OCF Resource to Zigbee Cluster Mapping Specification

VERSION 2.2.6 | October 2022

CONTACT admin@openconnectivity.org

Legal Disclaimer

CONTENTS

Tables

Introduction

This document, and all the other parts associated with this document, were developed in response to worldwide demand for smart home focused Internet of Things (IoT) devices, such as appliances, door locks, security cameras, sensors, and actuators; these to be modelled and securely controlled, locally and remotely, over an IP network.

While some inter-device communication existed, no universal language had been developed for the IoT. Device makers instead had to choose between disparate frameworks, limiting their market share, or developing across multiple ecosystems, increasing their costs. The burden then falls on end users to determine whether the products they want are compatible with the ecosystem they bought into, or find ways to integrate their devices into their network, and try to solve interoperability issues on their own.

In addition to the smart home, IoT deployments in commercial environments are hampered by a lack of security. This issue can be avoided by having a secure IoT communication framework, which this standard solves.

The goal of these documents is then to connect the next 25 billion devices for the IoT, providing secure and reliable device discovery and connectivity across multiple OSs and platforms. There are multiple proposals and forums driving different approaches, but no single solution addresses the majority of key requirements. This document and the associated parts enable industry consolidation around a common, secure, interoperable approach.

The OCF specification suite is made up of nineteen discrete documents, the documents fall into logical groupings as described herein:

- Core framework
  - Core Specification
  - Security Specification
  - Onboarding Tool Specification
- Bridging framework and bridges
  - Bridging Specification
  - Resource to Alljoyn Interface Mapping Specification
  - OCF Resource to oneM2M Resource Mapping Specification
  - OCF Resource to BLE Mapping Specification
  - OCF Resource to EnOcean Mapping Specification
  - OCF Resource to LWM2M Mapping Specification
  - OCF Resource to UPlus Mapping Specification
  - OCF Resource to Zigbee Cluster Mapping Specification
  - OCF Resource to Z-Wave Mapping Specification
- Resource and Device models
  - Resource Type Specification
  - Device Specification
- Core framework extensions
  - Easy Setup Specification
  - Core Optional Specification
- OCF Cloud
  - Cloud API for Cloud Services Specification

– Device to Cloud Services Specification
– Cloud Security Specification

OCF Resource to Zigbee Cluster Mapping Specification

1 Scope

This document provides detailed mapping information between Zigbee defined Clusters and OCF defined Resources.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 30118-1 Information technology -- Open Connectivity Foundation (OCF) Specification -- Part 1: Core specification
https://www.iso.org/standard/53238.html
Latest version available at: https://openconnectivity.org/specs/OCF_Core_Specification.pdf

ISO/IEC 30118-2 Information technology – Open Connectivity Foundation (OCF) Specification – Part 2: Security specification
https://www.iso.org/standard/74239.html
Latest version available at: https://openconnectivity.org/specs/OCF_Security_Specification.pdf

ISO/IEC 30118-3 Information technology – Open Connectivity Foundation (OCF) Specification – Part 3: Bridging specification
https://www.iso.org/standard/74240.html
Latest version available at: https://openconnectivity.org/specs/OCF_Bridging_Specification.pdf

ISO/IEC 30118-4 Information technology – Open Connectivity Foundation (OCF) Specification – Part 4: Resource type specification
https://www.iso.org/standard/74241.html
Latest version available at:
https://openconnectivity.org/specs/OCF_Resource_Type_Specification.pdf

ISO/IEC 30118-5 Information technology – Open Connectivity Foundation (OCF) Specification – Part 5: Smart home device specification
https://www.iso.org/standard/74242.html
Latest version available at: https://openconnectivity.org/specs/OCF_Device_Specification.pdf

Derived Models for Interoperability between IoT Ecosystems, Stevens & Merriam, March 2016
https://www.iab.org/wp-content/IAB-uploads/2016/03/OCF-Derived-Models-for-Interoperability-Between-IoT-Ecosystems_v2-examples.pdf

Zigbee, *Zigbee Specification*, August 2015
http://www.zigbee.org/zigbee-for-developers/zigbee-3-0/

Zigbee Cluster Library Specification, Version 1.0
http://www.zigbee.org/zigbee-for-developers/zigbee-3-0/

ZigBee Lighting & Occupancy Device, Version 1.0
http://www.zigbee.org/zigbee-for-developers/zigbee-3-0/

3 Terms, definitions symbols and abbreviations

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 30118-1, ISO/IEC 30118-2, and ISO/IEC 30118-3 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

– ISO Online browsing platform: available at https://www.iso.org/obp
– IEC Electropedia: available at http://www.electropedia.org/

3.1.1
Zigbee Attribute
data entity which represents a physical quantity or state within Zigbee.

3.1.2
Zigbee Cluster
one or more *Zigbee Attributes* (3.1.1), commands, behaviours, and dependencies, which supports an independent utility or application function.

3.1.3
Zigbee Server
cluster interface which is listed in the input cluster list of the simple descriptor on an endpoint.

3.1.4
Zigbee 3.0 Server
Zigbee Server (3.1.3) which is built on Zigbee 3.0 stack

3.1.5
Zigbee Client
cluster interface which is listed in the output cluster list of the simple descriptor on an endpoint.

3.1.6
Zigbee 3.0 Client
Zigbee Client (3.1.5) which is built on Zigbee 3.0 stack

3.1.7
Zigbee Device
unique device identifier and a set of mandatory and optional clusters to be implemented on a single Zigbee endpoint.

3.1.8
Zigbee 3.0 Device
Zigbee Device (3.1.7) which is built on Zigbee 3.0 stack

4 Document conventions and organization

4.1 Conventions

In this document a number of terms, conditions, mechanisms, sequences, parameters, events, states, or similar terms are printed with the first letter of each word in uppercase and the rest lowercase (e.g., Network Architecture). Any lowercase uses of these words have the normal technical English meaning.

In this document, to be consistent with the IETF usages for RESTful operations, the RESTful operation words CRUDN, CREATE, RETRIVE, UPDATE, DELETE, and NOTIFY will have all letters capitalized. Any lowercase uses of these words have the normal technical English meaning.

4.2 Notation

In this document, features are described as required, recommended, allowed or DEPRECATED as follows:

Required (or shall or mandatory).

> These basic features shall be implemented to comply with the Mapping Specification. The phrases “shall not”, and “PROHIBITED” indicate behavior that is prohibited, i.e. that if performed means the implementation is not in compliance.

Recommended (or should).

> These features add functionality supported by the Mapping Specification and should be implemented. Recommended features take advantage of the capabilities the Mapping Specification, usually without imposing major increase of complexity. Notice that for compliance testing, if a recommended feature is implemented, it shall meet the specified requirements to be in compliance with these guidelines. Some recommended features could become requirements in the future. The phrase “should not” indicates behavior that is permitted but not recommended.

Allowed (or allowed).

> These features are neither required nor recommended by the Mapping Specification, but if the feature is implemented, it shall meet the specified requirements to be in compliance with these guidelines.

Conditionally allowed (CA)

> The definition or behaviour depends on a condition. If the specified condition is met, then the definition or behaviour is allowed, otherwise it is not allowed.

Conditionally required (CR)

> The definition or behaviour depends on a condition. If the specified condition is met, then the definition or behaviour is required. Otherwise the definition or behaviour is allowed as default unless specifically defined as not allowed.

DEPRECATED

> Although these features are still described in this document, they should not be implemented except for backward compatibility. The occurrence of a deprecated feature during operation of an implementation compliant with the current document has no effect on the implementation’s operation and does not produce any error conditions. Backward compatibility may require that a feature is implemented and functions as specified but it shall never be used by implementations compliant with this document.

Strings that are to be taken literally are enclosed in “double quotes”.

Words that are emphasized are printed in *italic*.

5 Theory of operation

5.1 Interworking approach

The interworking between ZigBee Clusters and OCF defined Resources is modelled using the derived model syntax described in Derived Models for Interoperability between IoT Ecosystems.

5.2 Mapping syntax

5.2.1 Introduction

Within the defined syntax for derived modelling used by this document there are two blocks that define the actual Property-Property equivalence or mapping. These blocks are identified by the keywords "x-to-ocf" and "x-from-ocf". Derived Models for Interoperability between IoT Ecosystems does not define a rigid syntax for these blocks; they are free form string arrays that contain pseudo-coded mapping logic.

Within this document we apply the rules in defined in clause 5.2 to these blocks to ensure consistency and re-usability and extensibility of the mapping logic that is defined.

5.2.2 General

All statements are terminated with a carriage return.

5.2.3 Value assignment

The equals sign (=) is used to assign one value to another. The assignee is on the left of the operator; the value being assigned on the right.

5.2.4 Property naming

All Property names are identical to the name used by the original model; for example, from the OCF Temperature Resource the Property name "temperature" is used whereas when referred to the derived ecosystem then the semantically equivalent Property name is used.

The name of the OCF defined Property is prepended by the ecosystem designator "ocf" to avoid ambiguity (e.g. "ocf.step")

5.2.5 Range

The range on the OCF side is fixed.

5.2.6 Arrays

An array element is indicated by the use of square brackets "[]" with the index of the element contained therein, e.g. range [1]. All arrays start at an index of 0.

5.2.7 Default mapping

There are cases where the specified mapping is not possible as one or more of the Properties being mapped is optional in the source model. In all such instances a default mapping is provided. (e.g. "transitiontime = 1")

5.2.8 Conditional mapping

When a mapping is dependent on the meeting of other conditions then the syntax:

If "condition", then "mapping".

is applied.

E.g. if onoff = false, then ocf.value = false

5.2.9 Method invocation

The invocation of a command from the derived ecosystem as part of the mapping from an OCF Resource is indicated by the use if a double colon "::" delimiter between the applicable resource, service, interface or other construct identifier and the command name. The command name always includes trailing parentheses which would include any parameters should they be passed.

For example, when dealing with the "on()" command for Zigbee On/off Cluster this gives a complete command invocation as: "zb.command.onoff::on()".

6 Zigbee translation

6.1 Operational scenarios

The overall goal is to make Bridged Zigbee 3.0 Servers appear to OCF Clients as if they were native OCF Servers in the local network or cloud environment

The mapping between the OCF data models and Zigbee Clusters is specified in 9. Programmatic (i.e. On-the-fly) data model translation is not supported.

Figure 1 shows an overview of a Zigbee 3.0 Bridge Platform and its general topology. It exposes Zigbee 3.0 Servers to OCF Clients. Each Bridged Zigbee 3.0 Server is represented as a Virtual OCF Server. The Zigbee 3.0 Bridging Function supports Asymmetric bridging. The scope of this document is the asymmetric bridging to expose the Zigbee Server to OCF. The asymmetric bridging to expose an OCF Server to a Zigbee Client is out of scope.

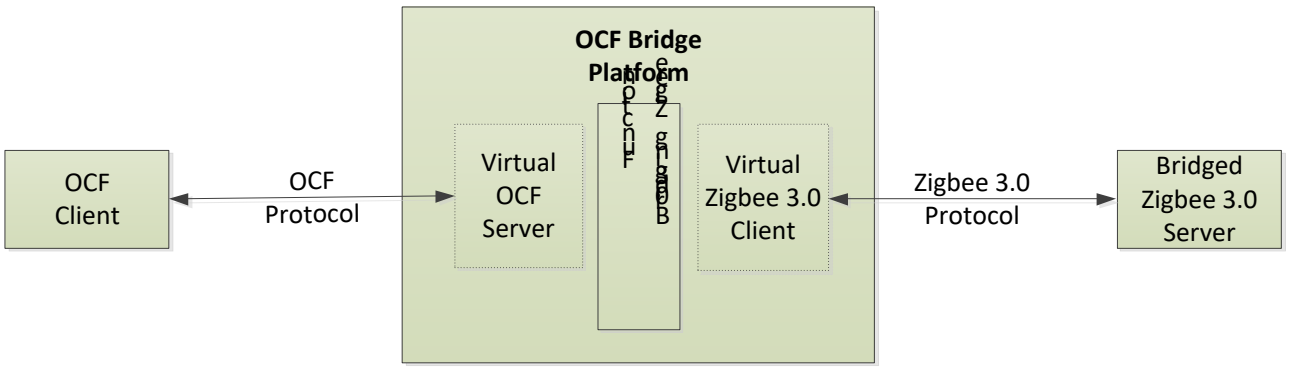

Figure 1 – OCF Zigbee Bridge Platform and Components

6.2 Requirements specific to Zigbee bridging function

6.2.1 Requirements specific to Zigbee

This document refers to Zigbee 3.0 or higher. Zigbee 3.0 is built on Zigbee Pro 2015 or newer, which enhances the IEEE 802.15.4 standard by adding a mesh network and security layers along with an application framework. Low power support is not the scope of this document.

An OCF Zigbee Bridging Function shall act as a Zigbee Coordinator in network layer. A Zigbee Coordinator is responsible for initiating and maintaining the devices on the network. An OCF Zigbee Bridge Platform will act as Zigbee Client towards the Zigbee 3.0 Devices in the application layer. Users can expect that a certified OCF Bridge Platform will be able to talk to Zigbee 3.0 Devices, without the user having to buy some other device.

6.2.2 Exposing Zigbee 3.0 servers to OCF clients

The nature of how Zigbee Devices are structured may be different than how an OCF Device is structured. The mapping of the structure of a Zigbee device on an OCF Device is given by Table 1.

A Zigbee Server cluster may map to one or more OCF Resources. If a specific Zigbee Server cluster has specific commands, one or more OCF Resources corresponding to the specific command attributes may be additionally needed.

A Zigbee Attribute of a Zigbee Server cluster typically maps to an OCF Resource Property. However, in some special cases, multiple attributes are mapped to a single OCF Resource Property e.g., "CurrentX" and "CurrentY" of the Zigbee color control cluster map to the "csc" Property in the "oic.r.colour.csc" (Colour Space Coordinates) Resource because of the difference in the data types, i.e., "csc" is an array, but "CurrentX" and "CurrentY" map to a number.

Table 2 is a mapping example of this rule

Table 1 – Translation Rule between Zigbee and OCF Data Models

From Zigbee	mapping count	To OCF	mapping count
Zigbee Device	**1**	OCF Device	**1**
Zigbee Cluster	**1**	OCF Resource	**n**
Zigbee Attribute	**1**	OCF Resource Property	**1**

Table 2 – Zigbee to OCF Mapping Example (Color Temperature Light)

From Zigbee		To OCF	
Zigbee 3.0 Device	0x010c (Color Temperature Light)	OCF Device	oic.d.light (Light)
Zigbee Server Cluster	0x0006 (On/Off)	OCF Resource(s)	oic.r.switch.binary (Binary Switch)
	0x0300 (Color Control Cluster)		oic.r.colour.hs (Colour Hue and Saturation)
			oic.r.colour.csc (Colour Space Coordinates)
			oic.r.colour.colourtemperature (Colour Temperature)
Zigbee Attribute	0x0000 (OnOff of On/Off Cluster)	OCF Resource Property	value (of Binary Switch Resource)
	0x0003 (CurrentX of Color Control Cluster)		csc (of Colour Space Coordinates)
	0x0004 (CurrentY of Color Control Cluster)		

If a Zigbee 3.0 Device, Zigbee Server Cluster, Zigbee Attribute are enlisted in the well-defined set, the Bridging Function shall follow the requirements for translating it to an OCF Device, OCF Resource, or OCF Resource Property (i.e. "deep translation").

