

# **ZIGBEE SHUTTER CONTROL J1(-R)**

## REFERENCE MANUAL



Thank you for purchasing the ubisys shutter control J1, a ZigBee window covering adaptor with integrated smart meter for bidirectional, single-phase AC motors.

You have decided for a high-quality product with first-rate support!

This reference manual provides operating and maintenance instructions, interface specifications, command references and more. It is primarily intended for system integrators, not end-users.

An installation guide specific to your region is available as a separate document and is included in printed form in the product package.

If you have any questions or need additional support, please visit the support pages that best fit your background:

If you are a consumer (private household) or installer, please visit the Smart Home support pages at <http://www.ubisys.de/en/smarthome/support.html> for contact details.

As a commercial customer, please visit the Engineering support pages at <http://www.ubisys.de/en/engineering/support.html> for contact details.

## 2. Contents

1. Overview	2
2. Contents	3
3. Features	5
4. Installation	6
4.1. Mains Powered Operation	6
4.2. Low-Voltage Operation (Maintenance and Early System Integration Stages only)	6
5. Initial Device Start-up	7
6. Man-Machine Interface (MMI)	8
6.1. Power-Cycle Sequencing Factory Reset	9
7. ZigBee Interface	10
7.1. Application Endpoint #0 – ZigBee Device Object	12
7.2. Application Endpoint #1 – Window Covering Device	13
7.2.1. Basic Cluster (Server)	13
7.2.2. Identify Cluster (Server)	13
7.2.3. Groups Cluster (Server)	14
7.2.4. Scenes Cluster (Server)	14
7.2.5. Window Covering Cluster (Server)	15
7.3. Application Endpoint #2 – Window Covering Controller	21
7.3.1. Basic Cluster (Server)	21
7.3.2. Identify Cluster (Server)	21
7.3.3. Scenes Cluster (Client)	21
7.3.4. Window Covering Cluster (Client)	22
7.4. Application Endpoint #3 – Metering	23
7.4.1. Basic Cluster (Server)	23
7.4.2. Metering Cluster (Server)	23
7.4.3. Electrical Measurement Cluster (Server)	23
7.5. Application Endpoint #232 – Device Management	25
7.5.1. Basic Cluster (Server)	25
7.5.2. Identify Cluster (Client)	26
7.5.3. Commissioning Cluster (Server)	26
7.5.4. Over-the-Air Upgrade Cluster (Client)	26
7.5.5. Device Setup Cluster (Server)	26

7.6.	Application Endpoint #242 – ZigBee Green Power	32
7.6.1.	Green Power Cluster (Server)	32
7.6.2.	Green Power Cluster (Client)	33
8.	Physical Dimensions	35
9.	Ordering Information	36
10.	General Terms & Conditions of Business	37
11.	Declaration of Conformity	38
12.	Revision History	39
13.	Contact	40

### 3. Features

- ZigBee 3.0 Certified AC shutter control with integrated smart meter and ZigBee router functionality
- Controls single-phase, bidirectional AC motors at 230V~, up to 500VA
- Employs ubisys WaveStep™ technology for advanced positioning, e.g. “go to 50% lift, 45° tilt angle”
- Two configurable 230V~ inputs, pre-configured for operating the locally attached motor. Individually reconfigurable as window covering controller and scene selector switches for use with momentary or stationary switches
- Local control even works when the device is not joined to a network (not commissioned yet)
- Supports groups, scenes, bindings, and reporting
- Reliable, unattended, maintenance-free operation;  
J1 designed for in-wall flush-mounting,  
J1-R designed for DIN rail mounting (fuse cabinet)
- Employs solid state switching for highest efficiency and durability
- Supports blinds, shutters, windows, flaps, etc.
- Standard ZigBee Window Covering Cluster with manufacturer-specific extensions (for calibration only)
- Made in Germany using high-quality, enduring parts for many years of life expectancy
- Low power dissipation: 0.3W
- On-board MCU: Advanced 32-bit ARM micro-controller running at 48MHz with 64KB SRAM
- On-board PHY: Texas Instruments CC2520, 5dBm transmit power, -98dBm receiver sensitivity
- On-board inverted-F antenna
- Leverages ubisys IEEE 802.15.4 MAC and ZigBee stack for best-in-class reliability and performance
- Supports all channels in the 2.4 GHz band, i.e. channels 11-26 as per IEEE 802.15.4:  
Primary = { 11, 15, 20, 25 }; Secondary = { 12, 13, 14, 16, 17, 18, 19, 21, 22, 23, 24, 26 }
- Supports joining centralized and distributed security networks as router
- Supports forming simple centralized security networks as Coordinator and Trust Center
- Supports forming distributed security networks as router
- Three pre-configured Trust Center Link-Keys for joining:
  - o Global Default Trust Center Link-Key (“ZigBeeAlliance09”)
  - o ZigBee 3.0 Global Distributed Security Link-Key<sup>1</sup>
  - o Device-individual link-key derived from installation code – also printed as text and QR barcode
- Extended neighbour table with up to 78 entries for routers and end-devices -  
more than three times the capacity required by the standard (25)
- Extended routing table with up to 96 entries for ad hoc and many-to-one routes -  
nearly ten times the capacity required by the standard (10)
- Extended buffering for sleeping end-devices with up to 24 buffers -  
24 times the capacity required by the standard (1)
- Extended APS duplicate rejection table with up to 64 slots -  
64 times the capacity required by the standard (1)
- Extensive transmit and receive queues for optimum through-put and minimum packet drop rate
- Reliable and scalable network-wide broadcasts featuring passive acknowledgments
- Reliable packet forwarding with automatic network-level retries
- Very sophisticated routing algorithm for reliable ad hoc routing – avoids routing loops even in case of concurrent route requests with overlapping source/destination
- Firmware upgradable over-the-air during normal operation using ZigBee OTA Upgrade Cluster
- Man-Machine-Interface: A push-button and a LED for network steering, factory reset etc.
- Flame retardant housing (V-0); black, RAL 9005

---

<sup>1</sup> Since application firmware 1.16. Prior, pre-certification key (D0:...:DF)

### 4.1. Mains Powered Operation

Please refer to the hardware installation guide, which was included in the product package. It provides detailed installation instructions.

### 4.2. Low-Voltage Operation (Maintenance and Early System Integration Stages only)

Instead of applying dangerous mains voltage to the shutter control J1, you may connect a low-voltage DC source (e.g. 12V=, 24V=, 48V=) to the supply lines of J1. Its built-in universal power supply is able to operate from both kinds of supplies, direct and alternating current. This can be useful during maintenance or early stages of system integration. In this mode, the ZigBee interface is operational and you can run OTA firmware upgrades, for example. However, the inputs and outputs are non-operational and attempting to switch the output can result in a reboot, because the device detects a malfunction.

For DC operation (laboratory use), connect the DC ground (0V, “minus”, negative) of your power supply to the phase input of J1 (marked “L”, brown) and the DC supply voltage (12...48VDC, “plus”, positive) to the neutral input of J1 (marked “N”, blue). Once powered, you should observe the normal device start-up behaviour (the green LED will flash a few times).

If you are not using a safe low-voltage power-supply, utilizing the services of a professional installer/electrician is imperative.

## 5. Initial Device Start-up

The device will search for an open ZigBee network to join when you first apply power to it.

Open the ZigBee network for joining and then power-up the device.

J1 will blink quickly to indicate a search in progress. Once a search cycle is complete it will either blink five times slowly to indicate it has joined a network, or blink three times quickly to indicate a joining failure – for example no network found, not permitted onto the network, etc. It will continue to search for a network in case of failure.

Subsequently, when the device is power-cycled, it will blink five times slowly to indicate it is operating as a router on the network, ten times slowly to indicate that it is operating as a coordinator and trust center or blink quickly to indicate it is searching for a network to join. When commissioned, the router's LED will remain off after the five/ten blink cycles during normal operation. Afterwards, it will turn on only for as long as it is permitting other devices to join the network.

After reboot, if the device is properly commissioned, it resumes operation doing a “silent rejoin”, i.e. it does not broadcast a device announcement in this case. After joining a network, the device will prolong the joining window by three minutes via a fresh ZDO permit joining request<sup>2</sup>.

Notice: When the device is power-cycled, it does not have knowledge of the shutter's current position. It has to reach the top or bottom limit once after a power-cycle to learn the current position; then it is able to maintain positioning using ubisys' unique WaveStep™ technology. For safety reasons the device will not start moving after a power-cycle in order to seek the top or bottom position. Instead, it will wait for any command that instigates motion, and if necessary, will try to find the upper bound as a reference point first, before moving to the requested target position.

For positioning using lift and tilt values, the device must be calibrated as part of the commissioning process. Otherwise only move up/down and stop commands are available.

---

<sup>2</sup> Stretching the joining duration is done in products with ubisys ZigBee Stack version 1.48 or above.

J1 offers a push-button behind a tiny hole in the front-face of the device, as well as a LED right next to it<sup>3</sup>. This man-machine interface provides access to a menu. In addition, it provides a handy shortcut to factory reset the device<sup>4</sup>.

**To factory reset the device, keep the button pressed for approximately 10 seconds until the LED starts to flash.** This is equivalent to selecting menu item #5.

