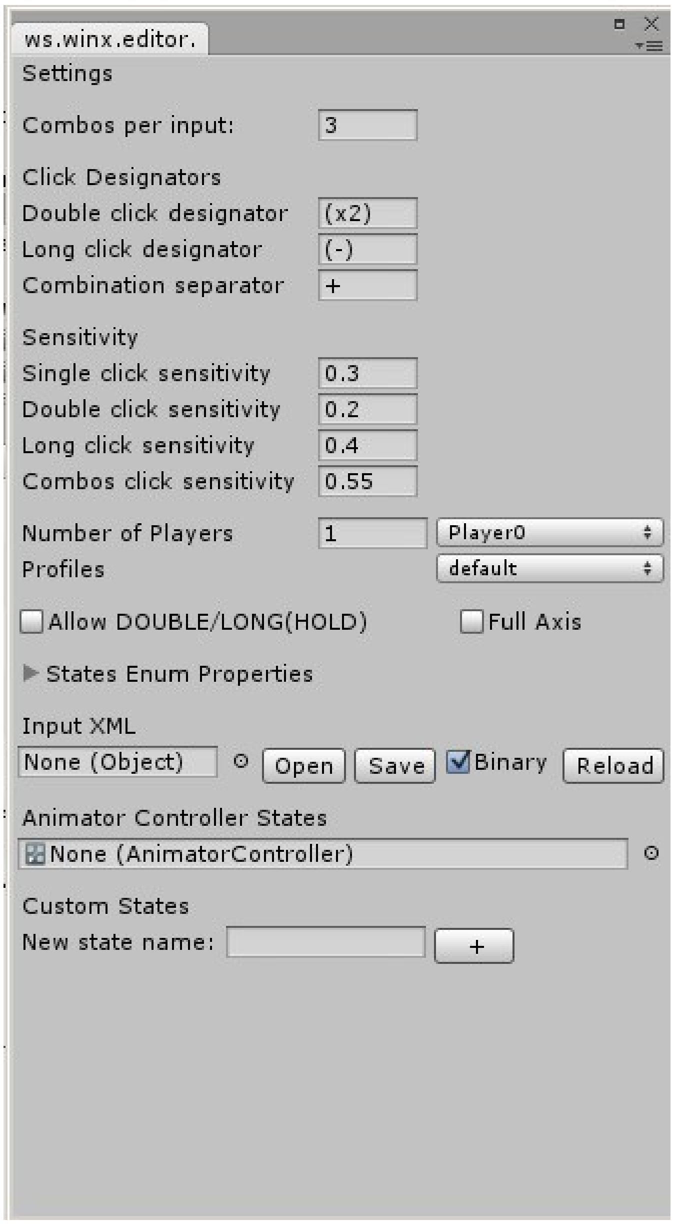
1) Install the package
2) Put gmcs.rsp and smcs.rsp in root
3) Put „InputSettings.xml“ in root StreamingAssets folder from package StreamingAssets folder if you want to hit play on test demo
4) Connect your devices
5) Open Window->InputMapper->InputMapper editor



ws.winx.editor.

Settings

Combos per input: 3

Click Designators
Double click designator (x2)
Long click designator (-)
Combination separator +

Sensitivity
Single click sensitivity 0.3
Double click sensitivity 0.2
Long click sensitivity 0.4
Combos click sensitivity 0.55

Number of Players 1 Player0
Profiles default

☐ Allow DOUBLE/LONG(HOLD) ☐ Full Axis

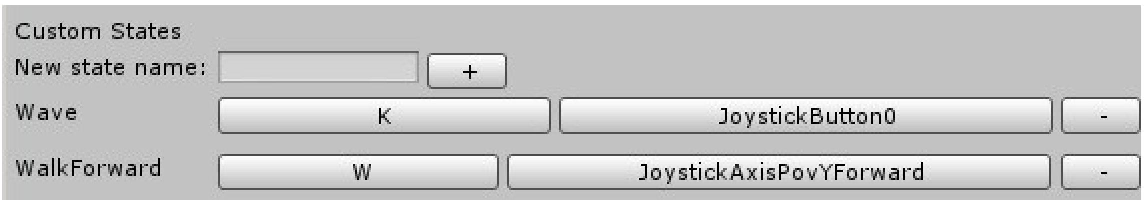
▶ States Enum Properties

Input XML
None (Object) Open Save ☑ Binary Reload

Animator Controller States
None (AnimatorController)

