Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten insbesondere für den Fall der Patenterteilung oder GM-Eintragung

MANUAL

RESI®
...feel the spirit of a new generation

HOW TO CONTROL AND MONITOR EMERGENCY LAMPS

Great care has been taken in the creation of the text, illustrations and program examples in this manual. RESI Informatik & Automation GmbH company, the editors and publishers accept no responsibility for any inadvertent omission of entries or for typographical or other errors herein. Nor can they be held responsible or liable for consequences arising from any errors herein.

This manual may not be copied in part or whole in any form including electronic media without the written consent of RESI Informatik & Automation GmbH. Neither may it be transferred in any other language suitable for machines or data processing facilities. Also rights for reproduction through lecture, radio or television transmission are reserved.

This documentation as well as the corresponding software are subject to copyright law and protected by RESI company. All rights reserved.

© Copyright 2009-2014 RESI Informatik & Automation GmbH

1 History

Date	Editor	Description
19.12.14	DI HC Sigl	First version

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

2 Contents

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

3 General Information

This document describes the usage of a dali emergency system in combination with our controlling units RESI-Tx and our RESI-DALI-MODBUS converter.

To test the system by yourself, you have to have

- 1 pcs RESI-DALI-MODBUS converter
- 1 pcs RESI-DALI-PS
- 1 pcs RESI-T4 controller
- 1 pcs laptop or PC with WinXP, Win7 or Win8
- 1 pcs VOSSLOH & SCHWABE Dimmable electronic ballast part no ELXd 224.601
- 1 pcs ALVIT R2641 EMERGENCY FOR FLUORESCENT LAMP module with NiCd accumulator pack
- 1 pcs ALVIT MAT3 Autotest module with LED lamp
- 1 pcs power supply 24Vdc and 230Vac
- 1 pcs RESI-IDE-2006 V1.11 Software package
- 1 pcs RESI-VISXP software package

The correct wiring of the ALVIT system is not part of our documentation. We supply the SIBASIC sample sources as text files with the extension .SIB. You can open this files with any text editor (eg. Notepad++).

If you want to order a working demonstration package for this system, please contact us under our suport email: help@RESI.cc.

We do not provide confidential information from the DALI norm EN 62386. You can buy the following documents for more details of controlling DALI devices. For emergency lights EN 62386-202 is necessary.

Number IEC	Title
62386-101	System
62386-102	Control Gear
62386-201	Fluorescent Lamps (Device type 0)
62386-202	Self-contained emergency lighting (device type 1)
62386-203	discharge lamps (excluding fluorescent lamps) (device type 2)
62386-204	Low voltage halogen lamps (device type 3)
62386-205	Supply voltage controller for incandescent lamps (device type 4)
62386-206	Conversion from digital signal into d. c. voltage (device type 5)
62386-207	LED modules (device type 6)
62386-208	Switching function (device type 7)
62386-209	Colour control (device type 8)

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

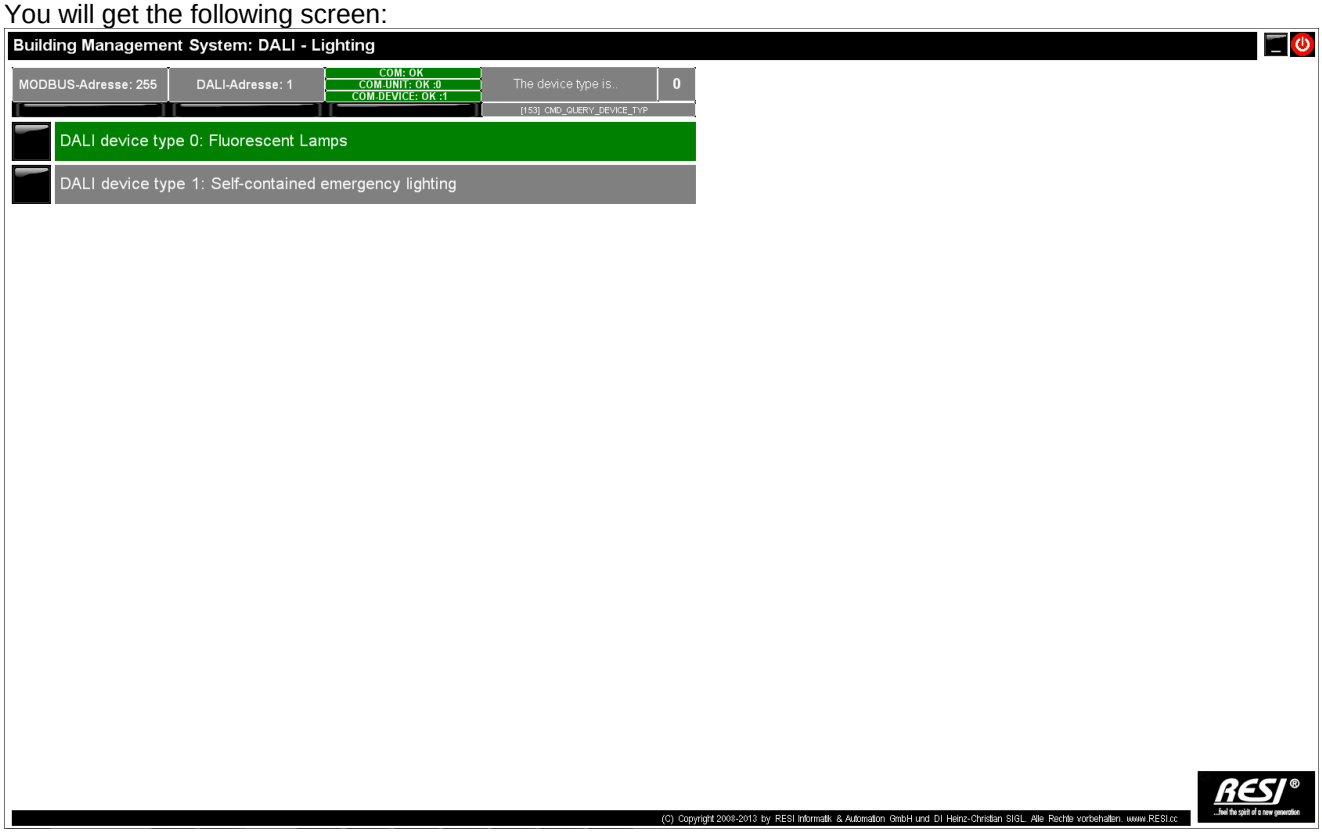
Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

4 System description

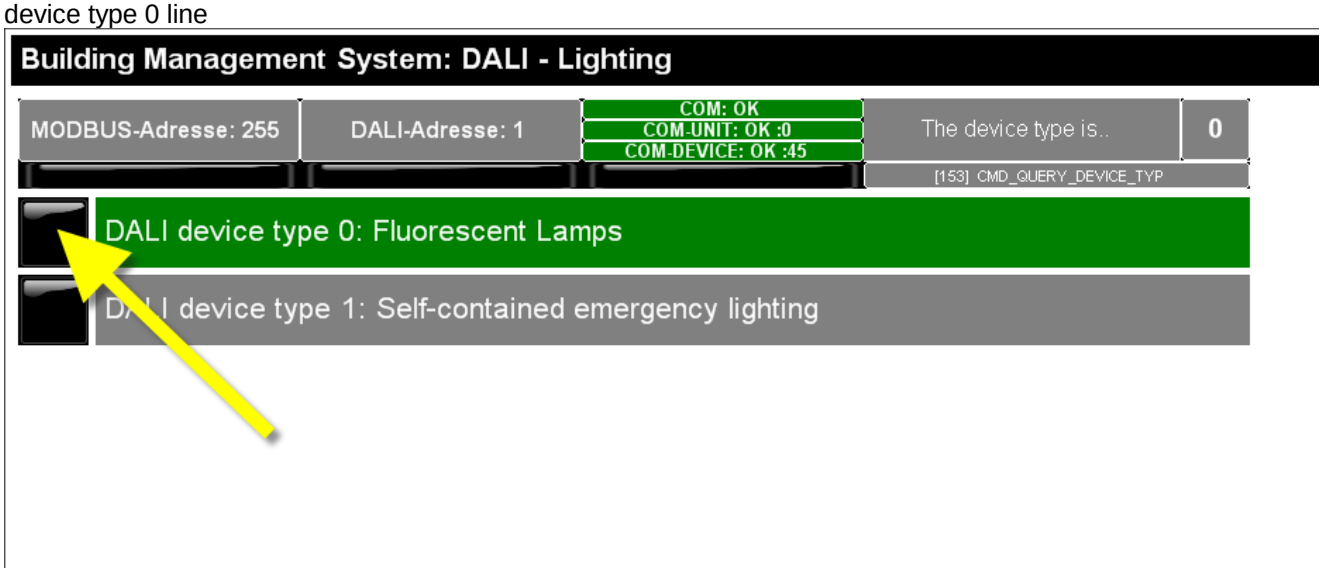
4.1 The visualization

We created a very simple demo visualization to show the basic functions of the system. After the correct setup of the software you can start the RESI visualization stored in the path DALI_EMERGENCY_LIGHTS\VISU. Start the program RESI_VISXP.exe in this folder.

You will get the following screen:



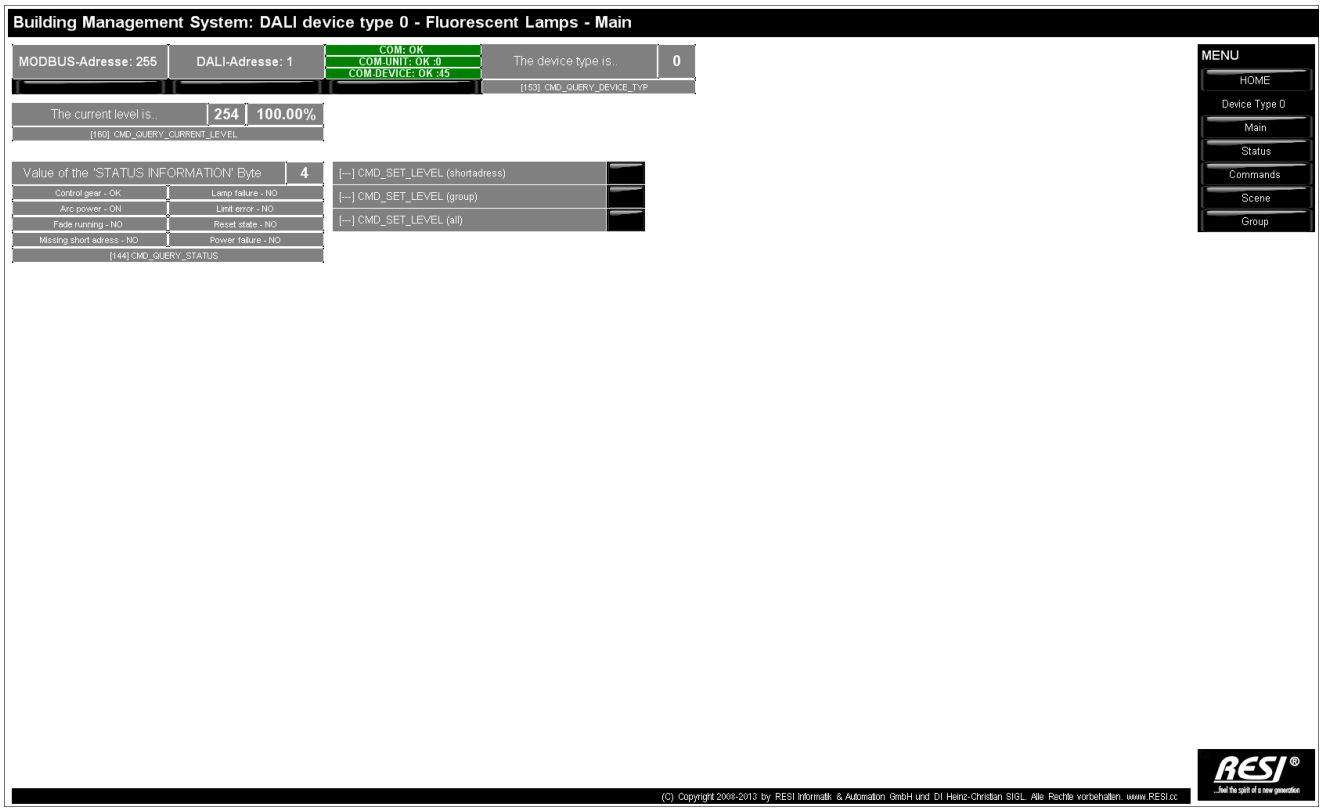
You will see, that you need two DALI devices, the one with device type 0 is the DALI fluorescent lamp itself, and the other one with the device type 1 is the emergency unit. Click on the black button on the left side of the DALI device type 0 line



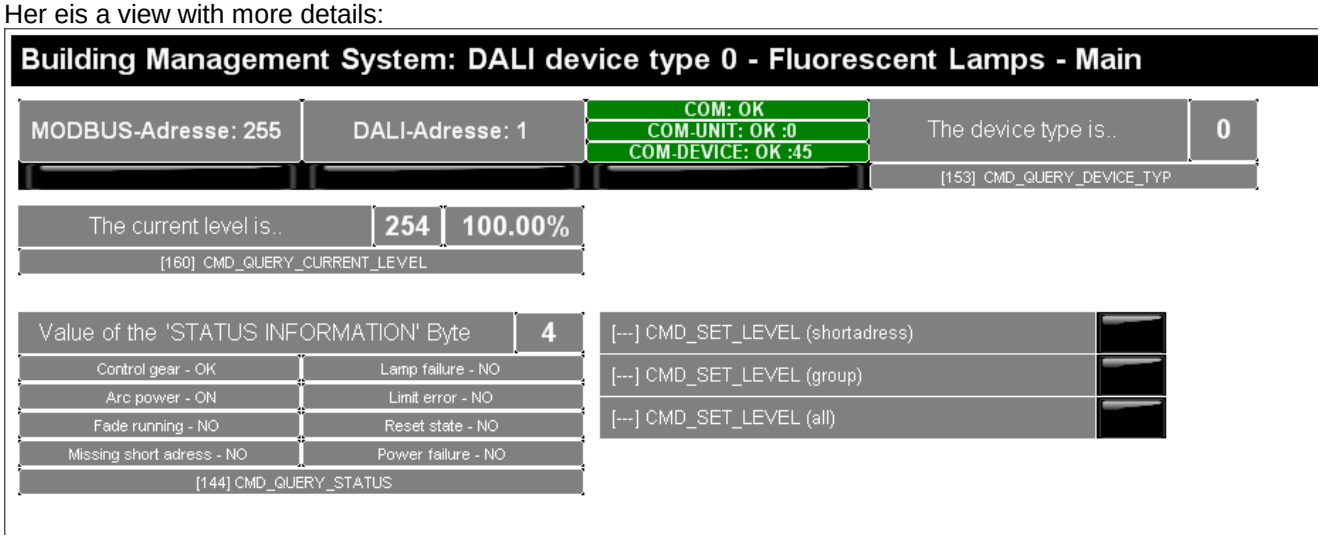
to activate the page for testing the fluorescent lamp. You will get the following page:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung



Her eis a view with more details:



The software displays the complete actual status of the DALI lamp. The current level of the lamp is displayed (254,100%). The STATUS INFORMAITON byte of the lamp is displayed and a panel with three commands (SET LEVEL for short address, SET LEVEL for groups, SET LEVEL for all) is available. In each section you will see the corresponding DALI command for requesting the information, eg. For The current level is, the command CMD_QUERY_CURRENT_LEVEL. Search in the documented source code for this command definition. You will find this line in the file DALI.sib

```
intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
```

When you search again, you will find a SIBASIC function for getting the current level:

```
//intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
function int DALI_QueryCurrentLevel(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_CURRENT_LEVEL)
endfunction
```

