W&T

www.WuT.de

Manual

Installation, Startup and Application

Web-IO Digital 4.0

valid for:

#57737	Web-IO 4.0 Digital 2xIn, 2xOut
#57730	Web-IO 4.0 Digital 12xIn, 12xOut
#57731	Web-IO 4.0 Digital 12xIn, 12xOut, 1xRS232

Release 1.79 02/2026

Subject to error and alteration:
Since it is posssible that we make mistakes, you mustn't use any of our statements without verification. Please, inform us of any error or misunderstanding you come about, so we can identify and eliminate it as soon as possible.

Carry out your work on or with W&T products only to the extent that they are described here and after you have completely read and understood the manual or guide. We are not liable for unauthorized repairs or tampering. When in doubt, check first with us or with your dealer.

Content

1. Legal notices

Warning notice system

This manual contains notices that must be observed for your personal safety as well as to prevent damage to equipment. The notices are emphasized using a warning sign. Depending on the hazard level the warning notices are shown in decreasing severity as follows.

⚠DANGER

Indicates a hazard which results in death or severe injury if no appropriate preventive actions are taken.

⚠WARNING

Indicates a hazard which can result in death or severe injury if no appropriate preventive actions are taken.

⚠CAUTION

Indicates a hazard that can result in slight injury if no appropriate preventive actions are taken.

⚠NOTE

Indicates a hazard which can result in equipment damage if no appropriate preventive actions are taken.

If more than one hazard level pertains, the highest level of warning is always used. If the warning sign is used in a warning notice to warn of personal injury, the same warning notice may have an additional warning of equipment damage appended.

Qualified personnel

The product described in this manual may be installed and placed in operation only by personnel who are qualified for the respective task.

The documentation associated with the respective task must be followed, especi-

ally the safety and warning notices contained therein.

Qualified personnel are defined as those who are qualified by their training and experience to recognize risks when handling the described products and to avoid possible hazards.

Disposal

Electronic equipment may not be disposed of with normal waste, but rather must be brought to a proper electrical scrap processing facility.

Symbols on the product

Symbol	Explanation
	CE Mark The product conforms to the requirements of the relevant EU Directives.
	UKCA Mark The product complies with the requirements of the applicable directives of the United Kingdom (UK)
	WEEE Mark The product may not be disposed of with normal waste, but rather in accordance with local disposal regulations for electrical scrap.

2. Safety notices

General notices

This manual is intended for the installer of the Web-IOs described in the manual and must be read and understood before starting work. The devices are to be installed and put in operation only by qualified personnel.

Intended use

⚠DANGER

The Digital Web-IOs manufactured by Wiesemann & Theis are network remote switches with integrated web server and digital in- and outputs. They are used as a remote switching and monitoring unit, accessible via TCP/IP-Ethernet using various web and network protocols in accordance with the present manual.

Non-intended use is any other use or any modification to the described devices.

Electrical safety

⚠WARNING

Before beginning any kind of work on the Web-IO you must completely disconnect it from power. Be sure that the device cannot be inadvertently turned on again!

The Web-IO may be used only in enclosed and dry rooms.

The device should not be subjected to high ambient temperatures or direct sunlight, and it should be kept away from heat sources. Please observe the limits with respect to maximum ambient temperature.

Ventilation openings must be clear of any obstacles. A distance of 10-15 cm between the Web-IO and nearby heat sources must be maintained.

Input voltage and output currents must not exceed the rated values in the specification.

When installing be sure that no stray wires stick out through the ventilation slit

of the Web-IO into the housing. Ensure that no individual wires stand off from leads, that the lead is fully contained in the clamp and that the screws are tightly fastened. Fully tighten screws on unused terminals.

The power supply used for the Web-IOs must absolutely ensure safe isolation of the low-voltage side from the supply mains according to EN60950-1 and must have “LPS“ designation.

EMC

⚠NOTE

Only shielded network cables may be used for connecting the Web-IOs to the network.

In this case the Web-IOs meet the noise immunity limits for industrial applications and the stricter emissions limits for households and small businesses. Therefore there are no EMC-related limitations with respect to the usability of the devices in such environments.

The complete Declarations of Conformity for the devices described in the manual can be found on the corresponding Internet page at the W&T homepage: http://www.wut.de.

Batteries

The Web-IO Digital 4.0 contains a 3V lithium-manganese dioxide button battery type CR1632 for backing up the internal clock. This battery has an expected lifetime of 10 years and must be replaced only by a battery of the same type.

When using the Web-IO Digital 4.0 in a network environment with access to a time server, the battery is not essential for correct function of the device and can be removed.

⚠CAUTION
The battery may be removed or replaced by an electrotechnical specialist, only.

To remove the battery, open the housing as follows:

#57730, #57730
Press a pointed object against the side latch hook of the housing while pulling the base of the housing out of the top shell.

Remove the circuit board stack from the bottom of the housing.

The battery for the clock module is located in a holder on the mainboard.

After removing/replacing the battery, reassemble in reverse order.

#57737
Remove the green power supply terminal on the bottom of the device. Gently squeeze the body of the case at the front of the narrow sides with your thumb and forefinger. Now pull the housing with the printed circuit board out of the housing.

The battery for the clock module is located in a holder on the mainboard.

After removing/replacing the battery, reassemble in reverse order.

Batteries and rechargeables must not be disposed of with normal waste, recycling of used batteries and rechargeables is required by law. Used batteries may contain harmful substances which can damage the environment or your health if not disposed of properly.

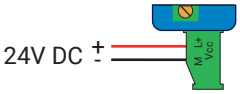
Batteries also contain important raw materials such iron, zinc, manganese or nickel and are recycled.

3. Quick Startup

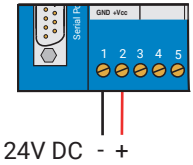
Network connection

Supply voltage

#57737



#57730 und 57731



For the first test leave the input and outputs unwired.

IP address assignment

Install Wutility-Tool (Download: http://wut.de/wutility)

After starting Wutility your Web-IO appears in the device list. If multiple devices are shown, please identify your device by the Mac address, that is printed on a white sticker at the device: “EN = 00c0:3d.....“ . If there is a DHCP server in your network, you can use the assigned IP address for a first test. Using the IP address icon in WuTility you can assign a free static IP address instead to the Web-IO.

Function test

Open the Web page of the Web-IO in a web browser using the address *http://<IP-address of the Web-IO>*.

4. Product introduction

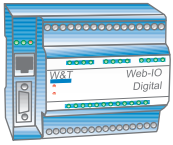
Hardware

The Web-IO devices differ in their mechanical design and hardware configuration:

#57737 - Web-IO 4.0 Digital 2xIn, 2xOut

Network interface:	RJ45 10/100BaseT / PoE
Power:	Screw terminal 24 … 48V DC or PoE
Inputs:	2 inputs -30 … +30V DC
	Switching threshold +8V (±1V)
Outputs:	2 outputs
	min. +6V, max. +30V DC
	Current driving max. 500mA
	with PoE total max. 125mA

#57730 - Web-IO 4.0 Digital 12xIn, 12xOut



Network interface:	RJ45 10/100BaseT
Power:	Screw terminal 12 … 24V DC
Inputs:	12 inputs -30 … +30V DC
	Switching threshold +8V (±1V)
Outputs:	12 outputs
	min. +6V, max. +30V DC
	Current driving max. 500mA
Data interface:	1 x RS232 (only for IP assignment)
Reset button:	for manual restart

Network security

All available network services are configurable and must first be enabled by the administrator. By default only browser access, inventorying via Wutility, and the port for initializing firmware updates are enabled. DHCP is also enabled.

You can explicitly specify for all communication paths whether the outputs may be accessible.

A list of the currently open TCP and UDP ports can be found in the navigation tree under *Port list*.

Access rights

The Web-IO is configured and operated by using a web browser. There are three authorization levels for access:

Guest
The guest has read-access to the status of inputs, counters and outputs without logging in.

User
A user can switch the outputs after logging in with a password if it is enabled for access via the browser.

Administrator
After logging in with a password the administrator has unrestricted configuration and access rights.

By default no passwords are assigned for the Web-IO. Simply click on the Login button.

After login the navigation tree on the left side can be used to open the enabled configuration areas. For help and information about the respective configuration possibilities click the *Info* buttons on the right side.

Clicking the *Apply* button makes the settings immediately effective.

For all other descriptions affecting the configuration, access with administrator

login is required.

Application and access possibilities

Browser access

Using password protected access, the status of inputs, counters and outputs can be monitored by browser access. You can also switch the outputs with the required access rights.

It is also possible to upload a web page created entirely according to your own needs to the device.

Email sending

The Web-IO offers the option of sending email messages depending on IO states or at fixed intervals. The Web-IO also supports authentication procedures prescribed by public providers.

Box-to-Box

Two Web-IOs can be configured so that the outputs of the first Web-IO follow the inputs of the second. This works in both directions when configured accordingly.

Integration into existing systems

The Web-IO allows communication using several protocols for integration into existing systems:

MQTT

In the context of Industry 4.0 and the “Internet of Things”, MQTT is an innovative communication channel.

The Web-IO can determine the status of the IOs via MQTT *Publish* to an MQTT broker and even accept the request to perform a switching action via MQTT *Subscribe*.

REST
REST (Representational State Transfer) is another web-based protocol that can be used to integrate the Web-IO into the environment of Industry 4.0 and the Internet of Things.

Web-API - HTTP requests / AJAX
The status of inputs, counters and outputs can be queried using HTTP requests. In addition the outputs can be directly controlled using HTTP requests.

OPC DA / OPC UA
Together with the W&T OPC Server the Web-IO can be accessed from any OPC client applications.

SNMP
The status of inputs, counters and outputs as well as the configuration and error status can be obtained via SNMP. For easy integration into SNMP systems, a private MIB is available which can be downloaded directly from the device.

Modbus-TCP
With Modbus TCP, the Web IO supports one of the most common industrial protocols. Any Modbus-TCP master can access the IOs by reading and writing the corresponding registers.

Individual applications
The Web-IO offers TCP and UDP socket access from your own applications.

In both cases the Web-IO supports addressing using command strings, but also by exchanging binary structures. With the support of HTTP requests your own web applications (e.g. with PHP or JavaScript) can also access the Web-IO.

Actions

Depending on configurable events at the IOs, the Web-IO can initiate actions such as sending an email message. Other actions include sending syslog messages or SNMP traps, writing to a file via FTP, sending data via TCP or UDP, or switching its own outputs.

