

**Joypad**

**COLLABORATORS**

	<i>TITLE :</i> Joypad		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		August 24, 2022	

**REVISION HISTORY**

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 . . . . .	2
1.4	joypadbuttons . . . . .	2
1.5	pressedrawkey . . . . .	3

---

## Chapter 1

# Joypad

### 1.1 Joypad

PureBasic - Joypad library

Cette bibliothèque vous permet de contrôler les joysticks/joypads branchés sur les ports 0, 1, 2 ou 3 de l'Amiga. Jusqu'à 7 boutons sont supportés (joypad CD32). La bibliothèque Amiga 'lowlevel.library' est utilisée pour permettre une compatibilité totale avec le système. Vous devez donc vous assurer que l'utilisateur de votre programme possède bien cette bibliothèque. Il est conseillé de l'inclure avec la distribution de votre programme (elle se trouve dans le répertoire PureBasic/Extras).

Commandes disponibles:

InitJoypad  
JoypadButtons  
JoypadMovement  
PressedRawKey  
Exemple:

Joypad example

### 1.2 initjoypad

SYNTAX

```
Resultat.l = InitJoypad()
```

FUNCTION

Initialise l'environnement nécessaire à la future gestion des joysticks. Si le 'Resultat' est NULL, alors la bibliothèque 'lowlevel.library' n'est pas disponible. Vous devez appeler cette commande avant d'utiliser

une commande de cette bibliothèque.

### 1.3 joypadmovement

#### SYNTAX

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

#### FUNCTION

Revoie le mouvement actuel du joystick (sur l'axe X et Y) qui est branché dans le 'Port' spécifié. Voici la liste des valeurs possibles:

```
0: Aucun mouvement
1: Haut
2: Diagonale Haut-Droite
3: Droite
4: Diagonale Bas-Droite
5: Bas
6: Diagonale Bas-Gauche
7: Gauche
8: Diagonale Haut-Gauche
```

Note à propos des ports:

```
Port 0: Situé directement sur l'Amiga (port le plus à droite).
        Il est en principe utilisé par la souris.
Port 1: Situé directement sur l'Amiga (à coté du port 0)
Port 2: Nécessite un adaptateur externe spécial
Port 3: Nécessite un adaptateur externe spécial
```

### 1.4 joypadbuttons

#### SYNTAX

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

#### FUNCTION

Renvoie une valeur décrivant quels sont les boutons actuellement appuyés sur un 'Port' donné. Tous les boutons peuvent être appuyés en même temps si nécessaire.

Pour savoir quels boutons sont appuyés, il faut utiliser les constantes suivantes, déjà déclarée dans le fichier AmigaOS\_Small:

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

Ensuite il suffit de faire une opération 'AND' (&) sur la valeur 'PressedButtons' et voir si le résultat est nul ou non. S'il n'est pas nul, le bouton est appuyé:

```
If PressedButtons & #PB_JOYPAD_BUTTON1
  PrintN("Le bouton 1 est appuyé !")
  B1 = 1
EndIf

If PressedButtons & #PB_JOYPAD_BUTTON7
  PrintN("Le bouton 7 est appuyé !")
  B7 = 1
EndIf

If B1=1 And B7=1
  PrintN("Les boutons 1 et 7 sont appuyés en même temps !")
EndIf
```

## 1.5 pressedrawkey

### SYNTAX

```
Key = PressedRawKey()
```

### STATEMENT

Renvoie le code brut (RawKey) de la touche actuellement appuyée. Ce code est fonction de la position de la touche sur le clavier, pas de sa lettre (par exemple le RawKey de la touche 'A' d'un clavier AZERTY aura la même valeur que la touche 'Q' d'un clavier QWERTY). Une valeur RawKey n'est pas une valeur ASCII.

C'est très utile pour les touches 'ESC', 'ENTER',...