Notice that the device also supports a power-cycle sequencing factory reset as detailed in section 6.1.

To enter the menu, press and hold the push-button for more than a second until you see three short flashes followed by a sequence of one blink, pause, one blink, pause, etc. This indicates that you have successfully entered the menu. With each short button press (less than a second), you advance through the menu. For example, pressing the button once brings you to the second menu item, which J1 indicates by two blinks, pause, two blinks pause, etc. Once you have reached the menu item that you want to run, press and hold the button for more than a second to execute the selected item. The exact meaning of each menu item depends on the particular device and may vary from firmware version to firmware version.

This is the menu for Shutter Control J1 running firmware version 1.07 and above<sup>5</sup>:

Menu Item #	Operation
1	<b>ZigBee Commissioning: Network Steering</b> While in this mode a single press on the button instigates ZigBee Network Steering ("EZ-mode"). If the device is on a network it will open the network for new devices for three minutes, otherwise it will attempt to join an open network. If the network is already open, the device will close the network instead such that repetitive presses toggle the network permit joining state. The LED is on, when the network is open for new devices, and off otherwise.
2	<b>ZigBee Commissioning: Finding &amp; Binding</b> Instigates the ZigBee Finding & Binding procedure ("EZ-mode") on an initiator or target endpoint. Target endpoints will enter identify mode for three minutes. Initiator endpoints will query targets and create suitable bindings to these targets. After leaving the menu in this mode, the first button press starts the selection of the endpoint and each subsequent press (within half a second) increments the endpoint number. Once the desired endpoint is reached, wait for the LED to blink the selected number of times. Then, press a single time to accept the selection or wait for three seconds to cancel the command. The LED will blink one time to confirm an affirmative choice, or two times to indicate the request has been cancelled.
3	<b>ZigBee Commissioning: Clear Bindings</b> Clears bindings on an initiator endpoint. Select this menu item and leave menu mode. Afterwards the device expects the selection of an initiator endpoint. The first button press starts the selection of the endpoint and each subsequent press (within half a second) increments the endpoint number. Once the desired endpoint is reached, wait for the LED to blink the selected number of times. Then, press a single time to accept the selection or wait for three seconds to cancel the command. The LED will blink one time to confirm an affirmative choice, or two times to indicate the request has been cancelled. This mode is active for one round only. Afterwards the device reverts to mode #1.
4	<b>ZigBee Commissioning: Set Device Role and Factory Reset</b> Selects the ZigBee device role for this device on the network, resets the remaining settings to factory defaults and restarts the device. Select this menu item and leave menu mode. Afterwards the device expects the selection of an option. The first button press starts the selection of the option and each

<sup>3</sup> The high-voltage inputs of J1/J1-R do not count as an MMI feature.

<sup>4</sup> The 10-second-press factory reset (equivalent to menu item #5) is available since firmware version 1.10.

<sup>5</sup> Notice: Prior versions have different menu items. Only the factory reset command (menu item #5) is identical across all released versions of the firmware.



	subsequent press (within half a second) increments the option number. Once the desired option is reached, wait for the LED to blink the selected number of times. Then, press a single time to accept the selection or wait for three seconds to cancel the command. The LED will blink one time to confirm an affirmative choice, or two times to indicate the request has been cancelled. The following options are available: Option #1: Join an existing ZigBee network as router Option #2: Form a new distributed security network as the first router Option #3: Form a new centralized security network as the coordinator and trust center
5	<b>Factory Reset</b> Put the device into the same state as it was when it left the factory, then reboot. The only exception is the outgoing network security frame counter, which is preserved across factory resets. The device will broadcast a network leave indication. <sup>6</sup>
6	<b>ZigBee Commissioning: Advanced Commands</b> Provides a number of advanced ZigBee commissioning command options. Select this menu item and leave menu mode. Afterwards the device expects the selection of an option. The first button press starts the selection of the option and each subsequent press (within half a second) increments the option number. Once the desired option is reached, wait for the LED to blink the selected number of times. Then, press a single time to accept the selection or wait for three seconds to cancel the command. The LED will blink one time to confirm an affirmative choice, or two times to indicate the request has been cancelled. The following options are available: Option #1: Perform a simple reset (reboot), then continue operating seamlessly (silent re-join) Option #2: Perform a simple reset (reboot), then re-join the network Option #3: Perform a full factory reset, including security frame counters and preserved settings
7	Reserved for internal use. Do not leave the menu in this mode.

## 6.1. Power-Cycle Sequencing Factory Reset

It is possible to instigate a factory reset using a special power-cycle sequence<sup>7</sup>. This is equivalent to selecting menu item #5, with the advantage that you need no access to the device itself (only to its power supply).

1. Power the device for at least four seconds.
2. Interrupt the power supply for at least a second.
3. Reapply power for less than two seconds but more than half a second.  
Notice that at the end of this cycle, the device is off and should remain off for at least a second.
4. Repeat the previous step two more times, for a total of three short power cycles.
5. Apply power to the device and leave it powered on.

The device will now factory reset and reboot.

<sup>6</sup> Since application firmware 1.11. Previously, the factory reset would result in a silent leave.

<sup>7</sup> Since application firmware 1.17

Please refer to the following IEEE and ZigBee Alliance documents, which apply to this product:

- [R1] IEEE Standard 802 – Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)
- [R2] ZigBee Specification, Revision 21, Document No. 05-3474-21
- [R3] ZigBee 2015 Layer PICS and Stack Profiles, Revision 6, Document No. 08-0006-06
- [R4] ZigBee Cluster Library Specification, Revision 5, Document No. 07-5123-05
- [R5] ZigBee Home Automation Public Application Profile 1.2, Revision 29, Document No. 05-3520-29
- [R6] ZigBee Smart Energy Standard 1.1b, Revision 18, Document No. 07-5356-18

### Device Anatomy

The shutter control J1 controls roller shades, lift & tilt blinds, windows, flaps, projector screens or other “window coverings”, which are controlled by a single-phase, bidirectional AC motor at rated voltage of 230V~/50Hz and an apparent power of up 500VA, which allows utilizing it for typical use cases in EMEA countries. It combines state-of-the-art solid-state technology with advanced firmware for a unique feature set. It also provides an integrated smart meter, which is capable of measuring active power, reactive power, apparent power, voltage, current and frequency<sup>8</sup>. In addition two high-voltage inputs are available for local control or remote control of other window covering target devices or groups. Being a mains-powered device, J1(-R) also acts as a ZigBee router improving network connectivity and meshing capabilities as well as serving as a connection point (“parent”) for sleeping end-devices.

J1 and J1-R have an identical ZigBee interface. They provide six ZigBee application endpoints:

Endpoint #	Profile	Application
0 (0x00)	0x0000: ZigBee Device Profile	ZigBee Device Object (ZDO) – standard management features
1 (0x01)	0x0104: Common Profile (HA)	Window Covering Device (0x0202). This endpoint provides control of the motor via the window covering cluster. It supports groups and scenes, as well as reporting for current lift and tilt position, as well as operational state (draft ZCL revision 6) feedback. The identify cluster allows for push-button commissioning as a target.
2 (0x02)	0x0104: Common Profile (HA)	Window Covering Controller (0x0203). This endpoint is used to transmit window covering commands triggered by any of the local high-voltage inputs.
3 (0x03)	0x0104: Common Profile (HA) <sup>9</sup>	Metering (0x0702). This endpoint provides metering and electrical measurement clusters to measure power, energy, voltage, current, and other figures.
200 (0xC8)	Private	Private. This endpoint contains a legacy private application and provided for backwards-compatibility, only. This feature is deprecated and will be removed in a future firmware release after a grace period.
232 (0xE8)	0x0104: Common Profile (HA)	Device Management (0x0507) <sup>10</sup>
242 (0xF2)	0xA1E0: Green Power Profile	ZigBee Green Power Combined Proxy and Sink <sup>11</sup>

<sup>8</sup> More advanced features like waveform sampling, total harmonic distortion and other quality figures available upon request

<sup>9</sup> Since application version 1.06, respectively. Prior, profile ID 0x0109 (Smart Energy) was used.

<sup>10</sup> Since application version 1.06, respectively. Prior, device ID 0x0008 (Range Extender) was used.

<sup>11</sup> Since application version 1.12.

The ubisys ZigBee manufacturer ID is 0x10F2. This manufacturer code is used to identify OTA upgrade images, for accessing manufacturer-specific ZCL attributes and commands etc.

### Installation Code

This shutter control has a pre-configured link key, which is derived from the installation code printed on the back of the control's housing in text format and as a two-dimensional barcode (QR code). The format specified in [R6], section 5.4.8.1.1 is used with a full 128-bit installation code + 16-bit CRC. The QR code contains additional information as illustrated in the following example:

```
ubisys2/R0/001FEE00000000FF/0F7C1CD805F91649EBA84580AA1CB432F51A/21
```

Here, "ubisys2" is the data format identifier, R0 is the model string (this is just an example, it would be "J1" or "J1-R" for this product), followed by the EUI-64, the installation code, and a checksum that covers the entire information (including model, EUI-64 and installation code), all separated by a single dash character ('/'). The check sum is an unsigned 8-bit integer, which is calculated by performing a byte-wise exclusive-or (XOR, " $\oplus$ ") of the ASCII characters of the model string, the binary representation of the EUI-64 (in big endian format), and the binary representation of the install code.