5. Installation and wiring

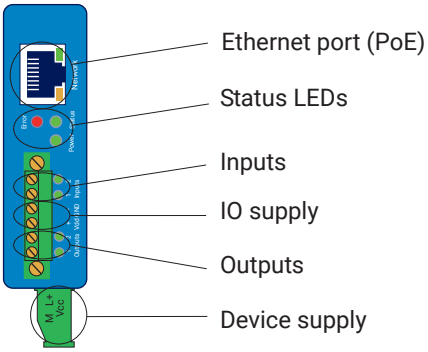
The described Web-IOs may be installed and wired by qualified personnel only. The generally applicable state of the art and corresponding prevailing regulations and standards must be observed.

#57737 - Web-IO Digital 2xIn, 2xOut

Installation

The Web-IO Digital 2xIn, 2xOut is intended for installation in a control cabinet. The Web-IO can be mounted on a 35mm top hat rail and has a width of 22 mm.

Connections and terminal assignment

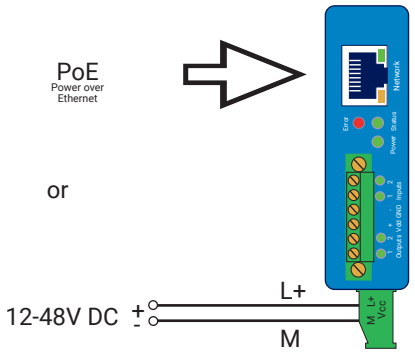


Connecting the supply voltage

The Web-IO is powered either by PoE (Power over Ethernet Class 2) or by a DC voltage between 12 and 48V. Power is connected to the green terminal at the bottom side of the device.

Only potential-free power supplies may be used for externally powering the Web-IO 57737. Their reference ground for the output voltage must not have any direct connection to the protective ground.

Simultaneous connection of an external power supply and a PoE infrastructure is not permitted.



With a typical industrial power supply of 24V, the Web-IO draws approx. 100mA of current.

Internal auxiliary voltage

After startup, a 24V auxiliary voltage can be output to the terminals *+Vdd* and *-GND* by selecting the menu item *Basic Settings » Inputs/Outputs » Internal I/O supply*.

The auxiliary voltage has a maximum load capacity of 150mA and can be used to drive the inputs through potential-free contacts. If the auxiliary voltage is activated, it is provided in the ON state at the outputs.

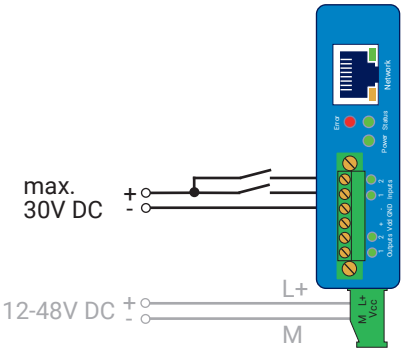
With PoE supply, the Web-IO does not require an external power supply if the total current consumption of the devices connected to the outputs does not exceed 150mA.

Under overload or short-circuit conditions the Web-IO turns off the auxiliary

voltage and generates an error message in the *Diagnostics* menu area. The *Delete report* button can be used to enable the auxiliary voltage again.

Input wiring

The Web-IO inputs are wired to the terminals labeled *Input 0* and *Input1*. The inputs are designed for voltages between -30V and +30V and are galvanically isolated from the internal circuitry by means of optocouplers.

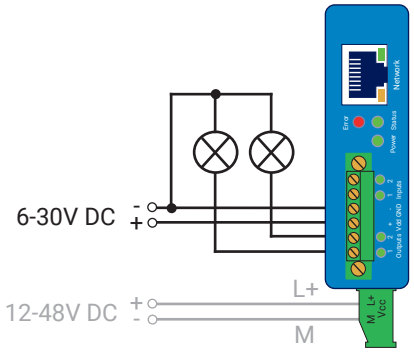


Positive voltages greater than 8V referenced to the *–GND* terminal are recognized as an *ON* signal and indicated as such by the corresponding LED.

If the auxiliary voltage has been enabled using
Basic settings >> Inputs/Outputs > Internal IO voltage, it can be tapped on the *+Vdd* terminal for driving the inputs through potential-free contacts.

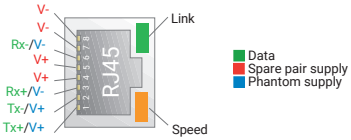
Output wiring

The outputs are current driving and have a maximum load capacity of 500mA each. The Vdd voltage which is applied to the terminals *+Vdd* and *-GND* is switched. When the auxiliary voltage is activated, it is switched out via the outputs. In this case the max. load is reduced to 150mA for both outputs.



Network connection

A shielded standard ethernet patch cable (min. CAT5) with RJ45 plugs can be used for the network connection.



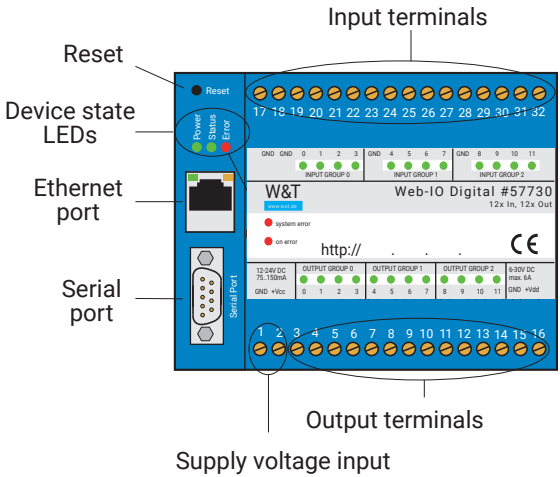
With PoE-enabled (Power over Ethernet) infrastructure, the Web-IO can be powered via the network connection.

#57730, #57731 - Web-IO Digital 12xIn, 12xOut

Installation

The Web-IO Digital 12xIn, 12xOut is intended for installation in a control cabinet. The Web-IO can be mounted on a 35mm top hat rail and has a width of 107 mm.

Connections and terminal assignment



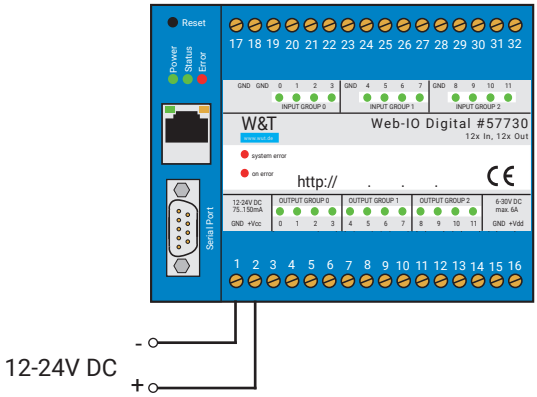
Terminal assignment

Terminal	Function
1	- GND / Device power supply
2	+ Vcc / Device power supply
3	Output 0
4	Output 1
5	Output 2
6	Output 3
7	Output 4
8	Output 5
9	Output 6
10	Output 7

Terminal	Function
11	Output 8
12	Output 9
13	Output 10
14	Output 11
15	- GND / Output power supply
16	+ Vdd / Output power supply
17	- GND / Input group 1
18	- GND / Input group 1
19	Input 0 / Input group 1
20	Input 1 / Input group 1
21	Input 2 / Input group 1
22	Input 3 / Input group 1
23	- GND / Input group 2
24	Input 4 / Input group 2
25	Input 5 / Input group 2
26	Input 6 / Input group 2
27	Input 7 / Input group 2
28	- GND / Input group 3
29	Input 8 / Input group 3
30	Input 9 / Input group 3
31	Input 10 / Input group 3
32	Input 11 / Input group 3

Connecting the supply voltage

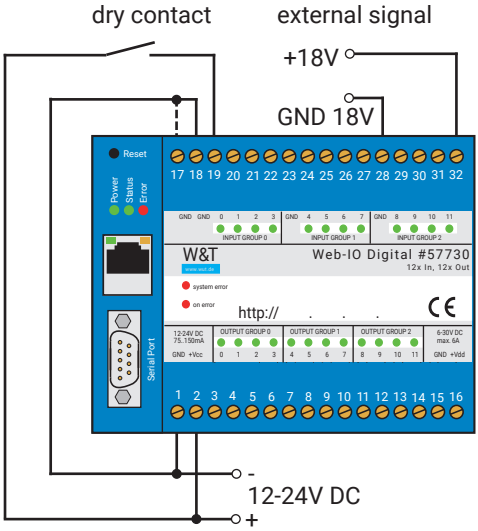
The Web-IO is powered at terminals 1 (*-GND*) and 2 (*+Vcc*) with a DC voltage between 12 and 24V.



With a typical industrial power supply of 24V, the Web-IO draws approx. 100mA of current.

Input wiring

The 12 inputs on the web-IO are divided into 3 groups of 4 inputs each. Each of the groups has its own reference ground (GND). The groups are galvanically isolated from each other and from the internal circuitry of the Web-IO.



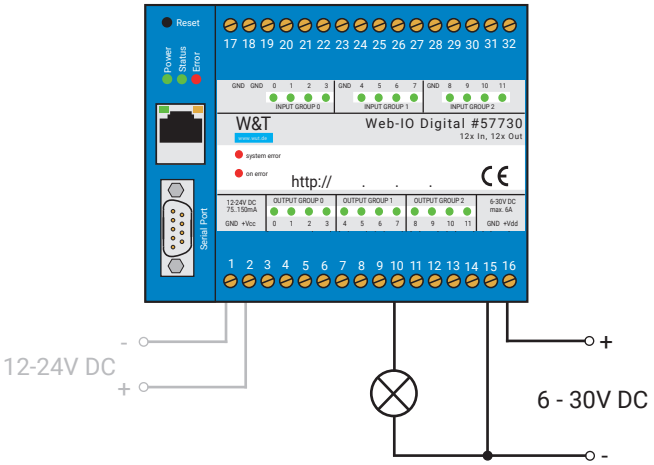
The galvanic isolation of the input groups allows the operation with different potentials. For example: Group 1 is connected to 18V and Group 2 to 24V.

The inputs are configured for voltages between -30V and +30V.

Positive voltages greater than 8V referenced to the *–GND* terminal are recognized as an *ON* signal and indicated as such by the corresponding LED.

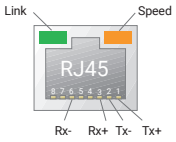
Output wiring

The outputs are current driving and have a maximum load capacity of 500mA each. The voltage Vdd, which is applied to terminals 15 (*-GND*) and 16 (*+Vdd*), is switched. The device supply Vcc and the output supply Vdd can be supplied from the same source if the same voltages (e.g. 24V) are used.



Terminal assignmentNetwork connection

A shielded standard ethernet patch cable (min. CAT5) with RJ45 plugs can be used for the network connection.