A Zigbee 3.0 Server Device maps to a single OCF Device Type. The OCF Device Type is provided by using the Device ID of the Zigbee 3.0 Server Device (The Device ID is allocated by the Zigbee Alliance and has the same meaning of the OCF Device Type). The Zigbee 3.0 Bridging Function has a table which includes the mapping information between the Zigbee Device ID and the OCF Device Type. Based on the table, the Zigbee 3.0 Bridging Function finds the OCF Device Type according to the Zigbee Device ID.

A Zigbee Device includes one or more Zigbee Server Clusters. If a Zigbee Cluster maps to multiple OCF Resources, the Zigbee Cluster may be translated as a Resource with a Collection Resource Type. The resource mapping between Zigbee Server Cluster and OCF Resources is defined in 9 for deep translation. The Zigbee 3.0 Bridging Function has a table which includes the mapping information between the identifier of Zigbee Cluster and OCF Resource Type(s). The Zigbee 3.0 Bridging Function obtains the list of cluster identifiers after the Virtual Zigbee 3.0 Client and Zigbee 3.0 Server Device are bound. Based on the table, the Zigbee 3.0 Bridging Function finds the OCF Resource Type(s) according to the identifier of Zigbee Cluster.

Since a Bridging Function knows all relationships between OCF Resources and Zigbee Server Clusters, the path component of URI can be free to choose. Maintaining relationship information and URI definition is implementation specific.

If a Zigbee operation fails, the Bridging Function send an appropriate OCF error response to the OCF Client. it constructs an appropriate OCF error message (e.g., diagnostic payload if using CoAP) from the Zigbee enumerated status value and Zigbee enumerated status (if any), using the form "<error name>: <error message>", with the <error name> taken from the Zigbee Status Code field and the <error message> taken from the Zigbee enumerated status, and the error code for the OCF network set to an appropriate value.

6.2.3 Translation for well-defined set

If a Zigbee 3.0 Device, Zigbee Server Cluster, Zigbee Attribute are enlisted in the well-defined set, the Bridging Function shall follow the requirements for translating it to an OCF Device, OCF Resource, or OCF Resource Property (i.e., "deep translation"). Table 3 is the list of Zigbee 3.0 devices and mandatory Zigbee Server Clusters with corresponding OCF devices and mandatory OCF Resources. Optional OCF Resources mapped with the specific Zigbee Server Clusters are enlisted in the well-defined set.

Table 3 – Zigbee 3.0 Device & Cluster – OCF Device & Resource mapping

Zigbee 3.0 Device Name (Device ID)	Zigbee 3.0 Mandatory Cluster	OCF Mandatory Resource Type	OCF Device Type (“rt”)	OCF Device Name
On/off light (0x0100)	On/off	oic.r.switch.binary,	oic.d.light	Light
Color Temperature Light (0x010c)	On/off, Level Control, Color Control	oic.r.switch.binary,	oic.d.light	Light

Extended Color Light (0x010d)	On/off, Level Control, Color Control	oic.r.switch.binary,	oic.d.light	Light
Dimmable Light (0x0101)	On/off, Level Control	oic.r.switch.binary,	oic.d.light	Light
Color Dimmable Light (0x0102)	On/off Level Control, Color Control	oic.r.switch.binary,	oic.d.light	Light
Temperature Sensor (0x0302)	Temperature Measurement	oic.r.temperature	oic.d.sensor	Generic Sensor
Thermostat (0x0301)	Thermostat	oic.r.temperature(2)	oic.d.thermostat	Thermostat
Window Covering Device (0x0202)	Window Covering	oic.r.openlevel	oic.d.blind	Blind
Smart Plug (0x0051)	On/off, Metering	oic.r.switch.binary,	oic.d.smartplug	Smart Plug
Mains Power Outlet (0x0009)	On/off	oic.r.switch.binary,	oic.d.smartplug	Smart Plug
On/off output (0x0002	On/off	oic.r.switch.binary,	oic.d.smartplug	Smart Plug
IAS Zone (0x0402)	IAS Zone	oic.r.ias.zone	oic.d.sensor	Generic Sensor
Occupancy Sensor (0x0107)	Occupancy Sensing	oic.r.sensor.presence	oic.d.sensor	Generic Sensor

6.2.4 Exposing a Zigbee 3.0 server as a virtual OCF server

Table 4 shows how OCF Platform properties, as specified in ISO/IEC 30118-1, shall be derived, typically from fields of Descriptor specified in Zigbee.

Table 4 – "oic.wk.p" Resource Type mapping

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Platform ID	pi	Unique identifier for the physical platform (UIUID); this shall be a UUID in accordance with IETF RFC 4122. It is recommended that the UUID be created using the random generation scheme (version 4 UUID) specific in the RFC.	Y	(none)	Bridging Function should return a randomly-generated UUID (Please see section 4.4 of IETF RFC 4122 for randomly-generated UUID)	
Manufacturer Name	mnmn	Name of manufacturer (not to exceed 16 characters)	Y	Manufacturer name (in DefaultLanguage, truncated to 16 characters)	Name of the manufacturer as a ZigBee character string Defined in Basic Cluster	Y
Manufacturer Details Link (URL)	mnml	URL to manufacturer (not to exceed 32 characters)	N	(none)	(none)	N
Model Number	mnmo	Model number as designated by manufacturer	N	Model Identifier	Model number (or other identifier) assigned by the manufacturer as a ZigBee character string Defined in Basic Cluster	Y
Date of Manufacture	mndt	Manufacturing date of device	N	DateCode	Date of manufacturer of the device in international date notation according to ISO 8601, i.e., YYYYMMDD, Defined in Basic Cluster	N
Platform Version	mnpv	Version of platform – string (defined by manufacturer)	N	(none)	(none)	N
OS Version	mnos	Version of platform resident OS – string (defined by manufacturer)	N	(none)	(none)	N
Hardware Version	mnhw	Version of platform hardware	N	HWVersion	Version number of the hardware of the device. Defined in Basic Cluster	N
Firmware version	mnfv	Version of device firmware	N	(none)	(none)	N

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Support link	mnsl	URI that points to support information from manufacturer	N	ProductURL	Link to a web page containing specific product information Defined in Basic Cluster	N
SystemTime	st	Reference time for the device	N	(none)	(none)	N
Vendor ID	vid	Vendor defined string for the platform. The string is freeform and up to the vendor on what text to populate it.	N	(none)	(none)	N

Table 5 shows how OCF Device Properties, as specified in Table 20 in ISO/IEC 30118-1, shall be derived, typically from fields of Descriptor or Attributes of Basic cluster specified in Zigbee and Zigbee Cluster Library Specification, respectively.

As specified in ISO/IEC 30118-2, the value of the “di” Property of OCF Devices (including Virtual OCF Devices) shall be established as part of Onboarding of that Virtual OCF Device.

Table 5 – "oic.wk.d" Resource Type mapping

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
(Device) Name	n	Human friendly name For example, “Bob’s Thermostat”	Y	User description if it exists, else Model Name if it exists, else translate Application Device Identifier (=Device ID)to Human friendly name by using Application Device Identifier value/description table	User description : Information that allows the user to identify the device using a user-friendly character string, such as “Bedroom TV” Defined in User Descriptor Model Name : character string representing the name of the manufacturer’s model of the device Defined in Complex Descriptor Application Device Identifier: device description supported on this endpoint Cluster Defined in Simple Descriptor	User description: N Model Name: N Application Device Identifier: Y
Spec Version	icv	Spec version of the core specification this device is implemented to,	Y	(none)	Spec version of the core specification that the Bridging Platform	

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
		The syntax is "core.major.minor"]			implements should return its own value	
Device UUID	di	Unique identifier for Device. This value shall be as defined in OCF Security Specification for Device UUID.	Y	(none)	Use as defined in the ISO/IEC 30118-2	
Protocol-Independent ID	piid	Unique identifier for OCF Device (UUID)	Y	(none)	Bridging Function should return a randomly-generated UUID (Please see section 4.4 of IETF RFC 4122 for randomly-generated UUID)	
Data Model Version	dmv	Spec version(s) of the vertical specifications this device data model is implemented to. The syntax is a comma separated list of "<vertical>.major. minor"]. <vertical> is the name of the vertical (i.e. sh for Smart Home)	Y	(none)	Bridging Function should return its own value.	
Localized Descriptions	ld	Detailed description of the Device, in one or more languages. This property is an array of objects where each object has a "language" field (containing an RFC 5646 language tag) and a "value" field containing the device description in the indicated language.	N	(none)	Zigbee provides Language and Character Set field only which specifies the language and character set used by the character strings by using ISO 639-1 language code	

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
Software Version	sv	Version of the device software.	N	ApplicationVersion	Version number of the application software contained in the device. Defined in Basic Cluster	Y
Manufacturer Name	dmn	Name of manufacturer of the Device, in one or more languages. This property is an array of objects where each object has a "language" field (containing an RFC 5646 language tag) and a "value" field containing the manufacturer name in the indicated language.	N	Manufacturer name	Name of the manufacturer as a ZigBee character string Defined in Basic Cluster	Y
Model Number	dmno	Model number as designated by manufacturer.	N	Model Identifier	Model number (or other identifier) assigned by the manufacturer as a ZigBee character string Defined in Basic Cluster	Y

Table 6 shows how OCF Device Configuration properties, as specified in Table 15 in ISO/IEC 30118-1 shall be derived.

Table 6 – "oic.wk.con" Resource Type mapping

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
(Device) Name	n	Human friendly name For example, “Bob’s Thermostat”	Y	User description if it exists, else Model Name if it exists, else translate Application Device Identifier (=Device ID)to Human friendly name by using Application Device Identifier value/description table	User description; Information that allows the user to identify the device using a user-friendly character string, such as “Bedroom TV” Defined in User Descriptor Model Name; character string representing the name of the manufacturer’s model of the device Defined in Complex Descriptor Application Device Identifier: device description supported on this endpoint Cluster Defined in Simple Descriptor	User description: N Model Name: N Application Device Identifier: Y
Location	loc	Provides location information where available.	N	(none)	(none)	
Location Name	locn	Human friendly name for location For example, “Living Room”.	N	(none)	(none)	
Currency	c	Indicates the currency that is used for any monetary transactions	N	(none)	(none)	
Region	r	Free form text Indicating the current region in which the device is located geographically. The free form text shall not start with a quote (").	N	(none)	(none)	
Localized Names	ln	Human-friendly name of the Device, in one or more languages. This property is an array of objects where each object has a "language" field (containing an RFC 5646 language tag) and a "value" field containing the	N	User description if it exists, else Model Name if it exists, else translate Application Device Identifier (=Device ID)to Human	User description; Information that allows the user to identify the device using a user-friendly character string, such as “Bedroom TV” Defined in User Descriptor Model Name; character string representing the	User description: N Model Name: N Application Device Identifier: Y

To OCF Property title	OCF Property name	OCF Description	OCF Mandatory	From Zigbee 3.0 Field name	Zigbee 3.0 Description	Zigbee 3.0 Mandatory
		device name in the indicated language. If this property and the Device Name (n) property are both supported, the Device Name (n) value shall be included in this array.		friendly name by using Application Device Identifier value/description table	name of the manufacturer's model of the device Defined in Complex Descriptor Application Device Identifier: device description supported on this endpoint Cluster Defined in Simple Descriptor	
Default Language	dl	The default language supported by the Device, specified as an RFC 5646 language tag. By default, clients can treat any string property as being in this language unless the property specifies otherwise.	N	ISO 639-1 language code (if it exists, else property is absent)	Language used for character strings.	N

7 Device type mapping

7.1 Introduction

This clause contains the mappings from Zigbee Device Types to OCF Device Types.

7.2 Zigbee device types to OCF device types

Table 7 captures the equivalency mapping between Zigbee defined Device Types (Please see reference Zigbee Cluster Library Specification) and OCF defined Device Types (please see reference ISO/IEC 30118-5).

Table 7 – Zigbee to OCF Device Type Mapping

Zigbee Device Type	Zigbee Device ID	OCF Device Type
On/off Output	0x0002	oic.d.smartplug
Mains Power Outlet	0x0009	oic.d.smartplug
Smart Plug	0x0051	oic.d.smartplug

On/Off Light	0x0100	oic.d.light
Dimmable Light	0x0101	oic.d.light
Color Dimmable Light	0x0102	oic.d.light
Color Temperature Light	0x010c	oic.d.light
Extended Color Light	0x010d	oic.d.light
Window Covering Device	0x0202	oic.d.blind
Thermostat	0x0301	oic.d.thermostat
Temperature Sensor	0x0302	oic.d.sensor
Occupancy Sensor	0x0107	oic.d.sensor
IAS Zone	0x0402	oic.d.sensor

8 Resource to zigbee cluster equivalence

8.1 Introduction

This clause introduces new Resource Types for mapping between Zigbee Clusters and OCF Resources and lists the complete set of applicable Zigbee Clusters and equivalent OCF Resource Type(s) in clause 8.2

8.2 Zigbee clusters to OCF resources

8.2.1 Introduction

Table 8 captures the equivalency mapping between Zigbee defined Clusters (see Zigbee Cluster Library Specification) and OCF defined Resource Types (see ISO/IEC 30118-4). Detailed Property by Property mappings are provided in clause 8.1.

clause 9 captures the mappings for mandatory server clusters for Zigbee 3.0 devices

Table 8 – Zigbee Server Cluster to OCF Resource Type Mapping

Zigbee Cluster	OCF Resource Type Name	OCF Resource Type ID	OCF Interface(s)
On/off	Binary Switch	oic.r.switch.binary	oic.if.a
Level Control	Dimming	oic.r.light.dimming	oic.if.a
Color Control	Colour Hue and Saturation, Colour Space Coordinates, Colour Temperature	oic.r.colour.hs, oic.r.colour.csc, oic.r.colour.colourtemperature,	oic.if.a
Thermostat	Temperature (3)	oic.r.temperature (3) * 1 for sensor, 2 for heater and cooler	oic.if.s oic.if.a
Window Covering	Window Covering	oic.r.windowcovering, oic.r.openlevel (4) * 2 for lift (percentage scale and cm scale), 2 for tilt (percentage scale and cm scale)	oic.if.rw oic.if.a
Temperature Measurement	Temperature	oic.r.temperature	oic.if.s
Occupancy Sensing	Presence Sensor	oic.r.sensor.presence	oic.if.s
IAS Zone	IAS Zone	oic.r.ias.zone	oic.if.rw

8.2.2 On/off

The APIs with "zcl.onoff" define the mapping between an instance of an OCF Binary Switch Resource and the Zigbee On/off Cluster. In clause 9.14 a RETRIEVE on an OCF Binary Switch Resource maps to a general Read command on a Zigbee On/off Cluster. The value of Zigbee Attribute in Zigbee On/off Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Binary Switch Resource. In clause 9.13 an UPDATE on a Binary Switch maps to a command invocation on either "on()" command or "off()" command of Zigbee On/off Cluster. "value = true" maps to "on()", "value = false" maps to "off()" of Zigbee On/off Cluster.

