

WHEELY-JOYSTICK-MOUSE

Assist Heidi, Technikum Wien



1 Inhalt

1.	Setting up the Microcontrollers.....	2
1.1	Introduction.....	2
1.2	Puck.js.....	2
1.3	Espruino Web DIE	3
1.4	Uploading the Code.....	3
1.5	Connecting to the Laptop	4
1.6	Adjusting the sensitivity and Speed	5
1.6.1	Adjust Sensitivity	5
1.6.2	Adjust Speed	5
1.6.3	Summary.....	5
2	Installing the Brackets.....	5

1. Setting up the Microcontrollers

1.1 Introduction

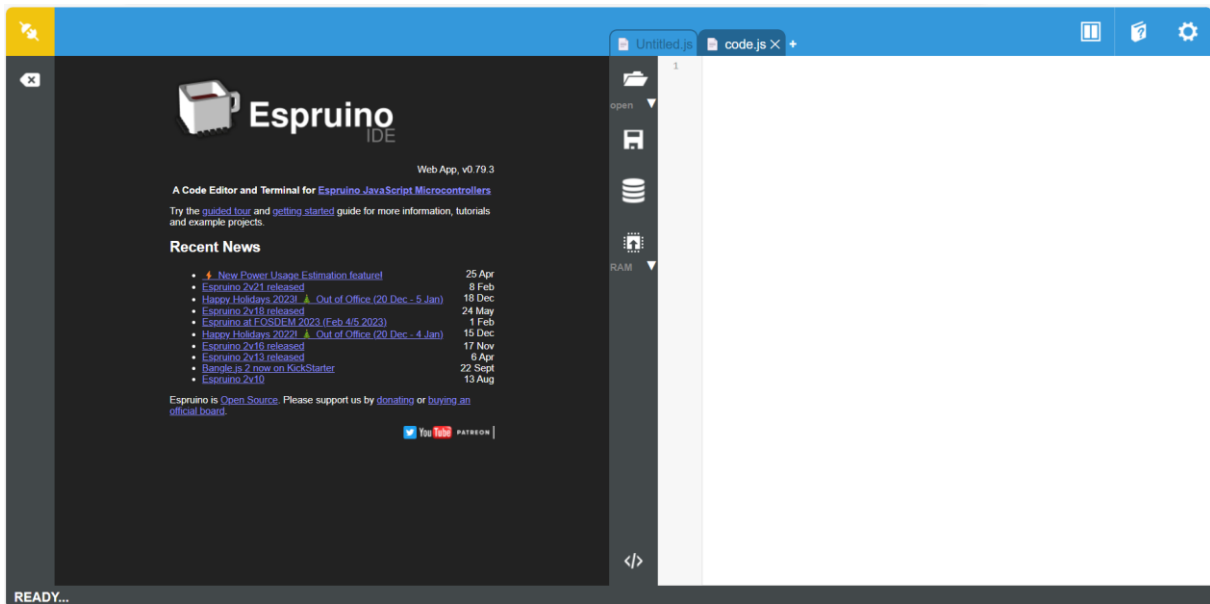
This manual provides a step-by-step guide on how to set up Puck.js as a Bluetooth Human Interface Device (HID) to control mouse movements and clicks using the Espruino Web IDE. Puck.js is a small, battery-powered Bluetooth device that can be programmed using JavaScript.

1.2 Puck.js



Puck.js is a JavaScript-powered microcontroller with built-in Bluetooth, accelerometer, magnetometer, and other sensors. It can be used for various applications, including as a Bluetooth HID device.

1.3 Espruino Web DIE



The Espruino Web IDE is an online integrated development environment for writing, uploading, and debugging JavaScript code on microcontrollers like Puck.js. It provides an easy-to-use interface for developing and testing your code

1.4 Uploading the Code

1. Prepare Your Puck.js and Computer:

- Make sure your Puck.js has a fresh battery.
- Turn on the Bluetooth on your computer.

