

# Input/Output (IO)

## OrxFile

File / file system module. Module that handles file / file system access.

## OrxInput

Input module. Module that handles generalized inputs (keyboard, mouse, joystick, ...)

Input events may be defined either in the config file or in code with `orxInput_Bind*()` functions. They can be changed on the fly, of course, and you can create inputs than don't necessarily map to physical device by using the `orxINPUT_TYPE_EXTERNAL` value of the enum.

It allows to group everything through the generic input system. Events can be bound to whatever any input source: physical device, GUI “buttons” by activating the input using `orxInput_SetValue()`, etc... This way, when going from one peripheral to the other, or when going to another platform, only the bindings need to be changed, the processing code itself doesn't change.

`orxInput_SetValue()` sets the value for one frame. Any value works. It all depends on what you expect when you read it. If mapped to a physical peripheral, buttons will set the value to 1 and joystick axes will set it to X with  $-1 \leq X \leq 1$ .

`orxInput_SetPermanentValue()` sets the value till someone sets it to something different.

## Temporary Disabling User Input

Just to make an input set that does nothing and toggle between it and the other ones as necessary.

## OrxJoystick

Joystick plugin module. Plugin module that handles joystick-like peripherals.

## OrxKeyboard

Keyboard plugin module. Plugin module that handles keyboard peripherals.

## OrxMouse

Mouse plugin module. Plugin module that handles mouse-like peripherals.

## Config

- [Input](#)
- [Joystick inputs](#)
- [Keyboard inputs](#)
- [Mouse inputs](#)
- [Touch events](#)

## Tutorials

- [Grey's Tutorial 4: Interaction and Physics](#) (Warning: You are recommended to start with [Grey's Tutorial 1](#))

From:

<https://orx-project.org/wiki/> - **Orx Learning**

Permanent link:

<https://orx-project.org/wiki/en/orx/reference/io/main?rev=1446085739>

Last update: **2017/05/30 00:50 (8 years ago)**

