



AN1325: ZCL Advanced Platform (ZAP) User's Guide

ZAP is an advanced configuration tool within Simplicity Studio that allows the developer to manage Zigbee endpoints, clusters and commands implemented by their device. ZAP works in concert with the Zigbee Application Framework to generate code for setting up the endpoints, clusters, attributes, and commands that constitute a Zigbee application. ZAP, which is also used in Matter configuration, is known as the Zigbee Cluster Configurator (ZCL) in the Zigbee context.

KEY POINTS

- Introduces ZCL and describes how to use it.
- Covers adding and modifying endpoints.
- Describe how to configure a cluster.
- Discusses adding custom clusters.

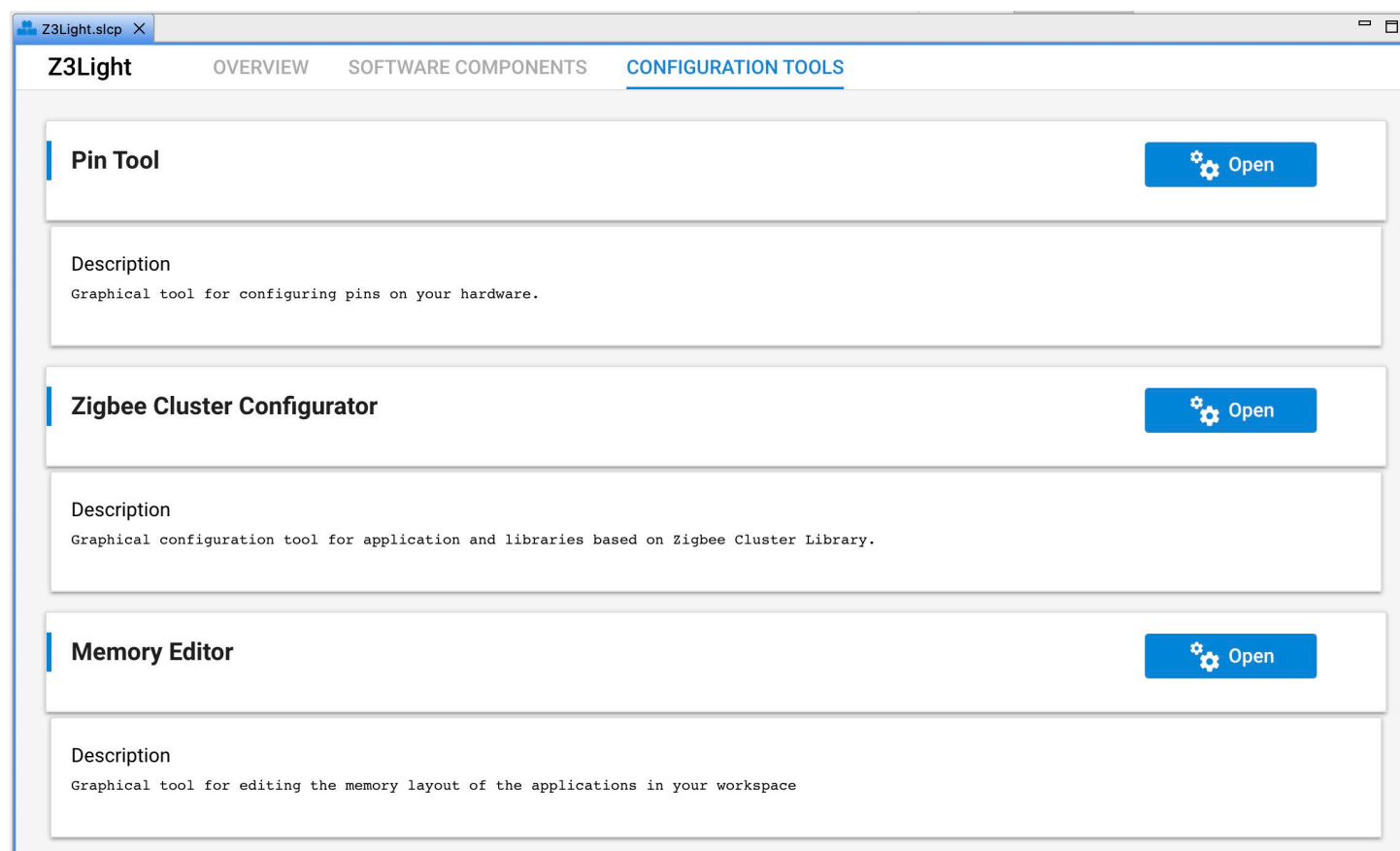
1 Introduction

ZAP, also known as the Zigbee Cluster Configurator (ZCL), is a separate tool within Simplicity Studio. It is used to generate code into the Simplicity Studio Zigbee project that is used to define important aspects of the Zigbee application, such as its endpoints, clusters, attributes, attribute storage, and commands. ZCL presents the user interface for managing these aspects of the Zigbee application. If you are familiar with the former Simplicity Studio AppBuilder, you can think of the Zigbee Cluster Configurator as replacing the ZCL tab in that interface.