Custom States
New state name: +

6) Create custom states



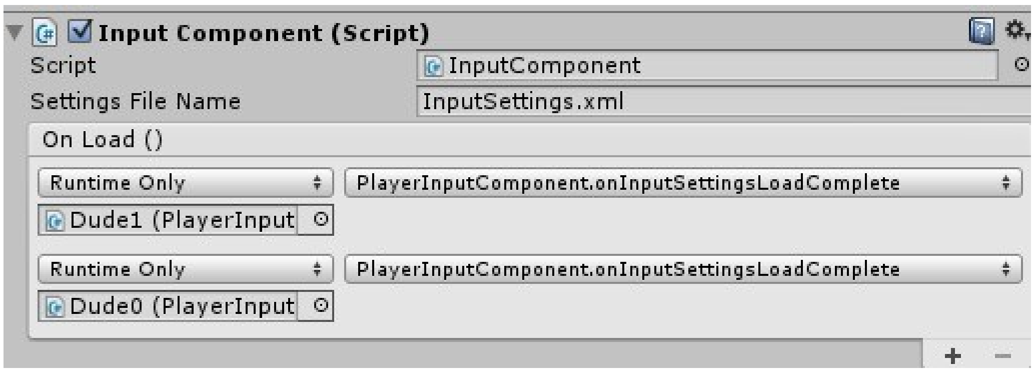
Custom States
New state name: +

Wave	K	JoystickButton0	-
WalkForward	W	JoystickAxisPovYForward	-

Advanced:

```
InputManager.MapStateToInput("Click_W+C_State",KeyCodeExtension.W.DOUBLE,KeyCodeExt
ension.C.LONG);
```

7) Map your input into combination by clicking buttons or moving joysticks. (single,double,long(hold) action)
8) Save your settings as .xml or .bin (States.cs would be generated for you so you can access your states as States.MyState)
9) Drag and drop InputComponent to some GameObject



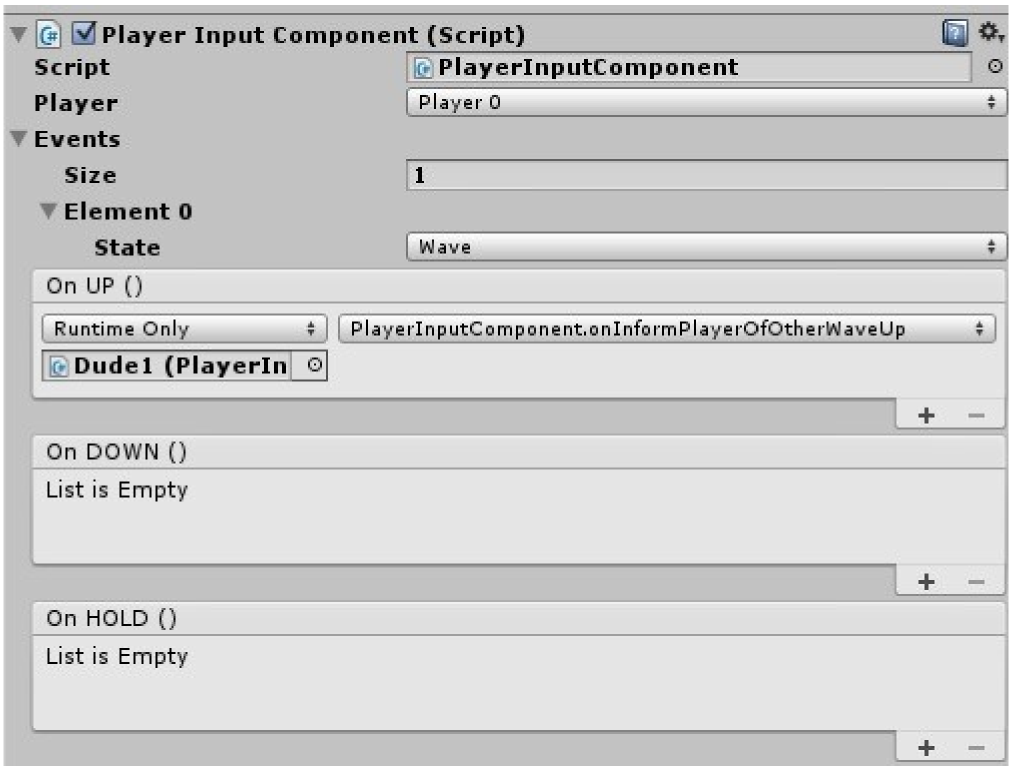
Input Component (Script)
Script: InputComponent
Settings File Name: InputSettings.xml
On Load ()
Runtime Only — PlayerInputComponent.onInputSettingsLoadComplete — Dude1 (PlayerInput
Runtime Only — PlayerInputComponent.onInputSettingsLoadComplete — Dude0 (PlayerInput

Component would handle loading and raise event when loading is complete.