2. Open Espruino Web IDE:

- Open your web browser and navigate to the Espruino Web IDE

3. Connect to Puck.js:

- Click on the `Connect` button (plug icon) located at the top left corner of the Espruino Web IDE.
- A pop-up window will appear with a list of available Bluetooth devices. Select your Puck.js from the list. It is usually named something like "Puck.js XXXX" where XXXX is the last few digits of its Bluetooth address.
- If prompted for permission to access Bluetooth, allow it.

5. Upload the Code:

- Click the `Send to Espruino` button (an icon with a right-facing arrow) located at the top middle of the Espruino Web IDE.
- Alternatively, you can use the keyboard shortcut `Ctrl + Enter` (Windows/Linux) or `Cmd + Enter` (Mac) to send the code.

6. Check Console Output:

- After uploading, you can see the console output in the right pane of the IDE. This is useful for debugging and checking if your code is running as expected.

7. Save Code to Flash Memory (Optional):

- If you want the code to persist even after the Puck.js is powered off and on again, you need to save it to flash memory.

- Click on the `Settings` gear icon at the top right corner.

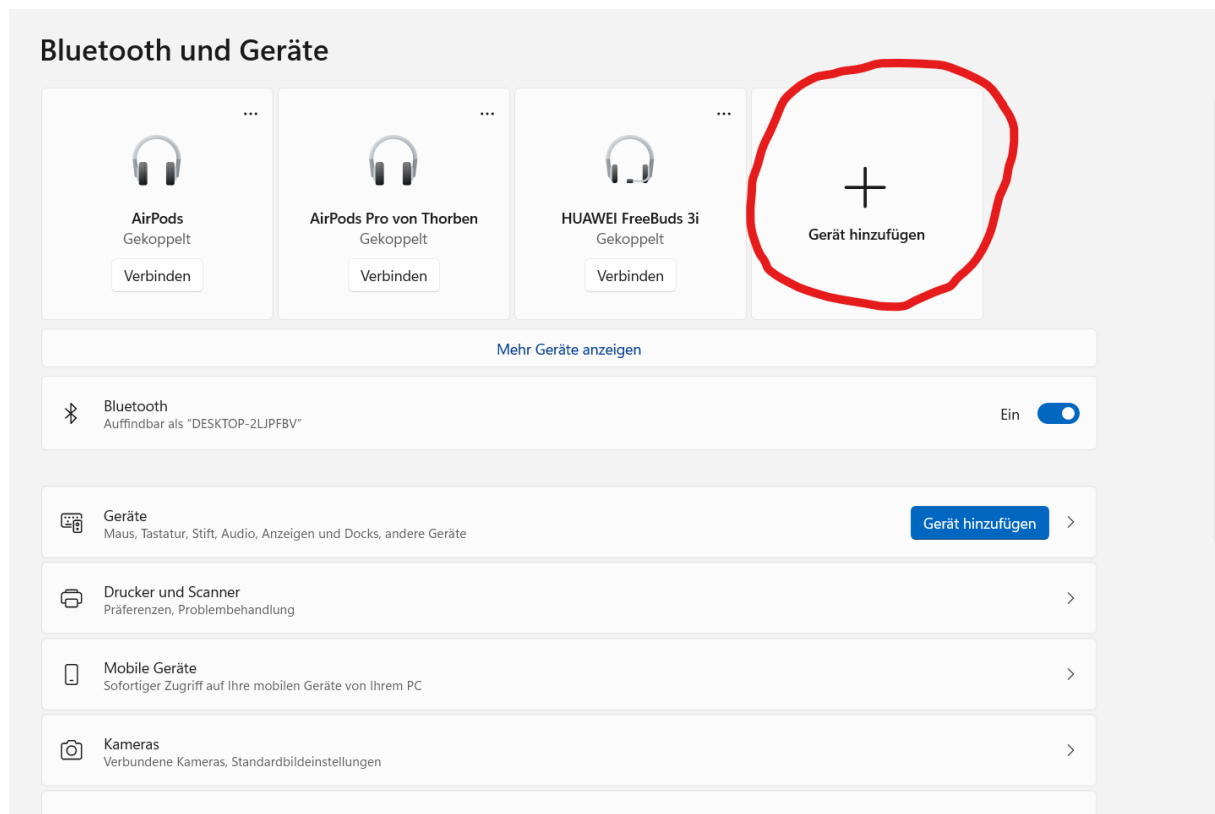
- Go to `Flasher` and then click on `Save on Send`.

