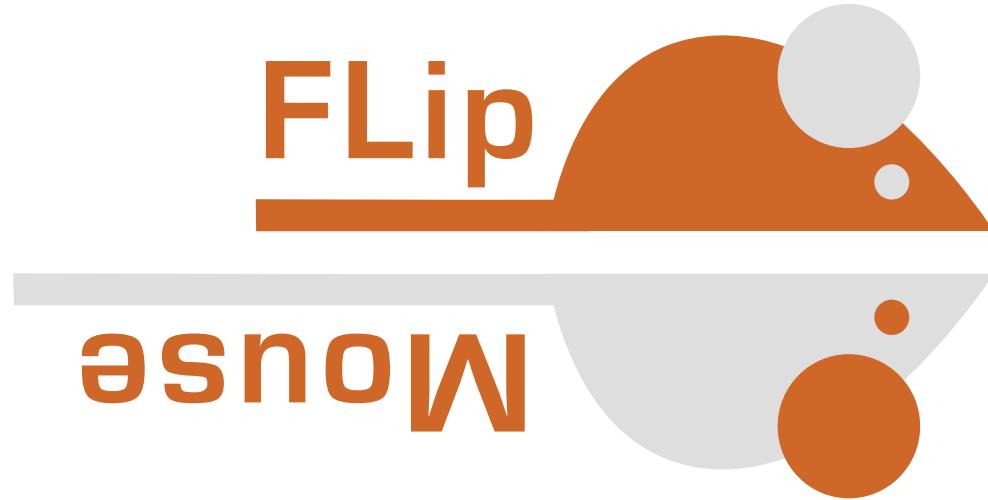


# FLipMouse V2 - Finger & Lip Mouse

User Manual v2.12, AsTeRICS Foundation



## Table of Contents

### FLipMouse V2 - Finger & Lip Mouse

#### Preface

#### Hardware overview

3D-Printed Housing

Bluetooth – Add-On Module

#### Instructions for proper mounting and use

Using the FLipMouse with Smart Phones or Tablets that provide USB-OTG

Using the FLipMouse via Bluetooth

#### The FLipMouse Configuration Manager

Connecting the FLipMouse Device

Changing Settings and Features

Operational modes – “memory slots”

Stick Configuration (Tab STICK-CONFIG)

Define stick usage (“Use stick for”)

Calibration of the stick’s middle position

Stick Orientation (Change with click on “ROTATE RIGHT”)

Sensitivity

Deadzone

Split axis control for sensitivity and deadzone

Maximum Speed

Acceleration

Drift compensation range + Drift compensation gain

Using the Stick for Joystick Movement

Sip/Puff Actions and Levels (Tab SIP AND PUFF)

StrongSip and StrongPuff

Assignment of different actions (Tab ACTIONS)

Action category “Mouse”

Action category: Joystick

Action category: Keyboard

Action category: Device  
Action category: Infrared  
Action category: Macro  
Using Configuration Slots (Tab SLOTS)  
General Settings (Tab GENERAL)  
USB/Bluetooth mode  
FLipMouse Firmware ("UPDATE FIRMWARE")  
Firmware Bluetooth-Add-on ("UPDATE BLUETOOTH FIRMWARE")  
View Force Levels (Tab VISUALIZATION)  
Infrared Code Recording and Replay

### **Using the Bluetooth module**

Installation of the Bluetooth module  
Connecting with a Bluetooth host device (pairing)

### **Mod your Mouthpiece?**

### **Instructions for cleaning and maintenance**

Hygienic considerations and risks

### **Updating the Firmware via Arduino IDE**

### **Building the Firmware**

### **Further instructions and troubleshooting**

### **Appendix: Macro commands and Key codes**

Supported Macro commands:

Supported Key Identifiers:

### **Contact Information**

### **Disclaimer**

### **Acknowledgement**

# Preface

---

The FLipMouse is an open source Assistive Technology module which is developed at the [University of Applied Sciences Technikum Wien](#) and delivered as a DIY-construction kit by the [AsTeRICS Foundation](#).

The FLipMouse allows people who cannot use standard computer input devices to control a computer mouse cursor or a joystick as well as typing desired keyboard keys or using infrared remote controls.

The FLipMouse device can be actuated with minimal finger- or lip-movements and/or sip & puff activities via a dedicated mouthpiece.

(**Important:** This documentation refers to Version V2 of the FLipMouse, the new Version V3 is described here: [FLipmouse V3](#).)

The FLipMouse can be configured via a web-based configuration application which is called *configuration manager* from now on. It can be found at the website [flipmouse.asterics.eu](#). This user manual includes a description of the configuration manager and explains how to use the different functions and features of the FLipMouse. The FLipMouse can be used on any Windows-, Mac- or Linux-computer without installation of special software, because it behaves like a standard mouse / joystick / keyboard device. By using the optional Bluetooth Add-On module, smart phones or tablet computers can be controlled - which will also be explained in this manual.

Furthermore, this manual covers important guidelines for using the FLipMouse, in particular cleaning procedures and hygienic precautions.

All design files for the electronics and the 3d-printed enclosure - as well as the software source code - are available as open source and are distributed via the AsTeRICS Foundation website and via GitHub. The construction kit for the FLipMouse can be purchased from [Hackerspaceshop.com](#). Detailed instructions are provided in the [FlipMouse construction manual](#). Have fun building and using your FLipMouse!

The AsTeRICS Foundation

[www.asterics-foundation.org](#)

# Hardware overview

---

The FLipMouse is available with a 3d-printed enclosure. Design files for the 3d-printed case are provided in the [Github repository](#). The primary method for interacting with the FLipMouse is the transparent stick ("Hygienic Stick" or "mouthpiece") which is attached to the front of the FLipMouse. The *Hygienic Stick* is made by the company *QuadJoy*. Replacement sticks can be ordered via the homepage: [www.quadjoy.com](http://www.quadjoy.com). We want to thank Bill and Debra Street of *QuadJoy* for letting us use their mouthpiece for our FLipMouse construction kit! (Please note the cleaning instructions in the dedicated section of this manual.)



Figure 1: Using the FLipMouse for computer control via lip interaction

Additionally, two external momentary switches can be attached to the jack plug connectors on the left side of the FlipMouse enclosure and one button is already integrated on the device ("B1", see Figure 3).

## Users can interact with the FLipMouse in several ways:

1. by touching the mouthpiece with the lips or fingers and applying small forces in vertical or horizontal direction
2. by increasing or reducing pressure in the mouthpiece (puffing or sipping)
3. by actuating (up to) 3 momentary switches / pushbuttons
4. if desired, a longer plastic tube can be attached to the pressure sensor so that the joystick can be used with a finger and a separate mouthpiece allows sip/puff control. Alternatively, pressure can be influenced by attaching a rubber ball to the plastic tube (pillow switch / soft switch).

## The hardware features of the FLipMouse: (see pictures at the bottom of this list)

- (a) "Zero-way"-joystick / mouthpiece: very low forces are sufficient to create up/down/left/right movement
  - o The zero-way-joystick can be used with fingers or toes etc.

- o It can also be used as a mouthpiece (actuated by lips / mouth movements)
- o If desired, sip / puff activities into the mouthpiece can trigger additional functions
- (b) One built-in switch ("Button 1 / B1") on the device, e.g. to change the active configuration (functions)
- (c) Two 3.5 mm jack plug sockets for attaching external switches to trigger additional functions ("Button 2 / B2" and "Button 3 / B3")
- (d) 3 Indicator Light Emitting Diodes (LEDs) for showing the active configuration, calibration procedure etc.
- (e) Universal infrared remote control receiver and transmitter
- (f) "Hot Shoe" Adapter for mounting to a Manfrotto Magic Arm or similar mounting system
- (g) Firmware upgrade via FLipMouse web application on [flipmouse.asterics.eu](http://flipmouse.asterics.eu)
- (h) Optional add-on board for Bluetooth (e.g. to control smartphones or iOS devices)
- (i) Reset button

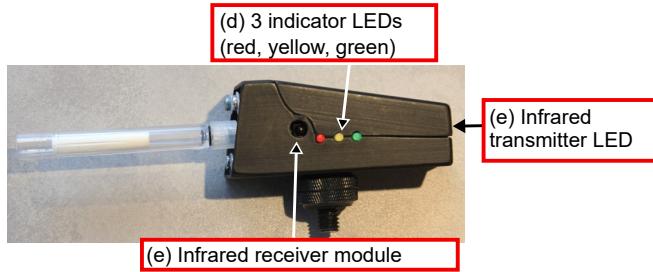


Figure 2: FLipMouse right side

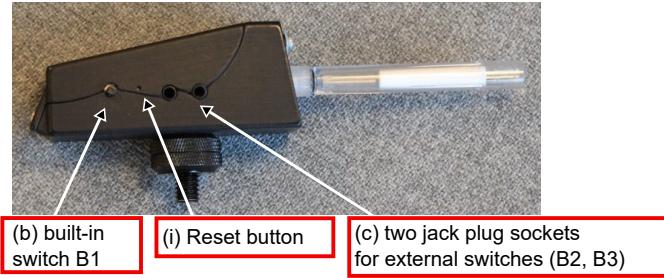


Figure 3: FLipMouse left side

On the right side of the FLipMouse enclosure, 3 LEDs indicate the current mode of operation (which can be changed to activate different speed settings or functional options). Furthermore, the Infrared (IR-) receiver module is accessible here. This modules can be used for recording arbitrary infrared remote control commands (for example to change volume or channel settings of a TV-set). The IR-signals can then be replayed from the IR-emitting diode at the rear side of the FLipMouse.

The left side of the FLipMouse gives access to a button (B1) which offers a configurable function (for example changing the operational mode). Next to this button is a hole which gives access to the reset-button of the FlipMouse (usually not needed).

Two 3,5mm Jack Plug sockets marked (B2) and (B3) allow the connection of external momentary switches. Standard switches with 3,5 mm mono jack plugs can be used.

## 3D-Printed Housing

---



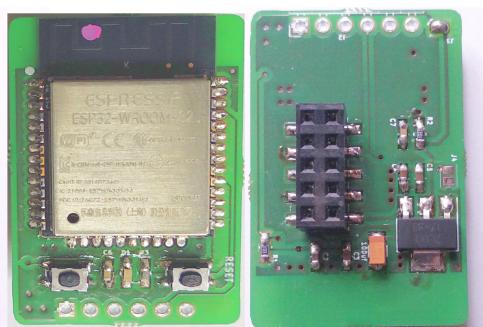
Figure 4: FLipMouse 3D-printed enclosure

Different alternatives for the FLipMouse housing are available, e.g. an acrylic version. The recommended version is the 3d printed enclosure, which is also documented in the construction manual. The 3D-design files can be downloaded from the [Github repository](#). Please note that the electronic components and PCB are the same as for the former acrylic housing variant – and the devices offer exactly the same functionality.

