# ZIGBEE CONTROL UNIT C4

REFERENCE MANUAL

ubisys®

# Table of Contents


1. Overview . . . 1
2. Features . . . 2
3. Installation . . . 5
   3.1. Mains Powered Operation . . . 5
   3.2. Low-Voltage Operation (Maintenance and Early System Integration Stages only) . . . 5
4. Initial Device Start-Up . . . 6
5. Man-Machine interface (MMI) . . . 8
   5.1. Power-Cycle Sequencing Factory Reset . . . 9
6. Zigbee Interface . . . 11
   6.1. Device Anatomy . . . 11
   6.2. Installation Code . . . 11
   6.3. Application Endpoint #0 - Zigbee Device Object . . . 12
   6.4. Application Endpoint #1 – Primary Level Control Switch . . . 13
      6.4.1. Basic Cluster (Server) . . . 13
      6.4.2. Identify Cluster (Server) . . . 13
      6.4.3. Scenes Cluster (Client) . . . 13
      6.4.4. On/off Cluster (Client) . . . 14
      6.4.5. Level Control Cluster (Client) . . . 14
   6.5. Application Endpoint #2 – Secondary Level Control Switch . . . 15
      6.5.1. Basic Cluster (Server) . . . 15
      6.5.2. Identify Cluster (Server) . . . 15
      6.5.3. Scenes Cluster (Client) . . . 15
      6.5.4. On/off Cluster (Client) . . . 15
      6.5.5. Level Control Cluster (Client) . . . 16
   6.6. Application Endpoint #3 – Tertiary Level Control Switch . . . 16
      6.6.1. Basic Cluster (Server) . . . 16
      6.6.2. Identify Cluster (Server) . . . 16
      6.6.3. Scenes Cluster (Client) . . . 17
      6.6.4. On/off Cluster (Client) . . . 17
      6.6.5. Level Control Cluster (Client) . . . 18
   6.7. Application Endpoint #4 – Quaternary Level Control Switch . . . 18
      6.7.1. Basic Cluster (Server) . . . 18
      6.7.2. Identify Cluster (Server) . . . 18
      6.7.3. Scenes Cluster (Client) . . . 18
      6.7.4. On/off Cluster (Client) . . . 19
      6.7.5. Level Control Cluster (Client) . . . 19
   6.8. Application Endpoint #5 – Primary Window Covering Controller . . . 20
      6.8.1. Basic Cluster (Server) . . . 20
      6.8.2. Identify Cluster (Server) . . . 20
      6.8.3. Scenes Cluster (Client) . . . 20

6.8.4. Window Covering Cluster (Client) . . . 20
6.9. Application Endpoint #6 – Secondary Window Covering Controller . . . 21
6.9.1. Basic Cluster (Server) . . . 21
6.9.2. Identify Cluster (Server) . . . 21
6.9.3. Scenes Cluster (Client) . . . 21
6.9.4. Window Covering Cluster (Client) . . . 22
6.10. Application Endpoint #232 - Device Management . . . 22
6.10.1. Basic Cluster (Server) . . . 23
6.10.2. Identify Cluster (Client) . . . 23
6.10.3. Commissioning Cluster (Server) . . . 24
6.10.4. Over-the-Air Upgrade Cluster (Client) . . . 24
6.10.5. Device Setup Cluster . . . 24
6.11. Application Endpoint #242 - Zigbee Green Power . . . 27
6.11.1. Green Power Cluster (Server) . . . 27
7. Physical Dimension . . . 30
8. Ordering Information . . . 31
9. General Terms & Conditions Of Business . . . 32
10. Declaration Of Conformity . . . 33
11. Revision History . . . 35
12. Contact . . . 36

# 1. Overview

Thank you for purchasing the ubisys Zigbee Control Unit C4.

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.

Copyright© 2014-2022 ubisys technologies GmbH, Düsseldorf, Germany.
All rights reserved.

## 2. Features

- Zigbee 3.0 Certified Control Unit for up to four inputs with integrated Zigbee Router functionality
- Four configurable 230V~ inputs, can be individually set-up as on/off, level control, window covering controller and scene selector switches use with momentary or stationary switches
- Reliable, unattended, maintenance-free operation; designed for in-wall flush-mounting
- 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:
  - Global Default Trust Center Link-Key (“ZigbeeAlliance09”)
  - Zigbee 3.0 Global Distributed Security Link-Key [1]
  - 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

- OEM and customized versions available upon request

[1] Since application firmware 1.05. Prior, pre-certification key (D0:..:DF)

# 3. Installation

## 3.1. Mains Powered Operation

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

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

Instead of applying dangerous mains voltage to the control unit C4, you may connect a low-voltage DC source (e.g. 12V=, 24V=, 48V=) to the supply lines of C4. 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 are non-operational.

