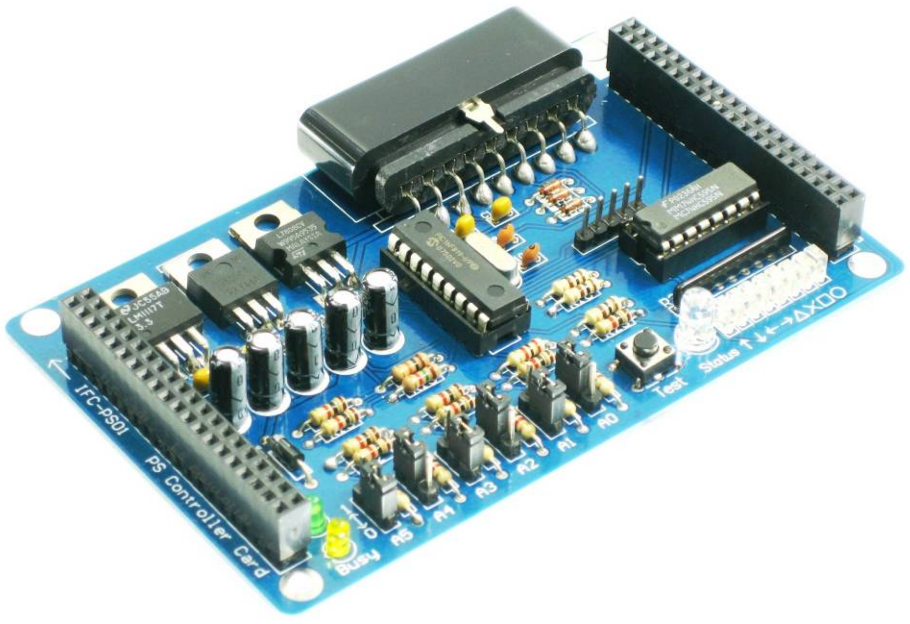
IFC-PS01 Interface Free Controller Play Station 2 Card



Card Library Functions

V1.0

Oct 2008

Function Prototype for Play Station 2 card (PS01)

This document explains the function prototype for IFC-PS01. IFC-PS01 can be stacked on IFC system as one of input card. The function prototype should be called in main program of MB00 in order to control/communicate with PS01. User can also find the explanation of function prototype in its header file, "iic_ps.h". Table 1 shows the function for PS01.

Function Prototype	Remarks	Parameter Description

unsigned char ps_sw(unsigned char **add**, unsigned char **button_number**)	ps_sw(**add_ps1**,**1**)	**add** = card address
		button_number = define which button to read.
		
		button_number = **0** (p_select)
		button_number = **1** (p_joyl)
		button_number = **2** (p_joyr)
		button_number = **3** (p_start)
		button_number = **4** (p_up)
		button_number = **5** (p_right)
		button_number = **6** (p_down)
		button_number = **7** (p_left)
		button_number = **8** (p_l2)
		button_number = **9** (p_r2)
		button_number = **10** (p_l1)
		button_number = **11** (p_r1)
		button_number = **12** (p_triangle)
		button_number = **13** (p_circle)
		button_number = **14** (p_cross)
		button_number = **15** (p_square)

unsigned char ps_joy(unsigned char **add**, unsigned char **joy_stick**)	ps_joy(**add_ps1**, **2**)	**add** = card address
		joy_stick = define which joy stick axis to read.
		
		type 1
		joy_stick = **0**(joy_lx)
		joy_stick = **1**(joy_ly)
		joy_stick = **2**(joy_rx)
		joy_stick = **3**(joy_ry)
		
		type 2
		joy_stick = **4**(joy_lu)
		joy_stick = **5**(joy_ld)
		joy_stick = **6**(joy_ll)
		joy_stick = **7**(joy_lr)
		joy_stick = **8**(joy_ru)
		joy_stick = **9**(joy_rd)
		joy_stick **= 10**(joy_rl)
		joy_stick = **11**(joy_rr)

void ps_vibrate(unsigned char **add**, unsigned char **motor**, unsigned char **value**)	ps_vibrate(**add_ps1**,**1**,**1**)	**add** = card address **motor** = which motor to control (**1** or **2**) **value** = motor value If **motor** = **1**, **value** is either **1 or 0**. If **motor** = **2**, **value** is **0-255.**
unsigned char ps_stat(unsigned char **add**)	ps_stat(**add_ps1**)	**add** = card address

