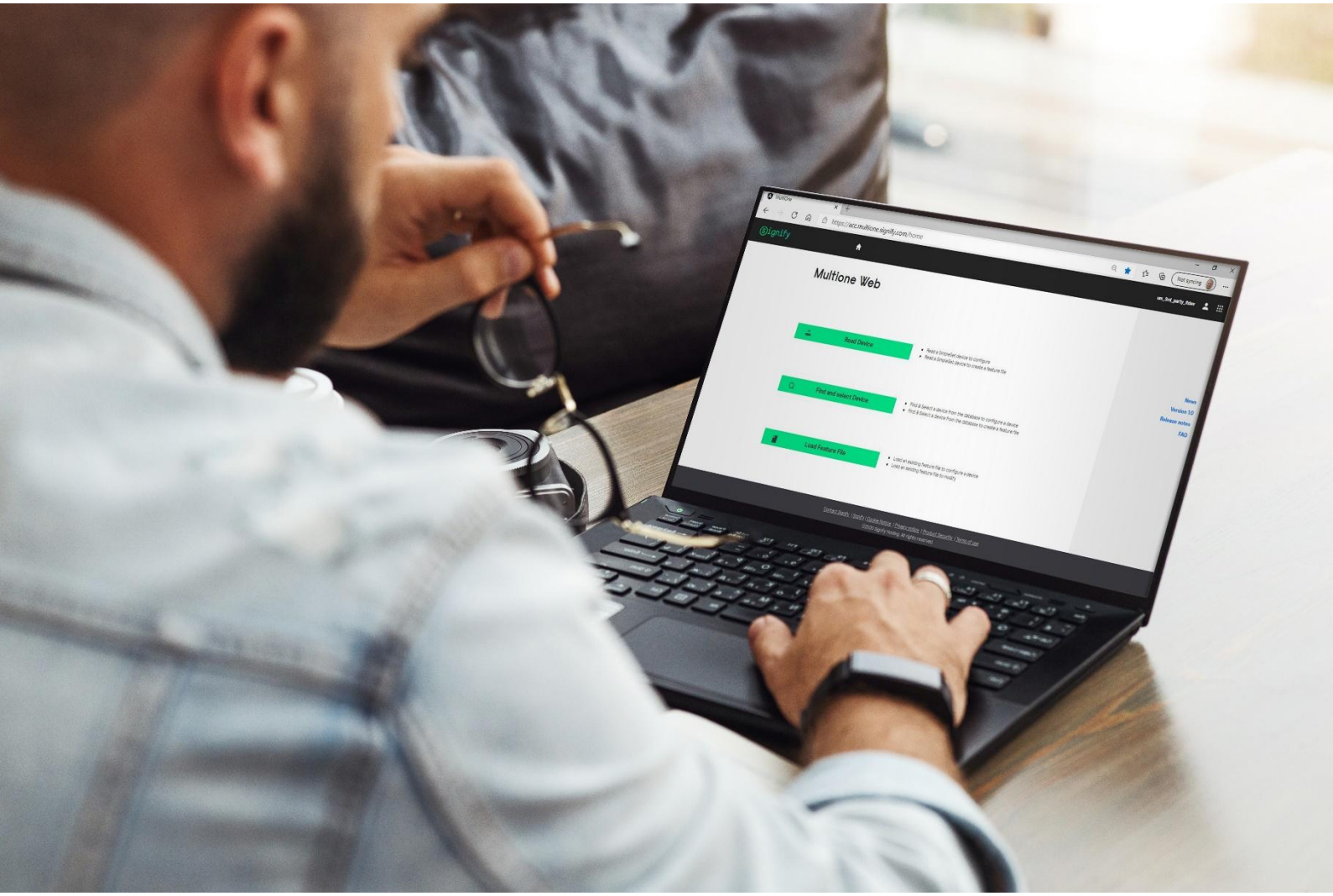
MultiOne by Signify

MultiOne Cloud

Configuration tooling

MultiOne Cloud 1.33



Application Guide for MultiOne Cloud

Document version September 2026

© 2022 Signify Holding. All rights reserved. The information provided herein is subject to change, without notice. Signify does not give any representation or warranty as to the accuracy or completeness of the information included herein and shall not be liable for any action in reliance thereon. The information presented in this document is not intended as any commercial offer and does not form part of any quotation or contract, unless otherwise agreed by Signify. All trademarks are owned by Signify Holding or their respective owners.

Signify

About this document

This document describes

1. How to connect to MultiOne Cloud using Postman.
2. The API (Application Programming Interface) calls for the generation of feature files.
3. Example syntax for all driver features useful for configuration.

Scope and Purpose

MultiOne cloud features and use of REST APIs (Representational State Transfer Application Programming Interface) with Postman application that supports your automation system.

Intended Audience

This document is intended for software experts and developers of the OEM who are involved to the fullest in the integration of MultiOne Cloud in the automation systems.
For access, contact multione@signify.com (or) regional Key Account Managers.

Privacy notes and Terms Agreement

For the privacy notes for MultiOne Cloud and MultiOne Web visit our site:
https://www.signify.com/global/privacy/legal-information/privacy-notice
The term of conditions for these tools can be found on:
https://www.signify.com/global/terms-of-use

Version history

Date	Changes
22 Oct 2020	Added items SR Power Removed in feature Dimming Interface “AmpDim”, since this is not yet supported Minor textual changes
22 Nov 2021	Added comment on override possibilities for the Dimming Interface minor textual change
15 Mar 2022	Encryption of feature file removed API commands modified to support multiple calls in one session Added features: - 0-10V interface Removed features: - Corridor Mode - Touch and Dim
11 Jun 2022	Added features Beta - MidnightShift calculator - MTP (Module Temperature Protection) - AmpDim Note – In this version ‘MidpointShift’ is changed to ‘MidnightShift’ The version is backward compatible Works for Production url and Beta url (See supported device List for June 2022) https://api.multione.signify.com/auth/api/v1 https://api.beta.multione.signify.com/auth/api/v1
13 Dec 2022	API call for list of device features Added features - Corridor Mode - Touch and Dim - OWP (OEM Write Protection) - Luminaire info Format 0 The version is backward compatible Works only for Production url (See supported device List for December 2022) https://api.multione.signify.com/auth/api/v1 No changes were made in the Beta environment Note (1) All the feature names are case insensitive. For E.g., Feature name ‘AOC’ can be written as aoc, Aoc Feature name ‘DimmingInterface’ can be written as dimminginterface, dimmingInterface Note (2) Under Dimming Interface, feature parameters of Selected Interface and Override Interface are case sensitive – Refer the Dimming Interface Section Note (3) Under feature name OWP/Owp/owp, feature parameters are case sensitive Refer the OWP section

16 Aug 2023	Updated features - AOC (Adjustable Output Current) - Midnight Shift Calculator - Driver Temperature Limit - Luminaire Info format 3 (DALI part 251) - AST (Adjustable Start-up Time) - ZTV (Zero to Ten Volt) - Corridor - Touch and Dim - OWP (OEM Write Protection) - Dynadimmer *The feature name "Dynadim" is deprecated. It is replaced with "Dynadimmer"* Added features - Retrieve driver production info - Coded Mains
1 Jan 2024	Updated features - MTP Added features - DALI 253M Added Features Metadata API call
3 Mar 2024	Updated features - SR Power/DALI Power supply *The feature name "SRPSU" is deprecated. It is replaced with "DALIPSU"* - Luminaire info Format 3 New search API call to present EDIT information
27 May 2024	Updated feature definitions Removed Beta references as it is no longer viable.
12 July 2024	Updated Features - FlexTune - Multi Config Mode - Dynadimmer 10 Scenes with CCT
31 August 2024	Updated Features - Dynadimmer 10 Scenes
11 October 2024	Updated Features - OWP – update for MultiOne Web to save feature file, set password and configure.
9 December 2024	Added API call for Device configuration file generation.