## Bluetooth – Add-On Module

---

The optional Bluetooth Add-On module allows connection of the FLipMouse as a Bluetooth mouse/keyboard to various mobile devices (smartphones, tablets, IOs devices). If this module is not contained in the DIY-Kit, it can be ordered individually from the AsTeRICS Foundation or built from the hardware design files available on Github (parts must be ordered individually). If you are interested in this feature, have a look at the [FLipMouse Wiki](#) where construction of the Add-On module is explained.



*Figure 5: Bluetooth Add-on Module*

# Instructions for proper mounting and use

## 1. Attach the "hygienic stick"

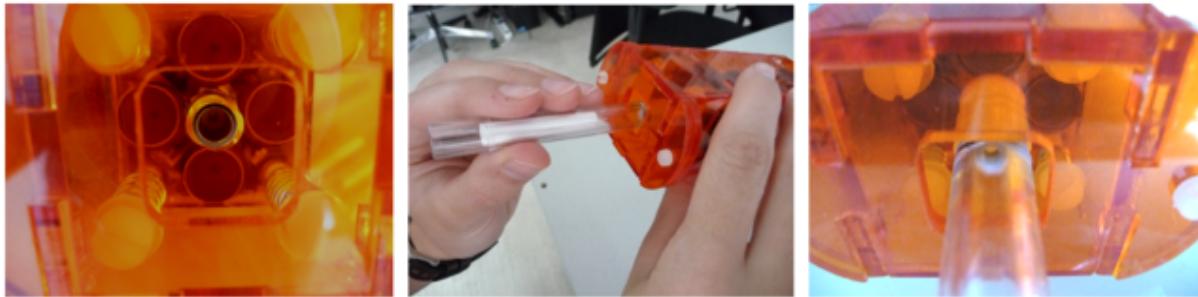


Figure 6: Attaching the hygienic stick

Use a clean "hygienic stick" and attach it to the front connector, turn clockwise (not too hard):

## 2. Mount the FLipMouse device appropriately for the user



Figure 7: Mounting the FlipMouse

The picture shows a "[Manfrotto Gelenkarm](#)" + "[SuperClamp](#)" mount combination. You could also use the [Manfrotto Magic Arm](#) or any other kind of mounting solution which fits to the HotShoe-Adapter of the FLipMouse. Be careful when attaching the mount to the 3/8" screw of the HotShoe-Adapter: hard forces might break the enclosure of the device.



Figure 8: Using the FLipMouse with Mouth/Lips

Ensure that the head of the user rests comfortably. The mouthpiece should gently touch the lips. It is not useful to insert the mouthpiece into the mouth! This could result in uncomfortable feelings and make it harder to control the lateral movement.

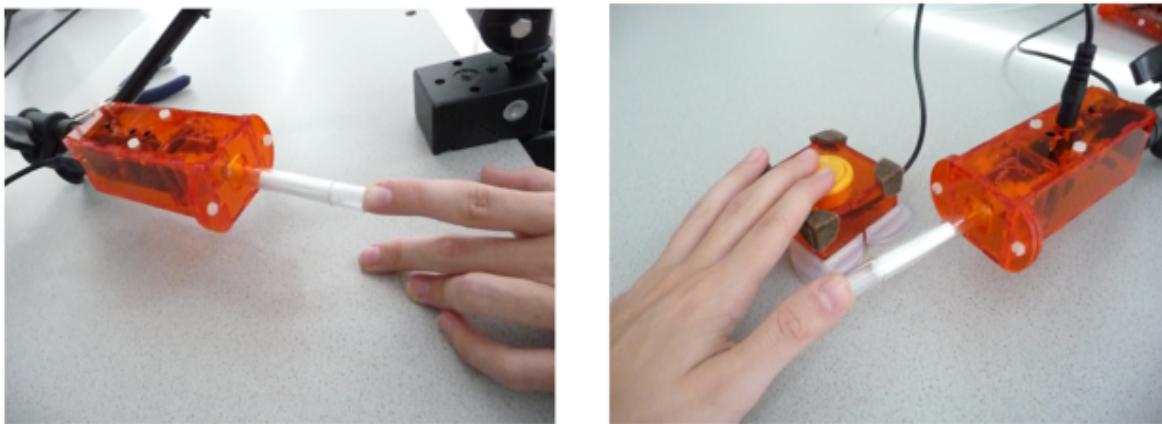


Figure 9: Using the FLIPMouse with fingers

If the user desires to actuate the FLIPMouse with a finger or the thumb, mount the system appropriately so that the hand is in a resting position and the finger can touch the joystick without stress. Attach one or two additional external switches if desired, and mount the switches at suitable locations (for example at the legs/toes/shoulder etc.).



Figure 10: Using the FLIPMouse as a ball-switch interface

If a soft ball interaction method is desired, a plastic tube can be attached to the front connector of the FLIPMouse using a *Luer Lock* adapter. This plastic tube can be connected to a soft ball (the picture above shows a cheap “dog toy” from a pet store). This solution could be used as a soft head-switch / pillow-switch or shoulder-switch and can be combined with other external switches.

### 3. Attaching the FLIPMouse to a Computer, Tablet or Smart Phone

When connecting the FLIPMouse via the provided USB-micro cable, do not touch the mouthpiece as long as the LEDs are blinking (the initial blinking indicates the zero point calibration phase). Wait until the device is recognized by the computer, then move the joystick / mouthpiece to check correct operation.



Figure 11: Connecting the FLIPMouse with the computer

**Please note** that every time you apply power to the FLipMouse device (respectively when you plug it in) the **zero-calibration is performed which is indicated by blinking all 3 LEDs**. It is important that you do not touch the joystick / mouthpiece until the LEDs stopped blinking.

## Using the FLipMouse with Smart Phones or Tablets that provide USB-OTG

The FLipMouse should work with any operating system which supports USB HID (Mouse/Keyboard/Joystick) devices, such as Windows, Linux or MacOS. Several Android devices provide an USB port with USB-OTG (“on-the-go”) functionality. If your device supports “OTG” you can attach the FLipMouse using an USB-OTG-adapter (see picture below) and it should work as a normal mouse (you will get a mouse cursor) or keyboard. You can test if your Android phone or tablet features the USB-OTG function with the “OTG checker” app.



Figure 12: Using the FLipMouse with the Smart Phone

## Using the FLipMouse via Bluetooth

The Bluetooth – Add-On-module for the FLipMouse allows wireless mouse/keyboard control of computers, tablets and smart phones. Furthermore, iPhones or iPads can be used via VoiceOver & assistive switch support. For more information see chapter [Using the Bluetooth Module](#).

# The FLipMouse Configuration Manager

The [FLipMouse Configuration Manager](#) offers a Graphical User Interface (GUI) to change and store settings of the FLipMouse device so that all features and functions can be adapted to personal preferences and needs. **Currently, the Google Chrome browser (or a Chromium-based browser) must be used.** The Configuration Manager is accessible by entering the following website: [flipmouse.asterics.eu](http://flipmouse.asterics.eu). The FLipMouse Configuration Manager sends and receives information from/to the FLipMouse. This information transfer occurs through a communication port (COM port).

The following figure shows the welcome page of the FLipmouse Configuration Manager:

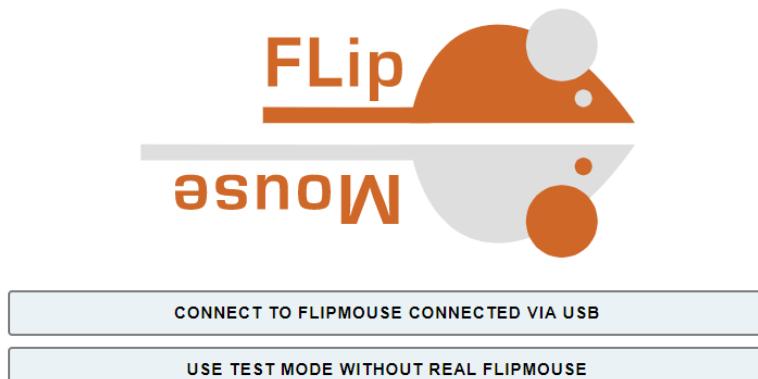


Figure 13: Welcome page of the FLipMouse Configuration Manager

## Connecting the FLipMouse Device

To connect the device, follow these steps:

1. Make sure your device is connected to your computer via an USB port (see page "Instructions for proper mounting and use").
2. Click CONNECT TO FLIPMOUSE CONNECTED VIA USB and select the appropriate COM port (communication port) in the selection box. If the selection box appears empty, this means that no communication port has been detected. In this case, please reconnect the FLipMouse device and wait for the COM port to be updated.
3. Once the COM port is selected, click the "Connect" button at the bottom of the selection box.

4. After successful connection of the COM port, the main window (see Figure 14) will be displayed and you should see live feedback of the stick movement. The port status at the top right corner of the window will show "connected".