For DC operation (laboratory use), connect the DC ground (0V, “minus”, negative) of your power supply to the phase input of C4 (marked “L”, brown) and the DC supply voltage (12…48VDC, “plus”, positive) to the neutral input of C4 (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.

## 4. 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.

C4 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. [2]

[2] Stretching the joining duration is done in products with ubisys Zigbee Stack version 1.48 or above.

# 5. Man-Machine interface (MMI)

C4 offers a push-button behind a tiny hole in the front-face of the device, as well as a LED right next to it [3]. This man-machine interface provides access to a menu. In addition, it provides a handy shortcut to factory reset the device [4].

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 5.1

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 C4 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 Control Unit C4 running firmware version 1.01 and above [5] :

| Menu Item # | Operation |
|---|---|
| 1 | **Zigbee Commissioning: Network Steering**<br>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 | **Zigbee Commissioning: Finding & Binding**<br>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 | **Zigbee Commissioning: Clear Bindings**<br>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 | **Zigbee Commissioning: Set Device Role and Factory Reset**<br>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 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:<br>Option #1: Join an existing Zigbee network as router<br>Option #2: Form a new distributed security network as the first router<br>Option #3: Form a new centralized security network as the coordinator and trust center |
| 5 | **Factory Reset**<br>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 [6]. |

| Menu Item # | Operation |
|---|---|
| 6 | **Zigbee Commissioning: Advanced Commands**<br>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:<br>Option #1: Perform a simple reset (reboot), then continue operating seamlessly (silent re-join)<br>Option #2: Perform a simple reset (reboot), then re-join the network<br>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. |

## 5.1. Power-Cycle Sequencing Factory Reset

It is possible to instigate a factory reset using a special power-cycle sequence [7]. 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.

[3] The high-voltage inputs of C4 do not count as an MMI feature.

[4] The 10-second-press factory reset (equivalent to menu item #5) is available since firmware version 1.02.

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

[6] Since application firmware 1.03. Previously, the factory reset would result in a silent leave.

[7] Since application firmware 1.06

# 6. Zigbee Interface

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 Base Device Behavior Specification, Revision 13, Document No. 13-0402-13
[R6] Zigbee PRO Green Power Feature Specification, Revision 26, Document No. 09-5499-26
[R7] Zigbee Home Automation Public Application Profile 1.2, Revision 29, Document No. 05-3520-29
[R8] Zigbee Smart Energy Standard 1.1b, Revision 18, Document No. 07-5356-18

## 6.1. Device Anatomy

The control unit C4 provides four high-voltage inputs for remote control of other on/off, level control and window covering target devices or groups. Being a mains-powered device, C4 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.

C4 provides ten Zigbee application endpoints:

| Endpoint # | Profile | Application |
|---|---|---|
| 0 (0x00) | 0x0000: Zigbee Device Profile | Zigbee Device Profile Zigbee Device Object (ZDO) - standard management features |
| 1 (0x01) | 0x0104: Common Profile (HA) | Level Control Switch (0x0001). This endpoint is used to transmit on/off or level control commands triggered by any of the local high-voltage inputs. |
| 2 (0x02) | 0x0104: Common Profile (HA) | Level Control Switch (0x0001). This endpoint is used to transmit on/off or level control commands triggered by any of the local high-voltage inputs. |
| 3 (0x03) | 0x0104: Common Profile (HA) | Level Control Switch (0x0001). This endpoint is used to transmit on/off or level control commands triggered by any of the local high-voltage inputs. |
| 4 (0x03) | 0x0104: Common Profile (HA) | Level Control Switch (0x0001). This endpoint is used to transmit on/off or level control commands triggered by any of the local high-voltage inputs. |
| 5 (0x03) | 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. |
| 6 (0x03) | 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. |
| 200 (0xC8) | Private | Private. This endpoint contains a legacy private application and is 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 (0x0502) [8]. |
| 242 (0xF2) | 0xA1E0: Green Power Profile | Zigbee Green Power Proxy [9] |

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.