8.2.3 Level control

The APIs with "zcl.levelcontrol" define the mapping between an instance of an OCF Dimming Resource and the Zigbee Level Control Cluster. In clause 9.11, a RETRIEVE on an OCF Dimming Resource maps to a general Read command on a Zigbee Level Control Cluster. The value of Zigbee Attribute in Zigbee Level Control Cluster is retrieved via the general Read command and

mapped with the value of OCF Property in OCF Dimming Resource. In clause 9.10, an UPDATE on a "dimmingSetting" maps to a command invocation on "movetolevel(level,transitiontime=0)" of Zigbee Level Control Cluster.

8.2.4 Color control

The APIs with "zcl.colorcontrol" define the mapping between instances of OCF Colour Resources and the Zigbee Color Control Cluster. The OCF Colour Resources are OCF Hue and Saturation Resource, OCF Colour Space Coordinate Resource, OCF Colour Temperature Resource.

The APIs with "zcl.colorcontrol_hs" define the mapping between an instance of OCF Hue and Saturation Resources and the Zigbee Color Control Cluster. In clause 9.7, a RETRIEVE on an are OCF Hue and Saturation Resource maps to a general Read command on a Zigbee Color Control Cluster. The values of Zigbee Attributes in Zigbee Color Control Cluster are retrieved via the general Read command and mapped with those of OCF Properties in OCF Hue and Saturation Resource. In clause 9.6, an UPDATE on OCF Colour Hue and Saturation Resource maps to a command invocation on "movetohueandsaturation(hue,saturation,transitiontime=0)" of Zigbee Color Control Cluster.

The APIs with "zcl.colorcontrol_csc" define the mapping between an instance of OCF Colour Space Coordinate Resource and the Zigbee Color Control Cluster. In clause 9.3, a RETRIEVE on an OCF Colour Space Coordinate Resource maps to a general Read command on a Zigbee Color Control Cluster. The values of Zigbee Attributes in Zigbee Color Control Cluster are retrieved via the general Read command and mapped with those of OCF Properties in OCF Colour Space Coordinate Resource. In clause 9.2, an UPDATE on OCF Colour Space Coordinate Resource maps to a command invocation on "movetocolor(colorx,colory,transitiontime=0)" of Zigbee Color Control Cluster.

The APIs with "zcl.colorcontrol_ct" define the mapping between an instance of OCF Colour Temperature Resource and the Zigbee Color Control Cluster. In clause 9.4, a RETRIEVE on an OCF Colour Temperature Resource maps to a general Read command on a Zigbee Color Control Cluster. The values of Zigbee Attributes in Zigbee Color Control Cluster are retrieved via the general Read command and mapped with those of OCF Properties in OCF Colour Temperature Resource. In clause 9.5, an UPDATE on OCF Colour Temperature Resource maps to a command invocation on "movetocolortemperature(colortemperature,transitiontime=0)" of Zigbee Color Control Cluster.

8.2.5 Thermostat

The APIs with "zcl.thermostat" define the mapping between 3 instances of OCF Temperature Resources and the Zigbee Thermostat Cluster. The 3 instances of OCF Temperature Resources are for sensor, heater, and cooler respectively.

The API with "zcl.thermostat_currenttemperature" defines the mapping between an instance of OCF Temperature Resource and the Zigbee Thermostat Cluster for sensor. In clause 9.17, a RETRIEVE on an OCF Temperature Resource maps to a general Read command on a Zigbee Thermostat Cluster. The value of Zigbee Attribute in Zigbee Thermostat Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Temperature Resource. The value represents the current temperature.

The API with "zcl.thermostat_heat" defines the mapping between an instance of OCF Temperature Resource and the Zigbee Thermostat Cluster for heater. In clause 9.18, an UPDATE on "temperature" of OCF Temperature Resource maps to "setpointraiselower(mode=heat mode, amount)" on a Zigbee Thermostat Cluster.

The API with "zcl.thermostat_cool" defines the mapping between an instance of OCF Temperature Resource and the Zigbee Thermostat Cluster for cooler. In clause 9.16, an UPDATE on

"temperature" of OCF Temperature Resource maps to "setpointraiselower(mode=cool mode, amount)" on a Zigbee Thermostat Cluster.

8.2.6 Window covering

The APIs with "zcl.windowcovering" define the mapping between 5 instances of OCF Resources and the Zigbee Window Covering Cluster. The 5 instances of OCF Resources are the instance of OCF Window Covering Resource and the 4 instances of OCF Open Level Resources. The 4 instances of OCF Open Level Resources are for lift level with percentage scale, lift level with centimetre scale, tilt level with percentage scale, tilt level with centimetre scale.

The API with "zcl.windowcovering_conf" defines the mapping between an instance of OCF Window Covering Resource and the Zigbee window Covering Cluster. In clause 9.20, a RETRIEVE on an OCF Window Covering Resource maps to a general Read command on a Zigbee Window Covering Cluster. The values of Zigbee Attributes in Zigbee Window Covering Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Window Covering Resource. In clause 9.19, an UPDATE on OCF Window Covering Resource maps to a general Write command on a Zigbee Window Covering Cluster.

The API with "zcl.windowcovering_liftpercentage" defines the mapping between an instance of OCF Open Level Resource and the Zigbee window Covering Cluster for lift with percentage scale. In clause 9.22, a RETRIEVE on an OCF Open Level Resource maps to a general Read command on "CurrentPositionLiftPercentage" of Zigbee Window Covering Cluster. The value of Zigbee Attribute in Zigbee Window Covering Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Open Level Resource. In clause 9.21, an UPDATE on OCF Open Level Resource maps to "gotoliftpercentage(percentageliftvalue)" on a Zigbee Window Covering Cluster.

The API with "zcl.windowcovering_liftposition" defines the mapping between an instance of OCF Open Level Resource and the Zigbee window Covering Cluster for lift with centimetre scale. In clause 9.24, a RETRIEVE on an OCF Open Level Resource maps to a general Read command on "CurrentPosition-Lift" of Zigbee Window Covering Cluster. The value of Zigbee Attribute in Zigbee Window Covering Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Open Level Resource. In clause 9.23, an UPDATE on OCF Open Level Resource maps to "gotoliftvalue(liftvalue)" on a Zigbee Window Covering Cluster.

The API with "zcl.windowcovering_tiltpercentage" defines the mapping between an instance of OCF Open Level Resource and the Zigbee window Covering Cluster for tilt with percentage scale. In clause 9.26, a RETRIEVE on an OCF Open Level Resource maps to a general Read command on "CurrentPositionTiltPercentage" of Zigbee Window Covering Cluster. The value of Zigbee Attribute in Zigbee Window Covering Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Open Level Resource. In clause 9.25, an UPDATE on OCF Open Level Resource maps to "gototiltpercentage(percentagetiltvalue)" on a Zigbee Window Covering Cluster.

The API with "zcl.windowcovering_tiltposition" defines the mapping between an instance of OCF Open Level Resource and the Zigbee window Covering Cluster for tilt with centimetre scale. In clause 9.28, a RETRIEVE on an OCF Open Level Resource maps to a general Read command on "CurrentPosition-Tilt" of Zigbee Window Covering Cluster. The value of Zigbee Attribute in Zigbee Window Covering Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Open Level Resource. In clause 9.27, an UPDATE on OCF Open Level Resource maps to "gototiltvalue(tiltvalue)" on a Zigbee Window Covering Cluster.

8.2.7 Temperature measurement

The API with "zcl.temperaturemeasurement" defines the mapping between an instance of an OCF Temperature Resource and the Zigbee Temperature Measurement Cluster for sensor. In clause 9.15, a RETRIEVE on an OCF Temperature Resource maps to a general Read command on a

Zigbee Temperature Measurement Cluster. The value of Zigbee Attribute in Zigbee Temperature Measurement Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Temperature Resource. The value represents the current temperature.

8.2.8 Occupancy sensing

The API with "zcl.occupancysensing" defines the mapping between an instance of an OCF Presence Sensor Resource and the Zigbee Occupancy Sensing Cluster. In clause 9.12, a RETRIEVE on an OCF Presence Sensor Resource maps to a general Read command on a Zigbee Occupancy Sensing Cluster. The value of Zigbee Attribute in Zigbee Occupancy Sensing Cluster is retrieved via the general Read command and mapped with the value of OCF Property in OCF Presence Sensor.

8.2.9 IAS zone

The API with "zcl.iaszone" defines the mapping between an instance of an OCF IAS Zone Resource and the Zigbee IAS Zone Cluster. In clause 9.9, a RETRIEVE on an IAS Zone Resource maps to a general Read command on a Zigbee IAS Zone Cluster. The values of Zigbee Attributes in Zigbee IAS Zone Cluster are retrieved via the general Read command and mapped with those of OCF Properties in OCF IAS Zone Resource. In clause 9.8, an UPDATE on OCF IAS Zone Resource maps to a general Write command on a Zigbee IAS Zone Cluster.

9 Detailed mapping APIs

9.1 Introduction

This clause provides an API and mapping description that aligns with the Derived Modelling syntax described in Derived Models for Interoperability between IoT Ecosystems for all Module Classes and Resources that are within scope.

The derived model definitions presented in clause 9 are formatted for readability, and so may appear to have extra line breaks.

9.2 Color control cluster - color space - control

9.2.1 Derived model

The derived model: "zcl.colorcontrol_csc.control.movetocolor".

9.2.2 Property definition

Table 9 provides the detailed per Property mapping for "zcl.colorcontrol_csc.control.movetocolor".

Table 9 – The Property mapping for "zcl.colorcontrol_csc.control.movetocolor".

Zigbee Property name	OCF Resource	To OCF	From OCF
colory	oic.r.colour.csc	N/A	colory= ocf.csc[1]*65536 & transitiontime=0zcl.command.colorcontrol::movetocolor(colorx,colory,transitiontime).
colorx	oic.r.colour.csc	N/A	colorx =ocf.csc[0]*65536 & transitiontime=0zcl.command.colorcontrol::movetocolor(colorx,colory,transitiontime).

Table 10 provides the details of the Properties that are part of "zcl.colorcontrol_csc.control.movetocolor".

Table 10 – The Properties of "zcl.colorcontrol_csc.control.movetocolor".

Zigbee Property name	Type	Required	Description
colory	number	no	Move to certain value(s) of color coordinates as fast as possible with transitiontime=0. transitiontime is set by Zigbee 3.0 translator.
colorx	number	no	Move to certain value(s) of color coordinates as fast as possible with transitiontime=0. transitiontime is set by Zigbee 3.0 translator.

9.2.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.colorcontrol_csc.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Color Control Cluster - Color Space - Control",
  "definitions": {
    "zcl.colorcontrol_csc.control.movetocolor": {
      "properties": {
        "colorx": {
          "type": "number",
          "description": "Move to certain value(s) of color coordinates as fast as possible with
transitiontime=0. transitiontime is set by Zigbee 3.0 translator.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.csc",
            "x-from-ocf": [
              "colorx =ocf.csc[0]*65536 & transitiontime=0",
              "zcl.command.colorcontrol::movetocolor(colorx,colory,transitiontime)."
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        },
        "colory": {
          "type": "number",
          "description": "Move to certain value(s) of color coordinates as fast as possible with
transitiontime=0. transitiontime is set by Zigbee 3.0 translator.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.csc",
            "x-from-ocf": [
              "colory= ocf.csc[1]*65536 & transitiontime=0",
              "zcl.command.colorcontrol::movetocolor(colorx,colory,transitiontime)."
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.colorcontrol_csc.control.movetocolor"}
  ]
}
```


9.3 Color control cluster - color space - information

9.3.1 Derived model

The derived model: "zcl.colorcontrol_csc.info".

9.3.2 Property definition

Table 11 provides the detailed per Property mapping for "zcl.colorcontrol_csc.info".

Table 11 – The Property mapping for "zcl.colorcontrol_csc.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
currentX	oic.r.colour.csc	ocf.csc[0] = currentX/65536	N/A
currentY	oic.r.colour.csc	ocf.csc[1] = currentY/65536	N/A

Table 12 provides the details of the Properties that are part of "zcl.colorcontrol_csc.info".

Table 12 – The Properties of "zcl.colorcontrol_csc.info".

Zigbee Property name	Type	Required	Description
currentX	integer	no	current value of the normalized chromaticity value x, as defined in the CIE xy Color Space
currentY	integer	no	current value of the normalized chromaticity value y, as defined in the CIE xy Color Space

9.3.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.colorcontrol_csc.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Color Control Cluster - Color Space - Information",
  "definitions": {
    "zcl.colorcontrol_csc.info": {
      "type": "object",
      "properties": {
        "currentX": {
          "type": "integer",
          "description": "current value of the normalized chromaticity value x, as defined in the
CIE xy Color Space",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.csc",
            "x-to-ocf": [
              "ocf.csc[0] = currentX/65536"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "currentY": {
          "type": "integer",
          "description": "current value of the normalized chromaticity value y, as defined in the
CIE xy Color Space",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.csc",
            "x-to-ocf": [
              "ocf.csc[1] = currentY/65536"
            ],
```

```
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.colorcontrol_csc.info"}
  ],
  "required": ["currentx","currenty"]
}
```

9.4 Color control cluster - color temperature - information

9.4.1 Derived model

The derived model: "zcl.colorcontrol_ct.control.movetocolortemperature".

9.4.2 Property definition

Table 13 provides the detailed per Property mapping for "zcl.colorcontrol_ct.control.movetocolortemperature".

Table 13 – The Property mapping for "zcl.colorcontrol_ct.control.movetocolortemperature".

Zigbee Property name	OCF Resource	To OCF	From OCF
colortemperature	oic.r.colour.colourtemperature	N/A	colourtemperature=ocf.ct & transitiontime=0zcl.command.colorcontrol::movetocolortemperature(colortemperature,transitiontime)

Table 14 provides the details of the Properties that are part of "zcl.colorcontrol_ct.control.movetocolortemperature".

Table 14 – The Properties of "zcl.colorcontrol_ct.control.movetocolortemperature".

Zigbee Property name	Type	Required	Description
colortemperature	integer	no	Move to certain value of colortemperature as fast as possible with transitiontime=0. transitiontime is set by Zigbee 3.0 translator.