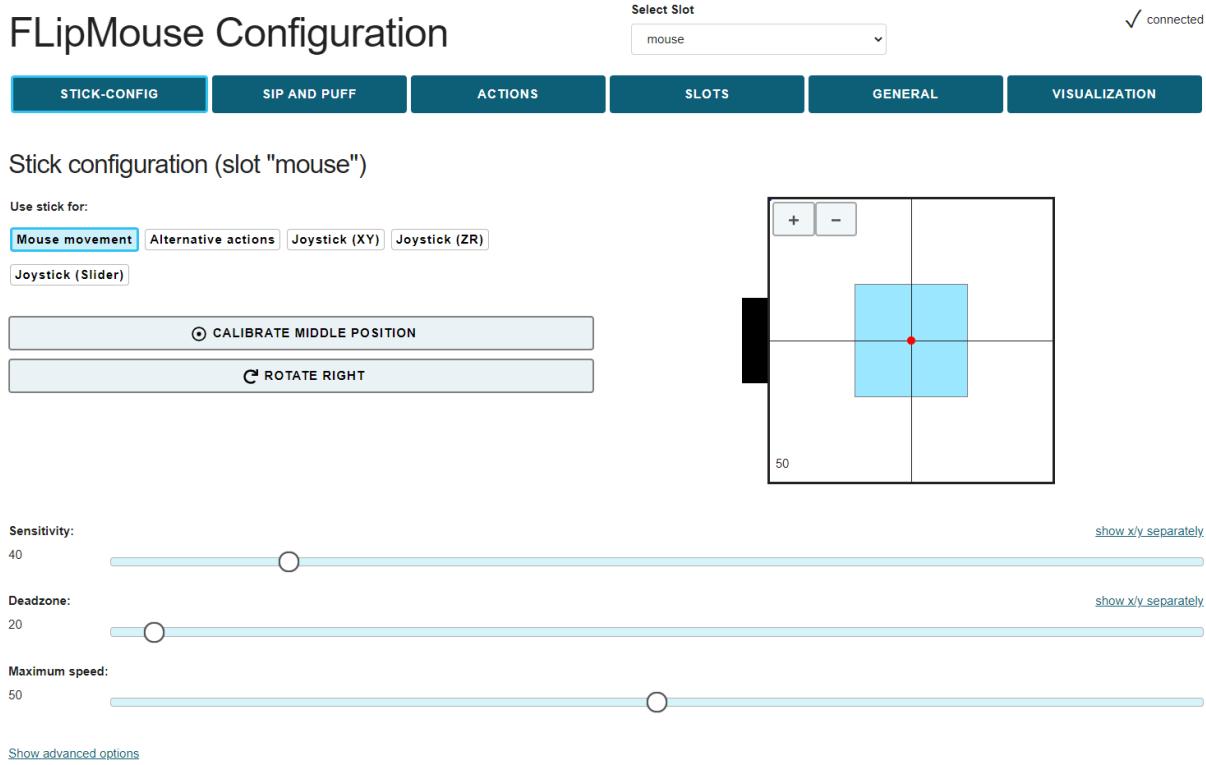


Figure 14: FLipMouse Configuration Manager

## Changing Settings and Features

### Operational modes – “memory slots”

The FLipMouse Configuration Manager allows adjustment of all important settings (operational modes). The settings are stored in individual **memory slots** (for example one slot for fast mouse operation, one slot for slow mouse operation, one slot for keyboard key generation etc.). All settings can be stored into (or loaded from) the FLipMouse device. The settings can also be saved to (or loaded from) a file on your computer.

Settings which were stored into the FLipMouse stay valid also if the power supply / USB cable is removed. When the FLipMouse is plugged in the next time, the settings will be available – also if you use another computer or operating system!

### Stick Configuration (Tab STICK-CONFIG)

Using the STICK-CONFIG tab of the FLipMouse Configuration Manager, you will be able to change the *Sensitivity*, *Deadzone* and *MaximumSpeed* of the FLipMouse stick. Furthermore, you can change *Acceleration*, *Drift compensation range* and *Drift compensation gain* - these settings are visible only if you activate "*Show advanced options*". You may also *Calibrate the middle position* of the stick if necessary and you can change the *Orientation* to reflect the actual mounting position of the FLipMouse.

## Define stick usage (“Use stick for”)

On the top of the STICK-CONFIG tab, the primary function of the stick can be chosen. By default, the stick is creating mouse cursor movements. However, the stick can also be used for alternative actions (such as pressing the key 'A' when the stick is pushed up), which can be specified in the ACTIONS tab (see section [Assignment of different actions](#)). Furthermore, the stick can emulate a real joystick or gamepad (see section [Using the stick for joystick movement](#)).

## Calibration of the stick’s middle position

When using the stick for cursor movement, the cursor should stay in place when the stick is not touched. Sometimes the resting position of the stick may become inaccurate and cause unwanted cursor movements, even if the stick is not used. If this happens, you can either increase the deadzone value (see below) or initiate a calibration. When the Configuration Manager is active, a calibration can be initiated by pressing the “*Calibrate Middle Position*” button. Once you click this button, do not move the FLipMouse stick, otherwise the calibration will be done incorrectly. The LEDs on the device will flash and the speaker will emit a low tone, indicating the start of the calibration. This will last for about 1 second - then the calibration sequence is accomplished. Calibration can also be assigned as an action in response to button pressing, sip/puff actions, or alternative FLipMouse stick actions, so that it can be performed on demand, while using the FLipMouse device.

## Stick Orientation (Change with click on “ROTATE RIGHT”)

The FLipMouse is mounted using a HotShoe 3/8" mounting screw which is located on the side of the FLipMouse case. The orientation can be changed according to the user's preferences. If the FLipMouse is mounted e.g. upside-down, the stick orientation can be selected accordingly so that the up/down/left/right movements are still interpreted correctly. A click on “ROTATE RIGHT” changes the orientation by 90°. Orientation settings for 0 / 90 / 180 and 270 degrees are possible, so that any mounting orientation is possible.

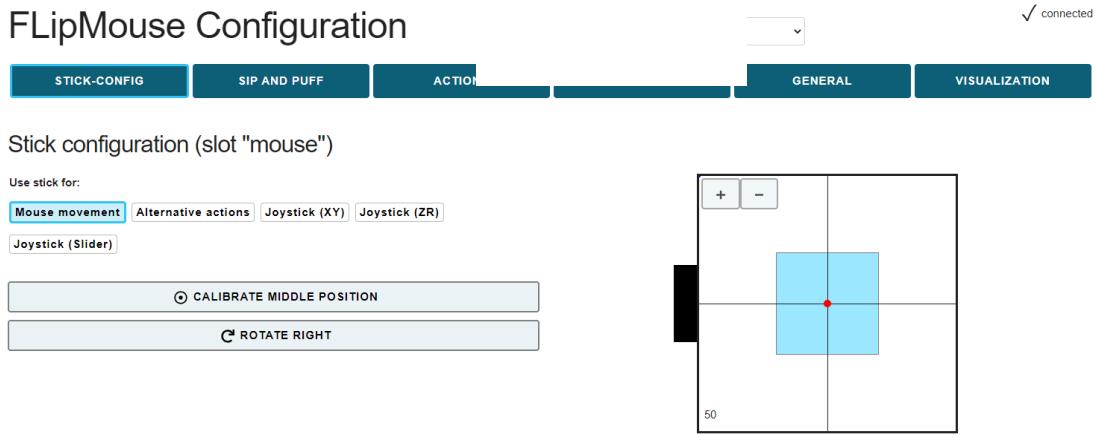


Figure 15: FLipMouse Configuration Manager: Tab STICK-CONFIG, Scroll bars

The scroll bars allow changing the parameters of the stick and the behavior of the mouse cursor according to the preferences of the user. The behavior of the following characteristics can be changed:

## Sensitivity

When using the stick for cursor movement, the sensitivity of the stick can be adjusted via the *Sensitivity* scroll bar. A smaller value results in slower cursor movement. To change the value, click and drag the scroll bar knob or click the bar next to the knob.

## Deadzone

The *deadzone* setting defines a passive zone for the stick movements: when the deadzone value is low, very slight stick movements will already cause cursor movement (or execute the assigned alternative function – see chapter *Assignment of different actions*). If the deadzone setting is too low, the cursor starts unintended drifting especially if a stronger force has been applied before. In this case, increase the deadzone value so that cursor drifting does not occur under normal operating conditions for a given user. (However, for some users it could be desired to use a very small deadzone in order to allow cursor movements with minimal force). For alternative actions (such as key presses), it makes sense to use a bigger deadzone value in order to avoid unintended actions.

## Split axis control for sensitivity and deadzone

If desired, the sensitivity- and deadzone-values for horizontal or vertical movements can be changed individually. In order to do that, select "show x/y separately", as shown below:

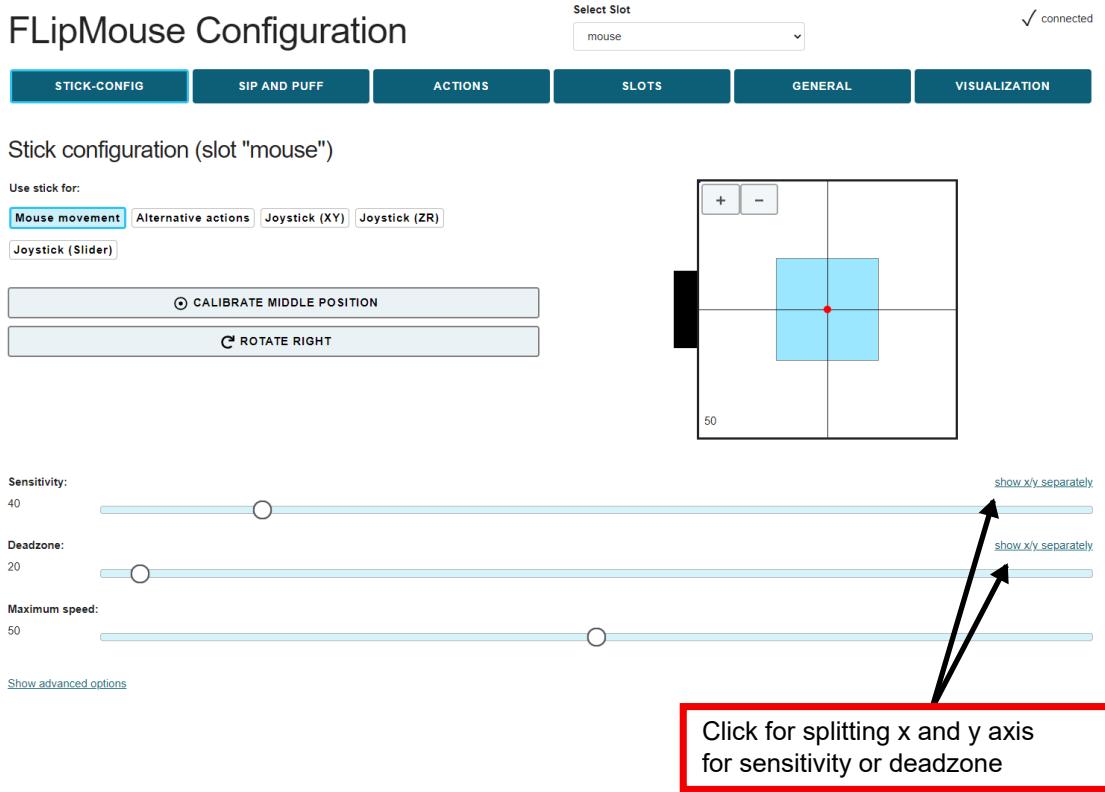


Figure 16: FLipMouse Configuration Manager: Tab STICK-CONFIG, splitting x/y

## Maximum Speed

Turning the maximum speed setting to a low level is useful if a user desires to limit the cursor speed, keeping high sensitivity / acceleration.

