Introduction

Thank you for purchasing the WE Control API (Model 707741). This User's Manual contains information about the ASCII commands that are used when the WE Control API is used with the WE7512 PSI Module. For information about the installation, program model, functions, and other information pertaining to the WE Control API, see the “WE Control API User’s Manual” (IM 707741-61E).

Notes

- The contents of this manual describe the WE Control API Ver. 3.0.4.0. If you are using another version of the API, the information given in this manual may differ from that of API that you are using.

- The contents of this manual are subject to change without prior notice as a result of continuing improvements to the instrument's performance and functions.

- Every effort has been made in the preparation of this manual to ensure the accuracy of its contents. However, should you have any questions or find any errors, please contact your nearest YOKOGAWA dealer as listed on the back cover of this manual.

- Copying or reproducing any or all of the contents of this manual without YOKOGAWA's permission is strictly prohibited.

Trademarks

- Windows is a trademark of Microsoft Corporation.
- Adobe and Acrobat are trademarks of Adobe Systems Incorporated.
- IBM is a registered trademark of International Business Machines Corporation.
- Other product names are trademarks or registered trademarks of their respective holders.

Revisions

1st Edition: January 1999
## ASCII Commands

<table>
<thead>
<tr>
<th>ASCII Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CH&lt;x&gt;:Input:Range</td>
<td>Sets/ Gets the analog input range.</td>
</tr>
<tr>
<td>Input:Digital(8bit)</td>
<td>Gets the 8-bit digital input value.</td>
</tr>
<tr>
<td>Input:Digital(8bit):DI Condition Window</td>
<td>Sets/ Gets the input time of the 8-bit digital signal.</td>
</tr>
<tr>
<td>CH&lt;x&gt;:Output:Range</td>
<td>Sets/ Gets the analog output range.</td>
</tr>
<tr>
<td>CH&lt;x&gt;:Output</td>
<td>Sets/ Gets the analog output value.</td>
</tr>
<tr>
<td>CH&lt;x&gt;:Output:Output:Output</td>
<td>Turns ON/OFF the analog output/ Gets the ON/OFF state.</td>
</tr>
<tr>
<td>Output:Digital(12bit):Output</td>
<td>Turns ON/OFF the 12-bit digital output/ Gets the ON/OFF state.</td>
</tr>
<tr>
<td>Output:Digital(8bit):Output</td>
<td>Turns ON/OFF the 8-bit digital output/ Gets the ON/OFF state.</td>
</tr>
<tr>
<td>Output:Digital(12bit)</td>
<td>Sets/ Gets the 12-bit digital output value.</td>
</tr>
<tr>
<td>Output:Digital(8bit)</td>
<td>Sets/ Gets the 8-bit digital output value.</td>
</tr>
</tbody>
</table>

### CH<x>:Input:Range

**Description**
Sets/ Gets the analog input range.

**Parameter**
10 V | 5 V | 0-10 V

**Example (VisualBasic)**
```vbnet
err = WeSetControl (hMo, "CH1:Input:Range", "10 V")
' Set the input range of analog CH1 to ±10 V.
Dim value As Variant
e
err = WeGetControl (hMo, "CH1:Input:Range", value)
value ➔ 10.0E+00
```

### Input:Digital(8bit)

**Description**
Gets the 8-bit digital input value.

**Parameter**
0 to 255

**Example (VisualBasic)**
```vbnet
err = WeGetControl (hMo, "Input:Digital(8bit)", value)
value ➔ 0
```
Input:Digital(8bit):DI Condition Window
Description
Sets/Gets the input time of the 8-bit digital signal.

Parameter
100 μ to 1 s

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "Input:Digital(8bit):DIConditionWindow", 100e-6)
' Set the input time to 100 ms.
Dim value As Variant
err = WeGetControl (hMo, "Input:Digital(8bit):DIConditionWindow", value)
value → 100.0E-06
```

CH<x>:Output:Range
Description
Sets/Gets the analog output range.

Parameter
10 V | 5 V | 0-10 V

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "CH1:Output:Range", "10 V")
' Sets the output range of analog CH1 to ±10 V.
Dim value As Variant
err = WeGetControl (hMo, "CH1:Output:Range", value)
value → 10.0E+00
```

CH<x>:Output
Description
Sets/Gets the analog output value.

Parameter
When the range is ±10 V: –10.000 to 9.995 V
When the range is ±5 V: –5.000 to 4.998 V
When the range is 0 to 10 V: 0 to 9.998 V

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "CH1:Output", "1.24")
' Set the analog output value to 1.24 V.
Dim value As Variant
err = WeGetControl (hMo, "CH1:Output", value)
value → 1.240E+00
```

CH<x>:Output:Output
Description
Turns ON/OFF the analog output/Gets the ON/OFF state.

Parameter
Off | On

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "CH1:Output:Output", "On")
' Turn ON the analog output of CH1.
Dim value As Variant
err = WeGetControl (hMo, "CH1:Output:Output", value)
value → On
```
Output:Digital(12bit):Output

Description
Turns ON/OFF the 12-bit digital output/Gets the ON/OFF state.

Parameter
Off | On

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "Digital(12bit):Output", "On")
' Turn ON the 12-bit digital output.
Dim value As Variant
err = WeGetControl (hMo, "Output:Digital(12bit):Output", value)
value → On
```

Output:Digital(8bit):Output

Description
Turns ON/OFF the 8-bit digital output/Gets the ON/OFF state.

Parameter
Off | On

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "Output:Digital(8bit):Output", "On")
' Turn ON the 8-bit digital output.
Dim value As Variant
err = WeGetControl (hMo, "Output:Digital(8bit):Output", value)
value → On
```

Output:Digital(12bit)

Description
Sets/Gets the 12-bit digital output value.

Parameter
0 to 4095

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "Output:Digital(12bit)", "4095")
' Set the 12-bit digital output to 0xfff.
Dim value As Variant
err = WeGetControl (hMo, "Output:Digital(12bit)", value)
value → 4095
```

Output:Digital(8bit)

Description
Sets/Gets the 8-bit digital output value.

Parameter
0 to 255

Example (VisualBasic)
```vbnet
err = WeSetControl (hMo, "Output:Digital(8bit)", "255")
' Set the 8-bit digital output to 0xff.
Dim value As Variant
err = WeGetControl (hMo, "Output:Digital(8bit)", value)
value → 255
```
Event
None

Valid API functions

<table>
<thead>
<tr>
<th>API</th>
<th>Valid?</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>WeStart</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>WeStop</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>WeStartSingle</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>WeStartWithEvent</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>WeIsRun</td>
<td>Yes</td>
<td></td>
</tr>
<tr>
<td>WeLatchData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeGetAcqDataInfo</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeGetAcqData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeGetScaleData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeGetMeasureParam</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeSaveAcqData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeSaveScaleData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeSaveAsciiData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeGetCurrentData</td>
<td>Yes</td>
<td>The analog input value can be retrieved. The data are double-precision real values.</td>
</tr>
<tr>
<td>WeGetAcqDataEx</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeGetAcqDataSize</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeSaveAcqHeader</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeSavePatternData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeLoadPatternData</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeStartEx</td>
<td>No</td>
<td></td>
</tr>
<tr>
<td>WeStopEx</td>
<td>No</td>
<td></td>
</tr>
</tbody>
</table>