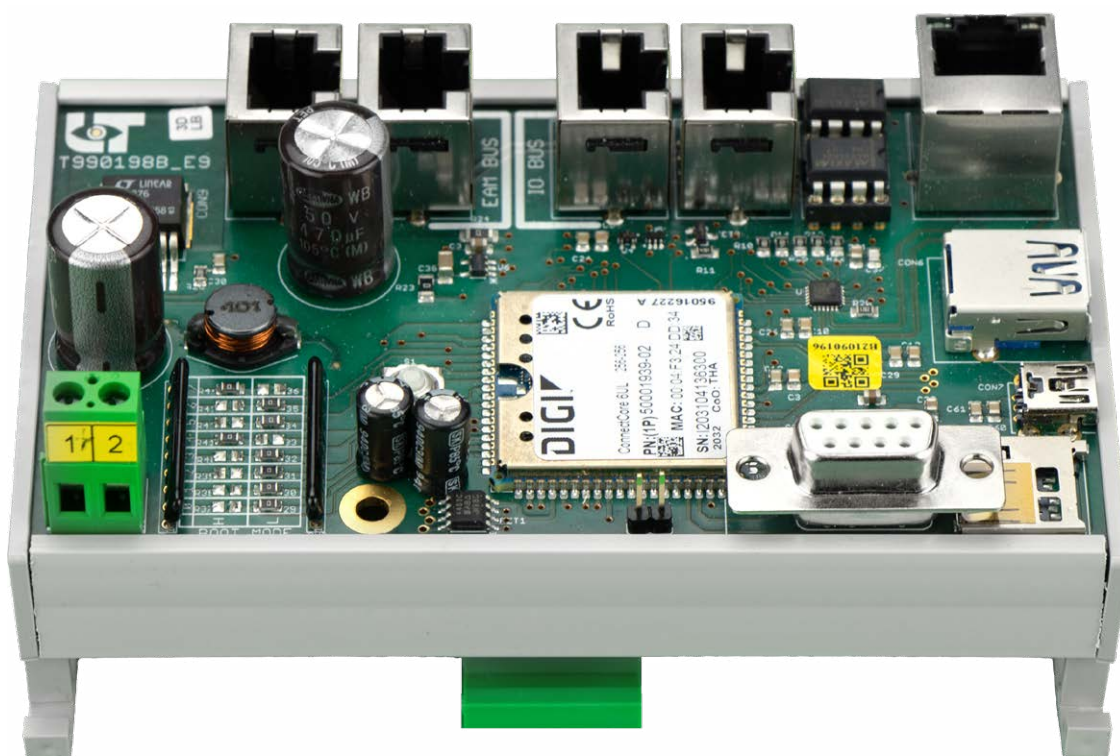




## MD2400 UNIVERSAL INTERFACE FOR CONNECTION TO A BUILDING MANAGEMENT SYSTEM - Art. 244391 -

### RELEASE NOTES



## 1. INHOUD

---

<b>1. INTRODUCTION</b>	<b>4</b>
1.1 ETHERNET FRAME	4
<b>2. UDP COMMUNICATION METHOD</b>	<b>5</b>
2.1 MD2400 FIRE DETECTION CONTROL PANEL TO BUILDING MANAGEMENT SYSTEM	5
2.2 BUILDING MANAGEMENT SYSTEM TO MD2400 FIRE DETECTION CONTROL PANEL	6
2.3 STRUCTURE UDP PACKET	7
2.3.1 LIMOTEC UDP HEADER	8
2.3.2 LIMOTEC UDP DATA	9
<b>3. UDP PROTOCOL</b>	<b>10</b>
3.1 OVERVIEW OF PACKETS	10
3.2 PACKET DETAIL	10
3.2.1 CODE_HEARTBEAT (0X00)	10
3.2.2 CODE_ACKNOWLEDGE_EXTERN (0X01)	11
3.2.3 CODE_ACKNOWLEDGE (0X02)	11
3.2.4 CODE_ABSENT (0X04)	11
3.2.5 CODE_DETECTOR_EVENT (0X10)	12
3.2.6 CODE_CENTRAL_FAULT_EVENT (0X12)	13
3.2.7 CODE_CENTRAL_EVENT (0X14)	14
3.2.8 CODE_CHANGE_TECHNICIAN (0X16)	14
3.2.9 CODE_SET_CHANGE_STATE (0X17)	15
3.2.10 CODE_CHANGE_STATE (0X18)	16
3.2.11 CODE_STARTUP_CENTRAL (0X1A)	17
3.2.12 CODE_STARTUP_EXTERN (0X1B)	17
3.2.13 CODE_REMOVE_EVENT (0X1C)	18
3.2.14 CODE_ASK_ZONE_STATUS (0X21)	18
3.2.15 CODE_ZONE_STATUS (0X22)	19
3.2.16 CODE_ASK_DETECTOR_VALUES (0X23)	19
3.2.17 CODE_DETECTOR_VALUES (0X24)	20
3.2.18 CODE_DO_CENTRAL_ACTION (0X25)	21
3.2.19 CODE_TEST_ALARM (0X27)	21

<b>4. MODBUS COMMUNICATION METHOD</b> .....	<b>22</b>
4.1 BMS REQUESTS DATA FROM THE MD2400 FIRE DETECTION CONTROL PANEL .....	22
4.2 BMS MODIFIES DATA ON THE MD2400 FIRE DETECTION CONTROL PANEL .....	22
4.3 STRUCTURE OF A MODBUS TCP PACKET .....	22
4.3.1 MODBUS APPLICATION PROTOCOL HEADER .....	23
4.3.2 PROTOCOL DATA UNIT .....	23
4.3.3 EXAMPLES .....	23
<b>5. MODBUS PROTOCOL</b> .....	<b>26</b>
5.1 COMPONENTS OF THE MD2400 FIRE DETECTION CONTROL PANEL (REGISTERS 1–2046) .....	26
5.1.1 MODBUS ADDRESSES .....	26
5.1.2 REGISTER DATA .....	26
5.2 GROUPS OF THE MD2400 FIRE DETECTION CONTROL PANEL (REGISTERS 4101–4164) .....	27
5.2.1 MODBUS ADDRESSES .....	27
5.2.2 REGISTER DATA .....	27
5.3 LOOPS OF THE MD2400 FIRE DETECTION CONTROL PANEL (REGISTERS 4301–4316) .....	27
5.3.1 MODBUS ADDRESSES .....	27
5.3.2 REGISTER DATA .....	27
5.4 MD2400 FIRE DETECTION CONTROL PANEL IN A TOKEN-RING (REGISTERS 4451–4472) .....	28
5.4.1 MODBUS ADDRESSES .....	28
5.4.2 REGISTER DATA .....	28