23 April 2025	Documentations now can be accessed through MyLighting portal, links and references in the document has been changed.
07 July 2025	DALI Part 251 update - Luminaire information formats 4&5 added
07 Agusut 2025	ZTV Update - ZTV update including versions 3-4-5-6
16 December 2025	DC Emergency update
19 February 2026	New Feature - DALI 102 Addressing
02 April 2026	New Feature - DALI 102 and 207 configuration
22 June 2026	New Feature - AOC MultiChannel
07 August 2026	New Feature - Over temperature protection (OTP)
26 August 2026	New Feature - Direct sensor control
21 September 2026	TD Update - Touch Dim with CCT

Table of Contents

About this document 2
Version history 3
Introduction 8
API sequence for MultiOne 9
API calls for MultiOne Cloud 10
Authentication 11
Configuration 11
Supported features 12
Feature configuration 13
AOC – Adjustable Output Current 13
AOC MultiChannel – Adjustable Output Current Multi Channel 14
ALO (Adjustable Light Output) 15
CLO (Constant Light Output) 16
CLOlite (Constant Light Output Lite) 17
Coded Mains 18
Dimming Interface 20
Dynadimmer 21
*Dynadimmer 10 scenes** 23
Dynadimmer 10 scenes with CCT 26
Midnight Shift Calculator 28
DC Emergency 29
Minimum Dim Level 30
LineSwitch 31
DALI Power Supply Unit (DALIPSU) 33
DTL (Driver Temperature Limit) 34
Luminaire Info format (DALI part 251) 35
AST (Adjustable Start-up Time) 37
ZTV (Zero to Ten Volt) 38
DALI 102 Addressing 39
DALI 102 and 207 Configuration 40
Amp Dim 42
MTP (Module Temperature Protection) 43
Corridor 45
Touch and Dim 46
Touch and Dim with CCT (only for FlexTune Drivers) 47
DiiA 253 48

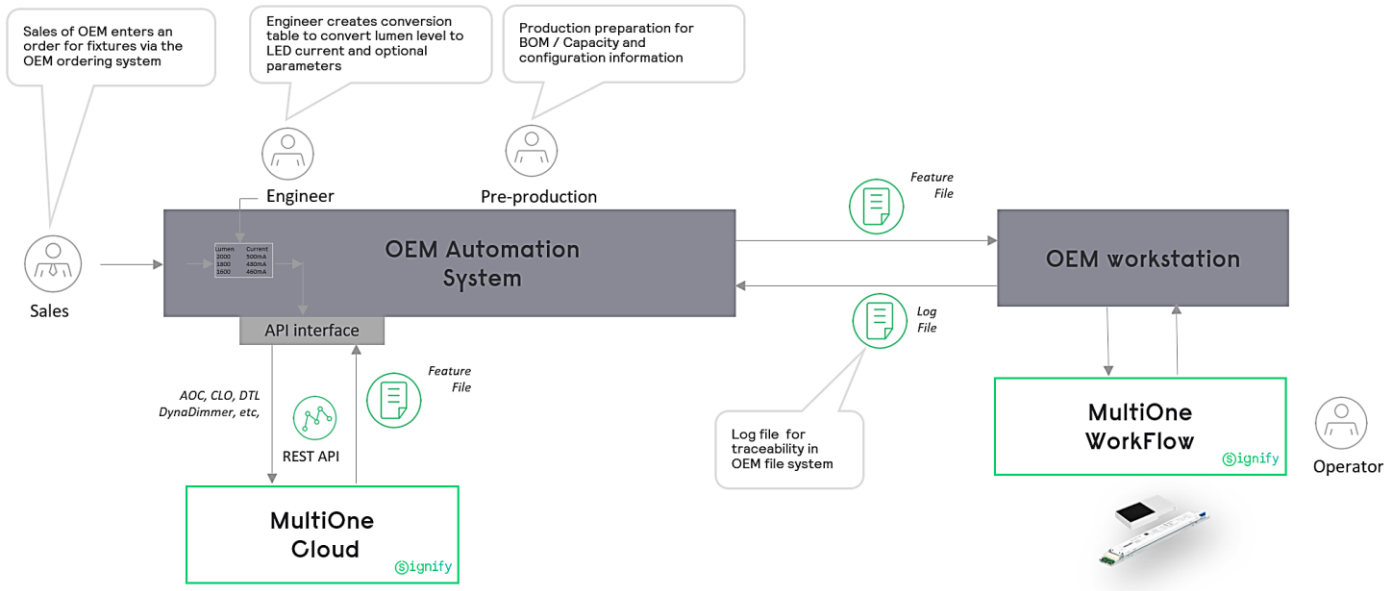
OWP (OEM Write Protection) 49
FlexTune 50
Direct Sensor Control(DSC) 51
Multi Config Mode 52
LSI 53
Over temperature protection (OTP) 54
Multiple feature configuration with a single API call 55
Feature file Generation 56
References 57

List of Tables

Table 1 : Main API Calls 10
Table 2 : Additional API Calls 10
Table 3 : Feature call for AOC – Adjustable Output Current 13
Table 3 : Feature call for AOC – Adjustable Output Current 14
Table 4: Feature call for ALO (Adjustable Light Output) 15
Table 5: Feature call for CLO – Constant Light Output 16
Table 6: Feature call for CLOlite – Constant Light Output Lite 17
Table 7: Feature call for Coded Mains 18
Table 8: Feature call for Dimming Interface 20
Table 9: Feature call for Dynadimmer 21
Table 10: Feature call for Dynadimmer 23
Table 11: Feature call for Dynadimmer 26
Table 12: Feature call for Midnight Shift Calculator 28
Table 13: Feature call for DC Emergency 29
Table 14: Feature call for Minimum Dim Level 30
Table 15: Feature call for Lineswitch 31
Table 16: Feature call for DALI Power Supply Unit 33
Table 17: Feature call for Driver Temperature Limit 34
Table 18: Feature call for Luminaire Info format (DALI part 251) 36
Table 19: Feature call for AST- Adjustable Start-up Time 37
Table 20: Feature call for ZTV- Zero to Ten Volt 38
Table 21: DALI 102 Adressing 39
Table 22: DALI 102 and 207 Configuration 40
Table 23: Feature call for AmpDim 42
Table 24: Feature call for MTP – Module Temperature Protection 43
Table 25: Feature call for Corridor 45
Table 26: Feature call for Touch and Dim 46
Table 27: Feature call for Touch and Dim 47
Table 27: Feature call for DALI 253M 48
Table 28: Feature call for OWP – OEM Write Protection 49
Table 29: Feature call for FlexTune 50
Table 30: Feature call for Multi Config Mode 52

Introduction

MultiOne Cloud can decrease time to market and optimize flexibility by connecting your automated planning/resource system. This new tool generates feature files compatible with MultiOne Engineering and MultiOne Workflow that is used in production. It also supports the latest configurable drivers sold from 2022 onwards. MultiOne Cloud is built on the open standard of REST API standardization in JavaScript Object Notation (JSON) format which makes it accessible via many different platforms. MultiOne Cloud is generally used by OEMs who have been using an automation system or planning to implement it. There are also other systems for e.g., excel etc. to which REST API can be connected.



