

# W&T

www.WuT.de

## Manual

Installation, Startup and Application

### Web-IO Digital 4.0

gvalid for:

#57733

Web-IO 4.0 Digital 4xIn/Out

#57736

Web-IO 4.0 Digital 16xIn/Out

Release 1.52 December 2021

© 10/2021 by Wiesemann und Theis GmbH  
Microsoft, MS-DOS, Windows, Winsock and Visual Basic  
are registered trademarks of the Microsoft Corporation.

Subject to error and alteration:

Since it is possible 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

<b>1. Legal notices</b> .....	<b>5</b>
Warning notice system .....	5
Qualified personnel .....	5
Disposal.....	6
Symbols on the product .....	6
<b>2. Safety notices</b> .....	<b>7</b>
General notices .....	7
Intended use.....	7
Electrical safety.....	7
Batteries .....	8
<b>3. Quick Startup</b> .....	<b>11</b>
Network connection.....	11
Supply voltage.....	11
IP address assignment .....	11
Function test .....	11
<b>4. Product introduction</b> .....	<b>12</b>
Hardware.....	12
Network security .....	13
Access rights.....	13
Application and access possibilities.....	14
Actions .....	15
<b>5. Installation and wiring</b> .....	<b>16</b>
Wiring #57733.....	17
Wiring #57736.....	22
Network connection.....	24

<b>6. Initial start-up .....</b>	<b>25</b>
Assigning the IP address .....	25
Changing the set IP parameters .....	26
<b>7. Basic settings.....</b>	<b>27</b>
Configuring Inputs and Outputs.....	27
Date / Time.....	28
Language / Info .....	28
Password .....	28
Certificates.....	29
<b>8. Basic applications.....</b>	<b>30</b>
Browser access .....	30
Sending email.....	32
Box-to-Box.....	33
<b>9. Integration into existing systems.....</b>	<b>34</b>
MQTT.....	34
REST	36
OPC DA.....	40
OPC UA.....	41
SNMP .....	43
Modbus-TCP .....	44
<b>10. Actions .....</b>	<b>47</b>
Trigger .....	47
Actions .....	49
<b>11. Access from own applications .....</b>	<b>53</b>
Access using TCP/IP sockets.....	53
<b>12. Appendix .....</b>	<b>56</b>
Alternatives for IP address assignment .....	56
Firmware update .....	57
Emergency access.....	58
<b>12. Technical data.....</b>	<b>59</b>

# 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, especial-

ly 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
	<p>CE Mark</p> <p>The product conforms to the requirements of the relevant EU Directives.</p>
	<p>WEEE Mark</p> <p>The product may not be disposed of with normal waste, but rather in accordance with local disposal regulations for electrical scrap.</p>

## 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.

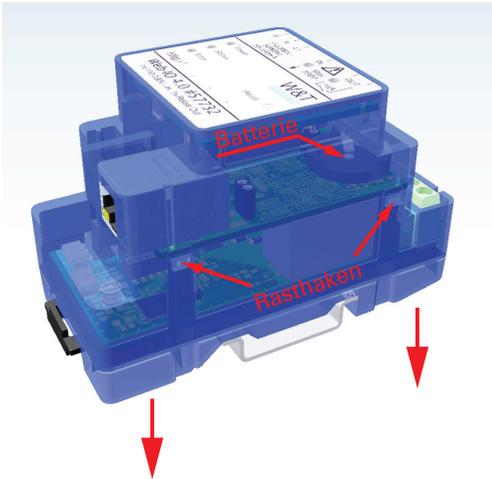
When using the Web-IO Digital 4.0 in a network environment with access to a time server, the battery is not essential for correct function of the device and can be removed.

### CAUTION

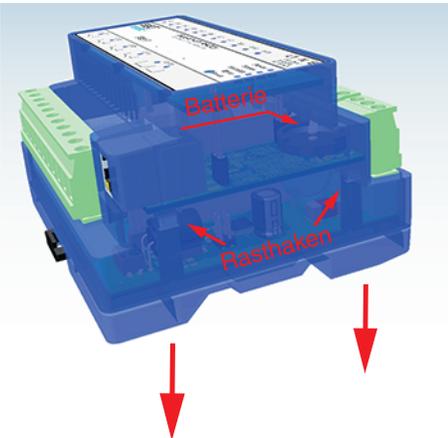
The battery may be removed or replaced by an electrotechnical specialist, only.