Advanced:

```
 InputManager.loadSettings(Path.Combine(Application.streamingAssetsPath,"Inpu
tSettings.xml"));
```

10) Drag and drop PlayerComponent to character's game objects

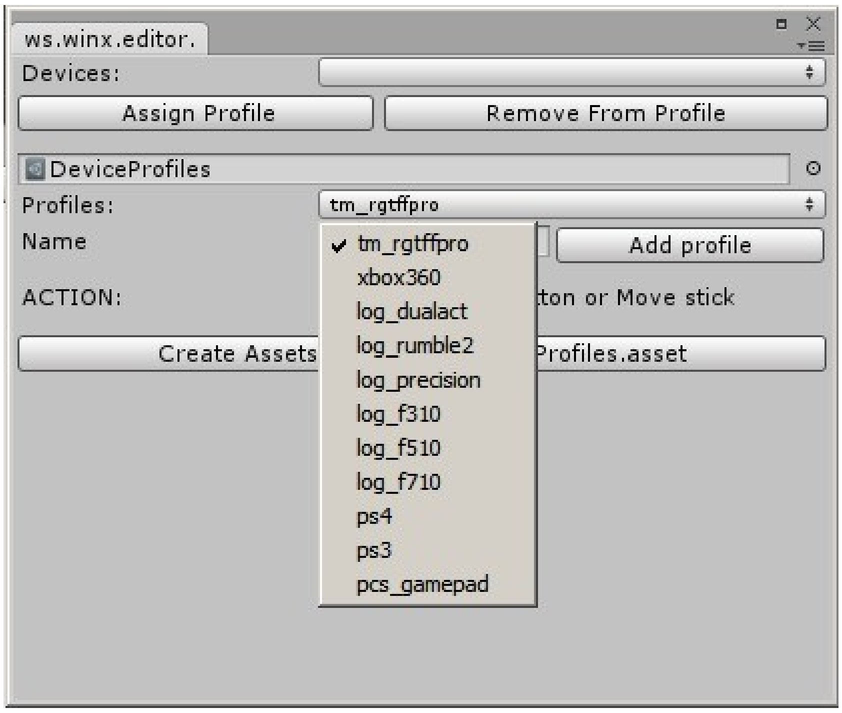


Player Input Component (Script)
Script: PlayerInputComponent
Player: Player 0
Events
Size: 1
Element 0
State: Wave
On UP ()
Runtime Only — PlayerInputComponent.onInformPlayerOfOtherWaveUp — Dude1 (PlayerIn
On DOWN ()
List is Empty
On HOLD ()
List is Empty