For the example above, this calculation yields:

```
52 ('R')  $\oplus$  30 ('0')  $\oplus$   
00  $\oplus$  1F  $\oplus$  EE  $\oplus$  00  $\oplus$  00  $\oplus$  00  $\oplus$  00  $\oplus$  FF  $\oplus$   
0F  $\oplus$  7C  $\oplus$  1C  $\oplus$  D8  $\oplus$  05  $\oplus$  F9  $\oplus$  16  $\oplus$  49  $\oplus$  EB  $\oplus$  A8  $\oplus$  45  $\oplus$  80  $\oplus$  AA  $\oplus$  1C  $\oplus$  B4  $\oplus$  32  $\oplus$  F5  $\oplus$  1A = 21
```

Notice: The data format "ubisys1" is identical to "ubisys2" except for the check sum, which is present, but invalid. If you encounter "ubisys1" labels, then do not verify the trailing check sum field.

## 7.1. Application Endpoint #0 – ZigBee Device Object

Please refer to the ZigBee Specification [R2] for details on the ZigBee Device Object (ZDO) and the protocol used for over-the-air communication, called the ZigBee Device Profile (ZDP). Notice that the ZDP is fundamentally different from typical application endpoints, which build on the ZigBee foundation framework and the ZigBee Cluster Library (ZCL).

The ubisys ZigBee shutter control J1/J1-R supports the following ZDO services:

Primitive	Description
nwk_addr_req/ nwk_addr_rsp	<b>Network address request/response</b> Translates a 64-bit IEEE address into a 16-bit network short address. Use only when really required, because this message employs a network-wide broadcast (flooding)
ieee_addr_req/ ieee_addr_rsp	<b>IEEE address request/response</b> Translates a 16-bit network short address into a 64-bit IEEE address.
node_desc_req/ node_desc_rsp	<b>Node descriptor request/response</b> Returns information such as the manufacturer ID, power supply, etc.
power_desc_req/ power_desc_rsp <sup>12</sup>	<b>Power descriptor request/response</b> Returns information such as the power source and mode.
active_ep_req/ active_ep_rsp	<b>Active endpoints request/response</b> Returns a set of available application endpoints on the device.
simple_desc_req/ simple_desc_rsp	<b>Simple descriptor request/response</b> Returns a descriptor for a certain application endpoint with a list of available services (clusters).
match_desc_req/ match_desc_rsp	<b>Match descriptor request/response</b> Searches for a certain cluster or set of clusters and returns the matching endpoints, if any.
device_annce	<b>Device announcement</b> Advertises the presence of a new device in the network.
parent_annce/ parent_annce_rsp <sup>13</sup>	<b>Parent announcement/response</b> This is part of the ZigBee 2015 end-device child management feature.
system_server_discovery_req/ system_server_discovery_rsp <sup>14</sup>	<b>System server discovery request/response</b> Provides the ability to discover system servers, in particular the network manager.
bind_req/ bind_rsp	<b>Bind request/response</b> Creates an application binding
unbind_req/ unbind_rsp	<b>Unbind request/response</b> Removes an application binding
mgmt_nwk_disc_req/ mgmt_nwk_disc_rsp <sup>15</sup>	<b>Management: Network discovery request/response</b> Instructs the device to perform a network discovery and report the results back.
mgmt_lqi_req/ mgmt_lqi_rsp	<b>Management: Neighbor table request/response</b> Returns information about neighboring devices, including the link quality, device type etc.
mgmt_rtg_req/ mgmt_rtg_rsp	<b>Management: Routing table request/response</b> Returns information about routes established on the device.
mgmt_bind_req/ mgmt_bind_rsp	<b>Management: Binding table request/response</b> Returns information about application bindings on the device.
mgmt_leave_req/ mgmt_leave_rsp	<b>Management: Leave request/response</b> Makes the device leave the network or removes one of its end-device children.
mgmt_permit_joining_req/ mgmt_permit_joining_rsp	<b>Management: Permit joining request/response</b> Opens the network for new devices to join.
mgmt_nwk_update_req/ mgmt_nwk_update_notify <sup>16</sup>	<b>Management: Network update request/response/notification</b> Performs energy scans, instigates a channel change or assigns the network manager.

<sup>12</sup> Available in ZigBee stack version 1.60 and above. Legacy ZCP requirement – do not use in applications

<sup>13</sup> Available in ZigBee stack version 1.56 and above.

<sup>14</sup> Available in ZigBee stack version 1.50 and above.

<sup>15</sup> Available in ZigBee stack version 1.61 and above.

<sup>16</sup> Available in ZigBee stack version 1.61 and above.

## 7.2. Application Endpoint #1 – Window Covering Device

Use this application endpoint to control the attached motor via the load outputs, which are marked with a wave crossed by an arrow and an up/down tag. In case of J1, these are the red (upwards, primary) and black (downwards, secondary) wires.

This is a Finding & Binding target endpoint.

The application endpoint exposes the following clusters:

Cluster	Direction	Description
0x0000	Inbound (Server)	<b>Basic</b> Provides basic information about the device, such as the manufacturer ID, vendor and model name, stack profile, ZCL version, production date, hardware revision etc. Allows a factory reset of attributes, without the device leaving the network.
0x0003	Inbound (Server)	<b>Identify</b> Allows to put the endpoint into identify mode. Useful for identifying/locating devices and required for Finding & Binding.
0x0004	Inbound (Server)	<b>Groups</b> Allows adding this endpoint to one or more groups. Afterwards the endpoint can be addressed using the group address. This is also a prerequisite for scenes. You may also query group membership and delete group associations.
0x0005	Inbound (Server)	<b>Scenes</b> Allows storing one or more scenes per group, where each scene consists of a pre-set on/off state value. You may either store the current values as a scene, or provide scene settings when adding a scene, or delete scenes.
0x0102	Inbound (Server)	<b>Window Covering</b> Provides up/down/stop and go to lift/tilt position control of the attached blind/screen/window.

### 7.2.1. Basic Cluster (Server)

Please refer to section 7.5.1 for details.

### 7.2.2. Identify Cluster (Server)

In identify mode, the up and down directions will be activated alternatingly once every second.

Caution: Make sure the attached motor and mechanics can handle this switching rate or physically disconnect the load!

Attributes supported:

Attribute	Type	Description
0x0000	unsigned16	<b>IdentifyTime</b> The remaining time, in seconds, the device will be identifying itself. Notice that J1 will move the connected blind up/down a little in identify mode.

Cluster commands supported:

Command	Description
---------	-------------

0x00	<b>Identify</b> Puts the device into identify mode, or terminates identify mode.
0x01	<b>Query Identify</b> Determines whether the device is currently identifying, and returns the remaining time if so.

### 7.2.3. Groups Cluster (Server)

Attributes supported:

Attribute	Type	Description
0x0000	bitmap8, read-only	<b>NameSupport</b> Always set to 0: This implementation does not support storing names for groups.

Cluster commands supported:

Command	Description
0x00	<b>Add Group</b> Adds the endpoint to a group.
0x01	<b>View Group</b> Determines whether the device belongs to a group and returns the group name, if supported.
0x02	<b>Get Group Membership</b> Returns the set of groups this endpoint belongs to.
0x03	<b>Remove Group</b> Removes this endpoint from the specified group. Also removes all scenes that refer to this group.
0x04	<b>Remove All Groups</b> Removes this endpoint from all groups. Also removes all scenes that refer to any of the existing groups.
0x05	<b>Add Group if Identifying</b> Adds this endpoint to the group, if the endpoint is identifying.

### 7.2.4. Scenes Cluster (Server)

Notice: For the scenes cluster to work as expected, the device needs to be calibrated first. Until calibration is complete, only limited scene support is available. In particular, a non-zero lift value will cause the shutter to be fully closed, while a lift percentage of 0% will cause the shutter to be fully opened. When storing a scene, the device considers the direction of the last move command. If the device was moving upwards, it will store 0% as the scene target value for lift; conversely, if it was moving downwards it will store 100% as the scene target value.

Attributes supported:

Attribute	Type	Description
0x0000	unsigned8, read-only	<b>SceneCount</b> Holds the total number of scenes (across all groups) currently stored on the device.
0x0001	unsigned8, read-only	<b>CurrentScene</b> If the SceneValid attribute is true, this attribute, together with the CurrentGroup attribute, indicates the currently active scene.
0x0002	unsigned8, read-only	<b>CurrentGroup</b> If the SceneValid attribute is true, this attribute, together with the CurrentScene attribute, indicates the currently active scene.

0x0003	bool, read-only	<b>SceneValid</b> If true, the scene identified by CurrentGroup and CurrentScene is currently active, i.e. all device attribute values match the values in the scene field set.
0x0004	bitmap8, read-only	<b>NameSupport</b> Always set to 1: This implementation does support storing names for scenes.