To remove the battery, open the housing as follows:

#57733



#57736



Use a pointed object to press on the latching hooks on the side of the housing and at the same time pull the bottom of the housing out of the top shell.

Then remove the stack of PCBs downwards from the housing.

The buffer battery for the clock module is located in a holder on the upper PCB. 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.

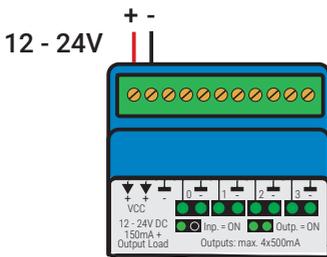
## 3. Quick Startup

### Network connection

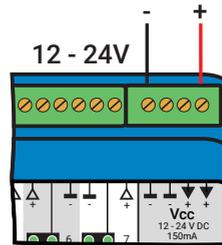


### Supply voltage

#57733



#57736



*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

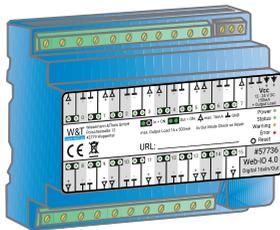
The Web-IO devices differ in their mechanical design and hardware configuration:

#### #57733 - Web-IO 4.0 Digital 4xIn/Out



Network interface:	RJ45 10/100BaseT / PoE
Power:	Screw terminal 12 ... 24V DC
Inputs/Outputs:	4 channels optionally as input or output Input switching threshold +9.5V (+/-1V) Output 12 ... 24V DC current driving max. 500mA

#### #57736 - Web-IO 4.0 Digital 16xIn/Out



Network interface:	RJ45 10/100BaseT
Power:	12 .. 24V DC
Inputs/Outputs:	16 channels optionally as input or output Input switching threshold +9.5V (+/-1V) Output 12 ... 24V DC current driving max. 250mA

Gerätstatus, Fehlerstatus und Status der Inputs/Outputs werden über entsprechende LEDs signalisiert.

## 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.

### Montage #57733

The Web-IO 4.0 Digital 4xIn/Out is intended for installation in the control cabinet or sub-distribution. For mechanical fixation, the Web-IO should be snapped onto a 35mm top-hat rail according to DIN EN 50022. In doing so, the Web IO takes up 45mm in width.

### Montage #57736

The Web-IO 4.0 Digital 4xIn/Out is intended for installation in the control cabinet or sub-distribution. For mechanical fixation, the Web-IO should be snapped onto a 35mm top-hat rail according to DIN EN 50022. In doing so, the Web IO takes up 107mm in width.

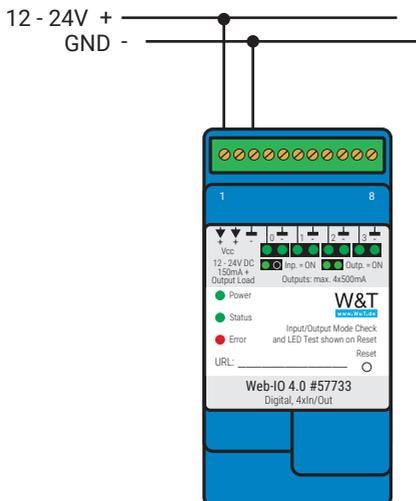
## Terminal assignment 57733

Terminal	Description / Function
1	+ Vcc - Device supply 12-24V 150mA@24V
2	+ Vcc - Device supply 12-24V 150mA@24V
3	GND / Device supply
4	Input 0 Switching threshold +8V / Output 0 24V max. 500mA
5	GND - Output 0
6	Input 1 Switching threshold +8V / Output 1 24V max. 500mA
7	GND - Output 1
8	Input 2 Switching threshold +8V / Output 2 24V max. 500mA
9	GND - Output 2
10	Input 3 Switching threshold +8V / Output 3 24V max. 500mA
11	GND - Output 3

## Wiring #57733

### Connection of the supply voltage $V_{DD}$

Connection of the supply voltage The Web IO is supplied with a DC voltage between 12 and 24V.



With a voltage supply of 24V, which is typical in industry, the Web IO draws approx. 100mA current.