## 1. INTRODUCTION

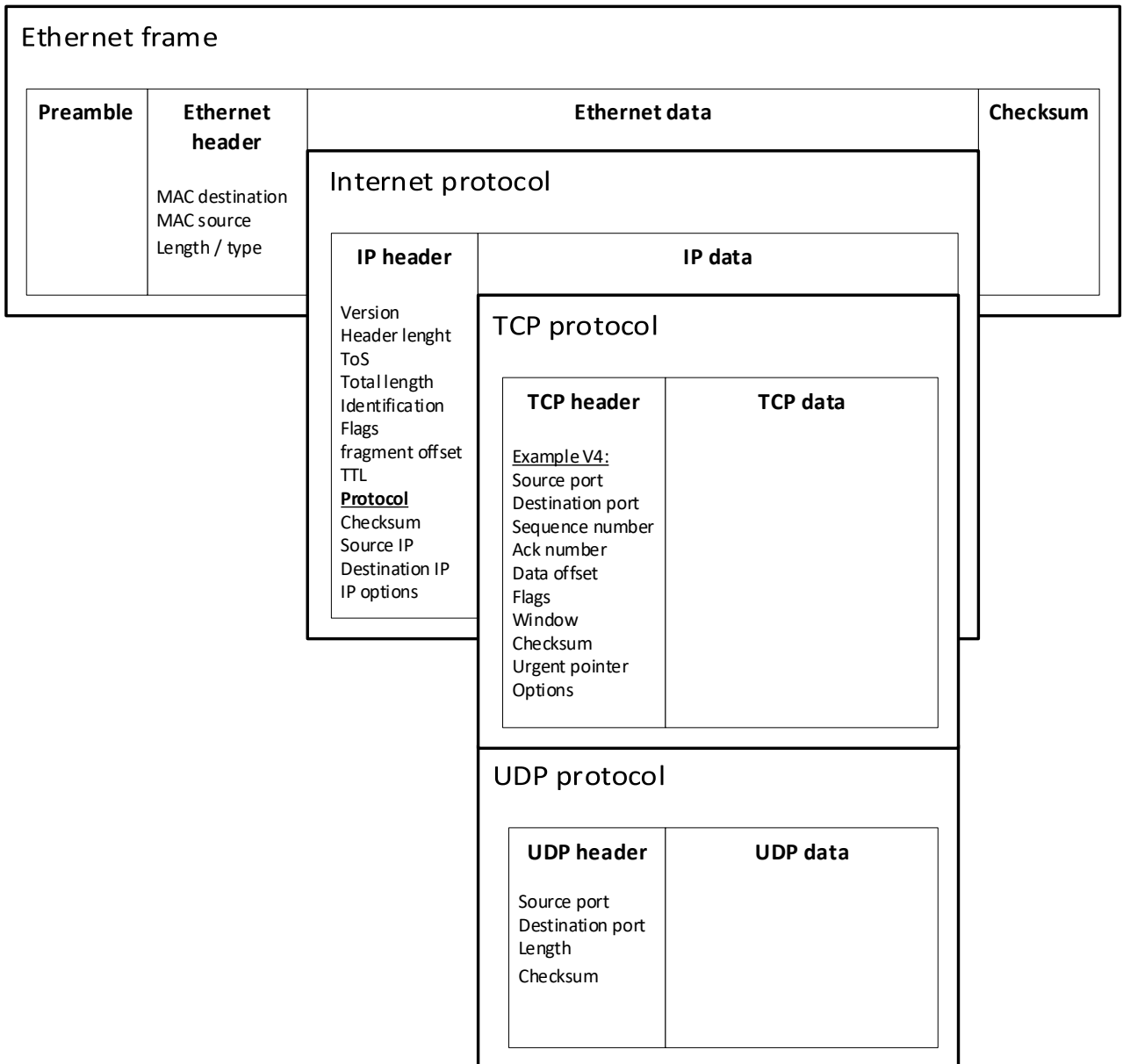
The document below is intended for the system integrator who wishes to implement the MD2400 UDP or MD2400 MODBUS TCP/IP protocol in their software. The external party that connects to the MD2400 fire detection control panel using the UDP or MODBUS protocol is hereafter referred to as the “BUILDING MANAGEMENT SYSTEM.”

It applies only to MD2400 fire detection control panels equipped with firmware version V2.3 or higher.

The selection between UDP or MODBUS protocol can only be performed by a Limotec installer.

The MD2400 fire detection control panel contains a maximum of 16 loops, each of which can be equipped with a maximum of 126 components. These components can in turn be placed into groups. The exact configuration of the fire detection control panel depends on the programming and must be obtained through the installer of the fire detection system.

### 1.1 ETHERNET FRAME



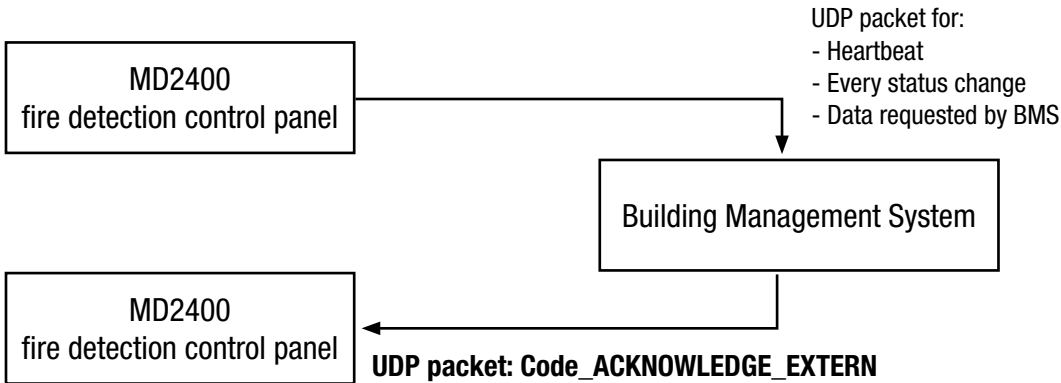
## 2. UDP COMMUNICATION METHOD

Both the MD2400 fire detection control panel and the building management system (BMS) can initiate a UDP protocol. Every protocol that is sent must be acknowledged by the other party.

### 2.1 MD2400 FIRE DETECTION CONTROL PANEL TO BUILDING MANAGEMENT SYSTEM

The MD2400 fire detection control panel will send a UDP protocol to the building management system in the following cases:

- Heartbeat
- Any change in the status of the fire detection control panel
- Requested data from the building management system (See paragraph 2.2)

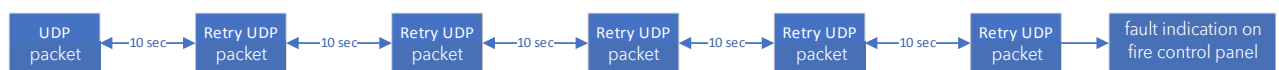


MD2400	ACKNOWLEDGE
CODE_HEARTBEAT	CODE_ACKNOWLEDGE_EXTERN
CODE_DETECTOR_EVENT	CODE_ACKNOWLEDGE_EXTERN
CODE_CENTRAL_FAULT_EVENT	CODE_ACKNOWLEDGE_EXTERN
CODE_CENTRAL_EVENT	CODE_ACKNOWLEDGE_EXTERN
CODE_CHANGE_TECHNICIAN	CODE_ACKNOWLEDGE_EXTERN
CODE_CHANGE_STATE	CODE_ACKNOWLEDGE_EXTERN
CODE_STARTUP_CENTRAL	CODE_ACKNOWLEDGE_EXTERN
CODE_REMOVE_EVENT	CODE_ACKNOWLEDGE_EXTERN
CODE_ZONE_STATUS	CODE_ACKNOWLEDGE_EXTERN
CODE_DETECTOR_VALUES	CODE_ACKNOWLEDGE_EXTERN
CODE_ABSENT	CODE_ACKNOWLEDGE_EXTERN

Each of these transmitted protocols must be acknowledged within the configured “time-out” period (default 10 seconds) by a CODE\_ACKNOWLEDGE\_EXTERN from the building management system.

If this acknowledgment is not received within the specified time, the MD2400 fire detection control panel will make several consecutive retry attempts (default 5) to resend the protocol.