## 6.2. Installation Code

This control unit has a pre-configured link key, which is derived from the installation code printed on the back of the control unit'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 “C4” 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, “⊻”) 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') ⊻ 30 ('0') ⊻
00 ⊻ 1F ⊻ EE ⊻ 00 ⊻ 00 ⊻ 00 ⊻ 00 ⊻ FF ⊻
0F ⊻ 7C ⊻ 1C ⊻ D8 ⊻ 05 ⊻ F9 ⊻ 16 ⊻ 49 ⊻ EB ⊻ A8 ⊻ 45 ⊻ 80 ⊻ AA ⊻ 1C ⊻ B4 ⊻ 32 ⊻ F5
⊻ 1A = 21
```

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.

## 6.3. 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 Control Unit C4 supports the following ZDO services:

| Primitive | Description |
|---|---|
| nwk_addr_req/ nwk_addr_rsp | **Network address request/response** 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 | **IEEE address request/response** Translates a 16-bit network short address into a 64-bit IEEE address. |
| node_desc_req/ node_desc_rsp | **Node descriptor request/response** Returns information such as the manufacturer ID, power supply, etc. |
| power_desc_req/ power_desc_rsp [10] | **Power descriptor request/response** Returns information such as the power source and mode. |
| active_ep_req/ active_ep_rsp | **Active endpoints request/response** Returns a set of available application endpoints on the device. |
| simple_desc_req/ simple_desc_rsp | **Simple descriptor request/response** Returns a descriptor for a certain application endpoint with a list of available services (clusters). |
| match_desc_req/ match_desc_rsp | **Match descriptor request/response** Searches for a certain cluster or set of clusters and returns the matching endpoints, if any. |
| device_annce | **Device announcement** Advertises the presence of a new device in the network. |
| parent_annce/ parent_annce_rsp [11] | **Parent announcement/response** This is part of the Zigbee 2015 end-device child management feature. |
| system_server_discovery_req/ system_server_discovery_rsp [12] | **System server discovery request/response** Provides the ability to discover system servers, in particular the network manager. |

| Primitive | Description |
|---|---|
| bind_req/<br>bind_rsp | **Bind request/response**<br>Creates an application binding |
| unbind_req/<br>unbind_rsp | **Unbind request/response**<br>Removes an application binding |
| mgmt_nwk_disc_req/<br>mgmt_nwk_disc_rsp [13] | **Management: Network discovery request/response**<br>Instructs the device to perform a network discovery and report the results back. |
| mgmt_lqi_req/<br>mgmt_lqi_rsp | **Management: Neighbor table request/response**<br>Returns information about neighboring devices, including the link quality, device type etc. |
| mgmt_rtg_req/<br>mgmt_rtg_rsp | **Management: Routing table request/response**<br>Returns information about routes established on the device. |
| mgmt_bind_req/<br>mgmt_bind_rsp | **Management: Binding table request/response**<br>Returns information about application bindings on the device. |
| mgmt_leave_req/<br>mgmt_leave_rsp | **Management: Leave request/response**<br>Makes the device leave the network or removes one of its end-device children. |
| mgmt_permit_joining_req/<br>mgmt_permit_joining_rsp | **Management: Permit joining request/response**<br>Opens the network for new devices to join. |
| mgmt_nwk_update_req/<br>mgmt_nwk_update_notify [13] | **Management: Network update request/response/notification**<br>Performs energy scans, instigates a channel change or assigns the network manager. |

## 6.4. Application Endpoint #1 – Primary Level Control Switch

This is a Finding & Binding target endpoint.

### 6.4.1. Basic Cluster (Server)

Please refer to Section 6.10.1 for details.

### 6.4.2. Identify Cluster (Server)

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 | **IdentifyTime**<br>The remaining time, in seconds, the device will be identifying itself.<br>Notice that C4 currently does not provide any visual feedback in identify mode. |

Cluster commands supported:

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

### 6.4.3. Scenes Cluster (Client)

This cluster is available on devices running firmware version 1.01 and above.

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 | **Recall Scene**<br>Recalls a scene. The group address in the recall scene command payload is also used as the target for the command. |

### 6.4.4. On/off Cluster (Client)

This cluster uses the binding table for managing command targets. 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 a push-button (momentary switch, one stable position) or a rocker switch (two stable positions).

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x00 | **Turn off**<br>Turns the target device or group of devices off. |
| 0x01 | **Turn on**<br>Turns the target device or group of devices on. |
| 0x02 | **Toggle**<br>Toggles the target device state. Should not be used to control groups. |

### 6.4.5. Level Control Cluster (Client)

This cluster uses the binding table for managing command targets. 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 one or two push-buttons.