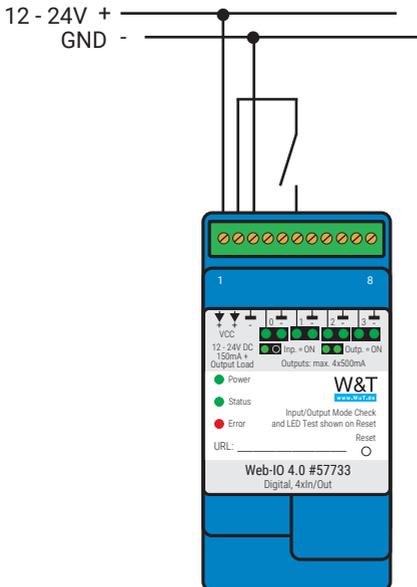
### NOTE

IO channels that are configured as outputs are also supplied via VDD. The loads to be switched must also be taken into account in the current consumption.

## Input wiring

Each of the four IO channels has a plus and a minus terminal.

The signal input for IO channels that are operated as inputs is on the respective plus terminal. When using potential-free contacts, we recommend switching the auxiliary voltage VDD-Out via the contact on the plus terminal. In this case, the minus terminals do not have to be connected. Here is an example of the wiring of input 1.

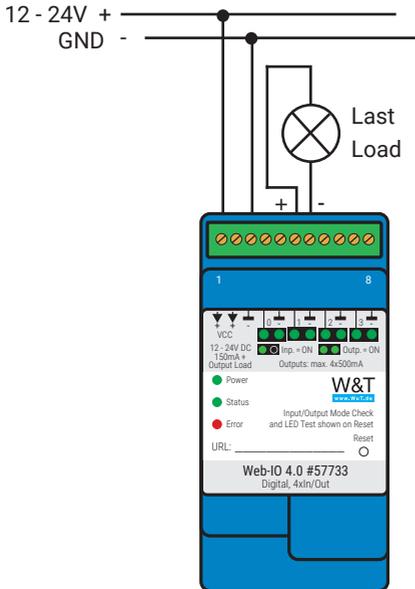


Alternatively, an external voltage of max. +24V DC can be applied to the plus terminal, whereby voltages greater than 8V (+/-1V) are recognised as ON signal. In this case, the reference ground must be applied to the respective minus terminal.

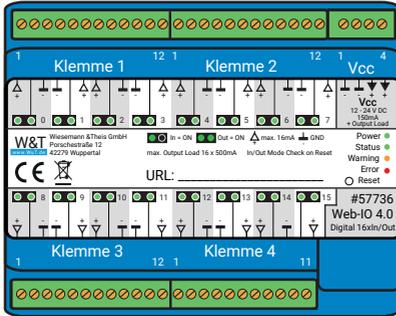
## Output wiring

IO channels that are configured as outputs operate as current drivers and can be loaded with max. 500mA each. The positive voltage applied to VDD .is switched to the positive terminal of the IO channel, whereby the negative terminal is used as reference ground.

Here is an example of the wiring of output 1.



## Terminal assignment #57736



Terminal Vcc	Description / Function
1	GND / Device supply
2	GND / Device supply
3	+ Vcc - Device supply 12-24V max. 2150mA@24V
4	+ Vcc - Device supply 12-24V max. 2150mA@24V

Terminal 1	Bezeichnung / Funktion
1	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
2	Input 0 <i>Switching threshold +8V</i> / Output 0 24V max. 250mA
3	GND / Reference GND for Output
4	GND / Reference GND for Output
5	Input 1 <i>Switching threshold +8V</i> / Output 1 24V max. 250mA
6	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
7	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
8	Input 2 <i>Switching threshold +8V</i> / Output 2 24V max. 250mA
9	GND / Reference GND for Output
10	GND / Reference GND for Output
11	Input 3 <i>Switching threshold +8V</i> / Output 3 24V max. 250mA
12	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation

Terminal 2	Description / Function
1	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
2	Input 4 <i>Switching threshold +8V</i> / Output 4 24V max. 250mA

Terminal 2	Description / Function
3	GND / Reference GND for Output
4	GND / Reference GND for Output
5	Input 5 <i>Switching threshold +8V / Output 5 24V max. 250mA</i>
6	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
7	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
8	Input 6 <i>Switching threshold +8V / Output 6 24V max. 250mA</i>
9	GND / Reference GND for Output
10	GND / Reference GND for Output
11	Input 7 <i>Switching threshold +8V / Output 7 24V max. 250mA</i>
12	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation

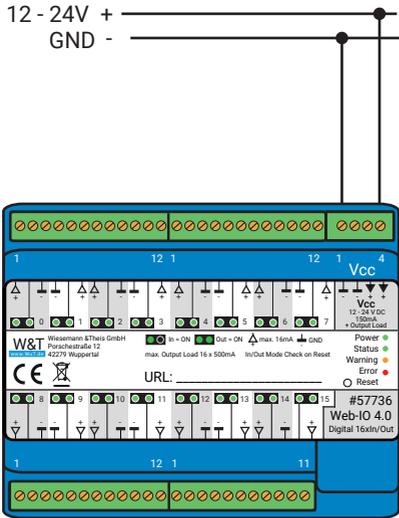
Terminal 3	Description / Function
1	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
2	Input 8 <i>Switching threshold +8V / Output 8 24V max. 250mA</i>
3	GND / Reference GND for Output
4	GND / Reference GND for Output
5	Input 9 <i>Switching threshold +8V / Output 9 24V max. 250mA</i>
6	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
7	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
8	Input 10 <i>Switching threshold +8V / Output 10 24V max. 250mA</i>
9	GND / Reference GND for Output
10	GND / Reference GND for Output
11	Input 11 <i>Switching threshold +8V / Output 11 24V max. 250mA</i>
12	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation

Terminal 4	Description / Function
1	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
2	Input 12 <i>Switching threshold +8V / Output 12 24V max. 250mA</i>
3	GND / Reference GND for Output
4	GND / Reference GND for Output
5	Input 13 <i>Switching threshold +8V / Output 13 24V max. 250mA</i>
6	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
7	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation
8	Input 14 <i>Switching threshold +8V / Output 14 24V max. 250mA</i>
9	GND / Reference GND for Output
10	Input 15 <i>Switching threshold +8V / Output 15 24V max. 250mA</i>
11	+ Vdd - Auxiliary voltage 24V 2mA@24V for Input activation

## Wiring #57736

### Connection of the supply voltage $V_{DD}$

The Web IO is supplied with a DC voltage between 12 and 24V.



With a voltage supply of 24V, which is typical in industry, the Web IO draws approx. 100mA current.

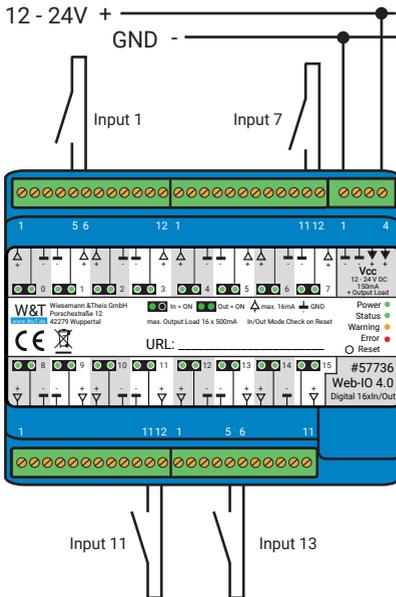
### NOTE

IO channels that are configured as outputs are also supplied via VDD. The loads to be switched must also be taken into account in the current consumption.

## Input wiring

Each of the 16 IO channels has a plus and a minus terminal.

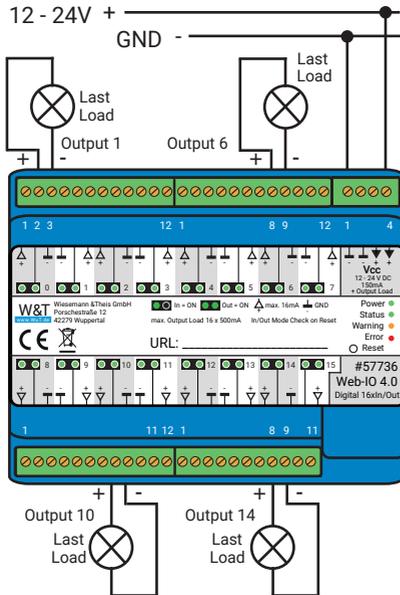
The signal input for IO channels that are operated as inputs is on the respective plus terminal. When using potential-free contacts, we recommend switching the auxiliary voltage VDD-Out via the contact on the plus terminal. In this case, the minus terminals do not have to be connected..