If, after these attempts, no acknowledgment is received from the building management system, this will result in a fault indication on the fire detection control panel.



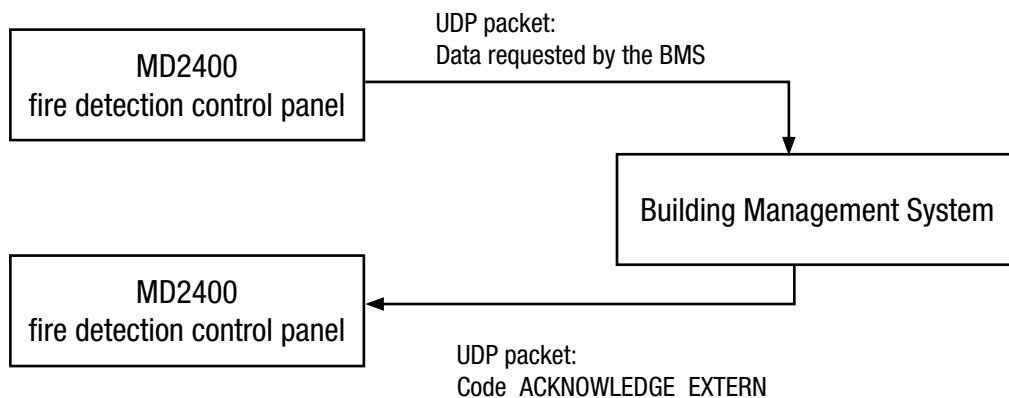
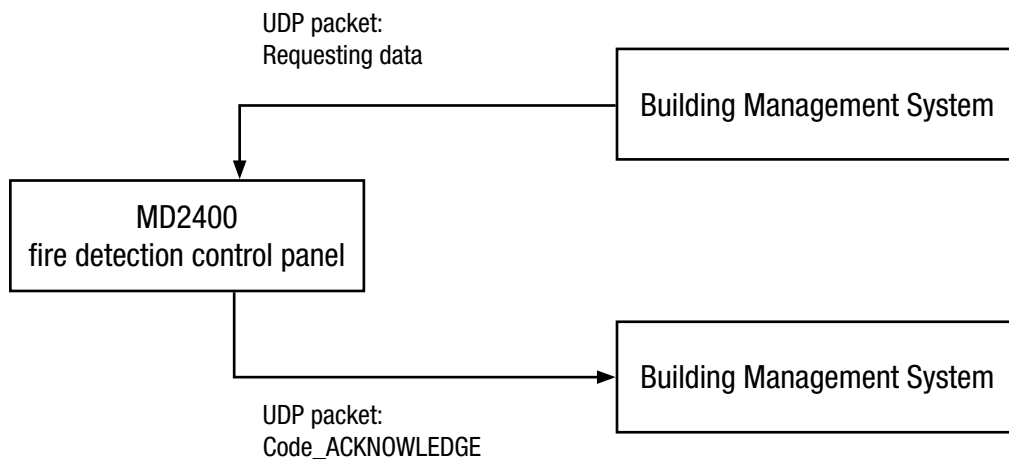
## 2.2 BUILDING MANAGEMENT SYSTEM TO MD2400 FIRE DETECTION CONTROL PANEL

The building management system can request the following information or perform the following operations on the fire detection control panel.

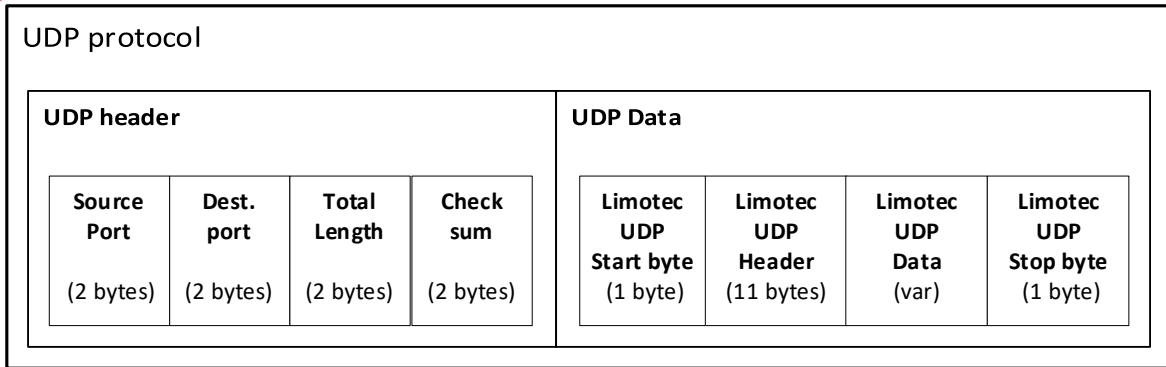
Building Management System	MD2400 fire detection control panel
CODE_STARTUP_EXTERN	CODE_ACKNOWLEDGE
CODE_SET_CHANGE_STATE	CODE_ACKNOWLEDGE
CODE_ASK_ZONE_STATUS*	CODE_ACKNOWLEDGE
CODE_ASK_DETECTOR_VALUES	CODE_ACKNOWLEDGE
CODE_DO_CENTRAL_ACTION	CODE_ACKNOWLEDGE
CODE_TEST_ALARM	CODE_ACKNOWLEDGE

Every UDP packet sent from the building management system is acknowledged by the MD2400 fire detection control panel with a CODE\_ACKNOWLEDGE.

After sending the CODE\_ACKNOWLEDGE, the MD2400 will transmit the requested information to the building management system, which must in turn acknowledge this again with a CODE\_ACKNOWLEDGE\_EXTERN.



### 2.3 STRUCTURE UDP PACKET



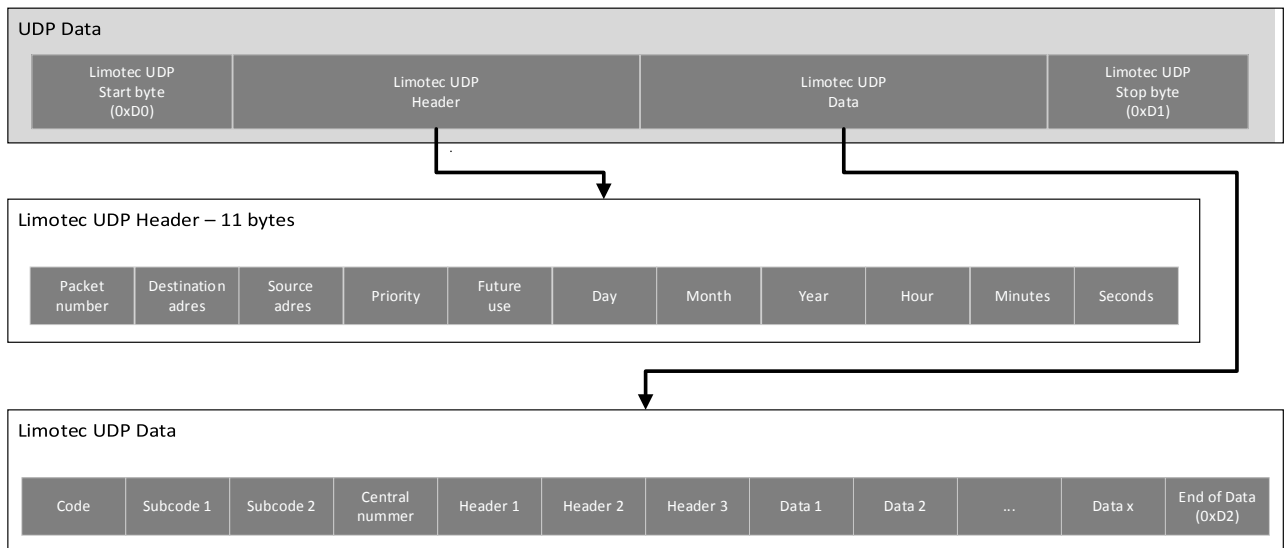
Structure of a complete UDP packet

The packets are transmitted according to the UDP protocol.