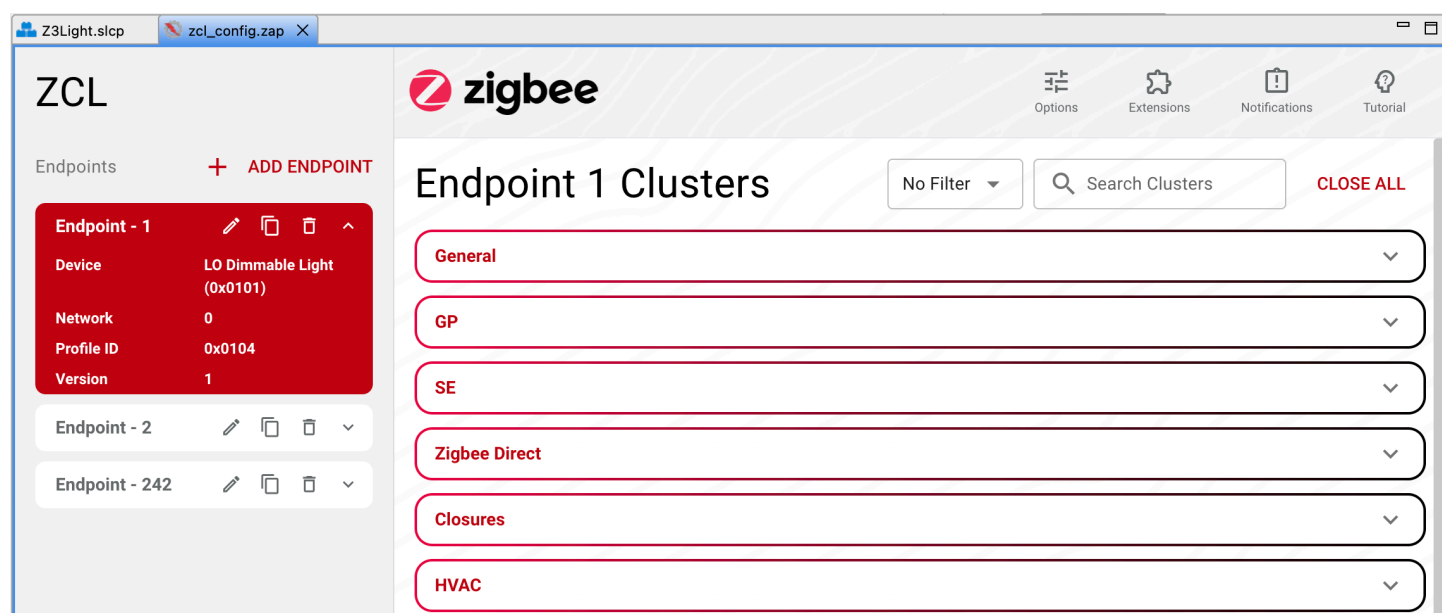
This document assumes that you have downloaded Simplicity Studio 5 and the Zigbee SDK and are familiar with other aspects of creating and configuring a Zigbee project. If not, see *QSG180: Zigbee EmberZNet SDK Quick-Start Guide for SDK 7.x and Higher* and the [Simplicity Studio v5 User's Guide](#).

2 About ZAP/ZCL

Simplicity Studio offers several ways to access or launch ZCL. The easiest way is through the Project Configurator's CONFIGURATION TOOLS tab. If the Project Configurator is not open, double-click the .slcp file in the Project Explorer view.

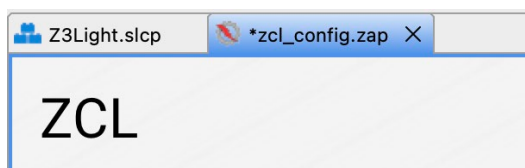


Click **Open** on the **Zigbee Cluster Configurator** card to open ZCL for your project. This launches ZCL in a new tab next to the <project>.slcp tab, with the title "zcl_config.zap." The main ZCL user interface is shown in the following figure.

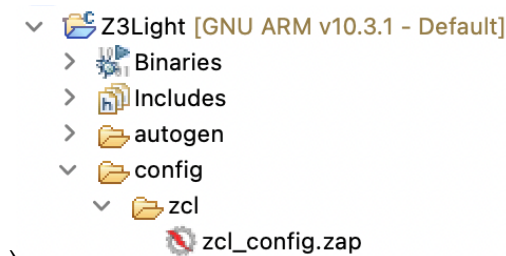


3 Saving and Generating Configuration Files

ZCL opens in edit mode, as noted by the asterisk on the tab. Save changes at any time using CTRL-S or any other file save function. If you close ZCL with unsaved changes, the tool prompts you to save.