The figure above represents the process flow. The sales personnel/technician enters the sales order, and the details are then converted to driver specific information. Based on this information in the automation system, the system makes specific API calls to connect with MultiOne Cloud as described in this document. In this way the feature files can be generated automatically and can be used in the production for configuring drivers.

APIs, short for Application Programming Interfaces, are software-to-software interfaces. Meaning, they allow different applications to talk to each other and exchange information or functionality. This allows businesses to access another business's data, piece of code, software, or services to extend the functionality of their own products — all while saving time and money (HubSpot, 2022).

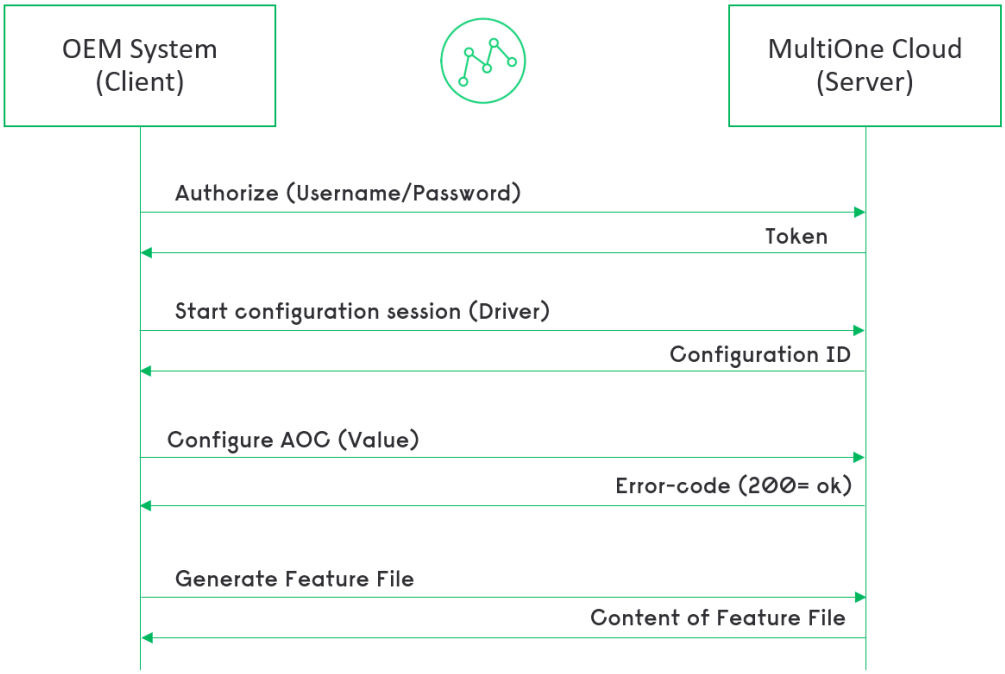
An API call is the process of a client application submitting a request to an API and that API retrieving the requested data from the external server or program and delivering it back to the client.

Request for the API calls for MultiOne Cloud are done in the following ways:

- **POST**: To authorize, to initiate configuration and configuration of features
- **GET**: To get the software version, list of device features and content of the feature file from MultiOne Cloud

There are many API testing platforms and one of the most common applications is Postman. This tool can be used for making API calls and in this document, the examples are provided using this tool.

API sequence for MultiOne



As shown in the above figure, the following steps are needed to create a feature file:

- Authentication (of the registered user) – MultiOne Cloud checks if the user is authorized to communicate with the server. When that is passed, a token is returned.
- Start a configuration session – For every driver configuration, a configuration ID is returned. Multiple sessions are allowed with same authentication.
- Set the feature configuration(s) – Configure features for the driver. Depending on the set values, error codes are returned
- Generate the feature file – After the configuration of features, via 'Get' feature file, you can get the content of the feature file. This content can be stored in your system for further use. Note – After the content has been stored, you cannot make changes to the content. You need to start a new session.

All the supporting documents for MultiOne Cloud can be found on Global MyLighting portal.
Link : [MyLighting: Log On]

API calls for MultiOne Cloud

Functions		API call
Authentication	The user account is authenticated, and a token is returned	{{url}}/auth/api/v1/authenticate
Start configuration	A configuration ID is generated for the entered 12NC	{{url}}/config/api/v1/configuration
Configure features	The features can be configured	{{url}}/config/api/v1/configuration/{{configurationId}}/features
Feature File	Feature file generated and stored by OEM.	{{url}}/feature/api/v1/configuration/{{configurationId}}/feature-file
Device Configuration File	Device configuration file generated and stored by OEM.	{{url}}/feature/api/v1/configuration/{{configurationId}}/device-configuration-file

Table 1 : Main API Calls

Functions		API Calls
Device Features	To get the list of device feature of the 12NC provided	{{url}}/search/api/v1/device/{{device12nc}}/features
Device Features Metadata	To get the details of device features of the 12NC provided	{{url}}/search/api/v1/device/{{device12nc}}/features/metadata
Midnight shift Calculator	To get the midnight shift value	{{url}}/config/api/v1/calculation/api/v1/midnight-shift
Retrieve driver production info	To get the 12nc, device name and production related information.	{{url}}/search/api/v1/data/devices/{{deviceId}}/production-information
Search API call	Present EDIT information	{{url}}/search/api/v1/devices/

Table 2 : Additional API Calls

To select production environment it is advised to define the url for MultiOne Cloud as a variable:

Production:
{{url}} – https://api.multione.signify.com/

Tip – When you make an API call to retrieve the list of device features, you can use the syntax from the response for easy implementation.

Authentication

With your account credentials you can start the authentication process.

```
{
  "password": "{{secret}}",
  "userName": "{{userId}}"
}
```

EXAMPLE

```
{
  "password": "MultiOneCloud",
  "userName": "multione"
}
```

Configuration

```
{
    "device12nc":“String”,
}
```

EXAMPLE

```
{
   "device12nc": "929002102906",
}
```

Supported features

- AOC (Adjustable Output Current)
- CLO (Constant Light Output)
- CLO Lite
- Dimming Interface selection
- ALO (Adjustable Light Output)
- ALO min
- Minimum dim level (different from 0-10V/1-10V dim level)
- DC emergency
- Luminaire info (DALI: part 251)
- DynaDimmer
- Dynadimmer 10 Scenes
- DynaDimmer 10 Scenes with CCT
- DTL (Driver temperature limit)
- LineSwitch
- SR/DALI power supply (DALI: part 250)
- 0-10V/1-10V Interface
- AST (Adjustable Startup Time)
- Midnight shift calculator
- MTP (Module Temperature Protection)
- AmpDim
- Corridor Mode
- Touch and Dim
- Coded Mains
- DALI 253M
- OWP (OEM Write Protection)
- FlexTune
- Multi Config Mode

Feature configuration

Below you can find some examples for configuration of features. See the design in guides of the driver for the available features. The syntax of the features is slightly changed (in accordance with REST definition). Please find below examples of the code. All the examples syntax is taken from Postman.

