Remapping Controller Inputs

Joystick buttons, mouses button or keyboard keys can be mapped. These bindings can be removed in code and rebound. This is especially useful in games where the user is allowed to customise their own keys or changing what joystick button or stick operates what control.

This article is more a presentation of a remapping routine rather than a step by step tutorial.



If you wish to simply use the existing built-in mappings for all controllers, check this article first. If you wish to learn about additional standard mappings for all controllers, see Overriding a Controller mapping.

It introduces the use of the following functions:

- `orxInput_GetActiveBinding`
- `orxInput_GetBindingList`
- `orxInput_Unbind`
- `orxInput_Bind`
- `orxInput_Save`

The routine will allow the user to press any key or joystick button assigned to the `XButton` label in order to trigger a message to the console. They press the `Enter` key to enter remap mode. Then pressing any joystick button will make the button the new one assigned to `XButton`.

Starting with the input config:

```
[MainInput]
KEY_ESCAPE      = Quit
KEY_ENTER  = Remap
JOY_12_1   = XButton ;Joystick binding
KEY_X      = XButton ;Keyboard alternative
```

In the above config, the `Escape` key is used as the usual quit key and the `Enter` key is the one used to put our game into “remap mode”.

Then there are two controls mapped to `XButton`, `JOY_12_1` which is the 12th button on your joystick controller or gamepad, and secondly, the X key.

Now to the routine itself with comments. Have a look though and I'll explain afterwards:

```
    /* Should quit? */
    if(orxInput_IsActive("Quit"))
    {
        /* Updates result */
        eResult = orxSTATUS_FAILURE;
    }
```

```
    if(orxInput_HasBeenActivated("XButton")){
        orxLOG("X Button");
    }

    if (orxInput_HasBeenActivated("Remap")) {
        orxLOG("Remap mode is on.");
        remapMode = orxTRUE;
    }

    if (remapMode) {
        orxINPUT_TYPE inputType;
        orxENUM buttonID;
        orxFLOAT value;

        //Get a button press. This will be the one assigned to 'XButton'
        if (orxInput_GetActiveBinding(&inputType, &buttonID, &value)) {
            if (inputType == orxINPUT_TYPE_JOYSTICK_BUTTON) { //only allow
if it's a joystick button press
                orxLOG("Remapping Type: %d ID: %d Value: %f", inputType,
buttonID, value); // 4 0 ?

                //do the remap here.

                //Get all inputs and buttonIDs assigned to XButton
                orxINPUT_TYPE inputTypes[orxINPUT_KU32_BINDING_NUMBER];
                orxENUM bindingButtonIDs[orxINPUT_KU32_BINDING_NUMBER];
                orxINPUT_MODE inputModes[orxINPUT_KU32_BINDING_NUMBER];
                orxInput_GetBindingList("XButton", inputTypes,
bindingButtonIDs, inputModes);

                //unbind all existing joystick buttons bound to 'XButton'
                for (orxU32 i = 0; i < orxINPUT_KU32_BINDING_NUMBER; i++) {
                    if (inputTypes[i] == orxINPUT_TYPE_JOYSTICK_BUTTON) {
                        orxSTATUS unBindSuccess = orxInput_Unbind("XButton",
i); //remove joystick binding
                    }
                }

                orxInput_Bind("XButton", orxINPUT_TYPE_JOYSTICK_BUTTON,
buttonID, orxINPUT_MODE_FULL, -1);
                orxInput_Save("gamepad.ini");
                orxLOG("Bound to button %d", buttonID);

                remapMode = orxFALSE;
            }
        }
    }
```

First, if either `Joystick button 12` or the X key is pressed, a line is logged to the console.

Next, remap mode can be entered by pressing the `Enter` key.

In remap mode, `orxInput_GetActiveBinding` is constantly called. This will get the very last input that was made whether it be a key, a mouse movement, a joystick press, etc. Here, if it was a joystick press, store it for later, and it will be used as the button to remap to.

`orxInput_GetBindingList` is then called which returns the entire list of bindings to `XButton`. There can be up to four physical inputs to a binding. `orxINPUT_KU32_BINDING_NUMBER` is a constant to 4.

We then loop through the entire list looking for any joystick button type input and unbind `XButton` from them.

`orxInput_Bind` is then called using the joystick button that was saved at the start using `orxInput_GetActiveBinding`. This is bound to `XButton`.

Finally, `orxInput_Save` is called to save the new input config section to a new file. This can then be loaded whenever your game is started.

Finally remap mode is turned off, and if you press the new joystick button that you chose, it will start to log to console instead of the previous joystick button.



Note: when using `orxInput_GetActiveBinding`, it only collects an input for the specific frame. A slow custom clock will miss values especially if they are fast inputs like mouse wheel clicks. Therefore, it is recommended that the **core clock** is used. See here for an example.

From:
https://www.orx-project.org/wiki/ - **Orx Learning**

Permanent link:
https://www.orx-project.org/wiki/en/tutorials/input/remapping_inputs?rev=1598877859

Last update: **2025/09/30 17:26 (14 months ago)**