9.4.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.colorcontrol_ct.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Color Control Cluster - Color Temperature - Information",
  "definitions": {
    "zcl.colorcontrol_ct.control.movetocolortemperature": {
      "properties": {
        "colortemperature": {
          "type": "integer",
          "description": "Move to certain value of colortemperature as fast as possible with
transitiontime=0. transitiontime is set by Zigbee 3.0 translator.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.colourtemperature",
            "x-from-ocf": [
```



```
              "colourtemperature=ocf.ct & transitiontime=0",
              "zcl.command.colorcontrol::movetocolortemperature(colortemperature,transitiontime)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.colorcontrol_ct.control.movetocolortemperature"}
  ]
}
```

9.5 Color control cluster - color temperature - information

9.5.1 Derived model

The derived model: "zcl.colorcontrol_ct.info".

9.5.2 Property definition

Table 15 provides the detailed per Property mapping for "zcl.colorcontrol_ct.info".

Table 15 – The Property mapping for "zcl.colorcontrol_ct.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
colorphysicalmax	oic.r.colour.colourtemperature	ocf.range[1] = colorphysicalmax	N/A
colortemphysicalmin	oic.r.colour.colourtemperature	ocf.range[0] = colortemphysicalmin	N/A
colortemperaturemired	oic.r.colour.colourtemperature	ocf.ct = colortemperaturemired	N/A

Table 16 provides the details of the Properties that are part of "zcl.colorcontrol_ct.info".

Table 16 – The Properties of "zcl.colorcontrol_ct.info".

Zigbee Property name	Type	Required	Description
Colorphysicalmax	integer	no	maximum mired value supported by the hardware
Colortemphysicalmin	integer	no	minimum mired value supported by the hardware
Colortemperaturemired	integer	yes	Scaled inverse of the current value of the color temperature

9.5.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.colorcontrol_ct.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Color Control Cluster - Color Temperature - Information",
  "definitions": {
    "zcl.colorcontrol_ct.info": {
      "type": "object",
      "properties": {
        "colortemperaturemired": {
          "type": "integer",
          "description": "Scaled inverse of the current value of the color temperature",
```

```
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.colourtemperature",
            "x-to-ocf": [
              "ocf.ct = colortemperaturemired"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "colortemphysicalmin": {
          "type": "integer",
          "description": "minimum mired value supported by the hardware",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.colourtemperature",
            "x-to-ocf": [
              "ocf.range[0] = colortemphysicalmin"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "colorphysicalmax": {
          "type": "integer",
          "description": "maximmum mired value supported by the hardware",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.colourtemperature",
            "x-to-ocf": [
              "ocf.range[1] = colorphysicalmax"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.colorcontrol_ct.info"}
  ],
  "required": ["colortemperaturemired","colortempphysicalmin","colortempphysicalmax" ]
}
```

9.6 Color control cluster - hue and saturation - control

9.6.1 Derived model

The derived model: "zcl.colorcontrol_hs.control.movetohueandsaturation".

9.6.2 Property definition

Table 17 provides the detailed per Property mapping for "zcl.colorcontrol_hs.control.movetohueandsaturation".

Table 17 – The Property mapping for "zcl.colorcontrol_hs.control.movetohueandsaturation".

Zigbee Property name	OCF Resource	To OCF	From OCF
saturation	oic.r.colour.hs	N/A	saturation=ocf.saturation & transitiontime=0zcl.command.colorcontrol::movetohueandsaturation(hue,saturation, transitiontime)

hue	oic.r.colour.hs	N/A	hue=ocf.hue/360 * 254 & transitiontime=0zcl.command.colorcontrol::movetohueandsaturation(hue,saturation,transitiontime)

Table 18 provides the details of the Properties that are part of "zcl.colorcontrol_hs.control.movetohueandsaturation".

Table 18 – The Properties of "zcl.colorcontrol_hs.control.movetohueandsaturation".

Zigbee Property name	Type	Required	Description
saturation	integer	no	Move to certain value(s) of hue or saturation or both as fast as possible with transitiontime=0. transitiontime is set by Zigbee 3.0 translator.
hue	integer	no	Move to certain value(s) of hue or saturation or both as fast as possible with transitiontime=0. transitiontime is set by Zigbee 3.0 translator.

9.6.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.colorcontrol_hs.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Color Control Cluster - Hue and Saturation - Control",
  "definitions": {
    "zcl.colorcontrol_hs.control.movetohueandsaturation": {
      "properties": {
        "hue": {
          "type": "integer",
          "description": "Move to certain value(s) of hue or saturation or both as fast as possible
with transitiontime=0. transitiontime is set by Zigbee 3.0 translator.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.hs",
            "x-from-ocf": [
              "hue=ocf.hue/360 * 254 & transitiontime=0",
              "zcl.command.colorcontrol::movetohueandsaturation(hue,saturation,transitiontime)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        },
        "saturation": {
          "type": "integer",
          "description": "Move to certain value(s) of hue or saturation or both as fast as possible
with transitiontime=0. transitiontime is set by Zigbee 3.0 translator.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.hs",
            "x-from-ocf": [
              "saturation=ocf.saturation & transitiontime=0",
              "zcl.command.colorcontrol::movetohueandsaturation(hue,saturation,transitiontime)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
```

```
    {"$ref": "#/definitions/zcl.colorcontrol_hs.control.movetohueandsaturation"}
  ]
}
```

9.7 Color control cluster - hue and saturation - information

9.7.1 Derived model

The derived model: "zcl.colorcontrol_hs.info".

9.7.2 Property definition

Table 19 provides the detailed per Property mapping for "zcl.colorcontrol_hs.info".

Table 19 – The Property mapping for "zcl.colorcontrol_hs.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
currentsaturation	oic.r.colour.hs	ocf.saturation = currentsaturation & maximumsaturation=254	N/A
currenthue	oic.r.colour.hs	ocf.hue = currenthue/254 * 360	N/A

Table 20 provides the details of the Properties that are part of "zcl.colorcontrol_hs.info".

Table 20 – The Properties of "zcl.colorcontrol_hs.info".

Zigbee Property name	Type	Required	Description
currentsaturation	integer	yes	current saturation value of the light
currenthue	integer	yes	current hue value of the light

9.7.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.colorcontrol_hs.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Color Control Cluster - Hue and Saturation - Information",
  "definitions": {
    "zcl.colorcontrol_hs.info": {
      "type": "object",
      "properties": {
        "currenthue": {
          "type": "integer",
          "description": "current hue value of the light",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.hs",
            "x-to-ocf": [
              "ocf.hue = currenthue/254 * 360"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "currentsaturation": {
          "type": "integer",
          "description": "current saturation value of the light",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.colour.hs",
            "x-to-ocf": [
              "ocf.saturation = currentsaturation & maximumsaturation=254"
            ],
            "x-from-ocf": [
```

```
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.colorcontrol_hs.info"}
  ],
  "required": ["currenthue","currentsaturation"]
}
```

9.8 IAS zone cluster - control

9.8.1 Derived model

The derived model: "zcl.iaszone.control".

9.8.2 Property definition

Table 21 provides the detailed per Property mapping for "zcl.iaszone.control".

Table 21 – The Property mapping for "zcl.iaszone.control".

Zigbee Property name	OCF Resource	To OCF	From OCF
currentzonesensitivitylevel	oic.r.ias.zone	N/A	currentzonesensitivitylevel = ocf.currentzonesensitivitylevelzcl.command.general::write (currentzonesensitivitylevel)

Table 22 provides the details of the Properties that are part of "zcl.iaszone.control".

Table 22 – The Properties of "zcl.iaszone.control".

Zigbee Property name	Type	Required	Description
currentzonesensitivitylevel	integer	no	Set a sensitivity level of IAS Zone

9.8.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.iaszone.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "IAS Zone Cluster - Control",
  "definitions": {
    "zcl.iaszone.control": {
      "properties": {
        "currentzonesensitivitylevel": {
          "type": "integer",
          "description": "Set a sensitivity level of IAS Zone",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.ias.zone",
            "x-from-ocf": [
              "currentzonesensitivitylevel = ocf.currentzonesensitivitylevel",
              "zcl.command.general::write(currentzonesensitivitylevel)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
```