Cluster commands supported:

Command	Description
0x00	<b>Add Scene</b> Adds a scene with or without a scene field set
0x01	<b>View Scene</b> Returns the scene field set, name and transition times for a scene.
0x02	<b>Remove Scene</b> Removes a scene from the scene table.
0x03	<b>Remove All Scenes</b> Removes all scenes that belong to a particular group.
0x04	<b>Store Scene</b> Stores the device's current state as a scene or updates a previously stored scene accordingly.
0x05	<b>Recall Scene</b> Reverts the device's current state using the values from the previously stored field set.
0x06	<b>Get Scene Membership</b> Returns the set of scenes (within the scope of the specified group) currently stored on the device.
0x40	<b>Enhanced Add Scene</b> Similar to Add Scene with higher transition time resolution (10ths of a second instead of once second)
0x41	<b>Enhanced View Scene</b> Similar to View Scene with higher transition time resolution (10ths of a second instead of once second)
0x42	<b>Copy Scene</b> Copies a scene in a single operation as an alternative to (Enhanced) View Scene, (Enhanced) Add Scene

### 7.2.5. Window Covering Cluster (Server)

Notice: This cluster uses the binding table for managing reporting targets. Currently, there is no default reporting configuration in place, so you need to configure reporting, if you want to receive feedback on lift & tilt position and operational status changes.

Standard attributes supported:

Attribute	Type	Description
0x0000	enum8, read-only	<b>WindowCoveringType</b> Specifies the type of device attached to the motor control output wires, for example: roller blind, lift & tilt blind, awning, flap, projector screen etc.
0x0001	unsigned16, read-only	<b>PhysicalClosedLimitLift</b> The bound for the fully closed position (lift height), in centimeters.
0x0002	unsigned16, read-only	<b>PhysicalClosedLimitTilt</b> The bound for the fully closed position (tilt angle), in units of 0.1°.
0x0003	unsigned16, read-only, reportable	<b>CurrentPositionLift</b> The current lift position (height), in centimeters.
0x0004	unsigned16, read-only, reportable	<b>CurrentPositionTilt</b> The current tilt position (angle), in units of 0.1°.
0x0005	unsigned16, read-only	<b>NumberOfActuationsLift</b> Notice: This attribute is present, but currently not maintained.

0x0006	unsigned16, read-only	<b>NumberOfActuationsTilt</b> Notice: This attribute is present, but currently not maintained.
0x0007	bitmap8, read-only	<b>ConfigurationAndStatus</b> Provides information about the status and capabilities of the window covering, e.g. if it supports positioning feedback via timer or encoder, etc.
0x0008	unsigned8, read-only, reportable	<b>CurrentPositionLiftPercentage</b> The current lift position (height), in percent. Where 0% corresponds to the open position (c.f. InstalledOpenLimitLift attribute), 100% to the closed position (c.f. InstalledClosedLimitLift attribute).
0x0009	unsigned8, read-only, reportable	<b>CurrentPositionTiltPercentage</b> The current tilt position (angle), in percent. Where 0% corresponds to the open position (c.f. InstalledOpenLimitTilt attribute), 100% to the closed position (c.f. InstalledClosedLimitTilt attribute).
0x000A	bitmap8, read-only, reportable	<b>OperationalStatus</b> This attribute contains two bits which will be set while the motor is active. Thus, devices that do not support positioning or move at a slow pace can still provide feedback.
0x0010	unsigned16, read-only <sup>17</sup>	<b>InstalledOpenLimitLift</b> Specifies a bound for the top position (lift height), in centimeters.
0x0011	unsigned16, read-only <sup>6</sup>	<b>InstalledClosedLimitLift</b> Specifies a bound for the bottom position (lift height), in centimeters.
0x0012	unsigned16, read-only <sup>6</sup>	<b>InstalledOpenLimitTilt</b> Specifies a bound for the open position (tilt angle), in units of 0.1°.
0x0013	unsigned16, read-only <sup>6</sup>	<b>InstalledClosedLimitTilt</b> Specifies a bound for the closed position (tilt angle), in units of 0.1°.
0x0014	unsigned16, read-only	<b>Velocity</b> Notice: This attribute is present, but currently not maintained.
0x0015	unsigned16, read-only	<b>AccelerationTime</b> Notice: This attribute is present, but currently not maintained.
0x0016	unsigned16, read-only	<b>DecelerationTime</b> Notice: This attribute is present, but currently not maintained.
0x0017	bitmap8, persistent	<b>Mode</b> Determines if the motor direction is reversed, the device is in calibration or maintenance mode, and whether it shall provide optical feedback.

#### Manufacturer-specific attributes supported:

Attribute	Type	Description
0x0000	enum8, persistent	<b>WindowCoveringType*</b> Writable version of the same standard attribute. This can be used to configure the proper window covering type, according to what is actually connected to J1(-R)'s motor output. Notice that the device may behave differently based on this selection, so it should be configured early in the commissioning process.
0x0007	bitmap8, persistent <sup>18</sup>	<b>ConfigurationAndStatus*</b> Writable version of the same standard attribute. This attribute is currently intended to disable the closed-loop behavior of the controller, in case a relay is connected to the output or the motor uses an electronic shut off at the top/bottom positions with a significant quiescence current level.
0x0010	unsigned16, persistent <sup>19</sup>	<b>InstalledOpenLimitLift*</b> Writable version of the same standard attribute.
0x0011	unsigned16, persistent <sup>7</sup>	<b>InstalledClosedLimitLift*</b> Writable version of the same standard attribute.

<sup>17</sup> In application versions prior to 1.05, this attribute was read/write and persistent. Now, use the manufacturer-specific mirrored versions to write this attribute.

<sup>18</sup> Available since application version 1.08. Prior a fixed default value of 0x18 ("lift/tilt closed-loop") was applied.

<sup>19</sup> Available since application version 1.05. Prior the standard attribute was read/write and persistent.



0x0012	unsigned16, persistent <sup>7</sup>	<b>InstalledOpenLimitTilt*</b> Writable version of the same standard attribute.
0x0013	unsigned16, persistent <sup>7</sup>	<b>InstalledClosedLimitTilt*</b> Writable version of the same standard attribute.
0x1000	unsigned8, persistent	<b>TurnaroundGuardTime</b> When an abrupt change of the direction occurs during motion, this guard time is applied to prevent damage to the motor, blind and any mechanical parts. Specified in units of 50ms. The default value of 10, which translates to 500ms, is applicable for most single-phase, bidirectional motors. Caution: Don't specify values, which are less than the minimum recommended by the motor manufacturer!
0x1001	unsigned16, persistent	<b>LiftToTiltTransitionSteps</b> Number of steps (measured in number of full AC waves), which a transition from lift to tilt takes for the attached blind. This applies to lift & tilt or tilt only blinds and specifies the time the blade requires for a full turn from fully open to fully closed.
0x1002	unsigned16, persistent	<b>TotalSteps</b> Total number of steps (measured in number of full AC waves), which a run from the fully open position to the fully closed position takes. This applies to all kinds of attached devices.
0x1003	unsigned16, persistent	<b>LiftToTiltTransitionSteps2</b> Number of steps (measured in number of full AC waves), which a transition from tilt to lift takes for the attached blind. This applies to lift & tilt or tilt only blinds and specifies the time the blade requires for a full turn from fully closed to fully open. Currently, this value should be equal to LiftToTiltTransitionSteps.
0x1004	unsigned16, persistent	<b>TotalSteps2</b> Total number of steps (measured in number of full AC waves), which a run from the fully closed position to the fully open position takes. This applies to all kinds of attached devices. This value might be significantly larger than TotalSteps, which presents the same value for the opposite direction.
0x1005	unsigned8, persistent	<b>AdditionalSteps</b> Specifies the additional steps (in percent) that the controller will run to make sure it really reaches the top or bottom position. For example, if the calculated height (TotalSteps) is 1,000 steps, the default value of 10% results in 100 additional steps that the controller will run the motor, before it shuts off power to the motor. Notice that the calculated position will be fixed at the bottom (or top) position in that case.
0x1006	unsigned16, persistent	<b>InactivePowerThreshold<sup>20</sup></b> This attribute provides a means to support motors with electronic shut-off, which draw electrical current even in their "off" state. It defines a threshold on the active power level, which discriminates the "active" (motor running) and "inactive" (motor off) states of the attached motor. You can disable this feature by writing a value of 0xFFFF. The value is specified in milliwatts, with a default setting of 0x1000 (approximately 4.1W), which should work out-of-the-box for most installations.
0x1007	unsigned16, persistent	<b>StartupSteps<sup>20</sup></b> Specifies the number of steps (measured in number of full AC waves) that the motor requires to exceed the "inactive threshold". This is primarily meant for motors with electronic shut-off and allows the motor to start, i.e. it is not being "inactive" during this time. The default value of 0x0020, or 32 full AC wave steps, should work out-of-the-box for most electronic motors on the market.

Cluster commands supported:

Command	Description
0x00	<b>Move up/open</b> Move upwards, towards the fully open position.
0x01	<b>Move down/close</b> Move downwards, towards the fully closed position.

<sup>20</sup> Availalbe since application version 1.13

0x02	<b>Stop</b> Stop all motion.
0x04	<b>Go to Lift Value</b> Moves to the specified lift value.
0x05	<b>Go to Lift Percentage</b> Moves to the specified lift percentage.
0x07	<b>Go to Tilt Value</b> Move to the specified tilt value.
0x08	<b>Go to Tilt Percentage</b> Move to the specified tilt percentage.

### 7.2.5.1. Calibration

The device must be calibrated once after installation before more advanced positioning features (other than move up/down and stop) can be used.