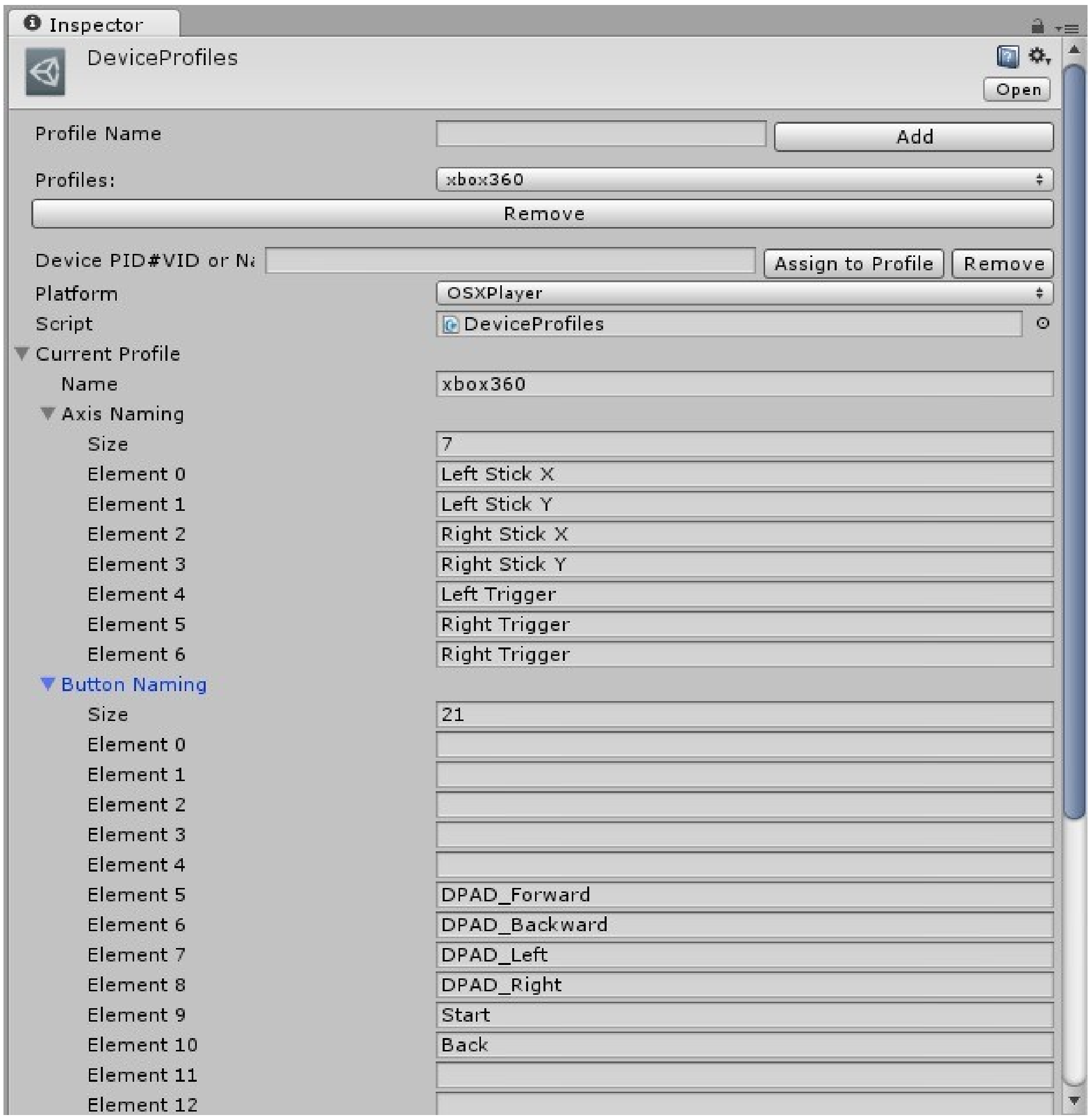
Advanced:

```
        InputManager.addEventListener ((int)States.Wave, Player).UP += onUp;
        InputManager.addEventListener ((int)States.Wave, Player).DOWN += onDown;




                public void onInformPlayerOfOtherWaveUp ()
                {
                                Debug.Log ("Inform "+Player+" of Player0 Wave UP");

                }

                void onHold ()
                {
                                Debug.Log (Player + ">Wave state trigger Hold");
                }

                void onUp ()
                {
                                Debug.Log (Player + ">Wave state trigger Up");
                }

                void onDown ()
                {
                                Debug.Log (Player + ">Wave state trigger Down");
                }

        // Update is called once per frame
        void Update ()
        {

        if (InputManager.GetInputDown ((int)States.Wave, Player, true)) {
                                                animator.Play ((int)States.Wave);
                                }

        if (InputManager.GetInputDown ((int)States.Jump, Player)) {
                                                animator.Play ((int)States.Jump);
                                }


        //
        //Math.Abs prevent code to function diffrently when axis is inverted
float forward = Math.Abs (InputManager.GetInput ((int)States.WalkForward, Player, 0.25f))
                                                - Math.Abs (InputManager.GetInput
((int)States.WalkBackward, Player, 0.25f));



        animator.SetFloat (forwardHash, forward);
                                //
                                //
float turn = Math.Abs (InputManager.GetInput ((int)States.TurnRight, Player, 0.25f))
                                                - Math.Abs (InputManager.GetInput
((int)States.TurnLeft, Player, 0.25f));

                                animator.SetFloat (turnHash, turn);
```

11) Create Device Profile by use of Window->InputMapper->Device Profiling



12) Click buttons or move axis to see the action and name the action. Data is stored in Resources>DataProfiles.asset for 85 devices.
13) Select DataProfiles.asset to add or edit profile's entries for designated OS. Add more devices in VID#PID (vendor ID product ID) or device name (UnityDriver only).