Alternatively, an external voltage of max. +24V DC can be applied to the plus terminal, whereby voltages greater than 8V (+/-1V) are recognised as ON signal. In this case, the reference ground must be applied to the respective minus terminal.

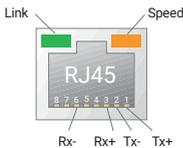
## Output wiring

IO channels that are configured as outputs operate as current drivers and can be loaded with max. 250mA each. The positive voltage applied to VDD .is switched to the positive terminal of the IO channel, whereby the negative terminal is used as reference ground..



## Network connection

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



## 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

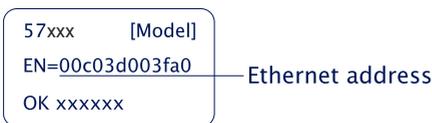
Off = 10MBit/s

### 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. In addition, it must be determined whether the respective IO channel is to work as input or as output.

#### 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.*

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 individual 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 accessed.

### 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](http://www.WuT.de/article%20number), e.g. [www.wut.de/57730](http://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

Placeholder	Description
<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“

## 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-I/O 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
```

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

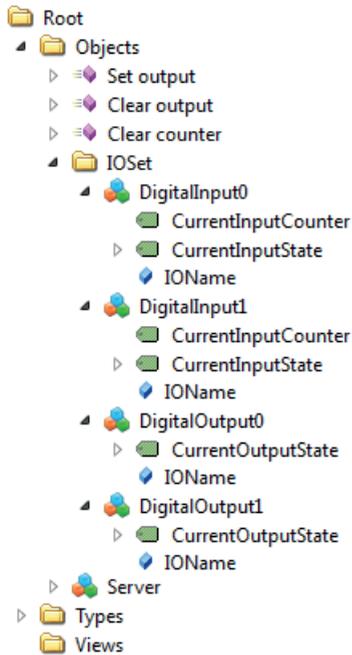
Basic265

Basic265-SHA256

AES128-SHA256 RsaOaep

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

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



Here are the most important variables that can be used to retrieve the states of the IO end points:

- 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)

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 to OFF
- 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.

## Modbus-TCP

The Modbus slave operation for the Web-IO can be activated via the menu item *Communication paths » Modbus-TCP*. Here you can also specify whether the outputs may be switched via Modbus TCP.

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

## Modbus-Memory

### Bit range:

adresse (hexadec.)	description	memory type	length (byte)	read bits with FC	read reg. with FC	Write bits with FC	write reg. with FC
1000	Input 0	bit	1	0x01, 0x02	-	-	-
1001	Input 1	bit	1	0x01, 0x02	-	-	-
1002	Input 2	bit	1	0x01, 0x02	-	-	-
1003	Input 3	bit	1	0x01, 0x02	-	-	-
1004	Input 4	bit	1	0x01, 0x02	-	-	-
1005	Input 5	bit	1	0x01, 0x02	-	-	-
1006	Input 6	bit	1	0x01, 0x02	-	-	-
1007	Input 7	bit	1	0x01, 0x02	-	-	-
1008	Input 8	bit	1	0x01, 0x02	-	-	-
1009	Input 9	bit	1	0x01, 0x02	-	-	-
100A	Input 10	bit	1	0x01, 0x02	-	-	-
100B	Input 11	bit	1	0x01, 0x02	-	-	-
1020	Output 0	bit	1	0x01, 0x02	-	0x05	0x0F
1021	Output 1	bit	1	0x01, 0x02	-	0x05	0x0F
1022	Output 2	bit	1	0x01, 0x02	-	0x05	0x0F
1023	Output 3	bit	1	0x01, 0x02	-	0x05	0x0F
1024	Output 4	bit	1	0x01, 0x02	-	0x05	0x0F
1025	Output 5	bit	1	0x01, 0x02	-	0x05	0x0F
1026	Output 6	bit	1	0x01, 0x02	-	0x05	0x0F
1027	Output 7	bit	1	0x01, 0x02	-	0x05	0x0F
1028	Output 8	bit	1	0x01, 0x02	-	0x05	0x0F
1029	Output 9	bit	1	0x01, 0x02	-	0x05	0x0F
102A	Output 10	bit	1	0x01, 0x02	-	0x05	0x0F
102B	Output 11	bit	1	0x01, 0x02	-	0x05	0x0F
1040	Alarm state 1	bit	1	0x01, 0x02	-	-	-
1041	Alarm state 2	bit	1	0x01, 0x02	-	-	-
1042	Alarm state 3	bit	1	0x01, 0x02	-	-	-
1043	Alarm state 4	bit	1	0x01, 0x02	-	-	-
1044	Alarm state 5	bit	1	0x01, 0x02	-	-	-
1045	Alarm state 6	bit	1	0x01, 0x02	-	-	-
1046	Alarm state 7	bit	1	0x01, 0x02	-	-	-
1047	Alarm state 8	bit	1	0x01, 0x02	-	-	-
1048	Alarm state 9	bit	1	0x01, 0x02	-	-	-
1049	Alarm state 10	bit	1	0x01, 0x02	-	-	-