Table 1 Function Prototype for Play Station (PS01)

IFC-PS01 is compatible on IFC series. It can be stacked on IFC system as one of input card. IFC-PS01 offers simple library function which returns the state of button status on Joy-stick. This section will give examples on the method to call this function in main program. **add** is the address for IFC-PS01 card. The address set on IFC-PS01 card must compatible with address define in source code.

unsigned char ps_sw(unsigned char **add**, unsigned char **button_number**)

This function is used for user to read digital value of PS2 Controller button. They are 16 **button_number** can read on PS2 controller such as triangle, square, up, bottom, left, right and many more. Every button on PS2 controller is defined as a specific number and specific name. User can choose either to use number or name for **button_number**. User can refer table 1 above to see button number. Below is an example to call this function in main program. LCD will display "circle" when p_circle on PS2 is pressed. This function will return 0 for pressed button and 1 if button is not pressed.

```
if(ps_sw(ps_add,13)==0)cp1_str("circle");
```

or

```
if(ps_sw(ps_add,p_circle)==0)cp1_str("circle");
```

unsigned char ps_joy(unsigned char **add**, unsigned char **joy_stick**)

The above function is used for user to read analog joystick value of PS2 controller. **joy_stick** is to define which joystick axis to read. They are 2 analog joy stick button on PS2 controller (left and right). Each joystick is read in axis. Each joystick axis is defined as a specific number and specific name. User is free to use either name or number for joy_stick. User can refer table 1 above to see the axis. From table 1, joystick axis is divided into 2 types. User can use either type 1 or type 2 to read the joystick. User can refer IFC-PS01 User's Manual for more detail on joystick axes. Below is an example of source code to call this function prototype. From the example below, CP1 will display return value of joystick axis when joy_rx is pushed to any axis.

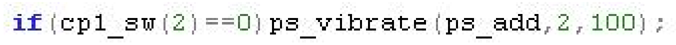
```
cp1_dec(ps_joy(ps_add,2),3);
```

or

```
cp1_dec(ps_joy(ps_add,joy_rx),3);
```

```
void ps_vibrate(unsigned char add, unsigned char motor, unsigned char value)
```

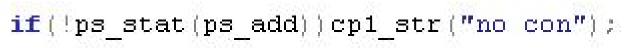
The above function is used for user to control the vibrator motor of the PS2 controller. **motor** is uses to define which motor to control the vibration. They are 2 **motor** on most PS2 controller which are motor 1 and motor 2. **value** is the status or speed value for selected motor. For motor 1, the value is either 1(ON) or 0(OFF). But for motor 2, user needs to set a speed **value**. The value is at range 0-255. Higher of speed value will result in greater vibration. Below is a sample code for user to call this function.



```
if(cp1_sw(2)==0)ps_vibrate(ps_add,2,100);
```

```
unsigned char ps_stat (unsigned char add)
```

The above function is used to obtain connection status of the PS2 controller with IFC-PS01 card. Example below shows calling of this function in main program.



```
if(!ps_stat(ps_add))cp1_str("no con");
```

From the sample code above, CP1 will display “no con” if no PS2 controller connect to IFC-PS01 card. This function will return 1 if PS2 controller is connected and return 0 if no PS2 controller is detected.

Note: User is reminded to add header file (iic.h and iic_ps.h) and object file (iic.o and iic_ps.o) for IFC-MB00 and IFC-PS01 each time open a new project for IFC. User also need to include card h file at the beginning of the program. Please refer sample source cod e for the example to include card h file.

Prepared by
Cytron Technologies Sdn. Bhd.
19, Jalan Kebudayaan 1A,
Taman Universiti,
81300 Skudai,
Johor, Malaysia.

Tel: *+607-521 3178*
Fax: *+607-521 1861*

URL: *www.cytron.com.my*
Email: *support@cytron.com.my*
sales@cytron.com.my