The features have in general the following syntax.

```
{
  "featureName": "String",
  "featureParameters": [
    {
       "Featureparmeter1": Boolean,
       " Featureparmeter2": Integer,
       " Featureparmeter3": “String”
    }
  ]
}
```

AOC – Adjustable Output Current

This feature enables the configuration of the maximum LED output current.

featureName	featureParameter	Type	Remark
AOC		String	Case Insensitive
	AocEnabled	Boolean	true/false
	AocCurrent	Integer	Value in mA Driver Specific

Table 3 : Feature call for AOC – Adjustable Output Current

EXAMPLE

```
[
  {
    "featureName": "AOC",
    "featureParameters": [
      {
        "AocEnabled": true,
        "AocCurrent": 300
      }
    ]
  }
]
```

AOC MultiChannel – Adjustable Output Current Multi Channel

This feature enables the configuration of the maximum LED output current.

featureName	featureParameter	Type	Remark
AOCMultiChannel		String	Case Insensitive
	NrOfChannels	Integer	1-4
	Channel1...4	Integer	Value in mA Driver Specific

Table 4 : Feature call for AOC – Adjustable Output Current

EXAMPLE


```
[
 {
  "featureName": "AOCMultiChannel",
  "featureParameters": [
   {
    "NrOfChannels": 4,
    "Channel1": 1600,
    "Channel2": 400,
    "Channel3": 400,
    "Channel4": 400
   }
  ]
 }
]
```

ALO (Adjustable Light Output)

This feature enables the configuration of the light level as a percentage of the configured LED current (AOC).

featureName	featureParameter	Type	Remark
ALO		String	Case Insensitive
	AloEnabled	Boolean	true/false
	AloPercentage	Integer	Value in %
	AloMinLevel	Integer	Value in % Driver Specific

Table 5: Feature call for ALO (Adjustable Light Output)

EXAMPLE

```
[
 {
  "featureName": "ALO",
  "featureParameters": [
   {
    "AloEnabled": true,
    "AloPercentage": 80,
    "AloMinLevel": 10
   }
  ]
 }
]
```

CLO (Constant Light Output)

This feature enables the compensation of the lumen depreciation to maintain a constant light level over time.

featureName	FeatureParameter	Type	Remark
CLO		String	Case Insensitive
	CloEnabled	Boolean	true/false
	CloPointx	X = Integer	Min 1 and Max 16 points
	Time	Integer	0 – 120 value in *1000 hours
	Percentage	Integer	% Of current

Table 6: Feature call for CLO – Constant Light Output

EXAMPLE


```
[
  {
    "featureName": "CLO",
    "featureParameters": [
      {
        "CloEnabled": true,
        "CloPoint1": {
          "Time": 0,
          "Percentage": 80
        },
        "CloPoint2": {
          "Time": 50,
          "Percentage": 90
        },
        "CloPoint3": {
          "Time": 100,
          "Percentage": 100
        }
      }
    ]
  }
]
```

CLOlite (Constant Light Output Lite)

This feature enables the compensation of the lumen depreciation to maintain a constant light level over time.

featureName	FeatureParameter	Type	Remark
CLOlite		String	Case Insensitive
	CloLiteEnabled	Boolean	true/false
	CloLiteInitialPowerLevel	Integer	Value in %
	CloLiteMaximumWorkingHours	Integer	0 – 120 value in *1000 hours

Table 7: Feature call for CLOlite – Constant Light Output Lite

EXAMPLE


```
[
  {
    "featureName": "CLOLITE",
    "featureParameters": [
      {
        "CloLiteEnabled": true,
        "CloLiteInitialPowerLevel": 90,
        "CloLiteMaximumWorkingHours": 100
      }
    ]
  }
]
```

Coded Mains

This feature enables control of the light level via the mains leads, an external transmitter sends Coded Mains commands and the device will react according the configured values.

featureName	FeatureParameter	Type	Remark
CodedMains		String	Case Insensitive
	FadeTime	Integer	0 - 15
	FastFadeTime	Integer	0 – 27
	FallbackDetectionTime	Integer	1 – 255
	PowerOnLevel	Integer	0 – 100 %
	FallbackScene	Integer	0 – 100 %
	Scenex	Integer	x = 1 – 10 max 10 scenes are allowed 0 – 100 %

Table 8: Feature call for Coded Mains

EXAMPLE


```
[
 {
  "featureName": "CodedMains",
  "featureParameters": [
   {
    "FadeTime": 7,
    "FastFadeTime": 27,
    "FallbackDetectionTime": 5,
    "PowerOnLevel": 100,
    "FallbackScene": 100,
    "Scene1": 0,
    "Scene2": 1,
    "Scene3": 10,
    "Scene4": 35,
    "Scene5": 50,
    "Scene6": 60,
    "Scene7": 75,
    "Scene8": 80,
    "Scene9": 85,
    "Scene10": 90,
    "Scene11": 95,
    "Scene12": 100
   }
  ]
 }
]
```

Dimming Interface

This feature enables the selection of the dimming interface (Dynadimmer, Lineswitch, etc.). If however, a DALI interface is present and a command is received it will automatically switch over to DALI.

featureName	FeatureParameter	Type	Remark
DimmingInterface		String	Case Insensitive
	SelectedInterface	String	Case Sensitive Dynadimmer, AmpDim, LineSwitch, ZTV and Dali
	OverrideInterface	String	Only Dynadimmer as SelectedInterface can be overriden by: “LineSwitch” Case Senstive String LineSwitch, ZTV OverrideInterface can contain multiple interfaces

Table 9: Feature call for Dimming Interface

EXAMPLE

```
[
 {
  "featureName": "DimmingInterface",
  "featureParameters": [
   {
    "SelectedInterface": "Dynadimmer",
    "OverrideInterface": ["LineSwitch"]
   }
  ]
 }
]
```

Dynadimmer

This feature enables autonomous dimming based on time or on the switch-on/switch-off behavior of the grid.

featureName	FeatureParameter	Type	Remark
Dynadimmer		String	Case Insensitive
	DynadimmerEnabled	Boolean	Optional - only when driver does not have interface selection
	MidnightShift	Integer	the midnight shift (value in minutes (0)) depends on the geographical location which can be calculated by a separate API call (Refer Midnight Shift Calculator section below)
	TimeReference	Boolean	true/false
	Scenex	Integer	x = 1 - 5 max 5 scenes are allowed
	Hour	Integer	1 - 24
	Minute	Integer	1 - 60
	Dim Level	Integer	1 - 100 % Value in %
	FadeTime	Integer	0 - 255 seconds
	OverrideMode	Boolean	Optional; only when driver does not have interface selection

Table 10: Feature call for Dynadimmer

EXAMPLE


```
[
  {
  "featureName": "Dynadimmer",
  "featureParameters": [
   {
    "DynadimmerEnabled": false,
    "MidnightShift": 0,
    "TimeReference": false,
    "Scene1": {
     "DimLevel": 50,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 0
    },
    "Scene2": {
     "DimLevel": 100,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 6
    }
   }
  ]
 }

]
```

*Dynadimmer 10 scenes**

This feature enables autonomous dimming based on time or on the switch-on/switch-off behavior of the grid.

featureName	FeatureParameter	Type	Remark
Dynadimmer		String	Case Insensitive
	DynadimmerEnabled	Boolean	Optional - only when driver does not have interface selection
	MidnightShift	Integer	the midnight shift (value in minutes (0)) depends on the geographical location which can be calculated by a separate API call (Refer Midnight Shift Calculator section below)
	TimeReference	Boolean	true/false
	Scenex	Integer	x = 1 - 10 max 5 scenes are allowed
	Hour	Integer	1 - 24
	Minute	Integer	1 - 60
	Dim Level	Integer	1 - 100 % Value in %
	FadeTime	Integer	0 - 255 seconds
	OverrideMode	Boolean	Optional; only when driver does not have interface selection