This protocol consists of a UDP header and UDP data. The document below only applies to the UDP data. The UDP header is assumed to be known.

The UDP data is composed of:

- Limotec UDP start byte 0xD0
- Limotec UDP header 11 bytes
- Limotec UDP data variable length
- Limotec UDP stop byte 0xD1

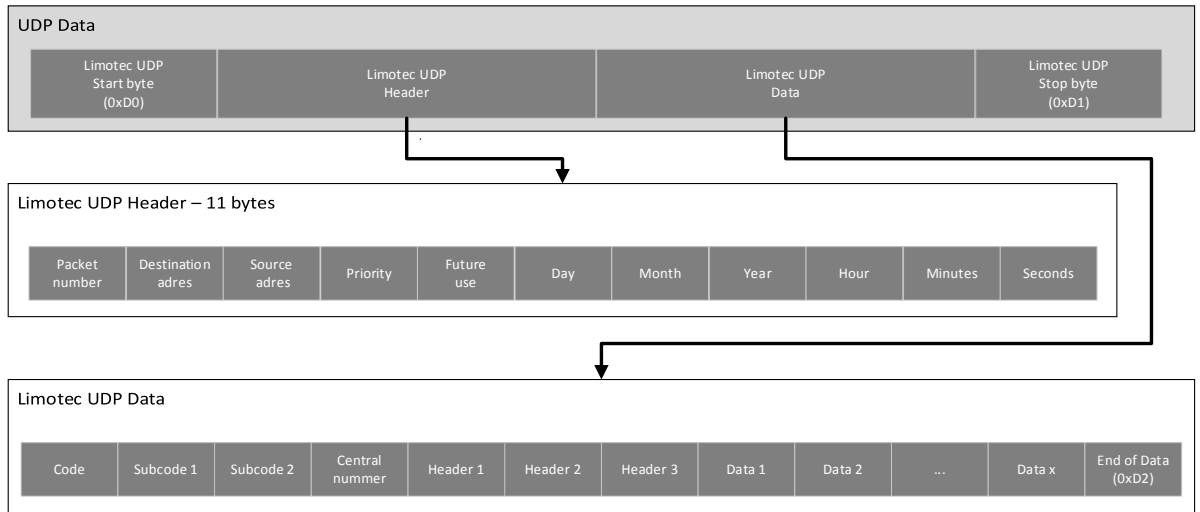


**2.3.1 LIMOTEC UDP HEADER**

Index	discription	data	format	info
1	Packet number	1 BYTE	HEX	The header always contains a “packet number” with a decimal value between 0 and 127 (0x00–0x7F). This is a counter that increments with every packet. Once the maximum value of 127 is reached, it is reset to 0.
2	Destination adres	1 BYTE	HEX	Default = 0
3	Source adres	1 BYTE	HEX	Default = 0
4	Priority	1 BYTE	HEX	<p>Priority 0x00 (0):</p> <p>CODE_HEARTBEAT = 0x00  CODE_ACK = 0x02  CODE_ABSENT = 0x04  CODE_ZONE_STATUS = 0x22  CODE_DETECTOR_VALUES = 0x24</p> <p>Priority 0x04 (4):</p> <p>CODE_CHANGE_STATE = 0x18 (of component)</p> <p>Priority 0x05 (5):</p> <p>CODE_CHANGE_TECHNICIAN = 0x16  CODE_STARTUP_CENTRAL = 0x1A  CODE_CENTRAL_EVENT = 0x14  CODE_CENTRAL_FAULT_EVENT = 0x12  CODE_DETECTOR_EVENT = 0x10  CODE_REMOVE_EVENT = 0x1C  CODE_CHANGE_STATE = 0x18</p>
5	Future use	1 BYTE	HEX	Default = 0
6	Day	1 BYTE	HEX	Each packet sent from the fire detection control panel includes a timestamp. This indicates the moment at which the packet was transmitted by the control panel.
7	Month	1 BYTE	HEX	
8	Year	1 BYTE	HEX	
9	Hour	1 BYTE	HEX	
10	Minutes	1 BYTE	HEX	
11	Seconds	1 BYTE	HEX	

**2.3.2 LIMOTEC UDP DATA**

The Limotec UDP data has a fixed structure, but the content differs per packet. This content is described in paragraph 3.2 “Detailed Packets”.



Index	discription	data	format	info
1	Code	1 BYTE	HEX	Content – paragraph 3.2 Detailed Packets.
2	Subcode 1	1 BYTE	HEX	
3	Subcode 2	1 BYTE	HEX	
4	Central number	1 BYTE	HEX	
5	Header 1	1 BYTE	HEX	
6	Header 2	2 BYTES	HEX	
7	Header 3	2 BYTES	HEX	
8 - ?	Data	Variable length	HEX of ASCII	
? + 1	End of data	1 BYTE	HEX	0xD2

### 3. UDP PROTOCOL

#### 3.1 OVERVIEW OF PACKETS

All codes supported by the protocol are listed below.

- Even codes: MD2400 fire detection control panel → Building Management System
- Oneven codes: Building Management System → MD2400 fire detection control panel

CODE_HEARTHBEAT	0x00
CODE_ACKNOWLEDGE_EXTERN	0x01
CODE_ACKNOWLEDGE	0x02
CODE_ABSENT	0x04
CODE_DETECTOR_EVENT	0x10
CODE_CENTRAL_FAULT_EVENT	0x12
CODE_CENTRAL_EVENT	0x14
CODE_CHANGE_TECHNICIAN	0x16
CODE_SET_CHANGE_STATE	0x17
CODE_CHANGE_STATE	0x18
CODE_STARTUP_CENTRAL	0x1A
CODE_STARTUP_EXTERN	0x1B
CODE_REMOVE_EVENT	0x1C
CODE_ASK_ZONE_STATUS	0x21
CODE_ZONE_STATUS	0x22
CODE_ASK_DETECTOR_VALUES	0x23
CODE_DETECTOR_VALUES	0x24
CODE_DO_CENTRAL_ACTION	0x25
CODE_TEST_ALARM	0x27

#### 3.2 PACKET DETAIL

##### 3.2.1 CODE\_HEARTBEAT (0x00)

Every minute, a packet is sent from the MD2400 fire detection control panel. This code is intended to monitor the communication between the MD2400 fire detection control panel and the building management system.

Code	0x00	CODE_HEARTBEAT	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.2 CODE\_ACKNOWLEDGE\_EXTERN (0x01)

Packets sent from the fire detection control panel must be acknowledged by the building management system with CODE\_ACKNOWLEDGE\_EXTERN.

Code	0x01	CODE_ACKNOWLEDGE_EXTERN	
Subcode 1	HEX	Number of the packet to be acknowledged	0x00 – 0x7F (0-127)
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.3 CODE\_ACKNOWLEDGE (0x02)

Packets sent from the building management system must be acknowledged by the fire detection control panel with CODE\_ACKNOWLEDGE.