Configuration changes made through ZCL are saved to the *zcl_config.zap* file. The .zap file is the backing data file for the ZCL configuration for your application. When you save the file, ZCL not only saves the .zap file into your project, but also automatically generates all the .c and .h files required by the Zigbee Application Framework for your Zigbee application. These files show up in two locations in your project. The .zap file is saved in the project's config > zcl folder.

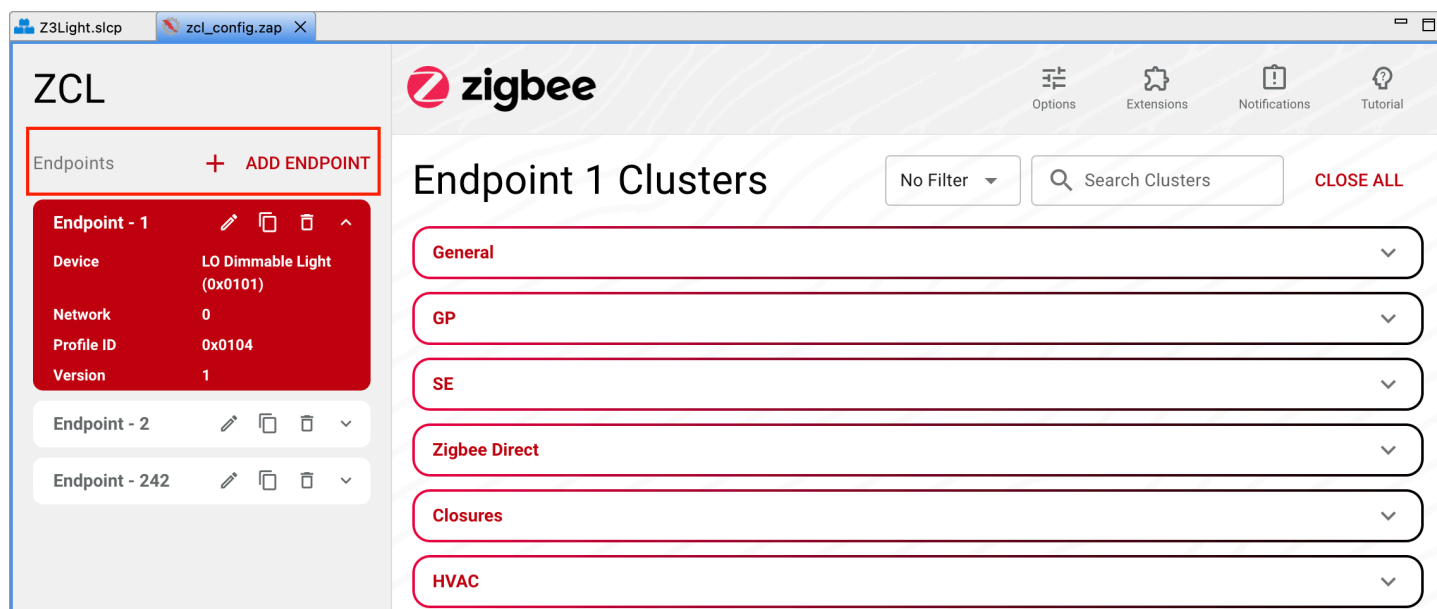


The files generated by ZCL for the application are placed in the project's autogen folder and all start with the “zap” prefix.



4 Adding Endpoints

A Zigbee application can have multiple endpoints. Each endpoint contains a device configuration made up of Clusters on that endpoint. Add a new endpoint to your application by clicking **ADD NEW ENDPOINT** in the top left corner of ZCL interface.

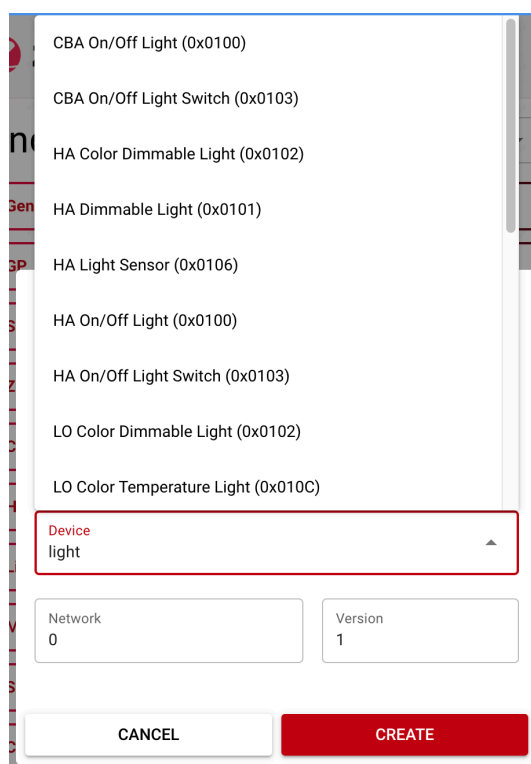


A dialog opens in which you can select the device type for the endpoint.

