

**Joypad**

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> Joypad		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 23, 2025	

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Joypad</b>	<b>1</b>
1.1	Joypad . . . . .	1
1.2	initjoypad . . . . .	1
1.3	joypadmovement . . . . .	1
1.4	joypadbuttons . . . . .	2
1.5	pressedrawkey . . . . .	3

# Chapter 1

## Joypad

### 1.1 Joypad

PureBasic – Joypad library

The PureBasic provide a full access to joypads plugged in the ports 0,1,2 or 3. It supports standard joysticks (1 buttons) and CD32 like joypads (7 buttons). The lowlevel.library is used to be OS friendly so be sure to include it in your final package as it's only part of Rom3.1. The lowlevel.library is provided in the extra drawer of the PureBasic package.

Commands summary:

```
InitJoypad
JoypadButtons
JoypadMovement
PressedRawKey
```

Example:

```
Joypad example
```

### 1.2 initjoypad

SYNTAX

```
Result.l = InitJoypad()
```

FUNCTION

Init all the joypad environment for later use. This command try to open the lowlevel.library, so if it fails it's probably this library which is not found. You must call this function before any other commands of this library.

### 1.3 joypadmovement

---

## SYNTAX

```
Movement.w = JoypadMovemnt (Port)
```

## FUNCTION

Return the actual joypad movement (on axis X and Y) plugged in the specified port. Here is the list for the returned value:

```
0: No activity
1: Up
2: Up-Right
3: Right
4: Down-Right
5: Down
6: Down-Left
7: Left
8: Up-Left
```

Note about the ports:

```
Port 0: On the Amiga. Used traditionnaly by the mouse
Port 1: On the Amiga. Near the first port, used by a joystick
Port 2: Need an external adaptater
Port 3: Need an external adaptater
```

## 1.4 joypadbuttons

## SYNTAX

```
PressedButtons.l = JoypadButtons (Port)
```

## FUNCTION

Returns a long typed value with a mask of currently pressed buttons. Any number of buttons can be pressed at the same time.

To know which button is actually pressed, you must use the following constants (self-explanatory) which are declared in the AmigaOS\_Small resident file (loaded automatically at the start of PureBasic):

```
#PB_JOYPAD_BUTTON1
#PB_JOYPAD_BUTTON2
#PB_JOYPAD_BUTTON3
#PB_JOYPAD_BUTTON4
#PB_JOYPAD_BUTTON5
#PB_JOYPAD_BUTTON6
#PB_JOYPAD_BUTTON7
```

And you have to perform some tests like:

```
If PressedButtons & #PB_JOYPAD_BUTTON1
    PrintN("Button 1 is pushed !")
    B1 = 1
EndIf

If PressedButtons & #PB_JOYPAD_BUTTON7
```

```
PrintN("Button 7 is pushed !")
B7 = 1
EndIf

If B1=1 And B7=1
PrintN("Buttons 1 & 7 are pushed at the same time !")
EndIf
```

## 1.5 pressedrawkey

### SYNTAX

Key = PressedRawKey()

### STATEMENT

Return the raw code of the key which is actually pushed or NULL if no key are pushed. Rawkey code are the same on any Amiga, indepedantly of the used keyboard. Rawkey are not ASCII values.