Code	0x02	CODE_ACKNOWLEDGE	
Subcode 1	HEX	Number of the packet to be acknowledged	0x00 – 0x7F (0-127)
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.4 CODE\_ABSENT (0x04)

When non-existent data is requested by the building management system from the fire detection control panel, this will result in a CODE\_ABSENT.

Example: CODE\_ASK\_ZONE\_STATUS requests a loop that is not present in the fire detection control panel. Since the loop does not exist in the control panel, it will send the code below to the building management system.

Code	0x04	CODE_ABSENT	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop	0x01 – 0x10 (1-16)
Header 2	HEX	Component	0x0001 – 0x007E (1-126)
Header 3	HEX	Group	0x0001 – 0x0040 (1-64)
End of data	HEX	0xD2	

### 3.2.5 CODE\_DETECTOR\_EVENT (0x10)

Every status change of a component is immediately reported by the MD2400 fire detection control panel to the building management system.

Code	0x10	CODE_DETECTOR_EVENT	
Subcode 1	HEX	0x01 (1) Open 0x02 (2) Pre-alarm 0x03 (3) Alarm1 0x04 (4) Alarm2 0x05 (5) Short 0x06 (6) Other faults 0x0A (10) Type of fault 0x0B (11) Dual adress	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop	0x01 – 0x10 (1-16)
Header 2	HEX	Component	0x0001 – 0x007E (1-126)
Header 3	HEX	Group	0x0001 – 0x0040 (1-64)
Data 0-19	ASCII	Name control panel	20 characters
Data 20-51	ASCII	Description of component status	32 characters
Data 52-101	ASCII	Name component	50 characters
Data 102-133	ASCII	Future use	32 characters
Data 134-165	ASCII	Name group	32 characters
End of Data	HEX	0xD2	

### 3.2.6 CODE\_CENTRAL\_FAULT\_EVENT (0x12)

This code covers general faults that may occur on the fire detection control panel. These faults are not related to components and are immediately reported by the MD2400 fire detection control panel to the building management system.

Code	0x12	CODE_CENTRAL_FAULT_EVENT	
Subcode 1	HEX	0x01 (1) Mains fault 0x02 (2) System fault 0x04 (4) Peripheral fault 0x06 (6) Loop communication fault 0x07 (7) Token fault 0x09 (9) Sounder circuit fault 0x0A (10) Power supply fault 0x0B (11) Redundancy fault 0x0C (12) Printer fault 0x0D (13) ESPA fault 0x0F (15) Battery fault 0x10 (16) Future use 0x1E (30) EAM fault	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop	0x00 – 0x10 (0-16)
Header 2	HEX	Specific fault	Subcode 1 = 0x01: 0x0001 = Loop fault 0x0004 = Parameter corrupt
Header 3	HEX	0x0000	
Data 0-19	A	Name control panel	20 characters
Data 20-38	A	Description	19 characters
39-70	A	Message	32 characters
End of data	HEX	0xD2	

### 3.2.7 CODE\_CENTRAL\_EVENT (0x14)

The operation of the 'silence', 'reset', and 'evacuation' buttons on the MD2400 fire detection control panel is immediately reported by the MD2400 fire detection control panel to the building management system.

Code	0x14	CODE_CENTRAL_EVENT	
Subcode 1	HEX	0x00 (0) None 0x01 (1) Silence 0x02 (2) Reset 0x03 (3) Evacuation	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.8 CODE\_CHANGE\_TECHNICIAN (0x16)

Every status change of technician mode is immediately reported by the MD2400 fire detection control panel to the building management system.

Code	0x16	CODE_CHANGE_TECHNICIAN	
Subcode 1	HEX	0x00 (0) Technician mode not active 0x01 (1) Technician mode local 0x02 (2) Technician mode sprinkler 0x03 (3) Technician mode global	
Subcode 2	HEX	0x00	
Central number	HEX	Central	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.9 CODE\_SET\_CHANGE\_STATE (0x17)

The building management system can change the status (in/out of service, in/out of test) of a component, a group, or a loop of the MD2400 fire detection control panel.

Changing the status of a component:

- Header 1 = loop number
- Header 2 = component number
- Header 3 = 0x0000

Changing the status of a group:

- Header 1 = 0x00
- Header 2 = 0x0000
- Header 3 = group number

Changing the status of a loop:

- Header 1 = loop number
- Header 2 = 0x0000
- Header 3 = 0x0000

MD2400 fire detection control panel response: **CODE\_CHANGE\_STATE (0x18)**

Code	0x17	CODE_SET_CHANGE_STATE	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop number	0x00 – 0x10 (0-16)
Header 2	HEX	Component number	0x0000 – 0x007E (0-126)
Header 3	HEX	Group number	0x0000 – 0x0040 (0-64)
Data	HEX	0x00 (0) in service 0x01 (1) out of service 0x02 (2) in test 0x03 (3) out of test	
End of data	HEX	0xD2	

### 3.2.10 CODE\_CHANGE\_STATE (0x18)

Changing the status (in/out of service, in/out of test) of a component, a group, or a loop of the MD2400 fire detection control panel will be immediately reported by the MD2400 fire detection control panel to the building management system.

Status originating from a component:

- Header 1 = loop number
- Header 2 = component number
- Header 3 = 0x0000

Status originating from a group:

- Header 1 = 0x00
- Header 2 = 0x0000
- Header 3 = group number

Status originating from a loop:

- Header 1 = loop number
- Header 2 = 0x0000
- Header 3 = 0x0000

Code	0x18	CODE_CHANGE_STATE	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop number	0x01 – 0x10 (1-16)
Header 2	HEX	Component number	0x0000 – 0x007E (0-126)
Header 3	HEX	Group number	0x0000 – 0x0040 (0-64)
Data	HEX	0x00 (0) in service 0x01 (1) out of service 0x02 (2) in test 0x03 (3) out of test	
End of data	HEX	0xD2	

### 3.2.11 CODE\_STARTUP\_CENTRAL (0x1A)

A restart of the MD2400 fire detection control panel results in the transmission of CODE\_STARTUP\_CENTRAL to the building management system. Upon receiving this code, all notifications may be cleared. If notifications are still present on the MD2400 fire detection control panel, they will be resent to the building management system.

Code	0x1A	CODE_STARTUP_CENTRAL	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.12 CODE\_STARTUP\_EXTERN (0x1B)

During the startup of the building management system, after a communication failure, and after restarting the building management system, this code must be sent to the MD2400 fire detection control panel.

Response from the MD2400 fire detection control panel: all notifications that may still be present will be resent to the building management system.

Code	0x1B	CODE_STARTUP_EXTERN	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.13 CODE\_REMOVE\_EVENT (0x1C)

If a fault is automatically reset, the MD2400 fire detection control panel sends the CODE\_REMOVE\_EVENT to the building management system. The data in "Subcode 1" indicates which notifications may be removed.

Code	0x1C	CODE_REMOVE_EVENT	
Subcode 1	HEX	0x00 (0) Component 0x01 (1) Loop fault 0x02 (2) System fault 0x04 (4) Peripheral fault 0x06 (6) Loop communication fault 0x07 (7) Token fault 0x09 (9) Sounder circuit fault 0x0A (10) Power supply fault 0x0B (11) Redundancy fault 0x0C (12) Printer fault 0x0D (13) ESPA fault 0x0F (15) Battery fault 0x10 (16) Future use 0x1E (30) EAM fault	0x00 = notifications originating from a component 0x01 and higher = notifications originating from the control panel. These notifications are identical to CODE_CENTRAL_FAULT_EVENT
Subcode 2	HEX	0x00	
Central number	HEX	Number of the fire detection control panel	0x00 - 0x1F (0-31)
Header 1	HEX	Loop	Only applicable if subcode 1 = 0x00 0x00 – 0x1A (0-16)
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
Data	HEX	Component addresses	Only applicable if subcode 1 = 0x00 0x01 – 0x7E (1-126)
End of data	HEX	0xD2	

### 3.2.14 CODE\_ASK\_ZONE\_STATUS (0x21)

With this code, the building management system can request the status of a loop. This status indicates whether the loop is in/out of service or in/out of test.

