domat
MEMBER OF CEZ ESCO

R091 Modbus TCP / DALI converter

Summary **R091 is a serial converter which acts as a Modbus TCP server (accepts Modbus TCP commands) and web server and controls a DALI (Digital Addressable Light Interface) bus with up to 64 DALI devices. R091 is a multi master of the DALI bus and provides the power supply for the bus. It also incorporates a web interface for manual entering of DALI commands including bus configuration and diagnostics commands. The R091 fully covers the functionality of previous types M090 and R090 and now supports standard DALI v2.**

Applications

- **integration of DALI light controllers to a Modbus TCP compatible SCADA or PLC**
- **configuration and control of DALI bus over a comfortable web interface, even on a remote basis**
- **DALI bus control over CGI commands**

Function

R091 acts as a DALI bus multimaster controller (according to EN 60929 ed. 4:2011 Annex E, static priority 4, setting time 16 ms, retry timeout 300 ms), and provides the power supply for the bus, see below. The Modbus and web commands are translated into DALI protocol telegrams and sent to the DALI devices. The responses from the light controllers are translated back to Modbus registers and available at the corresponding addresses, see tables below.
Another way is to use CGI commands where all web functions may be also communicated to the device as CGI requests.

Compatibility

The R091 converter was designed and tested so as to comply with the standard EN 60929. Compatibility with this norm is guaranteed e.g. by DALI Compliant certification. It is recommended to use only DALI Compliant certified ballasts and other devices. There are also components on the market marked as DALI Compatible. DALI Compatible means that the product uses DALI technology, but may not support all functions according to the standard, and it may not have been tested using methods described in

the standard. A DALI Compatible device will most probably be working correctly together with other products by the same manufacturer, but it may not work as expected with other manufacturers' products. We can not guarantee complete functionality with DALI Compatible devices.

Design notes

The DALI bus supports up to 64 light ballast addresses, up to 16 scenes, and up to 16 groups. Wire length and diameter must always be respected! For the complete overview of the DALI bus specification, see the EN 60929. The DALI bus uses 22.5 V operation voltage.

The bus devices are connected over a 2-pole connector, regardless of polarity. The Ethernet is connected over RJ45 connector with PoE (Power over Ethernet).

When specifying the DALI bus load, the amount and types of the DALI components must be selected so as not to exceed the guaranteed DALI bus current of the power supply. Using the **single master** topology, up to 64 control devices may be installed. A **multimaster** bus load shall not exceed the maximum total current for all devices (incl. input devices and application controllers) of 250 mA.

At the place of installation, heat dissipation must be guaranteed. Please do not exceed the maximum permitted working temperature of 45 °C, otherwise the proper function is not guaranteed and the converter could be damaged.

Technical data

Power supply	24V AC/DC ±20 %, 6W (G, G0, TE)
	or PoE (Power over Ethernet, 802.3af class 2 (36-57 V)
Consumption	1 VA (no DALI load) ... max. 6 VA (full load, 64 DALI slaves)
Communication	
Ethernet	1x Ethernet 10/100BaseT; galvanically insulated to 1 kV
	RJ45, 2 LED (link, data) integrated in the connector
DALI	standard EN 60929 ed. 4:2011 Annex E, 1200 bps
	The R091 is according to this standard a bus power supply and application controller - multimaster (collision avoidance/ detection, priority 4 setting time 16 ms, retry timeout 300 ms).
	Power supply with digital stabilizer and guaranteed accuracy 1 % over the whole range.
	Galvanic separation DALI bus is separated up to 1000 V
	Short circuit protection of DALI power supply electronic with automatic reset, short-circuit current I_k= 250 mA
	Overload sustainability of the DALI power supply - sustainable to unlimited bus short-circuit.
	Guaranteed current according to EN 62386-101: 250 mA
	Fully compliant with EN 62386-101 ed2:2015 incl. non-standard user profiles (8/16/24/25 bits)
4x LED	RUN, PWR, TxD, RxD
Housing	Polycarbonate box (certification UL94V0); 4U

Dimensions	See below
Terminals	Screw terminals M3, maximum wire cross-section 2,5 mm^2
Protection degree	IP20 (EN 60529)
Operating environment	
Ambient conditions	-5…45 °C; 5…95 % relative humidity; non-condensing gases and chemically non-aggressive conditions (according EN 60721-3-3 climatic class 3K3)
Storage conditions	-5…45 °C; 5…95 % relative humidity; non-condensing gases and chemically non-aggressive conditions (according EN 60721-3-1 climatic class 1K3)
Standards conformity	EMC EN 61000-6-2 ed.3:2005, EN 55022 ed.3:2010 electrical safety EN 60950-1 ed.2:2006 + A11:2009 + A12:2011 + A1:2010 + A2:2014 + Opr.1:2012 + Z1:2016 RoHS EN 50581:2012

Power supply

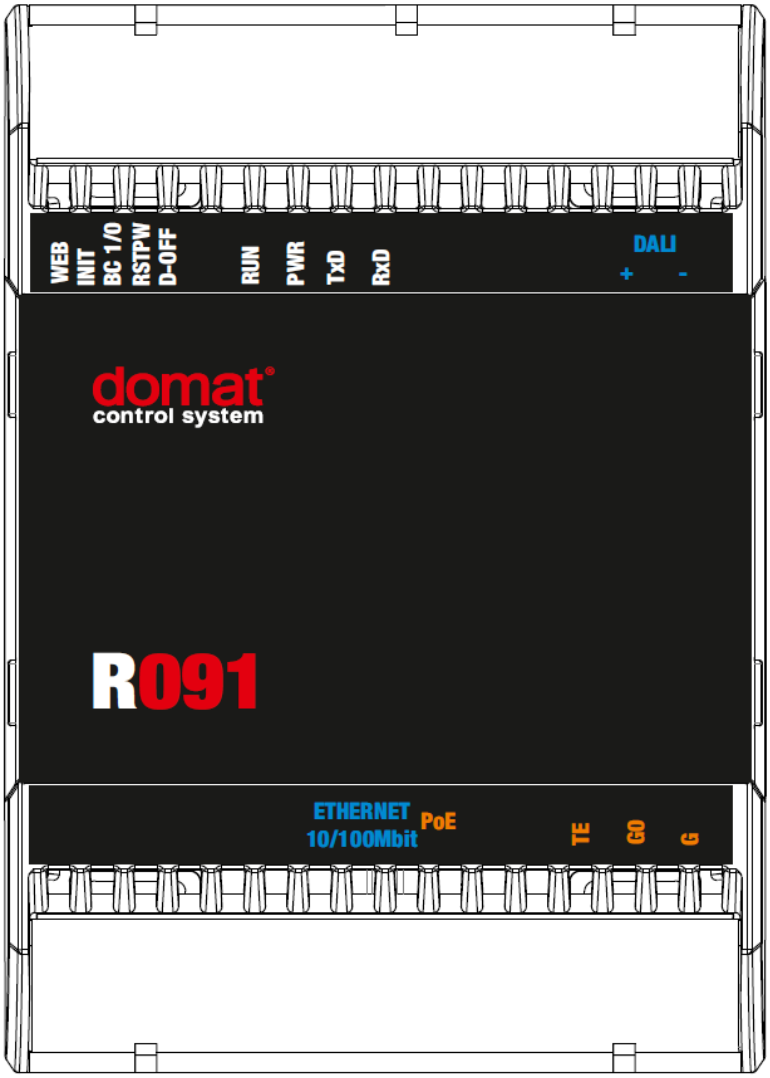
Alternative power supply (G/G0 terminals vs. PoE):

1. If the G/G0 power is applied first, the R091 is powered from this G/G0 external source. At power failure the power is switched over to PoE with a short dropout (device reset).

2. If the PoE power is applied first, the R091 is powered from the PoE. The switchover to G/G0 follows only if the G/G0 voltage is 27 V DC (19 V AC) and above.

3. If both G/G0 and PoE are applied at the same time, the R091 is powered from G/GO terminals. The device will not be damaged.

Schema



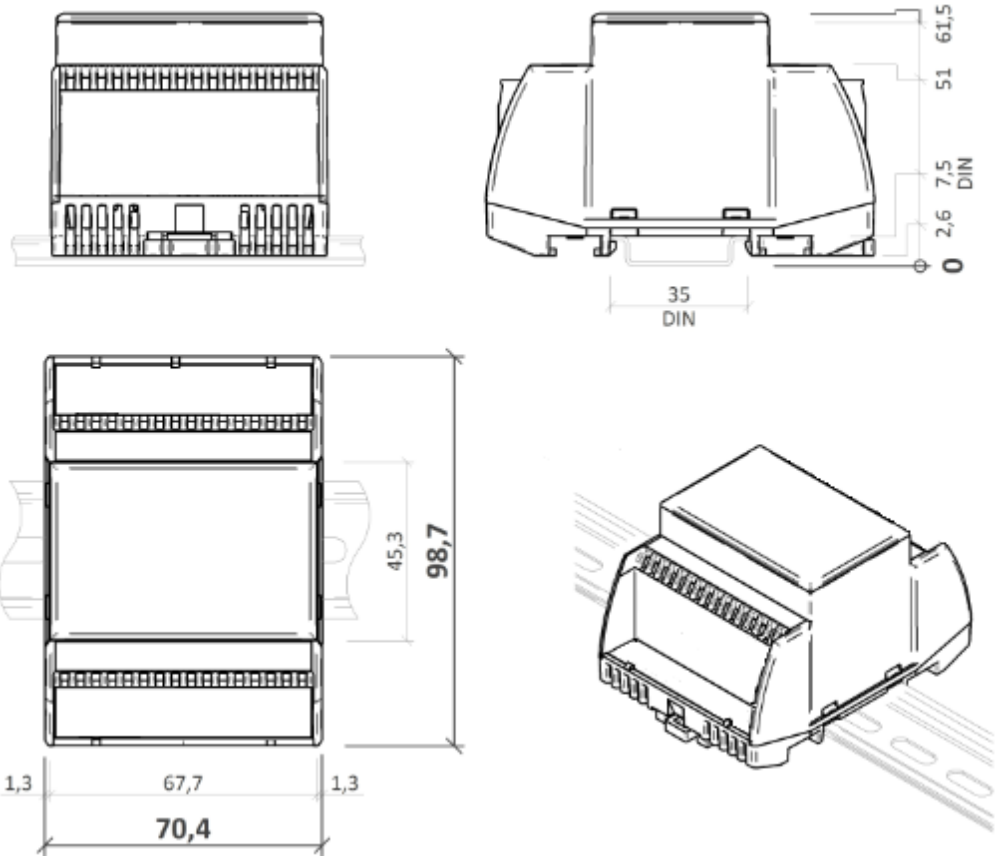
Terminals and connectors	
DALI	DALI bus, positive and negative
Ethernet, PoE	Network interface, PoE
G	power
G0	power, reference terminal
TE	optional connection for shielding

LED indication	
RUN	yellow LED – system cycle (OK: LED flashes periodically 1 s ON, 1 s OFF; ERROR: LED flashes in other pattern, LED is permanently ON or OFF)
PWR	green LED – power (ON: power OK; OFF: no power applied, weak or damaged power supply, …)
RxD	green LED – receiving data at DALI interface (flashing: receiving data; OFF: no data traffic)
TxD	red LED – transmitting data at DALI interface (flashing: transmitting data; OFF: no data traffic)
LINK/DATA	Ethernet activity

DIP switches

WEB	If ON at power-up, web, CGI, and FTP access is denied; set the switch to OFF and power cycle to enable web again.
INIT	If ON at power-up, configuration parameters are brought to defaults (see below); values are not saved to EEPROM
BC 1/0	When switched from OFF to ON, a central SET TO MAX command (broadcast) is sent, when switched from ON to OFF, a central OFF command is sent. For emergency manual control and testing.
RSTPW	Not used here.
D-OFF	When ON, the DALI bus power supply is disabled. Used if another DALI power supply is installed on the bus. The setting corresponds to the position of the DIP switch at power-up.