Table 11: Feature call for Dynadimmer

** 10 scenes feature currently only applicable for 929003876980 Xi LP 200W 0.3-1.05A S1 WL I165*

EXAMPLE


```
[
  {
  "featureName": "Dynadimmer",
  "featureParameters": [
   {
    "DynadimmerEnabled": false,
    "MidnightShift": 0,
    "TimeReference": false,
    "Scene1": {
     "DimLevel": 50,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 0
    },
    "Scene2": {
     "DimLevel": 100,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 6
    }
    "Scene3": {
     "DimLevel": 75,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 24
    }
     "Scene4": {
     "DimLevel": 50,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 8
    },
    "Scene5": {
     "DimLevel": 55,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 9
    }
    "Scene6": {
     "DimLevel": 70,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 21
   }
    }
    "Scene7": {
     "DimLevel": 65,
     "FadeTime": 180,
     "Minute": 0,
     "Hour": 22
    }
```


signify

```
      "Scene8": {
       "DimLevel": 70,
       "FadeTime": 180,
       "Minute": 0,
       "Hour": 23
      },
      "Scene9": {
       "DimLevel": 80,
       "FadeTime": 180,
       "Minute": 0,
       "Hour": 1
      }
      "Scene10": {
       "DimLevel": 55,
       "FadeTime": 180,
       "Minute": 0,
       "Hour": 3

   ]
  }

]
```

Dynadimmer 10 scenes with CCT

This feature enables autonomous dimming based on time or on the switch-on/switch-off behavior of the grid.

featureName	FeatureParameter	Type	Remark
Dynadimmer		String	Case Insensitive
	DynadimmerEnabled	Boolean	Optional - only when driver does not have interface selection
	MidnightShift	Integer	the midnight shift (value in minutes (0)) depends on the geographical location which can be calculated by a separate API call (Refer Midnight Shift Calculator section below)
	TimeReference	Boolean	true/false
	Scenex	Integer	x = 1 - 10 max 10 scenes are allowed
	Hour	Integer	1 - 24
	Minute	Integer	1 - 60
	Dim Level	Integer	1 - 100 % Value in %
	FadeTime	Integer	0 - 255 seconds
	OverrideMode	Boolean	Optional; only when driver does not have interface selection
	Cct	Integer	CCT Value in Kelvin. Range 1500-9999

Table 12: Feature call for Dynadimmer

EXAMPLE


```
{
"featureName": "Dynadimmer",
"featureParameters": [
  {
    "DynadimmerEnabled": false,
    "MidnightShift": 0,
    "TimeReference": false,
    "Scene1": {
      "DimLevel": 50,
      "FadeTime": 180,
      "Minute": 0,
      "Cct": 65535,
      "Hour": 0
    },
    "Scene2": {
      "DimLevel": 100,
      "FadeTime": 180,
      "Minute": 0,
      "Cct": 65535,
      "Hour": 6
    }
  }
]
```

Midnight Shift Calculator

The midnight shift depends on the geographical location. Enter the feature parameters to calculate the midnight shift in minutes. You can use the response value as input to configure Dynadimmer (Refer Dynadimmer section).

featureName	featureParameter	Type	Remark
	latitude	Integer	Degrees cannot be negative
	degrees	Integer	0 – 89
	minutes	Integer	0-60
	seconds	Integer	0-60
	hemisphere	String	North/South/East and West
	longitude	Integer	Degrees cannot be negative
	degrees	Integer	0 – 179
	minutes	Integer	0-60
	seconds	Integer	0-60
	hemisphere	String	North/South/East and West
	timeZone	Integer	GMT+1 enter 1, for GMT-1 enter -1, and so on.
	daylightSaving	Boolean	true/false

Table 13: Feature call for Midnight Shift Calculator

EXAMPLE


```
{
   "latitude": {
      "degrees": 1,
      "minutes": 0,
      "seconds": 0,
      "hemisphere": "North"
   },
   "longitude": {
      "degrees": 1,
      "minutes": 0,
      "seconds": 0,
      "hemisphere": "West"
   },
   "timeZone": 0,
   "daylightSaving": false
}
```


DC Emergency

This feature enables switch over to a lower light level when the grid has switched over from AC to DC.

featureName	FeatureParameter	Type	Remark
DcEmergency		String	Case Insensitive
	DcEmergencyLevel	Integer	157 - 254 85 ~1%, 25 ~100%
	DcEmergencyEnabled	Boolean	true/false
	DcEmergencyDimmingEnabled	Boolean	(MO: "Allow dimming")

Table 14: Feature call for DC Emergency

EXAMPLE


```
[
  {
     "featureName": "DcEmergency",
     "featureParameters": [
        {
           "DcEmergencyLevel": 157,
           "DcEmergencyEnabled": true,
           "DcEmergencyDimmingEnabled": true
        }
     ]
  }
]
```

Minimum Dim Level

This feature defines the minimum dim level.

featureName	FeatureParameter	Type	Remark
MinDimLevel		String	Case Insensitive
	Enabled	Boolean	true/false
	MinDimLevel	Integer	Value in %

Table 15: Feature call for Minimum Dim Level

Example

```
[
  {
    "featureName": "MinDimLevel",
    "featureParameters": [
      {
        "Enabled": true,
        "MinDimLevel": 10
      }
    ]
  }
]
```

LineSwitch

This feature enables the selection of a pre-configured light level via an extra pilot line.

featureName	FeatureParameter	Type	Remark
LineSwitch		String	Case Insensitive
	HighVLevel	Integer	0 - 100 Percentage
	LowVlevel	Integer	0 - 100 Percentage
	FadeUpTime	Integer	0 - 90 Seconds
	SwitchOffDelayTime	Integer	0 - 1800 Seconds
	FadeDownTime		
	LineswitchEnabled	Boolean	Optional. When for DimmingInterface: Lineswitch is selected: LineswitchEnabled = true
	2ndInativeDimLevel	Integer	0 - 100 Percentage
	2ndInactiveLevelDelayTime	Integer	0 - 1440 minutes
	2ndInativeLevelEnabled	Boolean	true/false

Table 16: Feature call for Lineswitch

EXAMPLE


```
[
  {
    "featureName": "LineSwitch",
    "featureParameters": [
      {
        "HighVLevel": 100,
        "LowVLevel": 50,
        "FadeUpTime": 0,
        "SwitchOffDelayTime": 0,
        "FadeDownTime": 0,
        "2ndInactiveDimLevel": 0,
        "2ndInactiveLevelDelayTime": 0,
        "2ndInactiveLevelEnabled": true
      }
    ]
  }
]
```

DALI Power Supply Unit (DALIPSU)

With this feature the power for an external device can be switched on or off.

featureName	FeatureParameter	Type	Remark
DALIPSU		String	Case Insensitive
	Enabled	Boolean	true/false
	HbEnable	Boolean	Optional, driver specific true/false
	LastGaspEnable	Boolean	Optional, driver specific true/false

Table 17: Feature call for DALI Power Supply Unit

EXAMPLE

```
[
  {
    "featureName": "DALIPSU",
    "featureParameters": [
      {
        "Enabled": true,
        "HbEnable": true,
        "LastGaspEnable": false
      }
    ]
  }
]
```

DTL (Driver Temperature Limit)

This feature protects the device and indirect the module from overheating.

featureName	FeatureParameter	Type	Remark
Dtl		String	Case Insensitive
	DtlOutputPercentage	Integer	0-100
	DtlDimStart	Integer	Driver Specific
	DtlDimStop	Integer	Optional Driver Specific
	DtlLampOff	Integer	Optional Driver Specific
	DtlEnabled	Boolean	Optional true/false

