DALI Controller Module

> **Note!**
> The described functionality and integration is available for **GRENTON DALI Controller DIN, Eth (INT-202-D-01)** with **2.0.5 (build 2440)** or higher.

> **Note!**
> DALI Controller is available for Object Manager in version **1.10.1 (build 242102)** and higher, and for CLU with firmware **5.13.08 (build 2442)** and higher.

1. General information

The DALI Controller module acts as a master device, in accordance with the DALI standard, it enables the operation of 64 ballasts - Control Gears, connected to the DALI bus.

> **Note!**
> The maximum number of ballasts (`DALI_GEAR` objects) assigned to one CLU Z-Wave is 128.

DALI Controller allows you to control all light control devices within the scope defined by the PN-EN 62386-102 standard, and the DT8 extension.

The module allows you to control single ballasts, as well as control by groups, each ballast can be assigned to 16 groups. Thanks to this, it is much easier to organize the lighting control and create advanced control scenarios.

2. Module configuration

> **Note!**
> Before starting any work with the DALI Controller module, it is necessary to update the interface database!

LED signaling

- The blue diode indicates the voltage on the DALI bus,
- The green diode indicates the current state of the module:
    - ON – no ballast configuration on module, DALI Discovery must be performed,
    - Flashes at 200 ms interval – DALI Discovery, the ballasts connected to the DALI bus are searched and local addresses assigned to them,
    - Flashes at 1 second interval – ballast configuration is on the module.

Adding a module to the project

After the CLU Discovery process has been executed, two objects appear in the project:

- `DALI_MASTER` - main object used to manage the module configuration,
- PowerSupplyVoltage - object for monitoring the voltage on the system bus.

A. Ballast addressing

The module configuration should start with addressing the DALI ballasts connected to the bus. The DALI Controller enables two types of addressing: fully automatic or manual.

Automatic addressing allows you to address the entire installation with one click, using the DALI Discovery process.

- In the `DALI_MASTER` object in the `Control` tab, call the `ResetGear (Broadcast)` method and then the `DALI_Discovery` with chosen parameter:
    - GetConfiguration,
    - NewWithoutLocalAddress,
    - ResetAllLocalAddress
- Calling the method with the `GetConfiguration` parameter initiates automatic retrieval of local ballast addresses located on the bus from the range 0 to 63.

> **Note!**
> During this process all addresses are retrieved, if there are duplicates on the bus they will remain unchanged, in which case the conflicts should be resolved using the ResolveAddressDuplicate method!
>
> The controller memory is not deleted, the previous configuration will remain!

- Calling the method with the `NewWithoutLocalAddress` parameter initiates automatic addressing of new ballasts without an assigned address on the bus, which will receive local addresses from the range 0 to 63. The address assignment will be confirmed by lighting up the given luminaire for 300 ms. During DALI Discovery with the `NewWithoutLocalAddress` parameter, the addresses of ballasts that were already on the bus will remain unchanged.

- Calling the method with the `ResetAllLocalAddress` parameter initiates automatic addressing of all ballasts on the bus, which will receive local addresses from 0 to 63. The assignment of the address will be confirmed by lighting up the given luminaire for 300 ms. It should be noted that after starting the addressing, all previous addresses will be deleted. During DALI Discovery with the `ResetAllLocalAddress` parameter, addresses are assigned to ballasts randomly,

- During DALI Discovery:
    - The green LED on the DALI Controller flashes at 200 ms interval,
    - The embedded feature `State` of the `DALI_MASTER` object takes the value 1.

The duration of the DALI Discovery depends on the number of ballasts (it can take up to several minutes for the maximum number of devices).

> **Note!**
> Do not perform any operations on the DALI Controller during DALI Discovery!

Manual addressing allows you to address individual ballasts using the `ChangeLocalAddress` method. It is helpful in the event that the ballast is not found after DALI Discovery, the address is doubled or we want a specific sequence of addresses in accordance with the assembly order.

