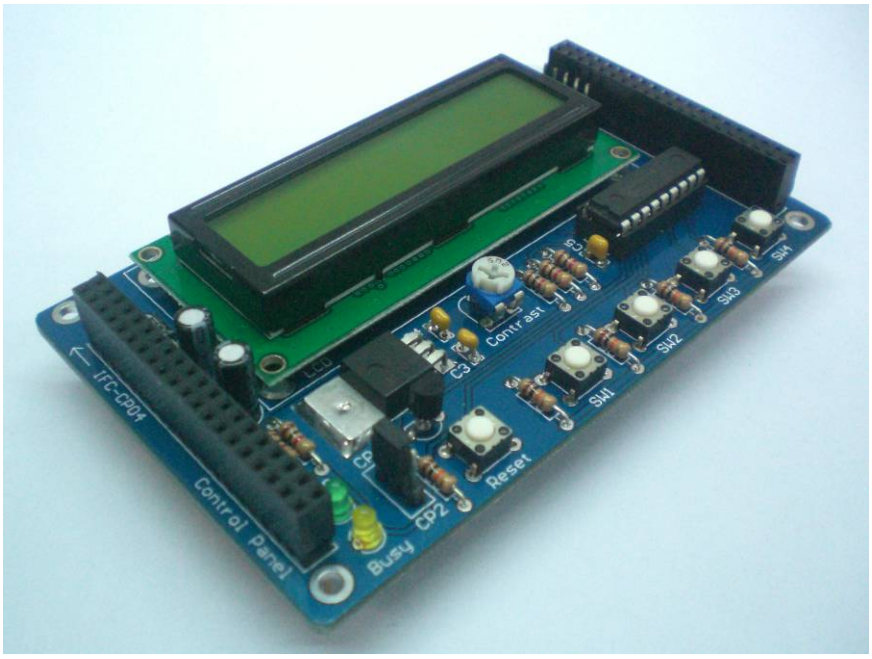
# IFC-CP04
# Interface Free Controller
# Control Panel




# Card Library Functions for Visual C# Express and Visual Basic Express

## V1.0

## Apr 2009

## Function Prototype for Control Panel (CP04)

This document explains the function prototype for controlling IFC-CP04 using PC through IFC-CI00. User may also use 'object browser' under Microsoft Visual C# to view the summary, parameter and return value description of IFC-CP04 function prototype. User need to add reference 'ifc_ci.dll' and 'ifc_cp.dll' for IFC-CI00 and IFC-CP04 card in order to control/communicate IFC-CP04 using PC. Please note that before user start the programming, user need to initialize the 'ifc.ifc_ci' and 'ifc.ifc_cp' in order to use the functions to control IFC-CP04. Example of creating a 'ifc.ifc_ci' class called 'ifc1' and 'ifc.ifc_cp' class called cp1:

```
static ifc.ifc_ci ifc1 = new ifc.ifc_ci(74);
ifc.ifc_cp cp1 = new ifc.ifc_cp(ifc1, 1);
ifc.ifc_cp cp2 = new ifc.ifc_cp(ifc1, 2);
```

For 'ifc.ifc_ci' class, user need to specified the COM Port that is connected to IFC-CI00 and for 'ifc.ifc_cp' class, user need to specified the IFC-CI00 in use and also the address for IFC-CP04. Please make sure that the address must be unique and different with other IFC card in the IFC system.

| Function Prototype | Examples | Summary | Parameter Description | Returns Value |
|---|---|---|---|---|
| **ifc_cp**(ifc.ifc_ci *ifc_ci*, byte *cp*) | **ifc.ifc_cp**(ifc1, 1)<br>**ifc.ifc_cp**(ifc1, 2) | Initializes a new instance of the ifc.ifc_cp class using the specified ifc.ifc_ci and address for IFC-CP04. | *ifc_ci :* ifc.ifc_ci in use.<br>*cp*: 1 for CP1 and 2 for CP2. (byte) | **ifc_cp**(ifc.ifc_ci *ifc_ci*, byte *cp*) |
| **ifc_cp**(ifc.ifc_ci *ifc_ci*, int *cp*) | **ifc.ifc_cp**(ifc1, 2) | Initializes a new instance of the ifc.ifc_cp class using the specified ifc.ifc_ci and address for IFC-CP04. | *ifc_ci*: ifc.ifc_ci in use.<br>*cp*: 1 for CP1 and 2 for CP2. (int) | **ifc_cp**(ifc.ifc_ci *ifc_ci*, int *cp*) |
| **cp_all_sw**() | **cp1.cp_all_sw**()<br>**cp2.cp_all_sw**() | To read all push buttons on IFC-CP04. | | Return all push button status in one byte. Bit 0 represent SW1, bit 1 represent SW2, bit 2 represent SW3, and bit 3 represent SW4. Representative bit clear the push button is pressed. (byte) |