The 'Create New Endpoint' dialog box is shown. It contains the following fields and controls:

- Endpoint:** A text input field containing the value '3'.
- Profile ID:** An empty text input field.
- Device:** A dropdown menu.
- Network:** A text input field containing the value '0'.
- Version:** A text input field containing the value '1'.
- Buttons:** 'CANCEL' and 'CREATE' buttons at the bottom.

From here, you can select whether you would like the endpoint to represent something like a Light or a Door Lock. You can find the Zigbee device type by entering the name of the device in the **Device** field.



CBA On/Off Light (0x0100)

CBA On/Off Light Switch (0x0103)

HA Color Dimmable Light (0x0102)

HA Dimmable Light (0x0101)

HA Light Sensor (0x0106)

HA On/Off Light (0x0100)

HA On/Off Light Switch (0x0103)

LO Color Dimmable Light (0x0102)

LO Color Temperature Light (0x010C)

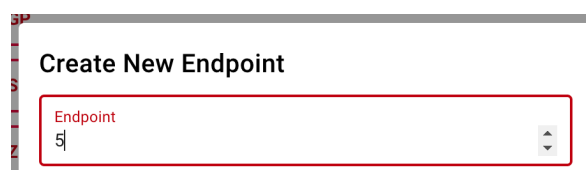
Device
light

Network
0

Version
1

CANCEL CREATE

To change the number of the endpoint on which you would like this device to appear, change the **Endpoint** setting.



Create New Endpoint

Endpoint
5

Once you have configured the endpoint, click **CREATE** to add the endpoint to your configuration.

5 Modifying an Endpoint

Select an endpoint to modify by clicking on the endpoint configuration on the left side of ZCL. The Endpoint highlighted with a blue border is the endpoint that you are in the process of modifying.

The screenshot shows the ZCL (Zigbee Cluster Library) interface. On the left, there's a sidebar with a list of endpoints. The first endpoint, 'Endpoint - 1', is highlighted with a blue border. It is a 'LO Dimmable Light' with network ID 0, profile ID 0x0104, and version 1. Below it are 'Endpoint - 2' and 'Endpoint - 242'. The main area is titled 'Endpoint 1 Clusters'. It has a search bar with 'No Filter' and 'Search Clusters' buttons, and a 'CLOSE ALL' link. Below the search bar is a list of clusters, each with a dropdown arrow: General, GP, SE, Zigbee Direct, Closures, HVAC, Lighting, Measurement & Sensing, and Security & Safety.

When the endpoint is highlighted, you can modify the clusters enabled on that endpoint in the cluster configuration editor on the right-side of ZCL.

Endpoint 1 Clusters

No Filter ▼

Search Clusters

CLOSE ALL

General ▼

GP ▼

SE ▼

Zigbee Direct ▼

Closures ▼

HVAC ▼

The dropdown tab highlighted below gives you options by which to view the clusters that are available or enabled on your endpoint. To see only clusters that are enabled on the endpoint, select the **Enabled Clusters** option.

Endpoint 1 Clusters

Enabled Clusters ▼

Search Clusters

CLOSE ALL

General

Cluster	Required Cluster	Cluster ID	Manufacturer Code	Enable	Configure
Basic	Server	0x0000	---	Server ▼	⚙️
Identify	Server	0x0003	---	Client & Server ▼	⚙️
Groups	Server	0x0004	---	Server ▼	⚙️
Scenes	Server	0x0005	---	Server ▼	⚙️
On/off	Server	0x0006	---	Server ▼	⚙️
Level Control	Server	0x0008	---	Server ▼	⚙️
⚠️ Over the Air Bootloading	Client	0x0019	---	Client ▼	⚙️

You can enable either the Client or Server (or both) sides of a cluster, by changing the **Client** and **Server** settings in the **Enable** column. If these settings are changed, you may be notified that components have been added to your project.