- Now, whenever you send code to the Puck.js, it will automatically be saved to the flash memory.

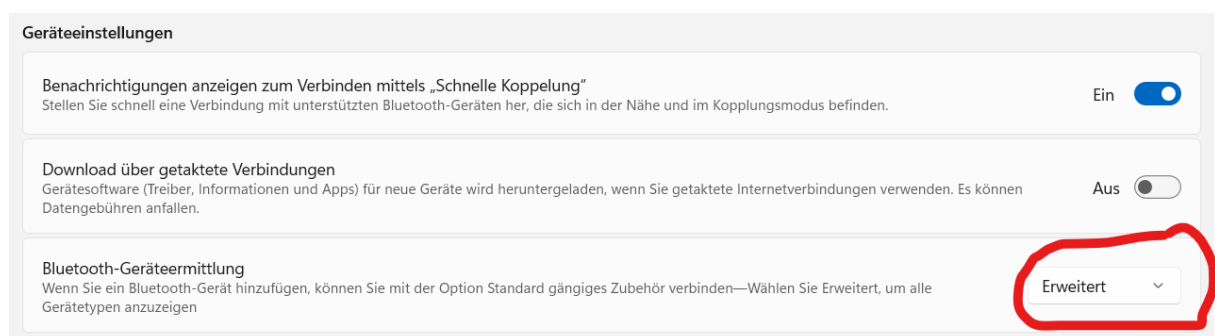
That's it! Your code should now be running on the Puck.js. You can disconnect by clicking the `Disconnect` button (plug icon with a cross) if needed.

1.5 Connecting to the Laptop

Now the Code is uploaded to the Puck.js you can simply connect it to the Laptop in the Bluetooth Settings



In Case the Microcontrollers don't show up you have to change the Bluetooth Settings to extended



1.6 Adjusting the sensitivity and Speed

```
// Function to handle accelerometer (tilt) events
function onAccel(a) {
  let x = 0, y = 0;
  const sensitivity = 500; // Adjust sensitivity as needed (lower value = higher
  sensitivity)
  const speed = 10; // Adjust speed for faster movement

  // Use accelerometer data to control mouse movement
  if (a.acc.y > sensitivity) y = speed;
  else if (a.acc.y < -sensitivity) y = -speed;
  if (a.acc.x > sensitivity) x = speed;
  else if (a.acc.x < -sensitivity) x = -speed;
```

1.6.1 Adjust Sensitivity

1. Open your code in the Espruino Web IDE.
2. Locate the function onAccel(a).
3. Find the line with const sensitivity = 500;.
4. Change the value of the sensitivity variable:
 - o Lower value: Increases sensitivity (the mouse reacts more to movements).
 - o Higher value: Decreases sensitivity (the mouse reacts less to movements).

1.6.2 Adjust Speed

1. In the same function onAccel(a), locate the line with const speed = 10;.
2. Change the value of the speed variable:
 - o **Higher value:** Increases the movement speed of the mouse.
 - o **Lower value:** Decreases the movement speed of the mouse.

1.6.3 Summary

- **Change sensitivity:** Adjust the value of the sensitivity variable. A lower value means higher sensitivity.
- **Change speed:** Adjust the value of the speed variable. A higher value means higher speed

2 Installing the Brackets

Now first, put the Microcontrollers in their Place in the Bracket and glue them there. Afterwards you should be able to put the Brackets at their Position by simply screwing it there for the Click part and for the Joystick you should be able to simply stick it on the rod of the previous joystick.