Response from the MD2400 fire detection control panel: **CODE\_ZONE\_STATUS (0x22) - Correct data**

**CODE\_ABSENT (0x04) - Non-existent data**

Code	0x21	CODE_ASK_ZONE_STATUS	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop	0x01 – 0x10 (1-16)
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.15 CODE\_ZONE\_STATUS (0x22)

The CODE\_ZONE\_STATUS is only generated by the MD2400 fire detection control panel after receiving and acknowledging the CODE\_ASK\_ZONE\_STATUS. This response indicates whether a loop is in/out of service or in/out of test.

Code	0x22	CODE_ZONE_STATUS	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop	0x00 – 0x10 (1-16)
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
Data	HEX	Bit 0 : out of service Bit 1 : in service	
End of data	HEX	0x0D	

### 3.2.16 CODE\_ASK\_DETECTOR\_VALUES (0x23)

With this code, the building management system can request the following information from a loop:

- Analog values per component + the programmed detection threshold (see PC-config-MD2400)
- Status (in service / out of service, in test / out of test, present) of all components in a loop, including the group number in which each component is programmed.

Response from the MD2400 fire detection control panel: **CODE\_DETECTOR\_VALUES (0x24)**

**The use of the CODE\_ASK\_DETECTOR\_VALUES command must be limited (recommendation: maximum one request per hour). Requesting this information too frequently will temporarily block the MD2400 fire detection control panel.**

Code	0x23	CODE_ASK_DETECTOR_VALUES	
Subcode 1	HEX	0x00 (0) analogue values + levels 0x01 (1) status + group	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop number	0x01 – 0x10 (1-16)
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.17 CODE\_DETECTOR\_VALUES (0x24)

The CODE\_DETECTOR\_VALUES is only generated by the MD2400 fire detection control panel after receiving and acknowledging the CODE\_ASK\_DETECTOR\_VALUES. In this response, the analog values per component are returned together with the programmed detection thresholds, or the status of a component together with its programmed group number.

Code	0x24	CODE_DETECTOR_VALUES	
Subcode 1	HEX	0x00 (0) analogue values + detection levels 0x01 (1) status + group number	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop number	0x01 – 0x10 (1-16)
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
0-251	HEX	Query subcode 1 = 0x00: Byte 1 analogue value Byte 2 detection level (value 10 if not programmed)  Query subcode 1 = 0x01 Byte 1 bit 0 out of service bit 1 in test bit 2 present Byte 2 group number	2 bytes per component
End of data	HEX	0x2D	

### 3.2.18 CODE\_DO\_CENTRAL\_ACTION (0x25) – from interface version V1.1 onwards

With this code, the building management system can perform the following operations on the fire detection control panel.

This code immediately triggers the execution of the corresponding actions on the fire detection control panel. Therefore, this code must be implemented and used with caution.

- Silence OFF command:
  - Stops the audible signal on the operator panels and repeater panels.
  - Control outputs may be deactivated. Contact the fire detection system installer for more information.
- Silence ON command:
  - Stops the audible signal on the operator panels and repeater panels.
  - Control outputs and/or sounders may be deactivated. Contact the fire detection system installer for more information.
- Reset of the fire detection control panel

Code	0x25	CODE_DO_CENTRAL_ACTION	
Subcode 1	HEX	0x00 (0) Silence OFF command 0x01 (1) Silence ON command 0x02 (2) Reset	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	0x00	
Header 2	HEX	0x0000	
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

### 3.2.19 CODE\_TEST\_ALARM (0x27) – from interface version V1.1 onwards

With this code, the building management system can activate specific components of the fire detection control panel into alarm. This code immediately triggers the respective component to go into alarm on the fire detection control panel.

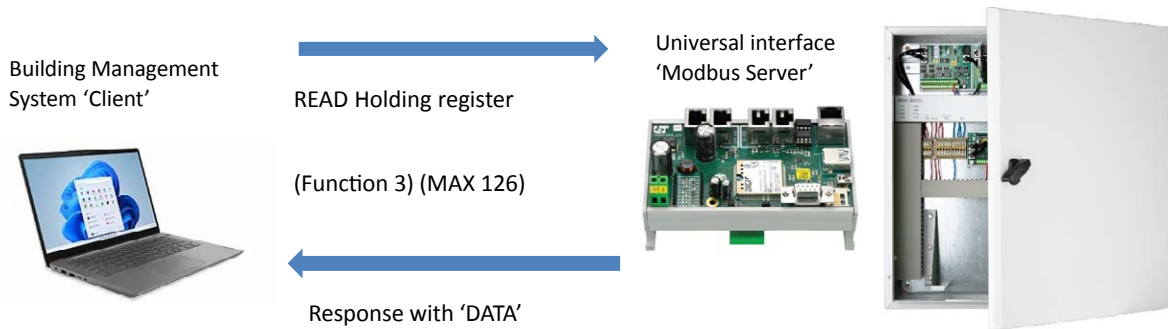
As a result, control outputs may be activated. Therefore, this code must be implemented and used with great caution. Contact the fire detection control panel installer for more information on the exact operation.

Code	0x27	CODE_TEST_ALARM	
Subcode 1	HEX	0x00	
Subcode 2	HEX	0x00	
Central number	HEX	Address of the fire detection control panel	0x00 – 0x1F (0-31)
Header 1	HEX	Loop number	0x01 – 0x10 (1-16)
Header 2	HEX	Component number	0x0000 – 0x007E (0-126)
Header 3	HEX	0x0000	
End of data	HEX	0xD2	

#### 4. MODBUS COMMUNICATION METHOD

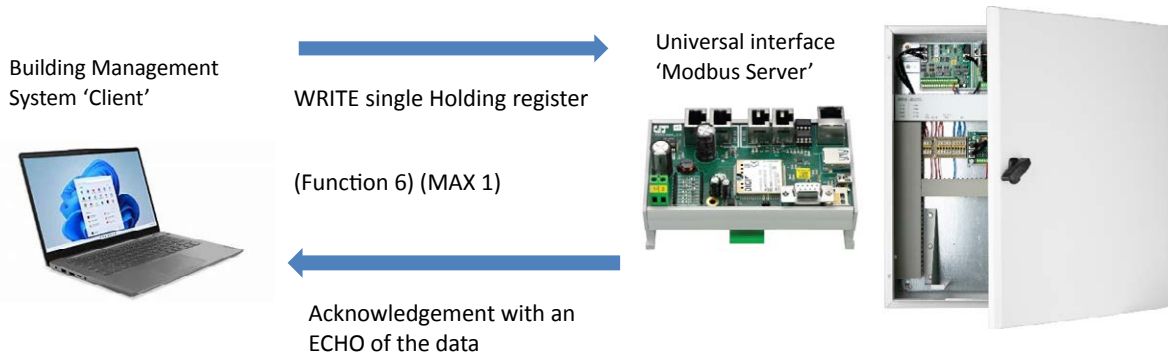
- Modbus TCP/IP is the Ethernet-based variant of the Modbus protocol. It encapsulates Modbus messages in TCP/IP packets and uses standard Ethernet infrastructure
- The MD2400 universal interface for integration with a BMS, when configured for MODBUS communication, functions as a MODBUS server
- Maximum of 2 clients can connect simultaneously
- Default communication port: 502 (adjustable by a Limotec installer)
- Slave id (Unit identifier) = 1
- Modbus TCP/IP