```
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.iaszone.control"}
  ]
}
```

9.9 IAS zone cluster - information

9.9.1 Derived model

The derived model: "zcl.iaszone.info".

9.9.2 Property definition

Table 23 provides the detailed per Property mapping for "zcl.iaszone.info".

Table 23 – The Property mapping for "zcl.iaszone.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
zoneID	oic.r.iaszone	ocf.zoneid=zoneID	N/A
numberofzonesensitivitylevelsupported	oic.r.iaszone	ocf.numzonesensitivitylevel= numberofzonesensitivitylevelsupported	N/A
zonestate	oic.r.iaszone	if zonestate=0x00, ocf.zonestate=falseif zonestate=0x01, ocf.zonestate=true	N/A
IAS_CIE_address	oic.r.iaszone	ocf.iascieaddress= IAS_CIE_address	N/A
zonetype	oic.r.iaszone	if zonetype=0x0000, ocf.zonetype=Standard CIEif zonetype=0x000d, ocf.zonetype=Motion sensorif zonetype=0x0015, ocf.zonetype=Contact switchif zonetype=0x0028, ocf.zonetype=Fire sensorif zonetype=0x002a, ocf.zonetype=Water sensorif zonetype=0x002b, ocf.zonetype=Carbon Monoxide (CO) sensorif zonetype=0x002c, ocf.zonetype=Personal emergency deviceif zonetype=0x002d, ocf.zonetype=Vibration/Movement sensorif zonetype=0x010f, ocf.zonetype=Remote Controlif zonetype=0x0115, ocf.zonetype=Key fobif zonetype=0x021d, ocf.zonetype=Keypadif zonetype=0x0225, ocf.zonetype=Standard Warning Deviceif zonetype=0x0226, ocf.zonetype=Glass break sensorif zonetype=0x0229, ocf.zonetype=Security repeaterif zonetype=0xffff, ocf.zonetype=Invalid Zone Type	N/A
zonestatus	oic.r.iaszone	if zonetype=0x0000 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x0000 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['system']if zonetype=0x0000 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x0000 & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['']if zonetype=0x000d & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x000d & zonestatus=xxxxxxxxxxxx1, ocf.zonestatus.alarms=['intrusion']if zonetype=0x000d & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x000d & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['presence']if zonetype=0x000d & zonestatus=xxxxxxxxxxx11, ocf.zonestatus.alarms=['intrusion','presence']if zonetype=0x0015 & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x0015 &	N/A

		zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['1stportalopenclose']if zonetype=0x0015 & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x0015 & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['2ndportalopenclose']if zonetype=0x0015 & zonestatus=xxxxxxxxxxx11, ocf.zonestatus.alarms=['1stportalopenclose','2ndportalope nclose']if zonetype=0x0028 & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x0028 & zonestatus=xxxxxxxxxxxx1, ocf.zonestatus.alarms=['fire']if zonetype=0x0028 & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x0028 & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['']if zonetype=0x002a & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x002a & zonestatus=xxxxxxxxxxx1, ocf.zonestatus.alarms=['wateroverflow']if zonetype=0x002a & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x002a & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['']if zonetype=0x002b & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x002b & zonestatus=xxxxxxxxxxxx1, ocf.zonestatus.alarms=['CO']if zonetype=0x002b & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x002b & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['cooking']if zonetype=0x002b & zonestatus=xxxxxxxxxxx11, ocf.zonestatus.alarms=['CO','cooking']if zonetype=0x002c & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x002c & zonestatus=xxxxxxxxxxx1, ocf.zonestatus.alarms=['fall']if zonetype=0x002c & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x002c & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['emergencybutton']if zonetype=0x002c & zonestatus=xxxxxxxxxxx11, ocf.zonestatus.alarms=['fall','emergencybutton']if zonetype=0x002d & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x002d & zonestatus=xxxxxxxxxxxx1, ocf.zonestatus.alarms=['movement']if zonetype=0x002d & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x002d & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['vibration']if zonetype=0x002d & zonestatus=xxxxxxxxxxx11, ocf.zonestatus.alarms=['movement','vibration']if zonetype=0x010f & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x010f & zonestatus=xxxxxxxxxxxx1, ocf.zonestatus.alarms=['panic']if zonetype=0x010f & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x010f & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['emergency']if zonetype=0x010f & zonestatus=xxxxxxxxxxx11, ocf.zonestatus.alarms=['panic','emergency']if zonetype=0x0115 & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x0115 & zonestatus=xxxxxxxxxxxx1, ocf.zonestatus.alarms=['panic']if zonetype=0x0115 & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x0115 & zonestatus=xxxxxxxxxxx1x, ocf.zonestatus.alarms=['emergency']if zonetype=0x0115 & zonestatus=xxxxxxxxxxx11, ocf.zonestatus.alarms=['panic','emergency']if zonetype=0x021d & zonestatus=xxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x021d & zonestatus=xxxxxxxxxxxx1, ocf.zonestatus.alarms=['panic']if zonetype=0x021d & zonestatus=xxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if	

		zonetype=0x021d & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['emergency']if zonetype=0x021d & zonestatus=xxxxxxxxxxxx11, ocf.zonestatus.alarms=['panic','emergency']if zonetype=0x0225 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x0225 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['glassbreak']if zonetype=0x0225 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x0225 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']if zonetype=0x0226 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x0226 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['']if zonetype=0x0226 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x0226 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']if zonetype=0x0229 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0x0229 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['']if zonetype=0x0229 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0x0229 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']if zonetype=0xffff & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']if zonetype=0xffff & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['']if zonetype=0xffff & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']if zonetype=0xffff & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']if zonestatus=xxxxxxxxxx0xx, ocf.zonestatus.tamper=falseif zonestatus=xxxxxxxxxx1xx, ocf.zonestatus.tamper=trueif zonestatus=xxxxxxxxx0xxx, ocf.zonebattery.charge=100 & ocf.zonebattery.lowbattery=falseif zonestatus=xxxxxxxxx1xxx, ocf.zonebattery.charge=100 & ocf.zonebattery.lowbattery=trueif zonestatus=xxxxxxx00xxxx, ocf.zonestatus.zonestatusreports='none'if zonestatus=xxxxxxx01xxxx, ocf.zonestatus.zonestatusreports='statuschangeonly' if zonestatus=xxxxxxx10xxxx, ocf.zonestatus.zonestatusreports='alarmclearonly' if zonestatus=xxxxxxx11xxxx, ocf.zonestatus.zonestatusreports='statuschangeandalarmclear'if zonestatus=xxxxxx0xxxxxx, ocf.zonestatus.fault=falseif zonestatus=xxxxxx1xxxxxx, ocf.zonestatus.fault=trueif zonestatus=xxxxx0xxxxxx, ocf.zonepowersource.powerSources=['AC (Mains) Power'] & ocf.zonepowersource.sourcefault=falseif zonestatus=xxxxx1xxxxxx, ocf.zonepowersource.powerSources=['AC (Mains) Power'] & ocf.zonepowersource.sourcefault=trueif zonestatus=xxxxx0xxxxxxx, ocf.zonestatus.test=falseif zonestatus=xxxxx1xxxxxxx, ocf.zonestatus.test=trueif zonestatus=xxxx0xxxxxxxx, ocf.zonepowersource.powerSources=['Internal Battery'] & oic.r.ias.zone.zonebattery.defect=false & oic.r.ias.zone.zonebattery.charge=100.if zonestatus=xxxx1xxxxxxxx, oic.r.ias.zone.zonepowersource.powerSources=['Internal Battery'] & oic.r.ias.zone.zonebattery.defect=true & oic.r.ias.zone.zonebattery.charge=100.	
currentzonesensitivitylevel	oic.r.iaszone	ocf.currentzonesensitivitylevel = currentzonesensitivitylevel	N/A

Table 24 provides the details of the Properties that are part of "zcl.iaszone.info".

Table 24 – The Properties of "zcl.iaszone.info".

Zigbee Property name	Type	Required	Description
zoneID	integer	no	Unique id allocated by IAS CIE
numberofzonesensitivitylevelsupported	integer	no	Total number of sensitivity levels supported by the IAS Zone
zonestate	boolean	yes	Enrollment status of IAS Zone false=not enrolled, true=enrolled
IAS_CIE_address	string	no	Address of IAS Control and Indicating Equipment (CIE)
zonetype	string	no	Zonetype and Meaning of Alarm1 and Alarm2 zonestatus
zonestatus	array	no	x is a variable. zonestatus in Zigbee maps to zonestatus, zonebattery, and zonepowersource in OCF. Data type of zonestatus in Zigbee is 16 bitmap (xxxxxxxxxxxxxxxx) : bit 0 = Alarm1, bit 1 = Alarm2, bit 2 = Tamper, bit 3 = Battery, bit 4 = Supervision reports, bit 5 = Restore reports, bit 6 = Trouble, bit 7 = AC (mains), bit 8 = Test, bit 9 = Battery Defect. Alarm1 : 1 = opened or alarmed 0 = closed or not alarmed, Alarm2 : 1 = opened or alarmed 0 = closed or not alarmed, Tamper : 1 = Tampered 0 = Not tampered, Battery : 1 = Low battery 0 = Battery OK, Supervision reports : 1 = Reports 0 = Does not report, Restore reports : 1 = Reports restore 0 = Does not report restore, Trouble : 1 = Trouble/Failure 0 = OK, AC (mains) : 1 = AC/Mains fault 0 = AC/Mains OK, Test : 1 = Sensor is in test mode 0 = Sensor is in operation mode, Battery Defect : 1 = Sensor detects a defective battery 0 = Sensor battery is functioning.
currentzonesensitivitylevel	integer	no	Sensitivity level of IAS Zone

9.9.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.iaszone.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "IAS Zone Cluster - Information",
    "definitions": {
      "zcl.iaszone.info": {
        "type": "object",
        "properties": {
          "zonestate": {
            "type": "boolean",
            "description": "Enrollment status of IAS Zone false=not enrolled, true=enrolled",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.iaszone",
              "x-to-ocf": [
                  "if zonestate=0x00, ocf.zonestate=false",
                  "if zonestate=0x01, ocf.zonestate=true"
              ],
              "x-from-ocf": [
                  "N/A"
              ]
            }
          },
          "zonetype": {
            "type": "string",
            "description": "Zonetype and Meaning of Alarm1 and Alarm2 zonestatus",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.iaszone",
              "x-to-ocf": [
                  "if zonetype=0x0000, ocf.zonetype=Standard CIE",
                  "if zonetype=0x000d, ocf.zonetype=Motion sensor",
                  "if zonetype=0x0015, ocf.zonetype=Contact switch",
                  "if zonetype=0x0028, ocf.zonetype=Fire sensor",
                  "if zonetype=0x002a, ocf.zonetype=Water sensor",
                  "if zonetype=0x002b, ocf.zonetype=Carbon Monoxide (CO) sensor",
                  "if zonetype=0x002c, ocf.zonetype=Personal emergency device",
                  "if zonetype=0x002d, ocf.zonetype=Vibration/Movement sensor",
                  "if zonetype=0x010f, ocf.zonetype=Remote Control",
                  "if zonetype=0x0115, ocf.zonetype=Key fob",
                  "if zonetype=0x021d, ocf.zonetype=Keypad",
                  "if zonetype=0x0225, ocf.zonetype=Standard Warning Device",
                  "if zonetype=0x0226, ocf.zonetype=Glass break sensor",
                  "if zonetype=0x0229, ocf.zonetype=Security repeater",
                  "if zonetype=0xffff, ocf.zonetype=Invalid Zone Type"
              ],
              "x-from-ocf": [
                  "N/A"
              ]
            }
          },
          "zonestatus": {
            "type": "array",
            "items": {
              "type": "integer"
            },
            "description": "x is a variable. zonestatus in Zigbee maps to zonestatus, zonebattery,
and zonepowersource in OCF. Data type of zonestatus in Zigbee is 16 bitmap (xxxxxxxxxxxxxxxx) : bit
0 = Alarm1, bit 1 = Alarm2, bit 2 = Tamper, bit 3 = Battery, bit 4 = Supervision reports, bit 5 =
Restore reports, bit 6 = Trouble, bit 7 = AC (mains), bit 8 = Test, bit 9 = Battery Defect.
Alarm1 : 1 = opened or alarmed 0 = closed or not alarmed, Alarm2 : 1 = opened or alarmed 0 = closed
or not alarmed, Tamper : 1 = Tampered 0 = Not tampered, Battery : 1 = Low battery 0 = Battery OK,
Supervision reports : 1 = Reports 0 = Does not report, Restore reports : 1 = Reports restore 0 =
Does not report restore, Trouble : 1 = Trouble/Failure 0 = OK, AC (mains) : 1 = AC/Mains fault 0 =
AC/Mains OK, Test : 1 = Sensor is in test mode 0 = Sensor is in operation mode, Battery Defect : 1
= Sensor detects a defective battery 0 = Sensor battery is functioning.",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.iaszone",
              "x-to-ocf": [
```



```
                "if zonetype=0x0000 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0000 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['system']",
                "if zonetype=0x0000 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0000 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']",

                "if zonetype=0x000d & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x000d & zonestatus=xxxxxxxxxxxxx1,
ocf.zonestatus.alarms=['intrusion']",
                "if zonetype=0x000d & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x000d & zonestatus=xxxxxxxxxxxx1x,
ocf.zonestatus.alarms=['presence']",
                "if zonetype=0x000d & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['intrusion','presence']",

                "if zonetype=0x0015 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0015 & zonestatus=xxxxxxxxxxxxx1,
ocf.zonestatus.alarms=['1stportalopenclose']",
                "if zonetype=0x0015 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0015 & zonestatus=xxxxxxxxxxxx1x,
ocf.zonestatus.alarms=['2ndportalopenclose']",
                "if zonetype=0x0015 & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['1stportalopenclose','2ndportalopenclose']",

                "if zonetype=0x0028 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0028 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['fire']",
                "if zonetype=0x0028 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0028 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']",

                "if zonetype=0x002a & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002a & zonestatus=xxxxxxxxxxxxx1,
ocf.zonestatus.alarms=['wateroverflow']",
                "if zonetype=0x002a & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002a & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']",

                "if zonetype=0x002b & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002b & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['CO']",
                "if zonetype=0x002b & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002b & zonestatus=xxxxxxxxxxxx1x,
ocf.zonestatus.alarms=['cooking']",
                "if zonetype=0x002b & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['CO','cooking']",

                "if zonetype=0x002c & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002c & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['fall']",
                "if zonetype=0x002c & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002c & zonestatus=xxxxxxxxxxxx1x,
ocf.zonestatus.alarms=['emergencybutton']",
                "if zonetype=0x002c & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['fall','emergencybutton']",

                "if zonetype=0x002d & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002d & zonestatus=xxxxxxxxxxxxx1,
ocf.zonestatus.alarms=['movement']",
                "if zonetype=0x002d & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x002d & zonestatus=xxxxxxxxxxxx1x,
ocf.zonestatus.alarms=['vibration']",
                "if zonetype=0x002d & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['movement','vibration']",

                "if zonetype=0x010f & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x010f & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['panic']",
                "if zonetype=0x010f & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x010f & zonestatus=xxxxxxxxxxxx1x,
ocf.zonestatus.alarms=['emergency']",
                "if zonetype=0x010f & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['panic','emergency']",

                "if zonetype=0x0115 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0115 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['panic']",
                "if zonetype=0x0115 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0115 & zonestatus=xxxxxxxxxxxx1x,
```



```
ocf.zonestatus.alarms=['emergency']",
                "if zonetype=0x0115 & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['panic','emergency']",

                "if zonetype=0x021d & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x021d & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['panic']",
                "if zonetype=0x021d & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x021d & zonestatus=xxxxxxxxxxxx1x,
ocf.zonestatus.alarms=['emergency']",
                "if zonetype=0x021d & zonestatus=xxxxxxxxxxxx11,
ocf.zonestatus.alarms=['panic','emergency']",

                "if zonetype=0x0225 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0225 & zonestatus=xxxxxxxxxxxxx1,
ocf.zonestatus.alarms=['glassbreak']",
                "if zonetype=0x0225 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0225 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']",

                "if zonetype=0x0226 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0226 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0226 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0226 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']",

                "if zonetype=0x0229 & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0229 & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0229 & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0x0229 & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']",

                "if zonetype=0xffff & zonestatus=xxxxxxxxxxxxx0, ocf.zonestatus.alarms=['']",
                "if zonetype=0xffff & zonestatus=xxxxxxxxxxxxx1, ocf.zonestatus.alarms=['']",
                "if zonetype=0xffff & zonestatus=xxxxxxxxxxxx0x, ocf.zonestatus.alarms=['']",
                "if zonetype=0xffff & zonestatus=xxxxxxxxxxxx1x, ocf.zonestatus.alarms=['']",

                "if zonestatus=xxxxxxxxxx0xx, ocf.zonestatus.tamper=false",
                "if zonestatus=xxxxxxxxxx1xx, ocf.zonestatus.tamper=true",

                "if zonestatus=xxxxxxxxx0xxx, ocf.zonebattery.charge=100 &
ocf.zonebattery.lowbattery=false",
                "if zonestatus=xxxxxxxxx1xxx, ocf.zonebattery.charge=100 &
ocf.zonebattery.lowbattery=true",

                "if zonestatus=xxxxxxxx00xxxx, ocf.zonestatus.zonestatusreports='none'",
                "if zonestatus=xxxxxxxx01xxxx, ocf.zonestatus.zonestatusreports='statuschangeonly'
",
                "if zonestatus=xxxxxxxx10xxxx, ocf.zonestatus.zonestatusreports='alarmclearonly' ",
                "if zonestatus=xxxxxxxx11xxxx,
ocf.zonestatus.zonestatusreports='statuschangeandalarmclear'",

                "if zonestatus=xxxxxxx0xxxxxx, ocf.zonestatus.fault=false",
                "if zonestatus=xxxxxxx1xxxxxx, ocf.zonestatus.fault=true",

                "if zonestatus=xxxxxx0xxxxxxx, ocf.zonepowersource.powerSources=['AC (Mains)
Power'] & ocf.zonepowersource.sourcefault=false",
                "if zonestatus=xxxxxx1xxxxxxx, ocf.zonepowersource.powerSources=['AC (Mains)
Power'] & ocf.zonepowersource.sourcefault=true",

                "if zonestatus=xxxxx0xxxxxxxx, ocf.zonestatus.test=false",
                "if zonestatus=xxxxx1xxxxxxxx, ocf.zonestatus.test=true",

                "if zonestatus=xxxx0xxxxxxxxx, ocf.zonepowersource.powerSources=['Internal
Battery'] & oic.r.ias.zone.zonebattery.defect=false & oic.r.ias.zone.zonebattery.charge=100.",
                "if zonestatus=xxxx1xxxxxxxxx,
oic.r.ias.zone.zonepowersource.powerSources=['Internal Battery'] &
oic.r.ias.zone.zonebattery.defect=true & oic.r.ias.zone.zonebattery.charge=100."
              ],
              "x-from-ocf": [
                 "N/A"
              ]
            }
          },
          "IAS_CIE_address": {
```



```
            "type": "string",
            "description": "Address of IAS Control and Indicating Equipment (CIE)",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.iaszone",
              "x-to-ocf": [
                "ocf.iascieaddress= IAS_CIE_address"
              ],
              "x-from-ocf": [
                  "N/A"
              ]
            }
          },
          "zoneID": {
            "type": "integer",
            "description": "Unique id allocated by IAS CIE",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.iaszone",
              "x-to-ocf": [
                "ocf.zoneid=zoneID"
              ],
              "x-from-ocf": [
                  "N/A"
              ]
            }
          },
          "numberofzonesensitivitylevelsupported": {
            "type": "integer",
            "description": "Total number of sensitivity levels supported by the IAS Zone",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.iaszone",
              "x-to-ocf": [
                "ocf.numzonesensitivitylevel= numberofzonesensitivitylevelsupported"
              ],
              "x-from-ocf": [
                  "N/A"
              ]
            }
          },
          "currentzonesensitivitylevel": {
            "type": "integer",
            "description": "Sensitivity level of IAS Zone",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.iaszone",
              "x-to-ocf": [
                "ocf.currentzonesensitivitylevel = currentzonesensitivitylevel"
              ],
              "x-from-ocf": [
                  "N/A"
              ]
            }
          }
        }
      }
    },
    "type": "object",
    "allOf": [
      {"$ref": "#/definitions/zcl.iaszone.info"}
    ],
    "required": [ "zonestate"]
  }
```

9.10 Level control cluster - control

9.10.1 Derived model

The derived model: "zcl.levelcontrol.control.moveto".

9.10.2 Property definition

Table 25 provides the detailed per Property mapping for "zcl.levelcontrol.control.moveto".

Table 25 – The Property mapping for "zcl.levelcontrol.control.moveto".

Zigbee Property name	OCF Resource	To OCF	From OCF
level	oic.r.light.dimming	N/A	level=ocf.dimmingSetting * 254 /100 , transitiontime=0zcl.command.levelcontrol::movetolevel(level,transitiontime)

Table 26 provides the details of the Properties that are part of "zcl.levelcontrol.control.moveto".

Table 26 – The Properties of "zcl.levelcontrol.control.moveto".

Zigbee Property name	Type	Required	Description
level	integer	no	Move to certain dimming value as fast as possible

9.10.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.levelcontrol.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Level Control Cluster - Control",
  "definitions": {
    "zcl.levelcontrol.control.moveto": {
      "properties": {
        "level": {
          "type": "integer",
          "description": "Move to certain dimming value as fast as possible ",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.light.dimming",
            "x-from-ocf": [
              "level=ocf.dimmingSetting * 254 /100 , transitiontime=0",
              "zcl.command.levelcontrol::movetolevel(level,transitiontime)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
   "allOf": [
     {"$ref": "#/definitions/zcl.levelcontrol.control.movetolevel"}
   ]
}
```

9.11 Level control cluster - information

9.11.1 Derived model

The derived model: "zcl.levelcontrol.info".

9.11.2 Property definition

Table 27 provides the detailed per Property mapping for "zcl.levelcontrol.info".

Table 27 – The Property mapping for "zcl.levelcontrol.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
currentlevel	oic.r.light.dimming	ocf.dimmingsetting = currentlevel/254 * 100	N/A

Table 28 provides the details of the Properties that are part of "zcl.levelcontrol.info".

Table 28 – The Properties of "zcl.levelcontrol.info".

Zigbee Property name	Type	Required	Description
currentlevel	integer	yes	current dimming value

9.11.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.levelcontrol.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Level Control Cluster - Information",
  "definitions": {
    "zcl.levelcontrol.info": {
      "type": "object",
      "properties": {
        "currentlevel": {
          "type": "integer",
          "description": "current dimming value",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.light.dimming",
            "x-to-ocf": [
              "ocf.dimmingsetting = currentlevel/254 * 100"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
    "allOf": [
      {"$ref": "#/definitions/zcl.levelcontrol.info"}
    ],
    "required": [ "currentlevel" ]
  }
```

9.12 Occupancy sensing cluster - information

9.12.1 Derived model

The derived model: "zcl.occupancysensing.info".