- In the `DALI_MASTER` object, after going to the `Control` tab, call the `ChangeLocalAddress` method with the parameters:
    - `ActualAddress` - current ballast address,
    - `AddressToSet` - new unoccupied address that will be assigned to the device,
- In the `DALI_MASTER` object, after going to the `Control` tab, call the `ResolveAddressDuplicate` method with the Address parameter:

- Calling the method resolves address conflicts on the Dali bus, in the case when the address is duplicated for one of the ballasts, the first free address that is not in the module configuration is assigned
- During `ResolveAddressDuplicate`:
    - The green diode on the DALI Controller module flashes at an interval of 200 ms,
    - The built-in feature `State` of the `DALI_MASTER` object takes the value 1.

> **Note!**
> Do not perform any operations on the DALI Controller during `ResolveAddressDuplicate`!

After the DALI Discovery

- The green LED on the DALI Controller flashes every 1 s (ballasts found) or is on continuously (no ballasts found),
- The embedded feature `State` of the `DALI_MASTER` object takes the value:
    - 3 - ballasts found,
    - 0 - no ballasts found,
- The embedded feature `NumberOfGear` of the `DALI_MASTER` returns the number of correctly found and addressed devices,
- The event `OnDALI_DiscoveryCompleated` is generated.

Operations possible on devices after DALI Discovery has ended

Using the methods of the `DALI_MASTER` object we can:

- Verify the device reporting to the given address - the `Identify` method,
- Restart the device at the given address - the `ResetGear` method,
- Set the value of the luminaire for the device at the given address - the `SetDAPCValue` method.

B. Adding ballasts to the project

After the ballast addressing process is completed with the `DALI_Discovery` and `SetLocalAddress` methods, CLU Discovery should be performed:

- New GEAR objects are added to the project to represent each DALI device (address) correctly found and added during the addressing process,
- The embedded `GearAddresses` feature of the `DALI_MASTER` object returns address numbers in the range 0 - 63, occupied by DALI devices,
- GEAR objects are in the `DALI_GEAR` and `DALI_GEAR_DT8` - Device Type 8 versions:
    - `DALI_GEAR` - all ballasts with basic control methods,
    - `DALI_GEAR_DT8` - ballasts for color control (RGBWA control mode) or color temperature (Tc control mode).

> **Note!**
> For correct operation of GEAR configuration and objects, CLU Discovery should be performed after each change in ballast addressing!

C. Ballast control

The control of a single ballast is carried out using a given `DALI_GEAR` / `DALI_GEAR_DT8` object using available methods or using the methods of the `DALI_MASTER` object (detailed functionalities can be found in the description of individual objects).

The ballast groups are controlled by the `DALI_MASTER` object using the `SetGroupDAPCValue`, `GroupSwitch`, `GroupSwitchOn`, `GroupSwitchOff` methods. In order to be able to control a given group of devices, it is necessary to:

- For the desired GEAR objects, set the value of the embedded feature `Group`. Each object can be assigned to 16 groups in the range 1 - 16, the next groups are given after a decimal point,
- After assigning objects to groups, send the configuration to CLUZ,
- After sending the configuration, the groups are sent by the DALI Controller. Embedded feature `State` of the `DALI_MASTER` object takes the value 4. The duration of the process depends on the number of devices for which the value of the `Group` feature has been changed, it can last up to 60 seconds,
- After correct grouping, the embedded feature of the `DALI_MASTER` object takes the value 3.

> **Note!**
> When assigning groups (after CLUZ restart / configuration sending) it is not possible to control the objects!