Dimensions



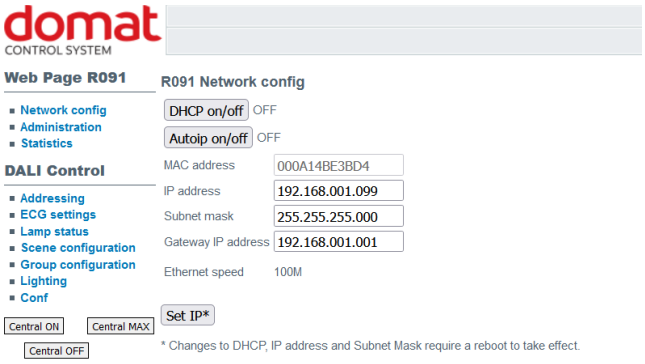
Dimensions are in *mm*.

Settings

The network properties are set over the R091's web interface. The default network settings are:

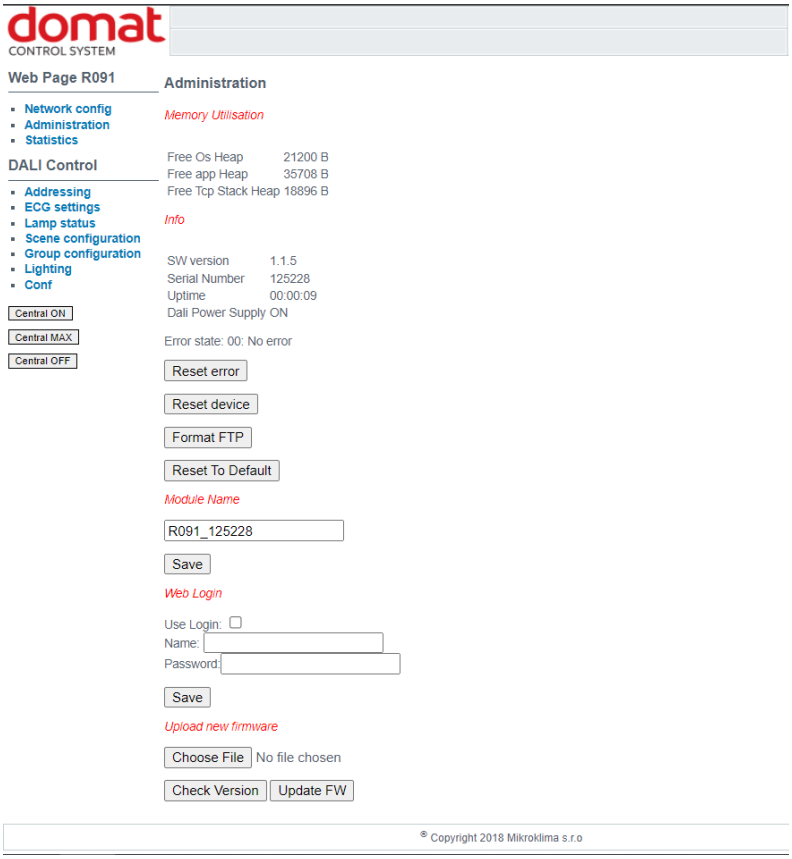
IP address	192.168.1.99
Network mask	255.255.255.0
Default gateway	192.168.1.1

All settings are stored in EEPROM.



Bringing the device to default settings:

1. Power off the R091.
2. Set the DIP switch 2 (INIT) to ON.
3. Apply power.
4. Locate the device at its default IP address and configure it as necessary.
5. Remove power.
6. Set the INIT switch to OFF.
7. Apply power again.
8. The R091 has the new settings.

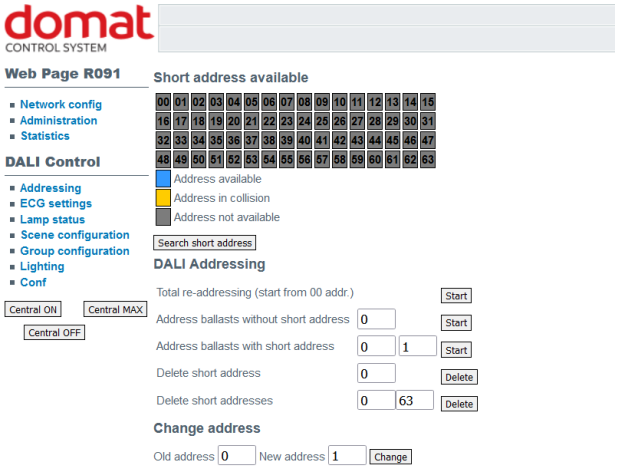


Administration

Web Login: Access to converter web interface (from fw v. 1.1.5 onward) can be limited with *username* and *password*.
Upload new firmware: New firmware can be chosen by clicking on button Choose file and then uploaded by *Update FW* button.

Web interface

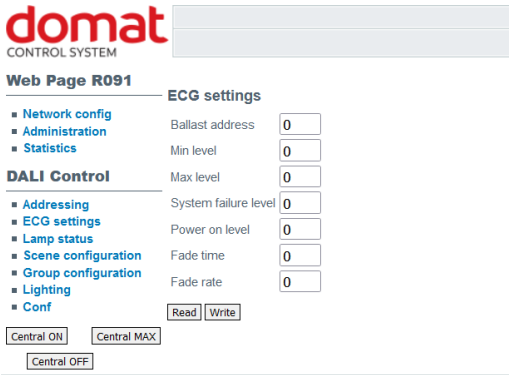
Over the web interface it is possible to set the R091 up (*Network config*), upload new firmware if necessary (*Administration*), and diagnose the interface (*Statistics*). The DALI Control menu is used to test if the DALI part is operating properly, to address the ballasts (*Addressing*), set the individual ballast parameters (*ECG settings*), see the states of the lamps at a glance (*Lamp status*), configure scenes and groups, issue group commands (*Ligthing*), and enable registers for simple control (*Conf*).



Addressing

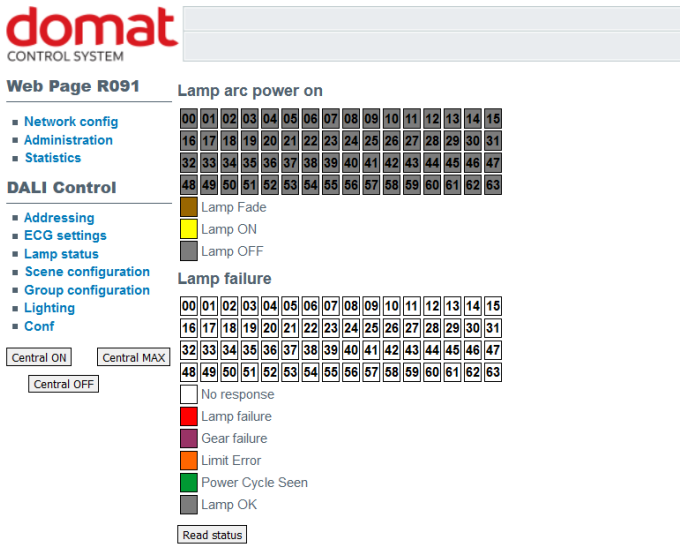
Total readdressing: Regardless of existing addressing, all ballasts will be readdressed starting from zero
Address ballast without short address: Enter the address (e.g. 1) to start with for addressing of ballasts without short address
Address ballast with short address: Enter the old address to start with, and the new starting address which replaces the old range
Delete short address: Enter the short address to delete (this ballast will have no short address after the operation is finished)
Delete short addresses: Enter the range of short addresses to delete (these ballasts will have no short address after the operation is finished)
Change address: Enter the address to change and new address

Please note that the numbering of the ballasts is:
0...63 at the web interface
0...63 in the Modbus telegrams.



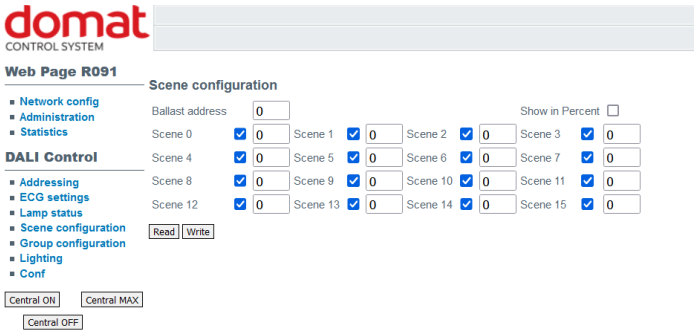
ECG Settings menu

In this menu, the parameters of a particular ballast can be read and set.
Ballast address: Set the ballast address to be configured.
Min level: Value (0...255) to set the ballast when receiving the Min level command
Max level: Value (0...255) to set the ballast when receiving the Max level command
System failure level: Value to set the ballast at DALI bus failure (e.g. short-circuit)
Power on level: Value to set the ballast between power on and receiving the first DALI command
Fade time: Time for change of value, see DALI standard
Fade rate: Frequency of changes (in steps/s), see DALI standard



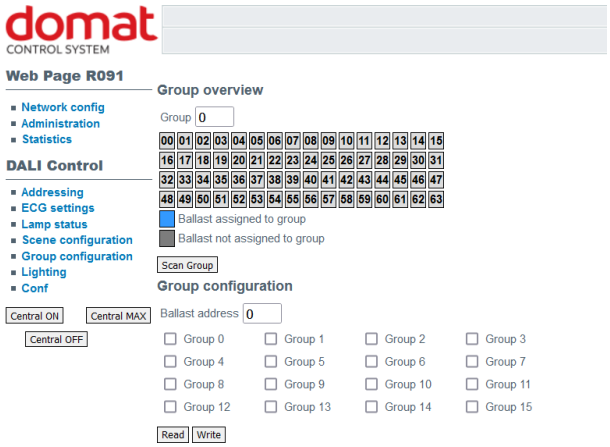
Lamp status – menu for ballast diagnostics

Lamp fade: Ballast is dimming
Lamp ON: Light is on (more than 0 %)
Lamp OFF: Light is off
Lamp failure table: DALI error according to DALI specification
Read status: Starts reading of status of all ballasts.



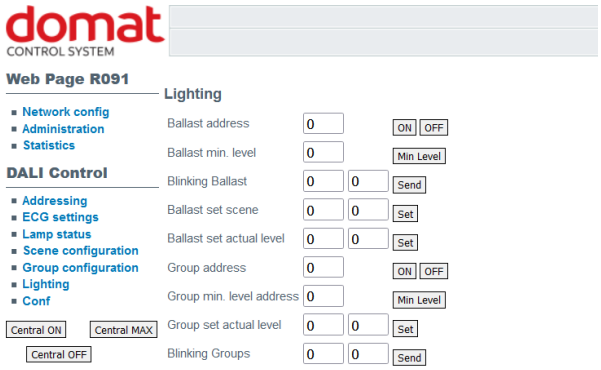
Scene configuration – menu for reading and configuration of scenes

Ballast address: Enter the ballast address to be configured
Read: Reads current setting of scenes in the ballast
(at scenes) tick box: a scene is defined, **number**: intensity for this scene
Show in Percent: Intensity is entered and displayed in % of a logarithmic scale, which fits better the human eye sensitivity, rather than as 0...255
Write: Saving of all scene settings to a ballast.