##### 4.1 BUILDING MANAGEMENT SYSTEM REQUESTS DATA FROM THE MD2400 FIRE DETECTION CONTROL PANEL

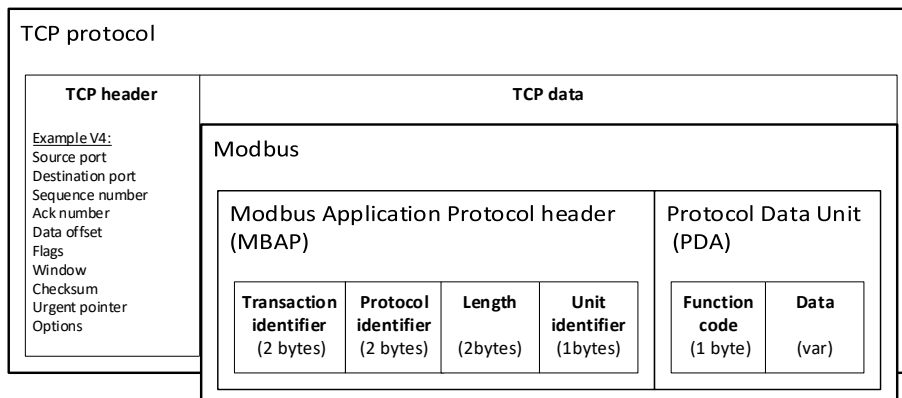


The MD2400 must be polled at an interval of no more than 10 seconds in order to report alarms in a timely manner. Failing to poll the fire detection control panel may generate a fault message on the MD2400.

##### 4.2 BUILDING MANAGEMENT SYSTEM MODIFIES DATA ON THE MD2400 FIRE DETECTION CONTROL PANEL



##### 4.3 STRUCTURE OF A MODBUS TCP PACKET



Structure of a complete MODBUS TCP packet

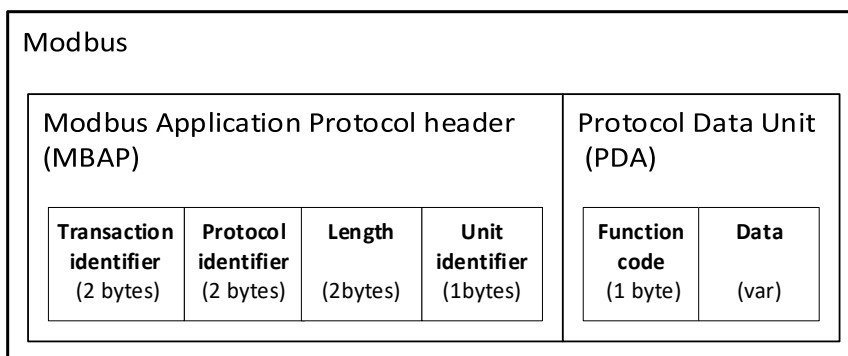
### 4.3.1 Modbus Application Protocol header

Index	discription	data	format	info
1	Transaction identifier	2 BYTE	HEX	Unique ID used to link Modbus 'requests' and Modbus 'responses'. The Modbus client sends a request to the Modbus server and places a unique code in the 'transaction identifier'. The server always answers the request using the same 'transaction identifier'.
2	Protocol identifier	2 BYTE	HEX	0x00 - Modbus TCP
3	Length	2 BYTE	HEX	Length of the remaining data (Unit Identifier + Protocol Data Unit (PDU))
4	Unit identifier	1 BYTE	HEX	0x01

### 4.3.2 Protocol Data Unit

Index	discription	data	format	info
1	Function code	1 BYTE	HEX	0x03 – read holding register (MAX 126) 0x06 – write holding register
2	Data	VAR BYTE	HEX	

### 4.3.3 Examples



#### 4.3.3.1 Reading the status of a single component of the fire detection control panel

Request from the building management system (Client):

Index	discription	data	info
1	Transaction identifier	0x00 0x05	The client generates a random number. In this example: "5"
2	Protocol identifier	0x00 0x00	Always 0
3	Length	0x00 0x06	6 data bytes follow
4	Unit identifier	0x01	Slave nummer 1
5	Function code	0x03	Read holding register
6.1	Data: Start register	0x00 0x0A	Requesting address 40010 loop 1 component 10 (Offset = 0 <sup>(1)</sup> )
6.2	Data Number of registers	0x00 0x01	Requesting a single register

(1) The offset can be configured by the installer of the fire detection control panel.  
Offset = 0 (default): Requesting address 'x' results in a response from register 'x'.  
Offset = 1: Requesting address 'x' results in a response from register 'x + 1'.

Response from the fire detection control panel (server):

Index	discription	data	info
1	Transaction identifier	0x00 0x05	The server responds with the same transaction identifier that was used by the client in the request.
2	Protocol identifier	0x00 0x00	Always 0
3	Length	0x00 0x05	5 data bytes follow
4	Unit identifier	0x01	
5	Function code	0x03	Read holding register
6.1	Data: Byte count	0x02	2 bytes data follow
6.2	Data: Requested register	0x81 0x00	Request of a single register 1001 0000 = status component 10 on loop 1 <ul style="list-style-type: none"> <li>• General fault</li> <li>• Open</li> </ul>

#### 4.3.3.2 Reading the statuses of multiple components of the fire detection control panel

Request from the building management system (Client):

Index	discription	data	info
1	Transaction identifier	0x00 0x0F	The client generates a random number. In this example: 'decimal 15'.
2	Protocol identifier	0x00 0x00	Always 0
3	Length	0x00 0x06	6 data bytes follow
4	Unit identifier	0x01	Slave nummer 1
5	Function code	0x03	Read holding register
6.1	Data: Start register	0x00 0x01	Request of address 40001 loop 1 component 1 (Offset = 0 <sup>(2)</sup> )
6.2	Data: Number of registers	0x00 0x14	Request of 20 registers (MAX 126)

Response from the fire detection control panel (server):

Index	discription	data	info
1	Transaction identifier	0x00 0x0F	The server responds with the same transaction identifier that was used by the client in the request.
2	Protocol identifier	0x00 0x00	Always 0
3	Length	0x00 0x2b	43 data bytes follow
4	Unit identifier	0x01	
5	Function code	0x03	Read holding register
6.1	Data: Byte count	0x28	40 bytes data follow
6.2	Data: 1st register	0x00 0x00	Register data 400001
6.3	Data: 2nd register	0x00 0x00	Register data 400002
6.4	Data: 3rd register	0x00 0x00	Register data 400003
6.5	Data: 4th register	0x00 0x00	Register data 400004
6.6	Data: 5th register	0x00 0x00	Register data 400005
6.7	Data: 6th register	0x00 0x00	Register data 400006
6.8	Data: 7th register	0x00 0x00	Register data 400007

(2) The offset can be configured by the installer of the fire detection control panel.  
 Offset = 0 (default): Requesting address 'x' results in a response from register 'x'.  
 Offset = 1: Requesting address 'x' results in a response from register 'x + 1'.