9.12.2 Property definition

Table 29 provides the detailed per Property mapping for "zcl.occupancysensing.info".

Table 29 – The Property mapping for "zcl.occupancysensing.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
occupancy	oic.r.sensor.presence	if occupancy =xxxxxxx0, then ocf.value = falseif occupancy =xxxxxxx1, then ocf.value = true	N/A

Table 30 provides the details of the Properties that are part of "zcl.occupancysensing.info".

Table 30 – The Properties of "zcl.occupancysensing.info".

Zigbee Property name	Type	Required	Description
occupancy	number	yes	x is a variable. Data type of occupancy in Zigbee is 8 bitmap (xxxxxxxx) while data type of value in OCF is boolean type i.e., true=occupied, false=unoccupied

9.12.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.occupancysensing.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Occupancy Sensing Cluster - Information",
  "definitions": {
    "zcl.occupancysensing.info": {
      "type": "object",
      "properties": {
        "occupancy": {
          "type": "number",
          "description": " x is a variable. Data type of occupancy in Zigbee is 8 bitmap (xxxxxxxx)
while data type of value in OCF is boolean type  i.e., true=occupied, false=unoccupied",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.sensor.presence",
            "x-to-ocf": [
              "if occupancy =xxxxxxx0, then ocf.value = false",
              "if occupancy =xxxxxxx1, then ocf.value = true"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.occupancysensing.info"}
  ],
  "required": [ "occupancy"]
}
```

9.13 On/Off cluster - control

9.13.1 Derived model

The derived model: "zcl.onoff.control.off".

The derived model: "zcl.onoff.control.on".

9.13.2 Property definition

Table 31 provides the detailed per Property mapping for "zcl.onoff.control.off".

Table 31 – The Property mapping for "zcl.onoff.control.off".

Zigbee Property name	OCF Resource	To OCF	From OCF
onoff	oic.r.switch.binary	N/A	if ocf.value = false, zcl.command.onoff::off().

Table 32 provides the details of the Properties that are part of "zcl.onoff.control.off".

Table 32 – The Properties of "zcl.onoff.control.off".

Zigbee Property name	Type	Required	Description
onoff	boolean	no	Turn off the device

Table 33 provides the detailed per Property mapping for "zcl.onoff.control.on".

Table 33 – The Property mapping for "zcl.onoff.control.on".

Zigbee Property name	OCF Resource	To OCF	From OCF
onoff	oic.r.switch.binary	N/A	if ocf.value = true, zcl.command.onoff::on().

Table 34 provides the details of the Properties that are part of "zcl.onoff.control.on".

Table 34 – The Properties of "zcl.onoff.control.on".

Zigbee Property name	Type	Required	Description
onoff	boolean	no	Turn on the device

9.13.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.onoff.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "On/Off Cluster - Control",
  "definitions": {
    "zcl.onoff.control.on": {
      "properties": {
        "onoff": {
          "type": "boolean",
          "description": "Turn on the device",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.switch.binary",
            "x-from-ocf": [
              "if ocf.value = true, zcl.command.onoff::on()."
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    },
    "zcl.onoff.control.off": {
      "properties": {
        "onoff": {
          "type": "boolean",
          "description": "Turn off the device",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.switch.binary",
            "x-from-ocf": [
              "if ocf.value = false, zcl.command.onoff::off()."
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.onoff.control.on"},
```

```
    {"$ref": "#/definitions/zcl.onoff.control.off"}
  ]
}
```

9.14 On/off cluster - information

9.14.1 Derived model

The derived model: "zcl.onoff".

9.14.2 Property definition

Table 35 provides the detailed per Property mapping for "zcl.onoff".

Table 35 – The Property mapping for "zcl.onoff".

Zigbee Property name	OCF Resource	To OCF	From OCF
onoff	oic.r.switch.binary	if onoff = false, then ocf.value = falseif onoff = true, then ocf.value = true	N/A

Table 36 provides the details of the Properties that are part of "zcl.onoff".

Table 36 – The Properties of "zcl.onoff".

Zigbee Property name	Type	Required	Description
onoff	boolean	yes	On/off status of the device

9.14.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.onoff.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "On/off Cluster - Information",
  "definitions": {
    "zcl.onoff": {
      "type": "object",
      "properties": {
        "onoff": {
          "type": "boolean",
          "description": "On/off status of the device",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.switch.binary",
            "x-to-ocf": [
              "if onoff = false, then ocf.value = false",
              "if onoff = true, then ocf.value = true"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.onoff.info"}
  ],
  "required": [ "onoff" ]
}
```

9.15 Temperature measurement cluster - information

9.15.1 Derived model

The derived model: "zcl.temperaturemeasurement.info".

9.15.2 Property definition

Table 37 provides the detailed per Property mapping for "zcl.temperaturemeasurement.info".

Table 37 – The Property mapping for "zcl.temperaturemeasurement.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
MeasuredValue	oic.r.temperature	ocf.temperature = MeasuredValue/100units = C	N/A
MinMeasuredValue	oic.r.temperature	ocf.range[0] = MinMeasuredValue/100	N/A
Tolerance	oic.r.temperature	ocf.precision = Tolerance/100	N/A
MaxMeasuredValue	oic.r.temperature	ocf.range[1] = MaxMeasuredValue/100	N/A

Table 38 provides the details of the Properties that are part of "zcl.temperaturemeasurement.info".

Table 38 – The Properties of "zcl.temperaturemeasurement.info".

Zigbee Property name	Type	Required	Description
MeasuredValue	number	yes	Measured value
MinMeasuredValue	number	yes	Minimum value of MeasuredValue
Tolerance	number	yes	Magnitude of the possible error
MaxMeasuredValue	number	yes	Maximum value of MeasuredValue

9.15.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.temperaturemeasurement.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Temperature Measurement Cluster - Information",
  "definitions": {
    "zcl.temperaturemeasurement.info": {
      "type": "object",
      "properties": {
        "MeasuredValue": {
          "type": "number",
          "description": "Measured value",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature",
            "x-to-ocf": [
              "ocf.temperature = MeasuredValue/100",
              "units = C"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "Tolerance": {
          "type": "number",
```

```
          "description": "Magnitude of the possible error",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature",
            "x-to-ocf": [
              "ocf.precision = Tolerance/100"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "MinMeasuredValue": {
          "type": "number",
          "description": "Minimum value of MeasuredValue",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature",
            "x-to-ocf": [
              "ocf.range[0] = MinMeasuredValue/100"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "MaxMeasuredValue": {
          "type": "number",
          "description": "Maximum value of MeasuredValue",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature",
            "x-to-ocf": [
              "ocf.range[1] = MaxMeasuredValue/100"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.temperaturemeasurement.info"}
  ],
  "required": [ "MeasuredValue","Tolerance","MinMeasuredValue","MaxMeasuredValue"]
}
```

9.16 Thermostat cluster - cool - control

9.16.1 Derived model

The derived model: "zcl.thermostat_cool.control.setpointraiselower".

9.16.2 Property definition

Table 39 provides the detailed per Property mapping for "zcl.thermostat_cool.control.setpointraiselower".

Table 39 – The Property mapping for "zcl.thermostat_cool.control.setpointraiselower".

Zigbee Property name	OCF Resource	To OCF	From OCF
amount	oic.r.temperature	N/A	if ocf.temperature is updated, then amount= ocf.temperature*100.zcl.command.thermostat::setpointraiselower(mode, amount)

Table 40 provides the details of the Properties that are part of "zcl.thermostat_cool.control.setpointraiselower".

Table 40 – The Properties of "zcl.thermostat_cool.control.setpointraiselower".

Zigbee Property name	Type	Required	Description
amount	number	no	Set the target temperature with cool mode. Mode=0x01 is set by Zigbee 3.0 translator

9.16.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.thermostat_cool.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Thermostat Cluster - Cool - Control",
  "definitions": {
    "zcl.thermostat_cool.control.setpointraiselower": {
      "type": "object",
      "properties": {
        "amount": {
          "type": "number",
          "description": "Set the target temperature with cool mode. Mode=0x01 is set by Zigbee 3.0
translator",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature",
            "x-from-ocf": [
              "if ocf.temperature is updated, then amount= ocf.temperature*100.",
              "zcl.command.thermostat::setpointraiselower(mode, amount)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
     {"$ref": "#/definitions/zcl.thermostat_cool.control.setpointraiselower"}
  ]
}
```

9.17 Thermostat cluster - current temperature - information

9.17.1 Derived model

The derived model: "zcl.thermostat_currenttemperature.info".

9.17.2 Property definition

Table 41 provides the detailed per Property mapping for "zcl.thermostat_currenttemperature.info".

Table 41 – The Property mapping for "zcl.thermostat_currenttemperature.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
localtemperature	oic.r.temperature	ocf.temperature=localtempearture/100units = C	N/A

Table 42 provides the details of the Properties that are part of "zcl.thermostat_currenttemperature.info".

Table 42 – The Properties of "zcl.thermostat_currenttemperature.info".

Zigbee Property name	Type	Required	Description
localtemperature	Number	no	current sensed temperature

9.17.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.thermostat_currenttemperature.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Thermostat Cluster - Current Temperature - Information ",
  "definitions": {
    "zcl.thermostat_currenttemperature.info": {
      "type": "object",
      "properties": {
        "localtemperature": {
          "type": "number",
          "description": "current sensed temperature",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature",
            "x-to-ocf": [
              "ocf.temperature=localtempearture/100",
              "units = C"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.thermostat_currenttemperature.info"}
  ],
  "required": [ "localtempearture"]
}
```

9.18 Thermostat cluster - heat - control

9.18.1 Derived model

The derived model: "zcl.thermostat_heat.control.setpointraiseLower".

9.18.2 Property definition

Table 43 provides the detailed per Property mapping for "zcl.thermostat_heat.control.setpointraiseLower".

Table 43 – The Property mapping for "zcl.thermostat_heat.control.setpointraiseLower".

Zigbee Property name	OCF Resource	To OCF	From OCF
amount	oic.r.temperature	N/A	if ocf.temperature is updated, then amount= ocf.temperature*100.zcl.command.thermostat::setpointraiseLower(mode, amount)

Table 44 provides the details of the Properties that are part of "zcl.thermostat_heat.control.setpointraiseLower".

Table 44 – The Properties of "zcl.thermostat_heat.control.setpointraiseLower".

Zigbee Property name	Type	Required	Description
Amount	number	no	Set the target temperature with heat mode. Mode=0x00 is set by Zigbee 3.0 translator

9.18.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.thermostat_heat.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Thermostat Cluster - Heat - Control",
  "definitions": {
    "zcl.thermostat_heat.control.setpointraiseLower": {
      "type": "object",
      "properties": {
        "amount": {
          "type": "number",
          "description": "Set the target temperature with heat mode. Mode=0x00 is set by Zigbee 3.0
translator",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.temperature",
            "x-from-ocf": [
              "if ocf.temperature is updated, then amount= ocf.temperature*100.",
              "zcl.command.thermostat::setpointraiseLower(mode, amount)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.thermostat_heat.control.setpointraiseLower"}
  ]
}
```

9.19 Window covering cluster - configuration - control

9.19.1 Derived model

The derived model: "zcl.windowcovering_conf.control".

9.19.2 Property definition

Table 45 provides the detailed per Property mapping for "zcl.windowcovering_conf.control".

Table 45 – The Property mapping for "zcl.windowcovering_conf.control".

Zigbee Property name	OCF Resource	To OCF	From OCF
Acceleration Time-Lift	oic.r.windowcovering	N/A	if ocf.liftaccelerationtime is updated, Acceleration Time-Lift=ocf.liftaccelerationtime.zcl.command.general::write(Acceleration Time-Lift)
Velocity-Lift	oic.r.windowcovering	N/A	if ocf.liftvelocity is updated, Velocity-Lift = ocf.liftvelocity.zcl.command.general::write(Velocity-Lift)

Deceleration Time-Lift	oic.r.windowcovering	N/A	if ocf.liftdecelerationtime is updated, Deceleration Time-Lift=ocf.liftdecelerationtime.zcl.command.general::write(Deceleration Time-Lift)
mode	oic.r.windowcovering	N/A	if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxxxxx0.if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxxxxx1.if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxxxx0x.if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxxxx1x.if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxxx0xx.if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxxx1xx.if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxx0xxx.if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxx1xxx.zcl.command.general::write(mode)

Table 46 provides the details of the Properties that are part of "zcl.windowcovering_conf.control".

Table 46 – The Properties of "zcl.windowcovering_conf.control".

Zigbee Property name	Type	Required	Description
Acceleration Time-Lift	integer	no	Set ramp up times to reaching the velocity setting (0.1sec).
Velocity-Lift	integer	no	Set velocity associtated with Lifting the Window Covering (cm/sec).
Deceleration Time-Lift	integer	no	Set ramp down times associated with stoping the velocity setting (0.1sec).
mode	integer	no	Set the mode. x is a variable. Data type of Mode in Zigbee is 8 bitmap (xxxxxxxx) while data type of mode in OCF is array with 4 Boolean type items(i.e., [Reversed Motor Direction, Calibration Mode, Maintenance Mode, LED]). Reversed Motor Direction : 0 = motor direction is normal, 1 = motor direction is reversed. Calibration Mode : 0 = run in normal mode, 1 = run in calibration mode. Maintenance Mode : 0 = motor is running normally, 1 = motor is running in maintenance mode. LED: 0 = LEDs are off, 1 = LEDs will display feedback.

9.19.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_conf.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Configuration - Control",
  "definitions": {
    "zcl.windowcovering_conf.control": {
      "properties": {
        "mode": {
          "type": "integer",
          "description": "Set the mode. x is a variable. Data type of Mode in Zigbee is 8 bitmap
(xxxxxxxx) while data type of mode in OCF is array with 4 Boolean type items(i.e., [Reversed Motor
```



```
Direction, Calibration Mode, Maintenance Mode, LED]). Reversed Motor Direction : 0 = motor
direction is normal, 1 = motor direction is reversed. Calibration Mode : 0 = run in normal mode, 1
= run in calibration mode. Maintenance Mode : 0 = motor is running normally, 1 = motor is running
in maintenance mode. LED: 0 = LEDs are off, 1 = LEDs will display feedback.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.windowcovering",
            "x-from-ocf": [
              "if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxxxxx0.",
              "if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxxxxx1.",
              "if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxxxx0x.",
              "if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxxxx1x.",
              "if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxxx0xx.",
              "if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxxx1xx.",
              "if ocf.mode is updated & ocf.mode = [false,x,x,x], Mode =xxxx0xxx.",
              "if ocf.mode is updated & ocf.mode = [true,x,x,x], Mode =xxxx1xxx.",
              "zcl.command.general::write(mode)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        },
        "Velocity-Lift": {
          "type": "integer",
          "description": "Set velocity associtated with Lifting the Window Covering (cm/sec).",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.windowcovering",
            "x-from-ocf": [
              "if ocf.liftvelocity is updated, Velocity-Lift = ocf.liftvelocity.",
              "zcl.command.general::write(Velocity-Lift)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        },
        "Acceleration Time-Lift": {
          "type": "integer",
          "description": "Set ramp up times to reaching the velocity setting (0.1sec).",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.windowcovering",
            "x-from-ocf": [
              "if ocf.liftaccelerationtime is updated, Acceleration Time-
Lift=ocf.liftaccelerationtime.",
              "zcl.command.general::write(Acceleration Time-Lift)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        },
        "Deceleration Time-Lift": {
          "type": "integer",
          "description": "Set ramp down times associated with stoping the velocity setting
(0.1sec).",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.windowcovering",
            "x-from-ocf": [
              "if ocf.liftdecelerationtime is updated, Deceleration Time-
Lift=ocf.liftdecelerationtime.",
              "zcl.command.general::write(Deceleration Time-Lift)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
```



```
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_conf.control"}
  ]
}
```

9.20 Window covering cluster - configuration - information

9.20.1 Derived model

The derived model: "zcl.windowcovering_conf.info".