Inspector
DeviceProfiles
Open

Profile Name	
Profiles:	xbox360
Device PID#VID or Na	
Platform	OSXPlayer
Script	DeviceProfiles
Current Profile	
Name	xbox360
Axis Naming	
Size	7
Element 0	Left Stick X
Element 1	Left Stick Y
Element 2	Right Stick X
Element 3	Right Stick Y
Element 4	Left Trigger
Element 5	Right Trigger
Element 6	Right Trigger
Button Naming	
Size	21
Element 0	
Element 1	
Element 2	
Element 3	
Element 4	
Element 5	DPAD_Forward
Element 6	DPAD_Backward
Element 7	DPAD_Left
Element 8	DPAD_Right
Element 9	Start
Element 10	Back
Element 11	
Element 12	

14) Open PlayerVSPlayerDemo.unity and hit PLAY.
15) Write custom device driver on high level (see XinputDriver.cs,WiimoteDriver.cs,ThrustMasterDriver.cs in source) and add it to the manger.

```
InputManager.AddDriver(new XinputDriver());
```

16) Set UnityDriver as default.

```
InputManager.hidInterface.defaultDriver=new UnityDriver();
```

```
Warring:Unity doesn't make distinction between triggers/axis/poitOfView and doesn't make
controller distinction as multiply instances of same type have same name and can hard code
index of devices no matter position in GetJoystickNames list
```

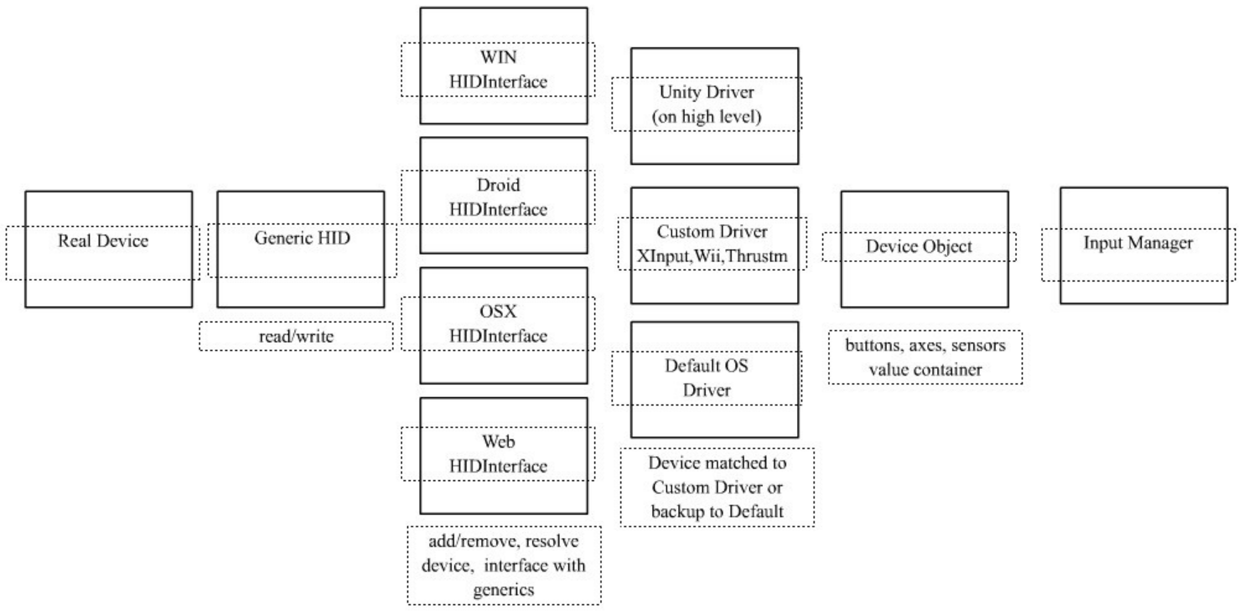
Still hard to understand? Go watch and read on bellow links.
https://www.youtube.com/watch?v=DAVy2xtkO6E

GITHUB: https://github.com/winalex/Unity3d-InputMapper

WiiDriver Demo
https://www.youtube.com/watch?v=tEwMcA2ZaMk

ThrustMasterDriver Demo (Win+Droid+Osx)
https://www.youtube.com/watch?v=uoH-RfopGzk

Web Player demo(thru Gamepad API):
https://www.youtube.com/watch?v=sXMWmn4Kc9c
MORE:
http://unity3de.blogspot.com/2014/06/unity-input-manger-pain-is-spreading.html


Real Device
Generic HID
read/write
WIN HIDInterface
Droid HIDInterface
OSX HIDInterface
Web HIDInterface
add/remove, resolve device, interface with generics
Unity Driver (on high level)
Custom Driver XInput,Wii,Thrustm
Default OS Driver
Device matched to Custom Driver or backup to Default
Device Object
buttons, axes, sensors value container
Input Manager

Feel free to contact me with your contribution, bugs reports, idea....
winxalex@gmail.com