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x05 | **Move with on/off**<br>Moves the level either up or down at a specified rate.<br>Maintains the companion on/off cluster's OnOff attribute. |
| 0x07 | **Stop with on/off**<br>Stops any level change in progress due to a move, move to level, step or recall scene command.<br>Maintains the companion on/off cluster's OnOff attribute. |

## 6.5. Application Endpoint #2 – Secondary Level Control Switch

This is a Finding & Binding initiator endpoint.

### 6.5.1. Basic Cluster (Server)

Please refer to Section 6.10.1 for details.

### 6.5.2. Identify Cluster (Server)

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 | **IdentifyTime**<br>The remaining time, in seconds, the device will be identifying itself.<br>Notice that C4 currently does not provide any visual feedback in identify mode. |

Cluster commands supported:

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

### 6.5.3. Scenes Cluster (Client)

This cluster is available on devices running firmware version 1.01 and above.

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 | **Recall Scene**<br>Recalls a scene. The group address in the recall scene command payload is also used as the target for the command. |

### 6.5.4. On/off Cluster (Client)

This cluster uses the binding table for managing command targets. 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 a push-button (momentary switch, one stable position) or a rocker switch (two stable positions).

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x00 | **Turn off**<br>Turns the target device or group of devices off. |
| 0x01 | **Turn on**<br>Turns the target device or group of devices on. |
| 0x02 | **Toggle**<br>Toggles the target device state. Should not be used to control groups. |

### 6.5.5. Level Control Cluster (Client)

This cluster uses the binding table for managing command targets. 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 one or two push-buttons.

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x05 | **Move with on/off**<br>Moves the level either up or down at a specified rate.<br>Maintains the companion on/off cluster's OnOff attribute. |
| 0x07 | **Stop with on/off**<br>Stops any level change in progress due to a move, move to level, step or recall scene command.<br>Maintains the companion on/off cluster's OnOff attribute. |

## 6.6. Application Endpoint #3 – Tertiary Level Control Switch

This is a Finding & Binding initiator endpoint.

### 6.6.1. Basic Cluster (Server)

Please refer to Section 6.10.1 for details.

### 6.6.2. Identify Cluster (Server)

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 | **IdentifyTime**<br>The remaining time, in seconds, the device will be identifying itself.<br>Notice that C4 currently does not provide any visual feedback in identify mode. |

Cluster commands supported:

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

### 6.6.3. Scenes Cluster (Client)

This cluster is available on devices running firmware version 1.01 and above.

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 | **Recall Scene**<br>Recalls a scene. The group address in the recall scene command payload is also used as the target for the command. |

### 6.6.4. On/off Cluster (Client)

This cluster uses the binding table for managing command targets. 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 a push-button (momentary switch, one stable position) or a rocker switch (two stable positions).

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x00 | **Turn off**<br>Turns the target device or group of devices off. |
| 0x01 | **Turn on**<br>Turns the target device or group of devices on. |
| 0x02 | **Toggle**<br>Toggles the target device state. Should not be used to control groups. |

### 6.6.5. Level Control Cluster (Client)

This cluster uses the binding table for managing command targets. 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 one or two push-buttons.

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x05 | **Move with on/off**<br>Moves the level either up or down at a specified rate.<br>Maintains the companion on/off cluster's OnOff attribute. |
| 0x07 | **Stop with on/off**<br>Stops any level change in progress due to a move, move to level, step or recall scene command.<br>Maintains the companion on/off cluster's OnOff attribute. |

## 6.7. Application Endpoint #4 – Quaternary Level Control Switch

This is a Finding & Binding initiator endpoint.

### 6.7.1. Basic Cluster (Server)

Please refer to Section 6.10.1 for details.

### 6.7.2. Identify Cluster (Server)

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 | **IdentifyTime**<br>The remaining time, in seconds, the device will be identifying itself.<br>Notice that C4 currently does not provide any visual feedback in identify mode. |

Cluster commands supported:

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

### 6.7.3. Scenes Cluster (Client)

This cluster is available on devices running firmware version 1.01 and above.