"Show advanced options":

## Acceleration

This setting is enabled by clicking "Show advanced options". The acceleration setting allows influencing the acceleration behavior of the mouse cursor: if acceleration is set to a low level, even a strong deflection of the stick will start a slow (but progressing) cursor movement which makes it easier to reach small targets precisely.

## Drift compensation range + Drift compensation gain

These settings are enabled by clicking “Show advanced options”.

Due to mechanical reasons, the force sensors of the FLipMouse show small inaccuracies which can result in mouse cursor drifting. This effect is critical if using very small deadzone settings (very sensitive mouse control): if you move the mouthpiece/joystick to one direction and then release it, the mouse cursor continues to slightly “drift” in this direction because the sensor values do not settle at the calibrated middle position. Since Version 2.9, the FLipMouse offers a “drift compensation” feature, where the software tries to calculate the expected drift caused by the last interaction and corrects the bias, so that the performance is improved especially for small deadzone settings.

The “drift compensation gain” value represents the level of correction, whereas the “compensation range” value represents the amount of force which is taken into account for the calculation.

Each of these values has a corresponding slider. It is also possible to set different values for the x-axis und the y-axis after clicking “show x/y separately” (see the following figure).

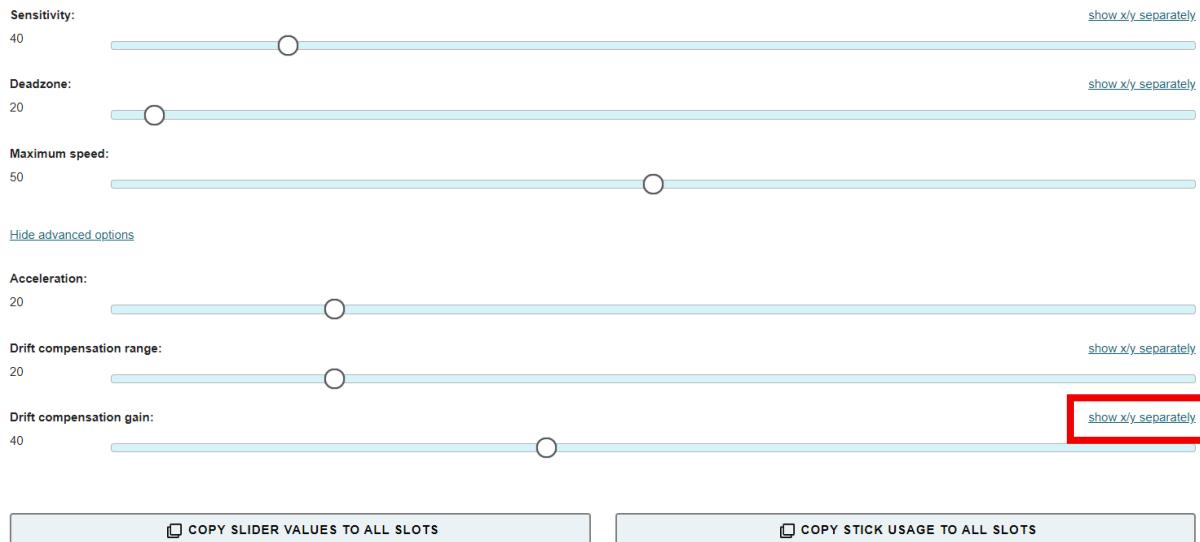


Figure 17: FLipMouse Configuration Manager: Tab STICK-CONFIG, separate x/y

The best values for a particular FLipMouse can be determined by experimentation.

For example:

1. Push the mouthpiece upwards and release it. If the mouse cursor continues to drift upwards, try to increase the vertical compensation value
2. Push the mouthpiece to the left and release it. If the mouse cursor drifts to the right, try to decrease the horizontal compensation value.

## Using the Stick for Joystick Movement

If you select one of the joystick modes in the STICK-CONFIG tab, moving the FFlipMouse stick up / down / left / right will result in joystick activities. The FFlipMouse will behave like a gamepad with 6 axes (*X/Y*, *Z/Z-Turn* and *Slider1/Slider2*). As the FFlipMouse stick can only deliver 2 axes of information at a time, the desired joystick axis must be selected.

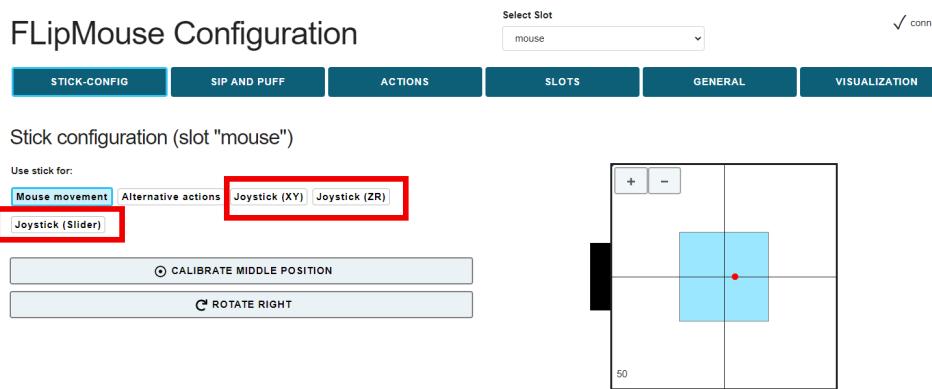


Figure 18: Set stick configuration for joystick movement

Please note that the joystick function must be supported by software applications running on the computer (for example computer games). Microsoft Windows offers a test software called "joy.cpl" – you can start this utility by entering "joy.cpl" into the search dialog.

The joystick controller provided by the FFlipMouse is called "Serial+Keyboard+Mouse+Joystick". When you select this device in the "joy.cpl" utility, click on the properties to display its preferences. The live values of the joystick axis and button activities will be displayed in a window similar to the one shown here:

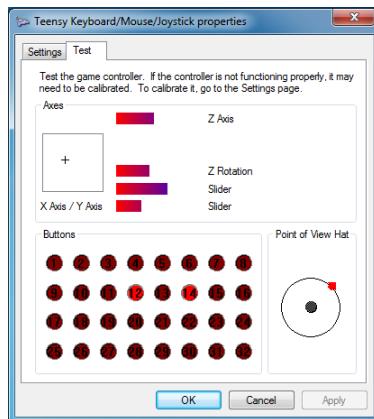


Figure 19: Joy.cpl Properties

Desired joystick button activities can be created using sip/puff or other interaction events by choosing "Press Joystick Button" from the action menu as described in the section [Assignment of different actions](#).

## Sip/Puff Actions and Levels (Tab SIP AND PUFF)

When using the FLipMouse stick with the mouth, the user can trigger actions by sipping or puffing into the mouthpiece. The tube is connected to a pressure sensor, which outputs a value that corresponds to the detected pressure. When the user sips, the sensor value decreases, and when the user puffs, the value increases.

You may assign actions for sipping or puffing using the tab "ACTIONS", as will be explained in the next section. You can select the thresholds for sipping and puffing as desired, using the tab "SIP AND PUFF". When the FLipMouse is connected, you can also monitor the current pressure values and the triggering of sip/puff actions in this tab:

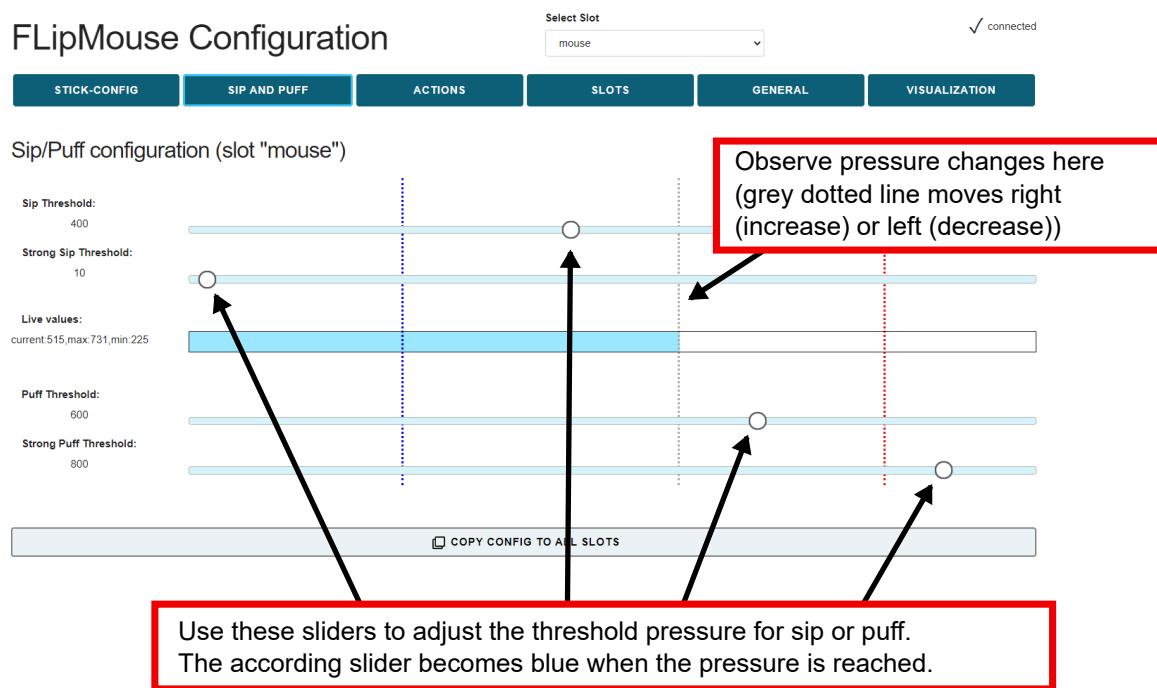


Figure 20: Tab SIP AND PUFF

Note that the idle pressure level (without sip or puff) is around 512, and the level increases when you puff into the mouthpiece, indicating increased pressure.

With the sliders, the threshold pressure for sip and puff can be regulated. The grey dotted line represents the actual pressure. The blue and red dotted lines represent the limits reached in this session.

## StrongSip and StrongPuff

The SIP AND PUFF tab allows to define additional threshold values for StrongSip or StrongPuff activities, which can then trigger distinct actions. For example, the next configuration slot could be activated by a strong puff. To enrich the functionality of the FFlipMouse especially for persons who cannot access external switches, additional StrongSip and StrongPuff gestures are available. These gestures allow to trigger actions via a strong sip or a strong puff followed by a stick movement (up / down / left / right). A strong sip or puff is indicated via an acoustic signal (high tone). If the stick is moved within one second, the corresponding action (for example "StrongSip+Up" is triggered). If the stick is not moved within one second, the single StrongSip or StrongPuff action will be triggered. Thus, a total of 10 additional actions can be performed.