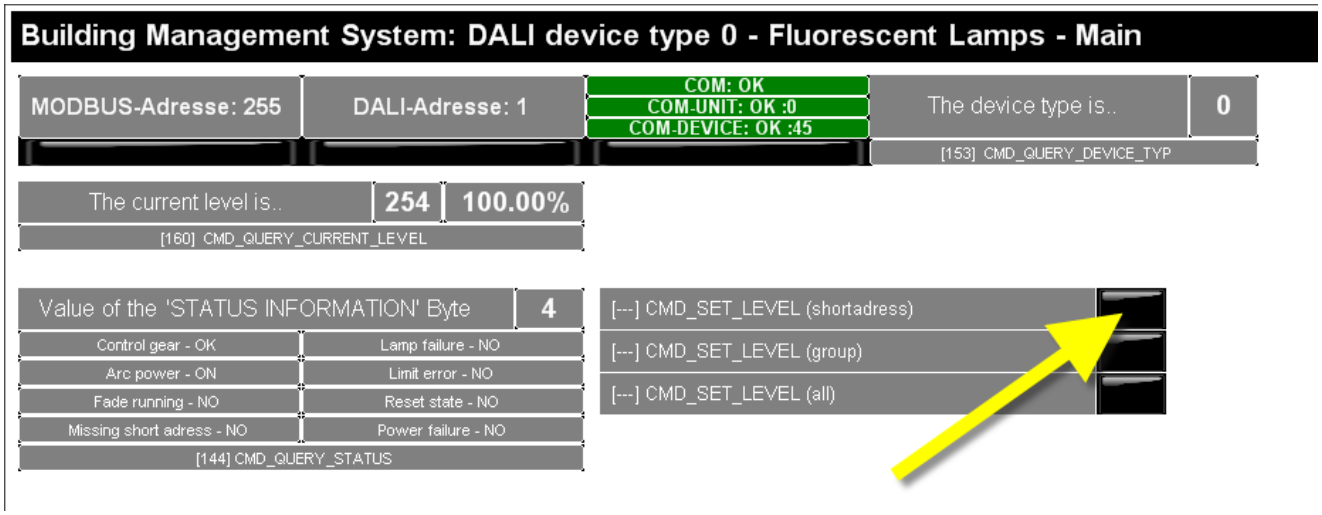
Now you see, that the software calls the universal function DALI_Query. So we search for this program segment:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
//-----------------------------------------------------------------------------------------------
function int DALI Query(num Handle,int Unit,int Adr,int Cmd)
  bit lComOk
  DALI_TempValue = Cmd + (Adr<<8)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB REG KRZ EXT CMD,DALI TempValue)then
    Me.Sleep(50)
    lComOk= ModbusMaster.ReadUnSignedInt16(Handle,Unit, MDB_REG_KRZ_EXT_CMD, DALI_QueryValue)
    if lComOk then
      if DALI QueryValue == 0x8000 then
         DALI_QueryOk = 0 //No answer from device
         DALI_R_COM_OK = false
         DALI R COM ERROR CNT DEVICE = DALI R COM ERROR CNT DEVICE + 1
      else
         DALI_QueryOk = 1 //Query was answered by device
         DALI_R_COM_OK = true
      endif
    else
      DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
      DALI_R_COM_OK = false
      DALI R COM ERROR CNT UNIT = DALI R COM ERROR CNT UNIT + 1
    endif
  else
    DALI CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI R COM OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  if DALI QueryOk != 1 then
    LOG_Message("DALI_Query  -  NOK  Adr:"+String.FormatInt(Adr,2)+"  Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  else
    LOG Message("DALI Query  -  OK  Adr:"+String.FormatInt(Adr,2)+"  Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  endif
  return DALI QueryOk
endfunction
```

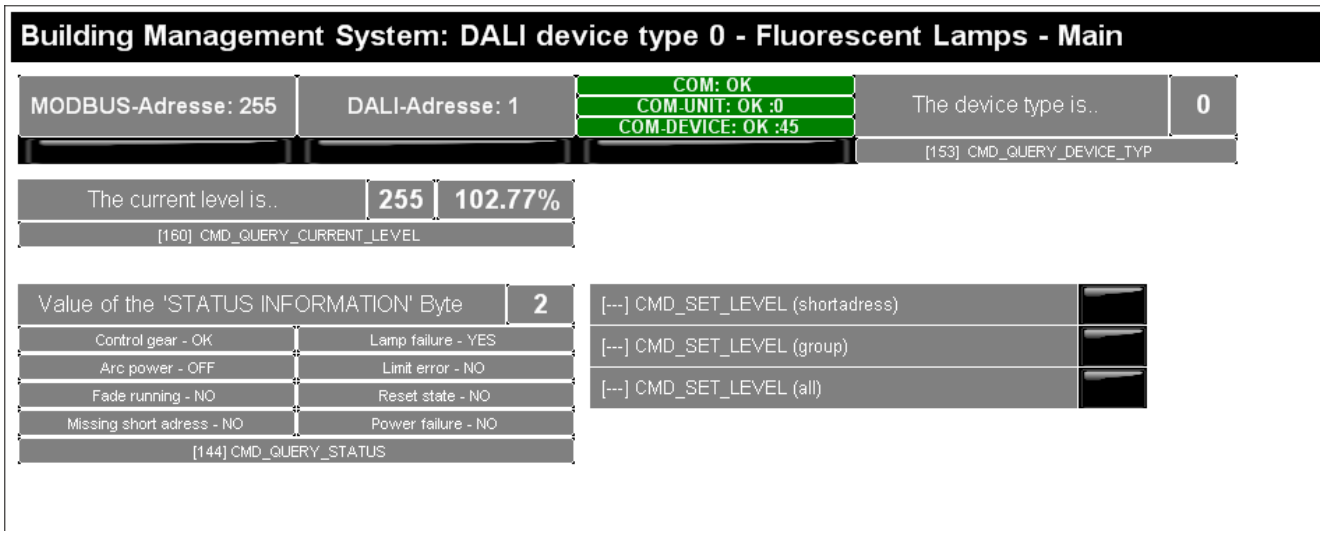
In this function our controller communicates with MODBUS/RTU master protocol with our RESI-DALI converter to send the correct command to the DALI lamp and to check the received information. This ist he same procedure for all types of commands. Refer to the source code for more details, what commands are supported by the DALI modules and how you can use them in your application. Our SIBASIC code is only for demonstration purpose.
Now click on the button for the command SET LEVEL:



The system will open a dialog with a question, if you really want to do that and then a dialog will be displayed where you can change the brightness level of the lamp between 0 and 254 as the DALI standard defines.
When you take a fluorescent lamp out of the lamp module, the status display will change to:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

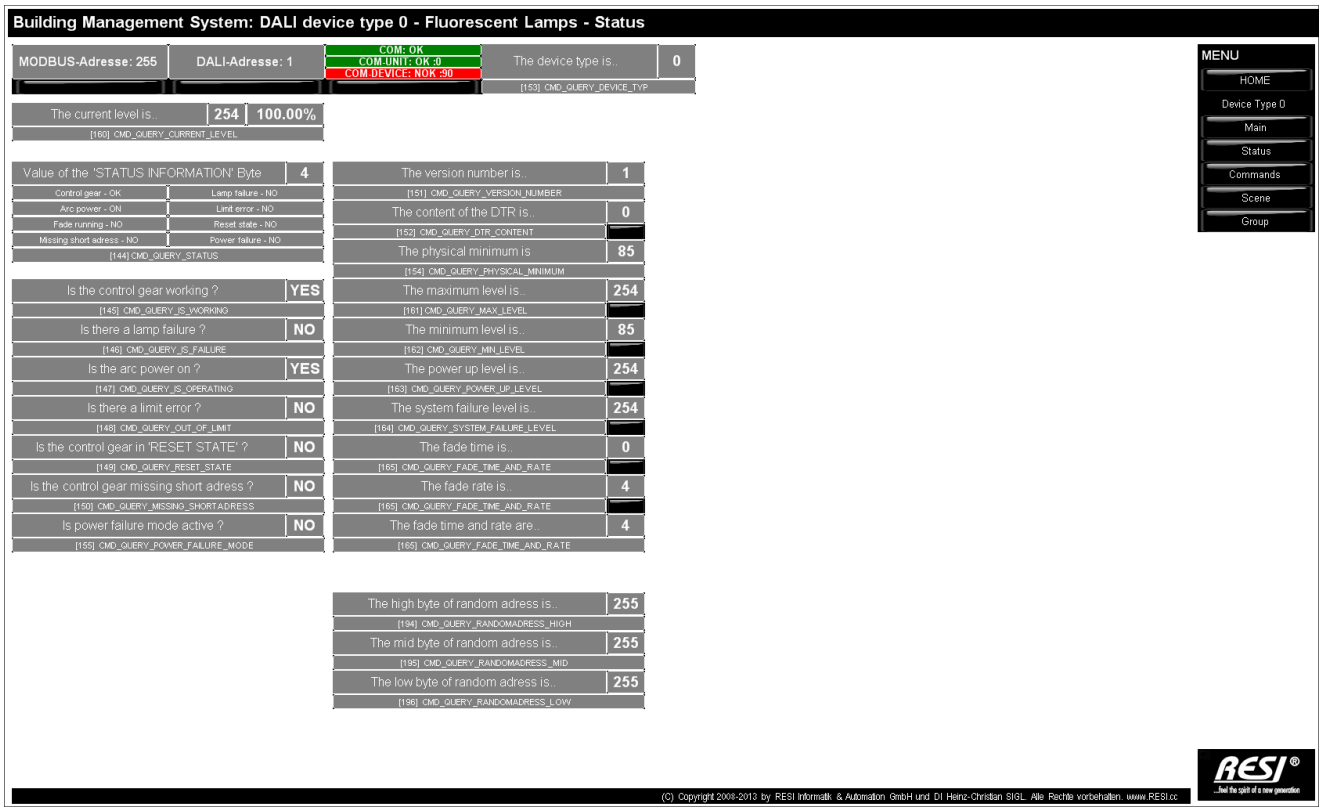


This page shows the basic DALI functions, you can do with a simple DALI lamp. On the right side you will find a menu:

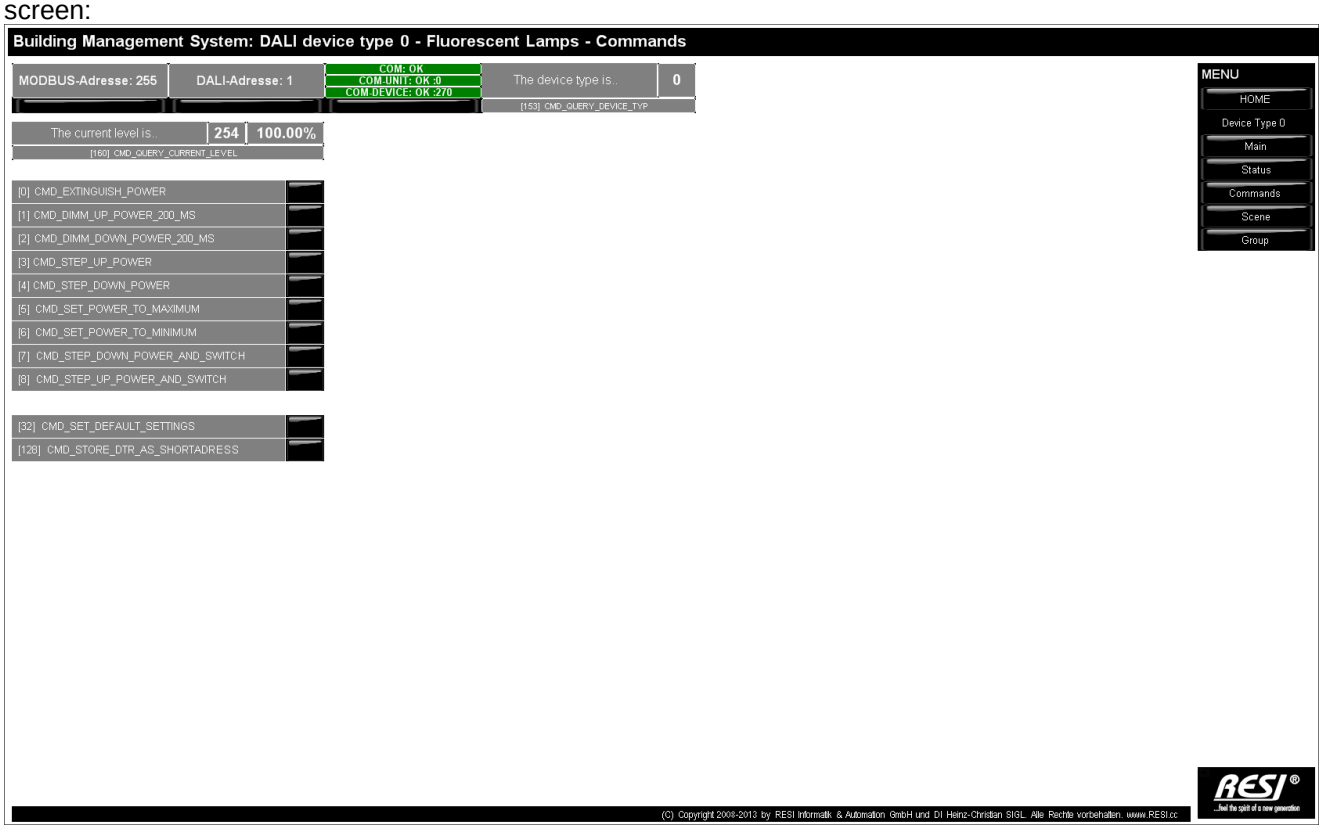
You are now on the MIAN page. Click on the page STATUS. You will get the following display:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung



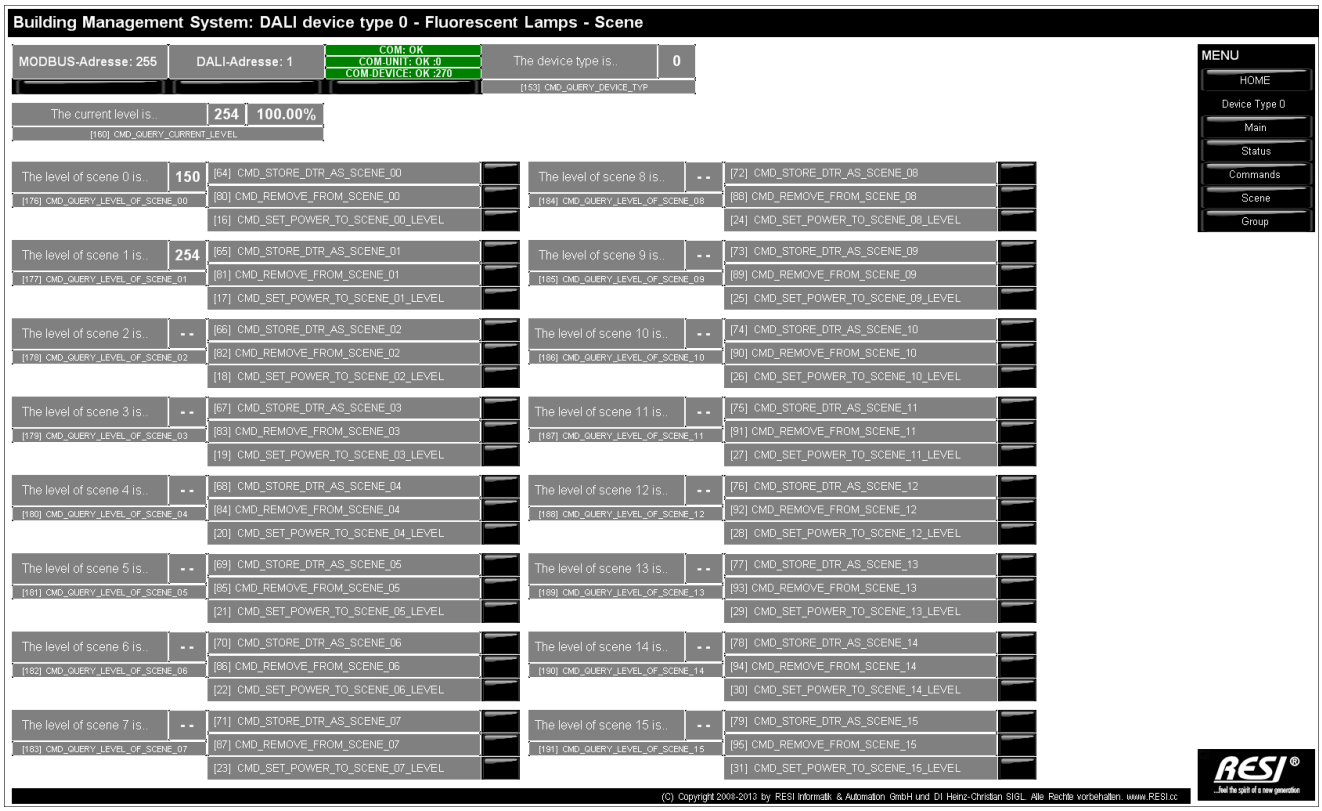
On this page you see all possible DALI status queries for a DALI fluorescent lamp module as defined in the standard. Whenever you find a black button, you can click on it and send a command (eg to change the minimum and maximum brightness level of the lamp). Click on the page COMMANDS, you will get the following screen:



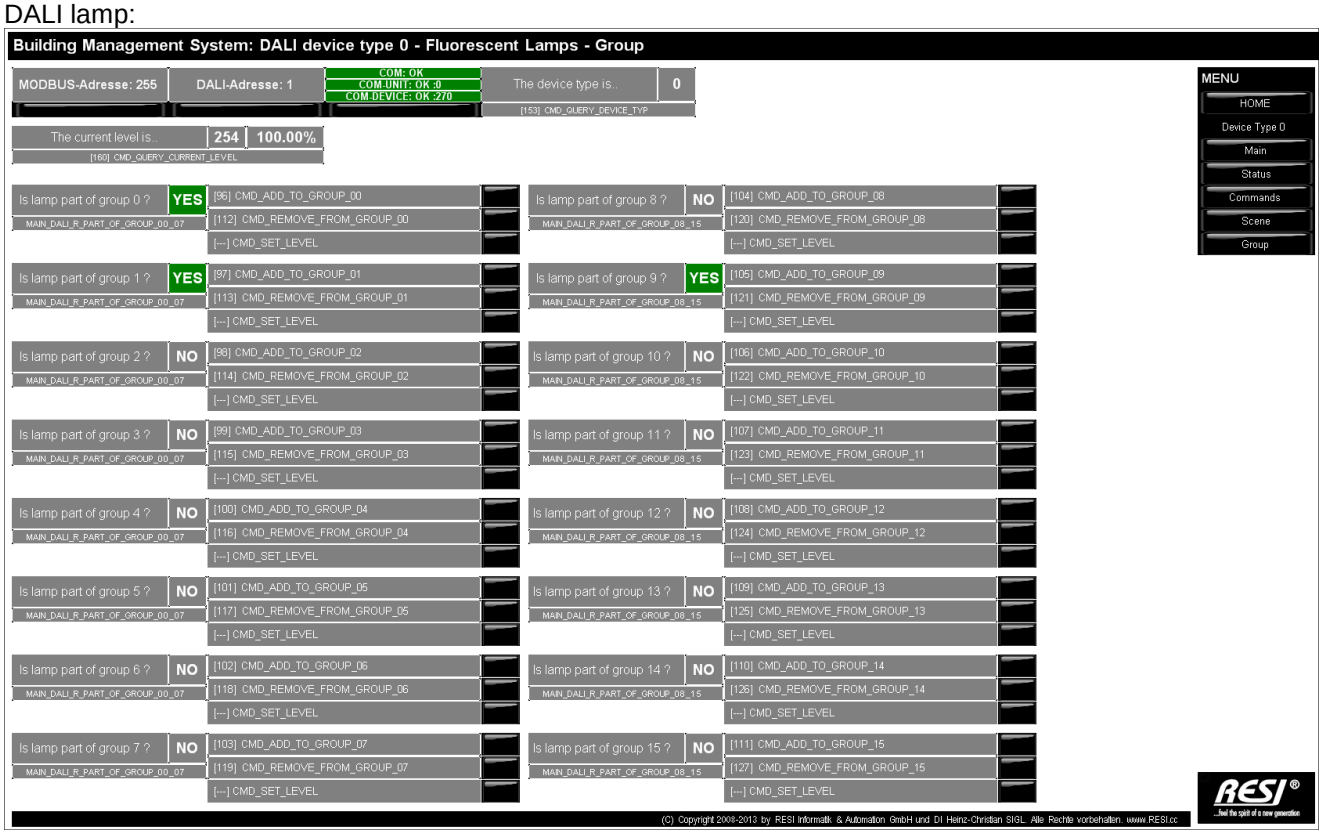
Again you will find a list view for all valid DALI commands of the standard DALI lamp module. When you click on the page SCENE button, you will see a page to configure all the scenes of a DALI lamp:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung



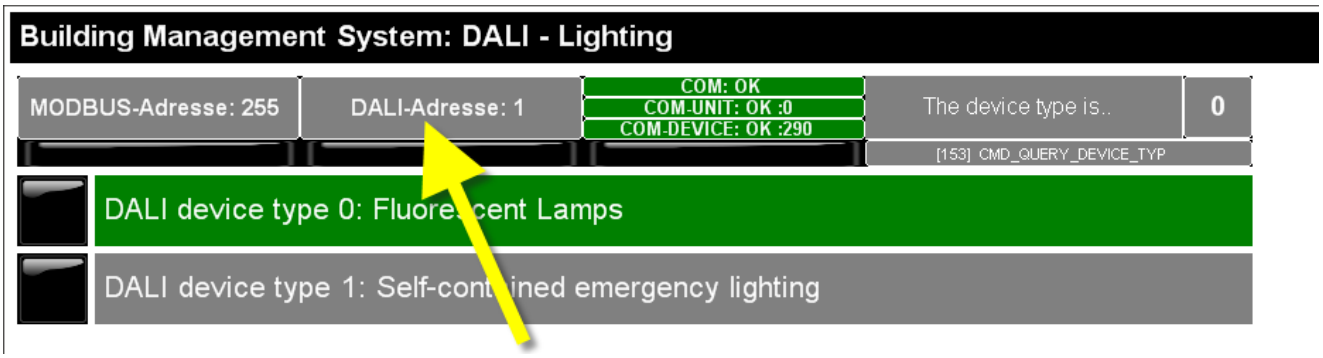
The same will happens with the GROUP button. You will see a page with all functions to define the groups of a DALI lamp:



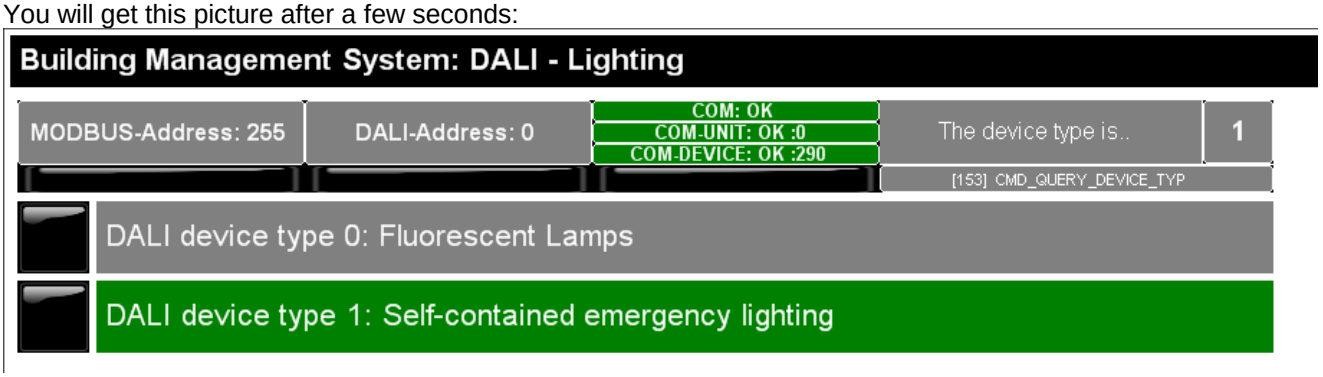
Again we do not discuss details for a standard DALI lamp module. This should be common sense for all, who work with a DALI lamp system. The button HOME leads to the main screen of the visualization. You might have noticed that on the main screen, only the DALI device type 0 is highlighted with green color. And you see the field DALI-Address with the current value of 1. This is the short address of the DALI lamp module. THE ALVIT module for the emergency lamp has in this sample the DALI address 0. So click on the black button below, and change the address to 0.

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

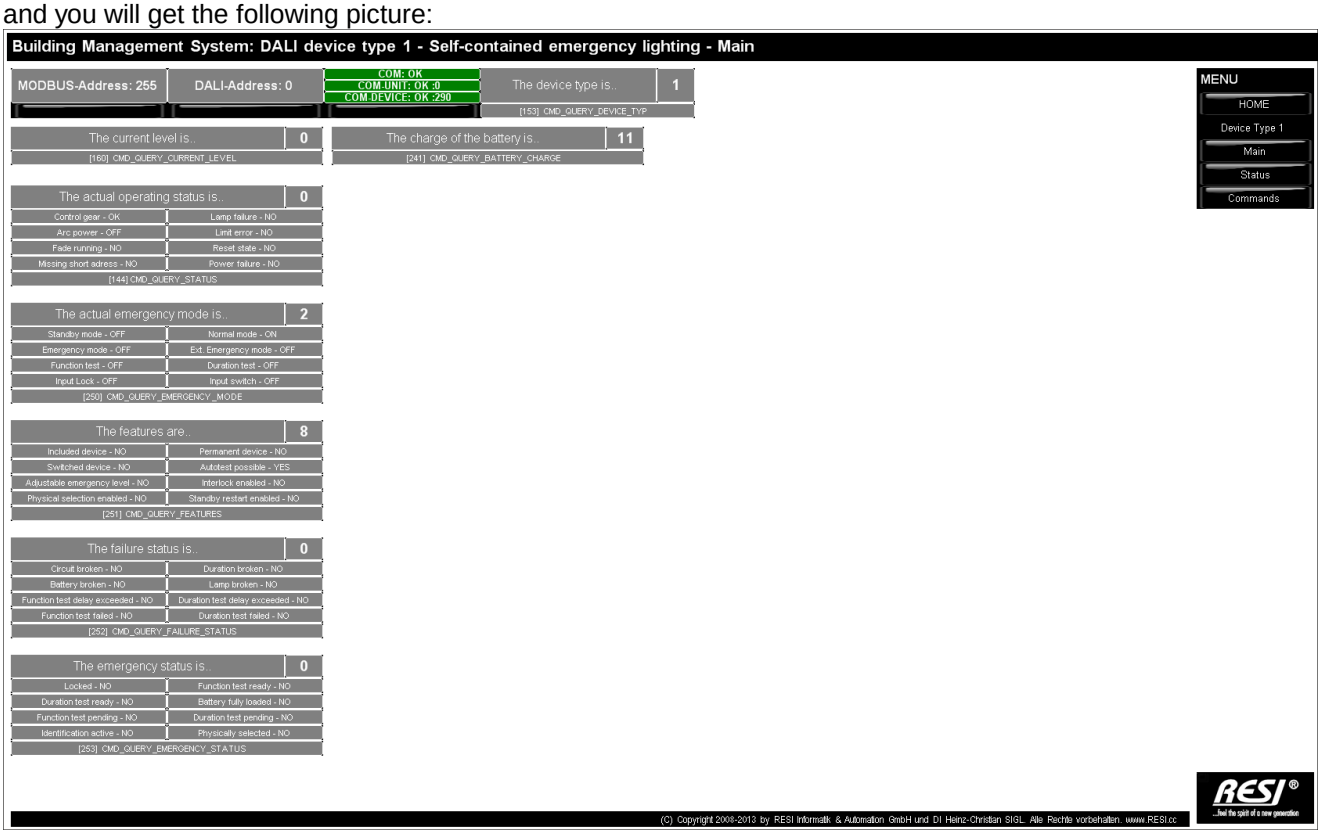
Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung



You will get this picture after a few seconds:



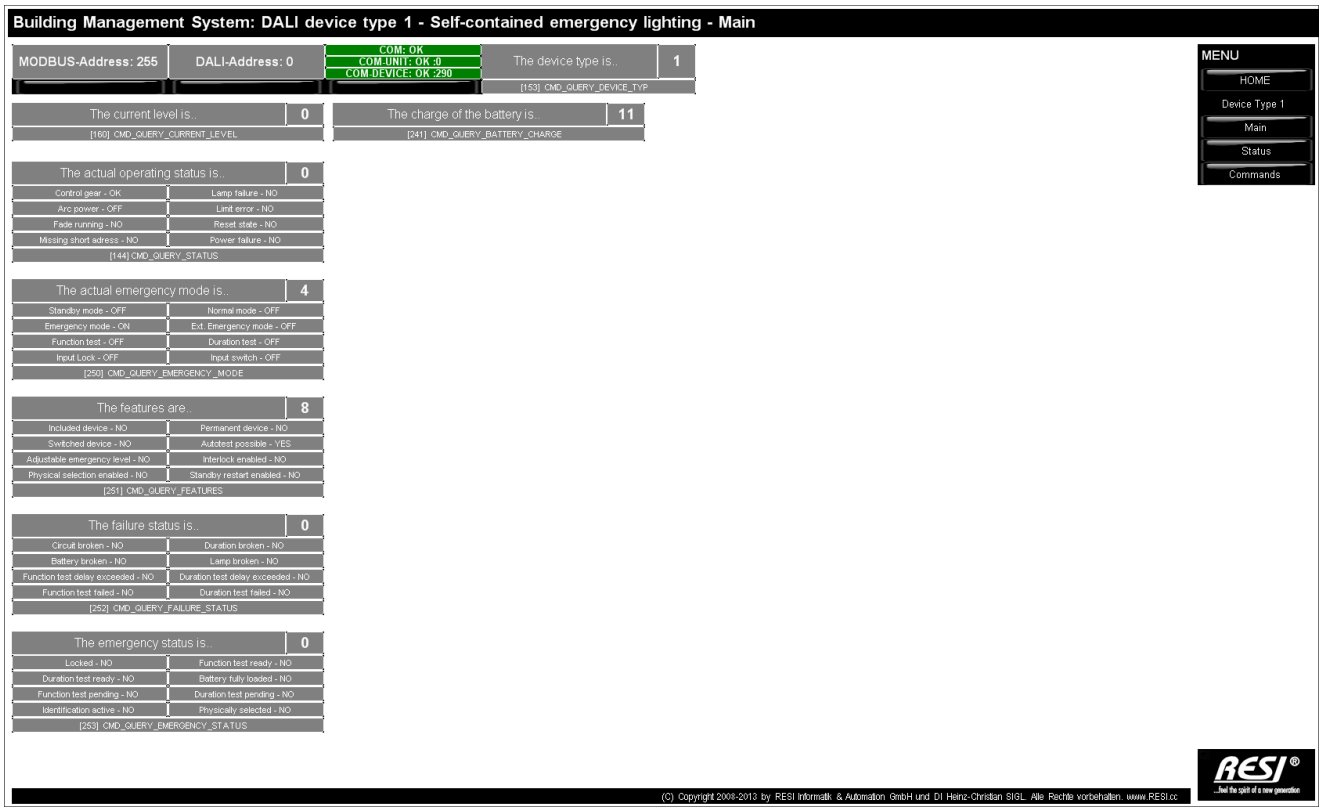
Now our demo system communicates with the emergency lamp module. So click on the button on the left side and you will get the following picture:



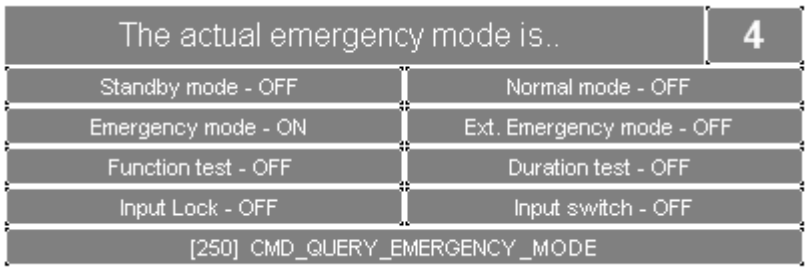
We will do a closer look on the parameters of the emergency module here. On the top you will find the actual level of the emergency module and the actual battery status (currently 11). When you switch off the 230Vac for the DALI lamp module, the display changes to:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung



Take a closer look to these fields:



You will notice that the normal mode is off due to the lack of 230Vac power supply to the DALI lamp and the emergency mode is on. So this is a good way to get the status of the emergency system with this command (CMD_QUERY_EMERGENCY_MODE). In the file DALI_DTY1.SIB you will find the declaration of this command:

```
intconst DALI_DTY1_CMD_QUERY_EMERGENCY_MODE =250  //YAAA AAA1 1111 1010  QUERY EMERGENCY MODE
```

Search also for the corresponding software module:

```
//intconst DALI_DTY1_CMD_QUERY_EMERGENCY_MODE =250  //YAAA AAA1 1111 1010  QUERY EMERGENCY MODE
function int DALI_DTY1_QueryEmergencyMode(num Handle,int Unit,int Adr)
  return DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY EMERGENCY MODE,CMD DEVICE TYPE 1)
endfunction
```

You see in comparison to a normal lamp, you need a different way to request the status for a emergency module. Again we search for the function DALI_Query_272_1F. This function is defined in the module DALI.sib:

```
//-------------------------------------------------------------------------------------------------
//==
//Specific execution commands 224 - 255
//==
// 1.) One forward frame (for querys)
// a.) Write the enable device type command with value = 1 [0xC101] to register 541 of RESI-DALI-
MODBUS
// b.) Write the short adress 0..63 (aaaa: highbyte) of the lamp and the command (cccc: lowbyte)
[0xaaaacccc] to register 511 of RESI-DALI-MODBUS
// c.) wait
// d.) Read the result of query from register 511 of RESI-DALI-MODBUS
//! If query value is equal 0x8000, the lamp haven't answered !
//-------------------------------------------------------------------------------------------------
function int DALI_Query_272_1F(num Handle,int Unit,int Adr,int Cmd, int DeviceType)
  //intconst CMD_ENABLE_DEVICE_TYPE = 0xC100
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  //intconst MDB_REG_SPEC_CMD = 541
  //intconst MDB_REG_KRZ_EXT_CMD = 511
  DALI_TempValue = CMD_ENABLE_DEVICE_TYPE + DeviceType

  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_SPEC_CMD,DALI_TempValue)then
    DALI_TempValue = Cmd + (Adr<<8)
    if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB REG KRZ EXT CMD,DALI TempValue)then
      Me.Sleep(50)
      if ModbusMaster.ReadUnSignedInt16(Handle,Unit, MDB_REG_KRZ_EXT_CMD, DALI_QueryValue) then
        if DALI_QueryValue == 0x8000 then
           DALI QueryOk = 0 //No answer from device
           DALI_R_COM_OK = false
           DALI_R_COM_ERROR_CNT_DEVICE = DALI_R_COM_ERROR_CNT_DEVICE + 1
        else
           DALI QueryOk = 1 //Query was answered by device
           DALI_R_COM_OK = true
        endif
      else
    DALI CmdOk = -1 //No answer from RESI-DALI-MODBUS
        DALI_R_COM_OK = false
        DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
      endif
    else
      DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
      DALI R COM OK = false
      DALI R COM ERROR CNT UNIT = DALI R COM ERROR CNT UNIT + 1
    endif
  else
    DALI CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI R COM OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  if DALI QueryOk != 1 then
    LOG_Message("DALI_Query_272_1F          -          NOK          Adr:"+String.FormatInt(Adr,2)+"
Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI QueryValue,4)+"["+String.FormatHex(DALI QueryValue,4)+"]")
  else
    LOG_Message("DALI_Query_272_1F          -          OK          Adr:"+String.FormatInt(Adr,2)+"
Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI QueryValue,4)+"["+String.FormatHex(DALI QueryValue,4)+"]")
  endif
  return DALI_QueryOk
endfunction
```

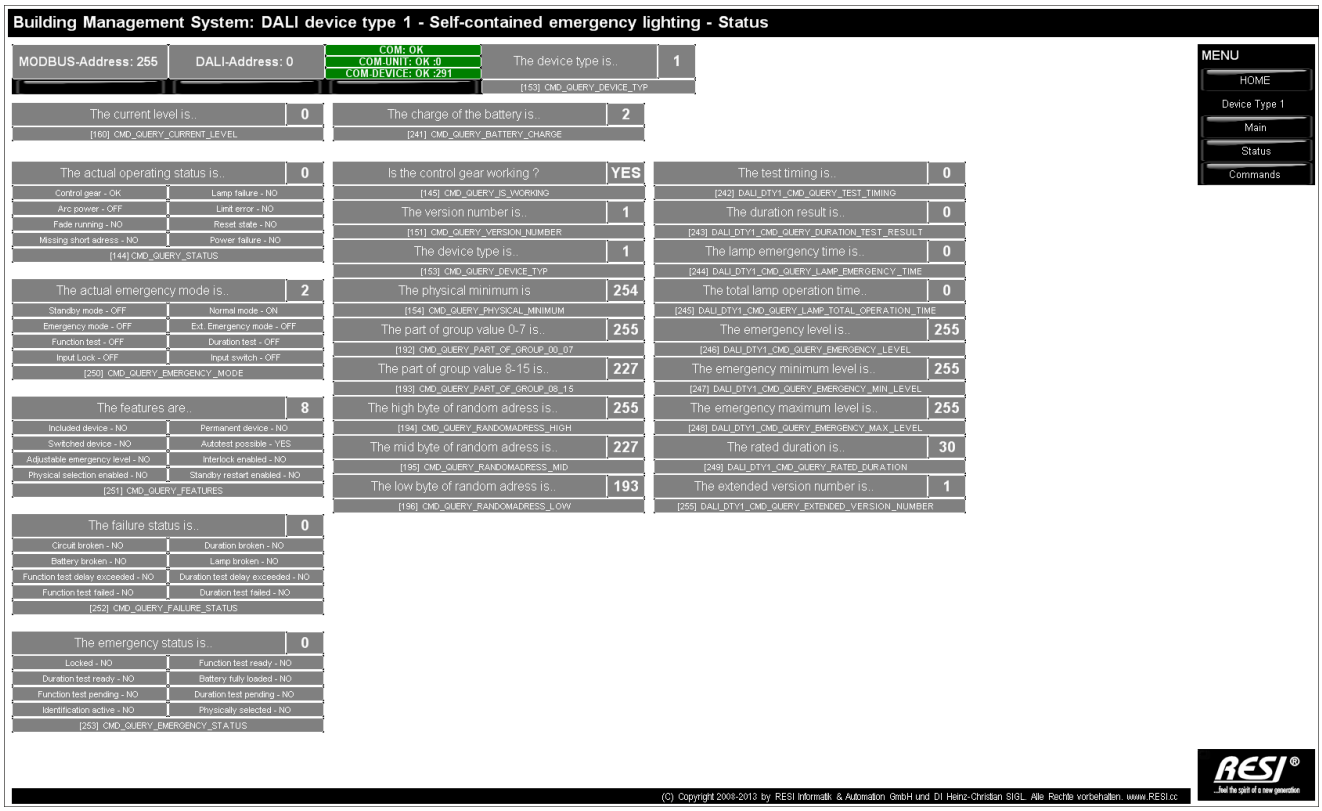
When you search for the usage of the function DALI_DTY1_QueryEmergencyStatus, you will find in the file DALI-DTY1.sib the following code:

```
    DALI_TempValue = DALI_DTY1_QueryEmergencyStatus(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
      DALI DTY1 R EMERGENCY STATUS = DALI QueryValue
       DALI_DTY1_R_EMERGENCY_STATUS_LOCKED = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS & 0x01)
      DALI_DTY1_R_EMERGENCY_STATUS_FUNCTION_TEST_READY =
           Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS & 0x02)
      DALI DTY1 R EMERGENCY STATUS DURATION TEST READY =
           Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS & 0x04)
      DALI_DTY1_R_EMERGENCY_STATUS_BATTERY_LOADED =
           Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS & 0x08)
      DALI DTY1 R EMERGENCY STATUS FUNCTION TEST PENDING =
           Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS & 0x10)
      DALI_DTY1_R_EMERGENCY_STATUS_DURATION_TEST_PENDING =
           Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS & 0x20)
      DALI_DTY1_R_EMERGENCY_STATUS_IDENTIFICATION_ACTIVE =
           Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS & 0x40)
      DALI DTY1 R EMERGENCY STATUS PHYSICAL SELECTED =
           Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS & 0x80)
    endif
```

No you will know how you have to interpret the result of the function, to get the correct status of the single bits. This method is valid for all requests in our demo software. So this should be a very easy way to build a own software module for using our converter to control and monitor DALI emergency lamps. When you switch to the page STATUS you will get the following picture:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung



On this page you see more status information about the DALI emergency lamp system.

The actual emergency mode is..	2
Standby mode - OFF	Normal mode - ON
Emergency mode - OFF	Ext. Emergency mode - OFF
Function test - OFF	Duration test - OFF
Input Lock - OFF	Input switch - OFF
[250] CMD_QUERY_EMERGENCY_MODE	

The features are..	8
Included device - NO	Permanent device - NO
Switched device - NO	Autotest possible - YES
Adjustable emergency level - NO	Interlock enabled - NO
Physical selection enabled - NO	Standby restart enabled - NO
[251] CMD_QUERY_FEATURES	

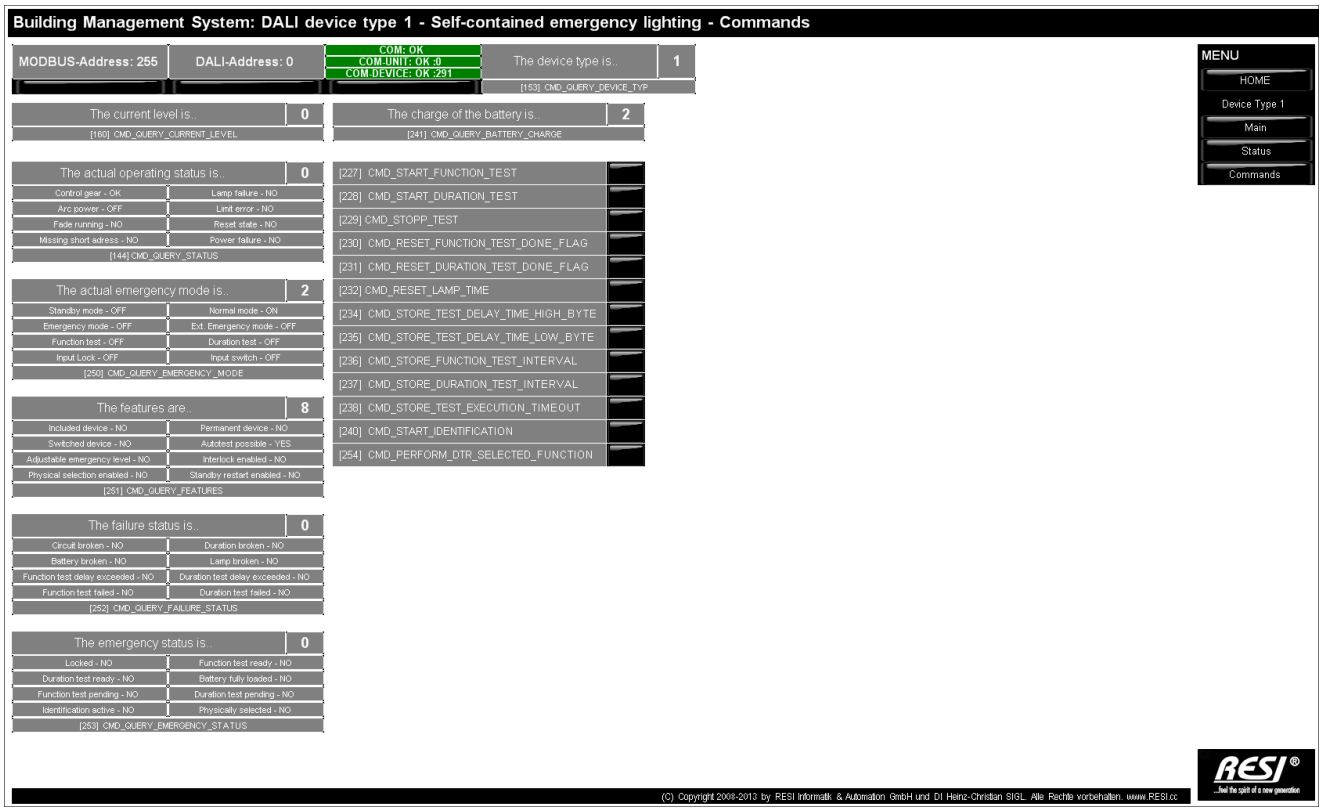
The failure status is..	0
Circuit broken - NO	Duration broken - NO
Battery broken - NO	Lamp broken - NO
Function test delay exceeded - NO	Duration test delay exceeded - NO
Function test failed - NO	Duration test failed - NO
[252] CMD_QUERY_FAILURE_STATUS	

The emergency status is..	0
Locked - NO	Function test ready - NO
Duration test ready - NO	Battery fully loaded - NO
Function test pending - NO	Duration test pending - NO
Identification active - NO	Physically selected - NO
[253] CMD_QUERY_EMERGENCY_STATUS	

Click on the Commands page and you will get the following command list:

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung



Here you will find a list of possible commands for the emergency lamp system. A very useful function is CMD_START_IDENTIFICATION. After sending this command, the LED of the emergency module flashes RED/YELLOW to display the selected lamp for about 30 seconds. Take the CMD_START_FUNCTION_TEST to switch the lamp from normal mode to emergency mode. When you want to end the emergency mode test, send the command CMD_STOPP_TEST.

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

5 The RESI-IDE software

5.1 General

We wrote demonstration software to access the ALVIT system via our DALI master module. Due to the fact that our RESI-T4 system is a very simple BASIC like language, we document the whole source code for reference implementations for our customers here.

5.2 SIBASIC module MAIN.SIB

```
taskname "MAIN"

include "LOG"
include "CYCLE"
include "DALI"
include "DALI_DTY1"

num hDali

int TempInt

sub Main

  bit lTrigger
  //---
  LOG_Init()
  LOG_Message("Start")
  CYCLE TimeInit()
  if DALI_K_COM != "" then
    hDali = ModbusMaster.CreateInterface(DALI_K_COM)
    DALI_Init(hDali)
    DALI_DTY1_Init(hDali)
  endif
  //---
  while true
    //================================================================
    Params.SyncIn()
    //------------------------------
    if Timer.Impulse(2000) then
      lTrigger = not lTrigger
    endif
    //------------------------------
    if Changed.Bit(lTrigger) then
    endif
    DALI_Update()
    DALI_DTY1_Update()
    //------------------------------
    Events.ProcessAll()
    //------------------------------
    CYCLE TimeCalc()
    //================================================================
    Params.SyncOut()
    //================================================================
  endwhile
endsub
//----------------------------------------------------------------
```

5.3 INCLUDE module LOG.SIB

```
// ================================================
// Parameter
// ================================================
bitin LOG_P_ENABLED = true
  symbol = "$auto: $param:Logging Ein / Aus:true"
endbitin

string LOG_PREFIX = ""
// ================================================
// Programmlogik
// ================================================

sub LOG_EnabledChange(string Name, bit Value)
  LOG_P_ENABLED = Value
endsub
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
sub LOG_Init()
  Params.OnSymbolChange(Params.IndexOfBit(LOG_P_ENABLED), 0, LOG_EnabledChange)
  string MyName
  MyName = Me.GetName()
  LOG_PREFIX = String.Sub(MyName, 0, String.FirstPosOf(MyName, "."))
endsub

sub LOG_Message(string Message)
  if LOG_P_ENABLED then
    Log.String(LOG_PREFIX+ ": " + Message)
  endif
endsub
sub LOG_Nachricht(string Nachricht)
  if LOG_P_ENABLED then
    Log.String(LOG_PREFIX+ ": " + Nachricht)
  endif
endsub
```

5.4 INCLUDE module CYCLE.SIB

```
// ================================================
// Parameter
// ================================================
numout R_CYCLETIME = 0.
  symbol = "$auto:_$param:Zykluszeit:false"
endnumout
//------------------------------------------------
num OldTime
num NewTime
// ================================================
// Programmlogik
// ================================================
sub CYCLE_TimeInit()
  //--------------
  OldTime = DateTime.ActualTime2Num()
  //--------------
endsub
sub CYCLE_TimeCalc()
  //--------------
  NewTime = DateTime.ActualTime2Num()
  R_CYCLETIME = (NewTime-OldTime)*24.*3600.
  OldTime=NewTime
  //--------------
endsub
// ================================================
```

5.5 INCLUDE module DALI.SIB

```
// ================================================
// Parameter für DALI Kommunikation
// ================================================
intin DALI_P_UNIT = 255
  symbol = "$auto:_$param:Modbus RTU unit Id (adress) of RESI-DALI-MODBUS:true"
endintin
intin DALI_P_ADR = 0
  symbol = "$auto:_$param:DALI short adress of the device:true"
endintin

bitout DALI_R_COM_OK = false
  symbol = "$auto:_$param:Error communication :false"
endbitout
bitout DALI_R_COM_OK_UNIT = false
  symbol = "$auto:_$param:Error communication with RESI-DALI-MODBUS:false"
endbitout
bitout DALI_R_COM_OK_DEVICE = false
  symbol = "$auto:_$param:Error communication with the device:false"
endbitout
intout DALI_R_COM_ERROR_CNT_UNIT = 0
  symbol = "$auto:_$param:Error coutervalue communication with RESI-DALI-MODBUS:false"
endintout
intout DALI_R_COM_ERROR_CNT_DEVICE = 0
  symbol = "$auto: $param:Error coutervalue communication with the device:false"
endintout

intin DALI_I_COM_ERROR_CNT_UNIT = 0
  symbol = "$auto: DALI_R_COM_ERROR_CNT_UNIT:Error coutervalue communication with RESI-DALI-
MODBUS:false"
endintin
intin DALI_I_COM_ERROR_CNT_DEVICE = 0
  symbol = "$auto:_DALI_R_COM_ERROR_CNT_DEVICE:Error coutervalue communication with the
device:false"
endintin

bitin DALI_P_COM_ERROR_RESET = false
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  symbol = "$auto:_$param: to reset Error counter:false"
endbitin

intin DALI P VIS UPDATE SCENE = 0
  symbol = "$auto:_$param:Updatevalue for visualisation:true"
endintin
// ================================================
intin DALI P CMD = 0
  symbol = "$auto:_$param:Command:false"
endintin
intin DALI P CMD PARAM 01 = 0
  symbol = "$auto:_$param:Param for command execution:false"
endintin
intin DALI_P_CMD_PARAM_02 = 0
  symbol = "$auto: $param:Param for command execution:false"
endintin
// ================================================
intout DALI R STATUS = 0
  symbol = "$auto: $param: Operating status value:false"
endintout
bitout DALI_R_STATUS_CONTROL_GEAR = false
  symbol = "$auto: $param: Operating status [OK;NOK]:false"
endbitout
bitout DALI_R_STATUS_FAILURE = false
  symbol = "$auto: $param: Failure status [NO;YES]:false"
endbitout
bitout DALI_R_STATUS_ARC_POWER = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI R STATUS LIMIT ERROR = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI R STATUS FADE RUNNING = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI R STATUS RESET STATE = false
  symbol = "$auto: $param: Operating status [NO;YES]:false"
endbitout
bitout DALI_R_STATUS_MISSING_SHORTADRESS = false
  symbol = "$auto: $param: Failure status [NO;YES]:false"
endbitout
bitout DALI_R_STATUS_POWER_FAILURE = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout

intout DALI_R_IS_WORKING = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_IS_FAILURE = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_IS_OPERATING = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R OUT OF LIMIT = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R RESET STATE = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_MISSING_SHORTADRESS = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_VERSION_NUMBER = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_DTR_CONTENT = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_DEVICE_TYP = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R PHYSICAL MINIMUM = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R POWER FAILURE MODE = 0
  symbol = "$auto: $param: .. :false"
endintout

intout DALI R CURRENT LEVEL = 0
  symbol = "$auto:_$param: .. :false"
endintout
numout DALI_R_CURRENT_LEVEL_100 = 0.
  symbol = "$auto: $param: .. :false"
endnumout

intout DALI_R_MAX_LEVEL = 0
```

```
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_MIN_LEVEL = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_POWER_UP_LEVEL = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_SYSTEM_FAILURE_LEVEL = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_FADE_TIME_AND_RATE = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R FADE TIME = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R FADE RATE = 0
  symbol = "$auto: $param: .. :false"
endintout

intout DALI R LEVEL OF SCENE 00 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R LEVEL OF SCENE 01 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_02 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_03 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_04 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R LEVEL OF SCENE 05 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R LEVEL OF SCENE 06 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_07 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_08 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_09 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_10 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R LEVEL OF SCENE 11 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R LEVEL OF SCENE 12 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_13 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_14 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_LEVEL_OF_SCENE_15 = 0
  symbol = "$auto:_$param: .. :false"
endintout

intout DALI_R_PART_OF_GROUP_00_07 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_PART_OF_GROUP_08_15 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI R PART OF GROUP = 0
  symbol = "$auto:_$param: .. :false"
endintout
stringout DALI R PART OF GROUP HEX = ""
  symbol = "$auto:_$param: .. :false"
endstringout

intout DALI R RANDOMADRESS HIGH = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_RANDOMADRESS_MID = 0
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_R_RANDOMADRESS_LOW = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_R_RANDOMADRESS = 0
  symbol = "$auto: $param: .. :false"
endintout
stringout DALI_R_RANDOMADRESS_HEX = ""
  symbol = "$auto:_$param: .. :false"
endstringout

// ====================================================
// port configuration
// ====================================================
stringin DALI_K_COM = "DALI_485"
endstringin
// ====================================================
// const for update visualisation
// ====================================================
intconst VIS_UPDATE_ALL = 0
intconst VIS_SCENE_MAIN = 1
intconst VIS_SCENE_STATUS = 2
intconst VIS_SCENE_DPS_COMMANDS = 3
intconst VIS_SCENE_SCENE = 4
intconst VIS_SCENE_GROUP = 5
// ====================================================
// const modbus params (registers of RESI-DALI-MODBUS)
// ====================================================
intconst MDB_REG_SPEC_CMD = 541
intconst MDB_REG_KRZ_LEVEL = 510
intconst MDB_REG_KRZ_EXT_CMD = 511
intconst MDB_REG_KRZ_EXT_CMD_100MS = 512
intconst MDB_REG_GRP_LEVEL = 520
intconst MDB_REG_GRP_EXT_CMD = 521
intconst MDB_REG_GRP_EXT_CMD_100MS = 522
intconst MDB_REG_ALL_LEVEL = 530
// ====================================================
// device type 0 DALI commands
// ====================================================
// Direct Power Set Commands
intconst CMD_EXTINGUISH_POWER            = 0x0000 //0
intconst CMD_DIMM_UP_POWER_200_MS        = 0x0001 //1
intconst CMD_DIMM_DOWN_POWER_200_MS      = 0x0002 //2
intconst CMD_STEP_UP_POWER               = 0x0003 //3
intconst CMD_STEP_DOWN_POWER             = 0x0004 //4
intconst CMD_SET_POWER_TO_MAXIMUM        = 0x0005 //5
intconst CMD_SET_POWER_TO_MINIMUM        = 0x0006 //6
intconst CMD_STEP_DOWN_POWER_AND_SWITCH  = 0x0007 //7
intconst CMD_STEP_UP_POWER_AND_SWITCH    = 0x0008 //8
// Set Power to scene level Commands
intconst CMD_SET_POWER_TO_SCENE_00_LEVEL = 0x0010 //16
intconst CMD_SET_POWER_TO_SCENE_01_LEVEL = 0x0011 //17
intconst CMD_SET_POWER_TO_SCENE_02_LEVEL = 0x0012 //18
intconst CMD_SET_POWER_TO_SCENE_03_LEVEL = 0x0013 //19
intconst CMD_SET_POWER_TO_SCENE_04_LEVEL = 0x0014 //20
intconst CMD_SET_POWER_TO_SCENE_05_LEVEL = 0x0015 //21
intconst CMD_SET_POWER_TO_SCENE_06_LEVEL = 0x0016 //22
intconst CMD_SET_POWER_TO_SCENE_07_LEVEL = 0x0017 //23
intconst CMD_SET_POWER_TO_SCENE_08_LEVEL = 0x0018 //24
intconst CMD_SET_POWER_TO_SCENE_09_LEVEL = 0x0019 //25
intconst CMD_SET_POWER_TO_SCENE_10_LEVEL = 0x001A //26
intconst CMD_SET_POWER_TO_SCENE_11_LEVEL = 0x001B //27
intconst CMD_SET_POWER_TO_SCENE_12_LEVEL = 0x001C //28
intconst CMD_SET_POWER_TO_SCENE_13_LEVEL = 0x001D //29
intconst CMD_SET_POWER_TO_SCENE_14_LEVEL = 0x001E //30
intconst CMD_SET_POWER_TO_SCENE_15_LEVEL = 0x001F //31
// Configuration Commands
intconst CMD_SET_DEFAULT_SETTINGS       = 0x0020 //32
intconst CMD_STORE_CURRENT_LEVEL_IN_DTR = 0x0021 //33
//
intconst CMD_STORE_DTR_AS_MAX_LEVEL             = 0x002A //42
intconst CMD_STORE_DTR_AS_MIN_LEVEL             = 0x002B //43
intconst CMD_STORE_DTR_AS_SYSTEM_FAILURE_LEVEL  = 0x002C //44
intconst CMD_STORE_DTR_AS_POWER_ON_LEVEL        = 0x002D //45
//
intconst CMD_STORE_DTR_AS_FADE_TIME  = 0x002E //46
intconst CMD_STORE_DTR_AS_FADE_RATE  = 0x002F //47
//
intconst CMD_STORE_DTR_AS_SCENE_00  = 0x0040 //64
intconst CMD_STORE_DTR_AS_SCENE_01  = 0x0041 //65
intconst CMD_STORE_DTR_AS_SCENE_02  = 0x0042 //66
intconst CMD_STORE_DTR_AS_SCENE_03  = 0x0043 //67
intconst CMD_STORE_DTR_AS_SCENE_04  = 0x0044 //68
intconst CMD_STORE_DTR_AS_SCENE_05  = 0x0045 //69
intconst CMD_STORE_DTR_AS_SCENE_06  = 0x0046 //70
intconst CMD_STORE_DTR_AS_SCENE_07  = 0x0047 //71
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
intconst CMD_STORE_DTR_AS_SCENE_08  = 0x0048 //72
intconst CMD_STORE_DTR_AS_SCENE_09  = 0x0049 //73
intconst CMD_STORE_DTR_AS_SCENE_10  = 0x004A //74
intconst CMD STORE DTR AS SCENE 11  = 0x004B //75
intconst CMD_STORE_DTR_AS_SCENE_12  = 0x004C //76
intconst CMD_STORE_DTR_AS_SCENE_13  = 0x004D //77
intconst CMD STORE DTR AS SCENE 14  = 0x004E //78
intconst CMD STORE DTR AS SCENE 15  = 0x004F //79
//
intconst CMD_REMOVE_FROM_SCENE_00  = 0x0050 //80
intconst CMD REMOVE FROM SCENE 01  = 0x0051 //81
intconst CMD_REMOVE_FROM_SCENE_02  = 0x0052 //82
intconst CMD_REMOVE_FROM_SCENE_03  = 0x0053 //83
intconst CMD_REMOVE_FROM_SCENE_04  = 0x0054 //84
intconst CMD REMOVE FROM SCENE 05  = 0x0055 //85
intconst CMD_REMOVE_FROM_SCENE_06  = 0x0056 //86
intconst CMD_REMOVE_FROM_SCENE_07  = 0x0057 //87
intconst CMD REMOVE FROM SCENE 08  = 0x0058 //88
intconst CMD REMOVE FROM SCENE 09  = 0x0059 //89
intconst CMD_REMOVE_FROM_SCENE_10  = 0x005A //90
intconst CMD_REMOVE_FROM_SCENE_11  = 0x005B //91
intconst CMD REMOVE FROM SCENE 12  = 0x005C //92
intconst CMD_REMOVE_FROM_SCENE_13  = 0x005D //93
intconst CMD_REMOVE_FROM_SCENE_14  = 0x005E //94
intconst CMD REMOVE FROM SCENE 15  = 0x005F //95
//
intconst CMD_ADD_TO_GROUP_00  = 0x0060 //96
intconst CMD_ADD_TO_GROUP_01  = 0x0061 //97
intconst CMD ADD TO GROUP 02  = 0x0062 //98
intconst CMD ADD TO GROUP 03  = 0x0063 //99
intconst CMD_ADD_TO_GROUP_04  = 0x0064 //100
intconst CMD_ADD_TO_GROUP_05  = 0x0065 //101
intconst CMD ADD TO GROUP 06  = 0x0066 //102
intconst CMD_ADD_TO_GROUP_07  = 0x0067 //103
intconst CMD_ADD_TO_GROUP_08  = 0x0068 //104
intconst CMD ADD TO GROUP 09  = 0x0069 //105
intconst CMD ADD TO GROUP 10  = 0x006A //106
intconst CMD_ADD_TO_GROUP_11  = 0x006B //107
intconst CMD_ADD_TO_GROUP_12  = 0x006C //108
intconst CMD ADD TO GROUP 13  = 0x006D //109
intconst CMD_ADD_TO_GROUP_14  = 0x006E //110
intconst CMD_ADD_TO_GROUP_15  = 0x006F //111
//
intconst CMD REMOVE FROM GROUP 00  = 0x0070 //112
intconst CMD_REMOVE_FROM_GROUP_01  = 0x0071 //113
intconst CMD_REMOVE_FROM_GROUP_02  = 0x0072 //114
intconst CMD REMOVE FROM GROUP 03  = 0x0073 //115
intconst CMD REMOVE FROM GROUP 04  = 0x0074 //116
intconst CMD_REMOVE_FROM_GROUP_05  = 0x0075 //117
intconst CMD_REMOVE_FROM_GROUP_06  = 0x0076 //118
intconst CMD REMOVE FROM GROUP 07  = 0x0077 //119
intconst CMD_REMOVE_FROM_GROUP_08  = 0x0078 //120
intconst CMD_REMOVE_FROM_GROUP_09  = 0x0079 //121
intconst CMD_REMOVE_FROM_GROUP_10  = 0x007A //122
intconst CMD REMOVE FROM GROUP 11  = 0x007B //123
intconst CMD_REMOVE_FROM_GROUP_12  = 0x007C //124
intconst CMD_REMOVE_FROM_GROUP_13  = 0x007D //125
intconst CMD REMOVE FROM GROUP 14  = 0x007E //126
intconst CMD REMOVE FROM GROUP 15  = 0x007F //127
//
intconst CMD_STORE_DTR_AS_SHORTADRESS  = 0x0080 //128
// ================================================
// device type 0 DALI query commands
// ================================================
intconst CMD QUERY STATUS              = 0x0090 //144
intconst CMD QUERY IS WORKING          = 0x0091 //145
intconst CMD_QUERY_IS_FAILURE          = 0x0092 //146
intconst CMD_QUERY_IS_OPERATING        = 0x0093 //147
intconst CMD QUERY OUT OF LIMIT        = 0x0094 //148
intconst CMD_QUERY_RESET_STATE         = 0x0095 //149
intconst CMD_QUERY_MISSING_SHORTADRESS = 0x0096 //150
intconst CMD_QUERY_VERSION_NUMBER      = 0x0097 //151
intconst CMD QUERY DTR CONTENT         = 0x0098 //152
intconst CMD_QUERY_DEVICE_TYP          = 0x0099 //153
intconst CMD_QUERY_PHYSICAL_MINIMUM    = 0x009A //154
intconst CMD QUERY POWER FAILURE MODE  = 0x009B //155

intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
intconst CMD_QUERY_MAX_LEVEL            = 0x00A1 //161
intconst CMD QUERY MIN LEVEL            = 0x00A2 //162
intconst CMD_QUERY_POWER_UP_LEVEL       = 0x00A3 //163
intconst CMD_QUERY_SYSTEM_FAILURE_LEVEL = 0x00A4 //164
intconst CMD_QUERY_FADE_TIME_AND_RATE   = 0x00A5 //165

intconst CMD_QUERY_LEVEL_OF_SCENE_00    = 0x00B0 //176
intconst CMD_QUERY_LEVEL_OF_SCENE_01    = 0x00B1 //177
intconst CMD_QUERY_LEVEL_OF_SCENE_02    = 0x00B2 //178
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
intconst CMD_QUERY_LEVEL_OF_SCENE_03    = 0x00B3 //179
intconst CMD_QUERY_LEVEL_OF_SCENE_04    = 0x00B4 //180
intconst CMD_QUERY_LEVEL_OF_SCENE_05    = 0x00B5 //181
intconst CMD QUERY LEVEL OF SCENE 06    = 0x00B6 //182
intconst CMD_QUERY_LEVEL_OF_SCENE_07    = 0x00B7 //183
intconst CMD_QUERY_LEVEL_OF_SCENE_08    = 0x00B8 //184
intconst CMD QUERY LEVEL OF SCENE 09    = 0x00B9 //185
intconst CMD QUERY LEVEL OF SCENE 10    = 0x00BA //186
intconst CMD_QUERY_LEVEL_OF_SCENE_11    = 0x00BB //187
intconst CMD_QUERY_LEVEL_OF_SCENE_12    = 0x00BC //188
intconst CMD QUERY LEVEL OF SCENE 13    = 0x00BD //189
intconst CMD_QUERY_LEVEL_OF_SCENE_14    = 0x00BE //190
intconst CMD_QUERY_LEVEL_OF_SCENE_15    = 0x00BF //191

intconst CMD QUERY PART OF GROUP 00 07  = 0x00C0 //192
intconst CMD_QUERY_PART_OF_GROUP_08_15  = 0x00C1 //193

intconst CMD QUERY RANDOMADRESS HIGH    = 0x00C2 //194
intconst CMD QUERY RANDOMADRESS MID     = 0x00C3 //195
intconst CMD_QUERY_RANDOMADRESS_LOW     = 0x00C4 //196
//
================================================================================================
=====
// special commands
//
================================================================================================
=====
intconst CMD_WRITE_DATA_TRANSFER_REGISTER    = 0xA300//Cmd-257
intconst CMD ENABLE DEVICE TYPE              = 0xC100//
intconst CMD ENABLE DEVICE TYPE 0            = 0xC100//Cmd-272 Wert = 0
intconst CMD_ENABLE_DEVICE_TYPE_1            = 0xC101//Cmd-272 Wert = 1
intconst CMD_ENABLE_DEVICE_TYPE_2            = 0xC102//Cmd-272 Wert = 2
intconst CMD ENABLE DEVICE TYPE 3            = 0xC103//Cmd-272 Wert = 3
intconst CMD_ENABLE_DEVICE_TYPE_4            = 0xC104//Cmd-272 Wert = 4
intconst CMD_ENABLE_DEVICE_TYPE_5            = 0xC105//Cmd-272 Wert = 5
intconst CMD ENABLE DEVICE TYPE 6            = 0xC106//Cmd-272 Wert = 6
intconst CMD ENABLE DEVICE TYPE 7            = 0xC107//Cmd-272 Wert = 7
intconst CMD_ENABLE_DEVICE_TYPE_8            = 0xC108//Cmd-272 Wert = 8
//
================================================================================================
=====
// device types
//
================================================================================================
=====
intconst CMD_DEVICE_TYPE_0 = 0x0000 //Fluorescent Lamps (Device type 0)
intconst CMD DEVICE TYPE 1 = 0x0001 //Self-contained emergency lighting (device type 1)
intconst CMD DEVICE TYPE 2 = 0x0002 //Discharge lamps (excluding fluorescent lamps) (device type 2)
intconst CMD_DEVICE_TYPE_3 = 0x0003 //Low voltage halogen lamps (device type 3)
intconst CMD_DEVICE_TYPE_4 = 0x0004 //Supply voltage controller for incandescent lamps (device type
4)
intconst  CMD_DEVICE_TYPE_5  =  0x0005  //Conversion  from  digital  signal  into  d.  c.  voltage  (device
type 5)
intconst CMD_DEVICE_TYPE_6 = 0x0006 //LED modules (device type 6)
intconst CMD DEVICE TYPE 7 = 0x0007 //Switching function (device type 7)
intconst CMD_DEVICE_TYPE_8 = 0x0008 //Colour control (device type 8)
//
================================================================================================
=====
int DALI_TempValue

int DALI QueryOk
int DALI_QueryValue
int DALI_CmdOk

num DALI Handle
//
================================================================================================
=====
//Level commands
//
================================================================================================
=====
//------------------------------------------------------------------------------------------------
--------
// 1.) Single device
// a.) Write the short adress 0..63 of the device(aaaa: highbyte) and the required level 0..254
(cccc: lowbyte) [0xaaaacccc] to register 510 of RESI-DALI-MODBUS
function int DALI_SetShortLevel(num Handle,int Unit,int Adr,int Value)
  //intconst MDB REG KRZ LEVEL = 510
  DALI_TempValue = ((Adr&0x3F)<<8) + (Value&0xFF)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_KRZ_LEVEL,DALI_TempValue)then
    DALI_CmdOk = 1 // Communication with RESI-DALI-MODBUS Ok
    DALI R COM OK = true
  else
    DALI_CmdOk = -1 // No answer from RESI-DALI-MODBUS
    DALI_R_COM_OK = false
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  return DALI_CmdOk
endfunction
//-------------------------------------------------------------------------------------------------
--------
// 2.) Group of devices
// a.) Write the group number 0..15 of the devices(aaaa: highbyte) and the required level 0..254
(cccc: lowbyte) [0xaaaacccc] to register 520 of RESI-DALI-MODBUS
function int DALI_SetGroupLevel(num Handle,int Unit,int Grp,int Value)
  //intconst MDB_REG_GRP_LEVEL = 520
  DALI_TempValue = ((Grp&0x0F)<<8) + (Value&0xFF)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_GRP_LEVEL,DALI_TempValue)then
    DALI_CmdOk = 1 // Communication with RESI-DALI-MODBUS Ok
    DALI_R_COM_OK = true
  else
    DALI_CmdOk = -1 // No answer from RESI-DALI-MODBUS
    DALI_R_COM_OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  return DALI_CmdOk
endfunction

//-------------------------------------------------------------------------------------------------
--------
// 3.) All devices
// a.) Write the required level 0..254 (cccc: lowbyte) [0x....cccc] to register 530 of RESI-DALI-
MODBUS
function int DALI_SetAllLevel(num Handle,int Unit,int Value)
  //intconst MDB_REG_ALL_LEVEL = 530
  DALI_TempValue = (Value&0xFF)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_ALL_LEVEL,DALI_TempValue)then
    DALI_CmdOk = 1 // Communication with RESI-DALI-MODBUS Ok
    DALI_R_COM_OK = true
  else
    DALI_CmdOk = -1 // No answer from RESI-DALI-MODBUS
    DALI_R_COM_OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  return DALI_CmdOk
endfunction
//
=================================================================================================
=====
//Specific execution commands 224 - 255
//==
// 1.) One forward frame (for querys)
// a.) Write the enable device type command with value = 1 [0xC101] to register 541 of RESI-DALI-
MODBUS
// b.) Write the short adress 0..63 (aaaa: highbyte) of the lamp and the command (cccc: lowbyte)
[0xaaaacccc] to register 511 of RESI-DALI-MODBUS
// c.) wait
// d.) Read the result of query from register 511 of RESI-DALI-MODBUS
//! If query value is equal 0x8000, the lamp haven't answered !

// 2.) Two forward frames (for commands like start,stop,store,reset..)
// a.) Write the enable device type command with value = 1 [0xC101] to register 541 of RESI-DALI-
MODBUS
// b.) Write the short adress (aaaa: highbyte) of the lamp and the command (cccc: lowbyte)
[0xaaaacccc] to register 512 of RESI-DALI-MODBUS
// ! The command will be automatically repeated from RESI-DALI-MODBUS within 100ms !

// ================================================
// Functions
// ================================================
//-------------------------------------------------------------------------------------------------
function int DALI_Write_DTR(num Handle,int Unit,int Wert)
  bit lComOk
  DALI_TempValue = CMD_WRITE_DATA_TRANSFER_REGISTER + (Wert & 0x00ff)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_SPEC_CMD,DALI_TempValue)then
    DALI_CmdOk = 1 //Communication with RESI-DALI-MODBUS Ok
    DALI_R_COM_OK = true
  else
    DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI_R_COM_OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  return DALI_CmdOk
endfunction
//-------------------------------------------------------------------------------------------------
function int DALI_Send_Cmd_1F(num Handle,int Unit,int Adr,int Cmd)
  //intconst MDB_REG_KRZ_EXT_CMD = 511
  DALI_TempValue = Cmd + (Adr<<8)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_KRZ_EXT_CMD,DALI_TempValue)then
    DALI_CmdOk = 1 //Communication with RESI-DALI-MODBUS Ok
    DALI_R_COM_OK = true
  else
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI_R_COM_OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  return DALI_CmdOk
endfunction
//-----------------------------------------------------------------------------------------------
//==
//Specific execution commands 224 - 255
//==
// 1.) One forward frame (for querys)
// a.) Write the enable device type command with value = 1 [0xC101] to register 541 of RESI-DALI-
MODBUS
// b.) Write the short adress 0..63 (aaaa: highbyte) of the lamp and the command (cccc: lowbyte)
[0xaaaacccc] to register 511 of RESI-DALI-MODBUS
// c.) wait
// d.) Read the result of query from register 511 of RESI-DALI-MODBUS
//! If query value is equal 0x8000, the lamp haven't answered !
//-----------------------------------------------------------------------------------------------
function int DALI_Query_272_1F(num Handle,int Unit,int Adr,int Cmd, int DeviceType)
  //intconst CMD_ENABLE_DEVICE_TYPE = 0xC100
  //intconst MDB REG SPEC CMD = 541
  //intconst MDB_REG_KRZ_EXT_CMD = 511
  DALI_TempValue = CMD_ENABLE_DEVICE_TYPE + DeviceType

  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB REG SPEC CMD,DALI TempValue)then
    DALI_TempValue = Cmd + (Adr<<8)
    if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_KRZ_EXT_CMD,DALI_TempValue)then
      Me.Sleep(50)
      if ModbusMaster.ReadUnSignedInt16(Handle,Unit, MDB REG KRZ EXT CMD, DALI QueryValue) then
        if DALI_QueryValue == 0x8000 then
           DALI_QueryOk = 0 //No answer from device
           DALI R COM OK = false
           DALI_R_COM_ERROR_CNT_DEVICE = DALI_R_COM_ERROR_CNT_DEVICE + 1
        else
           DALI QueryOk = 1 //Query was answered by device
           DALI R COM OK = true
        endif
      else
    DALI CmdOk = -1 //No answer from RESI-DALI-MODBUS
        DALI_R_COM_OK = false
        DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
      endif
    else
      DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
      DALI_R_COM_OK = false
      DALI R COM ERROR CNT UNIT = DALI R COM ERROR CNT UNIT + 1
    endif
  else
    DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI R COM OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  if DALI_QueryOk != 1 then
    LOG Message("DALI Query 272 1F          -          NOK          Adr:"+String.FormatInt(Adr,2)+"
Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  else
    LOG Message("DALI Query 272 1F          -          OK          Adr:"+String.FormatInt(Adr,2)+"
Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  endif
  return DALI_QueryOk
endfunction
//-----------------------------------------------------------------------------------------------
// 2.) Two forward frames (for commands like start,stop,store,reset..)
// a.) Write the enable device type command with value = 1 [0xC101] to register 541 of RESI-DALI-
MODBUS
// b.) Write the short adress (aaaa: highbyte) of the lamp and the command (cccc: lowbyte)
[0xaaaacccc] to register 512 of RESI-DALI-MODBUS
// ! The command will be automatically repeated from RESI-DALI-MODBUS within 100ms !
//-----------------------------------------------------------------------------------------------
function int DALI Send 272 2F(num Handle,int Unit,int Adr,int Cmd,int DeviceType)
  //intconst CMD_ENABLE_DEVICE_TYPE = 0xC100
  //intconst MDB_REG_SPEC_CMD = 541
  //intconst MDB REG KRZ EXT CMD 100MS = 512
  DALI TempValue = CMD ENABLE DEVICE TYPE + DeviceType

  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_SPEC_CMD,DALI_TempValue)then
    DALI TempValue = Cmd + (Adr<<8)
    if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_KRZ_EXT_CMD_100MS,DALI_TempValue)then
    DALI_CmdOk = 1 //Communication with RESI-DALI-MODBUS Ok
      DALI_R_COM_OK = true

    else
      DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
      DALI_R_COM_OK = false
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
      DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
    endif
  else
    DALI CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI_R_COM_OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  return DALI CmdOk
endfunction
//-----------------------------------------------------------------------------------------------
function int DALI Query(num Handle,int Unit,int Adr,int Cmd)
  bit lComOk
  DALI_TempValue = Cmd + (Adr<<8)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_KRZ_EXT_CMD,DALI_TempValue)then
    Me.Sleep(50)
    lComOk= ModbusMaster.ReadUnSignedInt16(Handle,Unit, MDB_REG_KRZ_EXT_CMD, DALI_QueryValue)
    if lComOk then
      if DALI QueryValue == 0x8000 then
         DALI QueryOk = 0 //No answer from device
         DALI_R_COM_OK = false
         DALI_R_COM_ERROR_CNT_DEVICE = DALI_R_COM_ERROR_CNT_DEVICE + 1
      else
         DALI_QueryOk = 1 //Query was answered by device
         DALI_R_COM_OK = true
      endif
    else
      DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
      DALI_R_COM_OK = false
      DALI R COM ERROR CNT UNIT = DALI R COM ERROR CNT UNIT + 1
    endif
  else
    DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI R COM OK = false
    DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
  endif
  if DALI QueryOk != 1 then
    LOG Message("DALI Query  -  NOK  Adr:"+String.FormatInt(Adr,2)+"  Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  else
    LOG Message("DALI Query  -  OK  Adr:"+String.FormatInt(Adr,2)+"  Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  endif
  return DALI_QueryOk
endfunction
//-----------------------------------------------------------------------------------------------
function int DALI_QuerYesNo(num Handle,int Unit,int Adr,int Cmd)
  bit lComOk
  DALI TempValue = Cmd + (Adr<<8)
  if ModbusMaster.WriteSignedInt16(Handle, Unit, MDB_REG_KRZ_EXT_CMD,DALI_TempValue)then
    Me.Sleep(50)
    lComOk= ModbusMaster.ReadUnSignedInt16(Handle,Unit, MDB REG KRZ EXT CMD, DALI QueryValue)
    if lComOk then
      if DALI_QueryValue == 0x8000 then
         DALI_QueryOk = 0 //No answer from device
         DALI R COM OK = false
         DALI_R_COM_ERROR_CNT_DEVICE = DALI_R_COM_ERROR_CNT_DEVICE + 1
      else
         DALI QueryOk = 1 //Query was answered by device
         DALI R COM OK = true
      endif
    else
      DALI CmdOk = -1 //No answer from RESI-DALI-MODBUS
      DALI_R_COM_OK = false
      DALI_R_COM_ERROR_CNT_UNIT = DALI_R_COM_ERROR_CNT_UNIT + 1
    endif
  else
    DALI_CmdOk = -1 //No answer from RESI-DALI-MODBUS
    DALI_R_COM_OK = false
    DALI R COM ERROR CNT UNIT = DALI R COM ERROR CNT UNIT + 1
  endif
  if DALI_QueryOk != 1 then
    LOG_Message("DALI_QueryYesNo           -           NOK           Adr:"+String.FormatInt(Adr,2)+"
Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  else
    LOG Message("DALI QueryYesNo           -           OK           Adr:"+String.FormatInt(Adr,2)+"
Cmd:"+String.FormatInt(Cmd,4)+"
Val:"+String.FormatInt(DALI_QueryValue,4)+"["+String.FormatHex(DALI_QueryValue,4)+"]")
  endif
  return DALI QueryOk
endfunction
//============================================================
// QUERY-Functions
//============================================================
//intconst CMD_QUERY_STATUS              = 0x0090 //144
function int DALI_QueryStatus(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_STATUS)
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
endfunction

//intconst CMD_QUERY_IS_WORKING          = 0x0091 //145
function int DALI_QueryIsWorking(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_IS_WORKING)
endfunction

//intconst CMD_QUERY_IS_FAILURE          = 0x0092 //146
function int DALI_QueryIsFailure(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_IS_FAILURE)
endfunction

//intconst CMD_QUERY_IS_OPERATING        = 0x0093 //147
function int DALI_QueryIsOperating(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_IS_OPERATING)
endfunction

//intconst CMD_QUERY_OUT_OF_LIMIT        = 0x0094 //148
function int DALI_QueryOutOfLimit(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_OUT_OF_LIMIT)
endfunction

//intconst CMD_QUERY_RESET_STATE         = 0x0095 //149
function int DALI_QueryResetState(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_RESET_STATE)
endfunction

//intconst CMD_QUERY_MISSING_SHORTADRESS = 0x0096 //150
function int DALI_QueryMissingShortadress(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_MISSING_SHORTADRESS)
endfunction

//intconst CMD_QUERY_VERSION_NUMBER      = 0x0097 //151
function int DALI_QueryVersionNumber(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_VERSION_NUMBER)
endfunction

//intconst CMD_QUERY_DTR_CONTENT         = 0x0098 //152
function int DALI_QueryDtrContent(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_DTR_CONTENT)
endfunction

//intconst CMD_QUERY_DEVICE_TYP          = 0x0099 //153
function int DALI_QueryDeviceTyp(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_DEVICE_TYP)
endfunction

//intconst CMD_QUERY_PHYSICAL_MINIMUM    = 0x009A //154
function int DALI_QueryPhysicalMinimum(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_PHYSICAL_MINIMUM)
endfunction

//intconst CMD_QUERY_POWER_FAILURE_MODE  = 0x009B //155
function int DALI_QueryPowerFailureMode(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_POWER_FAILURE_MODE)
endfunction

//intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
function int DALI_QueryCurrentLevel(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_CURRENT_LEVEL)
endfunction

//intconst CMD_QUERY_MAX_LEVEL            = 0x00A1 //161
function int DALI_QueryMaxLevel(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_MAX_LEVEL)
endfunction

//intconst CMD_QUERY_MIN_LEVEL            = 0x00A2 //162
function int DALI_QueryMinLevel(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_MIN_LEVEL)
endfunction

//intconst CMD_QUERY_POWER_UP_LEVEL       = 0x00A3 //163
function int DALI_QueryPowerUpLevel(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_POWER_UP_LEVEL)
endfunction

//intconst CMD_QUERY_SYSTEM_FAILURE_LEVEL = 0x00A4 //164
function int DALI_QuerySystemFailureLevel(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_SYSTEM_FAILURE_LEVEL)
endfunction

//intconst CMD_QUERY_FADE_TIME_AND_RATE   = 0x00A5 //165
function int DALI_QueryFadeTimeAndRate(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_FADE_TIME_AND_RATE)
endfunction
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
//intconst CMD_QUERY_LEVEL_OF_SCENE_xx   = 0x00B0- 0x00BF //176-191
function int DALI_QueryLevelOfScene(num Handle,int Unit,int Adr, int Scene)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_LEVEL_OF_SCENE_00 + (Scene & 0x000f))
endfunction

//intconst CMD_QUERY_PART_OF_GROUP_00_07  = 0x00C0 //192
function int DALI QueryPartOfGroup 00 07(num Handle,int Unit,int Adr)
  return DALI Query(Handle,Unit,Adr,CMD QUERY PART OF GROUP 00 07)
endfunction
//intconst CMD_QUERY_PART_OF_GROUP_08_15  = 0x00C1 //193
function int DALI QueryPartOfGroup 08 15(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_PART_OF_GROUP_08_15)
endfunction

function int DALI QueryPartOfGroup(num Handle,int Unit,int Adr)
  DALI_TempValue = DALI_Query(Handle,Unit,Adr,CMD_QUERY_PART_OF_GROUP_00_07)
  if DALI_TempValue >0 then
    DALI R PART OF GROUP 00 07 = DALI QueryValue
  else
    return DALI_TempValue
  endif
  DALI TempValue = DALI Query(Handle,Unit,Adr,CMD QUERY PART OF GROUP 08 15)
  if DALI_TempValue >0 then
    DALI_R_PART_OF_GROUP_08_15 = DALI_QueryValue
  else
    return DALI TempValue
  endif
  DALI_QueryValue = DALI_R_PART_OF_GROUP_00_07 + (DALI_R_PART_OF_GROUP_08_15<<8)
  return 1
endfunction


//intconst CMD QUERY RANDOMADRESS HIGH    = 0x00C2 //194
function int DALI_QueryRandomAdressHigh(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_RANDOMADRESS_HIGH)
endfunction
//intconst CMD QUERY RANDOMADRESS MID     = 0x00C3 //195
function int DALI_QueryRandomAdressMid(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_RANDOMADRESS_MID)
endfunction
//intconst CMD_QUERY_RANDOMADRESS_LOW     = 0x00C4 //196
function int DALI_QueryRandomAdressLow(num Handle,int Unit,int Adr)
  return DALI_Query(Handle,Unit,Adr,CMD_QUERY_RANDOMADRESS_LOW)
endfunction

function int DALI_QueryRandomAdress(num Handle,int Unit,int Adr)
  DALI TempValue = DALI Query(Handle,Unit,Adr,CMD QUERY RANDOMADRESS HIGH)
  if DALI TempValue >0 then
    DALI_R_RANDOMADRESS_HIGH = DALI_QueryValue
  else
    return DALI TempValue
  endif
  DALI_TempValue = DALI_Query(Handle,Unit,Adr,CMD_QUERY_RANDOMADRESS_MID)
  if DALI_TempValue >0 then
    DALI R RANDOMADRESS MID = DALI QueryValue
  else
    return DALI_TempValue
  endif
  DALI TempValue = DALI Query(Handle,Unit,Adr,CMD QUERY RANDOMADRESS LOW)
  if DALI_TempValue >0 then
    DALI_R_RANDOMADRESS_LOW = DALI_QueryValue
  else
    return DALI_TempValue
  endif
  DALI QueryValue      =      DALI R RANDOMADRESS LOW      +      (DALI R RANDOMADRESS MID<<8)      +
(DALI R RANDOMADRESS HIGH<<16)
  return 1
endfunction

//============================================================
// SET-Functions
//============================================================
//intconst CMD EXTINGUISH POWER = 0x0000 //0
function int DALI_ExtinguishPower(num Handle,int Unit,int Adr)
  return DALI_Send_Cmd_1F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_EXTINGUISH_POWER)
endfunction

//intconst CMD_DIMM_UP_POWER_200_MS        = 0x0001 //1
function int DALI_DimmUpPower200Ms(num Handle,int Unit,int Adr)
  return DALI Send Cmd 1F(DALI Handle,DALI P UNIT,DALI P ADR,CMD DIMM UP POWER 200 MS)
endfunction
//intconst CMD_DIMM_DOWN_POWER_200_MS      = 0x0002 //2
function int DALI_DimmDownPower200Ms(num Handle,int Unit,int Adr)
  return DALI Send Cmd 1F(DALI Handle,DALI P UNIT,DALI P ADR,CMD DIMM DOWN POWER 200 MS)
endfunction
//intconst CMD_STEP_UP_POWER               = 0x0003 //3
function int DALI_StepUpPower(num Handle,int Unit,int Adr)
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  return DALI_Send_Cmd_1F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STEP_UP_POWER)
endfunction
//intconst CMD_STEP_DOWN_POWER             = 0x0004 //4
function int DALI StepDownPower(num Handle,int Unit,int Adr)
  return DALI_Send_Cmd_1F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STEP_DOWN_POWER)
endfunction
//intconst CMD SET POWER TO MAXIMUM = 0x0005 //5
function int DALI SetPowerToMaximum(num Handle,int Unit,int Adr)
  return DALI_Send_Cmd_1F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_SET_POWER_TO_MAXIMUM)
endfunction
//intconst CMD SET POWER TO MINIMUM        = 0x0006 //6
function int DALI_SetPowerToMinimum(num Handle,int Unit,int Adr)
  return DALI_Send_Cmd_1F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_SET_POWER_TO_MINIMUM)
endfunction
//intconst CMD STEP DOWN POWER AND SWITCH  = 0x0007 //7
function int DALI_StepDownPowerAndSwitch(num Handle,int Unit,int Adr)
  return DALI_Send_Cmd_1F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STEP_DOWN_POWER_AND_SWITCH)
endfunction
//intconst CMD STEP UP POWER AND SWITCH    = 0x0008 //8
function int DALI_StepUpPowerAndSwitch(num Handle,int Unit,int Adr)
  return DALI_Send_Cmd_1F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STEP_UP_POWER_AND_SWITCH)
endfunction

//intconst CMD_SET_POWER_TO_SCENE_xx_LEVEL = 0x0010 - 0x001F //16-31
function int DALI SetPowerToSceneLevel(num Handle,int Unit,int Adr,int Scene)
  return    DALI Send Cmd 1F(DALI Handle,DALI P UNIT,DALI P ADR,CMD SET POWER TO SCENE 00 LEVEL    +
(Scene & 0x0f))
endfunction

//============================================================
// CONFIG-Functions
//============================================================

//intconst CMD_SET_DEFAULT_SETTINGS       = 0x0020 //32
function int DALI_SetDefaultSettings(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(DALI Handle,DALI P UNIT,DALI P ADR,CMD SET DEFAULT SETTINGS,CMD DEVICE TYPE 0)
endfunction
//intconst CMD_STORE_CURRENT_LEVEL_IN_DTR = 0x0021 //33
function int DALI StoreCurrentLevelInDtr(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STORE_CURRENT_LEVEL_IN_DTR,CMD_DEVICE_TYPE_
0)
endfunction