You might want to disable the positioning feature altogether, in case the particular setup is not compatible with the requirements of the positioning logic. To disable positioning altogether, clear the lift closed-loop and tilt closed-loop bits in `ConfigurationAndStatus*` (these are the bits #3 and #4).

Some motors with electronic shut-off draw significant power in the “off” state. For those motors, the inactive power threshold must be set appropriately, because the more precise bounds detection based on cease of current flow is not applicable. You should set the `InactivePowerThreshold` attribute value to a value that is above the motor's stand-by consumption and below the active power consumption when the motor is moving. Stand-by power loss is around 1W...2W for common electronic motors, so the default value of 0x1000 or ~4.1W should cover any motor on the market as it is also far from typical active power consumptions of 50W...250W. The default `StartupSteps` of 0x0020 (32 full AC waves) should also work for most installations.

#### Step 1

In order to calibrate the device, first choose the appropriate device type.

Value	Type	Capabilities
0	Roller Shade	Lift only
1	Roller Shade, two motors	Lift only
2	Roller Shade, exterior	Lift only
3	Roller Shade, two motors, exterior	Lift only
4	Drapery	Lift only
5	Awning	Lift only
6	Shutter	Tilt only
7	Tilt Blind, tilt only	Tilt only
8	Tilt Blind, lift & tilt	Lift & Tilt
9	Projector Screen	Lift only

Write attribute 0x10F2:0x0000 ("WindowCoveringType") accordingly.

### Step 2

Prepare calibration by setting these values:

Write attribute 0x10F2:0x0010 ("InstalledOpenLimitLift") as 0x0000 = 0cm.

Write attribute 0x10F2:0x0011 ("InstalledClosedLimitLift") as 0x00F0 = 240cm.

Write attribute 0x10F2:0x0012 ("InstalledOpenLimitTilt") as 0x0000 = 0°.

Write attribute 0x10F2:0x0013 ("InstalledClosedLimitTilt") as 0x0384 = 90.0°.

Write attribute 0x10F2:0x1001 ("LiftToTiltTransitionSteps") as 0xFFFF = invalid.

Write attribute 0x10F2:0x1002 ("TotalSteps") as 0xFFFF = invalid.

Write attribute 0x10F2:0x1003 ("LiftToTiltTransitionSteps2") as 0xFFFF = invalid.

Write attribute 0x10F2:0x1004 ("TotalSteps2") as 0xFFFF = invalid.

### Step 3

Enter calibration mode:

Write attribute 0x0017 ("Mode") as 0x02.

### Step 4

Send the "move down" command and "stop" after a few centimeters. This is to reach a starting position for calibration, which will allow J1 to detect the upper limit.

### Step 5

Send the "move up" command. When the device reaches its top position, J1 will recognize the upper bound.

### Step 6

After J1 has reached the top position and the motor has stopped, send the "move down" command. J1 will search for the lower bound and recognize the lower bound once it has been reached.

### Step 7

After J1 has reached the lower bound and the motor has stopped, send the "move up" command. J1 will search for the upper bound. Once the top position is reached, calibration of the total steps in both directions is complete.

When calibration is complete attributes 0x10F2:0x1002 and 0x10F2:0x1004 are different from 0xFFFF and hold the times that J1 has measured for your physical setup consisting of motor and blind. The values can be close together (typically lift & tilt blinds match closely) or have significant differences (down-to-up takes e.g. 100% longer than up-to-down due to gravity). Instead of auto-calibration you can also write the values directly if you know them.

### Step 8

In case of a tilt blind set attribute 0x10F2:0x1001 and 0x10F2:0x1003 to the time it takes for a lift-to-tilt transition (down) or a tilt-to-lift transition (up), respectively. Otherwise proceed with the next step.

### Step 9

To leave calibration mode, clear bit #1 in the Mode attribute, e.g. write attribute 0x0017 as 0x00.

## Hints

Once you leave calibration mode, move the blind down a little and then move it up. When it reaches the top position the relevant positioning attributes (absolute and percentage) should read as 0 (lift & tilt). You can now also use the “move to ...” commands to position the blind at e.g. 25% lift, 50% tilt etc. Also scene support becomes operational after calibration.

You should use the lift & tilt percentage in your UI, as this will result in a consistent user experience regardless of the absolute height/angle of the blind.

After a power-cycle you need to let the blind reach the upper position to regain position awareness, i.e. move down a little, and then move up. Notice that, for safety reasons, J1 does not start any motion unless explicitly requested to move up/down, move to lift/tilt or a relevant scene is recalled.

Notice: You can also visualize activity, i.e. motor running up/down using the draft (ZCLr06) attribute “OperationalStatus”, which is also reportable.

## 7.3. Application Endpoint #2 – Window Covering Controller

This is a Finding & Binding initiator endpoint.

### 7.3.1. Basic Cluster (Server)

Please refer to section 7.5.1 for details.

### 7.3.2. Identify Cluster (Server)

Notice: The identify cluster instance on this endpoint does not provide any visual or audible feedback. It is mainly present to comply with standards requirements.

Attributes supported:

Attribute	Type	Description
0x0000	unsigned16	<b>IdentifyTime</b> The remaining time, in seconds, the device will be identifying itself. Notice that J1 currently does not provide any visual feedback in identify mode.

Cluster commands supported:

Command	Description
0x00	<b>Identify</b> Puts the device into identify mode, or terminates identify mode.
0x01	<b>Query Identify</b> Determines whether the device is currently identifying, and returns the remaining time if so.

### 7.3.3. Scenes Cluster (Client)

This cluster is available on devices running firmware version 1.07 and above.

Notice: This cluster does not use the binding table for managing command targets. Instead, recall scene command templates include the group address and scene number to recall. The device setup cluster on the Device Management endpoint can be used to configure the behavior of the cluster, i.e. which identifiers will be sent in the message body, when the input state changes. Using the device setup cluster, you can configure this endpoint either to be used with a push-button or a switch.

This cluster instance is intended for recalling scenes on the local and/or remote devices using the local inputs and is not bound when factory fresh.

Cluster commands transmitted:

Command	Description
0x05	<b>Recall Scene</b> Recalls a scene. The group address in the recall scene command payload is also used as the target for the command.

### 7.3.4. Window Covering Cluster (Client)

Notice: This cluster uses the binding table for managing command targets. When factory fresh, this cluster is bound to endpoint #1 to enable local control. This works even when the device is not commissioned yet. In addition, the device setup cluster on the Device Management endpoint can be used to configure the behavior of the cluster, i.e. which of the commands mentioned below will actually be sent, when the input state changes. Using the device setup cluster, you can configure this endpoint either to be used with two push-buttons (momentary switches, one stable position) or two rocker switches (two stable positions).

Cluster commands transmitted:

Command	Description
0x00	<b>Move up/open</b> Makes the target device or group of devices move up, towards the fully open position.
0x01	<b>Move down/close</b> Makes the target device or group of devices move down, towards the fully closed position.
0x02	<b>Stop</b> Stops all motion.

## 7.4. Application Endpoint #3 – Metering

The metering endpoint provides information about energy consumption, instantaneous power, voltage, current, frequency, etc. This can be extended – upon customer request – with even more advanced features like waveform sampling, total harmonic distortion, or other quality measures.

The application endpoint exposes the following clusters:

Cluster	Direction	Description
0x0000	Inbound (Server)	<b>Basic</b> Provides basic information about the device, such as the manufacturer ID, vendor and model name, stack profile, ZCL version, production date, hardware revision etc. Allows a factory reset of attributes, without the device leaving the network.
0x0702	Inbound (Server)	<b>Metering</b> Provides metering information regarding energy and power.
0x0B04	Inbound (Server)	<b>Electrical Measurement</b> Provides electrical measurement figures, for example line frequency, active power, reactive power, apparent power, RMS voltage, RMS current, power factor, etc.

### 7.4.1. Basic Cluster (Server)

Please refer to section 7.5.1 for details.

### 7.4.2. Metering Cluster (Server)

Notice: This cluster uses the binding table for managing reporting targets. Currently, there is no default reporting configuration in place, so you need to configure reporting, if you want to receive feedback on energy/power changes.

Attributes supported:

Attribute	Type	Description
0x0000	unsigned48, read-only	<b>CurrentSummationDelivered</b> Indicates the current amount of electrical energy delivered to the load.
0x0001	unsigned48, read-only	<b>CurrentSummationReceived</b> Indicates the current amount of electrical energy generated by the attached device.
0x0002	unsigned48, read-only	<b>CurrentMaxDemandDelivered</b> Indicates the peak electrical power delivered to the load.
0x0003	unsigned48, read-only	<b>CurrentMaxDemandReceived</b> Indicates the peak electrical power generated by the attached device.
0x0200	bitmap8, read-only	<b>Status</b> Flags indicating current device status.
0x0300	enum8, read-only	<b>UnitOfMeasure</b> This is always kW.
0x0400	signed24, read-only, reportable	<b>InstantaneousDemand</b> The power currently delivered to the attached load, in Watts. A negative value indicates power generated by the attached device.

### 7.4.3. Electrical Measurement Cluster (Server)

Notice: This cluster uses the binding table for managing reporting targets. Currently, there is no default

reporting configuration in place, so you need to configure reporting, if you want to receive feedback on state changes.