Endpoint 1 Clusters

No Filter ▼

Search Clusters

CLOSE ALL

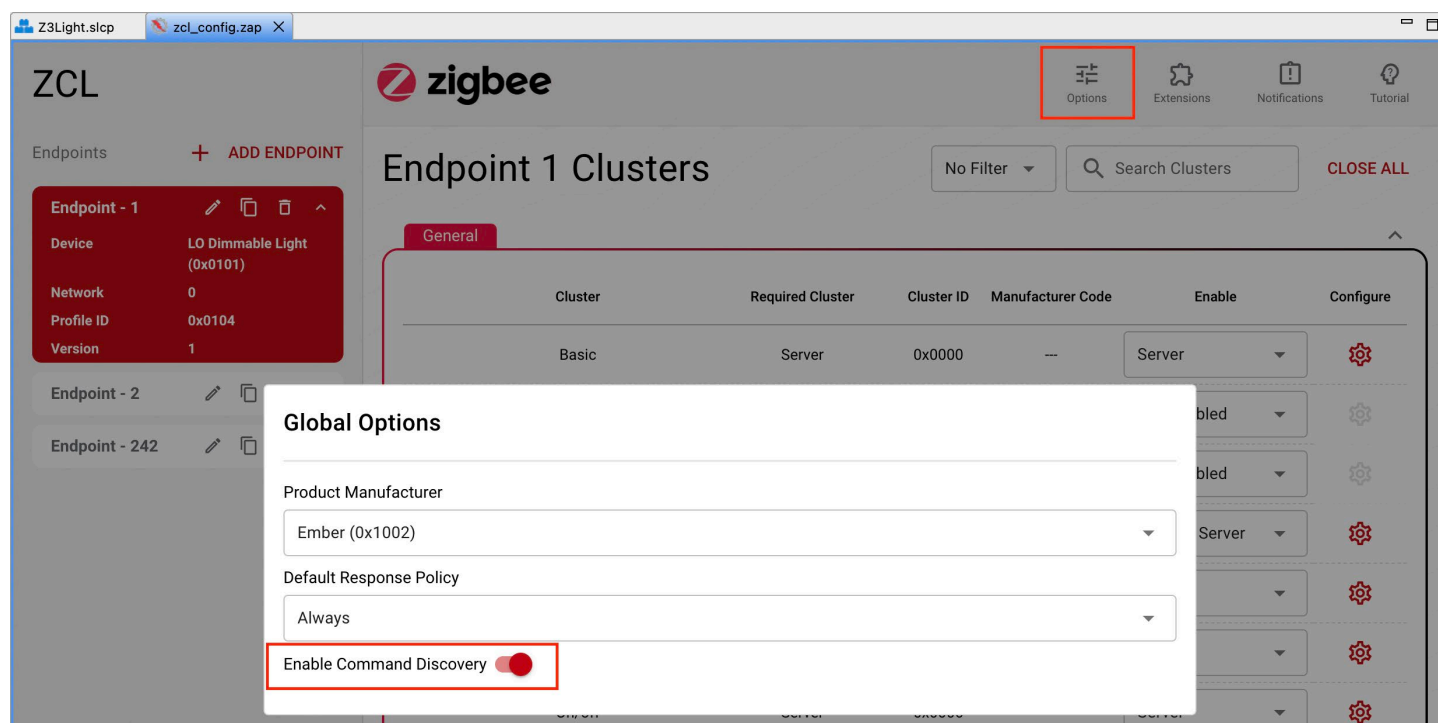
General

Cluster	Required Cluster	Cluster ID	Manufacturer Code	Enable	Configure
Basic	Server	0x0000	---	Server	
Power Configuration		0x0001	---	Not Enabled	
Device Temperature Configuration		0x0002	---	Client	
Identify	Server	0x0003	---	Server	
Groups	Server	0x0004	---	Client & Server	

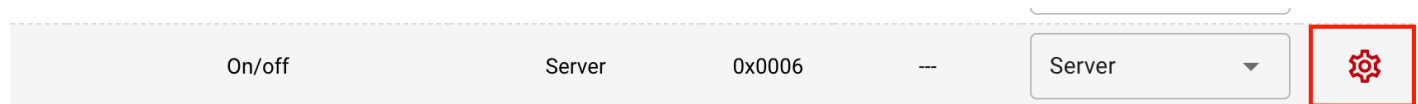
To remove the cluster entirely from the configuration, select 'Not Enabled' in the **Enable** menu.

6 Configuring a Cluster

Each cluster can be configured to implement Zigbee cluster attributes and commands. The **Enable Command Discovery** toggle, under the ZCL Global options, allows the list of commands supported by the device to be visible through the Zigbee General Command Discovery command interface.



To configure a cluster, click on the gear icon on the right side of the cluster row.