Table 18: Feature call for Driver Temperature Limit

EXAMPLE

```
[
  {
    "featureName": "Dtl",
    "featureParameters": [
      {
        "DtlEnabled": true,
        "DtlDimStart": 70,
        "DtlDimStop": 90,
        "DtlLampOff": 98,
        "DtlOutputPercentage": 60

      }
    ]
  }
]
```

Luminaire Info format (DALI part 251)

This feature enables the configuration of luminaire fixture information (according DALI part 251/ANSI C137.4). This information can be used by a control or traceability system to inform the users.

featureName	FeatureParameter	Type	Remark
LumInfoFormat		String	Case Insensitive
	OemGtin	String	13 digits
	OemIdentification	String	
	UseDeviceUid	Boolean	true/false
	ContentIdentifier	Integer	0, 3, 4 & 5
	LuminaireManufactureYear	Integer	1 - 99
	LuminaireManufactureWeek	Integer	1 - 99
	LuminaireManufactureYear	Integer	1 - 99
	Use DynamicDateIndicator	Boolean	true/false
	NominalInputPower	Integer	0 - 65535
	PowerMinDim	Integer	0 - 65535
	MinAcMainsVoltage	Integer	0 - 65535
	MaxAcMainsVoltage	Integer	0 - 65535
	NominalLightOutput	Integer	0 - 16777215
	CRI	Integer	0 - 255
	CCT	Integer	0 - 65535
	DistributionType	Integer	0 - 255
	LuminaireColor	String	This is a String of hexadecimal ascii values. Only the printable ascii character are allowed.
	LuminaireIdentification	String	This is a String of hexadecimal ascii values. Only the printable ascii character are allowed.
	AdditionalInfo	String	This is a String of hexadecimal ascii values. Only the printable ascii character are allowed.

Table 19: Feature call for Luminaire Info format (DALI part 251)

Example


```
[
  {
    "featureName": "LumInfoFormat",
    "featureParameters": [
      {
        "ContentIdentifier": 3,
        "OemGtin": "9999999999999",
        "OemIdentification": "",
        "UseDeviceUid": true,
        "LuminaireManufactureYear": 22,
        "LuminaireManufactureWeek": 40,
        "UseDynamicDateIndicator": true,
        "NominalInputPower": 0,
        "PowerMinDim": 0,
        "MinAcMainsVoltage": 194,
        "MaxAcMainsVoltage": 264,
        "NominalLightOutput": 3000,
        "CRI": 90,
        "CCT": 2700,
        "DistributionType": 2,
        "LuminaireColor": "",
        "LuminaireIdentification": "",
        "AdditionalInfo": ""
      }
    ]
  }
]
```

AST (Adjustable Start-up Time)

This feature enables the adjustment of the ramp-up time of the LED module at switching ON.

featureName	FeatureParameter	Type	Remark
AST		String	Case Insensitive
	AdjustableStartupTime	Integer	0 - 1000

Table 20: Feature call for AST- Adjustable Start-up Time

EXAMPLE

```
[
  {
    "featureName": "AST",
    "featureParameters": [
      {
        "AdjustableStartupTime": 700
      }
    ]
  }
]
```

ZTV (Zero to Ten Volt)

This feature enables control of the light level through analog Zero To Ten Volt input. 0-10V is used in North-America, in Europe this interface is standardized as 1-10V.

featureName	FeatureParameter	Type	Remark
ZTV		String	Case Insensitive
	ZtvMinDimLevel	Integer	1 - 100
	ZtvCurveSelection	Integer	Driver specific
	ZtvEnabled	Boolean	true/false
	ZtvCurveSelection	String	1-10V curved 0-10V linear 1-9V linear 0-10V logarithmic The supported curves are product specific
	ZtvDimToOffEnabled	Boolean	true/false
	FadeOnTime	Integer	0-2 Seconds, multiple of 0.5s
	FadeOffTime	Integer	0-2 Seconds, multiple of 0.5s
	DimToOffThreshold	Integer	0.5V/1V The support of dim to off is product specific
	ZtvMinDimCurrent	Integer	Physical min AOC – min AOC The Support of minimum dim current is product specific.

Table 21: Feature call for ZTV- Zero to Ten Volt

EXAMPLE

```
[
  {
    "featureName": "Ztv",
    "featureParameters": [
      {
        "ZtvMinDimLevel": 10,
        "ZtvCurveSelection": “1-10V Curved”,
        "ZtvEnabled": true,
        "ZtvDimToOffEnabled": false,
        "FadeOnTime": 0,
        "FadeOffTime": 1,
        "DimToOffThreshold": “0.5V”,
      }
    ]
  }
]
```

DALI 102 Addressing

This feature enables OEM users to wirelessly configure DALI 102 parameters related to addressing parameters, using NFC technology, providing more flexibility and simplicity for configuring drivers.

featureName	FeatureParameter	Type	Remark
DALIAddressing102		String	Case Insensitive
	ShortAddress	Integer	0 - 63
	Groups	String	Group0 – Group15

Table 22: DALI 102 Adressing

EXAMPLE

```
[
  {
    "featureName": "DALIAdressing102",
    "featureParameters": [
      {
        "ShortAddress": 255,
        "Groups": ["Group0", "Group4"],
      }
    ]
  }
]
```

DALI 102 and 207 Configuration

This feature enables OEM users to wirelessly configure DALI 102 and 207 parameters using NFC technology, providing more flexibility and simplicity for configuring drivers.

featureName	FeatureParameter	Type	Remark
daliconfiguration102and207		String	Case Insensitive
	PowerOnLevel	Integer	0 - 254
	SystemFailureLevel	Integer	0 - 254
	MinLevel	Integer	0 - 85
	MaxLevel	Integer	85 - 254
	FadeRate	Integer	2.8 – 358 seconds
	FadeTime	Integer	0 – 90.5 seconds
	ExtendedFadeTime	Integer	0 – 960 seconds
	FastFadeTime	Integer	0 – 675 miliseconds
	OperatingMode	Integer	0 - 254
	Scene0, Scene1, … , Scene15	Integer	0 - 254
	DimmingCurve	String	Logaritmic or Linear

Table 23: DALI 102 and 207 Configuration

EXAMPLE


```
{
    "featureName": "daliconfiguration102and207",
    "featureParameters": [
      {
        "PowerOnLevel": 254,
        "SystemFailureLevel": 254,
        "MinLevel": 85,
        "MaxLevel": 254,
        "FadeRate": 44.7,
        "ExtendedFadeTime": 0,
        "OperatingMode": 128,
        "Scene0": 255,
        "Scene1": 255,
        "Scene2": 255,
        "Scene3": 255,
        "Scene4": 255,
        "Scene5": 255,
        "Scene6": 255,
        "Scene7": 255,
        "Scene8": 255,
        "Scene9": 255,
        "Scene10": 255,
        "Scene11": 255,
        "Scene12": 255,
        "Scene13": 255,
        "Scene14": 255,
        "Scene15": 255,
        "DimmingCurve": "Logarithmic"
      }
    ]
  },
```


signify

Amp Dim

This feature enables control of the light level by regulating the mains voltage.

featureName	featureParameter	Type	Remark
AmpDim		String	Case Insensitive
	VmainsStart	Integer	120V – 277V
	VmainsStop	Integer	70V – 230V
	StartDimLevel	Integer	30% - 100%
	StopDimLevel	Integer	10% - 60%