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 | **Recall Scene**<br>Recalls a scene. The group address in the recall scene command payload is also used as the target for the command. |

### 6.7.4. On/off Cluster (Client)

This cluster uses the binding table for managing command targets. 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 a push-button (momentary switch, one stable position) or a rocker switch (two stable positions).

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x00 | **Turn off**<br>Turns the target device or group of devices off. |
| 0x01 | **Turn on**<br>Turns the target device or group of devices on. |
| 0x02 | **Toggle**<br>Toggles the target device state. Should not be used to control groups. |

### 6.7.5. Level Control Cluster (Client)

This cluster uses the binding table for managing command targets. 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 one or two push-buttons.

Cluster commands transmitted:

| Command | Description |
|---|---|
| 0x05 | **Move with on/off**<br>Moves the level either up or down at a specified rate.<br>Maintains the companion on/off cluster's OnOff attribute. |
| 0x07 | **Stop with on/off**<br>Stops any level change in progress due to a move, move to level, step or recall scene command.<br>Maintains the companion on/off cluster's OnOff attribute. |

## 6.8. Application Endpoint #5 – Primary Window Covering Controller

This is a Finding & Binding initiator endpoint.

### 6.8.1. Basic Cluster (Server)

Please refer to Section 6.10.1 for details.

### 6.8.2. Identify Cluster (Server)

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 | **IdentifyTime**<br>The remaining time, in seconds, the device will be identifying itself.<br>Notice that C4 currently does not provide any visual feedback in identify mode. |

Cluster commands supported:

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

### 6.8.3. Scenes Cluster (Client)

This cluster is available on devices running firmware version 1.01 and above.

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 | **Recall Scene**<br>Recalls a scene. The group address in the recall scene command payload is also used as the target for the command. |

### 6.8.4. Window Covering Cluster (Client)

This cluster uses the binding table for managing command targets. 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 | **Move up/open**<br>Makes the target device or group of devices move up, towards the fully open position. |
| 0x01 | **Move down/close**<br>Makes the target device or group of devices move down, towards the fully closed position. |
| 0x02 | **Stop**<br>Stops all motion. |

## 6.9. Application Endpoint #6 – Secondary Window Covering Controller

This is a Finding & Binding initiator endpoint.

### 6.9.1. Basic Cluster (Server)

Please refer to Section 6.10.1 for details.

### 6.9.2. Identify Cluster (Server)

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 | **IdentifyTime**<br>The remaining time, in seconds, the device will be identifying itself.<br>Notice that C4 currently does not provide any visual feedback in identify mode. |

Cluster commands supported:

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

### 6.9.3. Scenes Cluster (Client)

This cluster is available on devices running firmware version 1.01 and above.

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 | **Recall Scene**<br>Recalls a scene. The group address in the recall scene command payload is also used as the target for the command. |

### 6.9.4. Window Covering Cluster (Client)

This cluster uses the binding table for managing command targets. 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 | **Move up/open**<br>Makes the target device or group of devices move up, towards the fully open position. |
| 0x01 | **Move down/close**<br>Makes the target device or group of devices move down, towards the fully closed position. |
| 0x02 | **Stop**<br>Stops all motion. |

## 6.10. 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) | **Basic**<br>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) [14] | **Identify**<br>Allows to identify targets during Finding & Binding push-button commissioning. |
| 0x0015 | Inbound (Server) | **Commissioning**<br>Offers fine-grained commissioning options, for example, configuring a certain short address, channel mask, concentrator setup etc. This cluster is rarely used. |

| Cluster | Direction | Description |
|---|---|---|
| 0x0019 | Outbound (Client) | **OTA Upgrade**<br>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) | **Device Setup**<br>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.) |

### 6.10.1. 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,<br>read-only | **ZCLVersion**<br>The version of the cluster library |
| 0x0001 | unsigned8,<br>read-only | **ApplicationVersion**<br>The application version |
| 0x0002 | unsigned8,<br>read-only | **StackVersion**<br>The stack version |
| 0x0003 | unsigned8,<br>read-only | **HWVersion**<br>Specifies the hardware revision number |