//intconst CMD_STORE_DTR_AS_MAX_LEVEL             = 0x002A //42
function int DALI StoreDtrAsMaxLevel(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STORE_DTR_AS_MAX_LEVEL,CMD_DEVICE_TYPE_0)
endfunction
//intconst CMD STORE DTR AS MIN LEVEL             = 0x002B //43
function int DALI_StoreDtrAsMinLevel(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STORE_DTR_AS_MIN_LEVEL,CMD_DEVICE_TYPE_0)
endfunction
//intconst CMD_STORE_DTR_AS_SYSTEM_FAILURE_LEVEL  = 0x002C //44
function int DALI_StoreDtrAsSystemFailureLevel(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(DALI Handle,DALI P UNIT,DALI P ADR,CMD STORE DTR AS SYSTEM FAILURE LEVEL,CMD DEVIC
E_TYPE_0)
endfunction
//intconst CMD STORE DTR AS POWER ON LEVEL        = 0x002D //45
function int DALI_StoreDtrAsPowerOnLevel(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(DALI Handle,DALI P UNIT,DALI P ADR,CMD STORE DTR AS POWER ON LEVEL,CMD DEVICE TYPE
 0)
endfunction

//intconst CMD STORE DTR AS FADE TIME  = 0x002E //46
function int DALI_StoreDtrAsFadeTime(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STORE_DTR_AS_FADE_TIME,CMD_DEVICE_TYPE_0)
endfunction
//intconst CMD_STORE_DTR_AS_FADE_RATE  = 0x002F //47
function int DALI_StoreDtrAsFadeRate(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(DALI Handle,DALI P UNIT,DALI P ADR,CMD STORE DTR AS FADE RATE,CMD DEVICE TYPE 0)
endfunction

//intconst CMD STORE DTR AS SCENE xx  = 0x0040-0x004f //64-79
function int DALI_StoreDtrAsScene(num Handle,int Unit,int Adr,int Scene)
  return  DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STORE_DTR_AS_SCENE_00  +  (Scene  &
0x0f),CMD_DEVICE_TYPE_0)
endfunction

//intconst CMD_REMOVE_FROM_SCENE_xx  = 0x0050-0x005f //80-95
function int DALI_RemoveFromScene(num Handle,int Unit,int Adr,int Scene)
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  return  DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_REMOVE_FROM_SCENE_00  +  (Scene  &
0x0f),CMD_DEVICE_TYPE_0)
endfunction

//intconst CMD_ADD_TO_GROUP_xx  = 0x0060-0x006F //96-111
function int DALI_AddToGroup(num Handle,int Unit,int Adr,int Group)
  return   DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_ADD_TO_GROUP_00   +   (Group   &
0x0f),CMD_DEVICE_TYPE_0)
endfunction

//intconst CMD_REMOVE_FROM_GROUP_xx  = 0x0070-0x007f //112-127
function int DALI_RemoveFromGroup(num Handle,int Unit,int Adr,int Group)
  return  DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_REMOVE_FROM_GROUP_00  +  (Group  &
0x0f),CMD_DEVICE_TYPE_0)
endfunction

//intconst CMD_STORE_DTR_AS_SHORTADRESS  = 0x0080 //128
function int DALI_StoreDtrAsShortadress(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,CMD_STORE_DTR_AS_SHORTADRESS,CMD_DEVICE_TYPE_0)
endfunction
//----------------------
function num DALI_CalcPercent(int Level)
  num val,res
  val = Cvt.Int2Num(Level)
  res = Math.Pow(10.,((val-1.)/(253./3.))-1.)
  return res
endfunction
function num DALI_CalcLevel(int Percent)
  num val,res
  val = Cvt.Int2Num(Percent)
  res = (Math.Log10(val)+1.)*(253./3.)+1.
  return res
endfunction
//
sub DALI_OnWrite_P_CMD(string VarName,int Wert)
  int TempInt
   if Wert == 0  then //intconst CMD_EXTINGUISH_POWER = 0x0000 //0
     TempInt = DALI_ExtinguishPower(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 1  then //intconst CMD_DIMM_UP_POWER_200_MS        = 0x0001 //1
     TempInt = DALI_DimmUpPower200Ms(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 2  then //intconst CMD_DIMM_DOWN_POWER_200_MS      = 0x0002 //2
     TempInt = DALI_DimmDownPower200Ms(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 3  then //intconst CMD_STEP_UP_POWER               = 0x0003 //3
     TempInt = DALI_StepUpPower(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 4  then //intconst CMD_STEP_DOWN_POWER             = 0x0004 //4
     TempInt = DALI_StepDownPower(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 5  then //intconst CMD_SET_POWER_TO_MAXIMUM = 0x0005 //5
     TempInt = DALI_SetPowerToMaximum(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 6  then //intconst CMD_SET_POWER_TO_MINIMUM         = 0x0006 //6
     TempInt = DALI_SetPowerToMinimum(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 7  then //intconst CMD_STEP_DOWN_POWER_AND_SWITCH  = 0x0007 //7
     TempInt = DALI_StepDownPowerAndSwitch(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   elseif Wert == 8  then //intconst CMD_STEP_UP_POWER_AND_SWITCH    = 0x0008 //8
     TempInt = DALI_StepUpPowerAndSwitch(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   endif

   if Wert == 16  then //intconst CMD_SET_POWER_TO_SCENE_00_LEVEL = 0x0010 //16
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,0)
   elseif Wert == 17  then //intconst CMD_SET_POWER_TO_SCENE_01_LEVEL = 0x0011 //17
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
   elseif Wert == 18  then //intconst CMD_SET_POWER_TO_SCENE_02_LEVEL = 0x0012 //18
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,2)
   elseif Wert == 19  then //intconst CMD_SET_POWER_TO_SCENE_03_LEVEL = 0x0013 //19
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,3)
   elseif Wert == 20  then //intconst CMD_SET_POWER_TO_SCENE_04_LEVEL = 0x0014 //20
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,4)
   elseif Wert == 21  then //intconst CMD_SET_POWER_TO_SCENE_05_LEVEL = 0x0015 //21
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,5)
   elseif Wert == 22  then //intconst CMD_SET_POWER_TO_SCENE_06_LEVEL = 0x0016 //22
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,6)
   elseif Wert == 23  then //intconst CMD_SET_POWER_TO_SCENE_07_LEVEL = 0x0017 //23
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,7)
   elseif Wert == 24  then //intconst CMD_SET_POWER_TO_SCENE_08_LEVEL = 0x0018 //24
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
   elseif Wert == 25  then //intconst CMD_SET_POWER_TO_SCENE_09_LEVEL = 0x0019 //25
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,9)
   elseif Wert == 26  then //intconst CMD_SET_POWER_TO_SCENE_10_LEVEL = 0x001A //26
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
   elseif Wert == 27  then //intconst CMD_SET_POWER_TO_SCENE_11_LEVEL = 0x001B //27
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,11)
   elseif Wert == 28  then //intconst CMD_SET_POWER_TO_SCENE_12_LEVEL = 0x001C //28
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,12)
   elseif Wert == 29  then //intconst CMD_SET_POWER_TO_SCENE_13_LEVEL = 0x001D //29
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,13)
   elseif Wert == 30  then //intconst CMD_SET_POWER_TO_SCENE_14_LEVEL = 0x001E //30
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,14)
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
   elseif Wert == 31  then //intconst CMD_SET_POWER_TO_SCENE_15_LEVEL = 0x001F //31
     TempInt = DALI_SetPowerToSceneLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,15)
   endif
//
   if Wert == 32  then //intconst CMD_SET_DEFAULT_SETTINGS = 0x0020 //32
     TempInt = DALI_SetDefaultSettings(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   endif
//
  if Wert == 33  then //intconst CMD_STORE_CURRENT_LEVEL_IN_DTR = 0x0021 //33
     TempInt = DALI_StoreCurrentLevelInDtr(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 42  then //intconst CMD_STORE_DTR_AS_MAX_LEVEL             = 0x002A //42
     TempInt = DALI_StoreDtrAsMaxLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 43  then //intconst CMD_STORE_DTR_AS_MIN_LEVEL             = 0x002B //43
     TempInt = DALI_StoreDtrAsMinLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 44  then //intconst CMD_STORE_DTR_AS_SYSTEM_FAILURE_LEVEL  = 0x002C //44
     TempInt = DALI_StoreDtrAsSystemFailureLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 45  then //intconst CMD_STORE_DTR_AS_POWER_ON_LEVEL        = 0x002D //45
     TempInt = DALI_StoreDtrAsPowerOnLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 46  then //intconst CMD_STORE_DTR_AS_FADE_TIME  = 0x002E //46
     TempInt = DALI_StoreDtrAsFadeTime(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 47  then //intconst CMD_STORE_DTR_AS_FADE_RATE  = 0x002F //47
     TempInt = DALI_StoreDtrAsFadeRate(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  endif

  if Wert == 64  then //intconst CMD_STORE_DTR_AS_SCENE_00  = 0x0040 //64
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,0)
  elseif Wert == 65  then //intconst CMD_STORE_DTR_AS_SCENE_01  = 0x0041 //65
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
  elseif Wert == 66  then //intconst CMD_STORE_DTR_AS_SCENE_02  = 0x0042 //66
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,2)
  elseif Wert == 67  then //intconst CMD_STORE_DTR_AS_SCENE_03  = 0x0043 //67
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,3)
  elseif Wert == 68  then //intconst CMD_STORE_DTR_AS_SCENE_04  = 0x0044 //68
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,4)
  elseif Wert == 69  then //intconst CMD_STORE_DTR_AS_SCENE_05  = 0x0045 //69
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,5)
  elseif Wert == 70  then //intconst CMD_STORE_DTR_AS_SCENE_06  = 0x0046 //70
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,6)
  elseif Wert == 71  then //intconst CMD_STORE_DTR_AS_SCENE_07  = 0x0047 //71
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,7)
  elseif Wert == 72  then //intconst CMD_STORE_DTR_AS_SCENE_08  = 0x0048 //72
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
  elseif Wert == 73  then //intconst CMD_STORE_DTR_AS_SCENE_09  = 0x0049 //73
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,9)
  elseif Wert == 74  then //intconst CMD_STORE_DTR_AS_SCENE_10  = 0x004A //74
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
  elseif Wert == 75  then //intconst CMD_STORE_DTR_AS_SCENE_11  = 0x004B //75
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,11)
  elseif Wert == 76  then //intconst CMD_STORE_DTR_AS_SCENE_12  = 0x004C //76
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,12)
  elseif Wert == 77  then //intconst CMD_STORE_DTR_AS_SCENE_13  = 0x004D //77
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,13)
  elseif Wert == 78  then //intconst CMD_STORE_DTR_AS_SCENE_14  = 0x004E //78
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,14)
  elseif Wert == 79  then //intconst CMD_STORE_DTR_AS_SCENE_15  = 0x004F //79
     TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,15)
  endif

  if Wert == 80  then //intconst CMD_REMOVE_FROM_SCENE_00  = 0x0050 //80
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,0)
  elseif Wert == 81  then //intconst CMD_REMOVE_FROM_SCENE_01  = 0x0051 //81
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
  elseif Wert == 82  then //intconst CMD_REMOVE_FROM_SCENE_02  = 0x0052 //82
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,2)
  elseif Wert == 83  then //intconst CMD_REMOVE_FROM_SCENE_03  = 0x0053 //83
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,3)
  elseif Wert == 84  then //intconst CMD_REMOVE_FROM_SCENE_04  = 0x0054 //84
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,4)
  elseif Wert == 85  then //intconst CMD_REMOVE_FROM_SCENE_05  = 0x0055 //85
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,5)
  elseif Wert == 86  then //intconst CMD_REMOVE_FROM_SCENE_06  = 0x0056 //86
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,6)
  elseif Wert == 87  then //intconst CMD_REMOVE_FROM_SCENE_07  = 0x0057 //87
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,7)
  elseif Wert == 88  then //intconst CMD_REMOVE_FROM_SCENE_08  = 0x0058 //88
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
  elseif Wert == 89  then //intconst CMD_REMOVE_FROM_SCENE_09  = 0x0059 //89
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,9)
  elseif Wert == 90  then //intconst CMD_REMOVE_FROM_SCENE_10  = 0x005A //90
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
  elseif Wert == 91  then //intconst CMD_REMOVE_FROM_SCENE_11  = 0x005B //91
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,11)
  elseif Wert == 92  then //intconst CMD_REMOVE_FROM_SCENE_12  = 0x005C //92
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,12)
  elseif Wert == 93  then //intconst CMD_REMOVE_FROM_SCENE_13  = 0x005D //93
     TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,13)
  elseif Wert == 94  then //intconst CMD_REMOVE_FROM_SCENE_14  = 0x005E //94
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
   TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,14)
elseif Wert == 95  then //intconst CMD_REMOVE_FROM_SCENE_15  = 0x005F //95
   TempInt = DALI_RemoveFromScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,15)
endif

if Wert == 96  then //intconst CMD_ADD_TO_GROUP_00  = 0x0060 //96
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,0)
elseif Wert == 97  then //intconst CMD ADD TO GROUP 01  = 0x0061 //97
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
elseif Wert == 98  then //intconst CMD_ADD_TO_GROUP_02  = 0x0062 //98
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,2)
elseif Wert == 99  then //intconst CMD_ADD_TO_GROUP_03  = 0x0063 //99
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,3)
elseif Wert == 100  then //intconst CMD_ADD_TO_GROUP_04  = 0x0064 //100
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,4)
elseif Wert == 101  then //intconst CMD_ADD_TO_GROUP_05  = 0x0065 //101
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,5)
elseif Wert == 102  then //intconst CMD ADD TO GROUP 06  = 0x0066 //102
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,6)
elseif Wert == 103  then //intconst CMD_ADD_TO_GROUP_07  = 0x0067 //103
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,7)
elseif Wert == 104  then //intconst CMD ADD TO GROUP 08  = 0x0068 //104
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
elseif Wert == 105  then //intconst CMD_ADD_TO_GROUP_09  = 0x0069 //105
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,9)
elseif Wert == 106  then //intconst CMD ADD TO GROUP 10  = 0x006A //106
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
elseif Wert == 107  then //intconst CMD_ADD_TO_GROUP_11  = 0x006B //107
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,11)
elseif Wert == 108  then //intconst CMD ADD TO GROUP 12  = 0x006C //108
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,12)
elseif Wert == 109  then //intconst CMD_ADD_TO_GROUP_13  = 0x006D //109
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,13)
elseif Wert == 110  then //intconst CMD_ADD_TO_GROUP_14  = 0x006E //110
   TempInt = DALI_AddToGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,14)
elseif Wert == 111  then //intconst CMD ADD TO GROUP 15  = 0x006F //111
   TempInt = DALI AddToGroup(DALI Handle,DALI P UNIT,DALI P ADR,15)
endif

if Wert == 112  then //intconst CMD REMOVE FROM GROUP 00  = 0x0070 //112
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,0)
elseif Wert == 113  then //intconst CMD_REMOVE_FROM_GROUP_01  = 0x0071 //113
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
elseif Wert == 114  then //intconst CMD REMOVE FROM GROUP 02  = 0x0072 //114
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,2)
elseif Wert == 115  then //intconst CMD_REMOVE_FROM_GROUP_03  = 0x0073 //115
   TempInt = DALI RemoveFromGroup(DALI Handle,DALI P UNIT,DALI P ADR,3)
elseif Wert == 116  then //intconst CMD REMOVE FROM GROUP 04  = 0x0074 //116
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,4)
elseif Wert == 117  then //intconst CMD_REMOVE_FROM_GROUP_05  = 0x0075 //117
   TempInt = DALI RemoveFromGroup(DALI Handle,DALI P UNIT,DALI P ADR,5)
elseif Wert == 118  then //intconst CMD_REMOVE_FROM_GROUP_06  = 0x0076 //118
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,6)
elseif Wert == 119  then //intconst CMD_REMOVE_FROM_GROUP_07  = 0x0077 //119
   TempInt = DALI RemoveFromGroup(DALI Handle,DALI P UNIT,DALI P ADR,7)
elseif Wert == 120  then //intconst CMD_REMOVE_FROM_GROUP_08  = 0x0078 //120
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
elseif Wert == 121  then //intconst CMD REMOVE FROM GROUP 09  = 0x0079 //121
   TempInt = DALI RemoveFromGroup(DALI Handle,DALI P UNIT,DALI P ADR,9)
elseif Wert == 122  then //intconst CMD_REMOVE_FROM_GROUP_10  = 0x007A //122
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
elseif Wert == 123  then //intconst CMD REMOVE FROM GROUP 11  = 0x007B //123
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,11)
elseif Wert == 124  then //intconst CMD_REMOVE_FROM_GROUP_12  = 0x007C //124
   TempInt = DALI RemoveFromGroup(DALI Handle,DALI P UNIT,DALI P ADR,12)
elseif Wert == 125  then //intconst CMD REMOVE FROM GROUP 13  = 0x007D //125
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,13)
elseif Wert == 126  then //intconst CMD_REMOVE_FROM_GROUP_14  = 0x007E //126
   TempInt = DALI RemoveFromGroup(DALI Handle,DALI P UNIT,DALI P ADR,14)
elseif Wert == 127  then //intconst CMD_REMOVE_FROM_GROUP_15  = 0x007F //127
   TempInt = DALI_RemoveFromGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,15)
endif

if Wert == 128  then //intconst CMD_STORE_DTR_AS_SHORTADRESS  = 0x0080 //128
   TempInt = DALI_StoreDtrAsShortadress(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
endif

if Wert == 144 then //intconst CMD_QUERY_STATUS              = 0x0090 //144
  DALI_TempValue = DALI_QueryStatus(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI TempValue > 0 then
     DALI_R_STATUS = DALI_QueryValue
     DALI_R_STATUS_CONTROL_GEAR = Cvt.Int2Bit(DALI_R_STATUS & 0x01)
     DALI_R_STATUS_FAILURE = Cvt.Int2Bit(DALI_R_STATUS & 0x02)
     DALI R STATUS ARC POWER = Cvt.Int2Bit(DALI R STATUS & 0x04)
     DALI_R_STATUS_LIMIT_ERROR = Cvt.Int2Bit(DALI_R_STATUS & 0x08)
     DALI_R_STATUS_FADE_RUNNING = Cvt.Int2Bit(DALI_R_STATUS & 0x10)
     DALI_R_STATUS_RESET_STATE = Cvt.Int2Bit(DALI_R_STATUS & 0x20)
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
      DALI_R_STATUS_MISSING_SHORTADRESS = Cvt.Int2Bit(DALI_R_STATUS & 0x40)
      DALI_R_STATUS_POWER_FAILURE = Cvt.Int2Bit(DALI_R_STATUS & 0x80)
    endif
  elseif Wert == 145 then //intconst CMD QUERY IS WORKING          = 0x0091 //145
    DALI_TempValue = DALI_QueryIsWorking(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI R IS WORKING = DALI QueryValue
    endif
  elseif Wert == 146 then //intconst CMD_QUERY_IS_FAILURE          = 0x0092 //146
    DALI_TempValue = DALI_QueryIsFailure(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
      DALI_R_IS_FAILURE = DALI_QueryValue
    endif
  elseif Wert == 147 then //intconst CMD_QUERY_IS_OPERATING        = 0x0093 //147
    DALI TempValue = DALI QueryIsOperating(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
       DALI_R_IS_OPERATING = DALI_QueryValue
    endif
  elseif Wert == 148 then //intconst CMD QUERY OUT OF LIMIT        = 0x0094 //148
    DALI_TempValue = DALI_QueryOutOfLimit(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI R OUT OF LIMIT = DALI QueryValue
    endif
  elseif Wert == 149 then //intconst CMD_QUERY_RESET_STATE          = 0x0095 //149
    DALI TempValue = DALI QueryResetState(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI TempValue > 0 then
       DALI_R_RESET_STATE = DALI_QueryValue
    endif
  elseif Wert == 150 then//intconst CMD QUERY MISSING SHORTADRESS = 0x0096 //150
    DALI TempValue = DALI QueryMissingShortadress(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
       DALI_R_MISSING_SHORTADRESS = DALI_QueryValue
    endif
  endif

  if Wert == 151 then //intconst CMD QUERY VERSION NUMBER      = 0x0097 //151
    DALI TempValue = DALI QueryVersionNumber(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
       DALI_R_VERSION_NUMBER = DALI_QueryValue
    endif
  elseif Wert == 152 then //intconst CMD_QUERY_DTR_CONTENT          = 0x0098 //152
    DALI_TempValue = DALI_QueryDtrContent(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI R DTR CONTENT = DALI QueryValue
    endif
  elseif Wert == 153 then //intconst CMD_QUERY_DEVICE_TYP           = 0x0099 //153
    DALI TempValue = DALI QueryDeviceTyp(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI TempValue > 0 then
       DALI_R_DEVICE_TYP = DALI_QueryValue
    endif
  elseif Wert == 154 then //intconst CMD QUERY PHYSICAL MINIMUM    = 0x009A //154
    DALI_TempValue = DALI_QueryPhysicalMinimum(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI_R_PHYSICAL_MINIMUM = DALI_QueryValue
    endif
  elseif Wert == 155 then //intconst CMD_QUERY_POWER_FAILURE_MODE  = 0x009B //155
    DALI_TempValue = DALI_QueryPowerFailureMode(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI TempValue > 0 then
       DALI R POWER FAILURE MODE = DALI QueryValue
    endif
  elseif Wert == 160 then //intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
    DALI TempValue = DALI QueryCurrentLevel(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
       DALI_R_CURRENT_LEVEL = DALI_QueryValue
       DALI R CURRENT LEVEL 100=DALI CalcPercent(DALI R CURRENT LEVEL)
    endif
  elseif Wert == 161 then //intconst CMD_QUERY_MAX_LEVEL            = 0x00A1 //161
    DALI_TempValue = DALI_QueryMaxLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI TempValue > 0 then
       DALI_R_MAX_LEVEL = DALI_QueryValue
    endif
  elseif Wert == 162 then //intconst CMD_QUERY_MIN_LEVEL            = 0x00A2 //162
    DALI TempValue = DALI QueryMinLevel(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
       DALI_R_MIN_LEVEL = DALI_QueryValue
    endif
  elseif Wert == 163 then //intconst CMD QUERY POWER UP LEVEL       = 0x00A3 //163
    DALI_TempValue = DALI_QueryPowerUpLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI R POWER UP LEVEL = DALI QueryValue
    endif
  elseif Wert == 164 then //intconst CMD_QUERY_SYSTEM_FAILURE_LEVEL = 0x00A4 //164
    DALI_TempValue = DALI_QuerySystemFailureLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI TempValue > 0 then
       DALI_R_SYSTEM_FAILURE_LEVEL = DALI_QueryValue
    endif
  elseif Wert == 165 then //intconst CMD_QUERY_FADE_TIME_AND_RATE   = 0x00A5 //165
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    DALI_TempValue = DALI_QueryFadeTimeAndRate(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI_R_FADE_TIME_AND_RATE = DALI_QueryValue
       DALI_R_FADE_TIME = DALI_QueryValue >> 4
       DALI_R_FADE_RATE = DALI_QueryValue & 0x0f
    endif
  endif

  if Wert == 176 then //intconst CMD_QUERY_LEVEL_OF_SCENE_00    = 0x00B0 //176
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,0)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_00 = DALI_QueryValue
    endif
  elseif Wert == 177 then //intconst CMD_QUERY_LEVEL_OF_SCENE_01    = 0x00B1 //177
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_01 = DALI_QueryValue
    endif
  elseif Wert == 178 then //intconst CMD_QUERY_LEVEL_OF_SCENE_02    = 0x00B2 //178
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,2)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_02 = DALI_QueryValue
    endif
  elseif Wert == 179 then //intconst CMD_QUERY_LEVEL_OF_SCENE_03    = 0x00B3 //179
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,3)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_03 = DALI_QueryValue
    endif
  elseif Wert == 180 then //intconst CMD_QUERY_LEVEL_OF_SCENE_04    = 0x00B4 //180
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,4)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_04 = DALI_QueryValue
    endif
  elseif Wert == 181 then //intconst CMD_QUERY_LEVEL_OF_SCENE_05    = 0x00B5 //181
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,5)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_05 = DALI_QueryValue
    endif
  elseif Wert == 182 then //intconst CMD_QUERY_LEVEL_OF_SCENE_06    = 0x00B6 //182
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,6)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_06 = DALI_QueryValue
    endif
  elseif Wert == 183 then //intconst CMD_QUERY_LEVEL_OF_SCENE_07    = 0x00B7 //183
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,7)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_07 = DALI_QueryValue
    endif
  elseif Wert == 184 then //intconst CMD_QUERY_LEVEL_OF_SCENE_08    = 0x00B8 //184
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_08 = DALI_QueryValue
    endif
  elseif Wert == 185 then //intconst CMD_QUERY_LEVEL_OF_SCENE_09    = 0x00B9 //185
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,9)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_09 = DALI_QueryValue
    endif
  elseif Wert == 186 then //intconst CMD_QUERY_LEVEL_OF_SCENE_10    = 0x00BA //186
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_10 = DALI_QueryValue
    endif
  elseif Wert == 187 then //intconst CMD_QUERY_LEVEL_OF_SCENE_11    = 0x00BB //187
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,11)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_11 = DALI_QueryValue
    endif
  elseif Wert == 188 then //intconst CMD_QUERY_LEVEL_OF_SCENE_12    = 0x00BC //188
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,12)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_12 = DALI_QueryValue
    endif
  elseif Wert == 189 then //intconst CMD_QUERY_LEVEL_OF_SCENE_13    = 0x00BD //189
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,13)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_13 = DALI_QueryValue
    endif
  elseif Wert == 190 then //intconst CMD_QUERY_LEVEL_OF_SCENE_14    = 0x00BE //190
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,14)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_14 = DALI_QueryValue
    endif
  elseif Wert == 191 then //intconst CMD_QUERY_LEVEL_OF_SCENE_15    = 0x00BF //191
    DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,15)
    if DALI_TempValue > 0 then
       DALI_R_LEVEL_OF_SCENE_15 = DALI_QueryValue
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  endif
endif

if Wert == 192 then //intconst CMD QUERY PART OF GROUP 00 07  = 0x00C0 //192
  DALI_TempValue = DALI_QueryPartOfGroup_00_07(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
      DALI R PART OF GROUP 00 07 = DALI QueryValue
  endif
elseif Wert == 193 then //intconst CMD_QUERY_PART_OF_GROUP_08_15  = 0x00C1 //193
  DALI_TempValue = DALI_QueryPartOfGroup_08_15(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
      DALI_R_PART_OF_GROUP_08_15 = DALI_QueryValue
  endif
elseif Wert == 194 then //intconst CMD_QUERY_RANDOMADRESS_HIGH    = 0x00C2 //194
  DALI TempValue = DALI QueryRandomAdressHigh(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
      DALI_R_RANDOMADRESS_HIGH = DALI_QueryValue
  endif
elseif Wert == 195 then //intconst CMD QUERY RANDOMADRESS MID     = 0x00C3 //195
  DALI_TempValue = DALI_QueryRandomAdressMid(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
      DALI R RANDOMADRESS MID = DALI QueryValue
  endif
elseif Wert == 196 then //intconst CMD_QUERY_RANDOMADRESS_LOW     = 0x00C4 //196
  DALI TempValue = DALI QueryRandomAdressLow(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI TempValue > 0 then
      DALI_R_RANDOMADRESS_LOW = DALI_QueryValue
  endif
endif

if Wert == 257  then //intconst CMD_WRITE_DATA_TRANSFER_REGISTER    = 0xA300 //Cmd-257
   Params.SyncIn()
   TempInt = DALI Write DTR(DALI Handle,DALI P UNIT,DALI P CMD PARAM 01)
endif

if Wert == 128  then //intconst CMD STORE DTR AS SHORTADRESS  = 0x0080 //128
   TempInt = DALI StoreDtrAsShortadress(DALI Handle,DALI P UNIT,DALI P ADR)
elseif Wert == 0x10180 then
   Params.SyncIn()
   TempInt = DALI Write DTR(DALI Handle,DALI P UNIT,(DALI P CMD PARAM 01<<1)|0x01)
   TempInt = DALI_StoreDtrAsShortadress(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
endif


if Wert == 0x212A then
   TempInt = DALI_StoreCurrentLevelInDtr(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   TempInt = DALI StoreDtrAsMaxLevel(DALI Handle,DALI P UNIT,DALI P ADR)
elseif Wert == 0x212B then
   TempInt = DALI_StoreCurrentLevelInDtr(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   TempInt = DALI_StoreDtrAsMinLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
elseif Wert == 0x212C then
   TempInt = DALI_StoreCurrentLevelInDtr(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   TempInt = DALI_StoreDtrAsSystemFailureLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
elseif Wert == 0x212D then
   TempInt = DALI StoreCurrentLevelInDtr(DALI Handle,DALI P UNIT,DALI P ADR)
   TempInt = DALI_StoreDtrAsPowerOnLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
elseif Wert == 0x10012A then
   Params.SyncIn()
   TempInt = DALI Write DTR(DALI Handle,DALI P UNIT,DALI P CMD PARAM 01)
   TempInt = DALI_StoreDtrAsMaxLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
elseif Wert == 0x10012B then
   Params.SyncIn()
   TempInt = DALI_Write_DTR(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_01)
   TempInt = DALI_StoreDtrAsMinLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
elseif Wert == 0x10012C then
   Params.SyncIn()
   TempInt = DALI_Write_DTR(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_01)
   TempInt = DALI_StoreDtrAsSystemFailureLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
elseif Wert == 0x10012D then
   Params.SyncIn()
   TempInt = DALI_Write_DTR(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_01)
   TempInt = DALI_StoreDtrAsPowerOnLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
elseif Wert == 0x10012E then
   Params.SyncIn()
   TempInt = DALI_Write_DTR(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_01)
   TempInt = DALI StoreDtrAsFadeTime(DALI Handle,DALI P UNIT,DALI P ADR)
elseif Wert == 0x10012F then
   Params.SyncIn()
   TempInt = DALI_Write_DTR(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_01)
   TempInt = DALI StoreDtrAsFadeRate(DALI Handle,DALI P UNIT,DALI P ADR)
endif


if Wert >= 0x2140 and Wert <= 0x214F  then
   TempInt = DALI_StoreCurrentLevelInDtr(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
   TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,Wert - 0x2140)
elseif Wert >= 0x10140 and Wert <= 0x1014F  then
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    Params.SyncIn()
    TempInt = DALI_Write_DTR(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_01)
    TempInt = DALI_StoreDtrAsScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,Wert - 0x10140)
  endif