Serial port

The serial interface has a DTE assignment with the following pinout:

Pin	Function	Signal	Pin	Function	Signal
1	Input	DCD	6	Eingang	DSR
2	Input	RXD	7	Output	RTS
3	Output	TXD	8	Input	CTS
4	Output	DTR	9	Input	

Pin	Function	Signal	Pin	Function	Signal
5	--	GND			

On the Model 57730, the serial interface is used exclusively for configuration and emergency access (see Appendix).

With model #57731 the serial interface is additionally equipped with Com-Server functions (the separate manual for the Com-Server functions can be downloaded in the same area as this manual at WuT.de)

6. Initial start-up

After the Web-IO has been properly installed and wired, the power supply can be switched on. All three status LEDs should light up briefly. After approx. 5 seconds only the Power LED should remain on. The Status LED may flash. If a valid signal is detected on one of the inputs, the corresponding LED also lights up.

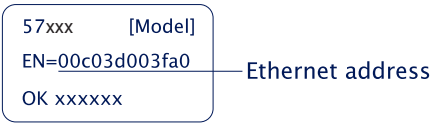
If the network connection is working, the green LED in the network socket signals an active link. The orange LED indicates the network speed:

On = 100MBit/s
Off = 10MBit/s

Assigning the IP address

At delivery, the Web-IO is set to IP address 0.0.0.0 and DHCP is activated.

Networks with DHCP
If a DHCP server is active in the network to which the Web-IO is connected, an IP address should be automatically assigned to the Web-IO. To be able to specifically access the Web-IO, you should configure a reservation in the DHCP server so that the Web-IO is always accessible under the same address. The required Ethernet address can be found on the white sticker on the device.



(If in doubt ask your network administrator)

Networks without DHCP
Install the WuTility program on a Windows PC (download from *http://www.WuT.de*). If you do not have a Windows PC, read the subchapter *Alternatives for IP address assignment* in the appendix.

When WuTility is started, the local subnet is scanned and all detected W&T network components are listed. Select your Web-IO and click the *IP address* icon. WuTility suggests the network parameters (subnet mask, gateway, DNS server) that also apply to the PC. If you want the Web-IO to work in the same subnet as the PC, you only need to adjust the IP address.

If you select *Address range > any network*, you can also enter parameters which are different from your local network, for example to pre-configure the Web-IO for a different network.

Changing the set IP parameters

To change the IP address, subnet mask, gateway or DNS server later, you can either use Wutility again or adjust the parameters using the browser under *Basic settings » Network*.

7. Basic settings

The further configuration of the Web-IO is done using a web browser. Enter the IP address of the Web IO in the browser command line. In the navigation tree, click *Login* and choose *Administrator* as user. By default, no password is assigned and a click on the login button is sufficient to configure the Web-IO with administrator rights.

Configuring Inputs and Outputs

In *Basic settings* » *Inputs/Outputs* you can give individual names to the inputs and outputs. These names replace the factory default names *Input n* and *Output n* in the visualization and any message texts.

Expanded Input settings

For special applications some input properties can be modified:

Input filters

A signal state must be present for the time in milliseconds entered here to be processed by the Web-IO. For example, bouncing of mechanical contacts can be suppressed.

Signal inversion

Normally signals greater than 8V are reliably detected as ON. Enabling *Signal inversion* means voltages greater than 8V are considered OFF.

Expanded output settings

For special applications some output properties can be modified:

Inverted output characteristics

Normally the outputs are switched off in OFF state (i.e. without signal) and switched on in ON state. By activating the inversion, the output configured in this way acts exactly reversed.

Pulse mode
By activating the *Pulse Mode*, the output automatically returns to the OFF state after the selected pulse duration when it is switched to the ON state. When switched on again during the pulse, the pulse duration starts counting again. *Reset Allowed* specifies that the output may also be switched to the OFF state during a current pulse.

Date / Time

In the *Date / Time* section you can define whether a periodical adjustment with a time server should take place. In addition, date and time can also be set manually. The configuration of a time zone and the daylight saving time can also be done here.

Language / Info

In addition to the language selection German or English, further display elements, including the logo, can be modified here.

Password

The passwords for administrator and user can be set in this section.

Please note that the same password should not be used for Administrators and Operators.

When assigning passwords, avoid the characters &, ?, # as well as country-specific special characters.

If the administrator password is no longer known, physical access to the Web IO is required to reset the passwords. See the chapter *Emergency Access* in the appendix of this manual.

Certificates

Protocols such as HTTPS or OPC UA are based on the TLS protocol. The encryption of the communication and the authentication of the communication partners is realized via certificates.

The Web IO identifies itself ex works with a self-signed certificate. Many applications consider such certificates to be a security risk. If the application requires secure authentication, the Web IO must be equipped with an individual certificate signed by a trusted certification authority.

Certificate Signing Request (CSR)

Here it is possible to generate a CSR with a new key pair and individual content.

By clicking the *Verify* button, the entered values are formally checked and the new key is generated. The new CSR can be downloaded via the *Download CSR* button.

Self signed certificate

A previously generated inidividual CSR can be self-signed by the device with the private key belonging to the CSR.

Upload certificate/upload certificate chain

A previously generated and downloaded CSR can be loaded into the device as a certificate after signature by an external certification authority. If a certificate chain belonging to the certificate is not already part of the certificate file, it can be uploaded separately afterwards. The files can be in PEM or DER format.

Install certificate/certificate chain

A previously uploaded certificate incl. associated certificate chain is installed in the device and used as a certificate within TLS connections after saving.

8. Basic applications

The Web-IO has a wide range of different communication channels and supports various standard protocols. We recommend that you only enable the communication channels that are actually required for your application. This limits the possibility of unauthorized access and manipulation.

First of all, we would like to introduce the three most frequently used communication channels:

Browser access

Access via a web browser has the special feature that, in addition to monitoring and operating the IOs, the configuration of the Web IO is also handled in this way if the user logs in accordingly:

Without login only the states of inputs and outputs can be observed.
With *User login* all settings and actions related to the IOs can be adjusted.
With *Administrator login* the entire configuration of the Web-IO can be acessed.

HTTP or HTTPS

Browser access for HTTP via port 80 is enabled by default. To change access to HTTPS or to change the port, select *Basic settings >> Network* in the navigation tree and then *Protocol* under *Access for Web services*. All other settings applicable in the browser can be made under *Web sites*.

Hide menu tree

When the configuration is complete, the display in the browser can be reduced to IO access. To do this, the option *Hide menu tree* must be activated under *Web pages » Browser access*. Via *http://<URL/IP of the Web-IO>/index* the menu tree can be shown temporarily and can be switched on again permanently via the option above.

IO access

For the access to the inputs, counters and outputs the Web-IO offers two prepared web pages:

Home

The *Home* page provides an overview of inputs, outputs and the configured actions. With the appropriate login, the outputs can be switched and the counter can be deleted. Both must first be enabled under *Web sites » Home*. By default this is disabled.

The menu point *Web sites » Home* offers several other display options for the *Home* page.

Direct access to the *Home* page without displaying the navigation tree is via *http://<URL/IP of the Web-IO>/home.*

If *Hide menu tree* is enabled, a password entry field appears on the *Home* page. After clicking the *Apply* button, outputs and counters can be operated until you leave the *Home* page again. Enabling *Web sites » Home » Save password for switching in browser* saves the password in the browser as a cookie and operation is immediately enabled again after opening the *Home* page in the same browser.

My Web page

The preloaded Web page in the Web-IO provides a compact overview of the IO states.

Under *Web sites » My Web page* the original website can be replaced by a self-designed one.

For this web page to dynamically update the states of inputs, counters and outputs, the option *Allow HTTP requests* must be activated under *Communication Channels » Web API*. You also specify here whether the outputs can be switched using HTTP requests.

Direct access to your own webpage without displaying the navigation tree is via *http://<URL/IP of the Web-IO>/user*

More details on programming your own Web pages can be found in the programming manual for the Web-IO. The manual for your Web-IO can be found on the respective Web data sheet page at www.WuT.de/article number, e.g. www.wut.de/57730

Sending email

A few basic settings are necessary in order to send email messages.

Network parameters

If you want to send via a mail server on the Internet, it is important that the basic network settings are correct. Check under *Basic settings » Network* especially whether *Gateway* and *DNS server* are specified correctly.

Mail server access

All mail server-specific settings can be made under *Communication paths » Mail*. The authentication method commonly used today is SSL/TLS. Further tips on the specific settings for the most common e-mail providers can be found in the info area under *Mail*.

Creating an email message

To create an email message, click the *Add* button under *Actions*. An input screen will appear for a new action.

Here you can determine the name for the action and what the trigger should be (e.g. the *ON* state of the input). A detailed description of the possibilities can be found in the *Actions* section.

Select *E-mail message* as the action. In the corresponding input mask you have the possibility to write an individual e-mail message. Use the placeholders described below, which are replaced by the current IO states, counter values, etc. when the e-mail is sent.

Placeholder	Description
<ix>	State of the inputs No. x (ON/OFF)
<ox>	State of the outputs No. x (ON/OFF)
<cx>	Counter state No. x
<i>	State of all inputs as hex. bit pattern
<o>	State of all outputs as hex. bit pattern
<dn>	Device Name
<inx>	Name of the input No. x

Placeholder	Description
<onx>	Name of the output No. x
<t>	Time stamp with date and time
<$y>	Year in format „YYYY“
<$m>	Month in format „MM“
<$d>	Day in format „DD“
<$h>	Hour in format „hh“
<$i>	Minutes in format “mm“
<$s>	Seconds in format „ss“
<hex: xx xx>	Any bytes as hexadecimal input
<rxxxx>	Modbus register value (Modbus TCP chapter)

Box-to-Box

Box-to-box operation connects two Web-IOs via the network so that the outputs of one follow the inputs of the other (ON at input 0 of Web-IO A switches output 0 of Web-IO B to ON).

In box-to-box mode, one Web IO must be configured as the master and the other as the slave. The master Web IO (client) establishes the connection to the slave Web IO (server). After successful setup of the connection, both Web-IOs work equally and the switching signals are transmitted in both directions.

9. Integration into existing systems

The Web-IO supports some common standards and protocols and can be easily integrated into many installed systems.

MQTT

After enabling MQTT and configuring in the menu branch *Communication paths » MQTT* the Web-IO supports two basic possibilities:

1. Passing the individual IO states and the counter value as an MQTT topic to an MQTT broker via MQTT publish.
2. Switching the output depending on topic contents received via MQTT subscribe.