D. RampTime

The DALI Controller supports the smooth change of the `DAPCValue` value using the `RampTime` parameter, in a logarithmic manner:

RampTime	Minimum fade time [s]	Nominal fade time [s]	Maximum fade time [s]
1	0,6	0,7	0,8
2	0,9	1,0	1,1
3	1,3	1,4	1,6
4	1,8	2,0	2,2
5	2,5	2,8	3,1
6	3,6	4,0	4,4
7	5,1	5,7	6,2
8	7,2	8,0	8,8
9	10,2	11,3	12,4
10	14,4	16,0	17,6
11	20,4	22,6	24,9
12	28,8	32,0	35,2
13	40,7	45,3	49,8
14	57,6	64,0	70,4
15	81,5	90,5	99,6

3. Objects

A. DALI_MASTER

FEATURES

Name	Description
`State`	`0` - no ballast configuration `1` - DALI Discovery `2` - Short circuit/no power supply on the bus `3` - ballast configuration is on the device `4` - saving information about groups
`NumberOfGear`	Number of active ballasts connected to the device. The feature value is refreshed after restarting the system or calling the UpdateMissingGears method
`GearAddresses`	Ballast addresses given during DALI_Discovery. The feature value is refreshed after Dali Discovery
`ActiveGears`	Addresses of active ballasts. The feature value is refreshed after system restart or calling the `UpdateMissingGears` method
`MissingGears`	Addresses of inactive ballasts. The feature value is refreshed after system restart or calling the `UpdateMissingGears` method

METHODS

Name	Description
`Identify`	Turns on the luminaire for 2 seconds
`ResetGear`	Restarts the ballast
`ResetLocalAddress`	Removes the LocalAddress of the selected ballast. `RemoveFromController` parameter: `True` - removes the ballast address from the controller memory `False` - does not remove the ballast address from the controller memory
`ChangeLocalAddress`	Changes the current LocalAddress of the ballast to the selected one, **address is not deleted from Dali Controller memory**
`RemoveGear`	Removes the ballast from the controller memory
`ResolveAddressDuplicate`	Resolves address conflicts on the Dali bus
`UpdateMissingGears`	Checks the activity of the ballasts that are in the configuration
`DALI_Discovery`	Searching for ballasts connected to the DALI bus and assigning them local addresses. When an address is assigned, the ballast is switched on for 300 ms. *Parameters:* `GetConfiguration`, `NewWithoutLocalAddress`, `ResetAllLocalAddress` A detailed description of the parameters can be found in the section [Addressing ballasts](#) During `DALI_Discovery`, do not perform any operations on the device!
`SetPowerOnLevel`	For the selected address, it saves in the ballast memory the DACPValue that is to be set after a restart/bus failure
`SetDAPCValue`	Sets the value of the power with which the luminaire shines. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SetLastActiveLevel`	If the ballast is off, turns on the ballast with the DACPValue value set before it was turned off
`SetGroupDAPCValue`	Sets the value of the power with which the luminaire shines for a given group. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`GroupSwitch`	Changes the output state for a ballast group to the opposite `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`GroupSwitchOn`	Turns on the luminaire for a given group. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`GroupSwitchOff`	Turns off the luminaire for a given group. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]

EVENTS

Name	Description
`OnDALI_DiscoveryCompleated`	Event occurring after the ballasts have been found and given local addresses
`OnDALI_BusPowerFailure`	Event generated: when there is no power supply to the DALI bus for more than 1s, when there is a short circuit on the DALI bus

B. DALI_GEAR

FEATURES

Name	Description
`Address`	Ballast address
`Group`	Ballast group numbers, subsequent groups from the 1-16 range are given after the decimal point. `0` - no belonging to any group
`DAPCValue`	The value of the power with which the luminaire shines

METHODS

Name	Description
`Identify`	Turns on the luminaire for 2 seconds
`SetDAPCValue`	Sets the value of the power with which the luminaire shines. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`Switch`	Changes the luminaire state to the opposite (0 / 254). `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SwitchOn`	Turns on the luminaire. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SwitchOff`	Turns off the luminaire. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`Hold`	Executes the function of illuminating / dimming the luminaire
`HoldUp`	Executes the function of illuminating the luminaire
`HoldDown`	Executes the function of dimming the luminaire