## Assignment of different actions (Tab ACTIONS)

The ACTIONS tab allows the mapping of user activities to desired FlipMouse functions (actions). The user activities are:

- pressing or releasing the 3 buttons (built-in button 1 or external button 2 or 3)
- stick movements (up/down/left/right) which exceed the deadzone threshold value
- sip and puff activities (see section StrongSip and StrongPuff).

The actions can be changed via clicking the entry in the action configuration table (see the blue box in figure 21). The table shows the actions for all user activities, either only for the currently active slot, or for all slots (which gives an overview of all actions). When clicking one particular action, a window pops up where you can change the action category (here for Button 1: Device) and the action itself (here: Load next slot), see Figure 22.

Bezeichnung	Slot "mouse"
Button 1	<a href="#">Load next slot</a> (empty)
Button 2	<a href="#">(empty)</a>
Button 3	<a href="#">(empty)</a>
Stick Up	<a href="#">Joystick (ZR)</a>
Stick Down	<a href="#">Joystick (ZR)</a>
Stick Left	<a href="#">Joystick (ZR)</a>
Stick Right	<a href="#">Joystick (ZR)</a>
Sip	<a href="#">Hold left mouse button</a> (empty)
Strong sip	<a href="#">Click right mouse button</a> (empty)
Puff	<a href="#">Calibrate stick</a> (empty)
Strong Puff	<a href="#">(empty)</a>
Strong Sip + Up	<a href="#">(empty)</a>
Strong Sip + Down	<a href="#">(empty)</a>
Strong Sip + Left	<a href="#">(empty)</a>
Strong Sip + Right	<a href="#">(empty)</a>
Strong Puff + Up	<a href="#">(empty)</a>
Strong Puff + Down	<a href="#">(empty)</a>
Strong Puff + Left	<a href="#">(empty)</a>
Strong Puff + Right	<a href="#">(empty)</a>

Figure 21: Assign different actions to user activities

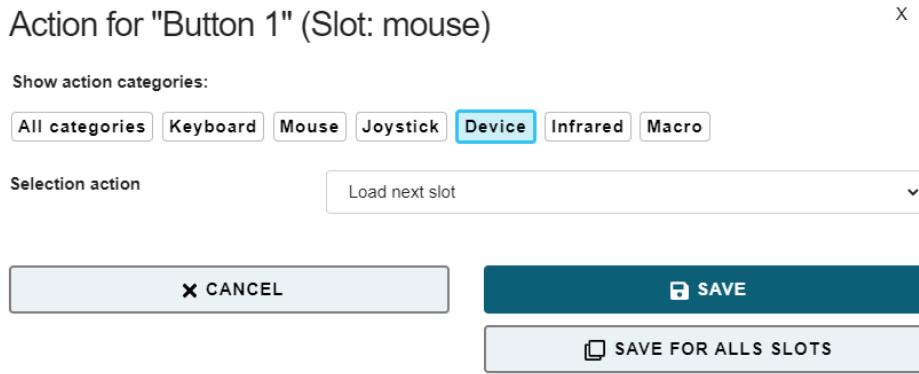


Figure 22: Pop-up window for choice of desired action

In the following, the individual action categories and different actions will be briefly described.

## Action category “Mouse”

### **Hold Left / Middle / Right Mouse Button**

The *hold* action presses a particular mouse button continuously (for example for dragging an item across the screen). The mouse button is released when the assigned user activity ends (for example when the sip/puff activity ends, when the stick is moved back to middle position or when an external button is released).

### **Click Left / Middle / Right Mouse Button**

Creates mouse button clicks. (The left/right or middle mouse button is pressed and released again after a short time).

### **Double Click Left Mouse Button**

Creates a double click with the left mouse button.

### **Toggle Left / Middle / Right Mouse Button**

The *toggle* action changes the state of a mouse button from pressed to not pressed and vice versa.

This is useful e.g. if keeping a user activity up for a longer time (e.g. for dragging an item or for keeping a key pressed while other keys are pressed/released).

### **Wheel Up / Down**

The actions *Wheel up* or *Wheel down* emulate the mouse scroll wheel. Triggering the *Wheel up* action results in upwards scrolling, while *wheel down* results in downward scrolling. This action is useful for example for reading documents or web sites.

### **Move Mouse horizontally / vertically (x-axis / y-axis)**

The *Move mouse horizontally (x-axis)* and *Move mouse vertically (y-axis)* functions generate computer mouse movements along the selected axes. Speed parameters can be set for these functions in the field that appears below. As long as the user activity is present, the mouse pointer is accelerated up to this maximum speed. Please note that:

A positive value for the X direction moves the mouse pointer to the right.

A negative value for the X direction moves the mouse pointer to the left.

A positive value for the Y direction moves the mouse pointer down.

A negative value for the Y direction moves the mouse pointer up.

## Action category: Joystick

### Joystick set x-/y-/z-axis/slider

These actions can be used to send a desired value to the joystick axis. The selectable joystick axes are: *X/Y/Z/Z-Turn / slider*. The joystick is moved back to middle position when the associated user activity ends.

### Press Joystick Button

This action can be used to press a desired joystick button. The FLipMouse device supports 32 buttons, so any value from 1 to 32 is allowed. The joystick button is released when the associated user activity ends.

### Joystick hat position

This action sets the orientation of the joystick "hat" (in degrees). Allowed values are:

-1, 0, 45, 90, 135, 180, 225, 270, 315. The value of -1 sets the hat to the middle (idle) position.

The joystick hat is moved back to middle position when the associated user activity ends.

## Action category: Keyboard

### Press / Hold / Toggle Keys

The *Press Keys* action allows pressing one or multiple keyboard keys. A second drop-down menu with possible key identifiers is displayed (**Add keys**). **When a key is selected from this menu, it has to be added (Click ADD)** to the *Insert keys* text field. Thus, multiple keyboard keys can be pressed simultaneously. The keys are released when the associated user activity ends. If you would like to remove or change the assigned keys, you must clear the currently assigned keys by pressing the "CLEAR" button next to the *Insert keys* field.

Common key combinations include:

KEY\_CTRL + Z: triggers undo function

KEY\_CTRL + C: triggers copy function

KEY\_CTRL + V: triggers paste function

KEY\_CTRL + KEY\_ALT + KEY\_DELETE

The *Hold Key* action keeps the key pressed until the user activity ends.

The *Toggle Key* action changes the key state from pressed to not pressed and vice versa with each user activity.

For a list of all supported key identifiers, see [appendix](#)

or have a look at the [FLipMouse Wiki / Github pages](#).

## Action for "Button 1" (Slot: mouse)

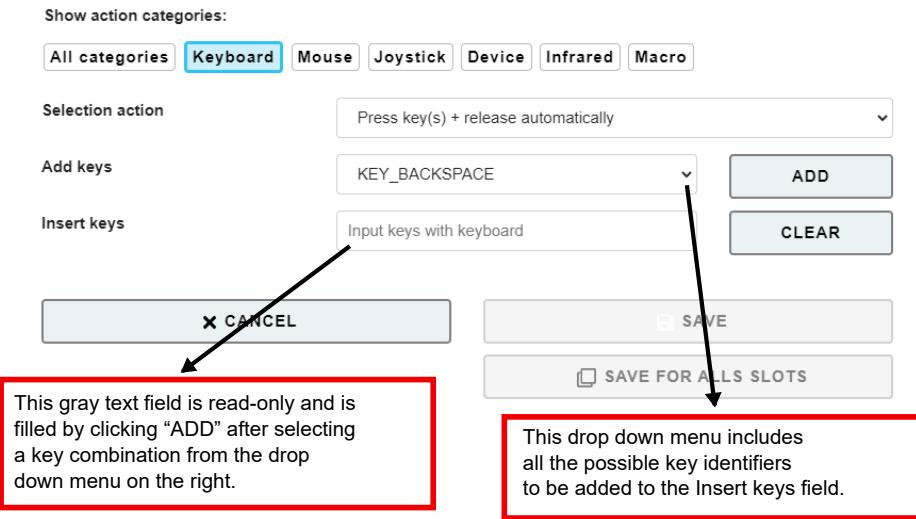


Figure 23: Add keys for the action press / hold / toggle keys

## Write Word

The *Write word* action allows typing a particular text/phrase whenever a user activity occurs (for example: write “Hello” whenever you move the FFlipMouse stick up). Selecting this action displays a blank text box under the drop-down menu where the text can be entered, as shown below:

## Action for "Button 2" (Slot: mouse)

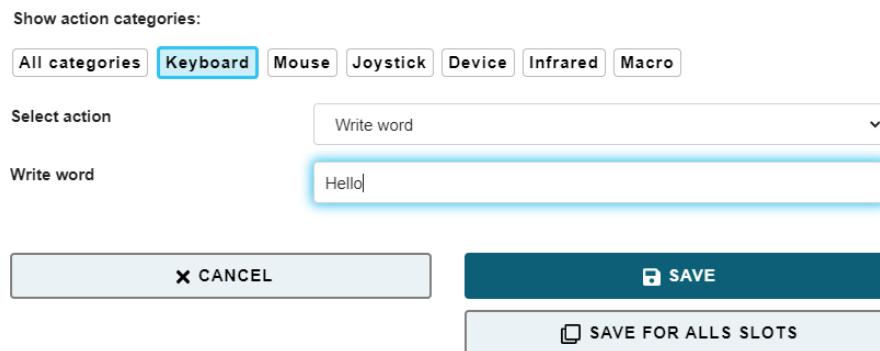


Figure 24: Write word function

In this example, "Hello" will be written each time button 2 of the FFlipMouse is pressed.

## Action category: Device

### No Command

If the action *No Command* is selected, no action will be assigned to the respective user activity.

### Load Next Slot

This action is only relevant if you have stored multiple FLipMouse configuration slots. This action switches to the next slot. If the last slot is already reached, triggering this action will switch to the first slot. When you switch the slot, the built-in LED bulbs will change accordingly, showing the active slot.

There are three built in LEDs (red, yellow-orange, green), which will display the binary number for the slot position of the configuration that you have selected. In other words, if you saved two configurations, e.g. "Game settings" and "Mouse", "Game settings" will be slot 1 and "Mouse" will be slot 2.

Additionally to the LEDs, an acoustic signal will indicate the change of the slot.