104A	Alarm state 11	bit	1	0x01, 0x02	-	-	-
104B	Alarm state 12	bit	1	0x01, 0x02	-	-	-
1060	Exception State	bit	1	0x01, 0x02	-	-	-
1068	Config. state	bit	1	0x01, 0x02	-	-	-
1800	Alarm trigger 1	bit	1	0x01, 0x02	-	0x05	0x0F
1801	Alarm trigger 2	bit	1	0x01, 0x02	-	0x05	0x0F
1802	Alarm trigger 3	bit	1	0x01, 0x02	-	0x05	0x0F
1803	Alarm trigger 4	bit	1	0x01, 0x02	-	0x05	0x0F
1804	Alarm trigger 5	bit	1	0x01, 0x02	-	0x05	0x0F
1805	Alarm trigger 6	bit	1	0x01, 0x02	-	0x05	0x0F
1806	Alarm trigger 7	bit	1	0x01, 0x02	-	0x05	0x0F
1807	Alarm trigger 8	bit	1	0x01, 0x02	-	0x05	0x0F
1808	Alarm trigger 9	bit	1	0x01, 0x02	-	0x05	0x0F
1809	Alarm trigger 10	bit	1	0x01, 0x02	-	0x05	0x0F
180A	Alarm trigger 11	bit	1	0x01, 0x02	-	0x05	0x0F
180B	Alarm trigger 12	bit	1	0x01, 0x02	-	0x05	0x0F

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

16- and 32-bit range:

adresse (hexadec.)	description	memory type	length (byte)	read bits with FC	read reg. with FC	Write bits with FC	write reg. with FC
2000	Inputs 0 - 11	16-bit	2	-	0x03, 0x04	-	-
2002	Outputs 0 - 11	16-bit	2	-	0x03, 0x04	-	-
2004	Alarm state 1 - 12	16-bit	2	-	0x03, 0x04	-	-
2006	Diagnosis Error count	16-bit	2	-	0x03, 0x04	-	0x06, 0x10
2007	Diagnostic state 0 - 15	16-bit	2	-	0x03, 0x04	-	-
2008	Diagnostic state 16 - 31	16-bit	2	-	0x03, 0x04	-	-
2009	Diagnostic state 32 - 47	16-bit	2	-	0x03, 0x04	-	-
200A	Diagnostic state 48 - 63	16-bit	2	-	0x03, 0x04	-	-
200B	Diagnostic state 64 - 79	16-bit	2	-	0x03, 0x04	-	-
200C	Diagnostic state 80 - 95	16-bit	2	-	0x03, 0x04	-	-
200D	Exception/Conf.-State	16-bit	2	-	0x03, 0x04	-	-
5000	Inputs 0 - 11	32-bit	4	-	0x03, 0x04	-	-
5002	Outputs 0 - 11	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
5004	Alarm state 1 - 12	32-bit	4	-	0x03, 0x04	-	-
5006	Counter 0	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
5008	Counter 1	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
500A	Counter 2	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
500C	Counter 3	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
500E	Counter 4	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
5010	Counter 5	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
5012	Counter 6	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
5014	Counter 7	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
5016	Counter 8	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
5018	Counter 9	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
501A	Counter 10	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
501C	Counter 11	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
504A	Diagnosis Error count	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
504C	Diagnostic state 0 - 31	32-bit	4	-	0x03, 0x04	-	-
504E	Diagnostic state 32 - 63	32-bit	4	-	0x03, 0x04	-	-
5050	Diagnostic state 64 - 95	32-bit	4	-	0x03, 0x04	-	-
7000	virtuel Register 0	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7002	virtuel Register 1	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7004	virtuel Register 2	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7006	virtuel Register 3	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7008	virtuel Register 4	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
700A	virtuel Register 5	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
700C	virtuel Register 6	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
700E	virtuel Register 7	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7010	virtuel Register 8	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
.....	virtuel Register 9 - 23	32-bit	4	-	-	-	-
702E	virtuel Register 23	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7030	virtuel Register 24	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7032	virtuel Register 25	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7034	virtuel Register 26	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7036	virtuel Register 27	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