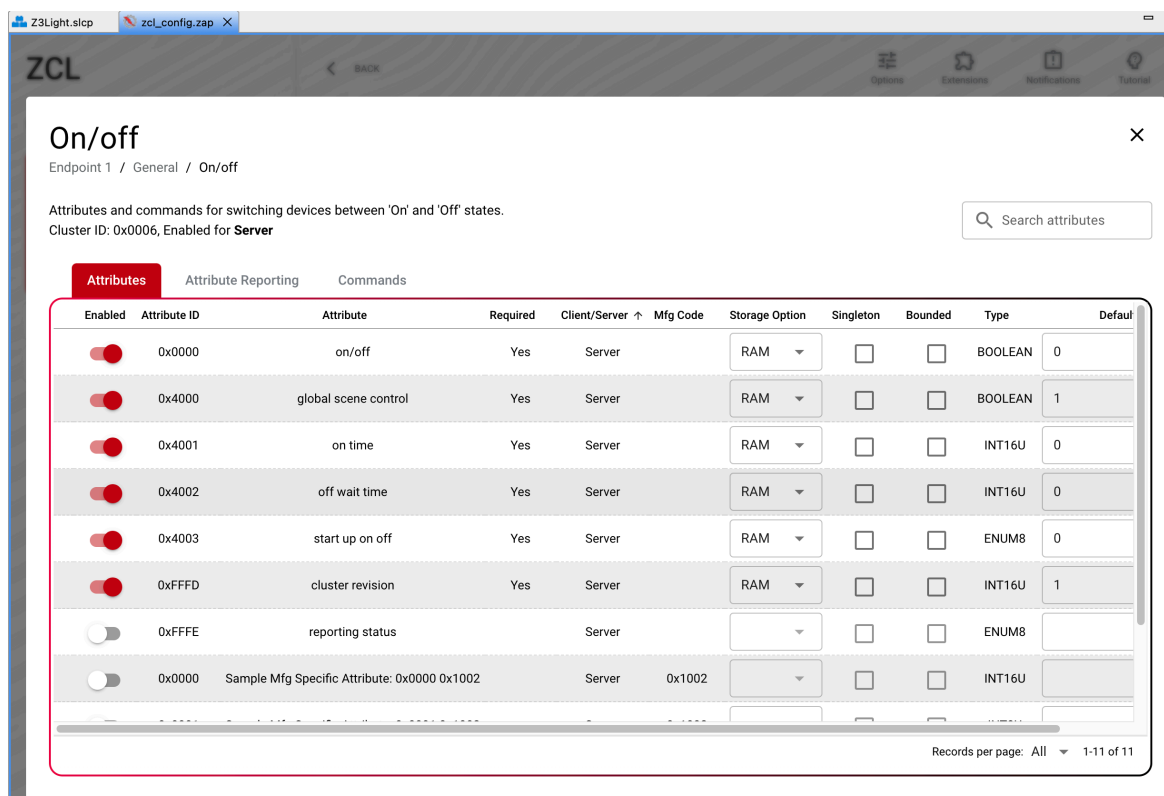
When you click the icon, a page opens specific to that cluster on that endpoint. In this interface, you can enable or disable attributes, manage how they are stored, manage attribute reporting, and manage the enabled commands on that cluster.

The cluster configuration interface consists of three tabs:

- Attributes
- Attribute Reporting
- Commands

6.1 Configuring Attributes

Attributes are configured through the first tab in the cluster configuration interface, as shown.



To enable or disable an attribute for a given cluster, click the **On/Off** toggle. When the toggle is shaded blue and to the right the attribute is on. If the toggle is grey and to the left, the attribute is off and thus not enabled for that cluster. In the figure below, the on/off attribute is implemented, whereas the reporting status attribute is not.

Enabled	Attribute ID	Attribute
<input checked="" type="checkbox"/>	0x0000	on/off
<input type="checkbox"/>	0xFFFE	reporting status

6.2 Configuring Attribute Reporting

Default attribute reporting is configured through the **Attribute Reporting** tab in the cluster configuration interface.

Attributes		Attribute Reporting		Commands					
Enabled	Attribute ID	Attribute	Reporting Policy	Client/Server ↑	Mfg Code	Min Interval	Max Interval	Type	Reportable Change
<div><div></div></div>	0x0000	on/off	suggested	Server		<div>1</div>	<div>65534</div>	BOOLEAN	not analog
<div><div></div></div>	0x4000	global scene control	optional	Server		<div>1</div>	<div>65534</div>	BOOLEAN	not analog

As with attribute enablement, default attribute reporting is controlled through the toggle interface to the left of the Attribute Reporting table. An attribute is set up to be reported by default when the toggle is on.

Attributes	Attribute Reporting	Commands
Enabled	Attribute ID	Attribute
<input checked="" type="checkbox"/>	0x0000	on/off

For instance, in the following example, Endpoint 1 only implements the Server side of the On/Off cluster. Therefore, it is only possible for the cluster to receive the Off command **In**, which is in fact enabled for that cluster.

On/off

Endpoint 1 / General / On/off

Attributes and commands for switching devices between 'On' and 'Off' states.