The following list shows the color codes and the acoustic signals of the LEDs for each slot position:

Active Slot	Acoustic Signal	Shining LEDs
Slot 1	one beep tone	red
Slot 2	two beep tones	yellow
Slot 3	three beep tones	red, yellow
Slot 4	four beep tones	green
Slot 5	five beep tones	red, green
Slot 6	six beep tones	yellow, green
Slot 7	seven beep tones	red, yellow, green

The next figure (Figure 25) shows a similar list but the shining bulbs in the according colors for the visual people. The different slots are called "Position" in this figure.

Position 1 will be displayed as 1 shining bulb:



Position 2 will be displayed as 1 shining bulb:



Position 3 will be displayed as 2 shining bulbs:



Position 4 will be displayed as 1 shining bulb:



Position 5 will be displayed as 2 shining bulbs:



Position 6 will be displayed as 2 shining bulbs:



Position 7 will be displayed as 3 shining bulbs:



Figure 25: Color codes of the LEDs

### **Load Slot by Name**

This action activates the configuration slot with the given name (if it exists). The name can be specified in the text field. This action is only relevant if you have stored multiple FFlipMouse configuration slots. The LEDs indicate the slot number as described above.

### **Calibrate middle position**

This action starts the calibration sequence for the middle position of the stick. Please refer to the description "["Calibration of the stick's middle position"](#)".

## **Action category: Infrared**

### **Play / Hold / Stop Infrared Command**

This action replays the infrared (IR)-code with the given command name. The existing (recorded) IR-commands can be selected from the drop-down menu. The *play* action sends the recorded code once, whereas the *hold* action repeats the code until the user activity ends. For more information about infrared codes please refer to section "["Infrared Code Recording and Replay"](#)".

## **Action category: Macro**

### **Execute Command Macro**

This action allows running a so-called command macro which consists of several individual actions. Thus, a desired sequence of actions can be performed which consists for example of a number of mouse moves, mouse clicks, text inputs or key presses. This action is powerful, but also a bit more complicated because the individual actions must be specified in form of so-called AT-commands which are separated by semicolons. For example: the following command macro will move the mouse cursor 100 steps to the left, 20 steps up, then wait 100 milliseconds and then perform a left mouse click: "MX 100; MY -20; WA 100; CL". For a list and detailed explanation of all supported AT-commands see [appendix](#) or have a look at the [FFlipMouse Wiki / Github pages](#).

## **Using Configuration Slots (Tab SLOTS)**

In the SLOTS tab you can create new slots, delete existing slots, download a single or all slots and upload a file containing a complete configuration with multiple slots. After starting the FFlipMouse Configuration Manager, only one default slot named "mouse" exists. When you create a slot using the "CREATE SLOT" button, **the current slot is copied and appended** after the last existing slot. Before creating the new slot, assign a name in the corresponding field ("insert name for new slot") that will help you remember the slot's purpose:

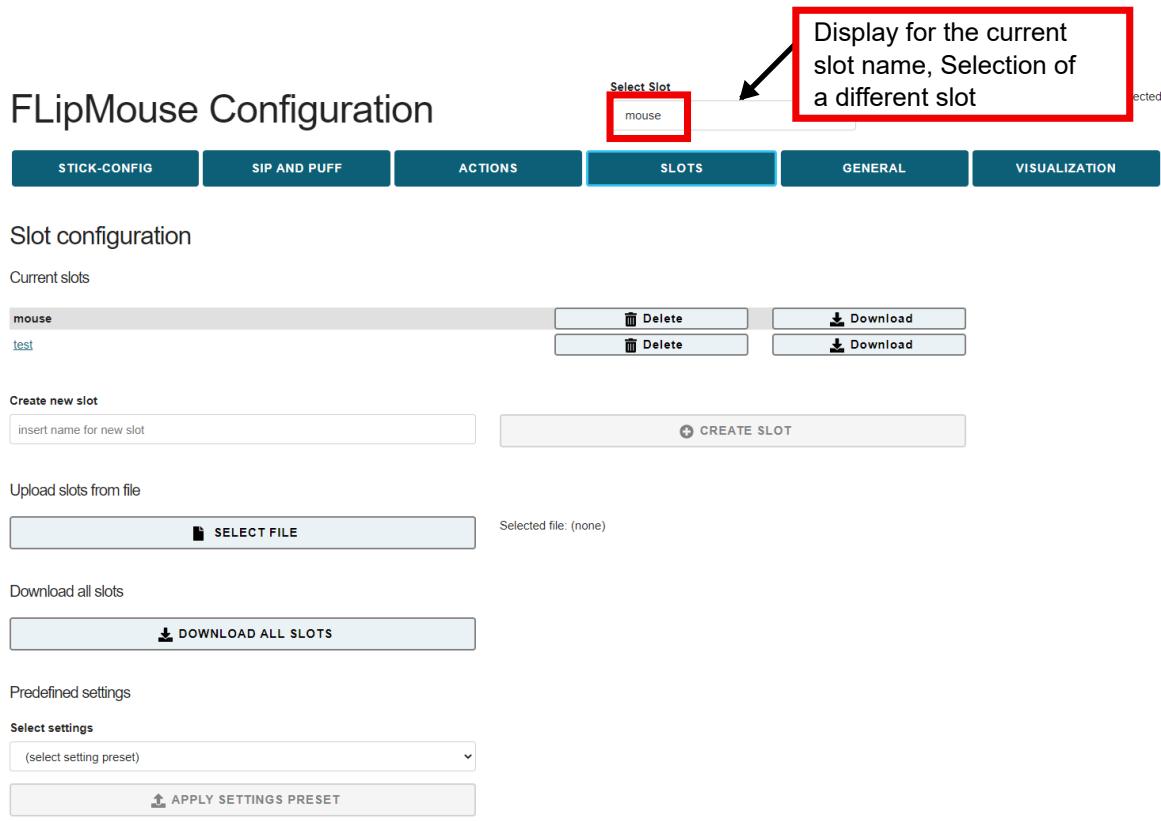


Figure 26: FLipMouse Configuration Manager: Tab SLOTS

At the top center of the Configuration Manager window (blue box in Figure 26) the current slot is shown and can be changed.

The settings of the current slot will be stored and remembered when changing to another slot. The maximum number of slots is 10.

### Delete a slot

A slot can be removed using the *Delete* button next to the according slot.

### Loading and saving the configuration to/from file

The button *Download all slots* allows transferring all current slots to a setting file (.set) which is stored on your computer. All settings can thus be transferred to the same or to another FLipMouse device. Multiple setups (for example for individual users or use-cases) can be kept on a computer and applied with a single click. A file chooser window will be opened which allows selection of the desired filename for saving or loading the configuration.

## General Settings (Tab GENERAL)

The GENERAL tab allows changing the USB/Bluetooth mode and apply firmware updates of the FLipmouse device and the Bluetooth Add-On module (if installed):

The screenshot shows the 'General' tab selected in the top navigation bar. A dropdown menu labeled 'Select Slot' is set to 'mouse'. A status indicator in the top right corner shows a checkmark and the word 'connected'. The main content area includes sections for 'Firmware versions' (with 'FLipMouse Firmware' and 'Firmware Bluetooth-Addon' tabs) and a 'Reset to default configuration' button.

Figure 27: General Settings

## USB/Bluetooth mode

This setting is only relevant if a Bluetooth Add-On module is used (see [FFlipMouse Wiki pages](#) for more information). Using this combo box selection, it is possible to decide if mouse / keyboard actions of a particular slot are sent via USB cable, via Bluetooth connection, or both. Thus, dedicated slots for USB and Bluetooth can be created, so that a user can switch e.g. from a laptop computer (where the FFlipMouse is connected via cable) to a wireless connection (phone / tablet) and back.

### FFlipMouse Firmware (“UPDATE FIRMWARE”)

Here you can see the installed version and the available version of the FFlipMouse software (firmware). If the installed version is older than the version which is available online in the FFlipMouse Github repository, the button caption shows “UPDATE FIRMWARE”. Clicking the button will attempt to download the latest firmware and install it. If the installed version is newer or the same as the available version, the button caption shows “OVERWRITE FIRMWARE” - clicking the button will also download and install the version which is available online.

### Firmware Bluetooth-Add-on (“UPDATE BLUETOOTH FIRMWARE”)

Here you can see the installed version and the available version of the Bluetooth Add-On module’s firmware. Similar to the FFlipMouse firmware update you can click the button “UPDATE / OVERWRITE BLUETOOTH FIRMWARE”, which will attempt to download the latest software (firmware) for the Bluetooth module and install it on the module. This process can take a few minutes and is only possible if a Bluetooth module is connected to the FFlipMouse. If there is no Bluetooth module connected to the FFlipMouse, a message will indicate that the installed version is unknown.

## View Force Levels (Tab VISUALIZATION)

The FLipMouse stick is attached to four force sensors – one for each movement direction (up, down, left, right). The numeric value of each of those sensors is displayed in the “VISUALIZATION” tab. The live values can be monitored when the FLipMouse device is connected. Whenever the stick is moved, the sensor values change. Specifically, a movement in each direction increases the respective sensor value. The sensor values also depend on the use of the adjustment screws which change the force which is applied to the sensor pads via 4 metal springs. Therefore, when assembling the FLipMouse, the live sensor values should be used as guidance. **When not touching the stick, the values should look similar to screenshot below. All values should decrease and raise similarly when moving the stick. Significant deviation in one direction indicates a sensor problem. The sensors should be checked and might need to be replaced.**

Visualization of current state

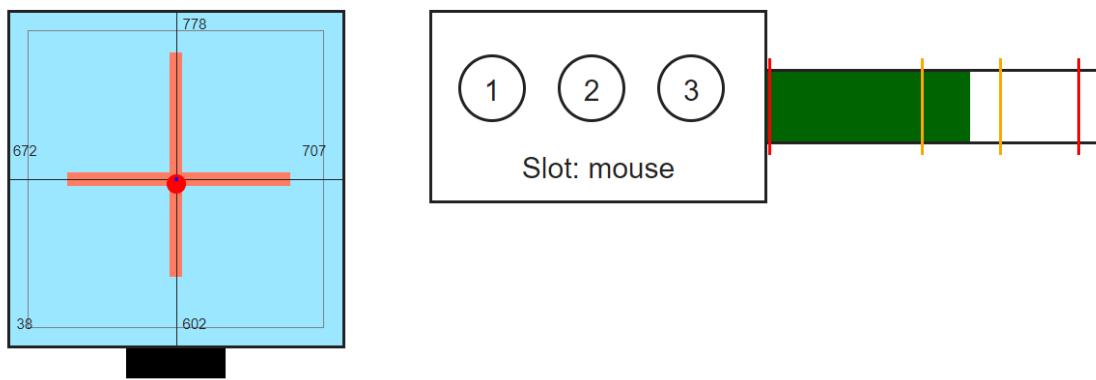


Figure 28: Left: Live display of force sensor values; Middle: current slot with the 3 buttons (if one button is pressed, the corresponding circle will get yellow; Right: Sip and Puff live value and current thresholds

## Infrared Code Recording and Replay

The FLipMouse device includes an infrared remote control receiver module and a high current IR LED so that it can record and replay remote control codes of many remote control devices used in consumer electronics. If you want to use this feature of the FLipMouse, make sure that the IR sending LED is points towards the device you want to control. For instructions how to connect an external IR LED to increase the transmit power, please visit the [FLipMouse Wiki pages](#).