Both cases are handled in the Web-IO as an action. A detailed description of the action philosophy used in the Web-IO can be found in the *Actions* section.

Publish IO states

To create a new MQTT publish, click the *Add* button under *Actions*. The input screen for a new action will appear.

Here you can specify a name for the action and what the initiator should be.

For example you can specify an input as the initiator and *ON* as the trigger state.

Choose *MQTT-Publish* as the action. In the following menu, enter the path to which the topic is to be written to the broker.

You can freely determine the contents of the topic, where the placeholders described in the infotext can be used.

Switching outputs via subscribe

You must also add a new action in this case. Choose *MQTT Subscribe* as the initiator.

Now enter the path via which the topic that contains the keyword for switching is transferred. As an action, configure *Switch Output » Switch this Web-IO Output*.

Then you determine in which state the output is to be switched or whether the state is to change.

Example:
A device writes the keyword ON as a topic in the path *wut/webio123/set0* of the broker specified in the Web IO. This path and topic are specified as an initiator under MQTT subscribe for the Web IO. As action the switching of the output to ON is determined.

The output is switched on each time ON is written. A second action can be used to determine how the output is to be switched off again.

The Web-IO as MQTT gateway

The flexible options offered by the Web-IO for configuring actions also allow the sending of e-mails, SNMP traps or messages via other communication channels, depending on the content of certain topics. More about this in the chapter *Actions*.

MQTT with W & T standard topics

For a quick integration without much configuration effort, the Web-IO offers the possibility to use predefined topics from W & T.

In order to work with W & T standard topics, MQTT must always be activated and configured under Communication paths >> MQTT. In addition, the Publish and Subscribe item must be enabled with W & T default topics.

In addition, you can select which IO states the Web-IO should publish to the configured broker and whether the switching of the outputs should be allowed by subscribe.

Structure of the standard topics

The structure of the topic path always follows the same pattern and consists of:

```
<Device name>/<get or set>/<function>/<IO-number>
```

The device name is in factory defaults :

```
wut-<last 6 digits of the MAC address>
```

The function direction is get (for publishing changes to input, output or counter)

and set for switching an output or deleting a counter.

Possible functions are `input`, `counter` or `output`

Via the IO number, starting at 0, the IO is specified.

Publish IO states
Example of the pulse of a state change at input 1:

```
wut-0a4711/get/input/1
```

Depending on the state, the payload will be `ON` or `OFF`.

Switching Outputs via Subscribe
Example for setting Output 5 using Subscribe:

```
wut-0a4711/set/output/5
```

Payload can be `ON`, `OFF` or `TOGGLE` to change state.

For reading and setting counters, the corresponding digits are transferred as a payload. To delete for example 0.

Both the topics and the payload are case-sensitive.

REST

The Web-IO uses REST (Representational State Transfer) to provide another web-based communication path.

Communication is carried out via Web-IO specific HTTP requests using the HTTP or HTTPS port specified under *Basic settings » Network » Access for Web services*.

To be able to exchange data via REST, access via *Communication paths >> Rest* must first be enabled.

If you wish to protect REST access against unauthorized manipulation, you can enable digest authentication. The requests must then take place as “admin” user with the Administrator password or as “operator” using the user password.

Here you can also specify whether REST is permitted to switch the outputs.

Read access

For read access REST uses the HTTP command GET.

The Web IO supports three formats for responses to REST requests:

- JSON
- XML
- Text

The format used for replies can be determined using the request. Using

```
http://<ip-adresse>/rest/json
```

for example opens the entire process image of the Web-IO in JSON format. The response body then looks as follows:

```
{
  “info” :
  {
    “request” : “ / rest / json”,
    “time” : “2016 - 09 - 09,
    09 : 42 : 54”,
    “ip” : “10.40.22.227”,
    “devicename” : “WEBIO - CAFE27”
  },
  “iostate” :
  {
    “input” : [
      {
        “number” : 0,
        “state” : 0
      },
      {
        “number” : 1,
        “state” : 0
      }
    ],
    “output” : [
      {
        “number” : 0,
        “state” : 0
      },
      {
        “number” : 1,
        “state” : 0
      }
    ],
    “counter” : [
      {
        “number” : 0,
        “state” : 0
      },
      {
```

```
        "number" : 1,
        "state" : 0
      }
    ]
  },
  "system" :
  {
    "time" :
    {
      "time" : "2016 - 09 - 09,
      09 : 42 : 54"
    },
    "diagnosis" : [
      {
        "time" : "06.09.2016 09 : 42 : 54",
        "msg" : "Gerätestatus : OK"
      }
    ],
    "diagarchive" : [
      {
        "time" : "06.09.2016 09 : 42 : 54",
        "msg" : "Gerätestatus : OK"
      }
    ]
  }
}
```


To query individual areas or points, you can formulate the request more detailed:

```
http://<ip-adresse>/rest/json/iostate/input
```

This causes the Web-IO to return the status of all inputs:


```
{
  "iostate" :
  {
    "input" : [
      {
        "number" : 0,
        "state" : 0
      },
      {
        "number" : 1,
        "state" : 0
      }
    ]
  }
}
```


The request

```
http://<ip-adresse>/rest/json/iostate/input/0
```

can be used to query the state of input 0.

```
{
  "iostate" :
  {
    "input" : [
      {
        "number" : 0,
        "state" : 0
      }
    ]
  }
}
```

Changing access

POST is used for accesses that change the switching state of the outputs or delete the counters.

For example to set the output to ON, a POST is sent to the following URL:

```
http://<ip-adresse>/rest/json/iostate/output/1
```

The following parameters are sent as payload:

```
Set=ON
```

The Web-IO sends the following response body:

```
{
  "iostate" :
  {
    "output" : [
      {
        "number" : 1,
        "state" : 1
      }
    ]
  }
}
```

The same URL can be used to turn the output off using the parameter Set=OFF or to change its state using Set=TOGGLE.

Clearing counters for example is done by using a POST to the following URL:

```
http://<ip-adresse>/rest/json/iostate/counterclear/1
```

No additional parameter needs to be sent.

The Web-IO responds:

```
{
  "iostate" :
  {
    "counter" : [
      {
        "number" : 1,
        "state" : 0
      }
    ]
  }
}
```

To receive the responses in one of the other formats, simply replace the keyword `json` with `xml` or `text`.

A detailed description of the supported REST requests and the structure of the replies can be found in the Web-IO Programming Manual (download at http://WuT.de.*). Follow the* Manual *link from the data sheet page for your Web-IO.*

OPC DA

The Web-IO is already preset for OPC operation by default. If you want to use OPC, you only have to activate OPC access under *Communication paths » OPC UA* and enable the switching of the outputs if required.

For your OPC client to communicate with the Web-IO the W&T OPC server must be installed. Access via third-party OPC servers is not provided.

Select the menu item *Devices » New I/O Device in the OPC* Server. Enter the IP address and password of your Web-IO and select the device type. Confirm with *OK*. Finally, you must accept the new entries as active configuration via the menu item *File » Save*.

OPC UA

In addition to the classic OPC access via the W&T OPC server, the Web IO can also be addressed directly via OPC UA.

The device provides OPC UA via a binary TCP protocol.

The preset port of the server service corresponds to the standard port for this application: 4840. The connection setup of your client is done accordingly with the call:

```
opc.tcp//<ip-adresse>:4840
```

Authentication

The device provides several authentication methods, with corresponding security policies. You have the choice between:

- No authentication — No security policy
- Sign — Security policy:
  Basic128 - RSA15
  Basic265
  Basic265-SHA256
  AES128-SHA256 RsaOaep
- Sign & Encrypt — Security policy:
  Basic128 - RSA15
  Basic265
  Basic265-SHA256
  AES128-SHA256 RsaOaep

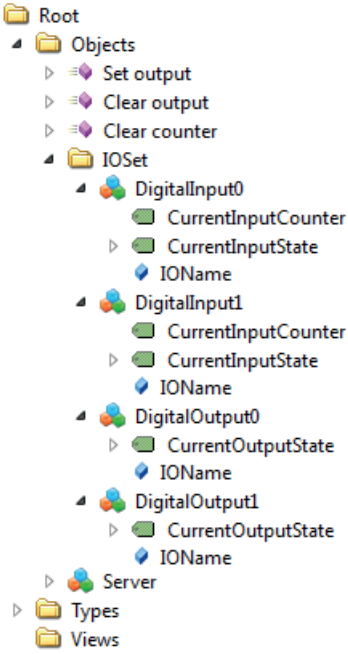
Also configure a UPC UA user name and password. If you select „No authentication“, this is not necessary.

Nodes und NodeIDs

The main nodes that can be used to retrieve the states of the IO endpoints are:

- CurrentInputCounter - Counter value of the pulses detected at the input
- CurrentInputState - Switching state of the inputs (ON or OFF)
- CurrentoutputState - Switching state of the outputs (ON or OFF)

The device provides you with the OPC UA tree shown in the following (here at the example of the Web-IO #57737).



A list of the most important nodes and the corresponding NodeIDs can be retrieved in the browser via `http://<ip-address>/opcua_nodes?PW=<password>&`.

If you want to replace the factory default NodeIDs with your own, download the node configuration in the menu branch *Communication paths >> OPC UA*. Enter the desired IDs behind the given IDs in the JSON file. Upload the modified file again and click *Apply*.

Changing the output switching states and clearing the counters is done by the following methods:

- Set output - sets the output defined by the index parameter to ON
- Clear output - sets the output defined by the index parameter toOFF
- Clear counter - sets the counter defined by the index parameter to 0

SNMP

Both the IOs and the configuration of the Web-IO can be accessed via SNMP. The assignment between parameters and values and the object identifiers (OID) is stored in the private MIB. The private MIB can be downloaded directly from the Web-IO under *Communication Channels » SNMP* (alternative download at http://www.WuT.de).

The MIB can easily be viewed with one of the common MIB browsers. This is the fastest way to get an overview of the assignment of the OIDs.

You can make all SNMP-related settings under *Communication paths » SNMP*. If the outputs are to be switchable via SNMP, they must be enabled here.

Opening an SNMP session

Read access is possible using SNMP-Get requests after enabling SNMP under *Communication paths >> SNMP*. Write/altering access requires a session login with an administrator password entry.

This is done using SNMP-SET via the OID which you can find in the MIB branch of your Web-IO under:

```
wtWebioEA….SessCntrl » wtWebioEA….SessCntrlPassword
```

Whether there is a valid session opened can be queried using a GET request to the OID:

```
wtWebioEA….SessCntrl » wtWebioEA….SessCntrlConfigMode.
```

(Return: 1 = valid session, 0 = no session.)