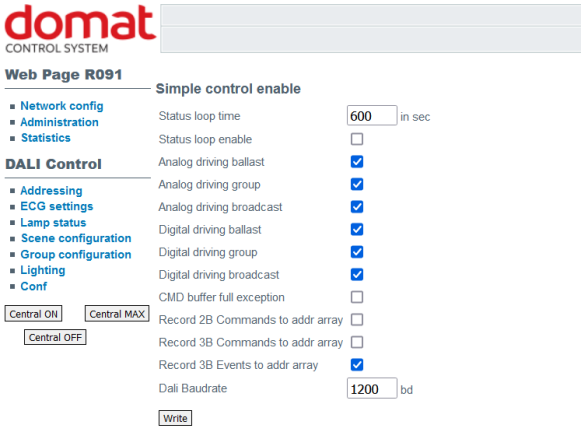
Group configuration

Group: Enter the group number which shall be displayed in the assignment below
Scan group: Displays ballasts which are assigned to the entered group
Ballast address: Enter the ballast address for which groups will be configured
Read: Reads assignment to groups for this address
Group x: tick = ballast is assigned to this group
Write: Writes the group assignment into the ballast



Lighting – Menu to control ballasts

Ballast address: Address to switch on (**ON**) or off (**OFF**)
Ballast min. level: Address to set to minimum level (**Min. Level**)
Blinking Ballast: Address and number of flashes. Activate by clicking **Send**
Ballast set scene: Address and scene number to be activated
Ballast set actual level: Address and light level to be set
Group address: Address of group to switch on (**ON**) or off (**OFF**)
Group min. level: Address of group to set to minimum level (**Min. Level**)
Group set actual level: Address of group and light level to be set
Blinking Groups: Address of group and number of flashes. Activate by clicking **Send**.



Conf – Menu to enable simple control functions

Status loop time: Period time to request ballast status. These requests bring substantial load to the bus, so it is recommended to keep the loop time as long as possible
Status loop enable: Enable ballast status requests.
Analog driving... / Digital driving...: Enable direct control of ballasts, groups and scenes over Modbus registers 95 to 182. Because of DALI bus load and possible collision of commands please only enable those commands which will be used.
CMD buffer full exception: When the buffer is full (and DALI commands are discarded), the R091 responds to a Modbus telegram with Modbus error 06 (Slave device busy). The client learns over Modbus that commands have been discarded. Should the Modbus error messages cause communication problems at the client side, do not check this option, and the telegrams will be confirmed correctly even when discarded. The number of discarded commands is listed in the *Statistics* menu.
Record 2B commands to addr array: All commands from application controllers to ballasts will be recorded in registers 601 and on.
Record 3B commands to addr array: All commands for configuration of input devices (switches, sensors, ...), will be recorded in registers 601 and on.
Record 3B events to addr array: All events sent by input devices will be recorded in registers 601 and on (typically pushing of a switch or light intensity change).
(In registers 801 and on, all telegrams are always recorded.)
Dali baudrate: Possibility to correct the baudrate for non-standard ballast types. Do not change if not necessary!

The web interface is useful for system commissioning: the DALI bus may be configured independently from the PLC program. As soon as the groups are configured and it is possible to control the DALI ballasts over the web interface, it is time to commission the PLC part.

When updating the firmware, update of the internal web pages may be necessary. The files are part of the firmware release package. For upload, use a FTP client with username/password: root / root99.

In the SoftPLC IDE, there is a special device for R091 so that the integration is easy – it is not necessary to map the Modbus registers via a generic Modbus driver, there are dedicated variables to control central commands, groups, scenes, as well as individual ballasts.

In the Merbon IDE, there are Modbus devices for direct commands through writing to Modbus registers.

Installation

Please ensure that when installed onto a DIN rail there is at least a 15 mm gap for air circulation at both vertical device sides, which is necessary for proper cooling of the device.

Modbus TCP communication

The supported Modbus functions are:

03 Read Holding Registers – read words
16 Force Multiple Registers – write words

The memory is divided into sections, and only registers which functionally stick together, like 95 to 158, 160 to 175, etc. are able to be written in a single command. In other words, the F16 command can not cross borders, like writing to registers 157 to 162 in one command.

The address space is accessible wordwise (16 bit words). See table below**.**

Tab. 1: Modbus table

Name	Address	Type	Description	Note
modul LSB	1 LSB	R	module ID lower byte	0x0191 hex
modul MSB	1 MSB	R	module ID upper byte	
firmware LSB	2 LSB	R	firmware version lower byte	
firmware MSB	2 MSB	R	firmware version upper byte	
	3 LSB	R	reserved	
status MSB	3 MSB	R	module status upper byte **bit 0** – 0: normal mode - 1: init mode **bit 1** – 1: at the next writing to any EEPROM registers, **all EEPROM data** will be written to EEPROM 0: data will only be written to RAM (changes will be lost after device restart) **bit 2 –** 1: web access blocked (WEB switch is ON) **bit 3** – 1**:** DALI FIFO full **bit 4** – 1: DALI bus shortcircuit detected, bus power is off **bit 5** – 1: DALI power supply disabled **bit 6** - 0 **bit 7** - 1	
Send twice mask	4 LSB	R, W RAM	**bit 0** = block 0 **bit 1** = block 1 **bit 2** = block 2 **bit 3** = block 3 **bit 4** = block 4 **bit 5** = block 5 **bit 6** = block 6 **bit 7** = block 7	**bit_x** = 1 -> send DALI command from **block x** twice within 100ms; works for all commands except for commands 32-128, 258 and 259, which are sent twice automatically set/reset of bit is done only by user
reserved	4 MSB	R RAM		
command mask	5 LSB	R,W RAM	**bit 0** = block 0 **bit 1** = block 1 **bit 2** = block 2 **bit 3** = block 3 **bit 4** = block 4 **bit 5** = block 5 **bit 6** = block 6 **bit 7** = block 7	By setting the bit, executing of the corresponding block is enabled. The device executes the enabled blocks one after another from **bit 7** to **bit 0**
command executed	5 MSB	R RAM		The set bit indicates the executed block, **bit 0** = **block 0** etc.
DALI command block 0	6 LSB	R,W RAM		DALI command for block 0 according to the tables
DALI address block 0	6 MSB	R,W RAM		DALI address for block 0

D0 block 0	7 LSB	R,W RAM	if the request is performed by a single DALI command which contains answer, the answer is in this block	additional data 0 for block 0 -> tables
D1 block 0	7 MSB	R,W RAM	If the request is performed by a single DALI command, then: **0x00** – no reply returned **0x55** – valid DALI reply returned, and stored in register **7LSB** **0x02 –** bus error **0x03** – DALI reply returned but damaged (data unrecognized) **0x04** - bus permanently short-circuited **0x05** – device is busy	***Note 1*** additional data 1 for block 0 -> tables
D2 block 0	8 LSB	R,W RAM		additional data 2 for block 0 -> tables
Trigger block 0	8 MSB	R,W RAM	By setting value of bit 0 from log 0 to log 1 the command will be executed. The value remains unchanged after command is finished, must be reset by the Modbus client. Reg. 5 indicates command in queue /executed command.	The command is equivalent to command in the corresponding bit of register 5 LSB.
DALI command block 1	9 LSB	R,W RAM		
DALI address block 1	9 MSB	R,W RAM		
D0 block 1	10 LSB	R,W RAM		
D1 block 1	10 MSB	R,W RAM		
D2 block 1	11 LSB	R,W RAM		
Trigger block 1	11 MSB	R,W RAM		
DALI command block 2	12 LSB	R,W RAM		
DALI address block 2	12 MSB	R,W RAM		
D0 block 2	13 LSB	R,W RAM		
D1 block 2	13 MSB	R,W RAM		
D2 block 2	14 LSB	R,W RAM		
Trigger block 2	14 MSB	R,W RAM		
DALI command block 3	15 LSB	R,W RAM		
DALI address block 3	15 MSB	R,W RAM		
D0 block 3	16 LSB	R,W RAM		

D1 block 3	16 MSB	R,W RAM		
D2 block 3	17 LSB	R,W RAM		
Trigger block 3	17 MSB	R,W RAM		
DALI command block 4	18 LSB	R,W RAM		
DALI address block 4	18 MSB	R,W RAM		
D0 block 4	19 LSB	R,W RAM		
D1 block 4	19 MSB	R,W RAM		
D2 block 4	20 LSB	R,W RAM		
Trigger block 4	20 MSB	R,W RAM		
DALI command block 5	21 LSB	R,W RAM		
DALI address block 5	21 MSB	R,W RAM		
D0 block 5	22 LSB	R,W RAM		
D1 block 5	22 MSB	R,W RAM		
D2 block 5	23 LSB	R,W RAM		
Trigger block 5	23 MSB	R,W RAM		
DALI command block 6	24 LSB	R,W RAM		
DALI address block 6	24 MSB	R,W RAM		
D0 block 6	25 LSB	R,W RAM		
D1 block 6	25 MSB	R,W RAM		
D2 block 6	26 LSB	R,W RAM		
Trigger block 6	26 MSB	R,W RAM		
DALI command block 7	27 LSB	R,W RAM		
DALI address block 7	27 MSB	R,W RAM		
D0 block 7	28 LSB	R,W RAM		
D1 block 7	28 MSB	R,W RAM		
D2 block 7	29 LSB	R,W RAM		
Trigger block 7	29 MSB	R,W RAM		

enable functions for simple control	30 LSB, MSB	R,W EEPROM default 0x7F hex (all enabled)	**bit0** – enable round for error states and status readout **bit1** – enable analogue intensity control – ballasts **bit2** – enable analogue intensity control - groups **bit3** – enable analogue intensity control - broadcast **bit4** – enable bit (on/off) control - ballasts **bit5** – enable bit (on/off) control – groups **bit6** – enable bit (on/off) control broadcast (central on/off) **bit7** – enable Modbus exception handler **bit8** – enable record 2B dali events in address event log (601 → 793) **bit9** – enable record 3B dali events in address event log (601 → 793) **bit10** – enable record 3B dali events in address event log (601 → 793)	
status of ballast 0	31 LSB	R RAM	**bit 0** - Status of ballast; "0" = OK **bit 1** -Lamp failure; "0" = OK **bit 2** - Lamp arc power on; "0" = OFF **bit 3** - Query: Limit Error; "0" = Last requested arc power level is between MIN..MAX LEVEL or OFF **bit 4** - Fade ready; "0" = fade is ready; "1" = fade is running **bit 5** - Query: "RESET STATE"? "0" = "No" **bit 6** - Query: Missing short address? "0" = "No" **bit 7** - Query: "POWER FAILURE"? "0" = "No"; "RESET" or an arc power control command has been received since last power-on. The "STATUS INFORMATION" shall be available in the RAM of the ballast and shall be updated regularly by the ballast according to the actual situation. The responses are same as command 144 responses from the standard DALI table.	See DALI standard.
status of ballast 0	31 MSB	R RAM	If **bit x = 1** **bit 0** - ballast comm failure **bit 1** - DALI bus busy **bit 2** - data frame error **bit 3** - DALI answer timeout **bit 4** - DALI short circuit **bit 5** - LPC processor comm timeout **bit 6** - FIFO full **bit 7** - repeated error	
status of ballast 1	32 LSB	R RAM	See status of ballast 0, LSB	
status of ballast 1	32 MSB	R RAM	See status of ballast 0, MSB	