Attributes supported:

Attribute	Type	Description
0x0000	bitmap32, read-only	<b>MeasurementType</b> Indicates the physical entities that this devices is able to measure.
0x0300	unsigned16, read-only	<b>Frequency</b> The frequency of the measured AC voltage (sine wave), in 0.001 Hz.
0x0304	signed32, read-only	<b>TotalActivePower</b> Total of delivered (positive) or generated (negative) active power, in W.
0x0305	signed32, read-only	<b>TotalReactivePower</b> Total of delivered (positive) or generated (negative) reactive power, in VAr.
0x0306	unsigned32, read-only	<b>TotalApparentPower</b> Total of delivered (positive) or generated (negative) apparent power, in VA.
0x0505	unsigned16, read-only	<b>RMSVoltage (L1)</b> Phase L1: RMS voltage.
0x0508	unsigned16, read-only	<b>RMSCurrent (L1)</b> Phase L1: RMS current.
0x050B	signed16, read-only	<b>ActivePower (L1)</b> Phase L1: Delivered (positive) or generated (negative) active power, in W.
0x050E	signed16, read-only	<b>ReactivePower (L1)</b> Phase L1: Delivered (positive) or generated (negative) reactive power, in VAr.
0x050F	unsigned16, read-only	<b>ApparentPower (L1)</b> Phase L1: Delivered (positive) or generated (negative) apparent power, in VA.
0x0510	signed8, read-only	<b>Powerfactor (L1)</b> Phase L1: Power factor of the attached load, in 0.01 units. Positive values indicate an inductive load (L), negative values a capacitive load (C), close to zero values a resistive load (R) and close to 1.00 purely capacitive/inductive loads.



## 7.5. Application Endpoint #232 – Device Management

This endpoint provides device management functions using standard ZigBee foundation paradigms and the ZigBee Cluster Library [R4]. You may use the standard ZCL frames to enumerate, read and write attributes, configure attribute reporting, invoke commands, etc.

The application endpoint exposes the following clusters:

Cluster	Direction	Description
0x0000	Inbound (Server)	<b>Basic</b> Provides basic information about the device, such as the manufacturer ID, vendor and model name, stack profile, ZCL version, production date, hardware revision etc. Allows a factory reset of attributes, without the device leaving the network.
0x0003	Outbound (Client) <sup>21</sup>	<b>Identify</b> Allows to identify targets during Finding & Binding push-button commissioning.
0x0015	Inbound (Server)	<b>Commissioning</b> Offers fine-grained commissioning options, for example, configuring a certain short address, channel mask, concentrator setup etc. This cluster is rarely used.
0x0019	Outbound (Client)	<b>OTA Upgrade</b> Pull-oriented firmware upgrade. Searches the network for mating servers and allows the server to control all stages of the upgrade process, including which image to download, when to download, at what rate and when to install the downloaded image.
0xFC00	Inbound (Server)	<b>Device Setup</b> This is a manufacturer-specific cluster used for basic device setup. For example, it can be used to assign physical inputs to endpoints and to control the behaviour of inputs (i.e. permanent switches, push-button switches, normally open vs. normally closed, on/off, level control or scene selection, etc.)

### 7.5.1. Basic Cluster (Server)

The basic cluster might be accessible via more than one endpoint. Most of its attributes are singleton attributes, i.e. all instances of the cluster share the same attribute value storage.

Attributes supported:

Attribute	Type	Description
0x0000	unsigned8, read-only	<b>ZCLVersion</b> The version of the cluster library
0x0001	unsigned8, read-only	<b>ApplicationVersion</b> The application version
0x0002	unsigned8, read-only	<b>StackVersion</b> The stack version
0x0003	unsigned8, read-only	<b>HWVersion</b> Specifies the hardware revision number
0x0004	string, read-only	<b>ManufacturerName</b> "ubisys"
0x0005	string, read-only	<b>ModelIdentifier</b> "J1 (5502)" or "J1-R (5602)"
0x0006	string, read-only	<b>DateCode</b> "YYYYMMDD-XX-FBV", where YYYY = year, MM = month, DD = day of production (hardware), XX = production facility, V = factory block version. For example, "20130411-DE-FB0" refers to a device built April 11, 2013 in Germany and programmed with factory block format version 0.

<sup>21</sup> Since application version 1.06. Prior, the server side of identify was available.

0x0007	enum8, read-only	<b>PowerSource</b> Always set to mains-powered, single phase (regardless of the connected supply voltage)
0x0010	string, persistent	<b>LocationDescription</b> Empty string by default. Might be set by commissioning software during installation
0x0011	unsigned8, persistent	<b>PhysicalEnvironment</b> "Unspecified" by default. Might be set by commissioning software during installation

Cluster commands supported:

Command	Description
0x00	<b>Reset to factory defaults.</b> Restores the values of all attributes of all clusters across all endpoints to their factory fresh default settings. Does not affect network settings or bindings. Notice: In earlier versions, up to and including version 1.10, the command did actually behave like a leave command with factory reset. This command has been deprecated and is no longer supported since version 1.11. Use <code>mgmt_leave_req</code> , instead.

### 7.5.2. Identify Cluster (Client)

The client side of the identify cluster on this endpoint is used for push-button commissioning.

Cluster commands transmitted:

Command	Description
0x01	<b>Query Identify</b> This command is broadcast during push-button commissioning to identify suitable binding targets. Once target devices are selected, they will be queried for matching descriptors.

### 7.5.3. Commissioning Cluster (Server)

This implementation supports all attributes and most commands. However, the cluster has limited value. It is rarely used and might be removed in the future.

### 7.5.4. Over-the-Air Upgrade Cluster (Client)

The image type for J1 is 0x7B04, for J1-R it is 0x7B07. You may request the latest firmware in ZigBee OTA image format to upgrade devices in the field using your own OTA server and back-end. If you operate the shutter control J1/J1-R together with the ZigBee/Ethernet Gateway ubisys G1, the latest firmware for J1 and J1-R will always be available automatically and you do not need to request it explicitly.

### 7.5.5. Device Setup Cluster (Server)

This cluster is manufacturer-specific and allows to change advanced device setup options, which are not covered by standard clusters defined in the ZigBee Cluster Library.

Attributes supported:

Attribute	Type	Description
0x0000	array of data8, persistent	<b>InputConfigurations</b> This array of 8-bit data holds exactly one entry per physical device input and allows disabling the input or inverting the “normal” signal level (normally closed vs. normally open).
0x0001	array of raw data, persistent	<b>InputActions</b> This array of raw data strings maps physical inputs to application endpoints on the device (for example, an on/off switch application) and translates level changes to appropriate cluster commands (e.g. on/off/toggle or move/stop).

### 7.5.5.1. InputConfigurations Attribute

This attribute is an array (ZCL data type 0x48) of 8-bit data (ZCL data type 0x08).

It is readable and writable and retains its value across reboots (persistent storage); when written the size of the attribute is not allowed to change, and the elements must be of the same type “8-bit data” (0x08). Each physical device input line has a one-to-one relation to a slot in this array.

For J1 and J1-R this relation is as follows:

Element #	Description
0x0000	<b>Configuration options for high-voltage physical input #1</b> Defines the configuration for the physical input marked with a switch symbol and the digit “1” in an upward pointing arrowhead. In case of J1, this is the white wire. Factory default value: 0x00
0x0001	<b>Configuration options for high-voltage physical input #2</b> Defines the configuration for the physical input marked with a switch symbol and the digit “2” in a downward pointing arrowhead. In case of J1, this is the grey wire. Factory default value: 0x00

Each element can take a combination of the following bit flags:

Flag	Bit(s)	Description
Disable	#7 (0x80)	When this bit is set, the input is disabled.
Invert	#6 (0x40)	Determines the “active” level. The default is active-high, meaning that a high voltage level (within 10% of the supply voltage) translates to an active input, and a low voltage level translates to an inactive signal. If this bit is set, the input is “active-low”, instead. For normally open circuits, leave this bit clear; for normally closed circuits make sure this bit is set.
RFU	#5...#0 (0x3F)	All other bits are reserved for future use and must be written as 0 and ignored when read.

### 7.5.5.2. InputActions Attribute

This attribute is an array (ZCL data type 0x48) of raw binary data (ZCL data type 0x41).

It is readable and writable and retains its value across reboots (persistent storage); when written, the elements must be of the same type “raw data” (0x41).

This attribute contains instructions, a kind of micro-code, which allow flexible reconfiguration of the commands sent in response to activity on the physical inputs.

The array elements take the following form:

Field	Data Type	Description
InputAndOptions	unsigned8	A four-bit input index in the LSBs and four option flags in the MSBs.
Transition	unsigned8	Specifies the level transition to which this particular instruction applies.
Endpoint	unsigned8	Each instruction must be assigned to a suitable source endpoint in the local device. Acceptable endpoint for J1/J1-R: Endpoint #2.
ClusterID	unsigned16	The 16-bit cluster ID that will be used to transmit the ZCL command payload, e.g. 0x0102 for the window covering cluster.
CommandTemplate	raw data	A variable length ZCL command template, which will be sent from the source endpoint specified above, using the cluster ID specified above. This can be a simple move up (0x00), move down (0x01) or stop (0x02) command, or more complex commands. There might be code in the device that modifies the command template according to intrinsic rules or option flag settings. Notice that this is the raw payload and does not include a leading length byte.