EVENTS:

Name	Description
`OnDAPCValueChange`	Event occurring when changing the `DAPCValue`
`OnSwitchOn`	Event occurring when the `DAPCValue` value is changed from 0 to the greater value
`OnSwitchOff`	Event occurring when the `DAPCValue` value is changed to 0

C. `DALI_GEAR_DT8`

FEATURES

Name	Description
`Address`	Ballast address
`Group`	Ballast group numbers, subsequent groups from the 1-16 range are given after the decimal point. `0` - no belonging to any group
`DAPCValue`	The value of the power with which the luminaire shines
`HSVValue`	Brightness value as per the HSV model (range: 0.00-1.00). The feature does not get the actual brightness of the luminaire! Set according to called `SetHSVValue` method
`HSVSaturation`	Color saturation value as per the HSV model (0.00-1.00). This feature does not get the actual color saturation of the luminaire! Set according to called `SetHSVSaturation` method
`HSVHue`	Color hue value as per the HSV model (0-360). The feature does not get the actual color of the luminaire! Set according to called `SetHSVHue` method
`ColourTemperature`	The feature does not retrieve the actual color state of the luminaire! Set based on the `SetColourTemperature` method being called.

METHODS

Name	Description
`Identify`	Turns on the luminaire for 2 seconds
`SetDAPCValue`	Sets the value of the power with which the luminaire shines
`Switch`	Changes the luminaire state to the opposite (0 / 254). `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SwitchOn`	Turns on the luminaire. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SwitchOff`	Turns off the luminaire. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`Hold`	Executes the function of illuminating / dimming the luminaire
`HoldUp`	Executes the function of illuminating the luminaire
`HoldDown`	Executes the function of dimming the luminaire
`SetHSVValue`	Sets brightness value (0.00-1.00). `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SetHSVSaturation`	Sets saturation value (0.00-1.00). `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SetHSVHue`	Sets hue value (0-360). `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SetRGBValue`	Sets the value of the R (Red), G (Green), B (Blue) channels. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SetWAFValue`	Sets the value of the W (White) channel, and the A (Amber) and F (Freecolor) parameters. `RampTime` parameter set on a logarithmic scale 0.8 - 90 [s]
`SetColorTemperature`	Sets the color temperature value, where 0 - physical minimum, 100 - physical maximum. RampTime parameter set on a logarithmic scale 0.8 - 90 [s]

EVENTS:

Name	Description
`OnDAPCValueChange`	Event occurring when changing the `DAPCValue`
`OnSwitchOn`	Event occurring when the `DAPCValue` value is changed from 0 to the greater value
`OnSwitchOff`	Event occurring when the `DAPCValue` value is changed to 0

D. PowerSupplyVoltage

FEATURES

Name	Description
`Value`	Current output value taking into account the scalar
`Value%`	Current percentage input value of the maximum value (`MaxValue` characteristic)
`Sensitivity`	Minimum change of input state when the `OnValueChange`, `OnValueLower` or `OnValueRise` event is generated
`MinValue`	Minimum value of the Value characteristic after exceeding which the `OnOutOfRange` event is generated
`MaxValue`	Maximum value of the Value characteristic after exceeding which the `OnOutOfRange` event is generated

METHODS

Name	Description
`SetSensitivity`	Sets input sensitivity value
`SetMinValue`	Sets `MinValue`
`SetMaxValue`	Sets `MaxValue`

EVENTS:

Name	Description
`OnValueChange`	Event resulting from changing input state
`OnValueLower`	Event occurs when a value lower than the value from the last reading appears at input
`OnValueRise`	Event occurs when a value higher than the value from the last reading appears at input
`OnOutOfRange`	Event resulting from exceeding the permissible range (`MinValue` : `MaxValue`)
`OnInRange`	Event occurs when value returns to `MinValue` / `MaxValue` range