…	…	…	…	…
status of ballast 63	94 LSB	R RAM	See status of ballast 0, LSB	
status of ballast 63	94 MSB	R RAM	See status of ballast 0, MSB	
ballast 0 intensity	95 LSB, MSB	R,W RAM	Analogue intensity value for ballast 0 (0...254). The telegram is sent on Modbus writing. Function must be enabled in register 30 bit 1.	***Note 2***
ballast 1 intensity	96 LSB, MSB	R,W RAM		
ballast 2 intensity	97 LSB, MSB	R,W RAM		
...	...	....		
ballast 63 intensity	158 LSB, MSB	R,W RAM		
error and status readout round trip	159 LSB, MSB	R,W EEPROM	Value is in sec. (0 – 65535). If the value is 0, status and error are not read.	default 600 sec
group intensity 0	160 LSB, MSB	R,W RAM	Analogue value of group 0 intensity (0-254). The telegram is sent on Modbus writing. Function must be enabled in register 30 bit 2.	
group intensity 1	161 LSB, MSB	R,W RAM		
group intensity 2	162 LSB, MSB	R,W RAM		
...	...	...		
group intensity 15	175 LSB, MSB	R,W RAM	see above	
analogue broadcast value	176 LSB, MSB	R,W RAM	Analogue value of all ballasts intensity (0-254) – central command	
bit control of ballasts 0-15	177 LSB, MSB	R,W RAM	0– switch off 1– switch on The telegram is sent on Modbus writing. Function must be enabled in register 30 bit 4.	**bit 0** –ballast 0 **bit 1** – ballast 1 **bit 2** – ballast 2 ...
bit control of ballasts 16-31	178 LSB, MSB	R,W RAM		
bit control of ballasts 32-47	179 LSB, MSB	R,W RAM		
bit control of ballasts 48-63	180 LSB, MSB	R,W RAM		

bit control of groups 0-15	181 LSB, MSB	R,W RAM	0– switch off 1– switch on The telegram is sent on Modbus writing. Function must be enabled in register 30 bit 5. Note that all groups are sent in each telegram.	***Note 3*** **bit 0** – group 0 **bit 1** – group 1 **bit 2** – group 2 etc.
bit broadcast control	182 LSB, MSB	R,W RAM	The telegram is sent on Modbus writing. Function must be enabled in register 30 bit 6.	**bit 0** – 0 = central OFF, 1 = central ON
DALI baudrate	183 LSB, MSB	R,W, EEPROM	DALI baudrate correction setting Default value is 1200 bps. Only change if ballasts require minor baudrate correction.	1056...1440 bps
reserved	184 - 500			
status	501 LSB, MSB	R, RAM	see reg. 3	mirror of register 3
reserved	502 LSB, MSB			
uptime LSW	503 LSB, MSB	R, RAM	device uptime in 0.1 s, LSW	
uptime MSW	504 LSB, MSB	R, RAM	device uptime in 0.1 s, MSW	
reserved	505 - 510			
command mask 2	511 LSB, MSB	R,W RAM	**bit 0** = block 8 **bit 1** = block 9 **bit 2** = block 10 **bit 3** = block 11 **bit 4** = block 12 **bit 5** = block 13 **bit 6** = block 14 **bit 7** = block 15	By setting the bit (rising edge), executing of the corresponding block is enabled. The module executes the enabled blocks one after another from **bit 7** to **bit 0.** Blocks 8 to 15 are used especially for sending commands of non-standard length. Unlike in reg. 5, there is no automatic reset of reg. 511 after command execution.
command 2 busy	512 LSB, MSB	R, RAM	**bit 0** = block 8 **bit 1** = block 9 **bit 2** = block 10 **bit 3** = block 11 **bit 4** = block 12 **bit 5** = block 13 **bit 6** = block 14 **bit 7** = block 15	A bit indicates a currently processed block. The Modbus master shall write into reg. 511, and then read regs. 511 and 512. Log. 1 in reg. 512 indicates that the command has been queued and is being executed. As soon as the corresponding bit in reg. 512 is reset, the command has been completed. At the same time, the RX area is updated (**dali RX data 0** to **dali RX num of bits**, reg. 561 and on). The Modbus master may then get the R091 ready for another execution of the block by writing 0 to reg. 511.

reserved	513 - 515			
dali TX data 0 block 8	516 LSB	R,W RAM	First byte of transmitted DALI message, block 8	
dali TX data 1 block 8	516 MSB	R,W RAM	Second byte of transmitted DALI message, block 8	
dali TX data 2 block 8	517 LSB	R,W RAM	Third byte of transmitted DALI message, block 8	
dali TX data 3 block 8	517 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 8	Value of 0x80 sends 1 in 25. bit
dali TX flags block 8	518 LSB	R,W RAM	TX attributes for the transmitted message, block 8	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)
dali TX No.of bits block 8	518 MSB	R,W RAM	No. of bits of the transmitted message, block 8	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	519 LSB			
reserved	519 MSB			
dali TX data 0 block 9	520 LSB	R,W RAM	First byte of transmitted DALI message, block 9	
dali TX data 1 block 9	520 MSB	R,W RAM	Second byte of transmitted DALI message, block 9	
dali TX data 2 block 9	521 LSB	R,W RAM	Third byte of transmitted DALI message, block 9	
dali TX data 3 block 9	521 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 9	Value of 0x80 sends 1 in 25. bit
dali TX flags block 9	522 LSB	R,W RAM	TX attributes for the transmitted message, block 9	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)
dali TX No.of bits block 9	522 MSB	R,W RAM	No. of bits of the transmitted message, block 9	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	523 LSB			
reserved	523 MSB			
dali TX data 0 block 10	524 LSB	R,W RAM	First byte of transmitted DALI message, block 10	
dali TX data 1 block 10	524 MSB	R,W RAM	Second byte of transmitted DALI message, block 10	

dali TX data 2 block 10	525 LSB	R,W RAM	Third byte of transmitted DALI message, block 10	
dali TX data 3 block 10	525 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 10	Value of 0x80 sends 1 in 25. bit
dali TX flags block 10	526 LSB	R,W RAM	TX attributes for the transmitted message, block 10	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)
dali TX No.of bits block 10	526 MSB	R,W RAM	No. of bits of the transmitted message, block 10	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	527 LSB			
reserved	527 MSB			
dali TX data 0 block 11	528 LSB	R,W RAM	First byte of transmitted DALI message, block 11	
dali TX data 1 block 11	528 MSB	R,W RAM	Second byte of transmitted DALI message, block 11	
dali TX data 2 block 11	529 LSB	R,W RAM	Third byte of transmitted DALI message, block 11	
dali TX data 3 block 11	529 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 11	Value of 0x80 sends 1 in 25. bit
dali TX flags block 11	530 LSB	R,W RAM	TX attributes for the transmitted message, block 11	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)
dali TX No.of bits block 11	530 MSB	R,W RAM	No. of bits of the transmitted message, block 11	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	531 LSB			
reserved	531 MSB			
dali TX data 0 block 12	532 LSB	R,W RAM	First byte of transmitted DALI message, block 12	
dali TX data 1 block 12	532 MSB	R,W RAM	Second byte of transmitted DALI message, block 12	
dali TX data 2 block 12	533 LSB	R,W RAM	Third byte of transmitted DALI message, block 12	
dali TX data 3 block 12	533 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 12	Value of 0x80 sends 1 in 25. bit

dali TX flags block 12	534 LSB	R,W RAM	TX attributes for the transmitted message, block 12	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)
dali TX No.of bits block 12	534 MSB	R,W RAM	No. of bits of the transmitted message, block 12	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	535 LSB			
reserved	535 MSB			
dali TX data 0 block 13	536 LSB	R,W RAM	First byte of transmitted DALI message, block 13	
dali TX data 1 block 13	536 MSB	R,W RAM	Second byte of transmitted DALI message, block 13	
dali TX data 2 block 13	537 LSB	R,W RAM	Third byte of transmitted DALI message, block 13	
dali TX data 3 block 13	537 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 13	Value of 0x80 sends 1 in 25. bit
dali TX flags block 13	538 LSB	R,W RAM	TX attributes for the transmitted message, block 13	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)
dali TX No.of bits block 13	538 MSB	R,W RAM	No. of bits of the transmitted message, block 13	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	539 LSB			
reserved	539 MSB			
dali TX data 0 block 14	540 LSB	R,W RAM	First byte of transmitted DALI message, block 14	
dali TX data 1 block 14	540 MSB	R,W RAM	Second byte of transmitted DALI message, block 14	
dali TX data 2 block 14	541 LSB	R,W RAM	Third byte of transmitted DALI message, block 14	
dali TX data 3 block 14	541 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 14	Value of 0x80 sends 1 in 25. bit
dali TX flags block 14	542 LSB	R,W RAM	TX attributes for the transmitted message, block 14	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)

dali TX No.of bits block 14	542 MSB	R,W RAM	No. of bits of the transmitted message, block 14	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	543 LSB			
reserved	543 MSB			
dali TX data 0 block 15	544 LSB	R,W RAM	First byte of transmitted DALI message, block 15	
dali TX data 1 block 15	544 MSB	R,W RAM	Second byte of transmitted DALI message, block 15	
dali TX data 2 block 15	545 LSB	R,W RAM	Third byte of transmitted DALI message, block 15	
dali TX data 3 block 15	545 MSB	R,W RAM	Fourth byte of transmitted DALI message, block 15	Value of 0x80 sends 1 in 25. bit
dali TX flags block 15	546 LSB	R,W RAM	TX attributes for the transmitted message, block 15	**bit 0** ... send twice **bits 1 to 3** ... send priority (0 = default - priority 5; value of 1 to 5 = priority 1 to 5, i.e. setting time 13,5 to 19,5 ms)
dali TX No.of bits block 15	546 MSB	R,W RAM	No. of bits of the transmitted message, block 15	0 ... 16 bits 1 ... 24 bits 2 ... 8 bits 3 ... 25 bits
reserved	547 LSB			
reserved	547 MSB			
reserved	548 ... 560			
dali RX data 0 block 8	561 LSB	R, RAM	first byte of received DALI reply, block 8; valid after the corresponding bit in **command 2 finished** is set	in this version, only 8 bit replies are supported
reserved	561 MSB			
dali RX flags block 8	562 LSB	R, RAM	attributes for the sent message and received reply, block 8; valid after the corresponding bit in **command 2 finished** is set	**bit 0** ... cmd successfully sent **bit 1** ... valid reply received **bit 2** ... bus busy (message could not be sent until timeout) **bit 3** ... answer violation (the reply was received as damaged) **bit 4** ... timeout no_answer (no reply received)
dali RX num of bits block 8	562 MSB	R, RAM	number of bits of the received reply, block 8; valid after the corresponding bit in **command 2 finished** is set	number = number of bits, 0 = no or damaged reply 8 = valid reply
reserved	563 LSB, MSB			
dali RX data 0 block 9	564 LSB	R, RAM	first byte of received DALI reply, block 9; valid after the corresponding bit in **command 2 finished** is set	in this version, only 8 bit replies are supported