### InputAndOptions

This field contains the physical input number in the lower four bits and indexes into the InputConfigurations array. For example, a value of 0 identifies the first physical input. All other bits are reserved for future use and must be written as 0, and ignored when read.

For J1 and J1-R, this field may take the values 0 and 1.

### Transition

The Transition field specifies when (i.e. under which conditions/circumstances) this particular instruction shall be executed.

Field	Bit(s)	Description
HasAlternate	#7 (0x80)	When this bit is set, there is another instruction that shall be executed in alternating order with the current one, each time the other conditions are met. In this case two entries must exist, one with the Alternate bit clear, and one with the Alternate bit set.
Alternate	#6 (0x40)	If this bit is set, this is the alternate instruction of a pair of two instructions which are executed in alternating order. If this flag is set, "HasAlternate" must be set as well.
RFU	#5...#4 (0x30)	These bits are reserved for future use and must be written as 0 and ignored when read.
Initial State	#3...#2 (0x0C)	These bits indicate the initial state of the input: 00b: Ignore: Don't care 01b: Pressed: The input was initially active (for less than a second) 10b: Kept pressed: The input has initially been active (for more than a second) 11b: Released. The input was initially inactive
Final State	#1...#0 (0x03)	These bits indicate the final state of the input: 00b: Ignore: Don't care 01b: Pressed: The input is active (for less than a second) 10b: Kept pressed: The input is still active (for more than a second) 11b: Released. The input is inactive

The "don't care" states are no real states, but rather placeholders for any other state. At any given time the input is either released, pressed or kept pressed.

### Endpoint

Identifies the source endpoint that will be used in the outgoing ZCL command frame. Notice, that you should only use an endpoint, which hosts a cluster with the appropriate cluster ID and role (client vs.

server).

#### ClusterID

Specifies the cluster ID that will be used in the outgoing ZCL command frame. The cluster ID must be an “operational” cluster, according to ZCL definitions.

#### CommandTemplate

This is the template of a ZCL command frame payload, which will be sent to the target device once the instruction is selected for execution by the universal input logic unique to ubisys ZigBee devices. This can be a simple toggle command, or a more complex move or recall scene command, for example.

Notice that J1 and J1-R will use their binding tables to determine the target for the command. The command may be sent to one or more targets, each being a group (specified as a group address) or a single application instance (specified as IEEE EUI-64 address and endpoint number).

## Examples

This is the complete default configuration for J1 and J1-R, which is aimed at dual push-button operation (momentary, one stable position):

```
41          element type: 0x41 (raw data)
04 00          element count: 0x0004 (4 entries)

06          element #1: six bytes
00          InputAndOptions: 0x00
0D          Transition: released -> pressed
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
00          ZCL Command Template: Move up/open

06          element #2: six bytes
00          InputAndOptions: 0x00
07          Transition: pressed -> released
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
02          ZCL Command Template: Stop

06          element #3: six bytes
01          InputAndOptions: 0x01
0D          Transition: released -> pressed
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
01          ZCL Command Template: Move down/close

06          element #4: six bytes
01          InputAndOptions: 0x01
07          Transition: pressed -> released
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
02          ZCL Command Template: Stop
```

A short press will move up/down and stop when released, while a long press will move up/down without stopping before the fully open or fully closed position is reached, respectively. This is particularly useful for lift & tilt blinds, but also generally suitable for all kinds of attached devices.

If stationary switches are connected to the inputs instead, the following instructions shall be used:

```
41          element type: 0x41 (raw data)
04 00          element count: 0x0004 (4 entries)

06          element #1: six bytes
00          InputAndOptions: 0x00
0D          Transition: released -> pressed
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
00          ZCL Command Template: Move up/open

06          element #2: six bytes
00          InputAndOptions: 0x00
03          Transition: any state -> released
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
02          ZCL Command Template: Stop

06          element #3: six bytes
01          InputAndOptions: 0x01
0D          Transition: released -> pressed
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
01          ZCL Command Template: Move down/close

06          element #4: six bytes
01          InputAndOptions: 0x01
03          Transition: any state -> released
02          Source: Endpoint #2 (hosts window covering client cluster on J1)
02 01          Cluster ID: 0x0102 - window covering
02          ZCL Command Template: Stop
```

Here, the blind moves as long as either switch is turned on. As soon as it is turned off, motion stops.

The ubisys Smart Home app for iOS and Android recognizes and generates a certain set of instructions, and lets users pick those easily. You may request an up-to-date list of optimized instructions by mailing [support@ubisys.de](mailto:support@ubisys.de) and asking for these recommended instructions for typical use cases. Avoid using other instructions.

## 7.6. Application Endpoint #242 – ZigBee Green Power

This endpoint provides the ZigBee Green Power feature according to the 2015 edition of the specification, i.e. including support for Green Power Devices with IEEE EUI-64 and bidirectional commissioning. The ZigBee traffic between Proxies and Sinks utilizes standard ZigBee foundation paradigms and the ZigBee Cluster Library [R4]. You may use the standard ZCL frames to enumerate, read and write attributes, invoke commands, etc.

The application endpoint exposes the following clusters:

Cluster	Direction	Description
0x0021	Inbound (Server)	<b>ZigBee Green Power Sink</b> Allows pairing the dimmer with ZigBee Green Power switches. The sink can directly receive Green Power frames but can also utilize proxies in the network.
0x0021	Outbound (Client)	<b>ZigBee Green Power Proxy</b> Allows sinks on the network to configure this device as a “Proxy”, i.e. access point for ZigBee Green Power Devices into the ZigBee mesh network.

### 7.6.1. Green Power Cluster (Server)

The server-side of the Green Power cluster provides the ZigBee Green Power Sink functionality, i.e. provides the ability to process Green Power frames. The device can receive those frames directly if they are in radio range; or it can process frames tunnelled over the ZigBee network via other Green Power Proxies. This implementation supports unidirectional and bidirectional<sup>22</sup> GPDs.

Attributes supported:

Attribute	Type	Description
0x0000	unsigned8, read-only	<b>gpsMaxSinkTableEntries</b> The number of sink table entries supported by this device
0x0001	extended raw binary, read-only, persistent	<b>SinkTable</b> Entries in the sink table create a pairing between the sink and a Green Power Device. For an operational pairing, one or more entries in the translation table are also required
0x0002	bitmap8	<b>gpsCommunicationMode</b> Determines the default communication mode between sink and proxies for new entries added by the sink to the sink table when paired with a GPD
0x0003	bitmap8	<b>gpsCommissioningExitMode</b> Specifies under which circumstances the sink should leave commissioning mode
0x0004	unsigned16	<b>gpsCommissioningWindow</b> Time, in seconds, before a commissioning session expires
0x0005	bitmap8, persistent	<b>gpsSecurityLevel</b> Determines the minimum security level a GPD is required to support as well as whether or not the Trust Center needs to be involved into the commissioning process
0x0006	bitmap24, read-only	<b>gpsFunctionality</b> Indicates Green Power features and building blocks supported by this device
0x0007	bitmap24, read-only	<b>gpsActiveFunctionality</b> Allows to disable certain Green Power features on this device
0x0020	bitmap8, persistent	<b>gpsSharedSecurityKeyType</b> Determines the security key type to use for devices with bidirectional commissioning capabilities, i.e. out-of-the-box individual key, shared GP key, etc.

<sup>22</sup> Bidirectional communication is currently limited to the commissioning stage



0x0021	key128, persistent	<b>gpSharedSecurityKey</b> The 128-bit AES-CCM* key that is being used to secure Green Power data frames
0x0022	key128, persistent	<b>gpLinkKey</b> The 128-bit AES-CCM* key that is being used to deliver keying material to Green Power devices

Cluster commands supported:

Command	Description
0x00	<b>GP Notification</b> Tunnels GP frames from a Green Power Device to one or more sinks or groups of sinks
0x04	<b>GP Commissioning Notification</b> Tunnels GP frames from a Green Power Device to a sink in commissioning mode
0x05	<b>GP Sink Commissioning Mode</b> Makes the sink enter or leave commissioning mode for a particular endpoint or all endpoints
0x09	<b>GP Pairing Configuration</b> Creates, updates or removes pairings by modifying the sink and translation table entries accordingly
0x0A	<b>GP Sink Table Request</b> Allows to query the sink table for a certain Green Power Device or read out the table in chunks

Cluster commands transmitted:

Command	Description
0x01	<b>GP Pairing</b> Creates, updates or removes proxy table entries
0x02	<b>GP Proxy Commissioning Mode</b> Makes the proxy enter commissioning mode for a particular sink, or leave commissioning mode
0x06	<b>GP Response</b> Tunnels GP data frames from a sink to a bidirectional Green Power Device
0x0A	<b>GP Sink Table Response</b> Conveys a set of sink table entries to a management application

## 7.6.2. Green Power Cluster (Client)

The client-side of the Green Power cluster provides the ZigBee Green Power Proxy functionality, i.e. makes the device act as an “access point” for Green Power Devices (GPDs). This implementation supports unidirectional and bidirectional<sup>23</sup> GPDs.