Cluster ID: 0x0006, Enabled for **Server**

Attributes		Attribute Reporting		Commands
Out	In	Direction	ID	Command
	<input checked="" type="checkbox"/>	Client → Server	0x00	Off

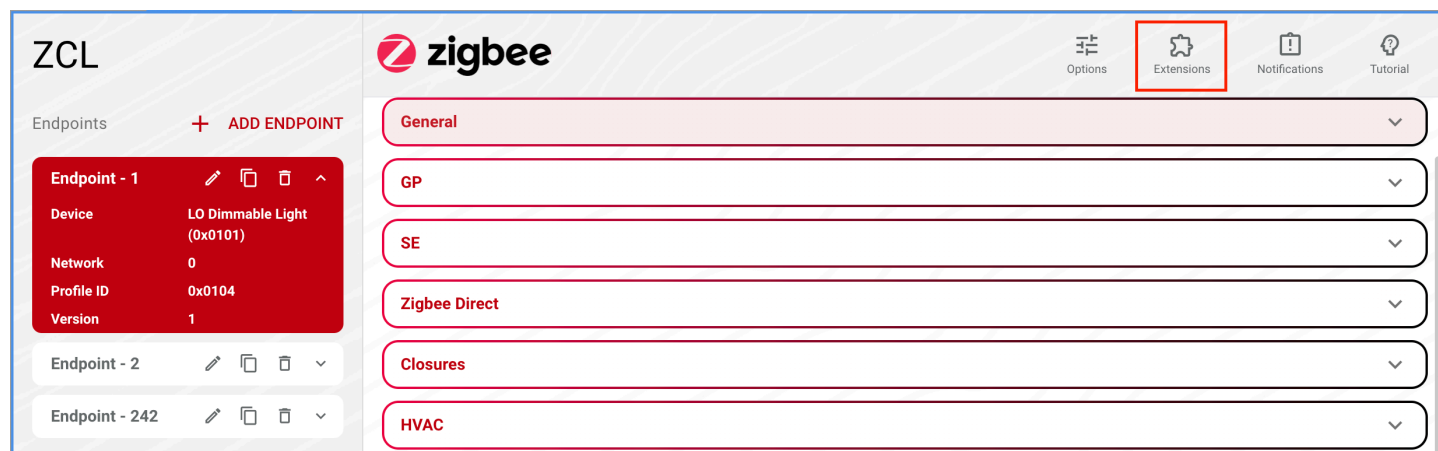
With this setting enabled, ZCL automatically generates command header code for the On/Off cluster's **Off** command.

Note: To ensure that enabled commands in the ZCL Configurator tool are handled, follow Section 6.4 ZCL Command Handling Callbacks of [UG491: Zigbee Application Framework Developer's Guide for SDK 7.0 and Higher](#).

7 Adding Custom Clusters

In Zigbee, developers can add their own custom clusters to their applications. This functionality is supported by the Zigbee Cluster Library (ZCL). The custom ZCL must be described in the Silicon Labs ZCL XML format. For examples of custom ZCL XML, see the file *sample-extensions.xml* in the <GSDK install location>/app/zcl directory. This XML file can be used as a reference for your custom ZCL XML file.

Once you have created your custom ZCL XML file, it can be added to your project through the **ZCL EXTENSIONS...** interface.







Clicking **ZCL EXTENSIONS...** opens the Custom ZCL page in ZCL.