| 0x0004 | string,<br>read-only | **ManufacturerName**<br>“ubisys” |
| 0x0005 | string,<br>read-only | **ModelIdentifier**<br>“C4 (5504)” |
| 0x0006 | string,<br>read-only | **DateCode**<br>“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. |
| 0x0007 | enum8,<br>read-only | **PowerSource**<br>Always set to mains-powered, single phase (regardless of the connected supply voltage) |
| 0x0010 | string,<br>persistent | **LocationDescription**<br>Empty string by default. Might be set by commissioning software during installation |
| 0x0011 | unsigned8,<br>persistent | **PhysicalEnvironment**<br>“Unspecified” by default. Might be set by commissioning software during installation |

Cluster commands supported:

| Command | Description |
|---|---|
| 0x00 | **Reset to factory defaults.**<br>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.<br>Notice: In earlier versions, up to and including version 1.02, the command did actually behave like a leave command with factory reset. This command has been deprecated and is no longer supported since version 1.03. Use mgmt_leave_req, instead. |

### 6.10.2. 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 | **Query Identify**<br>This command is broadcast during push-button commissioning to identify suitable binding targets.<br>Once target devices are selected, they will be queried for matching descriptors. |

### 6.10.3. 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.

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

The image type for C4 is 0x7B09. 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 control unit C4 together with the Zigbee/Ethernet Gateway ubisys G1, the latest firmware for C4 will always be available automatically and you do not need to request it explicitly.

### 6.10.5. Device Setup Cluster

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 | **InputConfigurations**<br>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 | **InputActions**<br>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). |

#### 6.10.5.1. 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 C4 this relation is as follows:

| Element # | Description |
|---|---|
| 0x0000 | **Configuration options for high-voltage physical input #1**<br>Defines the configuration for the physical input marked with a switch symbol and the digit “1”.<br>Factory default value: 0x00 |
| 0x0001 | **Configuration options for high-voltage physical input #2**<br>Defines the configuration for the physical input marked with a switch symbol and the digit “2”.<br>Factory default value: 0x00 |
| 0x0002 | **Configuration options for high-voltage physical input #3**<br>Defines the configuration for the physical input marked with a switch symbol and the digit “3”.<br>Factory default value: 0x00 |
| 0x0003 | **Configuration options for high-voltage physical input #4**<br>Defines the configuration for the physical input marked with a switch symbol and the digit “4”.<br>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 in inactive signal. If this bit is set, the input is “active-low”, instead.<br>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. |

### 6.10.5.2. 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.<br>Acceptable endpoints for C4: Endpoints #1, #2, #3, #4, #5 and #6. |
| ClusterID | unsigned16 | The 16-bit cluster ID that will be used to transmit the ZCL command payload, e.g. 0x0006 for the on/off cluster, 0x0008 for level control, etc. |
| 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.<br>Notice that this is the raw payload and does not include a leading length byte. |

### 6.10.5.3. 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 C4, this field may take the values 0, 1, 2, and 3.

### 6.10.5.4. 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. |

| Field | Bit(s) | Description |
|---|---|---|
| Initial State | #3…#2 (0x0C) | These bits indicate the initial state of the input:<br>00b: Ignore: Don't care<br>01b: Pressed: The input was initially active (for less than a second)<br>10b: Kept pressed: The input has initially been active (for more than a second)<br>11b: Released. The input was initially inactive |
| Final State | #1…#0 (0x03) | These bits indicate the final state of the input:<br>00b: Ignore: Don't care<br>01b: Pressed: The input is active (for less than a second)<br>10b: Kept pressed: The input is still active (for more than a second)<br>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.

#### 6.10.5.5. 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).

#### 6.10.5.6. 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.

#### 6.10.5.7. 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 C4 will use its binding table 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).

#### 6.10.5.8. Examples

This is the complete default configuration for C4, which assigns each input as a push-button (momentary, one stable position) to a corresponding on/off cluster instance on the primary, secondary, tertiary, and quaternary level control switch endpoint:

```
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
01          Source: Endpoint #1 (hosts primary on/off client cluster on C4)
06 00       Cluster ID: 0x0006 - on/off
02          ZCL Command Template: Toggle

06          element #2: six bytes
01          InputAndOptions: 0x01
0D          Transition: released -> pressed
02          Source: Endpoint #2 (hosts secondary on/off client cluster on C4)
06 00       Cluster ID: 0x0006 - on/off
02          ZCL Command Template: Toggle

06          element #3: six bytes
02          InputAndOptions: 0x02
0D          Transition: released -> pressed
03          Source: Endpoint #3 (hosts tertiary on/off client cluster on C4)
06 00       Cluster ID: 0x0006 - on/off
02          ZCL Command Template: Toggle

06          element #4: six bytes
03          InputAndOptions: 0x03
0D          Transition: released -> pressed
04          Source: Endpoint #4 (hosts quaternary on/off client cluster on C4)
06 00       Cluster ID: 0x0006 - on/off
02          ZCL Command Template: Toggle
```

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 and asking for these recommended instructions for typical use cases. Avoid using other instructions.

## 6.11. 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 | Outbound (Client) | **Zigbee Green Power Proxy**<br>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. |

### 6.11.1. 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 [15] GPDs.

Attributes supported:

| Attribute | Type | Description |
|---|---|---|
| 0x0010 | unsigned8, read-only | **gpsMaxSinkTableEntries**<br>The number of proxy table entries supported by this device |
| 0x0011 | extended raw binary, read-only, persistent | **ProxyTable**<br>Entries in the proxy table create a link between Green Power Devices and Green Power Sinks |
| 0x0016 | bitmap24, read-only | **gppFunctionality**<br>Indicates Green Power features and building blocks supported by this device |
| 0x0017 | bitmap24, read-only | **gpsActiveFunctionality**<br>Allows to disable certain Green Power features on this device |
| 0x0020 | bitmap8, persistent | **gpSharedSecurityKeyType**<br>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 | **gpSharedSecurityKey**<br>The 128-bit AES-CCM* key that is being used to secure Green Power data frames |
| 0x0022 | key128, persistent | **gpLinkKey**<br>The 128-bit AES-CCM* key that is being used to deliver keying material to Green Power devices |

Cluster commands supported:

| Command | Description |
|---|---|
| 0x01 | **GP Pairing**<br>Creates, updates or removes proxy table entries |
| 0x02 | **GP Proxy Commissioning Mode**<br>Makes the proxy enter commissioning mode for a particular sink, or leave commissioning mode |
| 0x06 | **GP Response**<br>Tunnels GP data frames from a sink to a bidirectional Green Power Device |
| 0x0B | **GP Proxy Table Request**<br>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 | **GP Notification**<br>Tunnels GP frames from a Green Power Device to one or more sinks or groups of sinks |
| 0x04 | **GP Commissioning Notification**<br>Tunnels GP frames from a Green Power Device to a sink in commissioning mode |
| 0x0B | **GP Proxy Table Response**<br>Conveys a set of proxy table entries to a sink or management application |

[8] Since application version 1.01. Prior, device ID 0x0008 (Range Extender) was used.

[9] Since application version 1.04.

[10] Available in Zigbee stack version 1.60 and above. Legacy ZCP requirement – do not use in applications.

[11] Available in Zigbee stack version 1.56 and above.

[12] Available in Zigbee stack version 1.50 and above.

[13] Available in Zigbee stack version 1.61 and above.

[14] Since application version 1.01. Prior, the server side of identify was available.

[15] Bidirectional communication is currently limited to the commissioning stage

# 7. Physical Dimension

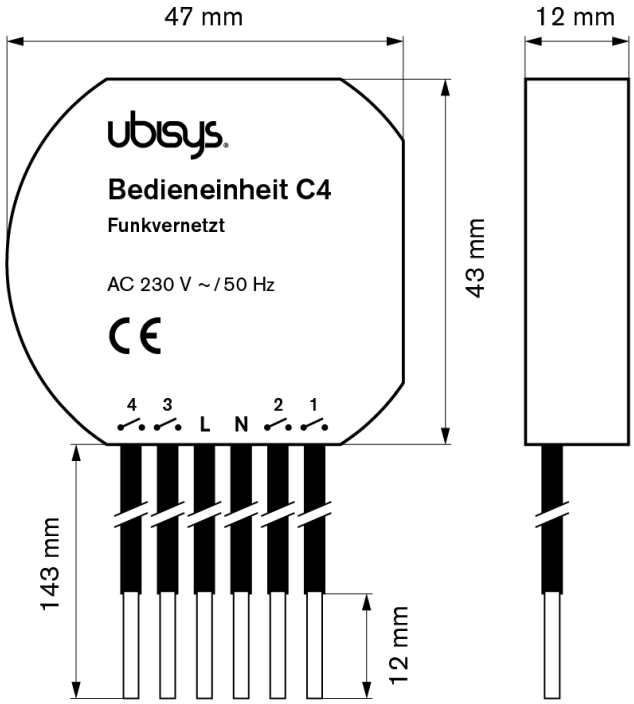


*Figure 1. Flush-mount In-wall Zigbee Control Unit C4 – Dimensions*

# 8. 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.

| Vendor/Brand | Order Code | Description |
|---|---|---|
| ubisys | 1120 | Zigbee Control Unit C4 (in-wall, flush-mount) |

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