  // Set-Level-Commands
  if Wert == 0x1000001 then
    Params.SyncSingle(Params.IndexOfInt(DALI P CMD PARAM 01) )
    TempInt = DALI_SetShortLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,DALI_P_CMD_PARAM_01)
  elseif Wert == 0x1000002 then
    Params.SyncSingle(Params.IndexOfInt(DALI P CMD PARAM 01) )
    TempInt = DALI_SetGroupLevel(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_02,DALI_P_CMD_PARAM_01)
  elseif Wert == 0x1000003 then
    Params.SyncSingle(Params.IndexOfInt(DALI_P_CMD_PARAM_01) )
    TempInt = DALI SetAllLevel(DALI Handle,DALI P UNIT,DALI P CMD PARAM 01)
  endif

endsub

sub DALI_OnWrite_P_COM_ERROR_RESET(string VarName,bit Wert)
  DALI_R_COM_ERROR_CNT_UNIT = 0
  DALI_R_COM_ERROR_CNT_DEVICE = 0
endsub

sub   DALI Init( num Handle)
  DALI Handle = Handle
  Params.OnSymbolWrite(Params.IndexOfInt(DALI_P_CMD), 0, DALI_OnWrite_P_CMD)
  Params.OnSymbolWrite(Params.IndexOfBit(DALI_P_COM_ERROR_RESET),                                    0,
DALI_OnWrite_P_COM_ERROR_RESET)
  DALI_R_COM_ERROR_CNT_UNIT = DALI_I_COM_ERROR_CNT_UNIT
  DALI_R_COM_ERROR_CNT_DEVICE = DALI_I_COM_ERROR_CNT_DEVICE
endsub
//================================================
sub   DALI_UpdateMain()
  //intconst CMD_QUERY_STATUS              = 0x0090 //144
  DALI_TempValue = DALI_QueryStatus(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_STATUS = DALI_QueryValue
    DALI_R_STATUS_CONTROL_GEAR = Cvt.Int2Bit(DALI_R_STATUS & 0x01)
    DALI_R_STATUS_FAILURE = Cvt.Int2Bit(DALI_R_STATUS & 0x02)
    DALI_R_STATUS_ARC_POWER = Cvt.Int2Bit(DALI_R_STATUS & 0x04)
    DALI_R_STATUS_LIMIT_ERROR = Cvt.Int2Bit(DALI_R_STATUS & 0x08)
    DALI_R_STATUS_FADE_RUNNING = Cvt.Int2Bit(DALI_R_STATUS & 0x10)
    DALI_R_STATUS_RESET_STATE = Cvt.Int2Bit(DALI_R_STATUS & 0x20)
    DALI_R_STATUS_MISSING_SHORTADRESS = Cvt.Int2Bit(DALI_R_STATUS & 0x40)
    DALI_R_STATUS_POWER_FAILURE = Cvt.Int2Bit(DALI_R_STATUS & 0x80)
  endif
  //intconst CMD_QUERY_DTR_CONTENT         = 0x0098 //152
  DALI_TempValue = DALI_QueryDtrContent(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_R_DTR_CONTENT = DALI_QueryValue
  endif
  //intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
  DALI_TempValue = DALI_QueryCurrentLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_R_CURRENT_LEVEL = DALI_QueryValue
     DALI_R_CURRENT_LEVEL_100=DALI_CalcPercent(DALI_R_CURRENT_LEVEL)
  endif
endsub
//================================================
sub   DALI_UpdateStatus()
  //-------------------------
  DALI_UpdateMain()
  //-------------------------
  //intconst CMD_QUERY_IS_WORKING          = 0x0091 //145
  DALI_TempValue = DALI_QueryIsWorking(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
   DALI_R_IS_WORKING = DALI_QueryValue
  endif
  //intconst CMD_QUERY_IS_FAILURE          = 0x0092 //146
  DALI_TempValue = DALI_QueryIsFailure(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_R_IS_FAILURE = DALI_QueryValue
  endif
  //intconst CMD_QUERY_IS_OPERATING        = 0x0093 //147
  DALI_TempValue = DALI_QueryIsOperating(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_R_IS_OPERATING = DALI_QueryValue
  endif
  //intconst CMD_QUERY_OUT_OF_LIMIT        = 0x0094 //148
  DALI_TempValue = DALI_QueryOutOfLimit(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_R_OUT_OF_LIMIT = DALI_QueryValue
  endif
  //intconst CMD_QUERY_RESET_STATE         = 0x0095 //149
  DALI_TempValue = DALI_QueryResetState(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    DALI_R_RESET_STATE = DALI_QueryValue
  endif
  //intconst CMD_QUERY_MISSING_SHORTADRESS = 0x0096 //150
  DALI_TempValue = DALI_QueryMissingShortadress(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_MISSING_SHORTADRESS = DALI_QueryValue
  endif
  //intconst CMD_QUERY_VERSION_NUMBER      = 0x0097 //151
  DALI_TempValue = DALI_QueryVersionNumber(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_VERSION_NUMBER = DALI_QueryValue
  endif
  //intconst CMD_QUERY_DEVICE_TYP          = 0x0099 //153
  DALI_TempValue = DALI_QueryDeviceTyp(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_DEVICE_TYP = DALI_QueryValue
  endif
  //intconst CMD_QUERY_PHYSICAL_MINIMUM    = 0x009A //154
  DALI_TempValue = DALI_QueryPhysicalMinimum(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_PHYSICAL_MINIMUM = DALI_QueryValue
  endif
  //intconst CMD_QUERY_POWER_FAILURE_MODE  = 0x009B //155
  DALI_TempValue = DALI_QueryPowerFailureMode(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_POWER_FAILURE_MODE = DALI_QueryValue
  endif
  //intconst CMD_QUERY_MAX_LEVEL           = 0x00A1 //161
  DALI_TempValue = DALI_QueryMaxLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_MAX_LEVEL = DALI_QueryValue
  endif
  //intconst CMD_QUERY_MIN_LEVEL           = 0x00A2 //162
  DALI_TempValue = DALI_QueryMinLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_MIN_LEVEL = DALI_QueryValue
  endif
  //intconst CMD_QUERY_POWER_UP_LEVEL      = 0x00A3 //163
  DALI_TempValue = DALI_QueryPowerUpLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_POWER_UP_LEVEL = DALI_QueryValue
  endif
  //intconst CMD_QUERY_SYSTEM_FAILURE_LEVEL = 0x00A4 //164
  DALI_TempValue = DALI_QuerySystemFailureLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_SYSTEM_FAILURE_LEVEL = DALI_QueryValue
  endif
  //intconst CMD_QUERY_FADE_TIME_AND_RATE  = 0x00A5 //165
  DALI_TempValue = DALI_QueryFadeTimeAndRate(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_FADE_TIME_AND_RATE = DALI_QueryValue
    DALI_R_FADE_TIME = DALI_QueryValue >> 4
    DALI_R_FADE_RATE = DALI_QueryValue & 0x0f
  endif
  //intconst CMD_QUERY_RANDOMADRESS_HIGH   = 0x00C2 //194
  //intconst CMD_QUERY_RANDOMADRESS_MID    = 0x00C3 //195
  //intconst CMD_QUERY_RANDOMADRESS_LOW    = 0x00C4 //196
  DALI_TempValue = DALI_QueryRandomAdress(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_RANDOMADRESS = DALI_QueryValue
    DALI_R_RANDOMADRESS_HEX = String.FormatHex(DALI_R_RANDOMADRESS,6)
  endif
endsub
//=================================================
sub   DALI_UpdateCommands()
  //-------------------------
  DALI_UpdateMain()
  //-------------------------
endsub
//=================================================
sub   DALI_UpdateScene()
  //-------------------------
  DALI_UpdateMain()
  //-------------------------
  //intconst CMD_QUERY_LEVEL_OF_SCENE_00   = 0x00B0 //176
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,0)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_00 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_01   = 0x00B1 //177
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_01 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_02   = 0x00B2 //178
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,2)
  if DALI_TempValue > 0 then
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    DALI_R_LEVEL_OF_SCENE_02 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_03    = 0x00B3 //179
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,3)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_03 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_04    = 0x00B4 //180
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,4)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_04 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_05    = 0x00B5 //181
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,5)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_05 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_06    = 0x00B6 //182
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,6)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_06 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_07    = 0x00B7 //183
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,7)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_07 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_08    = 0x00B8 //184
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_08 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_09    = 0x00B9 //185
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,9)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_09 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_10    = 0x00BA //186
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_10 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_11    = 0x00BB //187
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,11)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_11 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_12    = 0x00BC //188
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,12)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_12 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_13    = 0x00BD //189
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,13)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_13 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_14    = 0x00BE //190
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,14)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_14 = DALI_QueryValue
  endif
  //intconst CMD_QUERY_LEVEL_OF_SCENE_15    = 0x00BF //191
  DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,15)
  if DALI_TempValue > 0 then
    DALI_R_LEVEL_OF_SCENE_15 = DALI_QueryValue
  endif
endsub
//================================================
sub   DALI_UpdateGroup()
  //-------------------------
  DALI_UpdateMain()
  //-------------------------
  //intconst CMD_QUERY_PART_OF_GROUP_00_07  = 0x00C0 //192
  //intconst CMD_QUERY_PART_OF_GROUP_08_15  = 0x00C1 //193
  DALI_TempValue = DALI_QueryPartOfGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_R_PART_OF_GROUP = DALI_QueryValue
     DALI_R_PART_OF_GROUP_HEX = String.FormatHex(DALI_R_PART_OF_GROUP,4)
  endif
endsub
//================================================
sub   DALI_UpdateAll()
  //-------------------------
  DALI_UpdateMain()
  //-------------------------
  //intconst CMD_QUERY_IS_WORKING          = 0x0091 //145
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
DALI_TempValue = DALI_QueryIsWorking(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
 DALI_R_IS_WORKING = DALI_QueryValue
endif
//intconst CMD_QUERY_IS_FAILURE          = 0x0092 //146
DALI_TempValue = DALI_QueryIsFailure(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_IS_FAILURE = DALI_QueryValue
endif
//intconst CMD_QUERY_IS_OPERATING        = 0x0093 //147
DALI_TempValue = DALI_QueryIsOperating(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_IS_OPERATING = DALI_QueryValue
endif
//intconst CMD_QUERY_OUT_OF_LIMIT        = 0x0094 //148
DALI_TempValue = DALI_QueryOutOfLimit(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_OUT_OF_LIMIT = DALI_QueryValue
endif
//intconst CMD_QUERY_RESET_STATE         = 0x0095 //149
DALI_TempValue = DALI_QueryResetState(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_RESET_STATE = DALI_QueryValue
endif
//intconst CMD_QUERY_MISSING_SHORTADRESS = 0x0096 //150
DALI_TempValue = DALI_QueryMissingShortadress(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_MISSING_SHORTADRESS = DALI_QueryValue
endif
//intconst CMD_QUERY_VERSION_NUMBER      = 0x0097 //151
DALI_TempValue = DALI_QueryVersionNumber(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_VERSION_NUMBER = DALI_QueryValue
endif
//intconst CMD_QUERY_DTR_CONTENT         = 0x0098 //152
DALI_TempValue = DALI_QueryDtrContent(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_DTR_CONTENT = DALI_QueryValue
endif
//intconst CMD_QUERY_PHYSICAL_MINIMUM    = 0x009A //154
DALI_TempValue = DALI_QueryPhysicalMinimum(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_PHYSICAL_MINIMUM = DALI_QueryValue
endif
//intconst CMD_QUERY_POWER_FAILURE_MODE  = 0x009B //155
DALI_TempValue = DALI_QueryPowerFailureMode(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_POWER_FAILURE_MODE = DALI_QueryValue
endif
//intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
DALI_TempValue = DALI_QueryCurrentLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_CURRENT_LEVEL = DALI_QueryValue
   DALI_R_CURRENT_LEVEL_100=DALI_CalcPercent(DALI_R_CURRENT_LEVEL)
endif
//intconst CMD_QUERY_MAX_LEVEL            = 0x00A1 //161
DALI_TempValue = DALI_QueryMaxLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_MAX_LEVEL = DALI_QueryValue
endif
//intconst CMD_QUERY_MIN_LEVEL            = 0x00A2 //162
DALI_TempValue = DALI_QueryMinLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_MIN_LEVEL = DALI_QueryValue
endif
//intconst CMD_QUERY_POWER_UP_LEVEL       = 0x00A3 //163
DALI_TempValue = DALI_QueryPowerUpLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_POWER_UP_LEVEL = DALI_QueryValue
endif
//intconst CMD_QUERY_SYSTEM_FAILURE_LEVEL = 0x00A4 //164
DALI_TempValue = DALI_QuerySystemFailureLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_SYSTEM_FAILURE_LEVEL = DALI_QueryValue
endif
//intconst CMD_QUERY_FADE_TIME_AND_RATE   = 0x00A5 //165
DALI_TempValue = DALI_QueryFadeTimeAndRate(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
   DALI_R_FADE_TIME_AND_RATE = DALI_QueryValue
   DALI_R_FADE_TIME = DALI_QueryValue >> 4
   DALI_R_FADE_RATE = DALI_QueryValue & 0x0f
endif

//intconst CMD_QUERY_LEVEL_OF_SCENE_00    = 0x00B0 //176
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,0)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_00 = DALI_QueryValue
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_01    = 0x00B1 //177
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,1)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_01 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_02    = 0x00B2 //178
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,2)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_02 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_03    = 0x00B3 //179
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,3)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_03 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_04    = 0x00B4 //180
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,4)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_04 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_05    = 0x00B5 //181
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,5)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_05 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_06    = 0x00B6 //182
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,6)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_06 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_07    = 0x00B7 //183
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,7)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_07 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_08    = 0x00B8 //184
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,8)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_08 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_09    = 0x00B9 //185
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,9)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_09 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_10    = 0x00BA //186
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,10)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_10 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_11    = 0x00BB //187
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,11)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_11 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_12    = 0x00BC //188
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,12)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_12 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_13    = 0x00BD //189
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,13)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_13 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_14    = 0x00BE //190
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,14)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_14 = DALI_QueryValue
endif
//intconst CMD_QUERY_LEVEL_OF_SCENE_15    = 0x00BF //191
DALI_TempValue = DALI_QueryLevelOfScene(DALI_Handle,DALI_P_UNIT,DALI_P_ADR,15)
if DALI_TempValue > 0 then
   DALI_R_LEVEL_OF_SCENE_15 = DALI_QueryValue
endif

//intconst CMD_QUERY_PART_OF_GROUP_00_07  = 0x00C0 //192
//intconst CMD_QUERY_PART_OF_GROUP_08_15  = 0x00C1 //193
DALI_TempValue = DALI_QueryPartOfGroup(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
if DALI_TempValue > 0 then
    DALI_R_PART_OF_GROUP = DALI_QueryValue
    DALI_R_PART_OF_GROUP_HEX = String.FormatHex(DALI_R_PART_OF_GROUP,4)
endif

//intconst CMD_QUERY_RANDOMADRESS_HIGH    = 0x00C2 //194
//intconst CMD_QUERY_RANDOMADRESS_MID     = 0x00C3 //195
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  //intconst CMD_QUERY_RANDOMADRESS_LOW     = 0x00C4 //196
  DALI_TempValue = DALI_QueryRandomAdress(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_R_RANDOMADRESS = DALI_QueryValue
    DALI_R_RANDOMADRESS_HEX = String.FormatHex(DALI_R_RANDOMADRESS,6)
  endif
endsub
//================================================
sub   DALI_Update()
  //intconst CMD_QUERY_DEVICE_TYP          = 0x0099 //153
  DALI_TempValue = DALI_QueryDeviceTyp(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_R_DEVICE_TYP = DALI_QueryValue
  endif
  //
  if DALI_P_VIS_UPDATE_SCENE == VIS_UPDATE_ALL then
    DALI_UpdateAll()
    SystemVars.SetInt(LOG_PREFIX+" DALI_P_VIS_UPDATE_SCENE",-1)
  elseif DALI_P_VIS_UPDATE_SCENE == VIS_SCENE_MAIN then
    DALI_UpdateMain()
  elseif DALI_P_VIS_UPDATE_SCENE == VIS_SCENE_STATUS then
    DALI_UpdateStatus()
  elseif DALI_P_VIS_UPDATE_SCENE == VIS_SCENE_DPS_COMMANDS then
    DALI_UpdateCommands()
  elseif DALI_P_VIS_UPDATE_SCENE == VIS_SCENE_SCENE then
    DALI_UpdateScene()
  elseif DALI_P_VIS_UPDATE_SCENE == VIS_SCENE_GROUP then
    DALI_UpdateGroup()
  endif
  DALI_R_COM_OK_UNIT = not Changed.Int(DALI_R_COM_ERROR_CNT_UNIT)
  DALI_R_COM_OK_DEVICE = not Changed.Int(DALI_R_COM_ERROR_CNT_DEVICE)
endsub
```

5.6 INCLUDE module DALI_DTY1.SIB

```
//include "DALI"

// ================================================
// Parameter für DALI Kommunikation
// ================================================
intin DALI_DTY1_P_CMD = 0
  symbol = "$auto: $param: DALI_DTY1_Kommandonummer:false"
endintin
intin DALI_DTY1_P_CMD_PARAM_01 = 0
  symbol = "$auto: $param: Parameter 01 zur Ausführung des DALI_DTY1_Kommandos:false"
endintin
// ================================================
intout DALI_DTY1_R_STATUS = 0
  symbol = "$auto: $param: Operating status:false"
endintout
bitout DALI_DTY1_R_STATUS_CONTROL_GEAR = false
  symbol = "$auto: $param: Operating status [OK;NOK]:false"
endbitout
bitout DALI_DTY1_R_STATUS_FAILURE = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_STATUS_ARC_POWER = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_STATUS_LIMIT_ERROR = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_STATUS_FADE_RUNNING = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_STATUS_RESET_STATE = false
  symbol = "$auto: $param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_STATUS_MISSING_SHORTADRESS = false
  symbol = "$auto: $param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_STATUS_POWER_FAILURE = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
// ================================================
intout DALI_DTY1_R_IS_WORKING = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_DTY1_R_VERSION_NUMBER = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_DTY1_R_DTR_CONTENT = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_DTY1_R_DEVICE_TYP = 0
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_DTY1_R_PHYSICAL_MINIMUM = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_DTY1_R_CURRENT_LEVEL = 0
  symbol = "$auto: $param: .. :false"
endintout

intout DALI_DTY1_R_PART_OF_GROUP_00_07 = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_DTY1_R_PART_OF_GROUP_08_15 = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI_DTY1_R_PART_OF_GROUP = 0
  symbol = "$auto:_$param: .. :false"
endintout
stringout DALI DTY1 R PART OF GROUP HEX = ""
  symbol = "$auto:_$param: .. :false"
endstringout

intout DALI_DTY1_R_RANDOMADRESS_HIGH = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI DTY1 R RANDOMADRESS MID = 0
  symbol = "$auto:_$param: .. :false"
endintout
intout DALI DTY1 R RANDOMADRESS LOW = 0
  symbol = "$auto: $param: .. :false"
endintout
intout DALI_DTY1_R_RANDOMADRESS = 0
  symbol = "$auto: $param: .. :false"
endintout
stringout DALI_DTY1_R_RANDOMADRESS_HEX = ""
  symbol = "$auto: $param: .. :false"
endstringout