Index	discription	data	info
6.9	Data: 8th register	0x00 0x00	Register data 400008
6.10	Data: 9th register	0x00 0x00	Register data 400009
6.11	Data: 10th register	0x00 0x00	Register data 400010
6.12	Data: 11th register	0x00 0x00	Register data 400011
6.13	Data: 12th register	0x00 0x00	Register data 400012
6.14	Data: 13th register	0x00 0x00	Register data 400013
6.15	Data: 14th register	0x00 0x00	Register data 400014
6.16	Data: 15th register	0x00 0x00	Register data 400015
6.17	Data: 16th register	0x00 0x00	Register data 400016
6.18	Data: 17th register	0x00 0x00	Register data 400017
6.19	Data: 18th register	0x00 0x00	Register data 400018
6.20	Data: 19th register	0x00 0x00	Register data 400019
6.21	Data: 20th register	0x00 0x00	Register data 400020

#### 4.3.3.3. Modifying (Write) components of the fire detection control panel

Request from the building management system (Client):

Index	discription	data	info
1	Transaction identifier	0x00 0x12	The client generates a random number. In this example: "decimal 18"
2	Protocol identifier	0x00 0x00	Always 0
3	Length	0x00 0x06	6 data bytes follow
4	Unit identifier	0x01	Slave nummer 1
5	Function code	0x06	Write holding register
6.1	Data: Start register	0x00 0x0A	Modifying address 40010 loop 1 component 1 (Offset = 0 <sup>(3)</sup> )
6.2	Data: Value to be written	0x00 0x20	Write value 0x020 = decimal 32 → Place component out of service

Response from the fire detection control panel (server):

Index	discription	data	info
1	Transaction identifier	0x00 0x12	The server responds with the same transaction identifier that was used by the client in the request.
2	Protocol identifier	0x00 0x00	Always 0
3	Length	0x00 0x06	6 data bytes follow
4	Unit identifier	0x01	
5	Function code	0x06	Write holding register
6.1	Data: Start register	0x00 0xA0	Echo of the request : address 40010
6.2	Data: Written value	0x00 0x20	Echo of the request : 32

(3) The offset can be configured by the installer of the fire detection control panel.  
 Offset = 0 (default): Requesting address 'x' results in a response from register 'x'.  
 Offset = 1: Requesting address 'x' results in a response from register 'x + 1'.

## 5. MODBUS PROTOCOL

### 5.1 COMPONENTS OF THE MD2400 FIRE DETECTION CONTROL PANEL (REGISTERS 1–2046)

A maximum of 126 components can be connected to a single loop of the MD2400 fire detection control panel.

- Each loop is divided into a data block of 128 addresses.
- Modbus address = (loop number – 1) \* 128 + component address

#### 5.1.1 Modbus addresses

Modbus address range		Data	
(4)0001	(4)0126	Loop 1 – component 1	Loop 1 – Component 126
(4)0129	(4)0254	Loop 2 – component 1	Loop 2 – Component 126
(4)0257	(4)0382	Loop 3 – component 1	Loop 3 – Component 126
(4)0385	(4)0510	Loop 4 – component 1	Loop 4 – Component 126
(4)0513	(4)0638	Loop 5 – component 1	Loop 5 – Component 126
(4)0641	(4)0766	Loop 6 – component 1	Loop 6 – Component 126
(4)0769	(4)0894	Loop 7 – component 1	Loop 7 – Component 126
(4)0897	(4)1022	Loop 8 – component 1	Loop 8 – Component 126
(4)1025	(4)1150	Loop 9 – component 1	Loop 9 – Component 126
(4)1153	(4)1278	Loop 10 – component 1	Loop 10 – Component 126
(4)1281	(4)1406	Loop 11 – component 1	Loop 11 – Component 126
(4)1409	(4)1534	Loop 12 – component 1	Loop 12 – Component 126
(4)1537	(4)1662	Loop 13 – component 1	Loop 13 – Component 126
(4)1665	(4)1790	Loop 14 – component 1	Loop 14 – Component 126
(4)1793	(4)1918	Loop 15 – component 1	Loop 15 – Component 126
(4)1921	(4)2046	Loop 16 – component 1	Loop 16 – Component 126

#### 5.1.2 Register data

The data of a register consists of 2 bytes. Each bit corresponds to a specific status of a component.

READ ONLY
READ / WRITE

Byte	Status	Bit
Higher byte	Open	0
	Pre-alarm	1
	Alarm 1	2
	Alarm 2	3
	Short	4
	x	5
	Sprinkler alarm	6
General fault	7	
Lower byte	Technical message	0
	Fault type	1
	Double addressing	2
	x	3
	Evacuation	4
	Out of service	5
	In test	6
Wireless component fault	7	

## 5.2 GROUPS OF THE MD2400 FIRE DETECTION CONTROL PANEL (REGISTERS 4101–4164)

A maximum of 64 groups can be created in the MD2400 fire detection control panel. The groups are a collection of one or more components, spanning across different loops.

- Modbus address = 4100 + group address

### 5.2.1 Modbus addresses

Modbus address range		Data	
(4)4101	(4)4164	Group 1	Group 64

### 5.2.2 Register data

The data of a register consists of 2 bytes. Each bit corresponds to a specific status of a group.

READ ONLY
READ / WRITE

Byte	Status	Bit
Higher byte	x	0-7
Lower byte	x	0-2
	x	3
	Component fault	4
	Out of service	5
	In test	6
	Component	7

## 5.3 LOOPS OF THE MD2400 FIRE DETECTION CONTROL PANEL (REGISTERS 4301–4316)

An MD2400 fire detection control panel can contain a maximum of 16 loops. Each loop can contain between 1 and a maximum of 126 components.

- Modbus address = 4300 + loop address

### 5.3.1 Modbus addresses

Modbus address range		Data	
(4)4301	(4)4316	Loop 1	Loop 16

### 5.3.2 Register data

The data of a register consists of 2 bytes. Each bit corresponds to a specific status of a loop.

READ ONLY
READ / WRITE

Byte	Status	Bit
Higher byte	Line fault	0
	Parameter conformity fault	1
	x	2-6
	General fault	7
Lower byte	x	0-3
	Component fault	4
	Out of service	5
	In test	6
	Component	7

## 5.4 MD2400 FIRE DETECTION CONTROL PANEL IN A TOKEN-RING (REGISTERS 4451–4472)

Multiple MD2400 fire detection control panels can be interconnected in a network via a token-ring connection. A maximum of 32 fire detection control panels can be linked together. The MD2400 universal interface can display the status of another MD2400 fire detection control panel. For detailed information at component level, an MD2400 universal interface must always be installed for each individual fire detection control panel.

- Modbus address = 4450 + control panel address + 1

### 5.4.1 Modbus addresses

Modbus address range		Data	
(4)4451	(4)4472	Fire detection control panel 0	Fire detection control panel 31

### 5.4.2 Register data

The data of a register consists of 2 bytes. Each bit corresponds to a specific status of an MD2400 fire detection control panel.

READ ONLY
WRITE ONLY

Byte	Status	Bit
Higher byte	General out of service	0
	General in test	1
	Warning out of service	2
	x	3
	Sounders active	4
	x	5-6
	Reset <sup>(4)</sup>	7
Lower byte	General fire alarm	0
	Evacuation	1
	General fault	2
	System fault	3
	Power supply fault	4
	Silence	5
	Evacuation delay	6
	Sounder circuit fault	7

(4) The reset functionality will remove all notifications from the fire detection control panel and must therefore be implemented with the necessary caution and expertise.