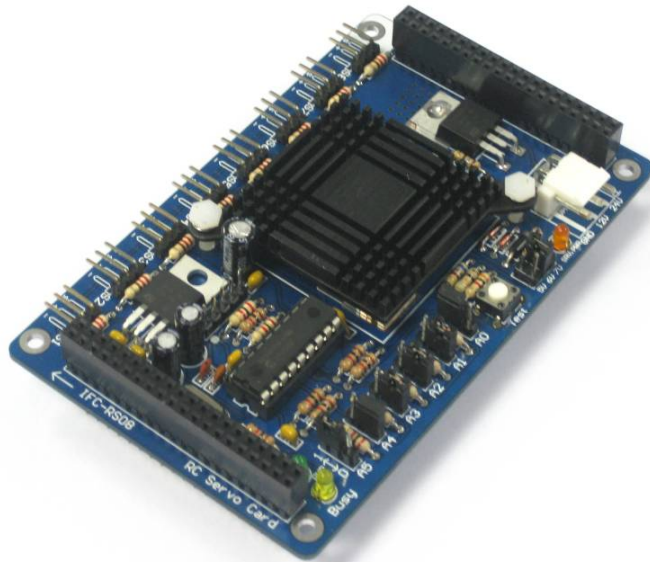




# **IFC-RS08 Interface Free Controller RC Servo Card**



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

**V1.0**

**Apr 2009**

Information contained in this publication regarding device applications and the like is intended through suggestion only and may be superseded by updates. It is your responsibility to ensure that your application meets with your specifications. No representation or warranty is given and no liability is assumed by Cytron Technologies Incorporated with respect to the accuracy or use of such information, or infringement of patents or other intellectual property rights arising from such use or otherwise. Use of Cytron Technologies's products as critical components in life support systems is not authorized except with express written approval by Cytron Technologies. No licenses are conveyed, implicitly or otherwise, under any intellectual property rights.

## Function Prototype for RC Servo card (RS08)

This document explains the function prototype for controlling IFC-RS08 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-RS08 function prototype. User need to add reference ‘ifc\_ci.dll’ and ‘ifc\_rs.dll’ for IFC-CI00 and IFC-RS08 card in order to control/communicate IFC-RS08 using PC. Please note that before user start the programming, user need to initialize the ‘ifc.ifc\_ci’ and ‘ifc.ifc\_rs’ in order to use the functions to control IFC-RS08. Example of creating a ‘ifc.ifc\_ci’ class called ‘ifc1’ and ‘ifc.ifc\_rs’ class called rs1:

```
static ifc.ifc_ci ifc1 = new ifc.ifc_ci(74);
ifc.ifc_rs rs1 = new ifc.ifc_rs(ifc1, 7);
```

For ‘ifc.ifc\_ci’ class, user need to specified the COM Port that is connected to IFC-CI00 and for ‘ifc.ifc\_rs’ class, user need to specified the IFC-CI00 in use and also the address for IFC-RS08. Please make sure that the address must be unique and different with other IFC card in the IFC system.

Function Prototype	Example	Summary	Parameter Description	Return Value
<b>ifc_rs</b> ( <a href="#">ifc.ifc_ci</a> ifc_ci, <a href="#">int</a> address)	<b>ifc.ifc_rs</b> ( <a href="#">ifc1</a> , <a href="#">6</a> )	Initializes a new instance of the ifc.ifc_rs class using the specified ifc.ifc_ci and address for IFC-RS08.	<i>ifc_ci</i> : ifc.ifc_ci in use. <i>address</i> : Address for IFC-PS01, in range of 0 to 63. (int)	
<b>ifc_rs</b> ( <a href="#">ifc.ifc_ci</a> ifc_ci, <a href="#">byte</a> address)	<b>ifc.ifc_rs</b> ( <a href="#">ifc1</a> , <a href="#">7</a> )	Initializes a new instance of the ifc.ifc_rs class using the specified ifc.ifc_ci and address for IFC-RS08.	<i>ifc_ci</i> : ifc.ifc_ci in use. <i>address</i> : Address for IFC-PS01, in range of 0 to 63. (byte)	
<b>void rs_en</b> ( <a href="#">int</a> servo_channel, <a href="#">bool</a> enable)	<b>rs1.rs_en</b> ( <a href="#">0</a> , <a href="#">true</a> )	To enable or disable RC servo(s).	<i>servo_channel</i> : RC servo channel in range of 0 to 8. 0 to select all, 1 to select S1, 2 to select S2... and 8 to select S8. (int) <i>enable</i> : True to enable the selected RC servo(s) and false to disable selected RC servo(s). (bool)	

<b>void rs_en</b> ( <a href="#">byte</a> <i>servo_channel</i> , <a href="#">bool</a> <i>enable</i> )	<b>rs1.rs_en</b> ( <a href="#">5</a> , <a href="#">true</a> )	To enable or disable RC servo(s).	<i>servo_channel</i> : RC servo channel in range of 0 to 8. 0 to select all, 1 to select S1, 2 to select S2... and 8 to select S8. (byte) <i>enable</i> : True to enable the selected RC servo(s) and false to disable selected RC servo(s). (bool)	
<b>void rs_pos_sp</b> ( <a href="#">int</a> <i>servo_channel</i> , <a href="#">int</a> <i>pos</i> , <a href="#">int</a> <i>ramp</i> )	<b>rs1.rs_pos_sp</b> ( <a href="#">0</a> , <a href="#">2000</a> , <a href="#">150</a> )	To determine position and the speed of selected RC servo.	<i>servo_channel</i> : RC servo channel in range of 0 to 8. 0 to select all, 1 to select S1, 2 to select S2... and 8 to select S8. (int) <i>pos</i> : RC servo position in range of 0 to 5000, which represent 0.5ms to 2.5ms. (int) <i>ramp</i> : Speed of RC servo in range of 1 to 255. 1 is the slowest speed and 255 is the fastest speed. 0 is to select the RC servo default maximum speed. (int)	
<b>rs_read_pos</b> ( <a href="#">int</a> <i>request_servo_channel</i> )	<b>rs1.rs_read_pos</b> ( <a href="#">0</a> )	To read the current position of selected RC servo.	<i>servo_channel</i> : RC servo channel in range of 1 to 8. 1 to select S1, 2 to select S2... and 8 to select S8. (int)	Return the current value of selected RC servo. (int)
<b>rs_read_pos</b> ( <a href="#">byte</a> <i>request_servo_channel</i> )	<b>rs1.rs_read_pos</b> ( <a href="#">8</a> )	To read the current position of selected RC servo.	<i>servo_channel</i> : RC servo channel in range of 1 to 8. 1 to select S1, 2 to select S2... and 8 to select S8. (byte)	Return the current value of selected RC servo. (int)

**Table 1      Function Prototype for RC Servo Card (RS08)**

*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](http://www.cytron.com.my)

*Email:* [support@cytron.com.my](mailto:support@cytron.com.my)  
[sales@cytron.com.my](mailto:sales@cytron.com.my)