7038	virtuel Register 28	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
703A	virtuel Register 29	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
703C	virtuel Register 30	32-bit	4	-	0x03, 0x04	-	0x06, 0x10
703E	virtuel Register 31	32-bit	4	-	0x03, 0x04	-	0x06, 0x10

A 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)
- specifies a range of from...to (e.g. weekday "2-4" stands for Tuesday to Thursday, whereas entering "\*" triggers the timer on all weekdays).
- / 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“

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 requestsAJAX

### 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 manual link on the data sheet page of your Web-IO.*

## 12. Appendix

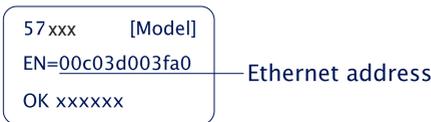
### Alternatives for IP address assignment

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 #57734.

- 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](http://www.WuT.de/article) number, e.g. [www.wut.de/57730](http://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.

## 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.

### Delete passwords

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

## 13. Technical data

### #57733

<b>Connections, displays and control elements:</b>	
Digital IOs:	4 channels selectable input/output operation As output: 12-24V DC / 500mA per channel Total fuse protection for all outputs 3A As input: max. 30V DC Switching threshold 9.5V +/- 1V „On“ current = 2.2 mA Integrated 32-bit pulse counter
Network:	10/100BaseT autosensing
Power supply:	12 ... 24V DC (approx. 100mA@24V)
Output supply	12 ... 24V DC
Galvanische Trennung:	Digital IOs - network: min. 1000 V
Conections:	1 x 11x screw terminal block for IOs and power 1 x RJ45 for network
Displays:	Status LEDs for network Error LEDs for system and application 8 LEDs for digital status
<b>Data transfer:</b>	
Protocols:	TCP- und UDP-Sockets, Client und Server SNMP inkl. Traps, SMTP E-Mail-Versand OPC-DA, OPC UA, Modbus-TCP, Inventarisierung
Response times:	Datatrafic: typ. 40 - 80ms
<b>Housing and other data:</b>	
Housing:	Plastic housing 90 x 45 x 56 mm (lxbxh)
Enclosure rating:	IP20
Weight:	ca. 140 g
Storage temperature:	-25°C .. 70°C
Operating temperature:	0°C .. 50°C
Permissible relative humidity:	5..95% RH (non-condensing)

## #57736

Connections, displays and control elements:	
Digital IOs:	16 channels selectable input/output operation As output: 12-24V DC / 250mA per channel Total fuse protection for all outputs 3A As input: max. 30V DC Switching threshold 9.5V +/- 1V „On“ current = 2.2 mA Integrated 32-bit pulse counter
Network:	10/100BaseT autosensing
Power supply:	12 ... 24V DC (approx. 100mA@24V)
Output supply	12 ... 24V DC
Galvanische Trennung:	Digital IOs - network: min. 1000 V
Conections:	3 x 12x screw terminal block for IOs 1 x 11x screw terminal block for IOs 1 x 4x screw terminal block for power 1 x RJ45 für Netzwerk
Displays:	Status LEDs for network Error LEDs for system and application 8 LEDs for digital status
Data transfer:	
Protocols:	TCP- und UDP-Sockets, Client und Server SNMP inkl. Traps, SMTP E-Mail-Versand OPC-DA, OPC UA, Modbus-TCP, Inventarisierung
Response times:	Datatraffic: typ. 40 - 80ms
Housing and other data:	
Housing:	Plastic housing 90x116x56 mm (lxwxh)
Enclosure rating:	IP20
Weight:	ca. 250 g
Storage temperature:	-25°C .. 70°C
Operating temperature:	0°C .. 50°C
Permissible relative humidity:	5..95% RH (non-condensing)



Wiesemann & Theis GmbH  
Porschestraße 12  
D-42279 Wuppertal

Mail [info@wut.de](mailto:info@wut.de)  
Web [www.wut.de](http://www.wut.de)

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