A session can be ended using SET to the OID

```
wtWebioEA….SessCntrl » wtWebioEA….SessCntrlLogout
```

During an SNMP session login attempts from the browser are rejected.

Access to inputs and outputs

Reading the inputs, counters and outputs is always possible using GET requests to the corresponding OID.

In the OID section

```
wtWebioEA….InOut
```

there are corresponding tables for this.

The MIB is symmetrically structured for the various Web-IO models. Input and output tables are kept, which have a different number of entries depending on the Web-IO type. In this way, the MIB remains compatible across devices..

Example: Querying the state of Input0

```
wtWebioEA….InOut » wtWebioEA….InputTable »
              wtWebioEA….InputEntry » wtWebioEA….InputState
```

An index is appended to the table entries for the individual IOs. For Input 0 for example „.1“ (return 0 = OFF and 1 = ON.)

There is also a corresponding table for the outputs:

```
wtWebioEA….InOut » wtWebioEA….OutputTable »
            wtWebioEA….OutputEntry » wtWebioEA….OutputState
```

Indexing works in the same way as for inputs. If a 1 is transferred via SNMP-SET, the output switches to ON, if a 0 is transferred, the output switches to OFF.

Switching the outputs requires a valid session.

Syslog

The menu item Communication Paths » Sylog can be used to configure the output of syslog messages for cold and warm starts, as well as other diagnostics. In addition, scalable debug information can be added to the output under Advanced Settings. Debug outputs can help, for example, to find out why emails are not being sent.

Modbus TCP

The Web-IO can be activated for Modbus slave operation via the menu item *Communication paths >> Modbus-TCP*. Here you can also specify whether the outputs may be switched via Modbus TCP. Normally, only one connection can be established to the server port provided (502) at any one time. If required, a further connection can be permitted via the Allow second connection checkbox.

The following tables show which function codes and register addresses are supported by the Web-IO.

Modbus memory addressing

Please note that the number of supported inputs, outputs, counters or alarms varies depending on the Web-IO model.

Bit range:

Description	Address (decimal)	Address (hexadecimal)	Storage model	Length (Byte)	Read bit with FC (decimal)	Read bit with FC (hexadecimal)	Write bit with FC (decimal)	Write bit with FC (hexadecimal)
Input 0	4096	0x1000	Bit	1	1, 2	0x01, 0x02		
Input 1	4097	0x1001	Bit	1	1, 2	0x01, 0x02		
Input 2	4098	0x1002	Bit	1	1, 2	0x01, 0x02		
Input 3	4099	0x1003	Bit	1	1, 2	0x01, 0x02		
Input 4	4100	0x1004	Bit	1	1, 2	0x01, 0x02		
Input 5	4101	0x1005	Bit	1	1, 2	0x01, 0x02		
Input 6	4102	0x1006	Bit	1	1, 2	0x01, 0x02		
Input 7	4103	0x1007	Bit	1	1, 2	0x01, 0x02		
Input 8	4104	0x1008	Bit	1	1, 2	0x01, 0x02		
Input 9	4105	0x1009	Bit	1	1, 2	0x01, 0x02		
Input 10	4106	0x100A	Bit	1	1, 2	0x01, 0x02		
Input 11	4107	0x100B	Bit	1	1, 2	0x01, 0x02		
Input 12	4108	0x100C	Bit	1	1, 2	0x01, 0x02		
Input 13	4109	0x100D	Bit	1	1, 2	0x01, 0x02		
Input 14	4110	0x100E	Bit	1	1, 2	0x01, 0x02		
Input 15	4111	0x100F	Bit	1	1, 2	0x01, 0x02		
Output 0	4128	0x1020	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 1	4129	0x1021	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F

Description	Address (decimal)	Address (hexadecimal)	Storage model	Length (Byte)	Read bit with FC (decimal)	Read bit with FC (hexadecimal)	Write bit with FC (decimal)	Write bit with FC (hexadecimal)
Output 2	4130	0x1022	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 3	4131	0x1023	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 4	4132	0x1024	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 5	4133	0x1025	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 6	4134	0x1026	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 7	4135	0x1027	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 8	4136	0x1028	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 9	4137	0x1029	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 10	4138	0x102A	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 11	4139	0x102B	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 12	4140	0x102C	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 13	4141	0x102D	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 14	4142	0x102E	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Output 15	4143	0x102F	Bit	1	1, 2	0x01, 0x02	5, 15	0x05, 0x0F
Action 1 state	4160	0x1040	Bit	1	1, 2	0x01, 0x02		
Action 2 state	4161	0x1041	Bit	1	1, 2	0x01, 0x02		
Action 3 state	4162	0x1042	Bit	1	1, 2	0x01, 0x02		
.....								
Action 28 state	4187	0x105B	Bit	1	1, 2	0x01, 0x02		
Action 29 state	4188	0x105C	Bit	1	1, 2	0x01, 0x02		
Action 30 state	4189	0x105D	Bit	1	1, 2	0x01, 0x02		
Exception state	4192	0x1060	Bit	1	1, 2	0x01, 0x02		
Config. state	4200	0x1068	Bit	1	1, 2	0x01, 0x02		
Action 1 Trigger	4160	0x1800	Bit	1	1, 2	0x01, 0x02		
Action 2 Trigger	4161	0x1801	Bit	1	1, 2	0x01, 0x02		
Action 3 Trigger	4162	0x1802	Bit	1	1, 2	0x01, 0x02		
.....								
Action 28 Trigger	4187	0x105B	Bit	1	1, 2	0x01, 0x02		
Action 29 Trigger	4188	0x105C	Bit	1	1, 2	0x01, 0x02		
Action 30 Trigger	4189	0x105D	Bit	1	1, 2	0x01, 0x02		

16 bit range:

Description	Address (decimal)	Address (hexadecimal)	Storage model	Length (Byte)	Read register with FC (decimal)	Read register with FC (hexadecimal)	Write register with FC (decimal)	Write register with FC (hexadecimal)
Inputs 0 - 15	8192	0x2000	16-Bit	2	3, 4	0x03, 0x04		
Outputs 0 - 15	8194	0x2002	16-Bit	2	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Action state 1 - 16	8196	0x2004	16-Bit	2	3, 4	0x03, 0x04		
Diag Error count	8198	0x2006	16-Bit	2	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Diag state 0 - 15	8199	0x2007	16-Bit	2	3, 4	0x03, 0x04		
Diag state 16 - 31	8200	0x2008	16-Bit	2	3, 4	0x03, 0x04		
Diag state 32 - 47	8201	0x2009	16-Bit	2	3, 4	0x03, 0x04		
Diag state 48 - 63	8202	0x200A	16-Bit	2	3, 4	0x03, 0x04		
Diag state 64 - 79	8203	0x200B	16-Bit	2	3, 4	0x03, 0x04		
Diag state 80 - 95	8204	0x200C	16-Bit	2	3, 4	0x03, 0x04		
Except./Conf.-state	8205	0x200D	16-Bit	2	3, 4	0x03, 0x04		

32 bit range:

Description	Address (decimal)	Address (hexadecimal)	Storage model	Length (Byte)	Read register with FC (decimal)	Read register with FC (hexadecimal)	Write register with FC (decimal)	Write register with FC (hexadecimal)
Inputs 0 - 15	20480	0x5000	32-Bit	4	3, 4	0x03, 0x04		
Outputs 0 - 15	20482	0x5002	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Action state 1 - 15	20484	0x5004	32-Bit	4	3, 4	0x03, 0x04		
Counter 0	20486	0x5006	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 1	20488	0x5008	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 2	20490	0x500A	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 3	20492	0x500C	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 4	20494	0x500E	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 5	20496	0x5010	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 6	20498	0x5012	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 7	20500	0x5014	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 8	20502	0x5016	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 9	20504	0x5018	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 10	20506	0x501A	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 11	20508	0x501C	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 12	20510	0x501E	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 13	20512	0x5020	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 14	20514	0x5022	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Counter 15	20516	0x5024	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10

Description	Address (decimal)	Address (hexadecimal)	Storage model	Length (Byte)	Read register with FC (decimal)	Read register with FC (hexadecimal)	Write register with FC (decimal)	Write register with FC (hexadecimal)
Diag Error count	20554	0x504A	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Diag state 0 - 31	20556	0x504C	32-Bit	4	3, 4	0x03, 0x04		
Diag state 32 - 63	20558	0x504E	32-Bit	4	3, 4	0x03, 0x04		
Diag state 64 - 95	20560	0x5050	32-Bit	4	3, 4	0x03, 0x04		
Diag state 96 - 127	20562	0x5052	32-Bit	4	3, 4	0x03, 0x04		
Diag state 128 - 159	20564	0x5054	32-Bit	4	3, 4	0x03, 0x04		
Diag state 160 - 191	20566	0x5056	32-Bit	4	3, 4	0x03, 0x04		
Serial number	24574	0x6000	32-Bit	8	3, 4	0x03, 0x04		
Ethernet Address	24578	0x6004	32-Bit	8	3, 4	0x03, 0x04		
Virtual Register 0	28672	0x7000	32-Bit	4	3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 1	28674	0x7002	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 2	28676	0x7004	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 3	28678	0x7006	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 4	28680	0x7008	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 5	28682	0x700A	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register ..								
Virtual Register 28	28728	0x7038	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 29	28730	0x703A	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 30	28732	0x703C	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10
Virtual Register 31	28734	0x703E	32-Bit		3, 4	0x03, 0x04	6, 16	0x06, 0x10

Register functions

All address details are to be understood as hexadecimal.

The Web-IO has various Modbus memory areas:

- Bit range (from address 0x1000)
- 16Bit range (from address 0x2000)
- 32-bit range or 2x16-bit range (from address 0x5000)

Addressing in the bit range is done bit by bit, i.e. 1 bit requires an address. In the 16-bit and 32-bit range, addressing takes place word by word (2 bytes).

Here is an overview of the most important registers:

The Inputs
can be found:

- in the bit range from 0x1000
- in the 16-bit range from 0x2000
- in the 32-bit range from 0x5000

The Outputs
can be found:

- in the bit range from 0x1020
- in the 16-bit range from 0x2002.
- in the 32-bit range from 0x5002.

The Counter
can be found in the 32-bit range from address 0x5004

Write counter values: Counters can be set to any value.

Modbus - virtual registers
The Web-IO provides 64 virtual 16-bit registers to which any values can be written by the Modbus master (high byte first). Writing to these registers does not trigger any special actions in the Web-IO. Instead, the virtual memory is used to transfer Modbus process data to web applications.

The address range for the virtual registers starts at 0x7000.