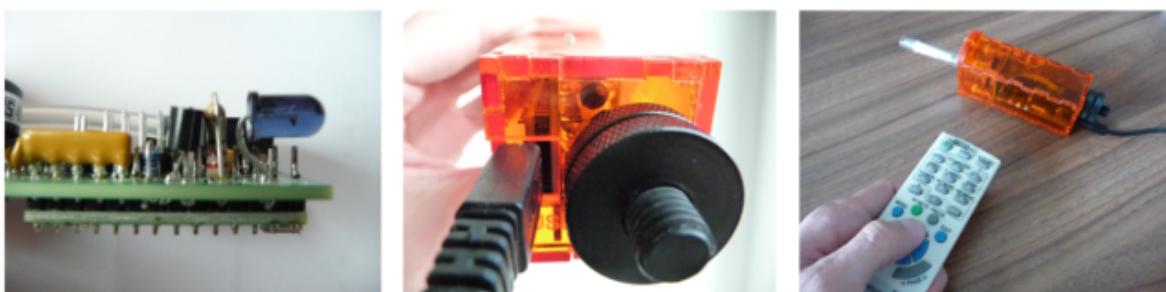


Figure 29: Infrared Code Recording

To record a new IR code, go to the tab ACTIONS. Click the desired action and select the action category “**Infrared**” in the popup-window. There, three different types of IR-actions can be selected from the combo box:

- *Play infrared command*: sends the recorded code once
- *Hold infrared command*: repeats the code until the user activity ends
- *Stop infrared command*: stops a currently playing IR-code

The existing commands can be chosen from the drop-down menu.

In this window you can also manage IR commands. Next to *New IR Command*, you can enter a name and then click RECORD - point your IR remote to the side of the FLipMouse where the LEDs stick out and send the IR command (there is a 10 seconds time limit for the recording phase). The new command is now stored in the FLipMouse and can be chosen and replayed by selecting its name in the drop down menu. If the command is not correctly replayed, try to increase the IR-Code timeout - for example to 250 milliseconds. Using *Delete IR command* commands can be deleted.

Action for "Button 3" (Slot: mouse) X

Show action categories:

All categories **Keyboard** Mouse Joystick Device Infrared **Macro**

Select action  Play infrared command

Play infrared command play

Manage IR Commands

New IR Command Name

Delete IR command play

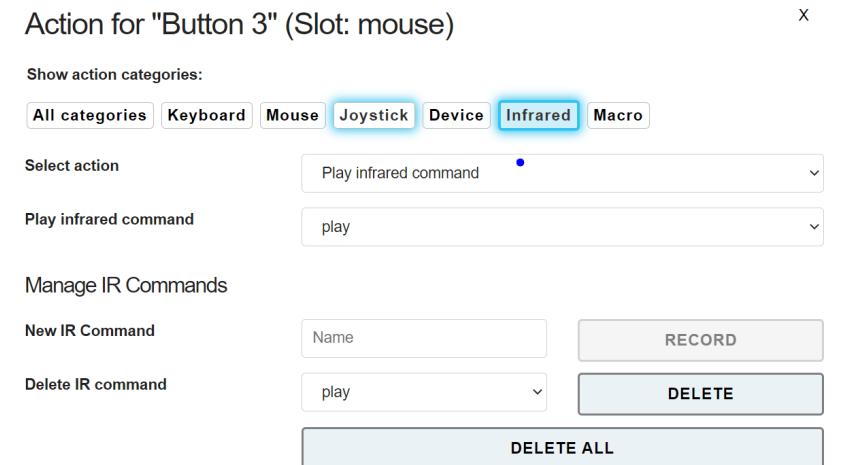


Figure 30: Infrared Recording and Replay Configuration Manager

# Using the Bluetooth module

---

The optional Bluetooth Add-on module allows connection and control of cell phones, tablets and computers with Bluetooth capability. If the FLipMouse is attached to a PC or laptop via USB, users may switch to Bluetooth operation and back to USB on demand. The Bluetooth module is available separately from the AsTeRICS Foundation or included in the appropriate version of the FLipMouse kit.

## Installation of the Bluetooth module

The Bluetooth module is plugged onto the internal 10-pin connector of the FLipMouse. To do this, open the FLipMouse housing and push the module onto the pin header as far as possible:

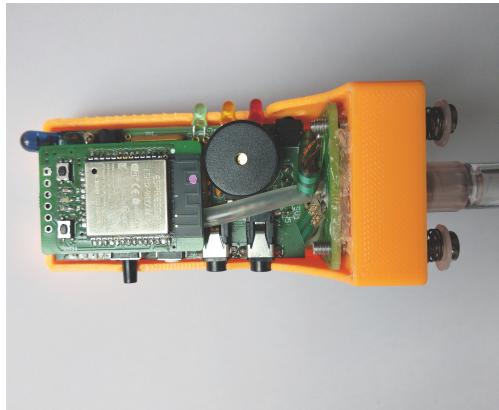


Figure 31: Bluetooth module

## Connecting with a Bluetooth host device (pairing)

The host device could be a cell phone with Android or iOS operating system, for example. The FLipMouse can be connected to a host device only if no device is currently connected, and thus pairing mode is active. In order to connect a device, open the Bluetooth settings of your Android or iOS device, enable BT, choose *add a new BT-device* and select the FLipMouse from the list of available devices. Then, open the GENERAL tab and activate Bluetooth operation for the desired slots (see section [General Settings](#)).

Note: If the Bluetooth module is in paring mode, the module's LED flashed fast (approx twice per second). If a connection is established, the LED flashes slowly (approx. once every 2 seconds). The LED can only be seen if the enclosure of the FLipMouse is opened.

# Mod your Mouthpiece?

Several users reported that they would prefer a different shape of the mouthpiece. It is of course possible to apply your own modifications and extensions which feel comfortable to you. Here we want to give some suggestions using affordable, disposable cigarette holders. The link between the *Hygienic Stick* and the cigarette holder is established with a small brass tube. It is also possible to cut the cigarette holder and insert it into the *Hygienic Stick* so that a tight fit is ensured.



Figure 32: Modified shape of the mouthpiece

Regardless of your desired mouthpiece modifications: Always take care for a hygienic operation and apply the cleaning procedures as described in the following chapter!

# Instructions for cleaning and maintenance

---

If the mouthpiece is used to actuate the cursor movements via lip/mouth movements, a careful cleaning procedure is essential to avoid bacteria and health problems! Read this section carefully and apply the needed cleaning procedures whenever necessary.

## **Exchanging the saliva filter:**

The saliva filter is placed inside the *Hygienic Stick*. Usually, no (or nearly no) saliva should enter the stick – but it strongly depends on the person using the mouthpiece and his/her condition. If saliva enters the *Hygienic Stick*, the saliva filters prevents the moisture entering the inner parts of the FLipMouse (respectively the pressure sensor). The filter can be replaced as shown below:



Figure 33.1: Removal of the saliva filter, using a normal screw to grab the filter



Figure 33.2: The removed filter



Figure 33.3: Inserting a new filter (pushing it back into the hygienic stick)

## **Disinfection of the mouthpiece/hygienic stick:**

To disinfect a mouthpiece, use boiling water. Place the *Hygienic Stick* (without saliva filter) in boiling water for at least five minutes. Then let the *Hygienic Stick* dry before the next application.

## **Ordering replacement sticks and filters:**

Saliva filters and *Hygienic Sticks* can be directly ordered from the [Quadjoy online shop](#). We would like to thank Bill and Debra Street for letting us use their *Quadjoy Hygienic Stick* together with our FLipMouse device – you guys are great!

## **Hygienic considerations and risks**

---

If the mouthpiece is actuated with the lips, it is important to respect the following hygienic precautions:

1. One mouthpiece must be used exclusively by one person
2. The filter tip must be replaced whenever saliva or dirt have formed
3. The mouthpiece must be cleaned whenever saliva or dirt have formed  
(see chapter "[Instructions for cleaning and maintenance](#)")

With the lack of hygiene concerning the mouthpiece, especially when the mouthpiece is used by more than one person, it is possible that

1. skin rash arises
2. viruses and bacteria are transmitted causing diseases
3. dirt formats in the mouthpiece, which is dangerous to health

With wrong positioning of the FLipMouse there is the risk of feeling uncomfortable after longer periods of use, and it is possible to get muscle cramps or muscle pain due to unusual head positioning – pay attention to a convenient position and schedule pauses if you work for longer time periods!

# Updating the Firmware via Arduino IDE

In addition to the possibility of updating the FFlipMouse firmware via the configuration manager, the firmware can also be updated via the Arduino IDE and the Teensyduino add-on / loader application. Find the most recent versions in the latest release package on [Github](#). Unzip the FFlipMouse.zip package and start the teensy.exe program (the Teensy Loader).

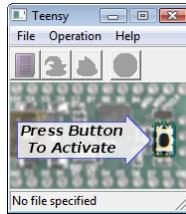


Figure 34: Teensy Loader Step 1

The Teensy Loader user interface should look as on the picture above (Figure 34). If there is any security warning (related to Windows User Access Control), please ensure that the Publisher is "PJRC.COM, LLC" and click on "Run".

As next step, activate the "download mode" of the FFlipMouse by using some pointed tool (needle, small screwdriver, ...) to press the reset button which is accessible via the small hole on the bottom side. After a few seconds (driver installation), the Teensy Loader GUI should indicate connection to the FFlipMouse as shown here in the following Figure:

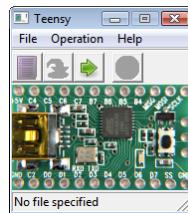


Figure 35: Teensy Loader Step 2

Choose "File → Open HEX file" and select the file "FFlipWare.hex" from the FFlipMouse folder (or location where you downloaded the release .zip file).