...	...	...	...	
dali RX num of bits block 15	583 MSB	R, RAM	number of bits of the received reply, block 15; valid after the corresponding bit in **command 2 finished** is set	number = number of bits, 0 = no or damaged reply 8 = valid reply
reserved	584 - 600			
input device 0 event byte 0	601 LSB	R, W, RAM	event from the input device 0 (button, sensor), byte 0 (MSB)	
input device 0 event byte 1	601 MSB	R, W, RAM	event from the input device 0 (button, sensor), byte 1	
input device 0 event byte 2	602 LSB	R, W, RAM	event from the input device 0 (button, sensor), byte 2 (LSB)	
input device 0 event counter	602 MSB	R, W, RAM	ring counter of events of input device 0	every incoming telegram from this input device increases the counter value; after reaching 255 it goes to 0, 1, ...
reserved	603 LSB, MSB			
input device 1 event byte 0	601 LSB	R, W, RAM	event from the input device 1 (button, sensor), byte 0 (MSB)	
...	...	...	...	
input device 63 event counter	791 MSB	R, W, RAM	ring counter of events of input device 63	
reserved	792			
input Broadcast event byte 0	793 LSB	R, W, RAM	event with broadcast address (button, sensor), byte 0-MSB	
input Broadcast event byte 1	793 MSB	R, W, RAM	event with broadcast address (button, sensor), byte 1-MSB	
input Broadcast event byte 0	794 LSB	R, W, RAM	event with broadcast address (button, sensor), byte 2-LSB	
input Broadcast event counter	794 MSB	R, W, RAM	See reg. input broadcast event counter	
reserved	795 - 799			
Event log index	800 LSB	R, RAM	Shows position of the latest record in the array addressed 801 ... 1001. The value is in range 0...99.	The incoming events from input devices are recorded also into this 100-records array, with cyclic overwrite.
reserved	800 MSB			
event record 0, byte 0	801 LSB	R, RAM	First record in the event log. Contains the first captured byte.	The first byte should contain the input device address. The value must be shifted to left by 1 bit.
event record 0, byte 1	801 MSB	R, RAM	First record in the event log. Contains the second captured byte.	
event record 0, byte 2	802 LSB	R, RAM	First record in the event log. Contains the third captured byte.	

event record 0, length	802 MSB	R, RAM	First record in the event log. Contains the captured packet length.	
event record 1, byte 0	803 LSB	R, RAM	Second record in the event log. Contains the first captured byte.	
event record 1, byte 1	803 MSB	R, RAM	Second record in the event log. Contains the second captured byte.	
event record 1, byte 2	804 LSB	R, RAM	Second record in the event log. Contains the third captured byte.	
event record 1, length	804 MSB	R, RAM	Second record in the event log. Contains the captured packet length.	
...	...	...	...	
event record 99, byte 0	999 LSB	R, RAM	100th record in the event log. Contains the first captured byte.	
event record 99, byte 1	999 MSB	R, RAM	100th record in the event log. Contains the second captured byte.	
event record 99, byte 2	1000 LSB	R, RAM	100th record in the event log. Contains the third captured byte.	
event record 99, length	1000 MSB	R, RAM	100th record in the event log. Contains the captured packet length.	
reserved	1001 - 1002			
serial number low	1003	R	Serial number of the device, LSW	
serial number high	1004	R	Serial number of the device, MSW	

LSB – lower byte
MSB – higher byte
LSW – lower word
MSW – higher word

Notes ***Note 1***: Possible reasons of error messages:
0x00 No reply: Bad R091 hardware, wiring problems, ...
0x02 Bus error: Appears when the bus is short-circuited. If there is no traffic on the bus, the microcontroller checks the bus every second. If a short-circuit is detected, the red LED goes on and the bus power is switched off. After 1 s the bus power is switched on again and the check is repeated. If there is no short-circuit, the converter goes to normal. If the problem persists, the check is performed again after 1 s. The problem may also be in the damaged analogue output circuits of the converter, which the processor can not distinguish from a real bus short-circuit.
0x03 Unrecognized reply: It may happen that at installations where the bus contains 50 – 60 ballasts more ballasts reply at the same time, or if there is a signal interference.

Note 2: For all analogue values, the maximum settable value depends on the particular ballast type. Some ballasts allow e.g. to set the analogue value in range of 80 to 250 only. If the current value is 80 and a command to set to 254 is sent, the new value of 254 is displayed in the Modbus map and sent to the ballast, but the ballast does not accept this value and keeps its previous setting of 80.

Note 3: Even if a single bit is changed only, the register to control all 16 groups is always set as a whole (Modbus function F16). If, for example, a ballast is assigned to groups 14 and 15, both groups are off, and there is a command issued to set group 14 to on and group 15 to off, the ballast just blinks shortly and goes to off.

Tab. 2: Standard DALI commands

Nr.	DALI command (bin)	DALI address	D0	D1	D2	Function
Commands 0 – 31: Indirect arc power control commands						
0	0000 0000	YAAA AAA 1	0	0	0	OFF - Extinguish the lamp immediately without fading.
1	0000 0001	YAAA AAA 1	0	0	0	UP – Dim up for 200 ms (execution time) using the selected 'FADE RATE', if this command is received again while it is been executed, the execution time shall be retriggered. This command shall only affect ballasts with burning lamps. No lamp shall be ignited (from 0) with this command. No change if the arc power output is already at the "MAX LEVEL".
2	0000 0010	YAAA AAA 1	0	0	0	DOWN – Dim down for 200 ms (execution time) using the selected 'FADE RATE', if this command is received again while it is been executed, the execution time shall be retriggered Lamp shall not be switched off via this command. No change if the arc power output is already at the "MIN LEVEL".
3	0000 0011	YAAA AAA 1	0	0	0	STEP UP - Set the actual arc power level one step higher immediately without fading. This command shall only affect ballasts with burning lamps. No lamp shall be ignited with this command. No change if the arc power output is already at the "MAX LEVEL".
4	0000 0100	YAAA AAA 1	0	0	0	STEP DOWN - Set the actual arc power level one step lower immediately without fading. Lamp shall not be switched off via this command. No change if the arc power output is already at the "MIN LEVEL".
5	0000 0101	YAAA AAA 1	0	0	0	RECALL MAX LEVEL - Set the actual arc power level to the "MAX LEVEL" without fading. If the lamp is off it shall be ignited with this command.
6	0000 0110	YAAA AAA 1	0	0	0	RECALL MIN LEVEL - Set the actual arc power level to the "MIN LEVEL" without fading. If the lamp is off it shall be ignited with this command.
7	0000 0111	YAAA AAA 1	0	0	0	STEP DOWN AND OFF - Set the actual arc power level one step lower immediately without fading.

						If the actual arc power level is already at the "MIN LEVEL", the lamp shall be switched off by this command.
8	0000 1000	YAAA AAA 1	0	0	0	ON AND STEP UP - Set the actual arc power level one step higher immediately without fading. If the lamp is switched off, the lamp shall be ignited with this command and shall be set to the "MIN LEVEL".
9-15	0000 1XXX					reserved
16-31	0001 XXXX	YAAA AAA 1	0	0	0	GO TO SCENE - the actual arc power level to the value stored for scene XXXX using the actual fade time. If the lamp is off, it shall be ignited with this command. If the value stored for scene XXXX is zero and the lamp is lit then the lamp shall be switched off by this command after the fade time.
Commands 32 – 128: Configuration commands These commands are automatically sent twice within 100 ms.						
32	0010 0000	YAAA AAA 1	0	0	0	RESET – After the second reception of the command, the variables in the persistent memory shall be changed to their reset values. It is not guaranteed that any commands are received properly within the next 300 ms by a ballast acting on this command.
33	0010 0001	YAAA AAA 1	0	0	0	STORE ACTUAL LEVEL IN THE DTR Store actual arc power level in the DTR without changing the current light intensity. If the ballast is in the process of fading it is the instantaneous level, not the target level that is stored.
34-41	0010 XXXX					reserved
42	0010 1010	YAAA AAA 1	0	0	0	STORE THE DTR AS MAX LEVEL - Save the value in "Data Transfer Register" as new "MAX LEVEL".
43	0010 1011	YAAA AAA 1	0	0	0	STORE THE DTR AS MIN LEVEL - Save the value in "Data Transfer Register" as new "MIN LEVEL". If this value is lower as the "PHYSICAL MIN. LEVEL" of the ballast, then store the "PHYSICAL MIN. LEVEL" as new "MIN LEVEL".
44	0010 1100	YAAA AAA 1	0	0	0	STORE THE DTR AS A SYSTEM FAILURE LEVEL - Save the value in "Data Transfer Register" as new "SYSTEM FAILURE LEVEL".
45	0010 1101	YAAA AAA 1	0	0	0	STORE THE DTR AS POWER ON LEVEL - Save the value in "Data Transfer Register" as new "POWER ON LEVEL".

46	0010 1110	YAAA AAA 1	0	0	0	STORE THE DTR AS FADE TIME - Save the value in "Data Transfer Register" as new "FADE TIME". FADE TME range is 0-15, where 0 means no FADE. The fade time specifies the time for changing the arc power level from the actual level to the requested level. In case of lamp off, the preheat and ignition time is not included in the fade time.
47	0010 1111	YAAA AAA 1	0	0	0	STORE DTR AS FADE RATE - Save the value in "Data Transfer Register" as new "FADE RATE". FADE RATE range is 1-15, where 1 means fastest dimming and 15 slowest dimming.
48-63	0011 XXXX					reserved
64-79	0100 XXXX	YAAA AAA 1	0	0	0	STORE DTR AS SCENE - Save the value in Data Transfer Register as a new level of the scene 0-15 (XXXX).
80-95	0101 XXXX	YAAA AAA 1	0	0	0	REMOVE FROM SCENE - Remove the ballast from scene 0-15 - XXXX. Removing the ballast from scene XXXX means storing 0xFF in scene register XXXX.
96-111	0110 XXXX	YAAA AAA 1	0	0	0	ADD TO GROUP - Add the ballast to group 0-15 - XXXX.
112-127	0111 XXXX	YAAA AAA 1	0	0	0	REMOVE FROM GROUP - Remove the ballast from group 0-15 - XXXX. Removing the ballast from group XXXX means storing "0" in the group register.
128	1000 0000	YAAA AAA 1	0	0	0	STORE DTR AS SHORT ADDRESS - Save the value in the DTR as new short address. The structure of the DTR shall be: XXXX XXXX = 0AAA AAA1 or 1111 1111 shall remove the short address.
129-143	1000 XXXX					reserved
Commands 144 – 155: Query commands						
144	1001 0000	YAAA AAA 1	0	0	0	QUERY STATUS – Answer is the following „STATUS INFORMATION“ byte: **bit 0** - Status of ballast; 0 = OK **bit 1** – Lamp failure; 0 = OK **bit 2** – Lamp arc power on; 0 = OK **bit 3** - Query: Limit Error; "0" = Last requested arc power level is between MIN..MAX LEVEL or OFF **bit 4** – Fade ready; "0" = fade is ready; "1" = fade is running **bit 5** – Query: "RESET STATE"? "0" = "No" **bit 6 –** Query: Missing short address? "0" = "No" **bit 7** – Query: "POWER FAILURE"? "0" = "No";