Virtual registers in web applications
The 64 registers (128 bytes) can be called up by web applications via an HTTP request.

```
modbusreg?PW=<password>&
```

The Web-IO responds with

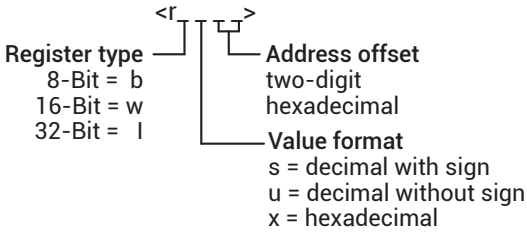
```
modbus;<High Byte1>;<Low Byte1>;<High Byte2>;<Low Byte2>;<High Byte3>;...
```

This means that all 64 registers (128 bytes) are output byte by byte separated by a semicolon after the word „modbus“.

Using JavaScript and programming techniques such as AJAX, process visualization can be implemented in the browser. In the simplest case, the virtual registers can be displayed in tabular form on the factory-set user page of the Web-IO (only if the Modbus operating mode has been activated in the WebIO).

Virtuelle Register in Actions

Although no actions can be triggered via the contents of the virtual registers, the values stored there can be integrated into the messages to be transmitted using placeholders (tags).



A more detailed description of the supported function codes and register addresses can be found in the Web-IO programming manual.

10. Actions

The Action principle allows the Web-IO to issue individual alarms and messages – but also to switch the outputs. This is done based on defined IO states or other events.

Up to 12 actions can be created and managed, whereby an individual name can be defined for each action.

Trigger

Inputs

Any input can be defined as an initiator. For the input you can specify whether a change from OFF to ON, a change from ON to OFF, or any state change should initiate an action.

Outputs

Any output can be specified as an initiator. For the output you can specify whether a change from OFF to ON, a change from ON to OFF, or any state change should initiate an action.

Counter

Any counter can be specified as the initiator. For the counter you must specify for which count value an action should be initiated. You also need to determine whether the counter is reset to zero after the action is initiated.

I/O combination

A combination of inputs and outputs can also initiate an action. Here you can specify whether the individual states should have an AND or OR operation performed.

Interval Timer

The Web-IO can be configured to perform actions at specified times. The times are entered in *Cron* format.

Valid characters:

- * represents all valid values in the respective input field (e.g. every minute or every hour)
- L Last day of the month
- - specifies a range of from…to (e.g. weekday "2-4" stands for Tuesday to Thursday, whereas entering "*" triggers the timer on all weekdays). In conjunction with "L", "-" is a minus sign, e.g. to define the penultimate day of the month ("L-1").
- / Interval within the specified range (e.g. minute "0-45/2" triggers the timer in a range between the 0th and 45th minute every two minutes (0, 2, 4, 6 ,8, 10, ... , 44)).
- , specifies an absolute value (e.g.: minute „0, 15 ,30" triggers the timer every full hour, every 15th minute and every 30th minute).

For example:
An action should be performed in the months of April to October every Monday at 8:00 a.m.

Minute:	0
Hour:	8
Date:	*
Month:	4-10
Day of week:	1

Device restart

The Web-IO distinguishes between two types when a restart is supposed to initiate an action:

- Cold start
  If the restart is initiated by hardware (applying/interrupting supply voltage or pressing the reset key) the Web-IO treats this as a cold start.

- Warm start
  A warm start can be initiated from the Web page under *Maintenance* by clicking the *Restart* button. Connecting to Port 8888 and using the administrator password will also cause a reset if the reset port is enabled.

MQTT Subscribe

If the Web IO receives the keyword configured as a topic, the action is executed. To do this, MQTT support must be activated under *Communication channels » MQTT*, and all necessary broker information must also be configured.

Actions

For actions which allow sending alarms, messages and other texts, placeholders can be used within the text which replace actual contents such as IO states, time etc. when performing an action.

Placeholder	Description
<ix>	State of the inputs No. x (ON/OFF)
<ox>	State of the outputs No. x (ON/OFF)
<cx>	Counter state No. x
<i>	State of all inputs as hex. bit pattern
<o>	State of all outputs as hex. bit pattern
<dn>	Device name
<inx>	Name of the input No. x
<onx>	Name of the output No. x
<t>	Time stamp with date and time
<$y>	Year in format „YYYY“
<$m>	Month in format „MM“
<$d>	Day in format „DD“
<$h>	Hour in format „hh“
<$i>	Minutes in format “mm“
<$s>	Seconds in format „ss“
<hex: xx xx>	Any bytes as hexadecimal input
<rxxxx>	Modbus register value (Modbus TCP chapter)

For text messages, a clear message can be stored in addition to the actual message that is sent upon triggering. The clear message is sent when the initiator for the action is no longer active – i.e. when the normal state returns. Sending messages takes different amounts of time, depending on the protocol. If the initiating state is only present for such a short time that the corresponding message could not be sent, only the clear message is sent.

Email message

The recipient, subject and contents of the email can be freely configured.

To send e-mail messages, access to the mail server must be configured and mail must be activated as a communication channel. All necessary settings can be made under *Communication paths » Mail.* In the info area you will find the general access data for the most common email providers.

SNMP trap

The IP address and host name of the SNMP server as well as the message texts can be freely configured.

To be able to send SNMP traps you must enable SNMP under *Communication paths » SNMP*. All other parameters which can be set there are not relevant for sending of SNMP traps.

MQTT publish

The Web-IO can write any information to an MQTT broker over a configurable path as an MQTT Topic.

To do this, access to the MQTT Broker must be configured under *Communication paths » MQTT.*

HTTP request

Another possible action is to send an HTTP request, as required by some devices, such as cameras, to trigger certain functions.

To do this, enter the complete URL with all parameters expected from the receiving device as the HTTP request.

Format:
`http://<Ip/Hostname>/<request>?Parameter1&Parameter2&ParameterN`

For those devices that require authentication with username and password, select which type of authentication should be used and fill in the appropriate fields.

With the request method you have the possibility to send a GET-request or via POST additional data.

TCP messages

When sending TCP messages the Web-IO operates as a TCP client. When initiating the action it opens a TCP connection to the specified TCP server address on the specified port, transmits the message or clear text, and then immediately closes the connection. Any replies from the server are ignored and discarded.

UDP messages

To be able to send UDP messages *UDP-Sockets* must be enabled in *UDP-Sockets ASCII-Mode* under *Communication paths » Socket-API*.

When sending UDP messages the Web-IO operates as an UDP peer. The message is transmitted in the form of an UDP datagram to the specified UDP peer address on the specified port. Any replies from the server are ignored and discarded.

Syslog messages

IP address and host name of the Syslog server, as well as the message texts can be freely configured.

To be able to send Syslog messages *Syslog* must be enabled under *Communication paths » Syslog*. All other parameters that can be set there are not relevant for sending Syslog messages.

FTP messages

The Web-IO can save message texts per FTP to a file.

To do this, FTP support must first be enabled under *Communication paths » FTP* and access to the FTP server must be configured.

The file name, message and clear texts can be freely formulated.

The options are used to distinguish whether *STOR* is used for each initiated action to completely overwrite the file or whether *APPEND* is used to append the message and clear texts continuously to the file.

Switching outputs

When switching outputs the Web-IO differentiates between switching its own outputs or switching the outputs on another Web-IO.

Switching the own outputs

The outputs can be switched to ON or OFF. Another possibility is to toggle the existing state.

Alternatively, several outputs can be switched simultaneously. For each selected output, you can specify whether it is to be set to ON or OFF.

Switching the outputs of another Web-IO

Also in this case, either one specific output or several outputs can be switched.

Specify the IP address of the Web-IO at which the outputs are to be switched. Specify the TCP port set as the browser access port for the destination Web IO. If the target Web-IO is protected with a password, this must also be entered.

For the destination Web-IO Allow HTTP requests must be enabled (Communication paths » Web-API) and the controlled outputs for switching from the browser and HTTP must be enabled.

The outputs of the older Web-IOs models #57630, #57631, #57634 und #57637 can also be switched. In this case the HTTP port of the Web-IO must be specified as the TCP port. The outputs must be set in *Output Mode* Menu.

Switching outputs as an action offers many interesting application possibilities.

Point-to-Point connection

Similar to box-to-box connections where the inputs on Web-IO A are mapped 1:1 to the outputs on Web-IO B, the switching state of one input can be mapped to any desired output on another Web-IO.

Point-to-Multipoint

By creating multiple actions which use an input as initiator, correspondingly more outputs on different Web-IOs can be controlled.

11. Access from own applications

In addition to the numerous standardized access possibilities, the Web-IO also offers the option of accessing from your own application.

This can be done via TCP/IP sockets from the common high-level languages. However, it is also possible to use common web techniques such as AJAX or PHP to communicate with the web IO.

Access using TCP/IP sockets

The Web-IO offers three ways to access using TCP/IP sockets:

- Command strings ASCII
- Binary structures BINARY
- HTTP requests AJAX

Command strings ASCII

The inputs and counters can be read and the outputs can be set by exchanging simple command strings.

Depending on the configuration the Web-IO operates in this mode as a TCP server or UDP peer.

A list of the supported commands and additional details on access via ASCII sockets can be found in the Web-IO programming manual. (download at http:www.WuT.de). Follow the manual link on the data sheet page of your Web-IO.

TCP server

To access the Web-IO as a TCP server using ASCII sockets, enable *TCP ASCII-Sockets* under *Communication paths » Socket-API*. Specify on which server port the Web-IO should accept connections. The Web-IO can provide up to four TCP connections on the specified port at the same time. Any additional connection attempt is rejected.

If the Web-IO does not receive a valid command within 30 seconds, it closes the connection and is then free for a new connection. The Web-IO behaves in the same way if an incorrect or unknown command is received.

The inputs are usually read using a polling procedure. Event-controlled processing is only possible after corresponding configuration of the input triggers.

UDP peer
To access the Web-IO via UDP using ASCII sockets, enable *UDP ASCII-Sockets* under *Communication paths » Socket-API*. Specify on which local UDP port the Web-IO should accept datagrams.

Via *Remote UDP-Port* you can define to which UDP-Port the answers of the Web-IO are sent. The entry *AUTO* specifies that the responses return to the port that is entered as the transmitter port in the received datagram.

The inputs are usually read using a polling procedure. Event-driven processing can be achieved by adding a corresponding action (see Actions section).

Binary structures BINARY

The Web-IO provides binary structures for various functions such as reading inputs, setting outputs, etc. Access takes place exclusively through the exchange of these structures.

In this mode the Web-IO can work as a TCP client, TCP server or UDP peer. Access can be password protected.

Four binary accesses are available which can be enabled and configured independently of each other under *Communication paths » Socket-API*.