Table 24: Feature call for AmpDim

EXAMPLE

```
[
  {
    "featureName": "AmpDim",
    "featureParameters": [
      {
        "VmainsStart": 230,
        "VmainsStop": 195,
        "StartDimLevel": 100,
        "StopDimLevel": 60
      }
    ]
  }
]
```

MTP (Module Temperature Protection)

This feature enables protection of the module from overheating.

featureName	featureParameter	Type	Remark
MTP		String	Case Insensitive
	SelectedNtc		Case Sensitive Ntc1, Ntc2, PhilipsLedLightEngines, and Custom
	NtcBeta	Integer	Beta value between 25°C and 85°C (Beta 25/85). 2000 – 5000 Ohm
	NtcResistance	Integer	1000 – 50000 Ohm R_NTC at 25°
	DimLevel	Integer	Min dim level - 100%
	DimStart	Integer	Driver Specific
	DimStop	Integer	Driver Specific
	Enabled	Boolean	true/false

Table 25: Feature call for MTP – Module Temperature Protection

EXAMPLE

For NTC1 (or NTC2):

```
[
 {
  "featureName": "MTP",
  "featureParameters": [
   {
    "SelectedNtc": "Ntc1",
    "DimLevel": 10,
    "DimStart": 70,
    "DimStop": 80,
    "Enabled": true
   }
  ]
 }
]
```

For Philips LED Light Engines:


```
[
 {
  "featureName": "MTP",
  "featureParameters": [
   {
    "SelectedNtc": "PhilipsLedLightEngines",
    "DimLevel": 10,
    "Enabled": true
   }
  ]
 }
]
```


For Custom:


```
[
 {
  "featureName": "MTP",
  "featureParameters": [
   {
    "SelectedNtc": "Custom",
    "NtcBeta": 3000,
    "NtcResistance": 25000,
    "DimLevel": 10,
    "DimStart": 70,
    "DimStop":  80,
    "Enabled": true
   }
  ]
 }
]
```

Corridor

This feature is an extension to Touch and Dim. Both features are controlled using mains cycles presented on the Touch and Dim or DALI interface.

featureName	featureParameter	Type	Remark
Corridor		String	Case Insensitive
	NormalLevel	Integer	1-254
	BackgroundLevel	Integer	1-254
	FadeTimeIndex	Integer	0-15
	ProlongTime	Integer	0-2540
	ActivationTime	Integer	0-254
	DelayTime	Integer	0-2550
	CorridorEnabled	Boolean	true/false

Table 26: Feature call for Corridor

EXAMPLE

```
[
 {
  "featureName": "Corridor",
  "featureParameters": [
   {
    "NormalLevel": 254,
    "BackgroundLevel": 170,
    "FadeTimeIndex": 12,
    "ProlongTime": 1800,
    "ActivationTime": 55,
    "DelayTime": 20,
    "CorridorEnabled": true
   }
  ]
 }
]
```

Touch and Dim

This feature enables the configuration of a Touch and Dim controller, which is a simple dim input used to control the dimming and switching of lamps via a device.

featureName	featureParameter	Type	Remark
TouchDim		String	Case Insensitive
	PowerOnLevel	Integer	Driver Specific
	IgnoreToggleLimits	Integer	true/false
	TouchDimEnabled	Boolean	true/false

Table 27: Feature call for Touch and Dim

EXAMPLE

```
[
 {
  "featureName": "TouchDim",
  "featureParameters": [
   {
    "PowerOnLevel": 255,
    "IgnoreToggleLimits": false,
    "TouchDimEnabled": true
   }
  ]
 }
]
```

Touch and Dim with CCT (only for FlexTune Drivers)

This feature enables the configuration of a Touch and Dim controller, which is a simple dim input used to control the dimming and switching of lamps via a device.

featureName	featureParameter	Type	Remark
TouchDim		String	Case Insensitive
	TouchDimEnabled	String	True/False
	PowerOnLevel	Integer	Driver Specific
	IgnoreToggleLimits	Integer	true/false
	TouchDimTwEnabled	Boolean	true/false
	PowerOnCCT	Integer	1500 - 9999 kelvin, 65535 (MASK, use last known)

Table 28: Feature call for Touch and Dim

EXAMPLE

```
{
 "featureName": "TouchDim",
 "featureParameters": [
  {
   "TouchDimEnabled": false,
   "PowerOnLevel": 255,
   "IgnoreToggleLimits": false,
   "TouchDimTwEnabled": false,
   "PowerOnCCT": 65535
  }
 ]
```

DiiA 253

This feature enables the configuration of Luminaire maintenance data, according to DiiA 253 Maintenance.

featureName	featureParameter	Type	Remark
Diia253Maintenance		String	Case Insensitive
	RatedLife	Integer	kilohours
	ReferenceTemperature	Integer	Celsius
	RatedStarts	Integer	100 starts

Table 29: Feature call for DALI 253M

EXAMPLE

```
[
  {
    "featureName": "Diia253Maintenance",
    "featureParameters": [
      {
        "RatedLife": 255,
        "ReferenceTemperature": 195,
        "RatedStarts": 65535
      }
    ]
  }
]
```

OWP (OEM Write Protection)

This feature enables the protection of one or multiple features. It prevents installers or others to alter the values which could damage the luminaire.

featureName	featureParameter	Remark
Owp		Case insensitive
	OwpEnabled	true/false
		Features are case sensitive. Use the following Dynadimmer, CLO, CLOLite, DALIPSU, LumInfoFormat, DCEmergency, MTP, DTL, FlexTune, LineSwitch, Corridor, AmpDim, ZTV, TouchDim, AST, MinDimLevel, ALO, ALOMin, AOC, CodedMains
	OwpPassword	

Table 30: Feature call for OWP – OEM Write Protection

EXAMPLE

```
[
 {
  "featureName": "Owp",
  "featureParameters": [
   {
    "OwpEnabled": [
     "Aoc",
     " LumInfoFormat"
    ],
    "OwpPassword": [
     1,
     2,
     3,
     4
    ]
   }
  ]
 }
]
```

FlexTune

This features enables the control of the color of the luminaire.

featureName	featureParameter	Type	Remark
FlexTune		String	Case Insensitive
	CoolCct	number	Configuration of coolest CCT (highest K)
	CoolRatedLm	number	Rated Lumen of cool LED at rated current (0-65535)
	CoolRatedCurrent	number	Rated Current of cool LED (0-10000)
	WarmCct	number	Configuration of warmest CCT (lowest K)
	WarmRatedLm	number	Rated Lumen of warm LED at rated current (0-65535)
	WarmRatedCurrent	number	Rated Current of warm LED (0-10000)
	LightOutputOverCct	string	Selection via Flexible or Constant
	CoolAOC	number	Current configuration for a defined light color (300 – 1050)
	WarmAOC	number	Hidden, see ``FlexTune Application Note`` for more information, Current configuration for a defined light color (300 – 1050)
	CctOffset	number	Offset of color (in kelvin)

Table 31: Feature call for FlexTune

EXAMPLE

```
[
 {
     "featureName": "FlexTune",
     "featureParameters": [
       {
          "CoolCct": 7042,
          "CoolRatedLm": 2200,
          "CoolRatedCurrent": 308,
          "WarmCct": 2198,
          "WarmRatedLm": 2040,
          "WarmRatedCurrent": 308,
          "LightOutputOverCct": "Constant",
          "CoolAOC": 700,
          "CctOffset": 0
       }
     ],
```