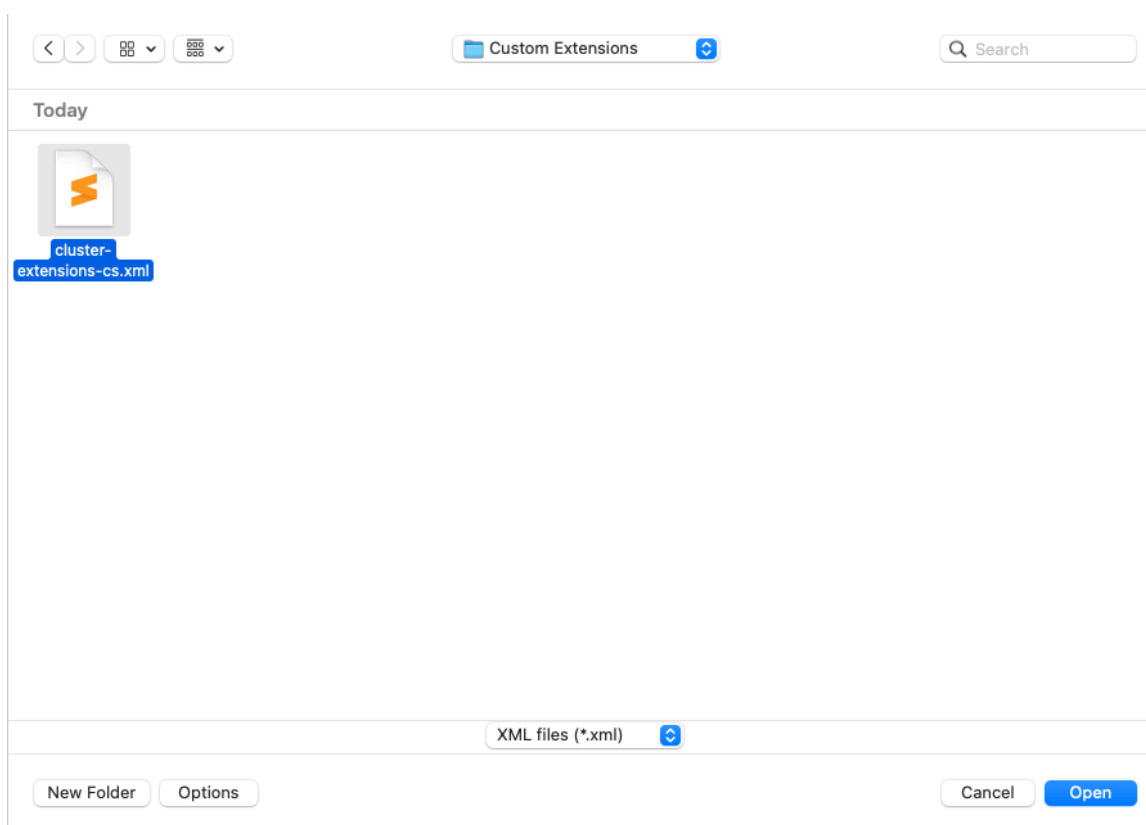
Add Custom ZCL

You can use this functionality to add custom ZCL clusters or commands to the Zigbee Clusters Configurator



zcl-zap.json	 DELETE	▼	
gen-templates.json	 DELETE	▼	

Click **Add** to add your custom ZCL XML to the project. Browse to the location of the custom ZCL XML, select a file, and click **Open**.



Once the custom ZCL XML has been read into ZCL, your custom clusters, attributes, commands, and so on are accessible to the configuration of your application.

Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!



IoT Portfolio
www.silabs.com/IoT



SW/HW
www.silabs.com/simplicity



Quality
www.silabs.com/quality



Support & Community
www.silabs.com/community

Disclaimer

Silicon Labs intends to provide customers with the latest, accurate, and in-depth documentation of all peripherals and modules available for system and software implementers using or intending to use the Silicon Labs products. Characterization data, available modules and peripherals, memory sizes and memory addresses refer to each specific device, and "Typical" parameters provided can and do vary in different applications. Application examples described herein are for illustrative purposes only. Silicon Labs reserves the right to make changes without further notice to the product information, specifications, and descriptions herein, and does not give warranties as to the accuracy or completeness of the included information. Without prior notification, Silicon Labs may update product firmware during the manufacturing process for security or reliability reasons. Such changes will not alter the specifications or the performance of the product. Silicon Labs shall have no liability for the consequences of use of the information supplied in this document. This document does not imply or expressly grant any license to design or fabricate any integrated circuits. The products are not designed or authorized to be used within any FDA Class III devices, applications for which FDA premarket approval is required or Life Support Systems without the specific written consent of Silicon Labs. A "Life Support System" is any product or system intended to support or sustain life and/or health, which, if it fails, can be reasonably expected to result in significant personal injury or death. Silicon Labs products are not designed or authorized for military applications. Silicon Labs products shall under no circumstances be used in weapons of mass destruction including (but not limited to) nuclear, biological or chemical weapons, or missiles capable of delivering such weapons. Silicon Labs disclaims all express and implied warranties and shall not be responsible or liable for any injuries or damages related to use of a Silicon Labs product in such unauthorized applications.

Note: This content may contain offensive terminology that is now obsolete. Silicon Labs is replacing these terms with inclusive language wherever possible. For more information, visit www.silabs.com/about-us/inclusive-lexicon-project

Trademark Information

Silicon Laboratories Inc.[®], Silicon Laboratories[®], Silicon Labs[®], SiLabs[®] and the Silicon Labs logo[®], Bluegiga[®], Bluegiga Logo[®], EFM[®], EFM32[®], EFR, Ember[®], Energy Micro, Energy Micro logo and combinations thereof, "the world's most energy friendly microcontrollers", Redpine Signals[®], WiSeConnect, n-Link, ThreadArch[®], EZLink[®], EZRadio[®], EZRadioPRO[®], Gecko[®], Gecko OS, Gecko OS Studio, Precision32[®], Simplicity Studio[®], Telegesis, the Telegesis Logo[®], USBXpress[®], Zentri, the Zentri logo and Zentri DMS, Z-Wave[®], and others are trademarks or registered trademarks of Silicon Labs. ARM, CORTEX, Cortex-M3 and THUMB are trademarks or registered trademarks of ARM Holdings. Keil is a registered trademark of ARM Limited. Wi-Fi is a registered trademark of the Wi-Fi Alliance. All other products or brand names mentioned herein are trademarks of their respective holders.



Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

www.silabs.com