In TCP Server mode, only one client can connect to the corresponding binary access at a time. Any further connection attempt will be rejected.

A detailed description of the supported HTTP binary structures and more details about access using BINARY sockets can be found in the Web-IO programming manual (download at http://www.WuT.de). Follow the manual link on the data sheet page of your Web-IO.

HTTP request

In addition to socket access the Web-IO can be addressed directly via HTTP using HTTP requests.

By default this access is blocked and must first be enabled using *Communication*

paths » Web-API.

A detailed description of the supported HTTP requests and more details about access using Web techniques such as AJAX and PHP can be found in the Web-IO programming manual (download at http://www.WuT.de). Follow the nanual link on the data sheet page of your Web-IO.

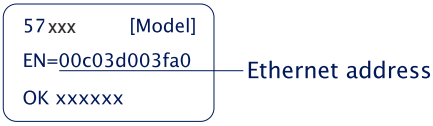
12. Appendix

Alternatives for IP address assignment

In case IP addresses cannot be assigned using DHCP or the Wutility Tool, the Web IO offers two further options:

Assigning the IP address using the ARP command
This method can be used when the Web-IO does not yet have an IP address and the entry is 0.0.0.0. Another prerequisite is that the Web-IO and computer are in the same network segment.

Read the Ethernet address of the Web-IO from the label at the side of the housing:



Now use the following command line from the ARP table of the computer to enter a static entry:

```
arp -s [IP address]  [MAC address]
```

Example under Windows:
```
arp -s 10.40.72.15 00-C0-3-00-3F-A0
```

Example under SCO UNIX:
```
arp -s 10.40.72.15 00:C0:3D:00:3F:A0
```

Then start the Web browser and enter

```
http://<IP address>
```

In Windows environments IP addresses may only be entered without leading zeros.

The Web-IO accepts the IP address of the first network packet sent to its Ethernet address as its own and saves it in non-volatile memory. All other settings can now

be made conveniently using web-based Management.

Assigning the IP address through the serial port

Only #57730 and #57731.

- connect the Web-IO to a serial port of your computer.
  For a standard PC, an RS232 crossover cable is required.
- open the used COM port in a terminal application (e.g., Easyterm) with the following parameters: 9600 baud, no parity, 8 bits, 1 stop bit, no handshake.
- Press the *Reset* button on the Web-IO for approx. 1 second.
- Now press the X button and hold it down until IPno. + <Enter> appears in the terminal program.
- Enter the IP address and confirm with Enter.

The Web-IO responds with the new IP address. If the serial input is incorrect, the Web-IO responds with FAIL.

Serial deactivation of the DHCP-/BOOTP client

The DHCP-/BOOTP function of the Web-IO can be turned off while serially assigning the IP address. We recommend doing this in any case where the IP assignment will not be expressly performed using DHCP/BOOTP.

To deactivate the DHCP/BOOTP client, enter the option „-0“ (zero) immediately following the IP address (no spaces!) and finish the entry with *<Enter>*.

```
xxx                                          -> Web-IO
IP no.+<ENTER>:                              <-Web-IO
10.40.72.15,255.255.255.0,10.40.1.1          -> Web-IO
10.40.72.15 ,255.255.255.0,10.40.1.1-1       <- Web-IO
```

This works in the same way if subnet mask and gateway are also entered. You can reactivate the function at any time later by using web-based management in the browser.

Firmware update

The firmware of the Web-IOs is continuously being improved to meet the constantly changing requirements of growing networks.

The current firmware for your Web-IO can be found on the respective Web data sheet page at www.WuT.de/article number, e.g. www.wut.de/57730

To install the firmware update, you need a Windows PC with the WuTility tool installed (included in the firmware archive) and unrestricted network access to the Web-IO.

Start WuTility, highlight your Web-IO in the inventory list and click on *Firmware* in the icon bar. Select the corresponding UHD file. WuTility will guide you through the update process.

Do not interrupt either power or the network connection during the update.

All the settings in the Web-IO are retained and the Web-IO should be immediately ready to use following the update.

Security advice

The following sections contain information and recommendations relevant from the point of view of IT security for commissioning, configuration, operation and maintenance of the Web IO models described in these instructions.

Function and typical application

Web-IOs offer the possibility to transmit or control the states of electrical switching signals via an Ethernet connection within higher protocol instances.

All Web-IO models are based on W&T's own operating system and ar free of open-source components and third-party software at their core. Out of the box, Web IOs are designed to operate in a secure network environment.

The factory settings focus on providing the lowest possible latency and therefore unsecured configuration access via HTTP.

In insecure network environments and/or with increased security requirements, additional measures must be taken to prevent unauthorized access.

With the exception of the display in the browser, all other access options to the inputs and outputs are disabled.

Requirements for integrators and operators

Depending on the individual network environment and the security requirements, the factory settings for operational use must be checked from a security perspective. Changes and/or additional measures may be required by the integrator or operator.

These include in particular:

- Selection of a secure password in terms of length and composition
- Deactivation of unneeded services and/or access restrictions through an upstream external firewall.
- Installation of an individual device certificate within a PKI environment.
- Protection of the Web IOs against unauthorized physical access

Further details on this can be found in the following of this chapter as well as in the previous descriptions of the individual operating modes.

Installation location

Der Installationsort des Web-IOs muss gewährleisten, dass keine unauthorisierten physikalischen Zugriffe erfolgen können (z.B. geeignet gesicherter Raum, Schaltschrank etc.). Ein physikalischer Zugriff auf das Web-IO birgt z.B. folgende Risiken:

- Decommissioning of the device (removing network cable, power supply ...) and loss of all connections to communication partners.
- Depending on the model, reset to factory settings by pressing and holding the reset button.

Startup

The commissioning of the Web IO is divided into the assignment of the IP address (DHCP, WuTility, static ARP entry, depending on the model serial port) and the subsequent further configuration via Web-Based-Management. With the factory setting, all configuration services are freely accessible. Commissioning must therefore be carried out in such a way that no unauthorized access can take place until the system password has been assigned and a secure configuration has been established.

A suitable measure is, for example, to perform commissioning via a point-to-point connection with the configuring computer. Only then is the Web IO then connected to the actual target network.

Password

Operational use of the Web IO without a password should not take place. The password is the central protection against unauthorized access to the configuration and management of the Web IO. Depending on the selected communication path, the password also protects access to the inputs and outputs

We recommend the use of a secure password with a length of at least 15 characters, consisting of upper and lower case letters, numbers and special characters (not allowed are &, # and /)

The system password is transferred to the Web IO in plain text for WBM access via HTTP. The transmission is only encrypted during configuration via HTTPS.

For password-protected access from supposedly insecure or public networks, additional measures such as the use of a VPN tunnel must be taken.

Registration for safety-related information

Devices can be registered with W&T via the WuTility inventory tool. In the event of security-relevant updates and/or information, we will notify you immediately by email.

In addition to the personal data provided, device-specific data is also stored during registration.

Operation and configuration

Ex works, all accesses or communication paths are deactivated except for browser access.

We recommend activating only those communication channels and services that are actually required for operation.

An overview of the possible communication channels can be found in the following table.

Communication path / protocol	Connection type	Active in factory defaults	Local port	Configurable	Remoteport	Configurable	Password-protected	unencrypted transfer
Wutility Inventory	UDP	X	8513	X	dynamic			
Wutility IP Assignment	UDP	X	68		67		X	X
DHCP	UDP	X	68		67			
HTTP	TCP-Server	X	80	X	dynamic		X	X
HTTPS	TCP-Server		443	X	dynamic		X	
DNS	UDP	X	dynamic		53			
NTP	UDP	X	dynamic		123			
Geräte-Reset	TCP-Server	X	8888	X	dynamic		X	X
Device update initialization	TCP-Server		8002	X	dynamic		X	X
Device update firmware data	UDP		69		dynamic		X	X
Mail	TCP-Client		dynamic		587	X	X	
Box-to-Box 1 Master	TCP-Client		dynamic	X	49157	X	X	
Box-to-Box 1 Slave	TCP-Server		49157	X	dynamic		X	
Box-to-Box 2 Master	TCP-Client		dynamic	X	49158	X	X	
Box-to-Box 2 Slave	TCP-Server		49158	X	dynamic		X	
MQTT	TCP-Client		dynamic		1883	X	X	X
SMQTT	TCP-Client		dynamic		8883	X	X	
REST (HTTP)	TCP-Server		80	X	dynamic		X	X
REST (HTTPS)	TCP-Server		443	X	dynamic		X	
Web-API (HTTP)	TCP-Server		80	X	dynamic		X	X
Web-API (HTTPS)	TCP-Server		443	X	dynamic		X	
TCP-ASCII-Socket Server	TCP-Server		42280	X	dynamic		X	X
UDP-ASCII-Socket Peer	UDP-Peer		42279	X	dynamic	X	X	X
BINARY 1 TCP Sockets	TCP-Client		dynamic	X	49153	X	X	
BINARY 1 TCP Sockets	TCP-Server		49153	X	dynamic		X	
BINARY 1 TCP Sockets	UDP-Peer		45889	X	45889	X		
BINARY 2 TCP Sockets	TCP-Client		dynamic	X	49154	X	X	
BINARY 2 TCP Sockets	TCP-Server		49154	X	dynamic		X	
BINARY 2 TCP Sockets	UDP-Peer		45890	X	45890	X		
BINARY 3 TCP Sockets	TCP-Client		dynamic	X	49155	X	X	
BINARY 3 TCP Sockets	TCP-Server		49155	X	dynamic		X	
BINARY 3 TCP Sockets	UDP-Peer		45891	X	45891	X		
BINARY 4 TCP Sockets	TCP-Client		dynamic	X	49156	X	X	
BINARY 4 TCP Sockets	TCP-Server		49156	X	dynamic		X	

Communication path / protocol	Connection type	Active in factory defaults	Local port	Configurable	Remoteport	Configurable	Password-protected	unencrypted transfer
BINARY 4 TCP Sockets	UDP-Peer		45892	X	45892	X		
Modbus-TCP	TCP-Server		502	X	dynamic			
OPC DA	TCP-Server		49159	X	dynamic		X	
OPC UA	TCP-Server		4840	X	dynamic		X	
SNMP V1	UDP-Peer		161		dynamic			
SNMP V2	UDP-Peer		161		dynamic		X	X
SNMP V3	UDP-Peer		161		dynamic		X	
SNMP-Trap	UDP-Peer		161		162	X		
SYSLOG	UDP-Peer		dynamic		514	X		
FTP control connection	TCP-Client		dynamic		21	X	X	X
FTP data connection (active)	TCP-Server		dynamic		dynamic			
FTP data connection (passive)	TCP-Client		dynamic		dynamic			
HTTP-Request (Actions)	TCP-Client		dynamic		80	X	X	X
HTTPS-Request (Actions)	TCP-Client		dynamic		443	X	X	
TCP-Message (Actions)	TCP-Cleint		dynamic		8000	X		
UDP-Message (Actions)	UDO-Peer		dynamic		8500	X		
Accesses for the Com-Server function (only 57731)								
Com-Server configuration (Telnet)	TCP-Server	X	1111	X	dynamic		X	X
Socket access serial data	TCP-Server	X	8000	X	dynamic			
Control access serial port	TCP-Server	X	9094	*	dynamic		X	X
Port reset	TCP-Server	X	9084	X	dynamic			
Configuration download	TCP-Server	X	8003		dynamic		X	X
Configuration upload	TCP-Server	X	8004		dynamic		X	X
Telnet	TCP-Server		6000	X	dynamic			
Telnet	TCP-Client		dynamic		0	X		
FTP	TCP-Server		7000	X	dynamic			
FTP	TCP-Client		dynamic		0	X		
Socket Client serial data	TCP-Client		dynamic		0	X		
Socket UDP Peer serial data	UDP-Peer		8000	X	0	X		
Socket for InQueueCopy	TCP-Server		0	X	dynamic			

The control port for serial access must always be 1094 higher than the TCP port configured for serial socket access.