Direct Sensor Control(DSC)

Direct Sensor Control is a feature which enables a driver to reactor to DALI occupancy- and illuminance events. The DALI occupancy- and illuminance events will trigger the configurated light behavior.

featureName	featureParameter	Type	Remark
DSC		String	Case Insensitive
	Enabled	String	true/false
	PresenceLevel	number	0 - 100% or Dynadimmer level (255)
	PresenceHoldTime	number	10 - 2540 seconds 2550 = no hold time
	NoPresenceLevel	number	0 - 100% or Dynadimmer level (255)
	NoPresenceFadeTime	number	0 - 255 seconds
	NoPresenceProlongTime	number	0 - 2540 seconds 2550 = no prolong time
	SunsetThreshold	number	1-1023 0 = Dynadimmer not started by sensor
	SunriseThreshold	number	Current configuration for a defined light color (300 – 1050)
	WarmAOC	number	1-1023 0 = Dynadimmer not stopped by sensor

EXAMPLE


```
[
  {
    "featureName": "DSC",
    "featureParameters": [
      {
        "Enabled": false,
        "PresenceLevel": 100,
        "PresenceFadeTime": 0,
        "PresenceHoldTime": 2550,
        "NoPresenceLevel": 10,
        "NoPresenceFadeTime": 0,
        "NoPresenceProlongTime": 2550,
        "SunsetThreshold": 0,
        "SunriseThreshold": 0
      }
    ]
  }
```

Multi Config Mode

This feature enables the selection of the configuration mode of multi config devices according DiiA specification.

featureName	featureParameter	Type	Remark
multiconfigmode		String	Case Insensitive
	ConfigurationMode	String	2x207 or 1x209 (Tc)

Table 32: Feature call for Multi Config Mode

EXAMPLE


```
{
    "featureName": "multiconfigmode",
    "featureParameters": [
      {
        "ConfigurationMode": "1x209 (Tc)"
      }
    ],
```

LSI

This feature enables control of light levels in occupancy sensing mode of operation.

featureName	featureParameter	Type	Remark
LSI		String	Case Insensitive
	Enabled	String	true/false
	Mode	String	"Occupancy sensor" "Photocell"
	ActiveDimLevel	Number	0.00 - 100.00%
	InactiveDimLevel	Number	0.00 - 100.00%
	Polarity	String	"Active when LSI input is low" "Active when LSI input is high"
	FadeOnTime	Number	0 - 90 seconds
	HoldOffDelay	Number	0 - 64800 seconds
	FadeOffTime	Number	0 - 90 seconds
	SwitchOffTime	Number	0 - 64800 seconds
	EnableOutputOff	String	true/false

Table 31 Feature call for LSI

EXAMPLE


```
[
 {
  "featureName": "LSI",
  "featureParameters": [
   {
    "Enabled": false,
    "Mode": "Occupancy sensor",
    "ActiveDimLevel": 100.0,
    "InactiveDimLevel": 50.0,
    "Polarity": "Active when LSI input is high",
    "FadeOnTime": 0,
    "HoldOffDelay": 0,
    "FadeOffTime": 0,
```


signify

```
    "SwitchOffDelay": 0,
    "EnableOutputOff": false
   }
  ]
 }
```


Over temperature protection (OTP)

Feature OTP consists of a configuration for protection the fixture combining the LED module temperature protection and driver temperature protection according to UL60730 or driver temperature protection only.

featureName	featureParameter	Type	Remark
OTP		String	Case Insensitive
	SelectedNTC	String	DLM Flex Compatible, Custom NTC
	DimStart	Number	0 to 65535 Ohm
	DimStop	Number	0 to 65535 Ohm
	DTP	Number	83-87 °C, 92-97 °C

Table 32 Feature call for OTP

EXAMPLE


```
[
 {
  "featureName": "OTP",
  "featureParameters": [
   {
    "SelectedNTC": "Custom NTC",
    "DimStart": 5000,
    "DimStop": 3003
   }
  ]
 }
```

Multiple feature configuration with a single API call

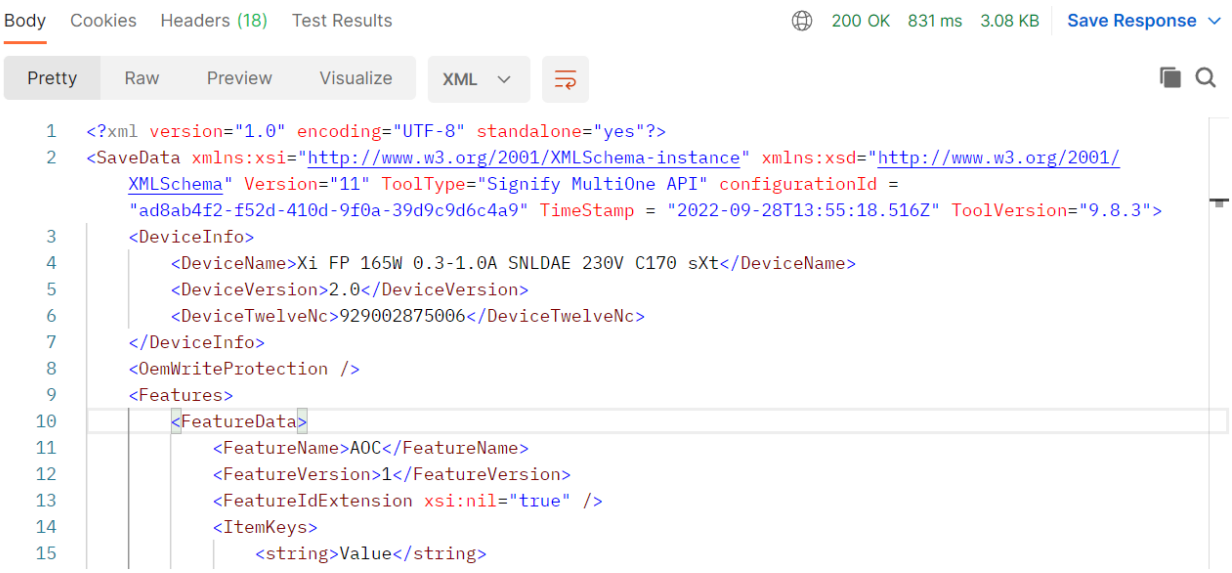
The below syntax is an example to configure multiple features in a single API call. Please note that the example does not include all the supported features.

EXAMPLE


```
[
  {
    "featureName": "AOC",
    "featureParameters": [
      {
        "AocEnabled": true,
        "AocCurrent": 500
      }
    ]
  },
  {
    "featureName": "CLO",
    "featureParameters": [
      {
        "CloEnabled": true,
        "CloPoint1": {
          "Time": 0,
          "Percentage": 80
        },
        "CloPoint2": {
          "Time": 100,
          "Percentage": 100
        }
      }
    ]
  },
  },
  {
    "featureName": "Dtl",
    "featureParameters": [
      {
        "DtlOutputPercentage": 50,
        "DtlDimStart": 10,
        "DtlDimStop": 88,
        "DtlLampOff": 93
      }
    ]
  }
]
```


signify

Feature file Generation



The above figure shows an example of generated content for a feature file from Postman platform. The content can then be saved by clicking ‘Save Response’. It must be saved in ‘xml’ file formats to be compatible with MultiOne Engineering and MultiOne Workflow.
Please note that Feature files are not stored in Signify Database.

References

HubSpot. (2022). Retrieved from HubSpot, Inc: https://blog.hubspot.com/website/api-calls