// ================================================
// ================================================
intout DALI_DTY1_R_BATTERIE_CHARGE = 0
  symbol = "$auto:_$param: Ladezustand der Batterie:false"
endintout
intout DALI DTY1 R TEST TIMING = 0
  symbol = "$auto:_$param: xxx :false"
endintout
intout DALI DTY1 R DURATION TEST RESULT = 0
  symbol = "$auto: $param: xxx :false"
endintout
intout DALI_DTY1_R_LAMP_EMERGENCY_TIME = 0
  symbol = "$auto: $param: xxx :false"
endintout
intout DALI_DTY1_R_LAMP_TOTAL_OPERATION_TIME = 0
  symbol = "$auto:_$param: xxx :false"
endintout
intout DALI_DTY1_R_EMERGENCY_LEVEL = 0
  symbol = "$auto:_$param: xxx :false"
endintout
intout DALI DTY1 R EMERGENCY MIN LEVEL = 0
  symbol = "$auto:_$param: xxx :false"
endintout
intout DALI DTY1 R EMERGENCY MAX LEVEL = 0
  symbol = "$auto:_$param: xxx :false"
endintout
intout DALI DTY1 R RATED DURATION = 0
  symbol = "$auto: $param: xxx :false"
endintout

intout DALI DTY1 R EMERGENCY MODE = 0
  symbol = "$auto:_$param: xxx :false"
endintout
bitout DALI_DTY1_R_EMERGENCY_MODE_STANDBY = false
  symbol = "$auto: $param: Operating status [OFF;ON]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_MODE_ON = false
  symbol = "$auto: $param: Operating status [OFF;ON]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY = false
  symbol = "$auto:_$param: Operating status [OFF;ON]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY_EXTENDED = false
  symbol = "$auto:_$param: Operating status [OFF;ON]:false"
endbitout
bitout DALI DTY1 R EMERGENCY MODE FUNCTION TEST = false
  symbol = "$auto:_$param: Operating status [OFF;ON]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_MODE_DURATION_TEST = false
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  symbol = "$auto:_$param: Operating status [OFF;ON]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_MODE_INPUT_LOCK = false
  symbol = "$auto: $param: Operating status [OFF;ON]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_MODE_INPUT_SWITCH = false
  symbol = "$auto: $param: Operating status [OFF;ON]:false"
endbitout

intout DALI_DTY1_R_FEATURES = 0
  symbol = "$auto: $param: xxx :false"
endintout
bitout DALI_DTY1_R_FEATURES_INCLUDED_DEVICE = false
  symbol = "$auto:_$param: Feature avaiable [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FEATURES_PERMANENT_DEVICE = false
  symbol = "$auto:_$param: Feature avaiable [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FEATURES_SWITCHED_DEVICE = false
  symbol = "$auto:_$param: Feature avaiable [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FEATURES_AUTOTEST_POSSIBLE = false
  symbol = "$auto:_$param: Feature avaiable [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FEATURES_ADJUSTABLE_EMERGENCY_LEVEL = false
  symbol = "$auto: $param: Feature avaiable [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FEATURES_INPUT_LOCK_ENABLED = false
  symbol = "$auto: $param: Feature avaiable [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FEATURES_PHYSICAL_SELECTION_ENABLED = false
  symbol = "$auto:_$param: Feature avaiable [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FEATURES_STANDBY_RESTART_ENABLED = false
  symbol = "$auto:_$param: Feature avaiable [NO;YES]:false"
endbitout

intout DALI_DTY1_R_FAILURE_STATUS = 0
  symbol = "$auto:_$param: xxx :false"
endintout
bitout DALI_DTY1_R_FAILURE_STATUS_CIRCUIT_BROKEN = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FAILURE_STATUS_DURATION_LOW = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FAILURE_STATUS_BATTERY_BROKEN = false
  symbol = "$auto: $param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FAILURE_STATUS_LAMP_BROKEN = false
  symbol = "$auto: $param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FAILURE_STATUS_FUNCTION_TEST_DELAY = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FAILURE_STATUS_DURATION_TEST_DELAY = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FAILURE_STATUS_FUNCTION_TEST = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_FAILURE_STATUS_DURATION_TEST = false
  symbol = "$auto:_$param: Failure status [NO;YES]:false"
endbitout

intout DALI_DTY1_R_EMERGENCY_STATUS = 0
  symbol = "$auto:_$param: xxx :false"
endintout
bitout DALI_DTY1_R_EMERGENCY_STATUS_LOCKED = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_STATUS_FUNCTION_TEST_READY = false
  symbol = "$auto: $param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_STATUS_DURATION_TEST_READY = false
  symbol = "$auto: $param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_STATUS_BATTERY_LOADED = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_STATUS_FUNCTION_TEST_PENDING = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_STATUS_DURATION_TEST_PENDING = false
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_STATUS_IDENTIFICATION_ACTIVE = false
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  symbol = "$auto:_$param: Operating status [NO;YES]:false"
endbitout
bitout DALI_DTY1_R_EMERGENCY_STATUS_PHYSICAL_SELECTED = false
  symbol = "$auto: $param: Operating status [NO;YES]:false"
endbitout

intout DALI DTY1 R EXTENDED VERSION NUMBER = 0
  symbol = "$auto: $param: xxx :false"
endintout
// ================================================
// Konfiguration
// ================================================
// ================================================
// Spezialkommandos für DALI-DALI_DTY1
// ================================================
//272+ 2 Forward Frame
intconst DALI_DTY1_CMD_START_FUNCTION_TEST =227  //YAAA AAA1 1110 0011  START FUNCTION TEST
intconst DALI DTY1 CMD START DURATION TEST =228  //YAAA AAA1 1110 0100  START DURATION TEST
intconst DALI DTY1 CMD STOPP TEST =229  //YAAA AAA1 1110 0101  STOP TEST
intconst  DALI_DTY1_CMD_RESET_FUNCTION_TEST_DONE_FLAG =230   //YAAA AAA1 1110 0110   RESET FUNCTION
TEST DONE FLAG
intconst  DALI DTY1 CMD RESET DURATION TEST DONE FLAG =231   //YAAA AAA1 1110 0111   RESET DURATION
TEST DONE FLAG
intconst DALI_DTY1_CMD_RESET_LAMP_TIME =232  //YAAA AAA1 1110 1000  RESET LAMP TIME
intconst  DALI DTY1 CMD STORE TEST DELAY TIME HIGH BYTE  =234   //YAAA AAA1 1110 1010   STORE TEST
DELAY TIME HIGH BYTE
intconst DALI_DTY1_CMD_STORE_TEST_DELAY_TIME_LOW_BYTE =235  //YAAA AAA1 1110 1011  STORE TEST DELAY
TIME LOW BYTE
intconst  DALI DTY1 CMD STORE FUNCTION TEST INTERVAL  =236   //YAAA AAA1 1110 1100   STORE FUNCTION
TEST INTERVAL
intconst  DALI_DTY1_CMD_STORE_DURATION_TEST_INTERVAL  =237   //YAAA AAA1 1110 1101   STORE DURATION
TEST INTERVAL
intconst  DALI DTY1 CMD STORE TEST EXECUTION TIMEOUT  =238    //YAAA  AAA1  1110  1110    STORE  TEST
EXECUTION TIMEOUT
intconst DALI_DTY1_CMD_START_IDENTIFICATION =240  //YAAA AAA1 1111 0000  START IDENTIFICATION
intconst  DALI DTY1 CMD PERFORM DTR SELECTED FUNCTION  =254   //YAAA  AAA1  1111  1110   PERFORM  DTR
SELECTED FUNCTION
//272+ 2 Forward Frame not supported from Alvit-MAT3/Alvit-2641
intconst DALI_DTY1_CMD_REST =224  //YAAA AAA1 1110 0000  REST
intconst DALI DTY1 CMD INHIBIT =225  //YAAA AAA1 1110 0001  INHIBIT
intconst DALI_DTY1_CMD_RESET_INHIBIT =226  //YAAA AAA1 1110 0010  RE-LIGHT/RESET INHIBIT
intconst  DALI_DTY1_CMD_STORE_DTR_AS_EMERGENCY_LEVEL  =233   //YAAA  AAA1  1110  1001   STORE  DTR  AS
EMERGENCY LEVEL
intconst DALI DTY1 CMD STORE PROLONG TIME =239  //YAAA AAA1 1110 1111  STORE PROLONG TIME
//272+ 1 Forward Frame
intconst DALI_DTY1_CMD_QUERY_BATTERY_CHARGE =241  //YAAA AAA1 1111 0001  QUERY BATTERY CHARGE
intconst DALI DTY1 CMD QUERY TEST TIMING =242  //YAAA AAA1 1111 0010  QUERY TEST TIMING
intconst DALI DTY1 CMD QUERY DURATION TEST RESULT =243  //YAAA AAA1 1111 0011  QUERY DURATION TEST
RESULT
intconst DALI_DTY1_CMD_QUERY_LAMP_EMERGENCY_TIME =244  //YAAA AAA1 1111 0100  QUERY LAMP EMERGENCY
TIME
intconst  DALI_DTY1_CMD_QUERY_LAMP_TOTAL_OPERATION_TIME  =245   //YAAA  AAA1  1111  0101   QUERY  LAMP
TOTAL OPERATION TIME
intconst DALI_DTY1_CMD_QUERY_EMERGENCY_LEVEL =246  //YAAA AAA1 1111 0110  QUERY EMERGENCY LEVEL
intconst  DALI DTY1 CMD QUERY EMERGENCY MIN LEVEL =247   //YAAA AAA1 1111 0111   QUERY EMERGENCY MIN
LEVEL
intconst  DALI_DTY1_CMD_QUERY_EMERGENCY_MAX_LEVEL =248   //YAAA AAA1 1111 1000   QUERY EMERGENCY MAX
LEVEL
intconst DALI DTY1 CMD QUERY RATED DURATION =249  //YAAA AAA1 1111 1001  QUERY RATED DURATION
intconst DALI_DTY1_CMD_QUERY_EMERGENCY_MODE =250  //YAAA AAA1 1111 1010  QUERY EMERGENCY MODE
intconst DALI_DTY1_CMD_QUERY_FEATURES =251  //YAAA AAA1 1111 1011  QUERY FEATURES
intconst DALI DTY1 CMD QUERY FAILURE STATUS =252  //YAAA AAA1 1111 1100  QUERY FAILURE STATUS
intconst DALI_DTY1_CMD_QUERY_EMERGENCY_STATUS =253  //YAAA AAA1 1111 1101  QUERY EMERGENCY STATUS

intconst  DALI DTY1 CMD QUERY EXTENDED VERSION NUMBER =255   //YAAA AAA1 1111 1111   QUERY EXTENDED
VERSION NUMBER

intconst DALI_DTY1_DEVICE_TYPE          = 0x0001
// ================================================
intconst VIS_DTY1_UPDATE_ALL = 10
intconst VIS_DTY1_SCENE_MAIN = 11
intconst VIS_DTY1_SCENE_STATUS = 12
intconst VIS DTY1 SCENE COMMANDS = 13
// ================================================
// Device Type 1 Functions
// ================================================

//intconst DALI_DTY1_CMD_QUERY_BATTERY_CHARGE =241  //YAAA AAA1 1111 0001  QUERY BATTERY CHARGE
function int DALI_DTY1_QueryBatterieCharge(num Handle,int Unit,int Adr)
  return DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY BATTERY CHARGE,CMD DEVICE TYPE 1)
endfunction

//intconst DALI_DTY1_CMD_QUERY_TEST_TIMING =242  //YAAA AAA1 1111 0010  QUERY TEST TIMING
function int DALI DTY1 QueryTestTiming(num Handle,int Unit,int Adr)
  return DALI_Query_272_1F(Handle,Unit,Adr,DALI_DTY1_CMD_QUERY_TEST_TIMING,CMD_DEVICE_TYPE_1)
endfunction
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
//intconst  DALI_DTY1_CMD_QUERY_DURATION_TEST_RESULT  =243   //YAAA  AAA1  1111  0011   QUERY  DURATION
TEST RESULT
function int DALI_DTY1_QueryDurationTestResult(num Handle,int Unit,int Adr)
  return
DALI_Query_272_1F(Handle,Unit,Adr,DALI_DTY1_CMD_QUERY_DURATION_TEST_RESULT,CMD_DEVICE_TYPE_1)
endfunction
//intconst  DALI DTY1 CMD QUERY LAMP EMERGENCY TIME  =244    //YAAA  AAA1  1111  0100    QUERY  LAMP
EMERGENCY TIME
function int DALI_DTY1_QueryLampEmergencyTime(num Handle,int Unit,int Adr)
  return
DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY LAMP EMERGENCY TIME,CMD DEVICE TYPE 1)
endfunction

//intconst  DALI_DTY1_CMD_QUERY_LAMP_TOTAL_OPERATION_TIME  =245   //YAAA  AAA1  1111  0101   QUERY  LAMP
TOTAL OPERATION TIME
function int DALI_DTY1_QueryLampTotalOperationTime(num Handle,int Unit,int Adr)
  return
DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY LAMP TOTAL OPERATION TIME,CMD DEVICE TYPE 1)
endfunction

//intconst DALI_DTY1_CMD_QUERY_EMERGENCY_LEVEL =246  //YAAA AAA1 1111 0110  QUERY EMERGENCY LEVEL
function int DALI DTY1 QueryEmergencyLevel(num Handle,int Unit,int Adr)
  return DALI_Query_272_1F(Handle,Unit,Adr,DALI_DTY1_CMD_QUERY_EMERGENCY_LEVEL,CMD_DEVICE_TYPE_1)
endfunction

//intconst DALI DTY1 CMD QUERY EMERGENCY MIN LEVEL =247  //YAAA AAA1 1111 0111  QUERY EMERGENCY MIN
LEVEL
function int DALI_DTY1_QueryEmergencyMinLevel(num Handle,int Unit,int Adr)
  return
DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY EMERGENCY MIN LEVEL,CMD DEVICE TYPE 1)
endfunction

//intconst DALI DTY1 CMD QUERY EMERGENCY MAX LEVEL =248  //YAAA AAA1 1111 1000  QUERY EMERGENCY MAX
LEVEL
function int DALI_DTY1_QueryEmergencyMaxLevel(num Handle,int Unit,int Adr)
  return
DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY EMERGENCY MAX LEVEL,CMD DEVICE TYPE 1)
endfunction

//intconst DALI DTY1 CMD QUERY RATED DURATION =249  //YAAA AAA1 1111 1001  QUERY RATED DURATION
function int DALI_DTY1_QueryRatedDuration(num Handle,int Unit,int Adr)
  return DALI_Query_272_1F(Handle,Unit,Adr,DALI_DTY1_CMD_QUERY_RATED_DURATION,CMD_DEVICE_TYPE_1)
endfunction

//intconst DALI_DTY1_CMD_QUERY_EMERGENCY_MODE =250  //YAAA AAA1 1111 1010  QUERY EMERGENCY MODE
function int DALI_DTY1_QueryEmergencyMode(num Handle,int Unit,int Adr)
  return DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY EMERGENCY MODE,CMD DEVICE TYPE 1)
endfunction

//intconst DALI_DTY1_CMD_QUERY_FEATURES =251  //YAAA AAA1 1111 1011  QUERY FEATURES
function int DALI DTY1 QueryFeatures(num Handle,int Unit,int Adr)
  return DALI_Query_272_1F(Handle,Unit,Adr,DALI_DTY1_CMD_QUERY_FEATURES,CMD_DEVICE_TYPE_1)
endfunction

//intconst DALI DTY1 CMD QUERY FAILURE STATUS =252  //YAAA AAA1 1111 1100  QUERY FAILURE STATUS
function int DALI_DTY1_QueryFailureStatus(num Handle,int Unit,int Adr)
  return DALI_Query_272_1F(Handle,Unit,Adr,DALI_DTY1_CMD_QUERY_FAILURE_STATUS,CMD_DEVICE_TYPE_1)
endfunction

//intconst DALI_DTY1_CMD_QUERY_EMERGENCY_STATUS =253  //YAAA AAA1 1111 1101  QUERY EMERGENCY STATUS
function int DALI_DTY1_QueryEmergencyStatus(num Handle,int Unit,int Adr)
  return DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY EMERGENCY STATUS,CMD DEVICE TYPE 1)
endfunction

//intconst DALI DTY1 CMD QUERY EXTENDED VERSION NUMBER =255  //YAAA AAA1 1111 1111  QUERY EXTENDED
VERSION NUMBER
function int DALI_DTY1_QueryExtendedVersionNumber(num Handle,int Unit,int Adr)
  return
DALI Query 272 1F(Handle,Unit,Adr,DALI DTY1 CMD QUERY EXTENDED VERSION NUMBER,CMD DEVICE TYPE 1)
endfunction

//============================================================
// SET-Functions
//============================================================
//intconst CMD_STORE_DTR_AS_SHORTADRESS  = 0x0080 //128
function int DALI DTY1 StoreDtrAsShortadress(num Handle,int Unit,int Adr)
  return DALI Send 272 2F(Handle,Unit,Adr,CMD STORE DTR AS SHORTADRESS,CMD DEVICE TYPE 1)
endfunction

//intconst DALI DTY1 CMD START FUNCTION TEST =227  //YAAA AAA1 1110 0011  START FUNCTION TEST
function int DALI_StartFunctionTest(num Handle,int Unit,int Adr)
  return DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_START_FUNCTION_TEST,CMD_DEVICE_TYPE_1)
endfunction
//intconst DALI DTY1 CMD START DURATION TEST =228  //YAAA AAA1 1110 0100  START DURATION TEST
function int DALI_StartDurationTest(num Handle,int Unit,int Adr)
  return DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_START_DURATION_TEST,CMD_DEVICE_TYPE_1)
endfunction
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
//intconst DALI_DTY1_CMD_STOPP_TEST =229  //YAAA AAA1 1110 0101  STOP TEST
function int DALI_StoppTest(num Handle,int Unit,int Adr)
  return DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_STOPP_TEST,CMD_DEVICE_TYPE_1)
endfunction
//intconst DALI_DTY1_CMD_RESET_FUNCTION_TEST_DONE_FLAG =230  //YAAA AAA1 1110 0110  RESET FUNCTION
TEST DONE FLAG
function int DALI ResetFunctionTestDoneFlag(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_RESET_FUNCTION_TEST_DONE_FLAG,CMD_DEVICE_TYPE_1)
endfunction
//intconst DALI DTY1 CMD RESET DURATION TEST DONE FLAG =231  //YAAA AAA1 1110 0111  RESET DURATION
TEST DONE FLAG
function int DALI_ResetDurationTestDoneFlag(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(Handle,Unit,Adr,DALI DTY1 CMD RESET DURATION TEST DONE FLAG,CMD DEVICE TYPE 1)
endfunction
//intconst DALI_DTY1_CMD_RESET_LAMP_TIME =232  //YAAA AAA1 1110 1000  RESET LAMP TIME
function int DALI ResetLampTime(num Handle,int Unit,int Adr)
  return DALI Send 272 2F(Handle,Unit,Adr,DALI DTY1 CMD RESET LAMP TIME,CMD DEVICE TYPE 1)
endfunction
//intconst  DALI_DTY1_CMD_STORE_TEST_DELAY_TIME_HIGH_BYTE =234   //YAAA AAA1 1110 1010   STORE TEST
DELAY TIME HIGH BYTE
function int DALI_StoreTestDelayTimeHighByte(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(Handle,Unit,Adr,DALI DTY1 CMD STORE TEST DELAY TIME HIGH BYTE,CMD DEVICE TYPE 1)
endfunction
//intconst  DALI_DTY1_CMD_STORE_TEST_DELAY_TIME_LOW_BYTE =235   //YAAA AAA1 1110 1011   STORE TEST
DELAY TIME LOW BYTE
function int DALI StoreTestDelayTimeLowByte(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_STORE_TEST_DELAY_TIME_LOW_BYTE,CMD_DEVICE_TYPE_1)
endfunction
//intconst  DALI DTY1 CMD STORE FUNCTION TEST INTERVAL =236  //YAAA AAA1 1110 1100  STORE FUNCTION
TEST INTERVAL
function int DALI_StoreFunctionTestInterval(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(Handle,Unit,Adr,DALI DTY1 CMD STORE FUNCTION TEST INTERVAL,CMD DEVICE TYPE 1)
endfunction
//intconst  DALI_DTY1_CMD_STORE_DURATION_TEST_INTERVAL =237  //YAAA AAA1 1110 1101  STORE DURATION
TEST INTERVAL
function int DALI_StoreDurationTestInterval(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_STORE_DURATION_TEST_INTERVAL,CMD_DEVICE_TYPE_1)
endfunction
//intconst  DALI_DTY1_CMD_STORE_TEST_EXECUTION_TIMEOUT  =238   //YAAA AAA1 1110 1110   STORE TEST
EXECUTION TIMEOUT
function int DALI StoreTestExecutionTimeOut(num Handle,int Unit,int Adr)
  return
DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_STORE_TEST_EXECUTION_TIMEOUT,CMD_DEVICE_TYPE_1)
endfunction
//intconst DALI DTY1 CMD START IDENTIFICATION =240  //YAAA AAA1 1111 0000  START IDENTIFICATION
function int DALI_StartIdentification(num Handle,int Unit,int Adr)
  return DALI_Send_272_2F(Handle,Unit,Adr,DALI_DTY1_CMD_START_IDENTIFICATION,CMD_DEVICE_TYPE_1)
endfunction
//intconst  DALI DTY1 CMD PERFORM DTR SELECTED FUNCTION =254   //YAAA AAA1 1111 1110   PERFORM DTR
SELECTED FUNCTION
function int DALI_PerformDtrSelectedFunction(num Handle,int Unit,int Adr)
  return
DALI Send 272 2F(Handle,Unit,Adr,DALI DTY1 CMD PERFORM DTR SELECTED FUNCTION,CMD DEVICE TYPE 1)
endfunction
//=============================================================================================
=========================
sub DALI_DTY1_OnWrite_P_CMD(string VarName,int Wert)
  int TempInt
  if Wert == 128  then //intconst CMD STORE DTR AS SHORTADRESS  = 0x0080 //128
     TempInt = DALI StoreDtrAsShortadress(DALI Handle,DALI P UNIT,DALI P ADR)
  elseif Wert == 0x10180 then
     Params.SyncIn()
     TempInt = DALI Write DTR(DALI Handle,DALI P UNIT,(DALI DTY1 P CMD PARAM 01<<1)|0x01)
     TempInt = DALI_StoreDtrAsShortadress(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  endif

  if Wert == 144 then //intconst CMD QUERY STATUS              = 0x0090 //144
    DALI_TempValue = DALI_QueryStatus(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI DTY1 R STATUS = DALI QueryValue
       DALI DTY1 R STATUS CONTROL GEAR = Cvt.Int2Bit(DALI DTY1 R STATUS & 0x01)
       DALI_DTY1_R_STATUS_FAILURE = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x02)
       DALI_DTY1_R_STATUS_ARC_POWER = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x04)
       DALI DTY1 R STATUS LIMIT ERROR = Cvt.Int2Bit(DALI DTY1 R STATUS & 0x08)
       DALI_DTY1_R_STATUS_FADE_RUNNING = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x10)
       DALI_DTY1_R_STATUS_RESET_STATE = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x20)
       DALI_DTY1_R_STATUS_MISSING_SHORTADRESS = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x40)
       DALI DTY1 R STATUS POWER FAILURE = Cvt.Int2Bit(DALI DTY1 R STATUS & 0x80)
    endif
  elseif Wert == 145 then //intconst CMD_QUERY_IS_WORKING          = 0x0091 //145
    DALI_TempValue = DALI_QueryIsWorking(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    if DALI_TempValue > 0 then
       DALI_DTY1_R_IS_WORKING = DALI_QueryValue
    endif
  elseif Wert == 151 then //intconst CMD QUERY VERSION NUMBER      = 0x0097 //151
    DALI_TempValue = DALI_QueryVersionNumber(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI DTY1 R VERSION NUMBER = DALI QueryValue
    endif
  elseif Wert == 152 then //intconst CMD_QUERY_DTR_CONTENT         = 0x0098 //152
    DALI_TempValue = DALI_QueryDtrContent(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI_DTY1_R_DTR_CONTENT = DALI_QueryValue
    endif
  elseif Wert == 153 then //intconst CMD_QUERY_DEVICE_TYP          = 0x0099 //153
    DALI TempValue = DALI QueryDeviceTyp(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
       DALI_DTY1_R_DEVICE_TYP = DALI_QueryValue
    endif
  elseif Wert == 154 then //intconst CMD QUERY PHYSICAL MINIMUM    = 0x009A //154
    DALI_TempValue = DALI_QueryPhysicalMinimum(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
       DALI DTY1 R PHYSICAL MINIMUM = DALI QueryValue
    endif
  elseif Wert == 160 then //intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
    DALI TempValue = DALI QueryCurrentLevel(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI TempValue > 0 then
       DALI_DTY1_R_CURRENT_LEVEL = DALI_QueryValue
    endif
   endif

  if Wert == 192 then //intconst CMD_QUERY_PART_OF_GROUP_00_07  = 0x00C0 //192
    DALI_TempValue = DALI_QueryPartOfGroup_00_07(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI TempValue > 0 then
        DALI_DTY1_R_PART_OF_GROUP_00_07 = DALI_QueryValue
    endif
  elseif Wert == 193 then //intconst CMD QUERY PART OF GROUP 08 15  = 0x00C1 //193
    DALI TempValue = DALI QueryPartOfGroup 08 15(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
        DALI_DTY1_R_PART_OF_GROUP_08_15 = DALI_QueryValue
    endif
  elseif Wert == 194 then //intconst CMD_QUERY_RANDOMADRESS_HIGH    = 0x00C2 //194
    DALI_TempValue = DALI_QueryRandomAdressHigh(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
        DALI DTY1 R RANDOMADRESS HIGH = DALI QueryValue
    endif
  elseif Wert == 195 then //intconst CMD_QUERY_RANDOMADRESS_MID     = 0x00C3 //195
    DALI TempValue = DALI QueryRandomAdressMid(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI TempValue > 0 then
        DALI_DTY1_R_RANDOMADRESS_MID = DALI_QueryValue
    endif
  elseif Wert == 196 then //intconst CMD QUERY RANDOMADRESS LOW     = 0x00C4 //196
    DALI_TempValue = DALI_QueryRandomAdressLow(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
        DALI_DTY1_R_RANDOMADRESS_LOW = DALI_QueryValue
    endif
  endif

  if Wert == 227 then //intconst DALI DTY1 CMD START FUNCTION TEST =227   //YAAA AAA1 1110 0011
START FUNCTION TEST
    TempInt = DALI_StartFunctionTest(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 228 then //intconst DALI_DTY1_CMD_START_DURATION_TEST =228  //YAAA AAA1 1110 0100
START DURATION TEST
    TempInt = DALI_StartDurationTest(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 229 then //intconst DALI_DTY1_CMD_STOPP_TEST =229   //YAAA AAA1 1110 0101   STOP
TEST
    TempInt = DALI StoppTest(DALI Handle,DALI P UNIT,DALI P ADR)
  elseif Wert == 230 then //intconst DALI_DTY1_CMD_RESET_FUNCTION_TEST_DONE_FLAG =230  //YAAA AAA1
1110 0110  RESET FUNCTION TEST DONE FLAG
    TempInt = DALI ResetFunctionTestDoneFlag(DALI Handle,DALI P UNIT,DALI P ADR)
  elseif Wert == 231 then //intconst DALI_DTY1_CMD_RESET_DURATION_TEST_DONE_FLAG =231  //YAAA AAA1
1110 0111  RESET DURATION TEST DONE FLAG
    TempInt = DALI_ResetDurationTestDoneFlag(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 232 then //intconst DALI DTY1 CMD RESET LAMP TIME =232   //YAAA AAA1 1110 1000
RESET LAMP TIME
    TempInt = DALI_ResetLampTime(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 234 then //intconst DALI DTY1 CMD STORE TEST DELAY TIME HIGH BYTE =234   //YAAA
AAA1 1110 1010  STORE TEST DELAY TIME HIGH BYTE
    TempInt = DALI_StoreTestDelayTimeHighByte(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 235 then //intconst DALI_DTY1_CMD_STORE_TEST_DELAY_TIME_LOW_BYTE =235  //YAAA AAA1
1110 1011  STORE TEST DELAY TIME LOW BYTE
    TempInt = DALI_StoreTestDelayTimeLowByte(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 236 then //intconst DALI_DTY1_CMD_STORE_FUNCTION_TEST_INTERVAL =236  //YAAA AAA1
1110 1100  STORE FUNCTION TEST INTERVAL
    TempInt = DALI StoreFunctionTestInterval(DALI Handle,DALI P UNIT,DALI P ADR)
  elseif Wert == 237 then //intconst DALI_DTY1_CMD_STORE_DURATION_TEST_INTERVAL =237  //YAAA AAA1
1110 1101  STORE DURATION TEST INTERVAL
    TempInt = DALI_StoreDurationTestInterval(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
  elseif Wert == 238 then //intconst DALI_DTY1_CMD_STORE_TEST_EXECUTION_TIMEOUT =238  //YAAA AAA1
1110 1110  STORE TEST EXECUTION TIMEOUT
    TempInt = DALI_StoreTestExecutionTimeout(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  elseif Wert == 240 then //intconst DALI DTY1 CMD START IDENTIFICATION =240  //YAAA AAA1 1111 0000
START IDENTIFICATION
    TempInt = DALI_StartIdentification(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  endif