Attributes supported:

Attribute	Type	Description
0x0010	unsigned8, read-only	<b>gppMaxProxyTableEntries</b> The number of proxy table entries supported by this device
0x0011	extended raw binary, read-only, persistent	<b>ProxyTable</b> Entries in the proxy table create a link between Green Power Devices and Green Power Sinks
0x0016	bitmap24, read-only	<b>gppFunctionality</b> Indicates Green Power features and building blocks supported by this device
0x0017	bitmap24, read-only	<b>gppActiveFunctionality</b> Allows to disable certain Green Power features on this device

<sup>23</sup> Bidirectional communication is currently limited to the commissioning stage

0x0020	bitmap8, persistent	<b>gpSharedSecurityKeyType</b> Determines the security key type to use for devices with bidirectional commissioning capabilities, i.e. out-of-the-box individual key, shared GP key, etc.
0x0021	key128, persistent	<b>gpSharedSecurityKey</b> The 128-bit AES-CCM* key that is being used to secure Green Power data frames
0x0022	key128, persistent	<b>gpLinkKey</b> The 128-bit AES-CCM* key that is being used to deliver keying material to Green Power devices

Cluster commands supported:

Command	Description
0x01	<b>GP Pairing</b> Creates, updates or removes proxy table entries
0x02	<b>GP Proxy Commissioning Mode</b> Makes the proxy enter commissioning mode for a particular sink, or leave commissioning mode
0x06	<b>GP Response</b> Tunnels GP data frames from a sink to a bidirectional Green Power Device
0x0B	<b>GP Proxy Table Request</b> Allows to query the proxy table for a certain Green Power Device or read out the table in chunks

Cluster commands transmitted:

Command	Description
0x00	<b>GP Notification</b> Tunnels GP frames from a Green Power Device to one or more sinks or groups of sinks
0x04	<b>GP Commissioning Notification</b> Tunnels GP frames from a Green Power Device to a sink in commissioning mode
0x0B	<b>GP Proxy Table Response</b> Conveys a set of proxy table entries to a sink or management application

## 8. Physical Dimensions

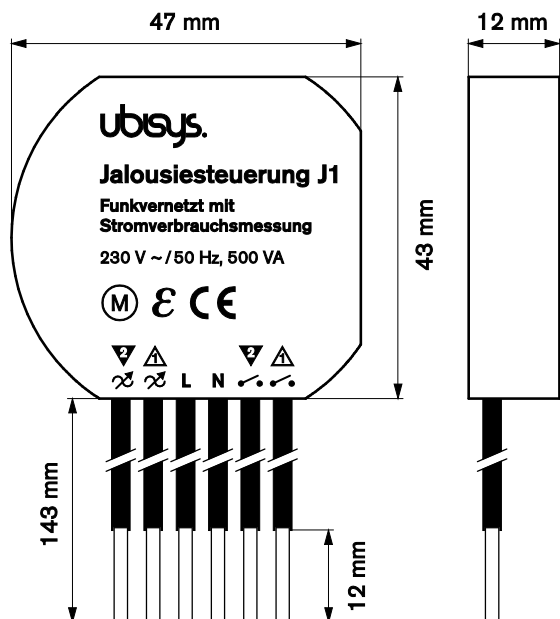


Figure 1: Flush-mount In-wall ZigBee Shutter Control J1 – Dimensions

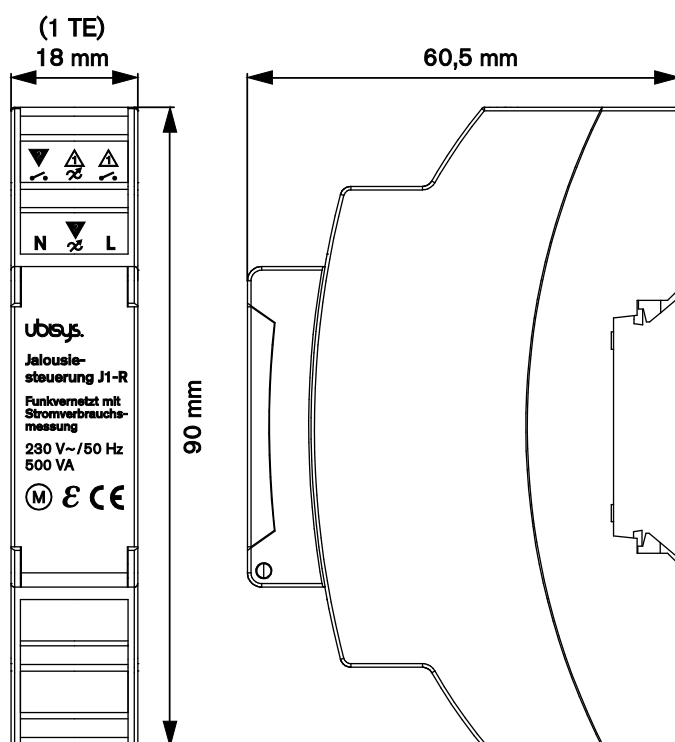


Figure 2: Rail-mount ZigBee Shutter Control J1-R – Dimensions

## 9. Ordering Information

The following tables list the product variants available. Use the specified order code for your orders. Please contact ubisys support if you require any customization.

Order Code	Description
1076	ZigBee Shutter Control J1 (in-wall, flush-mount)
1144	ZigBee Shutter Control J1-R (rail mount)

The housing is flame retardant (V-0), Color: Black, RAL 9005.

## 10. General Terms & Conditions of Business

When placing your order you agree to be bound by our General Terms & Conditions of Business, "Allgemeine Geschäftsbedingungen", which are available for download here:  
<http://www.ubisys.de/en/smarthome/terms.html>



We – ubisys technologies GmbH, Am Wehrhahn 45, 40211 Düsseldorf, Germany – declare under our sole responsibility that the ubisys ZigBee Shutter Controls J1 and J1-R with order codes as detailed in section 9 under the trade name “ubisys” to which this declaration relates are in conformity with the following directives and standards:

Directive/Standard	Description/Scope
2014/53/EU	Radio Equipment Directive (RED)
2004/108/EC	Electromagnetic Compatibility Directive (EMC)
2006/95/EC	Low Voltage Directive (LVD)
2002/96/EC	Waste Electrical and Electronic Equipment Directive (WEEE)
2002/95/EC	Restriction of Hazardous Substances Directive (RoHS)
EN 300 328	ERM; Wideband transmission systems; 2.4 GHz ISM band
EN 300 440	ERM; Radio equipment to be used in the 1 GHz to 40 GHz frequency range
EN 301 489	EMC
IEEE 802.15.4	IEEE Standard 802 – Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)
ZigBee 3.0	ZigBee 2015 with Green Power

Düsseldorf, Germany

**Place of issue**

April 28, 2016

**Date of issue**

Dr.-Ing. Arasch Honarbach

**Full name of Authorized Signatory**

Managing Director, Head of Research &amp; Development

**Title of Authorized Signatory**
**Signature**

UBISYS TECHNOLOGIES GMBH
   
 HARDWARE UND SOFTWARE DESIGN
   
 ENGINEERING UND CONSULTING
   
 AM WEHRHAHN 45
   
 40211 DÜSSELDORF
   
 info@ubisys.de
   
 www.ubisys.de

**Seal**

## 12. Revision History

Revision	Date	Remarks
1.0	11/25/2014	Initial Public Version
1.1	11/28/2014	Minor corrections.
1.2	12/01/2014	Fixed incorrect attribute IDs in section 7.2.5.1, Step 2.
1.3	01/06/2015	Includes documentation of ZigBee Network Steering and Finding & Binding ("EZ-mode"), as well as information on scenes cluster client instances in firmware revisions 1.07 and above.
1.4	01/15/2015	Added ConfigurationAndStatus* documentation for application firmware revisions 1.08 and above. Added a note on positioning after start-up.
1.5	02/13/2015	Mentioned a few input configuration options in the feature overview.
1.6	07/28/2015	Minor fix.
1.7	08/12/2015	Improved network steering in firmware revision 1.10 (toggle permit joining) and shortcut to factory reset by 10-second extended press.
1.8	11/02/2015	Starting with firmware revision 1.11, when the device is factory reset, it will broadcast a network leave indication
1.9	11/11/2015	Updated the ZDO description to include enhancements and additions made for ZigBee 2015 platform compliance
2.0	11/19/2015	Enhancements made in firmware revision 1.12 include added support for ZigBee Green Power and further changes to align with ZigBee 3.0 requirements.
2.1	01/21/2016	Firmware revision 1.13 provides support for positioning electronic motors, adding InactivePowerThreshold and StartupSteps attributes.
2.2	09/13/2016	Updated to ZigBee 3.0 Certified Product in application firmware 1.16; added information about the power-cycle sequence factory reset in application firmware 1.17. Corrected description of element data type for switch configuration (should have been 0x41 instead of 0x48).

## 13. Contact

### **UBISYS TECHNOLOGIES GMBH**

AM WEHRHAHN 45  
40211 DÜSSELDORF  
GERMANY

T: +49 (211) 54 21 55 - 00

F: +49 (211) 54 21 55 - 99

[www.ubisys.de](http://www.ubisys.de)

[info@ubisys.de](mailto:info@ubisys.de)