Select "Operation → Program". After a few seconds, you should see the notification "Download complete", as on the next picture (Figure 36). (If this step does not work in the first place, try to unplug and replug the FFlipMouse and restart the Teensy Loader software.)



Figure 36: Teensy Loader Step 3

To finish the firmware installation, select “Operation → Reboot”.

You should read “Reboot” and the FLipMouse should beep for a short time.

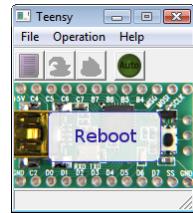


Figure 37: Teensy Loader Step 4

To update the GUI software, copy the file “FLipMouseGUI.exe” from the zip package to the desired location (replacing the old FLipMouseGUI.exe file).

# Building the Firmware

If you want to adapt the software (firmware) of the FLipMouse, follow these steps:

1. Clone the FLipMouse repository, <https://github.com/asterics/FLipMouse>
2. Download and install the Arduino IDE, <https://www.arduino.cc/en/main/software>
3. Download and install Teensyduino, [https://www.pjrc.com/teensy/td\\_download.html](https://www.pjrc.com/teensy/td_download.html)
4. Open FLipWare/FLipware.ino with the Arduino IDE, see: <https://github.com/asterics/FLipMouse/blob/master/FLipWare/FLipWare.ino>
5. Use the following settings in Arduino IDE:
  - o Tools -> Board -> Teensy LC
  - o Tools -> USB Type -> "Serial + Mouse + Keyboard + Joystick"

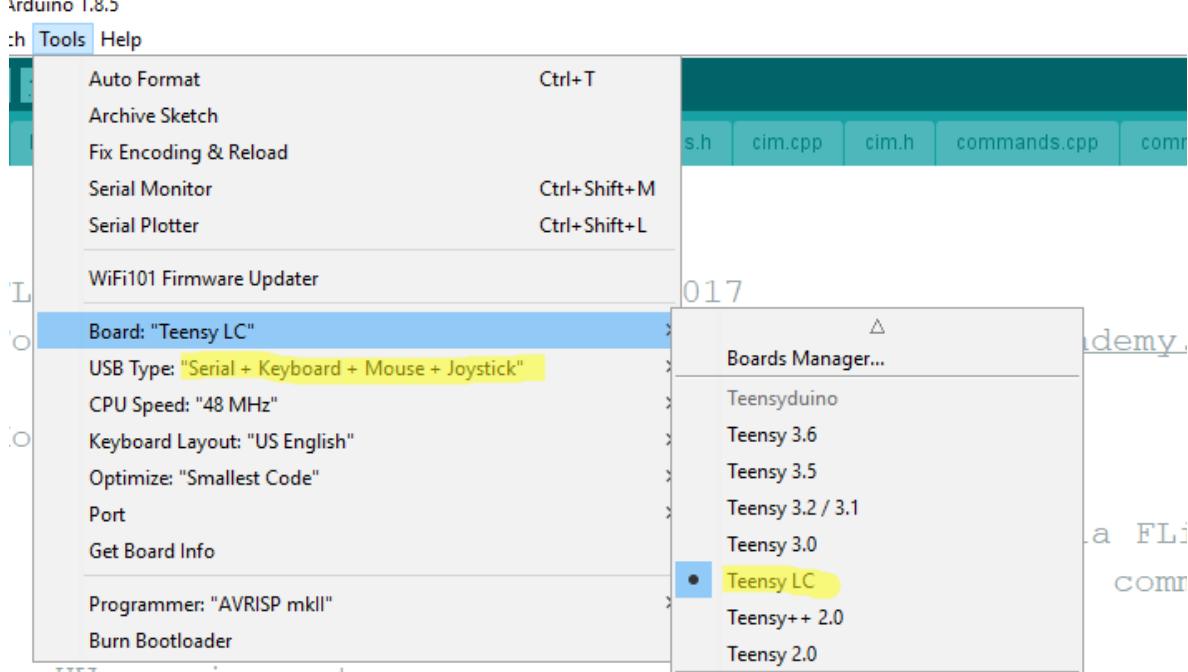


Figure 38: Adapting the Firmware

# **Further instructions and troubleshooting**

---

For further instruction and troubleshooting please refer to the [GitHub Wiki of the FLipMouse project](#).

# Appendix: Macro commands and Key codes

## Supported Macro commands:

Abbreviation	Function	example
CL	click left mouse button	
CR	click right mouse button	
CM	click middle mouse button	
CD	click double with left mouse button	
HL	hold the left mouse button	
HR	hold the right mouse button	
HM	hold the middle mouse button	
TL	toggle left mouse button	changes the state of the mouse button
TM	toggle middle mouse button	
TR	toggle right mouse button	
RL	release the left mouse button	
RR	release the right mouse button	
RM	release the middle mouse button	
WU	move mouse wheel up	
WD	move mouse wheel down	
MX	move mouse in x direction	MX 4 -> moves cursor 4 pixels to the right
MY	move mouse in y direction	MY -10 -> moves cursor 10 pixels up
RO	rotate stick orientation	RO 180 -> flips x and y movements of stick/mouthpiece
KW	keyboard write string	KW Hello! -> writes "Hello!" on the keyboard
KP	key press: Press keyboard keys (once). Keys are identified by keyboard shortcuts (see list below)	KP KEY_UP -> presses the "Cursor-Up" key; KP KEY_CTRL KEY_ALT KEY_DELETE presses all 3 keys

<b>Abbreviation</b>	<b>Function</b>	<b>example</b>
KH	key hold: hold keyboard keys	sKH KEY_LEFT -> holds the "Cursor-Left" key
KT	key toggle: toggle keyboard keys; the key will remain pressed until "AT KT" command is sent again or a "AT KR" command is sent	KT KEY_DOWN -> toggles the "Cursor-Down" key
KR	key release	KR KEY_UP -> releases the "Cursor-Up" key
RA	releases all = Release all currently pressed keys and mouse buttons	
WA	wait a certain number of milliseconds	WA 100 -> waits 100 milliseconds
NE	next slot: load next slot	
LO	Load slot per name, changes to the given slot	LO mouse
NC	No command	
JX	set joystick x axis	JX 512 -> sets the x-axis to middle position
JY	set joystick y axis	JY 1023 -> sets the y-axis to full up position
JZ	set joystick z axis	JZ 0 -> sets the z-axis to lowest poistion
JT	set joystick turn axis	JT 512 -> sets the rotation to middle position
JS	set joystick slider position	JS 512 -> sets the slider to middle position
JP	press joystick button	JP 1 -> presses joystick button 1
JR	release joystick button	JR 2 -> releases joystick button 2
JH	set joystick hat position	JH 45 -> sets joystick hat to 45 degrees; possible values are: 0, 45, 90, 135, 180, 225, 270, 315 and -1 to set center position
MM	mouse mode: cursor on (int=1) or alternative functions on (int=0)	MM1 -> switches to mouse mode

Abbreviation	Function	example
CA	Calibrate zero position	calibrates stick / mouthpiece middle position
IP	play infrared command	IP channelUp; plays infrared command "channelUp" (if this IR-command name has been recorded)

Please note that certain commands for adjusting FLipMouse settings are not shown in this table (for example setting thresholds or accelerations, recording IR-commands etc.). For a full list of commands please refer to the [FLipMouse wiki](#).

## Supported Key Identifiers:

Supported KEY Identifiers:	
<b>Letters</b>	KEY_A KEY_B KEY_C KEY_D KEY_E KEY_F KEY_G KEY_H KEY_I KEY_J KEY_K KEY_L KEY_M KEY_N KEY_O KEY_P KEY_Q KEY_R KEY_S KEY_T KEY_U KEY_V KEY_W KEY_X
<b>Digits</b>	KEY_1 KEY_2 KEY_3 KEY_4 KEY_5 KEY_6 KEY_7 KEY_8 KEY_9 KEY_0
<b>Function keys</b>	KEY_F1 KEY_F2 KEY_F3 KEY_F4 KEY_F5 KEY_F6 KEY_F7 KEY_F8 KEY_F9 KEY_F10 KEY_F11 KEY_F12 KEY_F13 KEY_F14 KEY_F15 KEY_F16 KEY_F17 KEY_F18 KEY_F19 KEY_F20 KEY_F21 KEY_F22 KEY_F23 KEY_F24
<b>Navigation keys</b>	KEY_UP KEY_DOWN KEY_LEFT KEY_RIGHT KEY_TAB KEY_PAGE_UP KEY_PAGE_DOWN KEY_HOME KEY_END
<b>Special keys</b>	KEY_ENTER KEY_SPACE KEY_BACKSPACE KEY_DELETE KEY_INSERT KEY_ESC KEY_NUM_LOCK KEY_SCROLL_LOCK KEY_CAPS_LOCK KEY_PAUSE  KEY_SEMICOLON KEY_COMMA KEY_PERIOD KEY_MINUS KEY_EQUAL KEY_SLASH KEY_BACKSLASH KEY_LEFT_BRACE KEY_RIGHT_BRACE KEY_QUOTE KEY_TILDE KEY_MENU
<b>Keypad keys</b>	KEYPAD_1 KEYPAD_2 KEYPAD_3 KEYPAD_4 KEYPAD_5 KEYPAD_6 KEYPAD_7 KEYPAD_8 KEYPAD_9 KEYPAD_0 KEYPAD_SLASH KEYPAD_ASTERIX KEYPAD_MINUS KEYPAD_PLUS KEYPAD_ENTER KEYPAD_PERIOD
<b>Keys for alternative functions</b>	KEY_SHIFT KEY_CTRL KEY_ALT KEY_RIGHT_ALT KEY_GUI KEY_RIGHT_GUI



# Contact Information

---

University of Applied Sciences Technikum Wien

Faculty of Electronic Engineering

Hochstädtplatz 6

A-1200 Wien

AUSTRIA

AsTeRICS Foundation

Webpage: [www.asterics-foundation.org](http://www.asterics-foundation.org)

Email: [office@asterics-foundation.org](mailto:office@asterics-foundation.org)

# **Disclaimer**

---

The University of Applied Sciences Technikum Wien and the AsTeRICS Foundation do not assume any warranty and liability for the functionality of the described devices, software and the correctness of the documents handed over.

Additionally, the UAS TW is not liable for any damages to health due to the use of described hardware or software. The provided software applications and hardware modules are to be used at own risk!

# Acknowledgement

---

This project was partly funded by the City of Vienna,  
Municipal department of Economic Affairs, Labor and Statistics (MA 23),  
in course of the projects AsTeRICS Academy (14-02), ToRaDes (18-04)  
and Wissensdrehscheibe für Barrierefreie Technologien (26-02)