						"RESET" or an arc power control command has been received after last power-on.
145	1001 0001	YAAA AAA 1	0	0	0	QUERY BALLAST - Ask if there is a ballast with the given address that is able to communicate. Answer shall be "Yes" or "No".
146	1001 0010	YAAA AAA 1	0	0	0	QUERY LAMP FAILURE - Ask if there is a lamp problem at the given address. Answer shall be "Yes" or "No".
147	1001 0011	YAAA AAA 1	0	0	0	QUERY LAMP POWER ON - Ask if there is a lamp operating at the given address. Answer shall be "Yes" or "No".
148	1001 0100	YAAA AAA 1	0	0	0	QUERY LIMIT ERROR - Ask if the last requested arc power level at the given address could not be met, because it is above the MAX LEVEL or below the MIN LEVEL. Answer shall be "Yes" or "No".
149	1001 0101	YAAA AAA 1	0	0	0	QUERY RESET STATE - Ask if the ballast is in "RESET STATE". Answer shall be "Yes" or "No".
150	1001 0110	YAAA AAA 1	0	0	0	QUERY MISSING SHORT ADDRESS - Ask if the ballast has no short address. Answer shall be "Yes" or "No". The answer shall be "Yes" if the ballast has no short address.
151	1001 0111	YAAA AAA 1	0	0	0	QUERY VERSION NUMBER - Ask for the version number of the IEC standard document met by the software and the hardware of the present ballast. The "VERSION NUMBER" shall be stored in a ROM. Answer shall be the 'VERSION NUMBER' as an 8 bit number 'XXXX 0000'. The first 4 bits (XXXX) represent the version number of this standard.
152	1001 1000	YAAA AAA 1	0	0	0	QUERY CONTENT DTR Answer shall be the content of the DTR as an 8 bit number.
153	1001 1001	YAAA AAA 1	0	0	0	QUERY DEVICE TYPE - Answer shall be an 8 bit number (x = 0 to 255). The standard device type shall return the answer 0 (this device type shall not react on the application extended commands 224 to 255). For the list of device types see command 272.
154	1001 1010	YAAA AAA 1	0	0	0	QUERY PHYSICAL MINIMUM LEVEL - Answer shall be the "PHYSICAL MIN. LEVEL" as an 8 bit number. The "PHYSICAL MIN. LEVEL" shall be stored in a ROM.
155	1001 1011	YAAA AAA 1	0	0	0	QUERY POWER FAILURE - Answer shall be "YES" if the ballast has not received a "RESET" or one of the following arc power control commands since the last power-on: "DIRECT ARC POWER CONTROL", "OFF", "RECALL MAX LEVEL", "RECALL MIN LEVEL", "STEP DOWN AND OFF", "ON AND STEP UP", "GO TO SCENE"

156-159	1001 11XX					reserved
Commands 160 – 165: Queries related to arc power parameter settings						
160	1010 0000	YAAA AAA 1	0	0	0	QUERY ACTUAL LEVEL - Answer shall be this level as an 8 bit number. During preheating and if a lamp error occurs the answer shall be "MASK".
161	1010 0001	YAAA AAA 1	0	0	0	QUERY MAX LEVEL - Answer shall be this level as an 8 bit number.
162	1010 0010	YAAA AAA 1	0	0	0	QUERY MIN LEVEL - Answer shall be this level as an 8 bit number.
163	1010 0011	YAAA AAA 1	0	0	0	QUERY POWER ON LEVEL - Answer shall be this level as an 8 bit number.
164	1010 0100	YAAA AAA 1	0	0	0	QUERY SYSTEM FAILURE LEVEL - Answer shall be this level as an 8 bit number.
165	1010 0101	YAAA AAA 1	0	0	0	QUERY FADE TIME / FADE RATE - Answer shall be XXXX YYYY where XXXX corresponds with the number of command 46 and where YYYY corresponds with the number of command 47.
166-175	1010 XXXX					reserved
Commands 176– 196: Queries related to system parameter settings						
176-191	1011 XXXX	YAAA AAA 1	0	0	0	QUERY SCENE LEVEL Answer shall be the arc power level of scene 0-15 - XXXX as an 8 bit number.
192	1100 0000	YAAA AAA 1	0	0	0	QUERY GROUPS 0-7 - One bit for each group in back channel data byte. Bit 0 = group 0. Bit1 = group 1 … "0" = not belonging to group. "1" = belonging to group.
193	1100 0001	YAAA AAA 1	0	0	0	QUERY GROUPS 8-15 - One bit for each group in back channel data byte. Bit 0 = group 8. Bit 1 = group 9 … "0" = not belonging to group. "1" = belonging to group.
194	1100 0010	YAAA AAA 1	0	0	0	QUERY RANDOM ADDRESS (H) - The 8 high bits of the random address
195	1100 0011	YAAA AAA 1	0	0	0	QUERY RANDOM ADDRESS (M) - The 8 mid bits of the random address.
196	1100 0100	YAAA AAA 1	0	0	0	QUERY RANDOM ADDRESS (L) - The 8 low bits of the random address.
197-223	110X XXXX					reserved
224-255	11XX XXXX	YAAA AAA 1	0	0	0	QUERY APPLICATION EXTENDED COMMANDS

256	0000 0000	1010 0001	0	0	0	TERMINATE – All special mode processes shall be terminated.
257	XXXX XXXX	1010 0011	0	0	0	DATA TRANSFER REGISTER (DTR) - Store 8 bit value XXXX XXXX to DTR.

Tab. 3: Address types

Short address	0-63	0AAAAAA1
Group address	0-15	100AAAA1
Broadcast		11111111
Direct control	0-63	0AAAAAA0
Direct control of one ballast	0-63	1AAAAAA0
Direct control of group	0-15	100AAAA0

Tab. 4: Advanced DALI commands

Nr.	DALI command	DALI address	D0	D1	D2	Function
258	1010 0101	XXXX XXXX				INITIALISE – This command shall be received a second time in the next 100 ms. No other commands addressing the same ballast shall be received between these two commands, otherwise these commands and command 258 shall be ignored. The command shall start or re-trigger a timer for 15 minutes; the addressing commands 259 to 270 shall only be processed within this period. All other commands shall still be processed during this period. This time period shall be aborted with the "TERMINATE" command. This command is sent automatically twice within 100ms. 0000 0000 – All ballasts shall react 0AAA AAA1 – Ballasts with the address AAA AAA shall react 1111 1111 – Ballasts without short address shall react
259	1010 01111	0000 0000				RANDOMISE – This command shall be received a second time in the next 100 ms. No other commands addressing the same ballast shall be received between these two commands, otherwise these commands and command 259 shall be ignored. The ballast shall generate a new random address on the request of this command. The new random address shall be available within a time period of 100 ms. This command is sent automatically twice within 100 ms.
260	1010 1001	0000 0000				COMPARE – The ballast shall compare its random address with the combined search address stored in SEARCHADDRH, SEARCHADDRM and SEARCHADDRL. If its random address is smaller or equal to the combined search address stored in SEARCHADDRH, SEARCHADDRM and SEARCHADDRL and the ballast is not withdrawn then the ballast shall generate a query ”YES”.
261	1010 1011	0000 0000				WITHDRAW – The ballast with its random address that is equal to the combined search address stored in SEARCHADDRH, SEARCHADDRM and SEARCHADDRL shall no longer respond to the COMPARE command. This ballast shall not be excluded from the

						initialization process.
262	1010 1101	0000 0000				reserved
263	1010 1111	0000 0000				reserved
264	1011 0001	HHHH HHHH				SEARCHADDRH
265	1011 0011	MMMM MMMM				SEARCHADDRM
266	1011 0101	LLLL LLLL				SEARCHADDRL
The final address is formatted as HHHHHHHHMMMMMMMMLLLLLLLL						
267	1011 0111	0AAA AAA1				PROGRAM SHORT ADDRESS – The ballast shall store the received 6 bit address as its short address if it is selected.
268	1011 1001	0AAA AAA1				VERIFY SHORT ADDRESS - The ballast shall give an answer ”YES” if the received short address is equal to its own short address.
269	1011 1011	0000 0000				QUERY SHORT ADDRESS – The ballast shall send the short address if the random address is the same as the search address or the ballast is physically selected. The structure of the answer shall have the format 0AAA AAA1. If no short address is stored the answer shall be "MASK".
270	1011 1101	0000 0000				PHYSICAL SELECTION – If a ballast receives this command, it shall cancel its physical selection and shall set the ballast to "Physical Selection Mode". In this mode the comparison of SEARCH and RANDOM ADDRESS shall be disabled.
271	1011 1111					reserved
272	1100 0001	XXXX XXXX				ENABLE DEVICE TYPE X - X = 0 to 255. This command shall be sent before an application extended command (224 – 255). This command can be processed without the use of the INITIALISE command. This command shall not be used for device type 0, because the application extended commands 224-255 are not used for this device type. X=0 – device for fluorescent lamps X=1 – device for emergency lighting X=2 – device for HID lamps X=3 - device for low voltage halogen lamps X=4 – device for dimming of incandescent lamps X=5 – device for conversion of digital signals according to E.4 into d.c. signals according to E.2 X=6 – LED X=7 – 255 - reserved
273	1100 0011					reserved
274	1100 0101					reserved
275	1100 0111					reserved
276	1100 1001				0x13	reserved
277	1101 0001				0x14	reserved

278	1101 0101				0x15	reserved
279	1101 0111				0x16	reserved

Tab. 5: Responses to advanced DALI commands

Nr.	DALI command	DALI address	D0	D1	D2	Function
260	1010 1001	0000 0000	Response	-	-	COMPARE – The ballast shall compare its random address with the combined search address stored in SEARCHADDRH, SEARCHADDRM and SEARCHADDRL. If its random address is smaller or equal to the combined search address stored in SEARCHADDRH, SEARCHADDRM and SEARCHADDRL and the ballast is not withdrawn then the ballast shall generate a query ”YES”.
268	1011 1001	0AAA AAA1	Response			VERIFY SHORT ADDRESS - The ballast shall give an answer ”YES” if the received short address is equal to its own short address.
269	1011 1011	0000 0000	Response			QUERY SHORT ADDRESS – The ballast shall send the short address if the random address is the same as the search address or the ballast is physically selected. The structure of the answer shall have the format 0AAA AAA1. If no short address is stored the answer shall be "MASK".

R091 converter special functions

The R091 converter contains 22 preprogrammed functions that are not part of the DALI standard. These functions trigger sequences of other commands.

Functions are controlled by registers that are normally used to control DALI functions.

The following table shows example of using these functions by block 0 (reg. 6, 7, 8).

However, any block can be used to activate them, so e.g. for block 1 registers 9, 10, 11 will be used, for block 2 registers 12, 13, 14, etc.

Triggering of these functions is done by setting the corresponding bits in register 5 – command mask.

Tab. 6: R091 converter special functions

No.	6 LSB	6 MSB	7 LSB	7 MSB	8 LSB	Function
1	-	YAAA AAA1	0100 XXXX	Value [0-254]	0000 0001	Store value as new parameter of scene XXXX (group address can be used as well)
2	-	0AAA AAA1	0110 XXXX 0111 XXXX	-	0000 0010	0110 XXXX = Add ballast to group XXXX 0111 XXXX = Remove ballast from group XXXX
3	-	YAAA AAA1	-	Value [0-15]	0000 0011	Store value as „FADE TIME“ (group address can be used as well)
4	-	YAAA AAA1	-	Value [1-15]	0000 0100	Store value as „FADE RATE“ (group address can be used as well)
5	-	YAAA AAA1	-	Value [0-254]	0000 0101	Store value as „MAX LEVEL“ (group address can be used as well)
6	-	YAAA AAA1	-	Value [0-254]	0000 0110	Store value as „MIN LEVEL“ (group address can be used as well)
7	-	YAAA AAA1	-	Value [0-255]	0000 0111	Store value as „SYSTEM FAILURE LEVEL“ (group address can be used as well)
8	-	YAAA AAA1	-	Value [0-254]	0000 1000	Store value as „POWER ON LEVEL“ (group address can be used as well)
9	-				0000 1001	Completely new addressing
10	-		Address to start with	0AAA AAA1	0000 1010	New addressing of all ballasts with given address
11	-		Address to start with		0000 1011	New addressing of ballasts without short address
12	-	0AAA AAA1	-	-	0000 1100	Deletes given short address of ballast
13	-	0AAA AAA1 (current address)	-	0AAA AAA1 (new address)	0000 1101	Changes current address to new address