9.20.2 Property definition

Table 47 provides the detailed per Property mapping for "zcl.windowcovering_conf.info".

Table 47 – The Property mapping for "zcl.windowcovering_conf.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
Velocity-Lift	oic.r.windowcovering	ocf.liftvelocity = Velocity-Lift	N/A
Windowcoveringtype	oic.r.windowcovering	if WindowCoveringType=0x00, ocf.windowcoveringtype= Rollershade.if WindowCoveringType=0x01, ocf.windowcoveringtype= RollerShade-2 Motor.if WindowCoveringType=0x02, ocf.windowcoveringtype= RollerShade-Exterior.if WindowCoveringType=0x03, ocf.windowcoveringtype= RollerShade-Exterior-2 Motor.if WindowCoveringType=0x04, ocf.windowcoveringtype= Drapery.if WindowCoveringType=0x05, ocf.windowcoveringtype= Awning.if WindowCoveringType=0x06, ocf.windowcoveringtype= Shutter.if WindowCoveringType=0x07, ocf.windowcoveringtype= Tilt Blind - Tilt Only.if WindowCoveringType=0x08, ocf.windowcoveringtype= Tilt Blind â€“ Lift and Tilt.if WindowCoveringType=0x09, ocf.windowcoveringtype= Projector Screen.	N/A
Config/Status	oic.r.windowcovering	if Config/Status =xxxxxxx0, ocf.configstatus.operational = falseif Config/Status =xxxxxxx1, ocf.configstatus.operational = trueif Config/Status =xxxxxx0x, ocf.configstatus.online = falseif Config/Status =xxxxxx1x, ocf.configstatus.online = trueif Config/Status =xxxxx0xx, ocf.configstatus.rotationdirection = 'normal'if Config/Status =xxxxx1xx, ocf.configstatus.rotationdirection = 'reversed'if Config/Status =xxxx0xxx, ocf.configstatus.controllift = 'openloop'if Config/Status =xxxx1xxx, ocf.configstatuscontrollift = 'closedloop'if Config/Status =xxx0xxxx, ocf.configstatus.controltilt =	N/A

		'openloop'if Config/Status =xxx1xxxx, ocf.configstatus.controltilt = 'closedloop'if Config/Status =xx0xxxxx, ocf.configstatus.closedloopliftcontrol = 'timer'if Config/Status =xx1xxxxx, ocf.configstatus.closedloopliftcontrol = 'encoder'if Config/Status =x0xxxxxx, ocf.configstatus.closedlooptiltcontrol = 'timer'if Config/Status =x1xxxxxx, ocf.configstatus.closedlooptiltcontrol = 'encoder'	
Deceleration Time-Lift	oic.r.windowcovering	ocf.liftdecelerationtime= Deceleration Time-Lift	N/A
Mode	oic.r.windowcovering	if Mode =xxxxxxx0, ocf.mode.motordirection = falseif Mode =xxxxxxx1, ocf.mode.motordirection = trueif Mode =xxxxxx0x, ocf.mode.calibration = falseif Mode =xxxxxx1x, ocf.mode.calibration = trueif Mode =xxxxx0xx, ocf.mode.maintenance = falseif Mode =xxxxx1xx, ocf.mode.maintenance = trueif Mode =xxxx0xxx, ocf.mode.ledfeedback = falseif Mode =xxxx1xxx, ocf.mode.ledfeedback = true	N/A
Acceleration Time-Lift	oic.r.windowcovering	ocf.liftaccelerationtime= Acceleration Time-Lift	N/A

Table 48 provides the details of the Properties that are part of "zcl.windowcovering_conf.info".

Table 48 – The Properties of "zcl.windowcovering_conf.info".

Zigbee Property name	Type	Required	Description
Velocity-Lift	integer	no	Velocity associated with Lifting the Window Covering (cm/sec).
Windowcoveringtype	string	yes	Type of Window Covering(i.e., [Rollershade,RollerShade-2 Motor, RollerShade-Exterior, RollerShade-Exterior-2 Motor, Drapery, Awning, Shutter, Tilt Blind - Tilt Only, Tilt Blind â€“ Lift and Tilt, Projector Screen])
Config/Status	integer	yes	x is a variable. Config/Status in Zigbee maps to configstatus in OCF. Data type of Config/Status in Zigbee is 8 bitmap (xxxxxxxx) : bit 0 = Operational, bit 1 = Online, bit 2 = Reversal, bit 3 = Control-Lift, bit 4 = Control-Tilt, bit 5 = Encoder-Lift, bit 6 = Encoder-Tilt. Operational: This status bit defines if

			the Window Covering is operational. 0 = Not Operational, 1 = Operational. Online: This status bit defines if the Window Covering is enabled for transmitting over the ZigBee network. 0 = Not Online, 1 = Online. Reversal: This status bit identifies if the direction of rotation for the Window Covering has been reversed in order for Open/Up commands to match the physical installation condition. 0 = Commands are normal, 1 = Open/Up Commands have been reversed. Control Lift: This status bit identifies if the window covering supports Open Loop or Closed Loop Lift Control. 0 = Lift control is Open Loop, 1 = Lift control is Closed. Control Tilt: This status bit identifies if the window covering supports Open Loop or Closed Loop Tilt Control. 0 = Tilt control is Open Loop, 1 = Tilt control is Closed. Encoder Lift: This status bit identifies if a Closed Loop Controlled Window Covering is employing an encoder for positioning the height of the window covering. 0 = Timer Controlled, 1 = Encoder Controlled. Encoder Tilt: This status bit identifies if a Closed Loop Controlled Window Covering is employing an encoder for tilting the window covering. 0 = Timer Controlled, 1 = Encoder Controlled.
Deceleration Time-Lift	integer	no	Ramp down times associated with stopping the velocity setting (0.1sec).
Mode	integer	yes	x is a variable. Mode in Zigbee maps to mode in OCF. Data type of Mode in Zigbee is 8 bitmap (xxxxxxxx) : bit 0 = Reversed Motor Direction, bit 1 = Calibration Mode, bit 2 = Maintenance Mode, bit 3 = LED. Reversed Motor Direction : 0 = motor direction is normal, 1 = motor direction is reversed. Calibration Mode : 0 = run in normal

			mode, 1 = run in calibration mode. Maintenance Mode : 0 = motor is running normally, 1 = motor is running in maintenance mode. LED: 0 = LEDs are off, 1 = LEDs will display feedback.
Acceleration Time-Lift	integer	no	Ramp up times to reaching the velocity setting (0.1sec).

9.20.3 Derived model definition

```
{
  "id": "http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_conf.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Configuration - Information",
  "definitions": {
    "zcl.windowcovering_conf.info": {
      "type": "object",
      "properties": {
        "Windowcoveringtype": {
          "type": "string",
          "description": "Type of Window Covering(i.e., [Rollershade,RollerShade-2 Motor,
RollerShade-Exterior, RollerShade-Exterior-2 Motor, Drapery, Awning, Shutter, Tilt Blind - Tilt
Only, Tilt Blind â€“ Lift and Tilt, Projector Screen])",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.windowcovering",
            "x-to-ocf": [
              "if WindowCoveringType=0x00,  ocf.windowcoveringtype= Rollershade.",
              "if WindowCoveringType=0x01,  ocf.windowcoveringtype= RollerShade-2 Motor.",
              "if WindowCoveringType=0x02,  ocf.windowcoveringtype= RollerShade-Exterior.",
              "if WindowCoveringType=0x03,  ocf.windowcoveringtype= RollerShade-Exterior-2 Motor.",
              "if WindowCoveringType=0x04,  ocf.windowcoveringtype= Drapery.",
              "if WindowCoveringType=0x05,  ocf.windowcoveringtype= Awning.",
              "if WindowCoveringType=0x06,  ocf.windowcoveringtype= Shutter.",
              "if WindowCoveringType=0x07,  ocf.windowcoveringtype= Tilt Blind - Tilt Only.",
              "if WindowCoveringType=0x08,  ocf.windowcoveringtype= Tilt Blind â€“ Lift and Tilt.",
              "if WindowCoveringType=0x09,  ocf.windowcoveringtype= Projector Screen."
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "Config/Status": {
          "type": "integer",
          "description": " x is a variable. Config/Status in Zigbee maps to configstatus in OCF.
Data type of Config/Status in Zigbee is 8 bitmap (xxxxxxxx) : bit 0 = Operational, bit 1 = Online,
bit 2 = Reversal, bit 3 = Control-Lift, bit 4 = Control-Tilt, bit 5 =  Encoder-Lift, bit 6 =
Encoder-Tilt. Operational: This status bit defines if the Window Covering is operational. 0 = Not
Operational, 1 = Operational. Online: This status bit defines if the Window Covering is enabled for
transmitting over the ZigBee network. 0 = Not Online, 1 = Online. Reversal: This status bit
identifies if the direction of rotation for the Window Covering has been reversed in order for
Open/Up commands to match the physical installation condition. 0 = Commands are normal, 1 = Open/Up
Commands have been reversed. Control Lift: This status bit identifies if the window covering
supports Open Loop or Closed Loop Lift Control. 0 = Lift control is Open Loop, 1 = Lift control is
Closed. Control Tilt: This status bit identifies if the window covering supports Open Loop or
Closed Loop Tilt Control. 0 = Tilt control is Open Loop, 1 = Tilt control is Closed. Encoder Lift:
This status bit identifies if a Closed Loop Controlled Window Covering is employing an encoder for
positioning the height of the window covering. 0 = Timer Controlled, 1 = Encoder Controlled.
Encoder Tilt: This status bit identifies if a Closed Loop Controlled Window Covering is employing
an encoder for tilting the window covering. 0 = Timer Controlled, 1 = Encoder Controlled.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.windowcovering",
            "x-to-ocf": [
              "if Config/Status =xxxxxxx0, ocf.configstatus.operational = false",
              "if Config/Status =xxxxxxx1, ocf.configstatus.operational = true",
```

```
            "if Config/Status =xxxxxx0x, ocf.configstatus.online = false",
            "if Config/Status =xxxxxx1x, ocf.configstatus.online = true",
            "if Config/Status =xxxxx0xx, ocf.configstatus.rotationdirection = 'normal'",
            "if Config/Status =xxxxx1xx, ocf.configstatus.rotationdirection = 'reversed'",
            "if Config/Status =xxxx0xxx, ocf.configstatus.controllift = 'openloop'",
            "if Config/Status =xxxx1xxx, ocf.configstatuscontrollift = 'closedloop'",
            "if Config/Status =xxx0xxxx, ocf.configstatus.controltilt = 'openloop'",
            "if Config/Status =xxx1xxxx, ocf.configstatus.controltilt = 'closedloop'",
            "if Config/Status =xx0xxxxx, ocf.configstatus.closedloopliftcontrol = 'timer'",
            "if Config/Status =xx1xxxxx, ocf.configstatus.closedloopliftcontrol = 'encoder'",
            "if Config/Status =x0xxxxx, ocf.configstatus.closedlooptiltcontrol = 'timer'",
            "if Config/Status =x1xxxxx, ocf.configstatus.closedlooptiltcontrol = 'encoder'"
          ],
          "x-from-ocf": [
            "N/A"
          ]
        }
      },
      "Mode": {
        "type": "integer",
        "description": "x is a variable. Mode in Zigbee maps to mode in OCF. Data type of Mode in
Zigbee is 8 bitmap (xxxxxxxx) : bit 0 = Reversed Motor Direction, bit 1 = Calibration Mode, bit 2 =
Maintenance Mode, bit 3 = LED. Reversed Motor Direction : 0 = motor direction is normal, 1 = motor
direction is reversed. Calibration Mode : 0 = run in normal mode, 1 = run in calibration mode.
Maintenance Mode : 0 = motor is running normally, 1 = motor is running in maintenance mode. LED: 0
= LEDs are off, 1 = LEDs will display feedback.",
        "x-ocf-conversion": {
          "x-ocf-alias": "oic.r.windowcovering",
          "x-to-ocf": [
            "if Mode =xxxxxxx0, ocf.mode.motordirection = false",
            "if Mode =xxxxxxx1, ocf.mode.motordirection = true",
            "if Mode =xxxxxx0x, ocf.mode.calibration = false",
            "if Mode =xxxxxx1x, ocf.mode.calibration = true",
            "if Mode =xxxxx0xx, ocf.mode.maintenance = false",
            "if Mode =xxxxx1xx, ocf.mode.maintenance = true",
            "if Mode =xxxx0xxx, ocf.mode.ledfeedback = false",
            "if Mode =xxxx1xxx, ocf.mode.ledfeedback = true"
          ],
          "x-from-ocf": [
            "N/A"
          ]
        }
      },
      "Velocity-Lift": {
        "type": "integer",
        "description": "Velocity associtated with Lifting the Window Covering (cm/sec).",
        "x-ocf-conversion": {
          "x-ocf-alias": "oic.r.windowcovering",
          "x-to-ocf": [
            "ocf.liftvelocity = Velocity-Lift"
          ],
          "x-from-ocf": [
            "N/A"
          ]
        }
      },
      "Acceleration Time-Lift": {
        "type": "integer",
        "description": "Ramp up times to reaching the velocity setting (0.1sec).",
        "x-ocf-conversion": {
          "x-ocf-alias": "oic.r.windowcovering",
          "x-to-ocf": [
            "ocf.liftaccelerationtime= Acceleration Time-Lift"
          ],
          "x-from-ocf": [
            "N/A"
          ]
        }
      },
      "Deceleration Time-Lift": {
        "type": "integer",
```



```
          "description": "Ramp down times associated with stoping the velocity setting (0.1sec).",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.windowcovering",
            "x-to-ocf": [
              "ocf.liftdecelerationtime= Deceleration Time-Lift"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_conf.info"}
  ],
  "required": [ "Windowcoveringtype","Config/Status","Mode"]
}
```

9.21 Window covering cluster - lift percentage - control

9.21.1 Derived model

The derived model: "zcl.windowcovering_liftpercentage.control.gotoliftpercentage".