Configuration via HTTPS / PKI environments if possible

The TLS protocol used by HTTPS provides encrypted and authenticated access to the web interface of the web IO. This also applies to access via the Web API and the rest access. To protect the exchanged configuration data, commands and the system password, we recommend activating HTTPS especially in insecure network environments. As protection against man-inthe-middle attacks, the self-signed default certificate should also be replaced by an individual, own certificate.

Encrypted communication

The hardware platform of the Web-IO combines low latency with low power consumption. As a result, the key length of the possible certificates is limited to 1024 bits and the Web IO supports TLS1.2 at most. In applications with higher requirements, additional measures may have to be taken (e.g. VPN).

TLS encrypted communication is possible in the following operating modes:

- HTTPS (Browser)
- HTTPS (Web-API)
- HTTPS (REST)
- MQTT (SMQTT)
- Mailing
- OPC UA

The computationally intensive TLS encryption functions can have an impact on the latencies of data transmission. For time-critical switching and acquisition tasks, protocols should therefore be tested for their compatibility with HTTPS accesses. This includes, in particular, any security scans in the network. In some cases, these open a large number of TLS connections within a short time and can thus lead to interruptions or timeouts of the data traffic.

Islanding of the subnet via router/firewall

For applications that communicate unencrypted with the Web IO, the communication partners (e.g. Web IO and PC) should be isolated in a separate network segment via a firewall to protect against spying. For example, with the aid of a W&T Microwall, this also protects the communication partners from damaging events (broadcast storms, overloads, etc.) in the main network.

Appropriate firewall rules limit cross-network access to the minimum necessary.

Firmware updates

W&T publishes firmware updates for the Web IOs in order to eliminate functional errors, possibly discovered vulnerabilities or also to extend functions.

The upload to the device is done with the help of the WuTility management tool.

Update files always contain the entire firmware or the entire system of the Web IO. For this reason, firmware updates are always associated with a restart of the Web IO and thus also an interruption of the operational mode. Individual configuration data (IP parameters, firewall rules, etc.) are not affected by a firmware update and are retained.

The Web IOs are based on S&T's own operating system and do not contain any third-party components at their core (e.g. Linux, external TCP stacks, etc.). Compromise with common malicious code existing for these systems is therefore not possible.

The firmware is uploaded via TFTP (UDP) and the system password is transmitted in plain text on the network side during this process. In insecure networks or in environments with increased security requirements, additional external measures are therefore required (e.g. VPN).

For more details on a firmware update, refer to the Firmware Update chapter.

Service, maintenance and decommissioning

Despite high quality standards, electronics can fail at any time, e.g. due to external events. Depending on the availability requirements of the respective application, we recommend taking appropriate precautions.

- Backup/storage of the device configuration
- If necessary, provision of a replacement device
- Documentation of the procedure for device replacement

During decommissioning, all confidential information stored in the Web IO (IP ranges, external access data, etc.) should be reset to the factory settings to protect them. This can be done either via the web-based management or via hardware by pressing and holding the reset button or the device-internal jumper.

Emergency access

In case you have forgotten the passwords for the Web-IO or simply want to reset the device to its factory defaults, there are model-dependent emergency accesses. In any case, you need physical access to the device.

Models #57730, #57731 (delivered until March 2018)

The emergency access for these models is established via the serial RS232 interface.

First connect the Web-IO serially to a computer. For a standard PC, a crossover RS232 cable (=null modem cable) is required.

Open the COM port used in a terminal program (e.g. Easyterm) using the following transmission parameters: 9600 baud, no parity, 8 bits, 1 stop bit, no handshake.

Delete passwords

Press the *Reset* key for approx 1 second. Then press the *P* key and hold it until the system error and on error LEDs begin to flash rapidly.

You may now access the Web-IO without a password.

Factory default reset

Press the reset key for approx 1 second. Then press the *F* key and hold it until the system error and on error LEDs begin to flash rapidly.

The configuration of the Web-IO now corresponds to the factory default settings.

For Web-IOs delivered from March 2018 onwards, the delivery status can be restored by pressing and holding (approx. 10 seconds) the *Reset* button. After approx. 30 seconds, the *Reset* button must be pressed again briefly to restart the Web-IO.

Models # 57730, # 57731 (delivered from March 2018)

On these models, the reset button (above the network jack) is recessed and can only be operated with a thin object.

Delete passwords

For Web-IOs delivered from March 2018, an emergency access can be activated by a long press (time window 3 - 7 seconds, the LEDs below the reset button flash slowly) of the recessed reset button. For about 5 minutes, an emergency page can be opened via the browser when the Web-IO IP address is called. Here all passwords can be deleted via a button.

Reset to factory settings

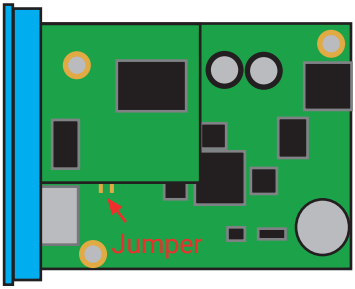
If the reset button is pressed for a long time (more than 7 seconds), the LEDs below the button start to flash quickly. The delivery state is restored. After approx. 30 seconds, the reset button must then be pressed short again to restart the Web-IO

Model #57737

In this model there is a jumper on the main board which is used to initiate a full reset to factory defaults.

First remove all connectors. Open the housing by pressing the case body slightly together with thumb and index finger - this causes the case walls to spread out. Now the front panel together with the circuit board can be pulled out.

There is a jumper (normally open) next to the network connector:



Close the jumper and power the Web-IO for approx. 30 seconds. The configuration has now been restored to its factory defaults.

11.Technical data

Web-IO #57730/#57731

Connections, displays and control elements:

Digital outputs:	12 x digital output 6V … 30V, 500mA Grouping of 2 or 4 outputs Maximum group current 2A Maximum total current 6A Short-circuit-protected
Digital inputs:	12 x digital inputs, Max. input voltage ±30 V Protected against polarity reversal within this range Switching threshold 8V ±1V „On“ current = 2.2 mA integrated 32-bit counter
Network:	10/100BaseT autosensing
COM-Port:	RS232 for IP assignment only
Power supply:	12 … 24V DC (approx. 100mA@24V)
Output supply:	6 … 30V DC
Connections:	2 x 16x screw terminal blocks for IOs and power 1 x RJ45 for network 1 x DB9 plug for RS232
Displays:	Status LEDs for network Error LEDs for system and application 24 LEDs for digital status

Housing and other data:

Housing:	Plastic housing for rail installation 106.8 x 87.8 x 62.6 (L x W x H)
Weight:	approx. 250 g

Web-IO #57737

Connections, displays and control elements:

Digital outputs:	2 x digital output 6V … 30V, 500mA Grouping of outputs Maximum total current 1A Short-circuit-protected
Digital inputs:	2 x digital inputs, Max. input voltage ±30 V Protected against polarity reversal within this range Switching threshold 8V ±1V „On“ current = 2.2 mA integrated 32-bit counter
Network:	10/100BaseT autosensing
Power supply:	Power-over-Ethernet (PoE) or 24 … 48V DC (approx. 100mA@24V)
Output supply:	6 … 30V DC over screw terminal or internal supply 24V max. 125mA
Connections:	1 x 6x screw terminal blocks for IOs and IO power 1 x 2x screw terminal blocks for power supply 1 x RJ45 for network
Displays:	Status LEDs for network Error LEDs for system and application 4 LEDs for digital status

Housing and other data:

Housing:	Plastic housing for top hat rail installation 105x22x75mm (LxWxH)
Weight:	approx. 125 g

Web-IO #57738

Connections, displays and control elements:

Digital outputs:	8 potential-free relay contacts for 30V/5A DC (2A when using inductive load) for 48V/5A AC (2A when using inductive load) max. 1800 switching cycles per hour
Digital inputs:	12 x digital inputs, Max. input voltage ±30 V Protected against polarity reversal within this range Switching threshold 4V ±1V „On“ current = 2.2 mA integrated 32-bit counter
Network:	10/100BaseT autosensing
Power supply:	12 … 24V DC (approx. 300mA@12V)
Connections:	1 x 4x screw terminal blocks for power supply 3 x 12x screw terminal blocks for IOs and power 1 x 11x screw terminal blocks for IOs and power 1 x RJ45 for network
Displays:	Status LEDs for network Error LEDs for system and application LEDs for digital status

Housing and other data:

Housing:	Plastic housing for top hat rail installation 90x116x56 mm (L x W x H)
Weight:	approx. 310 g

General data

Galvanic isolation:	Digital outputs - network: min. 1000 V Digital inputs - network: min. 2000 V Digital inputs - outputs: min. 1000 V
Protocols:	TCP- and UDP- Sockets, Client and Server SNMP incl. traps SMTP email sending OPC server Modbus-TCP Inventorying, group management
Response times:	Data and switching traffic: typically 40ms
Enclosure rating:	IP20
Storage temperature:	-25°C … 70°C
Operating temperature:	0°C … 60°C
Permissible relative humidity:	5 … 95% RH (non-condensing)

Wiesemann & Theis GmbH
Porschestraße 12
D-42279 Wuppertal

Mail info@wut.de
Web www.wut.de

Tel. +49 (0)202 2680-110
Fax +49 (0)202 2680-265