



Manufacturer: Control Concepts, Inc.  
Model: Websocket Server v1.0  
Device Type: Utility

**CONTACT SUPPORT:**

COMPANY NAME:	Control Concepts, Inc.
SUPPORT CONTACT:	Lynn Abraham
EMAIL ADDRESS:	support@controlconcepts.net
PHONE:	201-797-7900
ADDRESS:	16-00 Route 208, Fair Lawn, NJ 07410
NOTES:	Programmer: JJM
HAND-IN DATE:	11/31/2013

**GENERAL INFORMATION**

SIMPLWINDOWS NAME:	CCI Websocket Server v1.0
CATEGORY:	Utility
VERSION:	1.0
SUMMARY:	<p>This is a Utility module that implements a RFC-6455 WebSocket server capable of having a single connection per instance. The WebSocket server allows for two way communication to HTML5 compatible web pages. This WebSocket server utilizes a sub-protocol to allow for input/ouput control into and out of your Crestron 3 Series controller. The sub-protocol allows up to 500 digital, 100 analog, and 100 string input/output signals. A simple HTML5 webpage has been provided as an example (Websocket Webpage Example.html).</p> <p>This WebSocket server is compatible with the following web browsers.</p> <ul style="list-style-type: none"><li>Internet Explorer 10</li><li>Firefox 11 (PC)</li><li>Firefox 11 (Android)</li><li>Chrome 16 (PC, Mobile)</li><li>Safari 6 (Mac, iOS*)</li><li>Opera 12.10 (PC, Mobile)</li></ul>



Manufacturer: Control Concepts, Inc.  
 Model: Websocket Server v1.0  
 Device Type: Utility

	Android Browser (Not Supported)
	*iOS must be at least version 6.
<b>GENERAL NOTES:</b>	This module utilizes SIMPL# and is only compatible with 3-Series controllers
<b>CRESTRON HARDWARE REQUIRED:</b>	3-Series controller with Ethernet connectivity
<b>SETUP OF CRESTRON HARDWARE:</b>	Ethernet Connectivity
<b>VENDOR FIRMWARE:</b>	n/a
<b>VENDOR SETUP:</b>	n/a
<b>CABLE DIAGRAM:</b>	TCP/IP

**CONTROL:**

<u>Signal/Function Name</u>	<u>D,S,A</u>	<u>Digital, Serial, Analog signal property definition.</u>
fb*	D	<p>Digital Feedback Signals 1-500.</p> <p>When a signal goes <b>high</b>, it causes a sub-protocol message of <i>ON[&lt;signal_ID&gt;]</i> to be sent into the HTML5 websocket.onmessage event handler of your webpage.</p> <p>When a signal goes <b>low</b>, it causes a sub-protocol message of <i>OFF[&lt;signal_ID&gt;]</i> to be sent into the HTML5 websocket.onmessage event handler of your webpage.</p> <p><i>Examples (fb201):</i></p> <p>ON[201]</p> <p>OFF[201]</p>
an_fb*	A	<p>Analog Feedback Signals 1-100.</p> <p>When a signal <b>changes</b>, it causes a sub-protocol message of <i>LEVEL[&lt;signal_id&gt;,&lt;value&gt;]</i> to be sent into the HTML5 websocket.onmessage event handler of your webpage</p> <p><i>Example (an_fb50):</i></p> <p>LEVEL[50,100]</p> <p>LEVEL[50,65535]</p>
text-o*	S	String Output Signals 1-100



Manufacturer: Control Concepts, Inc.  
 Model: WebSocket Server v1.0  
 Device Type: Utility

		<p>When a signal <b>changes</b>, it causes a sub-protocol message of <code>STRING[&lt;signal_id&gt;,&lt;value&gt;]</code> to be sent into the HTML5 websocket.onmessage event handler of your webpage</p> <p><i>Example (text-o20):</i>  <code>STRING[20,Hello World!]</code>  <code>STRING[20,SIMPL# Rocks!]</code></p>
<b>Start_Server</b>	D	Pressing this signal will cause the WebSocket server to start listening for a client connection.
<b>Stop_Server</b>	D	Pressing this signal will disconnect any WebSocket clients and stop the server from listening for any additional clients.

**FEEDBACK:**

<b>press*</b>	D	<p>This digital signal will go <b>high</b> when a sub-protocol message of <code>PUSH[&lt;signal_id&gt;]</code> is sent out from your webpage utilizing the <code>websocket.send()</code> method.</p> <p>This signal will go <b>low</b> when a sub-protocol message of <code>RELEASE[&lt;signal_id&gt;]</code> is sent out from your webpage utilizing the <code>websocket.send()</code> method.</p> <p><i>Example (press101):</i>  <code>PUSH[101]</code>  <code>RELEASE[101]</code></p>
<b>an_act*</b>	A	<p>This analog signal will <b>change</b> when a sub-protocol message of <code>LEVEL[&lt;signal_id&gt;,&lt;value&gt;]</code> is sent out from your webpage utilizing the <code>websocket.send()</code> method.</p> <p><i>Example (an_act51):</i>  <code>LEVEL[51,255]</code>  <code>LEVEL[51,100]</code></p>
<b>text-i*</b>	S	<p>This string signal will <b>change</b> when a sub-protocol message of <code>STRING[&lt;signal_id&gt;,&lt;data&gt;]</code> is sent out from your webpage utilizing the <code>websocket.send()</code> method.</p> <p><i>Example (text-i32):</i>  <code>STRING[32,Testing 123]</code>  <code>STRING[32,PIN-1234]</code></p>



Manufacturer: Control Concepts, Inc.  
 Model: Websocket Server v1.0  
 Device Type: Utility

Server_Is_Running	D	Indicates high when the server is running.
Client_Is_Connected	D	Indicates high when a client is connected and successfully authenticated using RFC-6455 rules.

### PARAMETERS:

IP Port	A	IP Port the WebSocket will listen on. <i>(Must be an available port on the 3-Series controller.)</i>
---------	---	--

### TESTING:

OPS USED FOR TESTING:	CP3 1.007.0019
SIMPL WINDOWS USED FOR TESTING:	4.02.20
DEVICE DB USED FOR TESTING:	52.00.007.00
CRES DB USED FOR TESTING:	41.05.005.00
SYMBOL LIBRARY USED FOR TESTING:	875
SAMPLE PROGRAM:	CCI Websocket Server v1.0 Demo
REVISION HISTORY:	v1.0 Initial Release