9.21.2 Property definition

Table 49 provides the detailed per Property mapping for "zcl.windowcovering_liftpercentage.control.gotoliftpercentage".

Table 49 – The Property mapping for "zcl.windowcovering_liftpercentage.control.gotoliftpercentage".

Zigbee Property name	OCF Resource	To OCF	From OCF
percentageliftvalue	oic.r.openlevel	N/A	if ocf.openLevel is updated, percentage lift value = ocf.openLevel.zcl.command.windowcovering::gotoliftpercentage(percentageliftvalue)

Table 50 provides the details of the Properties that are part of "zcl.windowcovering_liftpercentage.control.gotoliftpercentage".

Table 50 – The Properties of "zcl.windowcovering_liftpercentage.control.gotoliftpercentage".

Zigbee Property name	Type	Required	Description
percentageliftvalue	integer	no	Adjust the window at the percentage lift value.

9.21.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_liftpercentage.control.json#"
,
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Lift Percentage - Control",
  "definitions": {
    "zcl.windowcovering_liftpercentage.control.gotoliftpercentage": {
      "properties": {
        "percentageliftvalue": {
          "type": "integer",
```

```
          "description": "Adjust the window at the percentage lift value.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-from-ocf": [
              "if ocf.openLevel is updated, percentage lift value = ocf.openLevel.",
              "zcl.command.windowcovering::gotoliftpercentage(percentageliftvalue)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_liftpercentage.control.gotoliftpercentage"}
  ]
}
```

9.22 Window covering cluster - lift percentage - information

9.22.1 Derived model

The derived model: "zcl.windowcovering_liftpercentage.info".

9.22.2 Property definition

Table 51 provides the detailed per Property mapping for "zcl.windowcovering_liftpercentage.info".

Table 51 – The Property mapping for "zcl.windowcovering_liftpercentage.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
CurrentPositionLiftPercentage	oic.r.openlevel	ocf.openLevel= CurrentPositionLiftPercentage	N/A

Table 52 provides the details of the Properties that are part of "zcl.windowcovering_liftpercentage.info".

Table 52 – The Properties of "zcl.windowcovering_liftpercentage.info".

Zigbee Property name	Type	Required	Description
CurrentPositionLiftPercentage	integer	yes	Position as a percentage between InstalledOpenLimit-Lift and InstalledClosedLimit-Lift

9.22.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_liftpercentage.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Lift Percentage - Information",
  "definitions": {
    "zcl.windowcovering_liftpercentage.info": {
      "type": "object",
      "properties": {
        "CurrentPositionLiftPercentage": {
          "type": "integer",
          "description": "Position as a percentage between InstalledOpenLimit-Lift and
InstalledClosedLimit-Lift",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
```

```
            "x-to-ocf": [
              "ocf.openLevel= CurrentPositionLiftPercentage"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_liftpercentage.info"}
  ],
  "required": ["CurrentPositionLiftPercentage"]
}
```

9.23 Window covering cluster - lift position - control

9.23.1 Derived model

The derived model: "zcl.windowcovering_liftposition.control.gotoliftvalue".

9.23.2 Property definition

Table 53 provides the detailed per Property mapping for "zcl.windowcovering_liftposition.control.gotoliftvalue".

Table 53 – The Property mapping for "zcl.windowcovering_liftposition.control.gotoliftvalue".

Zigbee Property name	OCF Resource	To OCF	From OCF
liftvalue	oic.r.openlevel	N/A	if ocf.openLevel is updated, lift value= ocf.openLevel.zcl.command.windowcovering::gotoliftvalue(liftvalue)

Table 54 provides the details of the Properties that are part of "zcl.windowcovering_liftposition.control.gotoliftvalue".

Table 54 – The Properties of "zcl.windowcovering_liftposition.control.gotoliftvalue".

Zigbee Property name	Type	Required	Description
liftvalue	integer	no	Adjust the window at the lift value.

9.23.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_liftposition.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Lift Position - Control",
  "definitions": {
    "zcl.windowcovering_liftposition.control.gotoliftvalue": {
      "properties": {
        "liftvalue": {
          "type": "integer",
          "description": "Adjust the window at the lift value.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-from-ocf": [
              "if ocf.openLevel is updated, lift value= ocf.openLevel.",
              "zcl.command.windowcovering::gotoliftvalue(liftvalue)"
```

```
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_liftposition.control.gotoliftvalue"}
  ]
}
```

9.24 Window covering cluster - lift position - information

9.24.1 Derived model

The derived model: "zcl.windowcovering_liftposition.info".

9.24.2 Property definition

Table 55 provides the detailed per Property mapping for "zcl.windowcovering_liftposition.info".

Table 55 – The Property mapping for "zcl.windowcovering_liftposition.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
CurrentPosition-Lift	oic.r.openlevel	ocf.openLevel= CurrentPosition-Lift	N/A
InstalledClosedLimit-Lift	oic.r.openlevel	ocf.range[0]= InstalledClosedLimit-Lift	N/A
InstalledOpenLimit-Lift	oic.r.openlevel	ocf.range[1]= InstalledOpenLimit-Lift	N/A

Table 56 provides the details of the Properties that are part of "zcl.windowcovering_liftposition.info".

Table 56 – The Properties of "zcl.windowcovering_liftposition.info".

Zigbee Property name	Type	Required	Description
CurrentPosition-Lift	integer	yes	Position of Window Covering from the top of the shade (cm)
InstalledClosedLimit-Lift	integer	yes	Close limit for lifting the Window Covering (cm)
InstalledOpenLimit-Lift	integer	yes	Open limit for lifting the Window Covering (cm)

9.24.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_liftposition.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Lift Position - Information",
  "definitions": {
    "zcl.windowcovering_liftposition.info": {
      "type": "object",
      "properties": {
        "InstalledClosedLimit-Lift": {
          "type": "integer",
          "description": "Close limit for lifting the Window Covering (cm)",
```

```
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-to-ocf": [
              "ocf.range[0]= InstalledClosedLimit-Lift"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "InstalledOpenLimit-Lift": {
          "type": "integer",
          "description": "Open limit for lifting the Window Covering (cm)",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-to-ocf": [
              "ocf.range[1]= InstalledOpenLimit-Lift"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        },
        "CurrentPosition-Lift": {
          "type": "integer",
          "description": "Position of Window Covering from the top of the shade (cm)",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-to-ocf": [
              "ocf.openLevel= CurrentPosition-Lift"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions//zcl.windowcovering_liftposition.info"}
  ],
  "required": [ "InstalledClosedLimit-Lift","InstalledOpenLimit-Lift","CurrentPosition-Lift"]
}
```

9.25 Window covering cluster - tilt percentage - control

9.25.1 Derived model

The derived model: "zcl.windowcovering_tiltpercentage.control.gototiltpercentage".

9.25.2 Property definition

Table 57 provides the detailed per Property mapping for "zcl.windowcovering_tiltpercentage.control.gototiltpercentage".

Table 57 – The Property mapping for "zcl.windowcovering_tiltpercentage.control.gototiltpercentage".

Zigbee Property name	OCF Resource	To OCF	From OCF
percentagetiltvalue	oic.r.openlevel	N/A	if ocf.openLevel is updated, percentage tilt value = ocf.openLevel.zcl.command.windowcovering::gototiltpercentage(percentagetiltvalue)

Table 58 provides the details of the Properties that are part of "zcl.windowcovering_tiltpercentage.control.gototiltpercentage".

Table 58 – The Properties of "zcl.windowcovering_tiltpercentage.control.gototiltpercentage".

Zigbee Property name	Type	Required	Description
percentagetiltvalue	integer	no	Adjust the window at the percentage tilt value.

9.25.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_tiltpercentage.control.json#"
,
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Tilt Percentage - Control",
  "definitions": {
    "zcl.windowcovering_tiltpercentage.control.gototiltpercentage": {
      "properties": {
        "percentagetiltvalue": {
          "type": "integer",
          "description": "Adjust the window at the percentage tilt value.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-from-ocf": [
              "if ocf.openLevel is updated, percentage tilt value = ocf.openLevel.",
              "zcl.command.windowcovering::gototiltpercentage(percentagetiltvalue)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_tiltpercentage.control.gototiltpercentage"}
  ]
}
```

9.26 Window covering cluster - tilt percentage - information

9.26.1 Derived model

The derived model: "zcl.windowcovering_tiltpercentage.info".

9.26.2 Property definition

Table 59 provides the detailed per Property mapping for "zcl.windowcovering_tiltpercentage.info".

Table 59 – The Property mapping for "zcl.windowcovering_tiltpercentage.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
CurrentPositionTiltPercentage	oic.r.openlevel	ocf.openlevel=CurrentPositionTiltPercentage	N/A

Table 60 provides the details of the Properties that are part of "zcl.windowcovering_tiltpercentage.info".

Table 60 – The Properties of "zcl.windowcovering_tiltpercentage.info".

Zigbee Property name	Type	Required	Description
CurrentPositionTiltPercentage	integer	yes	Tilt position as a percentage

9.26.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_tiltpercentage.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Tilt Percentage - Information",
  "definitions": {
    "zcl.windowcovering_tiltpercentage.info": {
      "type": "object",
      "properties": {
        "CurrentPositionTiltPercentage": {
          "type": "integer",
          "description": "Tilt position as a percentage",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-to-ocf": [
              "ocf.openlevel=CurrentPositionTiltPercentage"
            ],
            "x-from-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_tiltpercentage.info"}
  ],
  "required": ["CurrentPositionTiltPercentage"]
}
```

9.27 Window covering cluster - tilt position - control

9.27.1 Derived model

The derived model: "zcl.windowcovering_tiltposition.control.gototiltvalue".

9.27.2 Property definition

Table 61 provides the detailed per Property mapping for "zcl.windowcovering_tiltposition.control.gototiltvalue".

Table 61 – The Property mapping for "zcl.windowcovering_tiltposition.control.gototiltvalue".

Zigbee Property name	OCF Resource	To OCF	From OCF
tiltvalue	oic.r.openlevel	N/A	if ocf.openLevel is updated, tiltvalue= ocf.openLevel.zb.command.windowcovering::gototiltvalue(tiltvalue)

Table 62 provides the details of the Properties that are part of "zcl.windowcovering_tiltposition.control.gototiltvalue".

Table 62 – The Properties of "zcl.windowcovering_tiltposition.control.gototiltvalue".

Zigbee Property name	Type	Required	Description
tiltvalue	integer	no	Adjust the window at the tilt value.

9.27.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_tiltposition.control.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Tilt Position - Control",
  "definitions": {
    "zcl.windowcovering_tiltposition.control.gototiltvalue": {
      "properties": {
        "tiltvalue": {
          "type": "integer",
          "description": "Adjust the window at the tilt value.",
          "x-ocf-conversion": {
            "x-ocf-alias": "oic.r.openlevel",
            "x-from-ocf": [
              "if ocf.openLevel is updated, tiltvalue= ocf.openLevel.",
              "zb.command.windowcovering::gototiltvalue(tiltvalue)"
            ],
            "x-to-ocf": [
              "N/A"
            ]
          }
        }
      }
    }
  },
  "type": "object",
  "allOf": [
    {"$ref": "#/definitions/zcl.windowcovering_tiltposition.control.gototiltvalue"}
  ]
}
```

9.28 Window covering cluster - tilt position - information

9.28.1 Derived model

The derived model: "zcl.windowcovering_tiltposition.info".

9.28.2 Property definition

Table 63 provides the detailed per Property mapping for "zcl.windowcovering_tiltposition.info".

Table 63 – The Property mapping for "zcl.windowcovering_tiltposition.info".

Zigbee Property name	OCF Resource	To OCF	From OCF
InstalledOpenLimit-Tilt	oic.r.openlevel	ocf.range[1]= InstalledOpenLimit-Tilt	N/A
CurrentPosition-Tilt	oic.r.openlevel	ocf.openlevel= CurrentPosition-Tilt	N/A

Table 64 provides the details of the Properties that are part of "zcl.windowcovering_tiltposition.info".

Table 64 – The Properties of "zcl.windowcovering_tiltposition.info".

Zigbee Property name	Type	Required	Description
InstalledOpenLimit-Tilt	integer	yes	Open limit for tilting the Window Covering (0.1 degree)
CurrentPosition-Tilt	integer	no	Tilt position of Window Covering from open (0.1 degree)

9.28.3 Derived model definition

```
{
  "id":
"http://openinterconnect.org/zigbeemapping/schemas/zcl.windowcovering_tiltposition.info.json#",
  "$schema": "http://json-schema.org/draft-04/schema#",
  "description" : "Copyright (c) 2018 Open Connectivity Foundation, Inc. All rights reserved.",
  "title": "Window Covering Cluster - Tilt Position - Information",
  "definitions": {
    "zcl.windowcovering_tiltposition.info": {
      "type": "object",
        "properties": {
          "InstalledOpenLimit-Tilt": {
            "type": "integer",
            "description": "Close limit for tilting the Window Covering (0.1 degree)",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.openlevel",
              "x-to-ocf": [
                "ocf.range[0] = InstalledClosedLimit-Tilt"
              ],
              "x-from-ocf": [
                "N/A"
              ]
            }
          },
          "InstalledOpenLimit-Tilt": {
            "type": "integer",
            "description": "Open limit for tilting the Window Covering (0.1 degree)",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.openlevel",
              "x-to-ocf": [
                "ocf.range[1]= InstalledOpenLimit-Tilt"
              ],
              "x-from-ocf": [
                "N/A"
              ]
            }
          },
          "CurrentPosition-Tilt ": {
            "type": "integer",
            "description": "Tilt position of Window Covering from open (0.1 degree)",
            "x-ocf-conversion": {
              "x-ocf-alias": "oic.r.openlevel",
              "x-to-ocf": [
                "ocf.openlevel= CurrentPosition-Tilt"
              ],
              "x-from-ocf": [
                "N/A"
              ]
            }
          }
        }
      }
    },
    "type": "object",
    "allOf": [
      {"$ref": "#/definitions/zcl.windowcovering_tiltposition.info"}
    ],
    "required": [ "InstalledClosedLimit-Tilt","InstalledOpenLimit-Tilt","CurrentPosition-Tilt"]
```



```
}
```