## 9. 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:

www.ubisys.de/en/smarthome/terms.html

# 10. Declaration Of Conformity

We – ubisys technologies GmbH, Neumannstraße 10, 40235 Düsseldorf, Germany – declare under our sole responsibility that the ubisys Zigbee Control Unit C4 with order codes as detailed in Chapter 8 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) |
| 2014/30/EU | Electromagnetic Compatibility Directive (EMC) |
| 2014/35/EU | Low Voltage Directive (LVD) |
| 2012/19/EU | Waste Electrical and Electronic Equipment Directive (WEEE) |
| 2011/65/EU | Restriction of Hazardous Substances Directive (RoHS) |
| EN IEC 60669-2-1:2022+A11:2022 | Switches for household and similar fixed electrical installations - Part 2-1: Particular requirements - Electronic control devices (IEC 60669-2-1:2021) |
| EN 50663:2017 | Generic standard for assessment of low power electronic and electrical equipment related to human exposure restrictions for electromagnetic fields (10 MHz - 300 GHz) |
| EN 62479:2010 | Assessment of the compliance of low power electronic and electrical equipment with the basic restrictions related to human exposure to electromagnetic fields (10 MHz to 300 GHz) |
| EN IEC 61000-3-2:2019+A1:2021 | Electromagnetic compatibility (EMC) - Part 3-2: Limits - Limits for harmonic current emissions (equipment input current ≤16 A per phase) |
| EN 61000-3-3:A2:2013+A1:2019+A2:2021 | Electromagnetic compatibility (EMC) - Part 3-3: Limits - Limitation of voltage changes, voltage fluctuations and flicker in public low-voltage supply systems, for equipment with rated current ≤16 A per phase and not subject to conditional connection |
| ETSI EN 301 489-1 V2.2.3 (2019-11) | ElectroMagnetic Compatibility (EMC) standard for radio equipment and services; Part 1: Common technical requirements; Harmonised Standard for ElectroMagnetic Compatibility |
| ETSI EN 301 489-17 V3.2.4 (2020-09) | ElectroMagnetic Compatibility (EMC) standard for radio equipment and services; Part 17: Specific conditions for Broadband Data Transmission Systems; Harmonised Standard for ElectroMagnetic Compatibility |
| ETSI EN 300 328 V2.2.2 (2019-07) | Wideband transmission systems; Data transmission equipment operating in the 2,4 GHz band; Harmonised Standard for access to radio spectrum |
| IEEE 802.15.4:2020 | IEEE Standard 802 – Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) |
| Zigbee 3.0 | Zigbee 2017 with Green Power |

| | |
|---|---|
| Düsseldorf, Germany | October 29, 2025 |
| Place of issue | Date of issue |

| | |
|---|---|
| Dr.-Ing. Arasch Honarbacht | Managing Director, Head of Research & Development |
| Full name of Authorized Signatory | Title of Authorized Signatory |
| | |
| Signature | Seal |

# 11. Revision History

| Revision | Date | Remarks |
|---|---|---|
| 1.0 | 11/25/2014 | Initial Public Version |
| 1.1 | 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 1.01 and above. |
| 1.2 | 02/13/2015 | Feature overview updated to include a short overview of input configuration options |
| 1.3 | 08/12/2015 | Improved network steering in firmware revision 1.02 (toggle permit joining) and shortcut to factory reset by 10-second extended press. |
| 1.4 | 11/02/2015 | Starting with firmware revision 1.03, when the device is factory reset, it will broadcast a network leave indication |
| 1.5 | 11/11/2015 | Updated the ZDO description to include enhancements and additions made for Zigbee 2015 platform compliance |
| 1.6 | 11/17/2015 | Enhancements made in firmware revision 1.04 include added support for Zigbee Green Power and further changes to align with Zigbee 3.0 requirements. |
| 1.7 | 09/13/2016 | Updated to Zigbee 3.0 Certified Product in application firmware 1.05; added information about the power-cycle sequence factory reset in application firmware 1.06. Corrected description of element data type for switch configuration (should have been 0x41 instead of 0x48). |
| 1.8 | 09/16/2021 | Update Declaration of Conformity |
| 1.9 | 10/29/2025 | Update Declaration of Conformity |

## 12. Contact

ubisys technologies GmbH
Neumannstr. 10
40235 Düsseldorf
Germany

T: +49. 211. 54 21 55 - 19
F: +49. 211. 54 21 55 - 99

www.ubisys.de
info@ubisys.de