14	-	YAAA AAA1	number of winks [1-255]	wink time [1-255]	0000 1110	winks the addressed ballast; wink values must not be 0! (group address can be used as well)
15	-				0000 1111	Short addresses request [0-31]
16	-				0001 0000	Short addresses request [32-63]
17	-				0001 0001	Ballast status request [0-31]
18	-				0001 0010	Ballast status request [32-63]
19	-				0001 0011	“Lamp failure“ request [0-31]
20	-				0001 0100	“Lamp failure“ request [32-63]
21	-				0001 0101	“Lamp power on“ request [0-31]
22	-				0001 0110	“Lamp power on“ request [32-63]

Tab. 7: Answers for R091 converter special functions

No.	6 LSB	6 MSB	7 LSB	7 MSB	8 LSB	Function
1	-	-	-	-		
2	-	-	-	-	-	
3	-	-	-	-	-	
4	-	-	-	-	-	
5	-	-	-	-	-	
6	-	-	-	-	-	
7	-	-	-	-	-	
8	-	-	-	-	-	
9	-	-	Number of addressed ballasts [0-63]	-	-	Completely new addressing
10	-	-	Number of addressed ballasts [0-63]	-	-	New addressing of all ballasts with given address
11	-	-	Number of addressed ballasts [0-63]	-	-	New addressing of all ballasts without short address
12	-	-	-	-	-	
13	-	-	-	-	-	
14	-	-	-	-	-	
15	Addresses 0-7	Addresses 8-15	Addresses 16-22	Addresses 23-31	-	1 – Yes 0 - No
16	Addresses 32-39	Addresses 40-47	Addresses 48-55	Addresses 56-63	-	1 – Yes 0 - No
17	Addresses 0-7	Addresses 8-15	Addresses 16-22	Addresses 23-31	-	1 - Error 0 - OK

18	Addresses 32-39	Addresses 40-47	Addresses 48-55	Addresses 56-63	-	1 - Error 0 - OK
19	Addresses 0-7	Addresses 8-15	Addresses 16-22	Addresses 23-31	-	1 - Error 0 - OK
20	Addresses 32-39	Addresses 40-47	Addresses 48-55	Addresses 56-63	-	1 - Error 0 - OK
21	Addresses 0-7	Addresses 8-15	Addresses 16-22	Addresses 23-31	-	1 - On 0 - Off
22	Addresses 32-39	Addresses 40-47	Addresses 48-55	Addresses 56-63	-	1 - On 0 - Off

Light level control

It is possible to control the light level using two different ways. DALI recognizes direct and indirect light level control.

Indirect light level control
The command consists of 2 bytes.

Byte 1 DALI address (short address / group address / broadcast)
Byte 2 Standard or Extended DALI command – see tables above.

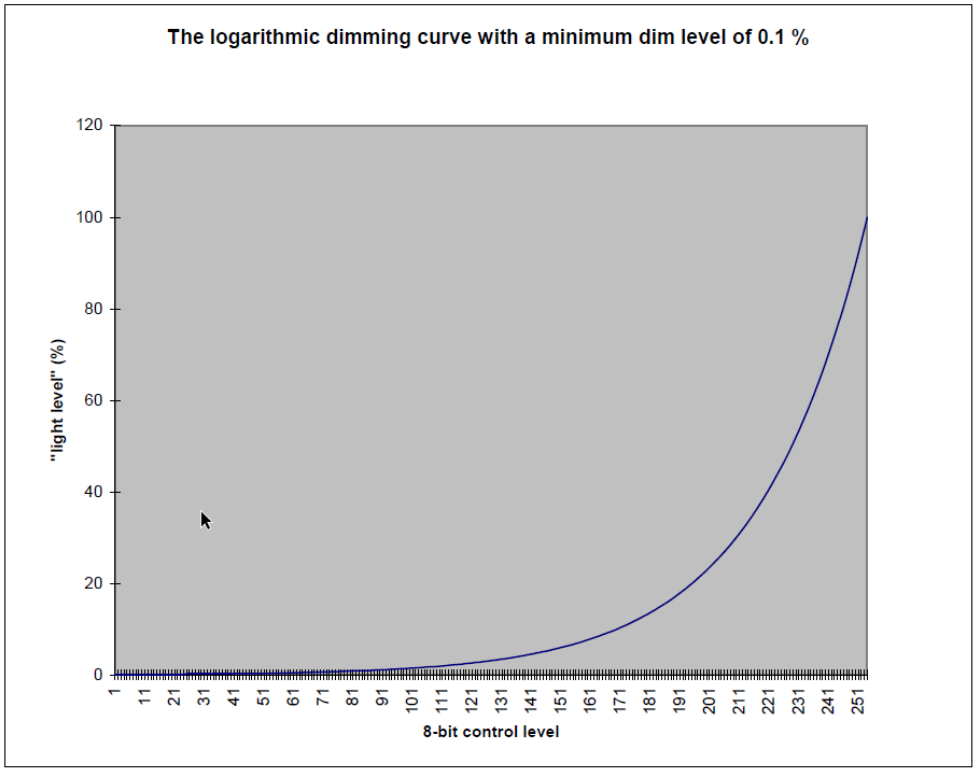
Direct light level control
The command consists of 2 bytes.

Byte 1 DALI address; this is the DALI ballast short address, with the first bit of 0.
Byte 2 Light level: a number in the range of 0...254 (note that the maximum range depends on the manufacturer setting).

This is a way how to control the light level directly without using of groups, scenes etc.

Scene control can use also percentage values. It is calculated using the following formula:

$$X(n) = 10^{\frac{n-1}{253/3}-1}, \left|\frac{X(n)-X(n+1)}{X(n)}\right| = const. = 2.8\%$$



Source: EN 60929

Examples of Modbus TCP commands

There are 8 "blocks" - (0 to 7) – which represent positions for the DALI commands. To execute a command:

- the block(s) must be filled with the data representing the command(s)
- bit(s) in Register 5 LSB which corresponds to the block to be executed must be set.

After executing of the command, the info bit in register 5 MSB is set so that the Modbus master can read that the execution was OK.

If a command is generating a response, the response is stored in the D0..D2 registers of the particular block.

More blocks may be filled at the same time and executed all together by writing a corresponding bit pattern into register 5 LSB.

Example 1:
Tx: 00 07 00 00 00 0D 01 10 00 05 00 03 06 0B 05 00 00 00 00

A Modbus TCP example telegram for **ballast Adr 6: set to max (DALI function 5)**. It is written in command block 0 (Modbus register 6, which is Modbus address 5).

00 07 00 00 00 0D 01	Details see Modbus TCP frame structure
10	Modbus F16, write multiple registers
00 05	Modbus address to be written to, address 5 = register 6
00 03	Number of 16bit registers to be written
06	Number of bytes to follow
0B 05 00 00 00 00	The data is 0B 05 for command block 0, other command blocks 1 and 2 are empty (00 00 00 00) - (this is how the client software, with which the example was done, is communicating: it writes three blocks at a time; other clients may only send the first couple of bytes).

The most important data is **0B 05**.

05: LSB = DALI command, see Tab 2 No.5
0B: MSB = 0000 1011 - the structure of a standard DALI command - see Tab 2:

Y AAA AAA 1, where
Y = 0 for Short address (see Tab 3), and
AAA AAA = 000 101 = 5 = DALI address of the ballast starting at 0, i.e. the 6th ballast.

In a similar way more command blocks may be filled, and then activated at the same time.

Another Modbus TCP telegram has to activate (execute) this command:

Address 6: execute block 0 (write 1 into Modbus register 5, or Modbus address 4)

00 08 00 00 00 09 01	Details see Modbus TCP frame structure
10	Modbus F16, write multiple registers
00 04	Modbus address to be written to, address 4 = register 5
00 01	Number of 16bit registers to be written

02	Number of bytes to follow
00 01	1 at Bit 0 means **Execute command block 0**. At this time the commands from Block D0 are sent to the DALI bus.

Example 2:

Sending the *Central On* command using data block 0

- Write 0xFF05 to register 6 (0xFF is broadcast address, 0x05 the *Recall Max Level* command)
- Write 0x01 to register 5. This starts executing the command. The command is in the queue or being executed until the bit 0x01 in register 5 is reset
- In case of error, the error code of 0x02 or 0x03 is readable in register 7 MSB. If the ballast responded correctly, the reply will be in register 7 LSB, and register 7 MSB will have value of 0x55.

Sending the *Central Off* command using data block 1

- Write 0xFF00 to register 9 (0xFF is broadcast address, 0x00 the *Off* command)
- Write 0x02 to register 5. This starts executing the command. The command is in the queue or being executed until the bit 0x02 in register 5 is reset
- In case of error, the error code of 0x02 or 0x03 is readable in register 10 MSB. If the ballast responded correctly, the reply will be in register 10 LSB, and register 10 MSB will have value of 0x55.

Sending the *Central Off* command using data block 8

- Write 0x00FF to register 516 (0xFF is broadcast address, 0x00 the *Off* command)
- Write 0x0000 to register 518. This will set a 16 bit packet, Send only once, and lowest priority.
- Write 0x0001 to register 511. This starts executing the command. The command is in the queue or being executed until the bit 0x01 in register 512 is reset
- In case of error, the error code is readable in register 562 MSB. If the ballast responded correctly, the reply will be in register 561 LSB, and register 562 LSB will have value of 0x03.

Sending the *Central On* command using data block 9

- Write 0x05FF to register 520 (0xFF is broadcast address, 0x05 the *Recall Max Level* command)
- Write 0x0000 to register 522. This will set a 16 bit packet, Send only once, and lowest priority.
- Write 0x0002 to register 511. This starts executing the command. The command is in the queue or being executed until the bit 0x02 in register 512 is reset
- In case of error, the error code is readable in register 564 MSB. If the ballast responded correctly, the reply will be in register 563 LSB, and register 564 LSB will have value of 0x03.

Sending a 24bit packet containing data (example: 01, 02, 03) using data block 8

- Write 0x0201 to register 516 (first two data bytes)
- Write 0x0003 to register 517 (third data byte)

- Write 0x0100 to register 518. This will set a 24 bit packet, Send only once, and lowest priority.
- Write 0x0001 to register 511. This starts executing the command. The command is in the queue or being executed until the bit 0x01 in register 512 is reset. The data [01, 02, 03] are sent.
- In case of error, the error code is readable in register 562 MSB. If the ballast responded correctly, the reply will be in register 561 LSB, and register 562 LSB will have value of 0x03.

Registers for simple control and status monitoring

To make Modbus communication easier, it is possible to read out statuses and control the ballasts also **using a simple Modbus read / write commands to dedicated Modbus registers 30 to 182**. These commands are converted to DALI commands in the converter, and sent to the DALI bus (unlike the standard commands, where the Modbus client actually has to compile the DALI telegrams and send them over Modbus). The Modbus client then may assign a separate register or bit to each ballast which makes the Modbus client engineering easier.

It is necessary to enable the required functions in Register 30 (see table above) for two reasons:

- this communication may bring extra load to the DALI bus, it is advisable to set e.g. the status update interval to the longest acceptable time

- only enabled command types are sent to the DALI bus – for security reason.

If these functions are not used, they should be disabled in register 30 or over the web interface.

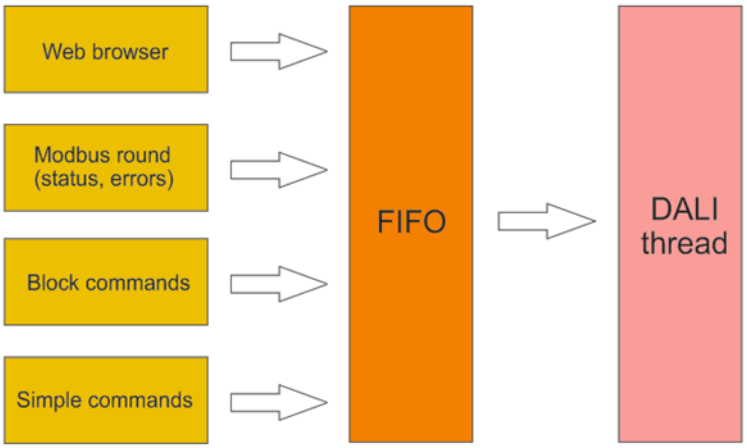
Note that if more commands to control a single DALI ballast are set over different registers, the last one is active. Make sure that the Modbus client does not send ambiguous commands over different registers which could spoil the intended DALI functionality.