  if Wert == 250 then //intconst DALI_DTY1_CMD_QUERY_EMERGENCY_MODE =250  //YAAA AAA1 1111 1010
QUERY EMERGENCY MODE
    DALI TempValue = DALI DTY1 QueryEmergencyMode(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
      DALI_DTY1_R_EMERGENCY_MODE = DALI_QueryValue
      DALI_DTY1_R_EMERGENCY_MODE_STANDBY = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x01)
      DALI DTY1 R EMERGENCY MODE ON = Cvt.Int2Bit(DALI DTY1 R EMERGENCY MODE & 0x02)
      DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x04)
      DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY_EXTENDED  =  Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE  &
0x08)
      DALI DTY1 R EMERGENCY MODE FUNCTION TEST = Cvt.Int2Bit(DALI DTY1 R EMERGENCY MODE & 0x10)
      DALI_DTY1_R_EMERGENCY_MODE_DURATION_TEST = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x20)
      DALI_DTY1_R_EMERGENCY_MODE_INPUT_LOCK = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x40)
      DALI DTY1 R EMERGENCY MODE INPUT SWITCH = Cvt.Int2Bit(DALI DTY1 R EMERGENCY MODE & 0x80)
    endif
  elseif Wert == 251 then //intconst DALI_DTY1_CMD_QUERY_FEATURES =251  //YAAA AAA1 1111 1011
QUERY FEATURES
    DALI TempValue = DALI DTY1 QueryFeatures(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
      DALI_DTY1_R_FEATURES = DALI_QueryValue
      DALI DTY1 R FEATURES INCLUDED DEVICE = Cvt.Int2Bit(DALI DTY1 R FEATURES & 0x01)
      DALI DTY1 R FEATURES PERMANENT DEVICE = Cvt.Int2Bit(DALI DTY1 R FEATURES & 0x02)
      DALI_DTY1_R_FEATURES_SWITCHED_DEVICE = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x04)
      DALI_DTY1_R_FEATURES_AUTOTEST_POSSIBLE = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x08)
      DALI DTY1 R FEATURES ADJUSTABLE EMERGENCY LEVEL = Cvt.Int2Bit(DALI DTY1 R FEATURES & 0x10)
      DALI_DTY1_R_FEATURES_INPUT_LOCK_ENABLED = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x20)
      DALI_DTY1_R_FEATURES_PHYSICAL_SELECTION_ENABLED = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x40)
      DALI DTY1 R FEATURES STANDBY RESTART ENABLED = Cvt.Int2Bit(DALI DTY1 R FEATURES & 0x80)
    endif
  elseif Wert == 252 then //intconst DALI_DTY1_CMD_QUERY_FAILURE_STATUS =252  //YAAA AAA1 1111 1100
QUERY FAILURE STATUS
    DALI TempValue = DALI DTY1 QueryFailureStatus(DALI Handle,DALI P UNIT,DALI P ADR)
    if DALI_TempValue > 0 then
      DALI_DTY1_R_FAILURE_STATUS = DALI_QueryValue
      DALI_DTY1_R_FAILURE_STATUS_CIRCUIT_BROKEN = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x01)
      DALI DTY1 R FAILURE STATUS DURATION LOW = Cvt.Int2Bit(DALI DTY1 R FAILURE STATUS & 0x02)
      DALI_DTY1_R_FAILURE_STATUS_BATTERY_BROKEN = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x04)
      DALI_DTY1_R_FAILURE_STATUS_LAMP_BROKEN = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x08)
      DALI DTY1 R FAILURE STATUS FUNCTION TEST DELAY  =  Cvt.Int2Bit(DALI DTY1 R FAILURE STATUS  &
0x10)
       DALI_DTY1_R_FAILURE_STATUS_DURATION_TEST_DELAY  =  Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS  &
0x20)
      DALI DTY1 R FAILURE STATUS FUNCTION TEST = Cvt.Int2Bit(DALI DTY1 R FAILURE STATUS & 0x40)
      DALI_DTY1_R_FAILURE_STATUS_DURATION_TEST = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x80)
    endif
  elseif Wert == 253 then //intconst DALI_DTY1_CMD_QUERY_EMERGENCY_STATUS =253  //YAAA AAA1 1111
1101  QUERY EMERGENCY STATUS
    DALI_TempValue = DALI_DTY1_QueryEmergencyStatus(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
    if DALI_TempValue > 0 then
      DALI DTY1 R EMERGENCY STATUS = DALI QueryValue
       DALI DTY1 R EMERGENCY STATUS LOCKED = Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS & 0x01)
      DALI_DTY1_R_EMERGENCY_STATUS_FUNCTION_TEST_READY = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS &
0x02)
      DALI DTY1 R EMERGENCY STATUS DURATION TEST READY = Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS &
0x04)
      DALI_DTY1_R_EMERGENCY_STATUS_BATTERY_LOADED  =  Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS  &
0x08)
      DALI DTY1 R EMERGENCY STATUS FUNCTION TEST PENDING = Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS
& 0x10)
      DALI_DTY1_R_EMERGENCY_STATUS_DURATION_TEST_PENDING = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS
& 0x20)
      DALI_DTY1_R_EMERGENCY_STATUS_IDENTIFICATION_ACTIVE = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS
& 0x40)
      DALI_DTY1_R_EMERGENCY_STATUS_PHYSICAL_SELECTED = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS &
0x80)
    endif
  elseif Wert == 254 then //intconst DALI_DTY1_CMD_PERFORM_DTR_SELECTED_FUNCTION =254  //YAAA AAA1
1111 1110  PERFORM DTR SELECTED FUNCTION
    TempInt = DALI PerformDtrSelectedFunction(DALI Handle,DALI P UNIT,DALI P ADR)
  endif

  if Wert == 257  then //intconst CMD WRITE DATA TRANSFER REGISTER    = 0xA300 //Cmd-257
    Params.SyncIn()
    TempInt = DALI_Write_DTR(DALI_Handle,DALI_P_UNIT,DALI_P_CMD_PARAM_01)
  endif

endsub

sub DALI_DTY1_OnWrite_P_COM_ERROR_RESET(string VarName,bit Wert)
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
//  DALI_DTY1_R_COM_ERROR_CNT_UNIT = 0
//  DALI_DTY1_R_COM_ERROR_CNT_LAMP = 0
endsub
//=============================================================================================================
=========================
//=============================================================================================================
=========================
sub   DALI DTY1 Init( num Handle)
  DALI_Handle = Handle
  Params.OnSymbolWrite(Params.IndexOfInt(DALI_DTY1_P_CMD), 0, DALI_DTY1_OnWrite_P_CMD)
endsub

sub DALI_DTY1_UpdateMain ()
  //intconst CMD_QUERY_CURRENT_LEVEL        = 0x00A0 //160
  DALI TempValue = DALI QueryCurrentLevel(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_CURRENT_LEVEL = DALI_QueryValue
  endif
  //intconst DALI DTY1 CMD QUERY BATTERY CHARGE =241  //YAAA AAA1 1111 0001  QUERY BATTERY CHARGE
  DALI_TempValue = DALI_DTY1_QueryBatterieCharge(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI DTY1 R BATTERIE CHARGE = DALI QueryValue
  endif
  //
  DALI TempValue = DALI QueryStatus(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI TempValue > 0 then
    DALI_DTY1_R_STATUS = DALI_QueryValue
    DALI_DTY1_R_STATUS_CONTROL_GEAR = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x01)
    DALI DTY1 R STATUS FAILURE = Cvt.Int2Bit(DALI DTY1 R STATUS & 0x02)
    DALI DTY1 R STATUS ARC POWER = Cvt.Int2Bit(DALI DTY1 R STATUS & 0x04)
    DALI_DTY1_R_STATUS_LIMIT_ERROR = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x08)
    DALI_DTY1_R_STATUS_FADE_RUNNING = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x10)
    DALI DTY1 R STATUS RESET STATE = Cvt.Int2Bit(DALI DTY1 R STATUS & 0x20)
    DALI_DTY1_R_STATUS_MISSING_SHORTADRESS = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x40)
    DALI_DTY1_R_STATUS_POWER_FAILURE = Cvt.Int2Bit(DALI_DTY1_R_STATUS & 0x80)
  endif
  DALI TempValue = DALI DTY1 QueryEmergencyMode(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_EMERGENCY_MODE = DALI_QueryValue
    DALI DTY1 R EMERGENCY MODE STANDBY = Cvt.Int2Bit(DALI DTY1 R EMERGENCY MODE & 0x01)
    DALI_DTY1_R_EMERGENCY_MODE_ON = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x02)
    DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x04)
    DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY_EXTENDED = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x08)
    DALI DTY1 R EMERGENCY MODE FUNCTION TEST = Cvt.Int2Bit(DALI DTY1 R EMERGENCY MODE & 0x10)
    DALI_DTY1_R_EMERGENCY_MODE_DURATION_TEST = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x20)
    DALI_DTY1_R_EMERGENCY_MODE_INPUT_LOCK = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_MODE & 0x40)
    DALI DTY1 R EMERGENCY MODE INPUT SWITCH = Cvt.Int2Bit(DALI DTY1 R EMERGENCY MODE & 0x80)
  endif
  DALI_TempValue = DALI_DTY1_QueryFeatures(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI DTY1 R FEATURES = DALI QueryValue
    DALI_DTY1_R_FEATURES_INCLUDED_DEVICE = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x01)
    DALI_DTY1_R_FEATURES_PERMANENT_DEVICE = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x02)
    DALI_DTY1_R_FEATURES_SWITCHED_DEVICE = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x04)
    DALI DTY1 R FEATURES AUTOTEST POSSIBLE = Cvt.Int2Bit(DALI DTY1 R FEATURES & 0x08)
    DALI_DTY1_R_FEATURES_ADJUSTABLE_EMERGENCY_LEVEL = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x10)
    DALI_DTY1_R_FEATURES_INPUT_LOCK_ENABLED = Cvt.Int2Bit(DALI_DTY1_R_FEATURES & 0x20)
    DALI DTY1 R FEATURES PHYSICAL SELECTION ENABLED = Cvt.Int2Bit(DALI DTY1 R FEATURES & 0x40)
    DALI DTY1 R FEATURES STANDBY RESTART ENABLED = Cvt.Int2Bit(DALI DTY1 R FEATURES & 0x80)
  endif
  DALI_TempValue = DALI_DTY1_QueryFailureStatus(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI TempValue > 0 then
    DALI_DTY1_R_FAILURE_STATUS = DALI_QueryValue
    DALI_DTY1_R_FAILURE_STATUS_CIRCUIT_BROKEN = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x01)
    DALI DTY1 R FAILURE STATUS DURATION LOW = Cvt.Int2Bit(DALI DTY1 R FAILURE STATUS & 0x02)
    DALI DTY1 R FAILURE STATUS BATTERY BROKEN = Cvt.Int2Bit(DALI DTY1 R FAILURE STATUS & 0x04)
    DALI_DTY1_R_FAILURE_STATUS_LAMP_BROKEN = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x08)
    DALI_DTY1_R_FAILURE_STATUS_FUNCTION_TEST_DELAY = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x10)
    DALI DTY1 R FAILURE STATUS DURATION TEST DELAY = Cvt.Int2Bit(DALI DTY1 R FAILURE STATUS & 0x20)
    DALI_DTY1_R_FAILURE_STATUS_FUNCTION_TEST = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x40)
    DALI_DTY1_R_FAILURE_STATUS_DURATION_TEST = Cvt.Int2Bit(DALI_DTY1_R_FAILURE_STATUS & 0x80)
  endif
  DALI TempValue = DALI DTY1 QueryEmergencyStatus(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_EMERGENCY_STATUS = DALI_QueryValue
    DALI DTY1 R EMERGENCY STATUS LOCKED = Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS & 0x01)
    DALI DTY1 R EMERGENCY STATUS FUNCTION TEST READY  =  Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS  &
0x02)
    DALI_DTY1_R_EMERGENCY_STATUS_DURATION_TEST_READY  =  Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS  &
0x04)
    DALI_DTY1_R_EMERGENCY_STATUS_BATTERY_LOADED = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS & 0x08)
    DALI_DTY1_R_EMERGENCY_STATUS_FUNCTION_TEST_PENDING = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS &
0x10)
    DALI DTY1 R EMERGENCY STATUS DURATION TEST PENDING = Cvt.Int2Bit(DALI DTY1 R EMERGENCY STATUS &
0x20)
    DALI_DTY1_R_EMERGENCY_STATUS_IDENTIFICATION_ACTIVE = Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS &
0x40)
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
    DALI_DTY1_R_EMERGENCY_STATUS_PHYSICAL_SELECTED   =   Cvt.Int2Bit(DALI_DTY1_R_EMERGENCY_STATUS   &
0x80)
  endif
endsub
//================================================
sub DALI_DTY1_UpdateStatus ()
  //-------------------------
  DALI DTY1 UpdateMain()
  //-------------------------
  DALI_TempValue = DALI_QueryIsWorking(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_IS_WORKING = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryVersionNumber(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_VERSION_NUMBER = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryPhysicalMinimum(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_PHYSICAL_MINIMUM = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryCurrentLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_CURRENT_LEVEL = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryPartOfGroup_00_07(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_PART_OF_GROUP_00_07 = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryPartOfGroup_08_15(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_PART_OF_GROUP_08_15 = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryRandomAdressHigh(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_RANDOMADRESS_HIGH = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryRandomAdressMid(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_RANDOMADRESS_MID = DALI_QueryValue
  endif
  DALI_TempValue = DALI_QueryRandomAdressLow(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_RANDOMADRESS_LOW = DALI_QueryValue
  endif

//--
  DALI_TempValue = DALI_DTY1_QueryTestTiming(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_TEST_TIMING = DALI_QueryValue
  endif
//--
  DALI_TempValue = DALI_DTY1_QueryDurationTestResult(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_DURATION_TEST_RESULT = DALI_QueryValue
  endif
//--
  DALI_TempValue = DALI_DTY1_QueryLampEMergencyTime(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_LAMP_EMERGENCY_TIME = DALI_QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryLampTotalOperationTime(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_LAMP_TOTAL_OPERATION_TIME = DALI_QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryEmergencyLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_EMERGENCY_LEVEL = DALI_QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryEmergencyMinLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_EMERGENCY_MIN_LEVEL = DALI_QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryEMergencyMaxLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_EMERGENCY_MAX_LEVEL = DALI_QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryRatedDuration(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_RATED_DURATION = DALI_QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryExtendedVersionNumber(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_EXTENDED_VERSION_NUMBER = DALI_QueryValue
  endif
endsub
//================================================
```

Proprietary data. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

```
sub DALI_DTY1_UpdateCommands ()
  //-------------------------
  DALI_DTY1_UpdateMain()
  //-------------------------
endsub
//=====================================================
sub DALI DTY1 UpdateAll ()
  //-------------------------
  DALI_DTY1_UpdateMain()
  //-------------------------
  DALI TempValue = DALI DTY1 QueryTestTiming(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_TEST_TIMING = DALI_QueryValue
  endif
  DALI TempValue = DALI DTY1 QueryDurationTestResult(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_DURATION_TEST_RESULT = DALI_QueryValue
  endif
  DALI TempValue = DALI DTY1 QueryLampEMergencyTime(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_LAMP_EMERGENCY_TIME = DALI_QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryLampTotalOperationTime(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI DTY1 R LAMP TOTAL OPERATION TIME = DALI QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryEmergencyLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI DTY1 R EMERGENCY LEVEL = DALI QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryEmergencyMinLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI DTY1 R EMERGENCY MIN LEVEL = DALI QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryEMergencyMaxLevel(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI DTY1 R EMERGENCY MAX LEVEL = DALI QueryValue
  endif
  DALI_TempValue = DALI_DTY1_QueryRatedDuration(DALI_Handle,DALI_P_UNIT,DALI_P_ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_RATED_DURATION = DALI_QueryValue
  endif

  DALI TempValue = DALI DTY1 QueryExtendedVersionNumber(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
     DALI_DTY1_R_EXTENDED_VERSION_NUMBER = DALI_QueryValue
  endif
endsub
//=====================================================
sub   DALI_DTY1_Update()
  DALI TempValue = DALI QueryDeviceTyp(DALI Handle,DALI P UNIT,DALI P ADR)
  if DALI_TempValue > 0 then
    DALI_DTY1_R_DEVICE_TYP = DALI_QueryValue
  endif
  //
  if DALI_P_VIS_UPDATE_SCENE == VIS_DTY1_UPDATE_ALL then
    DALI_DTY1_UpdateAll()
    SystemVars.SetInt(LOG PREFIX+" DALI P VIS UPDATE SCENE",-1)
  elseif DALI P VIS UPDATE SCENE == VIS DTY1 SCENE MAIN then
    DALI_DTY1_UpdateMain()
  elseif DALI_P_VIS_UPDATE_SCENE == VIS_DTY1_SCENE_STATUS then
    DALI DTY1 UpdateStatus()
  elseif DALI_P_VIS_UPDATE_SCENE == VIS_DTY1_SCENE_COMMANDS then
    DALI_DTY1_UpdateCommands()
  endif
endsub
//=====================================================

//-------------------------------------------------------------
```

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

5.7 List of global variables for the system

Name	Comment	Type	Value
__vm_cycles_per_s	Ø Zyklen pro Sekunde	num	84.60
__vm_version	Versionsinformationen	str[64]	VM 1.1.11 {vswincevm-arm}
__vm_tcp_stream_rx	TCP Bytes empfangen	str[64]	4.82 K
__vm_tcp_stream_tx	TCP Bytes gesendet	str[64]	263.91 K
__vm_mem_used	Speicherauslastung [%]	num	52.00
__sv_pers_wr_count	Schreibzähler persist. Werte	int	0
MAIN_LOG_P_ENABLED	Logging Ein / Aus	bit	false
MAIN_R_CYCLETIME	Zykluszeit	num	0.00
MAIN_DALI_P_UNIT	Modbus RTU unit Id (adress) of RESI-DALI-MODBUS	int	255
MAIN_DALI_P_ADR	DALI short adress of the device	int	1
MAIN_DALI_R_COM_OK	Error communication	bit	true
MAIN_DALI_R_COM_OK_UNIT	Error communication with RESI-DALI-MODBUS	bit	true
MAIN_DALI_R_COM_OK_DEVICE	Error communication with the device	bit	true
MAIN_DALI_R_COM_ERROR_CNT_UNIT	Error coutervalue communication with RESI-DALI-MODBUS	int	0
MAIN_DALI_R_COM_ERROR_CNT_DEVICE	Error coutervalue communication with the device	int	1
MAIN_DALI_P_COM_ERROR_RESET	to reset Error counter	bit	false
MAIN_DALI_P_VIS_UPDATE_SCENE	Updatevalue for visualisation	int	-1
MAIN_DALI_P_CMD	Command	int	0
MAIN_DALI_P_CMD_PARAM_01	Param for command execution	int	0
MAIN_DALI_P_CMD_PARAM_02	Param for command execution	int	0
MAIN_DALI_R_STATUS	Operating status value	int	0
MAIN_DALI_R_STATUS_CONTROL_GEAR	Operating status [OK;NOK]	bit	false
MAIN_DALI_R_STATUS_FAILURE	Failure status [NO;YES]	bit	false
MAIN_DALI_R_STATUS_ARC_POWER	Operating status [NO;YES]	bit	false
MAIN_DALI_R_STATUS_LIMIT_ERROR	Failure status [NO;YES]	bit	false
MAIN_DALI_R_STATUS_FADE_RUNNING	Operating status [NO;YES]	bit	false
MAIN_DALI_R_STATUS_RESET_STATE	Operating status [NO;YES]	bit	false
MAIN_DALI_R_STATUS_MISSING_SHORTADRESS	Failure status [NO;YES]	bit	false
MAIN_DALI_R_STATUS_POWER_FAILURE	Failure status [NO;YES]	bit	false
MAIN_DALI_R_IS_WORKING	..	int	0
MAIN_DALI_R_IS_FAILURE	..	int	0
MAIN_DALI_R_IS_OPERATING	..	int	0
MAIN_DALI_R_OUT_OF_LIMIT	..	int	0
MAIN_DALI_R_RESET_STATE	..	int	0
MAIN_DALI_R_MISSING_SHORTADRESS	..	int	0
MAIN_DALI_R_VERSION_NUMBER	..	int	0
MAIN_DALI_R_DTR_CONTENT	..	int	0
MAIN_DALI_R_DEVICE_TYP	..	int	0
MAIN_DALI_R_PHYSICAL_MINIMUM	..	int	0
MAIN_DALI_R_POWER_FAILURE_MODE	..	int	0
MAIN_DALI_R_CURRENT_LEVEL	..	int	0
MAIN_DALI_R_CURRENT_LEVEL_100	..	num	0.00

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

MAIN_DALI_R_MAX_LEVEL	..	int	0
MAIN_DALI_R_MIN_LEVEL	..	int	0
MAIN_DALI_R_POWER_UP_LEVEL	..	int	0
MAIN_DALI_R_SYSTEM_FAILURE_LEVEL	..	int	0
MAIN_DALI_R_FADE_TIME_AND_RATE	..	int	0
MAIN_DALI_R_FADE_TIME	..	int	0
MAIN_DALI_R_FADE_RATE	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_00	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_01	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_02	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_03	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_04	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_05	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_06	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_07	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_08	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_09	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_10	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_11	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_12	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_13	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_14	..	int	0
MAIN_DALI_R_LEVEL_OF_SCENE_15	..	int	0
MAIN_DALI_R_PART_OF_GROUP_00_07	..	int	0
MAIN_DALI_R_PART_OF_GROUP_08_15	..	int	0
MAIN_DALI_R_PART_OF_GROUP	..	int	0
MAIN_DALI_R_PART_OF_GROUP_HEX	..	str[64]	
MAIN_DALI_R_RANDOMADRESS_HIGH	..	int	0
MAIN_DALI_R_RANDOMADRESS_MID	..	int	0
MAIN_DALI_R_RANDOMADRESS_LOW	..	int	0
MAIN_DALI_R_RANDOMADRESS	..	int	0
MAIN_DALI_R_RANDOMADRESS_HEX	..	str[64]	
MAIN_DALI_DTY1_P_CMD	DALI_DTY1 Kommandonummer	int	0
MAIN_DALI_DTY1_P_CMD_PARAM_01	Parameter 01 zur Ausführung des DALI_DTY1 Kommandos	int	0
MAIN_DALI_DTY1_R_STATUS	Operating status	int	0
MAIN_DALI_DTY1_R_STATUS_CONTROL_GEAR	Operating status [OK;NOK]	bit	false
MAIN_DALI_DTY1_R_STATUS_FAILURE	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_STATUS_ARC_POWER	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_STATUS_LIMIT_ERROR	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_STATUS_FADE_RUNNING	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_STATUS_RESET_STATE	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_STATUS_MISSING_SHORTADRESS	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_STATUS_POWER_FAILURE	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_IS_WORKING	..	int	0
MAIN_DALI_DTY1_R_VERSION_NUMBER	..	int	0

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

MAIN_DALI_DTY1_R_DTR_CONTENT	..	int	0
MAIN_DALI_DTY1_R_DEVICE_TYP	..	int	0
MAIN_DALI_DTY1_R_PHYSICAL_MINIMUM	..	int	0
MAIN_DALI_DTY1_R_CURRENT_LEVEL	..	int	0
MAIN_DALI_DTY1_R_PART_OF_GROUP_00_07	..	int	0
MAIN_DALI_DTY1_R_PART_OF_GROUP_08_15	..	int	0
MAIN_DALI_DTY1_R_PART_OF_GROUP	..	int	0
MAIN_DALI_DTY1_R_PART_OF_GROUP_HEX	..	str[64]	
MAIN_DALI_DTY1_R_RANDOMADRESS_HIGH	..	int	0
MAIN_DALI_DTY1_R_RANDOMADRESS_MID	..	int	0
MAIN_DALI_DTY1_R_RANDOMADRESS_LOW	..	int	0
MAIN_DALI_DTY1_R_RANDOMADRESS	..	int	0
MAIN_DALI_DTY1_R_RANDOMADRESS_HEX	..	str[64]	
MAIN_DALI_DTY1_R_BATTERIE_CHARGE	Ladezustand der Batterie	int	0
MAIN_DALI_DTY1_R_TEST_TIMING	xxx	int	0
MAIN_DALI_DTY1_R_DURATION_TEST_RESULT	xxx	int	0
MAIN_DALI_DTY1_R_LAMP_EMERGENCY_TIME	xxx	int	0
MAIN_DALI_DTY1_R_LAMP_TOTAL_OPERATION_TIME	xxx	int	0
MAIN_DALI_DTY1_R_EMERGENCY_LEVEL	xxx	int	0
MAIN_DALI_DTY1_R_EMERGENCY_MIN_LEVEL	xxx	int	0
MAIN_DALI_DTY1_R_EMERGENCY_MAX_LEVEL	xxx	int	0
MAIN_DALI_DTY1_R_RATED_DURATION	xxx	int	0
MAIN_DALI_DTY1_R_EMERGENCY_MODE	xxx	int	0
MAIN_DALI_DTY1_R_EMERGENCY_MODE_STANDBY	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_MODE_ON	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_MODE_EMERGENCY_EXTENDED	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_MODE_FUNCTION_TEST	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_MODE_DURATION_TEST	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_MODE_INPUT_LOCK	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_MODE_INPUT_SWITCH	Operating status [OFF;ON]	bit	false
MAIN_DALI_DTY1_R_FEATURES	xxx	int	0
MAIN_DALI_DTY1_R_FEATURES_INCLUDED_DEVICE	Feature avaiable [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FEATURES_PERMANENT_DEVICE	Feature avaiable [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FEATURES_SWITCHED_DEVICE	Feature avaiable [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FEATURES_AUTOTEST_POSSIBLE	Feature avaiable [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FEATURES_ADJUSTABLE_EMERGENCY_LEVEL	Feature avaiable [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FEATURES_INPUT_LOCK_ENABLED	Feature avaiable [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FEATURES_PHYSICAL_SELECTION_ENABLED	Feature avaiable [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FEATURES_STANDBY_RESTART_ENABLED	Feature avaiable [NO;YES]	bit	false

Proprietary data, company confidential. All rights reserved.
Confié à titre de secret d'entreprise. Tous droits réservés.
Comunicado como segredo empresarial. Reservados todos os direitos.
Confidado como secreto industrial. Nos reservamos todos los derechos.

Weitergabe sowie Vervielfältigung dieser Unterlage, Verwertung und Mitteilung ihres Inhalts nicht gestattet, soweit nicht ausdrücklich zugestanden. Zuwiderhandlungen verpflichten zu Schadenersatz. Alle Rechte vorbehalten, insbesondere für den Fall der Patenterteilung oder GM-Eintragung

MAIN_DALI_DTY1_R_FAILURE_STATUS	xxx	int	0
MAIN_DALI_DTY1_R_FAILURE_STATUS_CIRCUIT_BROKEN	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FAILURE_STATUS_DURATION_LOW	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FAILURE_STATUS_BATTERY_BROKEN	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FAILURE_STATUS_LAMP_BROKEN	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FAILURE_STATUS_FUNCTION_TEST_DELAY	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FAILURE_STATUS_DURATION_TEST_DELAY	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FAILURE_STATUS_FUNCTION_TEST	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_FAILURE_STATUS_DURATION_TEST	Failure status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS	xxx	int	0
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_LOCKED	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_FUNCTION_TEST_READY	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_DURATION_TEST_READY	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_BATTERY_LOADED	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_FUNCTION_TEST_PENDING	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_DURATION_TEST_PENDING	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_IDENTIFICATION_ACTIVE	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EMERGENCY_STATUS_PHYSICAL_SELECTED	Operating status [NO;YES]	bit	false
MAIN_DALI_DTY1_R_EXTENDED_VERSION_NUMBER	xxx	int	0