Microbits)

```
import radio
radio.config(channel=your_number)
radio.on()
```

Radio Test

Sending: (on Microbit without car- open another code tab)

```
radio.send("send_me")
```

Receive: (on Microbit connected to car)

```
rec = radio.receive()
If rec != None:
    # received data stored in rec
```

Test the radio connection

We need two programs (sender and receiver)

Sender transmits different strings, receiver switches between happy and sad face when strings are received

TODO - SENDER:

- Import, configure and turn on radio
- Send string “ears”
- Wait for one second
- Send String “eyes”
- Wait for one second, then repeat

Compare your solution

Compare your solution tinyurl.com/ears-radio-send

```python
from microbit import *
import radio

radio.on()
radio.config(channel=1)

while True:
    radio.send("ears")
    sleep(1000)
    radio.send("eyes")
    sleep(1000)
```

Test the radio connection

We need two programs (sender and receiver)

TODO - RECEIVER:

- Import, configure and turn on radio
- Check if something has been received
- If “ears” was received -> show Image.HAPPY
- If “eyes” was received -> show Image.SAD

Compare your solution

Compare your solution

tinyurl.com/ears-radio-receive

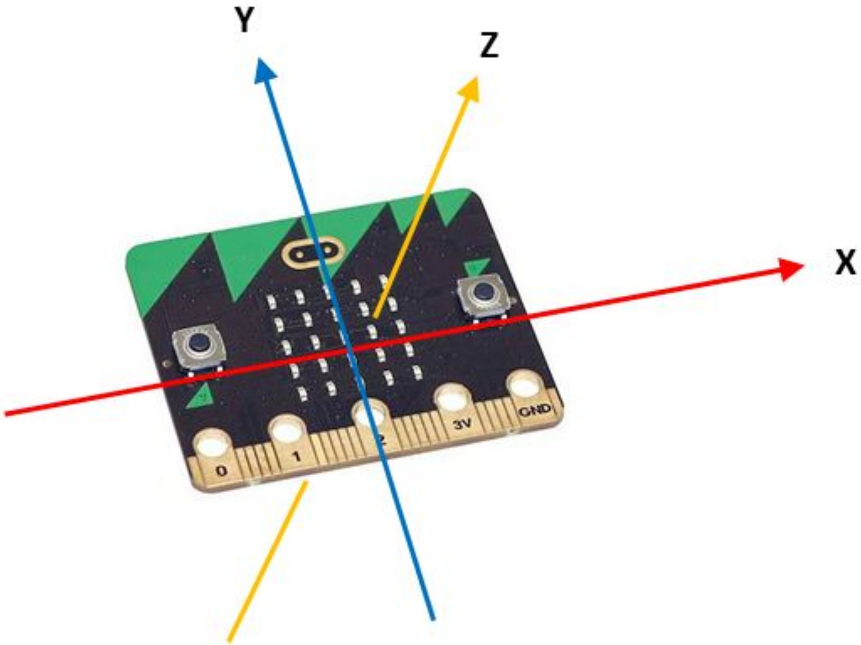
```
from microbit import *
import radio

radio.on()
radio.config(channel=1)

while True:
    rec = radio.receive()
    if rec != None:
        if rec == "ears":
            display.show(Image.HAPPY)
        elif rec == "eyes":
            display.show(Image.SAD)
```

Accelerometer

Useful to determine board tilt



Accelerometer

What we need:

```
accelerometer.get_x()
```

Returns value (-1023: tilted left, 0: horizontal, 1023: tilted right)

```
accelerometer.get_y()
```

Returns value (-1023: tilted to front, 0: horizontal, 1023: tilted to back)

Strings in Python

Concatenate strings

```
“I” + “ am “ + “ programming” == “I am programming”
```

Get first character of a string

```
str[0]
```

Get second character up to the last one

```
str[1:]
```

Strings in Python

Convert Integer to String

```
my_str = str(123)
```

Convert String to Integer

```
my_int = int("123")
```

Write Sender Code

TODO - SENDER:

- Read accelerometer x and y value
- Create a string for x value (e.g “x545”, “x-230”)
- Transmit string for x value
- Create a string for y value (e.g “y300”, “y-1000”)
- Transmit string for y value
- Repeat

Compare your solution

Compare your solution

tinyurl.com/ears-car-send

Write Receiver Code

TODO - RECEIVER:

- Check whether string has been received
- Extract letter and number from string
- If x value positive:
  - Set pin0 to x - y
  - Set pin1 to (-y)
- If x negative:
  - Set pin0 to (-y)
  - Set pin1 to (-x) - y
- CHECK THAT MOTOR VALUES ARE BETWEEN 0 AND 1023

Compare your solution

Compare your solution

tinyurl.com/ears-car-receive

NOW LET YOUR CAR RACE!