| | | | | |
|---|---|---|---|---|
| **void cp_bin**(byte *data*, byte *num_dig*) | **cp1.cp_bin**(10111101, 8)<br>**cp2.cp_bin**(10111101, 8) | To display binary number on LCD. | *data*: Binary number to be display on LCD. (byte)<br>*num_dig*: Number of bit to display in range of 1 to 16. (byte) | |
| **void cp_bin**(int *data*, int *num_dig*) | **cp1.cp_bin**(11110101, 12)<br>**cp2.cp_bin**(11110101, 12) | To display binary number on LCD. | *data*: Binary number to be display on LCD. (int)<br>*num_dig*: Number of bit to display in range of 1 to 16. (int) | |
| **void cp_blight**(byte *brightness*) | **cp1.cp_blight**(255)<br>**cp2.cp_blight**(255) | To set the brightness of the LCD backlight. | *brightness*: Brightness of the LCD in range of 0 to 255. (byte) | |
| **void cp_blight**(int *brightness*) | **cp1.cp_blight**(180)<br>**cp2.cp_blight**(180) | To set the brightness of the LCD backlight. | *brightness*: Brightness of the LCD in range of 0 to 255. (int) | |
| **void cp_char**(byte *data*) | **cp1.cp_char**('A')<br>**cp2.cp_char**('A') | To display a character on LCD. | *data*: Character to be display on LCD, in ASCII format. (byte) | |
| **void cp_char**(char *data*) | **cp1.cp_char**('@')<br>**cp2.cp_char**('@') | To display a character on LCD. | *data*: Character to be display on LCD, in ASCII format. (char) | |
| **void cp_clr**() | **cp1.cp_clr**()<br>**cp2.cp_clr**() | To clear the LCD and set the cursor back to position 0,0. | | |
| **void cp_dec**(byte *data*, byte *num_dig*) | **cp1.cp_dec**(12,2)<br>**cp2.cp_dec**(12,2) | To display decimal number on LCD. | *data*: Decimal number to be display on LCD. (byte)<br>*num_dig*: Number of digit to display in range of 1 to 16. (byte) | |

| | | | | |
|---|---|---|---|---|
| **void cp_dec**(int *data*, int *num_dig*) | **cp1.cp_dec**(1234,10)<br>**cp2.cp_dec**(1234,10) | To display decimal number on LCD. | *data*: Decimal number to be display on LCD. (int)<br>*num_dig*: Number of digit to display in range of 1 to 16. (int) | |
| **void cp_dec**(long *data*, long *num_dig*) | **cp1.cp_dec**(1234567890,10)<br>**cp2.cp_dec**(1234567890,10) | To display decimal number on LCD. | *data*: Decimal number to be display on LCD. (long)<br>*num_dig*: Number of digit to display in range of 1 to 16. (long) | |
| **void cp_goto**(byte *row*, byte *col*) | **cp1.cp_goto**(0, 0)<br>**cp2.cp_goto**(0, 0) | To set the cursor of the LCD to a specific location. | *row*: Selected Row of the cursor location, in range of 0 to 1. (byte)<br>*col*: Selected Column of the cursor location, in range of 0 to 15. (byte) | |
| **void cp_goto**(int *row*, int *col*) | **cp1.cp_goto**(1, 0)<br>**cp2.cp_goto**(1, 0) | To set the cursor of the LCD to a specific location. | *row*: Selected Row of the cursor location, in range of 0 to 1. (int)<br>*col*: Selected Column of the cursor location, in range of 0 to 15. (int) | |
| **void cp_str**(string *data*) | **cp1.cp_str**( "Cytron")<br>**cp2.cp_str**( "Cytron") | To display a string on LCD. | *data*: String to be display on LCD. (string) | |
| **cp_sw**(byte *button_number*) | **cp1.cp_sw**(1)<br>**cp2.cp_sw**(1) | To read push button on IFC-CP04. | *button_number*: Button number to read, in range of 1 to 4. (byte) | True if the button is being pressed, false otherwise. (bool) |
| **cp_sw**(int *button_number*) | **cp1.cp_sw**(4)<br>**cp1.cp_sw**(4) | To read push button on IFC-CP04. | *button_number*: Button number to read, in range of 1 to 4. (int) | True if the button is being pressed, false otherwise. (bool) |

**Table 1 Function Prototype for CP04 card**

*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*