To use the simple control commands in a proper way, it is necessary to understand the principle of command processing within the R091. There is an internal FIFO queue of 96 commands. The commands are read over the Modbus TCP interface or web pages, and put into the queue. On the queue output, the commands are translated into DALI telegrams, and sent to the DALI interface. **There is no feedback between the DALI command execution and the respective Simple command.** The Modbus server response to confirm a Simple command receipt only means that the command has been received by the R091, not that the command has been queued or executed at the DALI bus.

There are no exceptions, priorities, nor any other internal rules in the queue. As the DALI bus communication speed is 1200 bps while the Modbus TCP commands travel at the speed of Ethernet, it may happen that the queue gets full in case that the Simple commands are sent in a fast sequence.

If the queue is full, all incoming Modbus Simple commands are discarded. At the *Statistics* web page, there is a **Dali failure counter** item which counts the discarded commands. If this value is increasing steadily, it means that the Simple commands queue is permanently full and the Modbus communication should be less frequent.

In register 30 or over web can be set that when a command is discarded, the Modbus server returns a Modbus error 06 (Slave device busy).



Always select only the relevant simple command types at the *Conf* page. It is advised to disable the types of commands which are not used.

CGI commands

A CGI interface is available at the web server. Using CGI commands, all functions which are executable over the web pages can also be activated over CGI commands. The *ctrl.cgi* script must be called using a GET request with parameters which specify the operation type as well as arguments.

Example of a CGI request: ***ctrl.cgi?SrchShortAddr=0***
where ***ctrl.cgi*** is the CGI script
SrchShortAddr is the CGI command
=0 is the command argument.

Most of the commands are executed in several steps:

1. processing of input arguments and sending of the command to the DALI bus
2. periodical requesting if the command has been completed
3. after completion, the response with data is being assembled.

To simplify the implementation, the steps above are entered as command parameters (with the only exception, which is the *CentralOn* command).

Status	Code example	Response	Description
Trigger execution	*ctrl.cgi?SrchShortAddr=**0***	*{Status“:{"Code":01}}*	Starts executing a command.
Check end	*ctrl.cgi?SrchShortAddr=**1***	*{Status“:{"Code":01}}*	The command is being executed.
Check end	*ctrl.cgi?SrchShortAddr=**1***	*{Status“:{"Code":00}, data:{"d0":00, "d1":00}}*	Command executed successfully, the response with data is assembled if the command returns data.

GetLastData	*ctrl.cgi?SrchShortAddr=**2***	*{Status":{"Code":00}, data:{"d0":00, "d1":00}}*	Returns the last data immediately, no command is executed.

Not all commands do send or return data; see the command table below.

Every CGI command returns the operation status and also may return data. The reply is always formated as JSON and always contains the *Status* node (e.g. *{"Status":{"Code":00}}*). The Status node informs if the GET request was accepted and how the command, launched by the request, is being executed.

Status code table:

Description	**Code**
WEB_OK	0x00
WEB_OK_CMD_NOT_FINISHED	0x01
WEB_ERR_CMD_FIFO_FULL	0x81
WEB_ERR_CMD_IN_PROGRESS	0x82
WEB_ERR_UNKNOWN	0xFE
WEB_ERR_INVALID_GET_REQUEST	0xFF

The following table contains the list of all available CGI commands.
The values (...***=00000001*** etc.) are for illustration only, and must be replaced by the intended parameters and addresses.

Function	***Code***	**Response**	**Description**
Check if command was completed, used at Central On/Off	*ctrl.cgi?CheckFinished=1*	Status	To check repeatedly if the previous command has been completed
Central On/Off	*ctrl.cgi?CentralOn=1* *ctrl.cgi?CentralOn=0*	Status	Sends a central On or Off command to the bus
Search of short address	*ctrl.cgi?SrchShortAddr=x*	Status,SrchShortAddr: {d0:00,d1:00}	Launches searching of a ballast, the found addresses are returned as 64 bits in d0 and d1 which are 32bits variables as hex
Readdressing of all ballasts	*ctrl.cgi?TotReAddress=x*	Status	Returns status only. Use the *SrchShortAddr* command to get the new data
Addressing without short address	*ctrl.cgi?d0=000000001 &BalAddrWithoutShortAddr=x*	Status	d0 is the starting address. Use the *SrchShortAddr* command to get the new data
Addressing with short address	*ctrl.cgi?d0=000000001 &d1=000000002 &BalAddrWithShortAddr=x*	Status	d0 is the starting address, d1 is the starting address of the new range. Example: Old addressing 1...10, d0=3, d1=12, the result is 1, 2, 12, 13, 14, 15, 16, 17, 18, 19.
Change address	*ctrl.cgi?d0=000000001 &d1=000000002 &ChangeShortAddr=x*	Status	Change short address d0 to new address d1.
Delete address	*ctrl.cgi?d0=000000001 &DeleteShortAddr=x*		Delete the short address d0.

Read ballast status	*ctrl.cgi?GetLampStat=x*	Status,GetLampStatOn:{d0:00,d1:00},GetLampStatErr:{d0:00,d1:00}	Returns statuses of all ballasts in 2 nodes (On and Error) as 64 bits in d0 and d1 which are 32bits variables as hex
Read scenes	*ctrl.cgi?d0=00000001 &ReadScenes=x*	Status,GetScenes:{d0:00,d1:00,d2:00,d3:00}	d0 in the request is the address to read from, the response is in d0 to d3 as 16 scenes, 1 byte (0...255) each
Write scenes	*ctrl.cgi?d0=00000001 &d1=00000001 &d2=00000001 &d3=00000001 &d4=00000001 &WriteScenes=x*	Status	Writes to address in d0 the scenes pattern given in d1 to d4 as 16 scenes, 1 byte (0...255) each
Read groups	*ctrl.cgi?d0=00000001 &ReadGroups=x*	Status,GetGroups:{d0:00}	Reads configuration of groups for a ballast with short address set in d0; the response is returned in d0 as 32bit variable in hex
Write groups	*ctrl.cgi?d0=00000001 &d1=00000001 &WriteGroups=x*	Status	Writes group configuration to a ballast with short address set in d0, group configuration is in d1, d0 and d1 are 32 bit variables in hex
Read values from the ECG page	*ctrl.cgi?d0=00000001 &ReadEcg=x*	Status,GetGroups:{d0:00, d1:00}	Reads ECG parameters from a ballast with short address in d0, the response is in d0 to d1 as follows: d0:"Min level", "Max level", "System failure level", "Power on level"; d1:"Fade time", "Fade rate"; d0 and d1 are 32 bit variables in hex
Write values to the ECG page	*ctrl.cgi?d0=00000001 &d1=00000001 &d2=00000001 &WriteEcg=x*	Status	Writes ECG parameters to a ballast with short address in d0, the parameters must be in d1 to d2 as follows: d1:"Min level", "Max level", "System failure level", "Power on level"; d2:"Fade time", "Fade rate"; d0 to d2 are 32 bit variables in hex
Set a light to On	*ctrl.cgi?d0=00000001 &AddrOn=x*	Status	Sets a ballast with short address of d0 to On
Set a light to Off	*ctrl.cgi?d0=00000001 &AddrOff=x*	Status	Sets a ballast with short address of d0 to Off
Set a light to Min level	*ctrl.cgi?d0=00000001 &MinLevelAddr=x*	Status	Sets a ballast with short address of d0 to Min Level
Set scene	*ctrl.cgi?d0=00000001 &d1=00000001 &SetScene=x*	Status	Sets a ballast with short address of d0 to scene d1
Set a light to light level	*ctrl.cgi?d0=00000001 &d1=00000001 &SetActLevel=x*	Status	Sets a ballast with short address of d0 to level d1
Set a group to On	*ctrl.cgi?d0=00000001 &GrpOn=x*	Status	Sets a group with address of d0 to On
Set a group to Off	*ctrl.cgi?d0=00000001 &GrpOff=x*	Status	Sets a group with address of d0 to Off
Set a group to Min level	*ctrl.cgi?d0=00000001 &MinLevelGrp=x*	Status	Sets a group with address of d0 to Min level

Set a group to level	*ctrl.cgi?d0=00000001 &d1=00000001 &SetGrpActLevel=x*	Status	Sets a group with address of d0 to level d1
Read R091 configuration	*ctrl.cgi?ReadConfig=x*	Status,GetConfig:{d0:00 , d1:00}	Returns ballast configuration. d0 – period of status readout in seconds d1 – configuration bits: bit 0 - "Status loop enable" bit 1 - "Analog driving ballast" bit 2 - "Analog driving group" bit 3 - "Analog driving broadcast" bit 4 - "Digital driving ballast" bit 5 - "Digital driving group" bit 6 - "Digital driving broadcast"
Write R091 configuration	*ctrl.cgi?d0=00000001 &d1=00000001 &WriteConfig=x*	Status	Writes ballast configuration. d0 – period of status readout in seconds d1 – configuration bits: bit 0 - "Status loop enable" bit 1 - "Analog driving ballast" bit 2 - "Analog driving group" bit 3 - "Analog driving broadcast" bit 4 - "Digital driving ballast" bit 5 - "Digital driving group" bit 6 - "Digital driving broadcast"

Note that the CGI access is not protected. To disable CGI access for security reasons, use the WEB switch (see above). It is recommended to use the web services in closed networks only.

Firmware update

In case that firmware cannot be updated properly by web interface, try this:

The last firmware version is available on web link:
http://domat-int.com/en/downloads/software Section Firmware for Domat devices.

- Open the web page of the R091, go to *Administration*, and upload the new firmware file (*R091_fw_x_x_x.bin*)
- power off / on the R091
- connect to the R091 over FTP (name / password: root / root99)
- delete all web pages which are in the R091
- copy the new web pages from your PC to the R091
- disconnect the FTP server
- power off / on the R091

Changes in versions

11/2016 – The first datasheet version (M090).
12/2016 – Updated information about DALI commands, Modbus table and firmware update information.
01/2017 – Design notes merged with Bus design notes and all moved above the Technical data table.
01/2017 – New table “R091 converter special functions” added, minor updates to DALI functions tables.
01/2017 – New information about commands that have to be sent twice within 100ms
01/2018 – R091 datasheet splitted from R090. Modbus table enhanced – reg. 516 and on, CGI control added,
08/2018 – Typos fixed, minor Modbus table corrections, Modbus table enhanced – reg. 793, 794 and on, new schema and DIP switch description.
09/2018 – Adding registers (Trigger) to Modbus table, new web screenshots and screenshot description.
09/2018 – Compatibility section added.
08/2019 – Minor description modification.
07/2020 – Fixed: Swapped values (address / value) in commands 256 and 257.
10/2020 – Fixed: Wrong switch number (1) in “Set the DIP switch 2 (INIT) to ON.”
03/2021 – Description for register 7 MSB enhanced.
06/2021 – Default value of reg. 159 corrected, logo changed.
07/2021 – *Settings*: added option to secure web interface wth username and password.
10/2021 – Addition of specification for PoE.
02/2022 – Description for register 7 MSB enhanced.